



# **Windchill<sup>®</sup> Business Administrator's Guide**

**Windchill<sup>®</sup> 8.0**

**April 2007**

## **Copyright © 2006 Parametric Technology Corporation. All Rights Reserved.**

User and training documentation from Parametric Technology Corporation and its subsidiary companies (collectively “PTC”) are subject to the copyright laws of the United States and other countries and is provided under a license agreement that restricts copying, disclosure, and use of such documentation. PTC hereby grants to the licensed user the right to make copies in printed form of this documentation if provided on software media, but only for internal/personal use and in accordance with the license agreement under which the applicable software is licensed. Any copy made shall include the PTC copyright notice and any other proprietary notice provided by PTC. This documentation may not be disclosed, transferred, modified, or reduced to any form, including electronic media, or transmitted or made publicly available by any means without the prior written consent of PTC and no authorization is granted to make copies for such purposes.

Information described herein is furnished for general information only, is subject to change without notice, and should not be construed as a warranty or commitment by PTC. PTC assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

The software described in this document is provided under written license agreement, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. It may not be copied or distributed in any form or medium, disclosed to third parties, or used in any manner not provided for in the software licenses agreement except with written prior approval from PTC. UNAUTHORIZED USE OF SOFTWARE OR ITS DOCUMENTATION CAN RESULT IN CIVIL DAMAGES AND CRIMINAL PROSECUTION.

### **Registered Trademarks of PTC**

Advanced Surface Design, Advent, Arbortext, Behavioral Modeling, CADDs, Calculation Management Suite, Computervision, CounterPart, Create • Collaborate • Control, Designate, EPD, EPD.Connect, Epic Editor, Expert Machinist, Flexible Engineering, GRANITE, HARNESSDESIGN, Info\*Engine, InPart, Mathcad, Mathsoft, MECHANICA, Optegra, Parametric Technology, Parametric Technology Corporation, PartSpeak, PHOTORENDER, Pro/DESKTOP, Pro/E, Pro/ENGINEER, Pro/HELP, Pro/INTRALINK, Pro/MECHANICA, Pro/TOOLKIT, Product First, Product Development Means Business, Product Makes the Company, PTC, the PTC logo, PT/Products, Shaping Innovation, Simple • Powerful • Connected, StudyWorks, The Way to Product First, Wildfire, Windchill, Windchill DynamicDesignLink, and Windchill PDMLink.

### **Trademarks of PTC**

3B2, 3DPAINT, Arbortext Editor, Arbortext Content Manager, Arbortext Contributor, Arbortext Companion for Word®, Arbortext Advanced Print Publisher – Desktop, Arbortext Advanced Print Publisher – Enterprise, Arbortext Publishing Engine, Arbortext Dynamic Link Manager, Arbortext Styler, Arbortext Architect, Arbortext Digital Media Publisher, Associative Topology Bus, AutobuildZ, CDRS, CV, CVact, CVacc, CVdesign, CV DORS, CVMAC, CVNC, CVToolmaker, Create • Collaborate • Control • Communicate, EDACompare, EDAConduit, DataDoctor, DesignSuite, DIMENSION III, Distributed Services Manager, DIVISION, e/ENGINEER, eNC Explorer, Expert Framework, Expert MoldBase, Expert Toolmaker, FlexPDM, FlexPLM, Harmony, InterComm, InterComm Expert, InterComm EDACompare, InterComm EDAConduit, ISSM, KDiP, Knowledge Discipline in Practice, Knowledge System Driver, ModelCHECK, MoldShop, NC Builder, PDS Workbench, POLYCAPP, Pro/ANIMATE, Pro/ASSEMBLY, Pro/CABLING, Pro/CASTING, Pro/CDT, Pro/CMM, Pro/COLLABORATE, Pro/COMPOSITE, Pro/CONCEPT, Pro/CONVERT, Pro/DATA for PDGS, Pro/DESIGNER, Pro/DETAIL, Pro/DIAGRAM, Pro/DIEFACE, Pro/DRAW, Pro/ECAD, Pro/ENGINE, Pro/FEATURE, Pro/FEM POST, Pro/FICIENCY, Pro/FLY THROUGH, Pro/HARNESS, Pro/INTERFACE, Pro/LANGUAGE, Pro/LEGACY, Pro/LIBRARYACCESS, Pro/MESH, Pro/Model.View, Pro/MOLDESIGN, Pro/NC ADVANCED, Pro/NC CHECK, Pro/NC MILL, Pro/NC POST, Pro/NC SHEETMETAL, Pro/NC TURN, Pro/NC WEDM, Pro/NC Wire EDM, Pro/NETWORK ANIMATOR, Pro/NOTEBOOK, Pro/PDM, Pro/PHOTORENDER, Pro/PIPING, Pro/PLASTIC ADVISOR, Pro/PLOT, Pro/POWER DESIGN, Pro/PROCESS, Pro/REPORT, Pro/REVIEW, Pro/SCAN TOOLS, Pro/SHEETMETAL, Pro/SURFACE, Pro/VERIFY, Pro/Web.Link, Pro/Web.Publish, Pro/WELDING, ProductView, PTC Precision, Routed Systems Designer, Shrinkwrap, The Product Development Company, Validation Manager, Warp, Windchill MPMLink, Windchill PartsLink, Windchill ProjectLink, and Windchill SupplyLink.

### **Patents of Parametric Technology Corporation or a Subsidiary**

Registration numbers and issue dates follow. Additionally, equivalent patents may be issued or pending outside of the United States. Contact PTC for further information. GB2366639B 13-October-2004. GB2363208 25-August-2004. (EP/DE/GB)0812447 26-May-2004. GB2365567 10-March-2004. (GB)2388003B 21-January-2004. 6,665,569 B1 16-December-2003. GB2353115 10-December-2003. 6,625,607 B1 23-September-2003. 6,580,428 B1 17-June-2003. GB2354684B 02-July-2003. GB2384125 15-October-2003. GB2354096 12-November-2003. GB2354924 24-September-2003. 6,608,623 B1 19-August-2003. GB2353376 05-November-2003. GB2354686 15-October-2003. 6,545,671 B1 08-April-2003. GB2354685B 18-June-2003. GB2354683B 04-June-2003. 6,608,623 B1 19-August-2003. 6,473,673 B1 29-October-2002. GB2354683B 04-June-2003. 6,447,223 B1 10-Sept-2002. 6,308,144 23-October-2001. 5,680,523 21-October-1997. 5,838,331 17-November-1998. 4,956,771 11-September-1990. 5,058,000 15-October-1991. 5,140,321 18-August-1992. 5,423,023 05-June-1990. 4,310,615 21-December-1998. 4,310,614 30-April-1996. 4,310,614 22-April-1999. 5,297,053 22-March-1994. 5,513,316 30-April-1996. 5,689,711 18-November-1997. 5,506,950 09-April-1996. 5,428,772 27-June-1995. 5,850,535 15-December-1998. 5,557,176 09-November-1996. 5,561,747 01-October-1996. (EP)0240557 02-October-1986. 6,275,866 14-Aug-2001. 5,469,538 21-Nov-1995. 5,526,475 11-June-1996. 5,771,392 23-June-1998. 5,844,555 01-Dec-1998. PCT 03/05061 13-Feb-2003.

### **Third-Party Trademarks**

Adobe, Acrobat, Distiller, FrameMaker and the Acrobat logo are trademarks of Adobe Systems Incorporated. Cognos is a registered trademark of Cognos Corporation. IBM, AIX, and WebSphere are registered trademarks of IBM Corporation. Allegro, Cadence, and Concept are registered trademarks of Cadence Design Systems, Inc. Apple, Mac, Mac OS, Panther, and Tiger are trademarks or registered trademarks of Apple Computer, Inc. AutoCAD and Autodesk Inventor are registered trademarks of Autodesk, Inc. Baan is a registered trademark of Baan Company. CADAM and CATIA are registered trademarks of Dassault Systemes. DataDirect Connect is a registered trademark of DataDirect Technologies. CYA, iArchive, HOTbackup, and Virtual StandBy are trademarks or registered trademarks of CYA Technologies, Inc. DOORS is a registered trademark of Telelogic AB. Feature-Following Anti-Aliasing is a trademark of LightWorks Design. FLEXnet, InstallShield, and InstallAnywhere are trademarks or registered trademarks of Macrovision Corporation. Geomagic is a registered trademark of Raindrop Geomagic, Inc. EVERSINC, GROOVE, GROOVEFEST, GROOVE.NET, GROOVE NETWORKS, iGROOVE, PEERWARE, and the interlocking circles logo are trademarks of Groove Networks, Inc. Helix is a trademark of Microcadam, Inc. HOOPS is a trademark of Tech Soft America, Inc. HP, Hewlett-Packard, and HP-UX are registered trademarks of Hewlett-Packard Company. Advanced ClusterProven, ClusterProven, the ClusterProven design, Rational Rose, and Rational ClearCase are trademarks or registered trademarks of International Business Machines in the United States and other countries and are used under license. IBM Corporation does not warrant and is not responsible for the operation of this software product. I-DEAS, Metaphase, Parasolid, SHERPA, Solid Edge, TeamCenter, UG NX, and Unigraphics are trademarks or registered trademarks of UGS Corp. Intel is a registered trademark of Intel Corporation. IRIX is a registered trademark of Silicon Graphics, Inc. I Run and ISOGEN are registered trademarks of Alias Ltd. Linux is a registered trademark of Linus Torvalds. MainWin and Mainsoft are trademarks of Mainsoft Corporation. MatrixOne is a trademark of MatrixOne, Inc. Mentor Graphics and Board Station are registered trademarks and 3D Design, AMPLE, and Design Manager are trademarks of Mentor Graphics Corporation. MEDUSA and STHENO are trademarks of CAD Schroer GmbH. Microsoft, ActiveX, Excel, JScript, Windows, Windows NT, Windows 2000, Windows 2000 Server, Windows XP, Windows Server 2003, the Windows logo, Visual Basic, the Visual Basic logo, and Active Accessibility are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Moldflow is a registered trademark of Moldflow Corporation. Mozilla and Firefox are registered trademarks of the Mozilla Foundation. Netscape, Netscape Navigator, Netscape Communicator, and the Netscape N and Ship's Wheel logos are registered trademarks or service marks of Netscape Communications Corporation in the U.S. and other countries. Oracle and interMedia are registered trademarks of Oracle Corporation. OrbixWeb is a registered trademark of IONA Technologies PLC. PANTONE is a registered trademark and PANTONE CALIBRATED is a trademark of Pantone, Inc. PDGS is a registered trademark of Ford Motor Company. RAND is a trademark of RAND Worldwide. RetrievalWare is a registered trademark of Convera Corporation. RosettaNet is a trademark and Partner Interface Process and PIP are registered trademarks of RosettaNet, a nonprofit organization. SAP and R/3 are registered trademarks of SAP AG Germany. SolidWorks is a registered trademark of SolidWorks Corporation. SPARC is a registered trademark and SPARCStation is a trademark of SPARC International, Inc. (Products bearing the SPARC trademarks are based on an architecture developed by Sun Microsystems, Inc. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and in other countries. Sun, Sun Microsystems, the Sun logo, Solaris, UltraSPARC, Java and all Java based marks, and "The Network is the Computer" are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and in other countries. 3Dconnexion is a registered trademark of Logitech International S.A. TIBCO is a registered trademark and TIBCO ActiveEnterprise, TIBCO Designer, TIBCO Enterprise Message Service, TIBCO Rendezvous, TIBCO TurboXML, and TIBCO BusinessWorks are trademarks or registered trademarks of TIBCO Software Inc. in the United States and other countries. WebEx is a trademark of WebEx Communications, Inc. API Toolkit is a trademark of InterCAP Graphics Systems, Inc. BEA and WebLogic are registered trademarks of BEA Systems, Inc. BEA WebLogic Server and BEA WebLogic Platform are trademarks of BEA Systems, Inc. Documentum is a registered trademark of EMC Corporation. Elan License Manager and Softlock are trademarks of Rainbow Technologies, Inc. JAWS is a registered trademark of Freedom Scientific BLV Group, LLC in the United States and other countries. FileNET is a registered trademark of FileNET Corporation. Panagon is a trademark of FileNET Corporation. Galaxy Application Environment is a licensed trademark of Visix Software, Inc. Interleaf is a trademark of Interleaf, Inc. IslandDraw and IslandPaint are trademarks of Island Graphics Corporation. OSF/Motif and Motif are trademarks of the Open Software Foundation, Inc. Palm Computing, Palm OS, Graffiti, HotSync, and Palm Modem are registered trademarks, and Palm III, Palm IIIe, Palm IIIx, Palm V, Palm Vx, Palm VII, Palm, More connected, Simply Palm, the Palm Computing platform logo, all Palm logos, and HotSync logo are trademarks of Palm, Inc. or its subsidiaries. Proximity and Linguibase are registered trademarks of Proximity Technology, Inc. TeX is a trademark of the American Mathematical Society. UNIX is a registered trademark of The Open Group. X Window System is a trademark of X Consortium, Inc.

### **Third-Party Technology Information**

Certain PTC software products contain licensed third-party technology:

Rational Rose and Rational ClearCase are copyrighted software of IBM Corp.

RetrievalWare is copyrighted software of Convera Corporation.

VisTools library is copyrighted software of Visual Kinematics, Inc. (VKI) containing confidential trade secret information belonging to VKI.

HOOPS graphics system is a proprietary software product of, and is copyrighted by, Tech Soft America, Inc.

I Run and ISOGEN are copyrighted software of Alias Ltd.

Xdriver is copyrighted software of 3Dconnexion, Inc, a Logitech International S.A. company.

G POST is copyrighted software and a registered trademark of Intercim.

VERICUT is copyrighted software and a registered trademark of CGTech.

FLEXnet Publisher is copyrighted software of Macrovision Corporation.

Pro/PLASTIC ADVISOR is powered by Moldflow technology.

Fatigue Advisor nCode libraries from nCode International.

TetMesh GHS3D provided by Simulog Technologies, a business unit of Simulog S.A.

MainWin Dedicated Libraries are copyrighted software of Mainsoft Corporation.

DFORMD.DLL is copyrighted software from Compaq Computer Corporation and may not be distributed.

LightWork Libraries are copyrighted by LightWork Design 1990–2001.

Visual Basic for Applications and Internet Explorer is copyrighted software of Microsoft Corporation.

Parasolid is © UGS Corp.

TECHNOMATIX is copyrighted software and contains proprietary information of Technomatix Technologies Ltd.

TIBCO ActiveEnterprise, TIBCO Designer, TIBCO Enterprise Message Service, TIBCO Rendezvous, TIBCO TurboXML, and TIBCO BusinessWorks are provided by TIBCO Software Inc.

DataDirect Connect is copyrighted software of DataDirect Technologies.

Technology "Powered by Groove" is provided by Groove Networks, Inc.

Technology "Powered by WebEx" is provided by WebEx Communications, Inc.

Oracle 8i run time, Oracle 9i run time, and Oracle 10g run time are Copyright 2002–2004 Oracle Corporation. Oracle programs provided herein are subject to a restricted use license and can only be used in conjunction with the PTC software they are provided with.

Adobe Acrobat Reader and Adobe Distiller are copyrighted software of Adobe Systems Inc. and are subject to the Adobe End User License Agreement as provided by Adobe with those products.

Certain license management is based on Elan License Manager © 1989-1999 Rainbow Technologies, Inc. All rights reserved.

Portions compiled from Microsoft Developer Network Redistributable Sample Code, Copyright © 1998 by Microsoft Corporation.

The CD-ROM Composer and CD-ROM Consumer are based on Vivace CD-Web Composer Integrator © 1996-1997 KnowledgeSet Corporation. All rights reserved.

Larson CGM Engine 9.4, Copyright © 1992-2006 Larson Software Technology, Inc. All rights reserved.

Certain graphics-handling portions are based on the following technologies:

GIF: Copyright 1989, 1990 Kirk L. Johnson. The author disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the author be liable for any special, indirect, or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence, or other tortious action, arising out of or in connection with the use or performance of this software.

JPEG: This software is based in part on the work of the Independent JPEG Group.

PNG: Copyright 2004-2006 Glenn Randers-Pehrson.

TIFF: Copyright 1988-1997 Sam Leffler, Copyright © 1991-1997 Silicon Graphics, Inc. The software is provided AS IS and without warranty of any kind, express, implied, or otherwise, including without limitation, any warranty of merchantability or fitness for a particular purpose. In no event shall Sam Leffler or Silicon Graphics be liable for any special, incidental, indirect, or consequential damages of any kind, or any damages whatsoever resulting from loss of use, data or profits, whether or not advised of the possibility of damage, or on any theory of liability, arising out of or in connection with the use or performance of this software.

XBM, Sun Raster, and Sun Icon: Copyright, 1987, Massachusetts Institute of Technology.

ZLIB: Copyright 1995-2004 Jean-loup Gailly and Mark Adler.

PDFlib software is copyright © 1997-2005 PDFlib GmbH. All rights reserved.

PStill software is copyright © Dipl.- Ing. Frank Siebert, 1996-2005

Proximity Linguistic Technology provides Spelling Check/Thesaurus portions of certain software products, including: The Proximity/Bertelsmann Lexikon Verlag Database. Copyright © 1997 Bertelsmann Lexikon Verlag. Copyright © 1997, All Rights Reserved, Proximity Technology, Inc.; The Proximity/C.A. Strombertg AB Database. Copyright © 1989 C.A. Strombertg AB. Copyright © 1989, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Editions Fernand Nathan Database. Copyright © 1984 Editions Fernand Nathan. Copyright © 1989, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Espasa-Calpe Database. Copyright © 1990 Espasa-Calpe. Copyright © 1990, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Dr. Lluís de Yzaguirre i Maura Database. Copyright © 1991 Dr. Lluís de Yzaguirre i Maura Copyright © 1991, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Franklin Electronic Publishers, Inc. Database. Copyright © 1994 Franklin Electronic Publishers, Inc. Copyright © 1994, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Hachette Database. Copyright © 1992 Hachette. Copyright © 1992, All Rights Reserved, Proximity Technology, Inc.; The Proximity/IDE a.s. Database. Copyright © 1989, 1990 IDE a.s. Copyright © 1989, 1990, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Merriam-Webster, Inc. Database. Copyright © 1984, 1990 Merriam-Webster, Inc. Copyright © 1984, 1990, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Merriam-Webster, Inc./Franklin Electronic Publishers, Inc. Database. Copyright © 1990 Merriam-Webster Inc. Copyright © 1994 Franklin Electronic Publishers, Inc. Copyright © 1994, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Munksgaard International Publishers Ltd. Database. Copyright © 1990 Munksgaard International Publishers Ltd. Copyright © 1990, All Rights Reserved, Proximity Technology, Inc.; The Proximity/S. Fischer Verlag Database. Copyright © 1983 S. Fischer Verlag. Copyright © 1997, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Van Dale Lexicografie by Database. Copyright © 1995, 1997 Van Dale Lexicografie by. Copyright © 1996, 1997, All Rights Reserved, Proximity Technology, Inc.; The Proximity/William Collins Sons & Co. Ltd. Database. Copyright © 1984, 1990 William Collins Sons & Co. Ltd. Copyright © 1988, 1990, All Rights Reserved, Proximity Technology, Inc.; The Proximity/Zanichelli Database. Copyright © 1989 Zanichelli. Copyright © 1989, All Rights Reserved, Proximity Technology, Inc.

CimPro, IGES/Pro, and PS/Pro software are provided by CAD/CAM-E, Inc.

The Arbortext Import/Export feature includes components that are licensed and copyrighted by CambridgeDocs LLC (© 2002-2005 CambridgeDocs LLC). This functionality:

Includes software developed by the Apache Software Foundation (<http://www.apache.org>).

Redistributes JRE from Sun Microsystems. The Redistributable is complete and unmodified, and only bundled as part of the product. CambridgeDocs is not distributing additional software intended to supersede any component(s) of the Redistributable, nor has CambridgeDocs removed or altered any proprietary legends or notices contained in or on the Redistributable. CambridgeDocs is only distributing the Redistributable pursuant to a license agreement that protects Sun's interests consistent with the terms contained in the Agreement. CambridgeDocs agrees to defend and indemnify Sun and its licensors from and against any damages, costs, liabilities, settlement amounts and/or expenses (including attorney's fees) incurred in connection with any claim, lawsuit, or action by any third party that arises or results from the use or distribution of any and all Programs and/or Software. This product includes code licensed from RSA Security, Inc. ICU4J portions licensed from IBM (see terms at <http://www-128.ibm.com/developerworks/opensource/>).

Redistributes the Saxon XSLT Processor from Michael Kay, more information, including source code is available at <http://saxon.sourceforge.net/>.

Uses cxImage, an open source image conversion library that follows the zlib license. cxImage further uses the following image libraries which also ship (statically linked) with cxLib: zLib, LibTIFF, LibPNG, LibJPEG, JBIG-Kit, JasPer, LibJ2K. See <http://www.xdp.it/cximage.htm>.

Includes software developed by Andy Clark, namely Neko DTD. NekoDTD is © Copyright 2002, 2003, Andy Clark. All rights reserved. For more information, visit <http://www.apache.org/~andyc/neko/doc/index.html>.

Includes code which was developed and copyright by Steven John Metsker, and shipped with Building Parsers with Java, from Addison Wesley.

Uses controls from Infragistics NetAdvantage 2004, Volume 3, © Copyright 2004 Infragistics.

Word, FrameMaker, and Interleaf filters. Copyright © 2000 Blueberry Software. All rights reserved.

Portions of software documentation are used with the permission of the World Wide Web Consortium. Copyright © 1994–2006 World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal>. Such portions are indicated at their points of use.

Copyright and ownership of certain software components is with YARD SOFTWARE SYSTEMS LIMITED, unauthorized use and copying of which is hereby prohibited. YARD SOFTWARE SYSTEMS LIMITED 1987. (Lic. #YSS:SC:9107001)

Certain business intelligence reporting functionality is powered by Cognos.

MKM (Mathsoft Kernel Maple) © 1994 Waterloo Maple Software.

Microsoft Internet Explorer © 1995-2005 Microsoft Corporation.

Web Help © 2004 and Macromedia RoboHelp are copyrighted software of Adobe Systems Incorporated.

Portions of the Mathcad Solver © 1990-2002 by Frontline Systems, Inc.

Sentry Spelling-Checker Engine copyright © 1994-2003 Wintertree Software, Inc.

File Filters © 1986-2002 Circle Systems, Inc.

PDEFIT © 1995-2002 Dr. Klaus Schittkowski.

Hyphenation Copyright © 1986-1999, Computer Hyphenation Ltd. All rights reserved.

\*\*\*\*\*

METIS, developed by George Karypis and Vipin Kumar at the University of Minnesota, can be researched at <http://www.cs.umn.edu/~karypis/metis>. METIS is © 1997 Regents of the University of Minnesota.

Certain software components licensed in connection with the Apache Software Foundation, all rights reserved, and use is subject to the terms and limitations (and license agreement) at <http://www.apache.org>. Apache software is provided by its Contributors AS IS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, and any expressed or implied warranties, including, but not limited to, the implied warranties of title non-infringement, merchantability and fitness for a particular purpose are disclaimed. In no event shall the Apache Software Foundation or its Contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage. Apache software includes:

Apache Server, Tomcat, Xalan, Xerces, and Jakarta, Jakarta POI, Jakarta Regular Expression, Commons-FileUpload, and XML Beans

IBM XML Parser for Java Edition, the IBM SaxParser and the IBM Lotus XSL Edition

DITA-OT - Apache License Version

Jakarta-ORO (as used with Jena Software)

NekoHTML software developed by Andy Clark © Copyright 2002-2005, Andy Clark. All rights reserved.

Lucene (<http://lucene.apache.org>)

IzPack: Java-based Software Installers Generator (<http://www.izforge.com/izpack/start>)

Pop up calendar components Copyright © 1998 Netscape Communications Corporation. All Rights Reserved.

The following software is distributed under GNU Lesser General Public License (LGPL) which is at <http://www.gnu.org/copyleft/lesser.html> and is provided AS IS by authors with no warranty therefrom without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE (see the GNU LGPL for more details). Upon request PTC will provide the source code for such software for a charge no more than the cost of performing this distribution:

GTK+ - The GIMP Toolkit is licensed under the GNU Library General Public License (LGPL). You may obtain a copy of the source code at <http://www.gtk.org/>, which is likewise provided under the GNU LGPL.

OmniORB is distributed under the terms and conditions of the GNU General Public License – The OmniORB Libraries are released under the GNU LGPL.

Java Port copyright 1998 by Aaron M. Renn ([arenn@urbanophile.com](mailto:arenn@urbanophile.com)), is redistributed under the GNU LGPL. You may obtain a copy of the source code at <http://www.urbanophile.com/arenn/hacking/download.html>. The source code is likewise provided under the GNU LGPL.

JFreeChart is licensed under the GNU LGPL and can be found at <http://www.jfree.org>.

eXist, an Open Source Native XML Database, is redistributed under the GNU LGPL. You may obtain a copy of the source code at <http://exist.sourceforge.net/index.html>. The source code is likewise provided under the GNU LGPL.

UnZip (© 1990 2001 Info-ZIP, All Rights Reserved) is provided AS IS and WITHOUT WARRANTY OF ANY KIND. For the complete Info-ZIP license see <http://www.info-zip.org/doc/LICENSE>. "Info-ZIP" is defined as the following set of individuals: Mark Adler, John Bush, Karl Davis, Harald Denker, Jean-Michel Dubois, Jean-loup Gailly, Hunter Goatley, Ian Gorman, Chris Herborth, Dirk Haase, Greg Hartwig, Robert Heath, Jonathan Hudson, Paul Kienitz, David Kirschbaum, Johnny Lee, Onno van der Linden, Igor Mandrichenko, Steve P. Miller, Sergio Monesi, Keith Owens, George Petrov, Greg Roelofs, Kai Uwe Rommel, Steve Salisbury, Dave Smith, Christian Spieler, Antoine Verheijen, Paul von Behren, Rich Wales, and Mike White.

The Java™ Telnet Applet (StatusPeer.java, TelnetIO.java, TelnetWrapper.java, TimedOutException.java), Copyright © 1996, 97 Mattias L. Jügel, Marcus Meißner, is redistributed under the GNU General Public License. This license is from the original copyright holder and the Applet is provided WITHOUT WARRANTY OF ANY KIND. You may obtain a copy of the source code for the Applet at <http://www.mud.de/se/jta> (for a charge of no more than the cost of physically performing the source distribution), by sending e mail to [leo@mud.de](mailto:leo@mud.de) or [marcus@mud.de](mailto:marcus@mud.de)—you are allowed to choose either distribution method. Said source code is likewise provided under the GNU General Public License.

Eclipse SWT is distributed under the Eclipse Public License (EPL) (<http://www.eclipse.org/org/documents/epl-v10.php>) and is provided AS IS by authors with no warranty therefrom and any provisions which differ from the EPL are offered by PTC. Upon request PTC will provide the source code for such software for a charge no more than the cost of performing this distribution.

PDFBOX – Free software provided pursuant to the BSD license at <http://www.pdfbox.org/index.html> - Features.

zlib software Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

#ZipLib GNU software is developed for the Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA, copyright © 1989, 1991. PTC hereby disclaims all copyright interest in the program #ZipLib written by Mike Krueger. #ZipLib licensed free of charge and there is no warranty for the program, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holders and/or other parties provide the program AS IS without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the program is with you. Should the program prove defective, you assume the cost of all necessary servicing, repair, or correction.

The Java Getopt.jar file, copyright 1987 1997 Free Software Foundation, Inc.

CUP Parser Generator Copyright ©1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian—used by permission. The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages, or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action arising out of or in connection with the use or performance of this software.

Software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>): Copyright © 1998 2003 The OpenSSL Project. All rights reserved. This product may include cryptographic software written by Eric Young ([ey@cryptsoft.com](mailto:ey@cryptsoft.com)).

ImageMagick software is Copyright © 1999-2005 ImageMagick Studio LLC, a nonprofit organization dedicated to making software imaging solutions freely available. ImageMagick is freely available without charge and provided pursuant to the following license agreement: <http://www.imagemagick.org/script/license.php>.

Mozilla Japanese localization components are subject to the Netscape Public License Version 1.1 (at <http://www.mozilla.org/NPL>). Software distributed under the Netscape Public License (NPL) is distributed on an AS IS basis, WITHOUT WARRANTY OF ANY KIND, either expressed or implied (see the NPL for the rights and limitations that are governing different languages). The Original Code is Mozilla Communicator client code, released March 31, 1998 and the Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright © 1998 Netscape Communications Corporation. All Rights Reserved. Contributors: Kazu Yamamoto ([kazu@mozilla.gr.jp](mailto:kazu@mozilla.gr.jp)), Ryoichi Furukawa ([furu@mozilla.gr.jp](mailto:furu@mozilla.gr.jp)), Tsukasa Maruyama ([mal@mozilla.gr.jp](mailto:mal@mozilla.gr.jp)), Teiji Matsuba ([matsuba@dream.com](mailto:matsuba@dream.com)).

The following components are subject to the Mozilla Public License Version 1.0 or 1.1 at <http://www.mozilla.org/MPL> (the MPL): Gecko and Mozilla components and Charset Detector. Software distributed under the MPL is distributed on an AS IS basis, WITHOUT WARRANTY OF ANY KIND, either expressed or implied and all warranty, support, indemnity or liability obligations under PTC's software license agreements are provided by PTC (see the MPL for the specific language governing rights and limitations). Modifications to Mozilla source code are available under the MPL and are available upon request.

iText Library - Copyright © 1999-2006 by Bruno Lowagie and Paulo Soares. All Rights Reserved – source code and further information available at <http://www.lowagie.com/iText>.

iCal4j is Copyright © 2005, Ben Fortuna, All rights reserved. Redistribution and use of iCal4j in source and binary forms, with or without modification, are permitted provided that the following conditions are met: (i) Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer; (ii) Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution; and (iii) Neither the name of Ben Fortuna nor the names of any other contributors may be used to endorse or promote products derived from this software without specific prior written permission. iCal4j SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Independent JPEG Group's JPEG software. This software is Copyright © 1991-1998, Thomas G. Lane. All Rights Reserved. This software is based in part on the work of the Independent JPEG Group.

libpng, Copyright © 2004 Glenn Randers-Pehrson, which is distributed according to the disclaimer and license (as well as the list of Contributing Authors) at <http://www.libpng.org/pub/png/src/libpng-LICENSE.txt>.

ICU (International Components for Unicode) Copyright 1995-2001 International Business Machines Corporation and others. All rights reserved.

Perl support was developed with the aid of Perl Kit, Version 5.0. Copyright © 1989-2002, Larry Wall. All rights reserved.

Regular Expressions support was derived from copyrighted software written by Henry Spencer, Copyright © 1986 by University of Toronto.

SGML parser: Copyright © 1994, 1995, 1996, 1997, 1998 James Clark, 1999 Matthias Clasen.

XML parser and XSLT processing was developed using Libxml and Libxslt by Daniel Veillard, Copyright © 2001.

Curl software, Copyright ©1996 - 2005, Daniel Stenberg, <[daniel@haxx.se](mailto:daniel@haxx.se)>. All rights reserved. Software is used under the following permissions: Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use, or other dealings.

The cad2eda program utilizes wxWidgets (formerly wxWindows) libraries for its cross-platform UI API, which is licensed under the wxWindows Library License at <http://www.wxwindows.org>.

LAPACK libraries used are freely available at <http://www.netlib.org/> (authors are Anderson, E. and Bai, Z. and Bischof, C. and Blackford, S. and Demmel, J. and Dongarra, J. and Du Croz, J. and Greenbaum, A. and Hammarling, S. and McKenney, A. and Sorensen, D.).

The following software, which is provided with and called by certain PTC software products, is licensed under the GNU General Public License (<http://www.gnu.org/licenses/gpl.txt>) and is provided AS IS by the authors with no warranty therefrom without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE (see the GNU GPL for more details). Upon request PTC will provide the source code for such software for a charge no more than the cost of performing this distribution:

Ghost Script (<http://www.cs.wisc.edu/~ghost>);

The PJA (Pure Java AWT) Toolkit library (<http://www.eteks.com/pja/en>).

Launch4j: This program is free software licensed under the GPL license, the head subproject (the code which is attached to the wrapped jars) is licensed under the LGPL license. Launch4j may be used for wrapping closed source, commercial applications.

The following unmodified libraries distributed under the GNU-GPL: libstdc and #ziplib (each are provided pursuant to an exception that permits use of the library in proprietary applications with no restrictions provided that the library is not modified).

Rhino JavaScript engine, distributed with a form of the Mozilla Public License (MPL).



Java Advanced Imaging (JAI) is provided pursuant to the Sun Java Distribution License (JDL) at <http://www.jai.dev.java.net>. The terms of the JDL shall supersede any other licensing terms for PTC software with respect to JAI components.

May include Jena Software © Copyright 2000, 2001, 2002, 2003, 2004, 2005 Hewlett-Packard Development Company, LP. THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. Jena includes:

Jakarta–ORO software developed by the Apache Software Foundation (described above).

ICU4J software Copyright © 1995-2003 International Business Machines Corporation and others All rights reserved. Software is used under the following permissions: Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

May contain script.aculo.us (built on "prototype.conio.net"). Copyright © 2005 Thomas Fuchs (<http://script.aculo.us>, <http://mir.aculo.us>). Software is used under the following permissions: Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England. This software is based in part on the work of the Independent JPEG Group.

antlr-2.7.0.zip - ANTLR parser and lexer generator, version 2.7.0; Provided pursuant to: ANTLR 2 License <http://www.antlr.org/license.html>.

jpeg-6b.zip - JPEG image compression library, version 6.2. Used to create images for HTML output; Provided pursuant to: <http://www.faqs.org/faqs/jpeg-faq/part2>.

lpng120.zip - PNG image library version 1.2.0. <http://www.ijg.org/>; Provided pursuant to: <http://www.libpng.org/pub/png/src/libpng-LICENSE.txt>.

pcre-4.3-2-src.zip - Perl Compatible Regular Expression Library version 4.3. <http://www.pcre.org/>; Provided pursuant to: PCRE License

tiff-v3.4-tar.gz - Libtiff File IO Library version 3.4: (see also <http://www.libtiff.org/> <ftp://ftp.sgi.com/graphics/tiff>) Used by the image EFI library; Provided pursuant to: <http://www.libtiff.org/misc.html>.

zlib-1.2.1.tar.gz - Zip compression library version 1.2.1. <http://www.gzip.org/zlib/>; Provided pursuant to: Zlib.h License.

The Boost Library - Misc. C++ software from <http://www.boost.org/>; Provided pursuant to: Boost Software License [http://www.boost.org/more/license\\_info.html](http://www.boost.org/more/license_info.html) and [http://www.boost.org/LICENSE\\_1\\_0.txt](http://www.boost.org/LICENSE_1_0.txt).

**UNITED STATES GOVERNMENT RESTRICTED RIGHTS LEGEND**

This document and the software described herein are Commercial Computer Documentation and Software, pursuant to FAR 12.212(a)-(b) (OCT'95) or DFARS 227.7202-1(a) and 227.7202-3(a) (JUN'95), and are provided to the US Government under a limited commercial license only. For procurements predating the above clauses, use, duplication, or disclosure by the Government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software Clause at DFARS 252.227 7013 (OCT'88) or Commercial Computer Software-Restricted Rights at FAR 52.227 19(c)(1)-(2) (JUN'87), as applicable. 071906

**Parametric Technology Corporation, 140 Kendrick Street, Needham, MA 02494 USA**

# Contents

<b>Change Record .....</b>	<b>xxvii</b>
<b>About This Guide.....</b>	<b>xlvi</b>
Intended Audience.....	xlvi
Overview.....	xlvi
Chapter Contents .....	xlvi
Related Documentation .....	xlvi
Technical Support.....	xlvi
Documentation for PTC Products.....	xlvi
Comments .....	xlix
Documentation Conventions .....	xlix
<b>Getting Started.....</b>	<b>1-1</b>
Introduction.....	1-2
Regarding Arbortext Content Manager .....	1-2
Regarding Pro/INTRALINK 8.0 .....	1-3
Logging On as the Administrator .....	1-3
Windchill Foundation & PDM Home Page .....	1-4
Windchill PDMLink Home Page .....	1-5
Windchill ProjectLink Home Page.....	1-7
Creating the Default Organization Container.....	1-8
Establishing Administrators .....	1-12
Windchill Foundation & PDM Administrators .....	1-12
Windchill PDMLink Administrators .....	1-13
Windchill ProjectLink Administrators.....	1-14
Creating Users to Select as Administrators .....	1-15
The Next Steps.....	1-16
<b>Administration Overview .....</b>	<b>2-1</b>
Your Installed Windchill Runtime Architecture.....	2-2
Managing Your Windchill System.....	2-3
Your Installed Windchill Environment .....	2-3
Managing User Access to Data .....	2-6
Windchill PDMLink Container Hierarchy .....	2-7
Windchill ProjectLink Container Hierarchy.....	2-7

Container Hierarchy for Integral Windchill Solutions .....	2-8
Managing Access to Data through Access Control Rules .....	2-8
Managing Access to Data through Team Memberships .....	2-14
Managing Users .....	2-14
Managing Data .....	2-16
Data Types .....	2-17
Soft Types .....	2-17
Visualization Data .....	2-18
CAD Data .....	2-18
Dynamic Document Data .....	2-19
Document Data .....	2-19
Part Data .....	2-20
Audit Reports .....	2-20
Managing Windchill Processes .....	2-20
Managing User Collaboration .....	2-21
Additional Administrative Groups .....	2-21
Post-Installation Activities .....	2-22
About the windchill Command .....	2-23
About the windchill shell .....	2-25
About the xconfmanager Utility .....	2-26
Formatting Property Value Guidelines .....	2-28
<b>Administering Containers .....</b>	<b>3-1</b>
Distributed and Hierarchical Administration .....	3-2
Container Administrative Items .....	3-4
Container Configuration .....	3-4
Container Structure .....	3-6
Container Participation .....	3-8
Container Policies .....	3-10
Container Data Types and Attributes .....	3-14
Container Templates .....	3-16
Container Object Initialization Rules .....	3-19
Creating Containers .....	3-21
Removing Templates .....	3-21
Using Out-of-the-box Context Templates .....	3-22
Creating Context Templates .....	3-23
Creating Business XML Files .....	3-28
Administering Domains and Policies .....	3-28
Container and Domain Hierarchy Overview .....	3-29
Domains in the Site Container .....	3-32

Creating Domains .....	3-33
Defining Domain-based Policies .....	3-34
Using the Policy Administrator .....	3-34
Specifying Policy Rules in a Context Template .....	3-37
Assigning Domains to Folders in Windchill PDMLink or Windchill Foundation & PDM .....	3-38
Administering Object Initialization Rules .....	3-40
Loading Object Initialization Rules .....	3-40
Adding Object Initialization Rules .....	3-41
Accessing the Object Initialization Rules Administrator .....	3-41
How Object Initialization Rules Work .....	3-41
Determining the Composite Rule .....	3-42
Specifying Rules in the Object Initialization Rules Administrator .....	3-44
Content of Object Initialization Rules XML Documents .....	3-45
Example XML Document .....	3-47
Specifying Object Initialization Rules in a Context Template .....	3-47
Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects ..	3-48
Out-of-the-box Numbering Schemes .....	3-48
Additional Information about Numbering .....	3-49
Administering the Versioning of Parts, Documents, and CAD Documents .....	3-50
Understanding the Use of Versioning Schemes .....	3-50
Initial Versioning Rules .....	3-51
Harvard Series Versioning Schemes .....	3-52
Setting Up a File-based or State-based Versioning Scheme .....	3-59
Updating Versioning Schemes to Conform to the ASME Y 14.35M Standard .....	3-62
Searching for Principals .....	3-63
<b>Administering the Site .....</b>	<b>4-1</b>
Overview .....	4-2
Typical Duties of Site Administrators .....	4-2
Creating and Managing Organizations .....	4-3
Adding and Updating Members .....	4-4
Creating and Managing Site Folders and Documents .....	4-5
Changing Default Configuration Options for Windchill PDMLink and Windchill ProjectLink ..	4-5
Managing Site-level Types and Type-specific Attributes .....	4-7
Managing Site-level Templates .....	4-7
Auditing System Information .....	4-8
Creating and Managing Access Control Policies .....	4-8
Configuring External Vaults or Replication Sites to Optimize Performance .....	4-8
Configuring Numbering and Versioning Schemes and Units of Measure .....	4-9
Configuring and Managing CAD Publishing Utilities .....	4-10

Creating, Updating, and Managing Custom Reports.....	4-10
Importing and Exporting Information Among Systems .....	4-10
Managing Overall System Configuration and Preferences .....	4-10
Monitoring Enterprise Systems Transactions Log.....	4-11
Purging, Archiving, and Restoring Jobs .....	4-11
Managing Searches .....	4-11
Business Object Uniqueness Considerations .....	4-11
Out-of-the-Box Site Container Configuration .....	4-15
About the Site Tab .....	4-16
Best Practices .....	4-16
For Windchill PDMLink.....	4-16
For Windchill ProjectLink.....	4-18
<b>Administering Organizations.....</b>	<b>5-1</b>
Overview .....	5-2
Typical Duties of Organization Administrators .....	5-3
Managing Organization Members, Groups, and Roles .....	5-3
Creating, Updating, and Managing Organization Folders and Documents .....	5-4
Managing Organization-level Types and Type-specific Attributes .....	5-4
Managing Organization Templates and Object Creation Rules .....	5-5
Auditing Activities Within the Organization.....	5-5
Creating and Managing Access Control Policies .....	5-5
Configuring Numbering and Versioning Schemes .....	5-6
Monitoring and Managing Viewable Publishing.....	5-6
Viewing Project Reports .....	5-6
Importing and Exporting Information .....	5-6
Purging, Archiving, and Restoring Jobs .....	5-7
Out-of-the-Box Organization Templates .....	5-7
Container Structure .....	5-7
Container Participation.....	5-8
Container Access Control Policies .....	5-8
Container Data .....	5-13
Creating an Organization .....	5-13
Using the Organization Utilities Page .....	5-15
Changing an Established Internet Domain .....	5-16
Using the OrganizationSync Utility for User Organization Changes .....	5-16
Best Practices .....	5-18
For Both Windchill PDMLink and Windchill ProjectLink .....	5-18
For Windchill PDMLink.....	5-19
For Windchill ProjectLink.....	5-20

<b>Administering Products and Libraries .....</b>	<b>6-1</b>
Overview.....	6-2
Typical Duties of Product and Library Administrators.....	6-2
Managing Team Members and Roles.....	6-3
Managing Folders .....	6-4
Managing Templates .....	6-4
Managing Object Initialization Rules.....	6-4
Managing Access Policies .....	6-5
Configuring Numbering and Versioning Schemes.....	6-5
Managing the Life Cycle of Parts, Documents, CAD Documents, and Dynamic Documents....	6-5
Managing Viewable Publishing.....	6-6
Managing Custom Reports .....	6-6
Managing Related Reports .....	6-7
Importing and Exporting Information.....	6-7
Configuring External Vaults or Replication Sites to Optimize Performance.....	6-7
Out-of-the-box Product and Library Context Templates.....	6-8
Product Design Template .....	6-10
General Product and General Library Templates .....	6-31
Updating Access Control Rules .....	6-40
Creating a Product.....	6-40
Creating a Library .....	6-41
Administering Teams.....	6-42
Creating Custom Change Monitor Reports .....	6-44
Using the Product and Library Utilities Page.....	6-45
Configuring Product or Library Containers for Dynamic Documents.....	6-45
Configuring Part and Document Relationships.....	6-47
Part to Document Relationships .....	6-47
Part to Part Relationships .....	6-50
Document to Document Relationships .....	6-51
Allowing URLs as Content for Documents .....	6-52
Best Practices For Object Initialization Rules .....	6-52
<b>Administering Projects .....</b>	<b>7-1</b>
Overview.....	7-2
Typical Duties of Project Managers.....	7-2
Creating and Updating Project Space .....	7-2
Managing Project Team Members and Roles .....	7-3
Creating, Updating, and Managing Documents and Folders.....	7-4
Creating, Updating, and Managing Activities, Deliverables, Resources, and Action Items ..	7-5

Limiting Update Privileges for All Action Items .....	7-5
Managing Project Templates.....	7-5
Importing and Exporting Information .....	7-5
Optional Utilities (Optional Access Through Project Utilities Link) .....	7-6
Out-of-the-box Project Templates .....	7-7
Creating a Project .....	7-10
<b>Administering Windchill PDM Library.....</b>	<b>8-1</b>
Overview .....	8-2
Typical Duties of Windchill Foundation & PDM Administrators.....	8-2
Managing Groups and Roles.....	8-3
Managing Templates.....	8-3
Managing Object Initialization Rules .....	8-3
Managing Access Policies.....	8-3
Configuring Numbering and Versioning Schemes .....	8-4
Managing Viewable Publishing .....	8-4
Managing Custom Reports.....	8-4
Importing and Exporting Information .....	8-4
Configuring External Vaults or Replication Sites to Optimize Performance .....	8-5
Out-of-the-box Windchill Foundation & PDM Library Contents.....	8-5
Creating Groups for Use in Site Domain Policies .....	8-6
Improving Performance on Searches from Windchill Explorer .....	8-7
<b>Administering Principals.....</b>	<b>9-1</b>
Overview .....	9-2
Windchill Users.....	9-3
Windchill Groups .....	9-3
Windchill Organizations.....	9-4
Using the Principal Administrator .....	9-4
Best Practices for Windchill PDMLink and Windchill ProjectLink .....	9-7
Best Practices When Maintaining a Directory Service Outside of Windchill .....	9-7
Understanding Principals .....	9-7
Best Practices for Assigning Domains to Principals .....	9-8
Searching for Users and Groups .....	9-9
Managing Users.....	9-9
Changing User Passwords.....	9-10
Naming a User's Personal Cabinet .....	9-11
Deleting Users.....	9-12
Changing the Organization to which a User Belongs.....	9-13
Refreshing Team Membership for Users and Groups.....	9-14
Managing Groups .....	9-14



Working with Groups that are Maintained in a Directory Service.....	9-15
Deleting Groups .....	9-16
Managing Organizations.....	9-18
Deleting Organizations .....	9-19
Receiving Administrative Notifications.....	9-20
Managing the Principal Cache.....	9-20
Automatically Purging Entries from the Principal Cache .....	9-21
Manually Purging Entries from the Principal Cache .....	9-21
Maintaining the Connections between Principal Objects and their Directory Service Entries ..	9-22
<b>Administering Access Control .....</b>	<b>10-1</b>
Overview.....	10-2
About Access Control Rules .....	10-2
Creating and Managing Access Control Rules .....	10-4
Setting Permissions .....	10-4
About Access Control Lists.....	10-6
Deriving ACLs from Access Control Policies .....	10-7
How ACLs Work .....	10-8
Distributed Administration of Policy Rules .....	10-11
About Predefined Access Control Policy Rules.....	10-12
Managing Access to Enterprise Information.....	10-13
Domain Administered Information .....	10-14
Policy and Ad Hoc Access Controlled Information .....	10-14
Content Holder Information.....	10-15
Foldered Information.....	10-15
Life-Cycle Managed Information .....	10-19
Example Permissions Needed for Moving a Document .....	10-21
Windchill Foundation & PDM Example Permissions Needed for Creating a Part in a Shared Cabinet .....	10-22
Access Control Strategies for Cabinets in Windchill Foundation & PDM .....	10-23
Access to Cabinets .....	10-23
Access Control Strategies for Life-Cycle Managed Objects.....	10-25
Combining Access Control Strategies for Cabinets and Life-Cycle Managed Objects .....	10-25
<b>Using Types and the Type Manager .....</b>	<b>11-1</b>
Overview of Types and the Runtime Typing Capability .....	11-2
Effects of Deploying a New Type.....	11-3
Using Typing in Conjunction with Classification .....	11-4
When to Use Typing and When to Use Modeling .....	11-5
Localizing New Type Definitions .....	11-6
Migrating Existing Type Instances to a New Type Definition .....	11-7

Defining Additional Attributes .....	11-7
The Type Manager.....	11-10
Starting the Type Manager.....	11-10
Using the Type Manager .....	11-11
Setting Constraints .....	11-21
Specifying Windchill Types .....	11-24
Managing Types and Attributes for Parts, Documents, and Change Objects.....	11-25
Displaying Type Lists .....	11-25
Excluding Types from a Types List .....	11-27
Understanding Soft Attribute Behavior on Object Create and Update Windows, and on Information Pages .....	11-28
Managing Types and Attributes for CAD Documents .....	11-29
Renaming the CAD Document Soft Type.....	11-29
Additional Soft Types Related to CAD Documents .....	11-30
Managing Types and Attributes for Dynamic Documents.....	11-31
Renaming Dynamic Document and Burst Configuration Soft Types.....	11-31
Creating Additional Soft Types for Use with Arbortext Editor and the Arbortext Publishing Engine.....	11-32
Specifying the Logical Identifier for Types Used with Arbortext .....	11-33
Defining Windchill Soft Attributes for Attributes Used with Arbortext .....	11-33
Best Practices .....	11-34
For Windchill PDMLink and Windchill ProjectLink.....	11-34
For CAD Documents .....	11-37
For Searchable Types .....	11-39
<b>Administering Indexing .....</b>	<b>12-1</b>
Overview .....	12-2
About Indexing Rules.....	12-2
Creating and Managing Indexing Rules.....	12-3
About Indexing Policy .....	12-3
Considerations for Establishing Indexing Rules .....	12-4
About Indexing Lists.....	12-4
Defining a Collection .....	12-5
<b>Administering Notifications .....</b>	<b>13-1</b>
Overview .....	13-2
About Notification Rules.....	13-2
Creating and Managing Notification Rules.....	13-3
Notification Lists .....	13-4
Folder Subscription Notifications .....	13-5
About Notification .....	13-6

<b>Administering Life Cycles .....</b>	<b>14-1</b>
Overview.....	14-2
The Windchill Life Cycle Model .....	14-2
Windchill Solutions.....	14-3
Windchill Foundation & PDM .....	14-3
Additional Life Cycle Capabilities.....	14-3
Managing Life Cycle Processes .....	14-4
Promotion Process .....	14-9
Basic and Advanced Life Cycles .....	14-14
Out-of-the-Box Life Cycle Templates .....	14-15
Windchill Foundation & PDM .....	14-15
Windchill PDMLink.....	14-15
Windchill ProjectLink.....	14-18
Life Cycle Iteration.....	14-18
Testing an Updated Life Cycle.....	14-19
Viewing Iteration History .....	14-19
Creating a Life Cycle Template .....	14-19
Overriding and Reassigning Life Cycle and Team Templates .....	14-20
Preference for Manual Selection of Life Cycle and Team Templates.....	14-21
Preference for Reassigning Life Cycles and Resetting Team Templates.....	14-22
Life Cycle Properties .....	14-22
Defining Life Cycle Phases.....	14-24
State-based Revision Sequences by Life Cycle State.....	14-29
Defining Transitions .....	14-30
Additional Advanced Life Cycle Management Options .....	14-32
Predefined Life Cycle States .....	14-42
Import and Export .....	14-42
Preparing to Import or Export Life Cycles .....	14-42
Importing and Exporting Life Cycles .....	14-43
Access to Life Cycle Administration .....	14-45
Best Practices for Windchill PDMLink and Windchill ProjectLink .....	14-45
Life Cycle Support in Windchill ProjectLink .....	14-46
Best Practices for Managing Promotion Processes in Windchill PDMLink.....	14-47
Restrictions on Moving Items Between Containers .....	14-49

<b>Administering Teams and Roles .....</b>	<b>15-1</b>
Overview .....	15-2
Out-of-the-Box Team Templates .....	15-3
Windchill PDMLink Out-of-the-Box Team Templates .....	15-3
Windchill Foundation & PDM Out-of-the-Box Team Templates .....	15-5
Team Templates and Object Types .....	15-7
Team Roles Resolution .....	15-10
Initial Team Creation .....	15-10
Role Resolution Rules .....	15-10
Role Resolution Example .....	15-13
Defining Team Properties and Roles .....	15-17
Predefined Roles .....	15-19
Assigning Participants to Team Roles .....	15-20
Best Practices for Windchill PDMLink and Windchill ProjectLink .....	15-21
Refreshing Groups .....	15-22
<b>Administering Workflow Processes .....</b>	<b>16-1</b>
Overview .....	16-2
Workflow Iteration .....	16-3
Testing an Updated Workflow Process Template .....	16-3
Using the Workflow Process Editor .....	16-3
Working with Workflow Templates .....	16-4
Navigating a Process Diagram .....	16-8
Placing Process Nodes .....	16-8
Declaring Variables .....	16-12
Defining an Assigned Activity .....	16-14
Defining a Subprocess .....	16-26
Defining Connectors .....	16-27
Defining Links .....	16-28
Import and Export .....	16-30
Preparing to Import or Export Workflow Templates .....	16-30
Importing and Exporting Workflow Templates .....	16-31
Process Manager Toolbar Access Control .....	16-32
Viewing Workflow History .....	16-32
Selecting Events .....	16-33
Using the Workflow History Viewer .....	16-34
Configuring Worklist Fields .....	16-38
Workflow Instance States .....	16-40
Domain-based Workflows .....	16-41
Default Workflow Templates .....	16-42

Windchill PDMLink Default Workflows .....	16-43
Workflow Template Execution Flags .....	16-48
Process Flags .....	16-49
Activity Flags.....	16-53
Both Process and Activity Flags .....	16-55
Modifying Execution Flags .....	16-56
Electronic Signatures.....	16-58
Setting Up for Electronic Signatures .....	16-58
Requiring Electronic Signatures in a Workflow .....	16-60
Best Practices.....	16-62
Access Control.....	16-62
Using a Single Workflow in a Life Cycle Having Multiple State .....	16-63
For Windchill PDMLink and Windchill ProjectLink .....	16-63
Workflow Process Support in Windchill ProjectLink .....	16-64
<b>Administering Views and View Associations .....</b>	<b>17-1</b>
Overview.....	17-2
Views and View Associations .....	17-2
Managing Views and View Associations .....	17-3
Examining a View Structure.....	17-3
Creating a View .....	17-3
Renaming a View.....	17-3
Deleting a View .....	17-4
Creating a View Association .....	17-4
Inserting a View .....	17-4
<b>Administering Visualization Services.....</b>	<b>18-1</b>
Overview.....	18-2
File Types .....	18-2
Architecture .....	18-3
Troubleshooting.....	18-6
Windchill Visualization Service Loader .....	18-6
CAD Agent.....	18-9
Publishing CAD Documents .....	18-16
Visualization Collaboration .....	18-21
Exporting Watermarks for ProductView.....	18-22
Copy Forward Functionality .....	18-23
Copying Representations Forward .....	18-23
Copying Markups Forward.....	18-23
Copy Forward Properties .....	18-24
Windchill Visualization Service Properties.....	18-26

<b>Administering Audit Reports .....</b>	<b>19-1</b>
Overview .....	19-2
Enabling Auditing .....	19-2
Accessing Audit Reports .....	19-3
Example Audit Report .....	19-4
<b>Creating Business XML Files for Templates .....</b>	<b>A-1</b>
Overview .....	A-2
Creating Business XML Files for Organization Templates .....	A-2
OrgStructure Element.....	A-3
ContainerTemplate Element.....	A-3
loadXMLFile Element .....	A-4
TypeBasedRule Element.....	A-5
ProjectEnumType Element.....	A-5
DiscussionForumTemplate Element .....	A-5
NotebookTemplate Element.....	A-5
Creating Business XML Files for Product Templates .....	A-6
OrgStructure Element.....	A-7
DiscussionForumTemplate Element .....	A-7
NotebookTemplate Element.....	A-7
ExportedRoleMemberMap Element .....	A-7
ACLRule Element.....	A-7
TypeBasedRule Element.....	A-7
Creating Business XML Files for Library Templates .....	A-8
OrgStructure Element.....	A-9
DiscussionForumTemplate Element .....	A-9
NotebookTemplate Element.....	A-9
ExportedRoleMemberMap Element .....	A-9
ACLRule Element.....	A-9
TypeBasedRule Element.....	A-9
Creating Business XML Files for Project Templates.....	A-10
ExportedRoleMemberMap Element .....	A-11
OrgStructure Element.....	A-11
TypeBasedRule Element.....	A-11
ProductStructure Element .....	A-11
shareMapSet Element.....	A-20
projectPlan Element .....	A-21
ProjectEnumType Element.....	A-42
DiscussionForumTemplate Element .....	A-42
NotebookTemplate Element.....	A-43

Common XML Elements Used in Multiple Business XML Files .....	A-45
OrgStructure Element .....	A-45
loadXMLFile Element.....	A-52
TypeBasedRule Element .....	A-53
ExportedRoleMemberMap Element.....	A-56
ACLRule Element .....	A-59
ProjectEnumType Element .....	A-61

## Index





# Change Record

The following tables describe the major changes made in the guide for each update.

**Table 1 Changes for Windchill 8.0, Maintenance Release M040**

Chapter	Description
Chapter 3, <a href="#">Administering Containers</a>	<ul style="list-style-type: none"><li>Updated the <a href="#">File-based Scheme</a> section to include versioning name rules.</li><li>Updated the <a href="#">Creating Domains</a> section to include forward slash in domain names information.</li></ul>

Chapter	Description
Chapter 4, <a href="#">Administering the Site</a>	<ul style="list-style-type: none"> <li>Updated the <a href="#">Business Object Uniqueness Considerations</a> section.</li> <li>Updated the <a href="#">Managing Searches</a> section with latest version searching information, and reference to instructions for adding new searchable types.</li> <li>Added caution to the <a href="#">Creating and Managing Organizations</a> section.</li> <li>Added a new <a href="#">Setting Delimiters for Subscription Name Field</a> section to <a href="#">Best Practices For Windchill PDMLink</a>.</li> <li>Added a new <a href="#">Overriding and Reassigning Life Cycle and Team Templates</a> section to <a href="#">Best Practices For Windchill PDMLink</a>.</li> </ul>
Chapter 5, <a href="#">Administering Organizations</a>	<ul style="list-style-type: none"> <li>Updated the <a href="#">Using the OrganizationSync Utility for User Organization Changes</a> section.</li> </ul>
Chapter 6, <a href="#">Administering Products and Libraries</a>	<ul style="list-style-type: none"> <li>Updated the <a href="#">Revised or Saved Part to Related Document</a> section of <a href="#">Part to Document Relationships</a> to clarify that removing copy rules does not affect revise actions.</li> <li>Added new <a href="#">Managing Related Reports</a> section.</li> </ul>

Chapter	Description
Chapter 9, <a href="#">Administering Principals</a>	<ul style="list-style-type: none"> <li>Updated the <a href="#">Changing the Organization to which a User Belongs</a> section.</li> <li>Added a new <a href="#">Working with Groups that are Maintained in a Directory Service</a> section to <a href="#">Managing Groups</a>.</li> <li>Added a new section, <a href="#">Refreshing Team Membership for Users and Groups</a>.</li> </ul>
Chapter 11, <a href="#">Using Types and the Type Manager</a>	<ul style="list-style-type: none"> <li>Added statement about making attributes searchable to the <a href="#">Adding Attributes</a> section.</li> <li>Updated the following sections to support the use of the logical identifier when specifying types in Windhchill and Arbortext: <ul style="list-style-type: none"> <li><a href="#">Specifying Windchill Types</a></li> <li><a href="#">Managing Types and Attributes for Parts, Documents, and Change Objects</a></li> <li><a href="#">Managing Types and Attributes for CAD Documents</a></li> <li><a href="#">Managing Types and Attributes for Dynamic Documents</a></li> </ul> </li> </ul>

Chapter	Description
Chapter 14, <a href="#">Administering Life Cycles</a>	<ul style="list-style-type: none"> <li>Added property file for displaying annotations during promotion approval to the <a href="#">Promotion Process</a> section.</li> <li>Changed the description of requirements for Roles in the phase properties table on page 14-29 in the <a href="#">Defining Life Cycle Phases</a> section.</li> <li>Added preferences for manually selecting and reassigning life cycle and resetting team templates in the <a href="#">Overriding and Reassigning Life Cycle and Team Templates</a> section.</li> </ul>
Chapter 17, <a href="#">Administering Visualization Services</a>	<ul style="list-style-type: none"> <li>Updated <a href="#">Windchill Visualization Service Properties</a> section with new mark out of date properties.</li> <li>Added new <a href="#">Copy Forward Functionality</a> section.</li> </ul>

**Table 2 Changes for Windchill 8.0, Maintenance Release M030**

Chapter	Description
Chapter 1, <a href="#">Getting Started</a>	Added new section, <a href="#">Regarding Arbortext Content Manager</a> , with information on Arbortext Content Manager support.

Chapter	Description
Chapter 3, <a href="#">Administering Containers</a>	<p>Updated the <a href="#">Administering Object Initialization Rules</a> section:</p> <ul style="list-style-type: none"> <li>• Windchill ProjectLink does not use the default values set for folder path.</li> <li>• Customization is not supported for rule algorithms.</li> <li>• Specify types in the objType attribute using the logical identifier of the type.</li> </ul> <p>Updated the <a href="#">Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects</a> section.</p> <ul style="list-style-type: none"> <li>• Do not use numeric values other than 0 for the pad character.</li> </ul> <p>Updated the <a href="#">Administering the Versioning of Parts, Documents, and CAD Documents</a> section.</p> <ul style="list-style-type: none"> <li>• Do not remove series or series values that have been used.</li> <li>• Revision labels must be unique within the entire versioning scheme.</li> <li>• Using versioning schemes that conform to the ASME Y 14.35M revision standard is described in the <a href="#">Updating Versioning Schemes to Conform to the ASME Y 14.35M Standard</a> section.</li> </ul>
Chapter 4, <a href="#">Administering the Site</a>	Added new section, <a href="#">Business Object Uniqueness Considerations</a> .

Chapter	Description
Chapter 5, <a href="#">Administering Organizations</a>	<ul style="list-style-type: none"> <li>Added new section, <a href="#">Changing an Established Internet Domain</a>.</li> <li>Added multiorganization environment information to <a href="#">Creating Only One Organization Container for Windchill PDMLink</a> in the <a href="#">Best Practices</a> section.</li> </ul>
Chapter 7, <a href="#">Administering Projects</a>	<ul style="list-style-type: none"> <li>Added <a href="#">Hiding the Guests Role</a> section to <a href="#">Managing Project Team Members and Roles</a>.</li> <li>Added <a href="#">Limiting Update Privileges for All Action Items</a> section to <a href="#">Typical Duties of Project Managers</a>.</li> </ul>
Chapter 8, <a href="#">Administering Principals</a>	Removed creating similar organizations link information from <a href="#">Managing Organizations section</a> as this functionality is no longer supported.
Chapter 11, <a href="#">Using Types and the Type Manager</a>	<p>Added new <a href="#">Enhancing Performance on Systems with Many Soft Types</a> section to <a href="#">Best Practices</a>.</p> <p>Added character requirements for creating types.</p>
Chapter 19, <a href="#">Administering Audit Reports</a>	Added note regarding use of the forward slash ( / ) as a directory separator to the <a href="#">Enabling Auditing</a> section.

**Table 3 Changes for Windchill 8.0, Maintenance Release M020**

Chapter	Description
Chapter 2, <a href="#">Administration Overview</a>	Updated the <a href="#">Managing Data</a> section to include support for dynamic documents in Windchill PDMLink.

Chapter	Description
Chapter 3, <a href="#">Administering Containers</a>	<p>Updated the <a href="#">Windchill ProjectLink Context Templates</a> section to include role access for teams.</p> <p>Updated the <a href="#">Access to Creating Additional Context Templates</a> section to include the <b>Download</b> action.</p> <p>Added <a href="#">Understanding the Use of Versioning Schemes</a> section and added cautions to the <a href="#">Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects</a> and <a href="#">Administering the Versioning of Parts, Documents, and CAD Documents</a> sections about changing numbering and versioning schemes after objects have been created.</p> <p>Added name of Windchill PDMLink out-of-the-box state-based versioning file to the <a href="#">State-based Scheme</a> section.</p>
Chapter 6, <a href="#">Administering Products and Libraries</a>	<p>Added wt.inf.team .displayContainerListForGuest property description to the <a href="#">Out-of-the-box Container Participation</a> section.</p> <p>Added <a href="#">Configuring Product or Library Containers for Dynamic Documents</a> section.</p> <p>Added <a href="#">Configuring Part and Document Relationships</a> section.</p> <p>Added <a href="#">Allowing URLs as Content for Documents</a> section.</p>
Chapter 7, <a href="#">Administering Projects</a>	<p>Added wt.inf.team .displayContainerListForGuest property description to the <a href="#">Managing Project Team Members and Roles</a> section.</p> <p>Added <a href="#">Controlling the Visibility of Actions By Role</a> section to the <a href="#">Managing Project Team Members and Roles</a> section.</p>

Chapter	Description
Chapter 11, <a href="#">Using Types and the Type Manager</a>	<p>Updated the <a href="#">Overview of Types and the Runtime Typing Capability</a> section.</p> <p>Updated the <a href="#">For Windchill PDMLink and Windchill ProjectLink</a> Best Practices section.</p> <p>Soft attributes can now be added to WTPartsUsageLink objects and soft type and soft attributes are supported for change objects as described in sections:</p> <p style="padding-left: 40px;"> <a href="#">Defining Additional Attributes</a>  <a href="#">Restricting the Use of Soft Types and Soft Attributes</a> </p> <p>There are changes to how attributes for change objects are displayed. See section <a href="#">Client Changes</a>.</p> <p>Updated description of Immutable Constraint is in the <a href="#">Setting Constraints</a> section.</p> <p>Added following new sections:</p> <p style="padding-left: 40px;"> <a href="#">Specifying Windchill Types</a>  <a href="#">Managing Types and Attributes for Parts, Documents, and Change Objects</a>  <a href="#">Managing Types and Attributes for CAD Documents</a>  <a href="#">Managing Types and Attributes for Dynamic Documents</a> </p>
Chapter 13, <a href="#">Administering Notifications</a>	Added new <a href="#">Folder Subscription Notifications</a> section.



Chapter	Description
Chapter 14, <a href="#">Administering Life Cycles</a>	<p>Added new Two Level Approval Routing advanced life cycle template to <a href="#">Windchill ProjectLink</a> section.</p> <p>Added location of resource files to the <a href="#">Defining Life Cycle Phases</a> section.</p> <p>Modified <a href="#">State-based Revision Sequences by Life Cycle State</a> section to update XML file information.</p>
Chapter 15, <a href="#">Administering Teams and Roles</a>	Updated <a href="#">Refreshing Groups</a> section to clarify this functionality.
Chapter 16, <a href="#">Administering Workflow Processes</a>	Updated <a href="#">Access Control</a> section to remove limit restriction on ad-hoc access policies.
Appendix A, <a href="#">Creating Business XML Files for Templates</a>	Updated the <a href="#">ExportedRoleMemberMap Element</a> section to add project template support for exporting and importing role access data.

**Table 4 Changes for Windchill 8.0, Maintenance Release M010**

Chapter	Description
Chapter 3, <a href="#">Administering Containers</a>	<p>In the <a href="#">Installed Site Container Policies</a> section, added the access control rule set for the RelationshipMap object.</p> <p>Updated the <a href="#">Administering Domains and Policies</a> section to better describe domains and the association of domains to folders. The ability to add top-level domains in application containers is now supported.</p> <p>In the <a href="#">Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects</a> section, added a description of the preference you can use to remove the ability to override the autogenerated part number.</p>

Chapter	Description
Chapter 16, <a href="#">Administering Workflow Processes</a>	<p>Added guidelines to the <a href="#">Declaring Variables</a> section.</p> <p>Updated the <a href="#">Process Flags</a> section to include the wt.workflow.engine.useLatestTemplateIteration property.</p>
Chapter 17, <a href="#">Administering Views and View Associations</a>	Updated chapter to add the <b>[D]elete view</b> choice description and improve other choice descriptions.
Chapter 19, <a href="#">Administering Audit Reports</a>	Updated <a href="#">Enabling Auditing</a> section to clarify the use of the configAudit.xml file.
Appendix A, <a href="#">Creating Business XML Files for Templates</a>	<p>Added support for the ProductStructure element in project template XML files in the <a href="#">ProductStructure Element</a> section.</p> <p>Updated the <a href="#">OrgStructure Element</a> section to support the parentDomain element, better describe the domainName element, and correct the examples provided.</p> <p>Updated the <a href="#">ACLRule Element</a> section to better describe the domainName element.</p>

**Table 5 Changes for Windchill 8.0**

Chapter	Description
Chapter 1, <a href="#">Getting Started</a>	<p>Updated the <a href="#">Logging On as the Administrator</a> and <a href="#">Creating the Default Organization Container</a> sections to reflect base data changes for Windchill PDMLink.</p> <p>Updated the <a href="#">Establishing Administrators</a> section with e-mail sender information.</p>

Chapter	Description
Chapter 2, <a href="#">Administration Overview</a>	<p>Updated domain diagrams for containers in the <a href="#">Managing User Access to Data</a> section.</p> <p>Updated the <a href="#">About the windchill Command</a> and <a href="#">About the xconfmanager Utility</a> sections.</p>
Chapter 3, <a href="#">Administering Containers</a>	<p>Added <a href="#">Assigning Domains to Folders in Windchill PDMLink or Windchill Foundation &amp; PDM</a> section.</p> <p>Added <a href="#">Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects</a> section to provide an overview of numbering schemes.</p> <p>Added <a href="#">Administering the Versioning of Parts, Documents, and CAD Documents</a> section to document versioning schemes including the new state-based versioning.</p> <p>Added information about exporting and importing context templates in the <a href="#">Creating Context Templates</a> section.</p> <p>Added information about the formatting required for object initialization rules in the <a href="#">Administering Object Initialization Rules</a> section.</p> <p>Added information about the wt.org.client.applet.ActiveSearch property in the <a href="#">Searching for Principals</a> section.</p>

Chapter	Description
Chapter 4, <a href="#">Administering the Site</a>	<p>Added information about using the Purge Manager in the <a href="#">Purging, Archiving, and Restoring Jobs</a> section.</p> <p>Added information about requiring primary content in Windchill PDMLink documents to the <a href="#">Creating and Managing Site Folders and Documents</a> section.</p> <p>Added best practices information in the <a href="#">Setting Object Initialization Rules</a> and <a href="#">Setting Up Enhanced Life Cycle Templates</a> sections.</p>
Chapter 5, <a href="#">Administering Organizations</a>	<p>Added information about using the Purge Manager in the <a href="#">Purging, Archiving, and Restoring Jobs</a> section.</p> <p>Updated the <a href="#">Setting Default Preferences in Windchill ProjectLink to Collapse Tables</a> section because tables are collapsed out-of-the-box.</p>
Chapter 6, <a href="#">Administering Products and Libraries</a>	<p>Updated <a href="#">Configuring Numbering and Versioning Schemes</a> section to add site-based versioning.</p> <p>Added <a href="#">Managing the Life Cycle of Parts, Documents, CAD Documents, and Dynamic Documents</a> section.</p> <p>Updated <a href="#">Out-of-the-box Product and Library Context Templates</a> section to document Product Design template.</p> <p>Updated <a href="#">Administering Teams</a> section to document reassigning tasks.</p> <p>Added <a href="#">Creating Custom Change Monitor Reports</a> section.</p> <p>Added <a href="#">Best Practices For Object Initialization Rules</a> section for setting object initialization rules in individual product and library containers</p>

Chapter	Description
Chapter 9, <a href="#">Administering Principals</a>	<p>Added <a href="#">Searching for Users and Groups</a> section.</p> <p>Updated <a href="#">Managing Users</a> section.</p> <p>Updated chapter to reflect updated user interface.</p>
Chapter 11, <a href="#">Using Types and the Type Manager</a>	Included information about mapping CAD documents.
Chapter 14, <a href="#">Administering Life Cycles</a>	<p>Updated to include enhanced life cycle management.</p> <p>Added <a href="#">Basic and Advanced Life Cycles</a> section.</p>
Chapter 15, <a href="#">Administering Teams and Roles</a>	Added <a href="#">Refreshing Groups</a> section.
Chapter 16, <a href="#">Administering Workflow Processes</a>	<p>Added information about electronic signatures.</p> <p>Added information regarding two new promotion request workflows.</p> <p>Added <a href="#">Workflow Template Execution Flags</a> to document how to modify the workflow engine flags for an existing workflow.</p>
Chapter 18, <a href="#">Administering Visualization Services</a>	Updated table of Windchill Visualization Service properties and descriptions.
Appendix A, <a href="#">Creating Business XML Files for Templates</a>	New appendix describing content of business XML files that can be used when creating an organization template or importing a product, library, or project template.

**Table 6 Changes for Windchill 7.0, Maintenance Release M020**

Chapter	Description
Chapter 3, <a href="#">Administering Containers</a>	Updated to include a note about not modifying out-of-the-box rules and added information about the final and ignore AttrValue attributes.
Chapter 5, <a href="#">Administering Organizations</a>	Added a description of the OrganizationSync utility that can be used to update organization-related information.
Chapter 11, <a href="#">Using Types and the Type Manager</a>	Updated to include information about supported constraints for Windchill PDMLink and Windchill ProjectLink.
Chapter 14, <a href="#">Administering Life Cycles</a>	Updated to include information on how to import (into Windchill release 7.0) workflow templates exported from previous release of Windchill.
Chapter 15, <a href="#">Administering Teams and Roles</a>	Added a Refreshing Groups section to the Best Practices for Windchill PDMLink and Windchill ProjectLink section.
Chapter 16, <a href="#">Administering Workflow Processes</a>	<p>Updated to include:</p> <ul style="list-style-type: none"><li>• two new sections (Access Control and Workflow Process Support in Windchill ProjectLink) to the Best Practices for Windchill PDMLink and Windchill ProjectLink</li><li>• information on how to import into Windchill release 7.0 workflow templates exported from previous release of Windchill</li><li>• a description of DefineTeams activity</li><li>• a note that only required participants see routing choices</li><li>• a clarification of loop links.</li></ul>

**Table 7 Changes for Windchill 7.0**

Change	Description
Chapter 1, <a href="#">Getting Started</a>	This new chapter provides a road map for getting your Windchill solution set up to be usable as a test system and provides some basic information about your Windchill environment.
Chapter 2, <a href="#">Administration Overview</a>	This new chapter provides a general overview of your installed Windchill runtime architecture and Windchill environment. It also introduces you to the main Windchill administration areas and gives some basic information about how to manage your Windchill solution.
Chapter 3, <a href="#">Administering Containers</a>	This new chapter provides the overall details relating to working with containers. This chapter includes information that previously existed in Understanding Domains and Policies (chapter 1 in the previous Windchill Business Administrator's Guide).
Chapter 4, <a href="#">Administering the Site</a>	This new chapter provides an overview for administering sites and describes the typical duties that a site administrator performs.
Chapter 5, <a href="#">Administering Organizations</a>	The new chapter provides an overview for administering organizations and describes the typical duties that an organization administrator performs.
Chapter 6, <a href="#">Administering Products and Libraries</a>	This new chapter provides an overview for administering products and libraries in Windchill PDMLink and describes the typical duties that an administrator performs in the product and library contexts.

Change	Description
Chapter 7, <a href="#">Administering Windchill PDM Library</a>	This new chapter provides an overview for administering product and libraries in Windchill Foundation & PDM and describes the typical duties that an administrator performs. It also provides additional information about some of the main administrative tasks for the Windchill Foundation & PDM library container.
Chapter 8, <a href="#">Administering Projects</a>	This new chapter provides an overview for administering projects in Windchill ProjectLink and describes the typical duties that an administrator performs in the project context.
Chapter 9, <a href="#">Administering Principals</a>	This chapter includes information that previously existed in Administering Users and Groups (chapter 2 in the previous Windchill Business Administrator's Guide). The chapter documents the new user interface for the Principal Administrator and has been updated to incorporate organization as a new principal type.
Chapter 10, <a href="#">Administering Access Control</a>	<p>This chapter includes information that previously existed in Administering Access Control (chapter 5 in the previous Windchill Business Administrator's Guide). The chapter has been updated to include the following:</p> <ul style="list-style-type: none"> <li>• use of the Secured information feature</li> <li>• new Change Permissions permission</li> <li>• use of out-of-the-box policy rules</li> <li>• reorganization and update to Windchill 7.0.</li> </ul>



Change	Description
Chapter 11, <a href="#">Using Types and the Type Manager</a>	This chapter includes information that previously existed in Using Types and the Type Manager (chapter 3 in the previous Windchill Business Administrator's Guide). The chapter has been updated to reflect Windchill 7.0.
Chapter 12, <a href="#">Administering Indexing</a>	This chapter includes information that previously existed in Administering Indexing (chapter 6 in the previous Windchill Business Administrator's Guide). The chapter has been updated to incorporate organization as a new principal type.
Chapter 13, <a href="#">Administering Notifications</a>	This chapter includes information that previously existed in Administering Notifications (chapter 4 in the previous Windchill Business Administrator's Guide). The chapter has been updated to incorporate organization as a principal type.
Chapter 14, <a href="#">Administering Life Cycles</a>	This chapter includes information that previously existed in Administering Life Cycles (chapter 8 in the previous Windchill Business Administrator's Guide). The chapter has been updated to reflect Windchill 7.0.
Chapter 15, <a href="#">Administering Teams and Roles</a>	This chapter includes information that previously existed in Administering Teams and Roles (chapter 7 in the previous Windchill Business Administrator's Guide). The chapter has been updated to reflect Windchill 7.0.
Chapter 16, <a href="#">Administering Workflow Processes</a>	This chapter includes information that previously existed in Administering Workflow Processes (chapter 9 in the previous Windchill Business Administrator's Guide). The chapter has been updated to reflect Windchill 7.0.

Change	Description
Chapter 17, <a href="#">Administering Views and View Associations</a>	This chapter was previously chapter 10 in the Windchill Business Administrator's Guide.
Chapter 18, <a href="#">Administering Visualization Services</a>	This chapter includes information that previously existed in Administering Visualization Services (chapter 11 in the previous Windchill Business Administrator's Guide). The chapter has been updated to reflect Windchill 7.0 and includes information how to export watermarks to ProductView.
Chapter 19, <a href="#">Administering Audit Reports</a>	This new chapter provides information on how to enable and use the audit reporting feature.

# About This Guide

## Intended Audience

The *Windchill Business Administrator's Guide* serves as a reference guide for Windchill business and application administrators.

The following table illustrates the responsibilities and skills of each type of administrator:

	<b>Business Administrator</b>	<b>Applications Administrator</b>
<b>Responsibilities</b>	Use Windchill to achieve specific business goals, such as setting up workflows and entering users, user groups, ACLs, and so on.	Use Windchill applications to achieve goals appropriate for the application.
<b>Skills</b>	Understand the Windchill client and the business needs of enterprise users.	Understand the needs of particular application-user communities.

System administrators, who are responsible for keeping the system running, should refer to the *Windchill System Administrator's Guide*.

## Overview

The *Windchill Business Administrator's Guide* describes responsibilities and roles of Windchill business and application administrators, providing conceptual and background information to help them understand the nature of administrative tasks.

# Chapter Contents

The *Windchill Business Administrator's Guide* is composed of the following chapters and appendixes:

Chapter 1, Getting Started, provides a road map for getting your Windchill solution set up to be usable as a test system and provides some basic information about your Windchill environment.

Chapter 2, Administration Overview, provides a general overview of your installed Windchill runtime architecture and Windchill environment.

Chapter 3, Administering Containers, provides the overall details relating to working with containers.

Chapter 4, Administering the Site, provides an overview for administering sites and describes the typical duties that a site administrator performs.

Chapter 5, Administering Organizations, provides an overview for administering organizations and describes the typical duties that an organization administrator performs.

Chapter 6, Administering Products and Libraries, provides overview for administering products and libraries in Windchill PDMLink and describes the typical duties that an administrator performs in the product and library context.

Chapter 7, Administering Windchill PDM Library, provides an overview for administering product and libraries in Windchill Foundation & PDM and describes the typical duties that an administrator performs.

Chapter 8, Administering Projects, provides an overview for administering projects in Windchill ProjectLink and describes the typical duties that an administrator performs in the project context.

Chapter 9, Administering Principals, describes the principals (user, group, and organization objects) that are used in your Windchill solution. It also provides details about managing the principals.

Chapter 10, Administering Access Control, describes access control policies, consisting of rules that govern access to objects. It includes the Windchill model for storage of enterprise information and strategies for creating access rules that best serve your site's security needs.

Chapter 11, Using Types and the Type Manager, describes the definition of subtypes, attributes, and constraints.

Chapter 12, Administering Indexing, describes indexing policies and specifies the indexes into which an object is to be entered when a specified event occurs, and it describes how to define and bulk-load index collections when setting up a new site or upgrading to a new release.

Chapter 13, Administering Notifications, describes notification policies, which determine users and groups to be notified when specific events are applied to an object.

Chapter 14, Administering Life Cycles, describes life cycles, which define how an object matures and provides models for product commercialization.

Chapter 15, Administering Teams and Roles, describes the definition of teams and the mapping of team roles to principals.

Chapter 16, Administering Workflow Processes, describes workflow processes, which enable an organization to automate procedures in which information, tasks, and documents are passed among participants.

Chapter 17, Administering Views and View Associations, describes view setups and view associations for your Windchill system.

Chapter 18, Administering Visualization Services, describes the configuration and troubleshooting of the visualization service.

Chapter 19, Administering Audit Reports, provides information on how to enable and use the audit reporting feature.

Appendix A, Creating Business XML Files for Templates, provides information to assist in creating business XML files that can be used when creating an organization template, or creating or importing a product, library, or project template.

## Related Documentation

The following documentation may also be helpful:

- *Windchill System Administrator's Guide*
- *Windchill Installation and Configuration Guide - Windchill*
- *Windchill Foundation & PDM User's Guide*
- *Windchill PDMLink User's Guide*
- *Windchill ProjectLink User's Guide*
- *Windchill Archive Administrator's Guide*
- *Windchill Customizer's Guide*
- *Windchill Adapter Guide*
- *Windchill Performance Tuning Guide*
- properties.html file
- *Getting Started with Pro/INTRALINK 8.0*
- *Pro/INTRALINK 8.0 and Windchill PDMLink User's Guide*

- *Getting Started with Windchill PDMLink and Arbortext Editor*

Not all guides are available; only those guides for the products installed are available. If a guide you are interested in reading is not installed at your site, contact whoever did the installation. You can also access the guides from the Reference Documents section of the PTC Web site as described in [Documentation for PTC Products](#).

## Technical Support

Contact PTC Technical Support via the PTC Web site, phone, fax, or e-mail if you encounter problems using Windchill.

For complete details, refer to Contacting Technical Support in the *PTC Customer Service Guide* enclosed with your shipment. This guide can also be found under the Support Bulletins section of the PTC Web site at:

<http://www.ptc.com/support/index.htm>

The PTC Web site also provides a search facility that allows you to locate Technical Support technical documentation of particular interest. To access this page, use the following link:

<http://www.ptc.com/support/support.htm>

You must have a Service Contract Number (SCN) before you can receive technical support. If you do not have an SCN, contact PTC License Management using the instructions found in your *PTC Customer Service Guide* under Contacting License Management.

## Documentation for PTC Products

PTC provides documentation in the following forms:

- Help topics
- PDF books

To view and print PDF books, you must have the Adobe Acrobat Reader installed.

All Windchill documentation is included on the CD for the application and copied to the application when it is installed. In addition, all books [including those updated after release (for example, to support a hardware platform certification)] are available from the Reference Documents section of the PTC Web site at the following URL:

<http://www.ptc.com/appserver/cs/doc/refdoc.jsp>


## Comments

PTC welcomes your suggestions and comments on its documentation. You can submit your feedback through the online survey form at the following URL:

[http://www.ptc.com/go/wc\\_pubs\\_feedback](http://www.ptc.com/go/wc_pubs_feedback)

## Documentation Conventions

Windchill documentation uses the following conventions:

Convention	Item	Example
Bold	Names of elements in the user interface, such as buttons, and menu paths.  Required elements and keywords or characters in syntax formats.	Click <b>OK</b> .  Select <b>File &gt; Save</b> .  <b>create_&lt;tablename&gt;.sql</b>
Italic	Variable and user-defined elements in syntax formats. Angle brackets (< and >) enclose individual elements.	<b>create_&lt;tablename&gt;.sql</b>
Monospace	Examples Messages	JavaGen "wt.doc.*" F true  Processing completed.
	The <b>CAUTION</b> symbol indicates potentially unsafe situations which may result in minor injury, machine damage or downtime, or corruption or loss of software or data.	When you add a value to an enumerated type (for example, by adding a role in the RolesRB.java resource file), removing that value can result in a serious runtime error. Do not remove a role unless you are certain there is no reference to it within the system.





# 1

## Getting Started

This chapter provides a road map for getting your Windchill solution set up so that it is usable as a test system. The chapter also provides some basic information about the use of Windchill administrators and your Windchill environment.

<b>Topic</b>	<b>Page</b>
Introduction .....	1-2
Logging On as the Administrator .....	1-3
Creating the Default Organization Container.....	1-8
Establishing Administrators .....	1-12
The Next Steps .....	1-16

# Introduction

This guide provides business administration information for the following Windchill solutions:

Windchill Foundation & PDM

Windchill PDMLink

Windchill ProjectLink

After a Windchill solution installation is complete, the following basic activities have been accomplished:

- A Web server and servlet engine are installed and configured
- The Oracle database is installed and configured
- Any enterprise directory services that are going to be used are configured
- The Windchill solution is installed and has been started

If you want more information about these activities, see the *Windchill Installation and Configuration Guide - Windchill*.

Before you can get started with administrative activities in your Windchill solution, you must log on as the Administrator (defined during the installation). Additionally for Windchill PDMLink and Windchill ProjectLink, you may need to create a container for the installed default organization and establish additional administrators.

The next sections in this chapter describe how to log on, create an organization container, and establish additional administrators. The last section provides a guide to which additional chapters you may want to read next.

## Regarding Arbortext Content Manager

Parts of this guide apply to Arbortext Content Manager. The general information and information specific to Windchill PDMLink is relevant to Arbortext Content Manager administrators.

**Note:** As parts and CAD documents are not supported for Arbortext Content Manager, disregard references to those objects. References to "Arbortext" or "Arbortext products" refer to the Arbortext Editor and Arbortext Publishing Engine, not the Arbortext Content Manager solution.

For additional information on Arbortext Content Manager, see the *Arbortext Content Manager User's Guide*.

## Regarding Pro/INTRALINK 8.0

Parts of this guide apply to Pro/INTRALINK 8.0. The general information and information specific to Windchill PDMLink is relevant to Pro/INTRALINK 8.0 administrators.

**Note:** As parts and dynamic documents are not supported for Pro/INTRALINK 8.0, disregard references to those objects.

For information about what is supported in Pro/INTRALINK 8.0, see *Getting Started with Pro/INTRALINK 8.0*.

## Logging On as the Administrator

You can access your Windchill solution using a URL from a Web browser.

**Note:** The server manager and method server must be running (as well as the Web server and servlet engine) before Windchill can be accessed.

The URL string is formatted as follows:

```
http://<hostname>:<port>/<webapp>
```

The required parameters were defined when Windchill Info\*Engine was installed. You only need to include the port number when the Web server being used is using a port number other than 80 (default). If the Web server is using the default port, then you can enter `http://<hostname>/<webapp>` in your Web browser **Address** (or **Location**) text box. For example, if you specified Windchill for the `<webapp>` parameter, then the following URL entered in the Web browser **Address** text box will open the Windchill home page:

```
http://<hostname>/Windchill
```

**Tip:** If you are logging on using the same system where your solution is installed, you can use **localhost** as `<hostname>`.

Use the administrative user defined during the installation to log on. This user is a member of the Administrators group and has out-of-the box privileges that give you full control over all Windchill objects.

The home page that appears is unique to your Windchill solution. The following sections describe each home page.

## Windchill Foundation & PDM Home Page

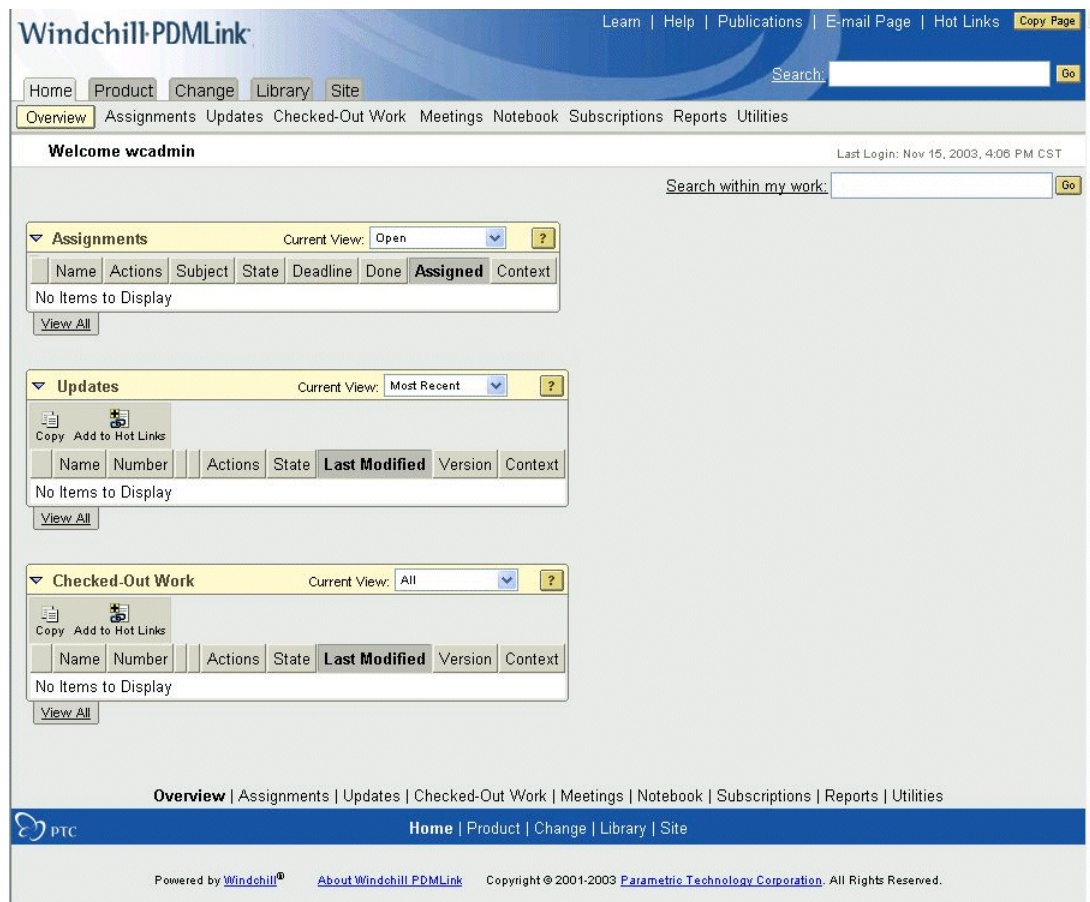
The Windchill Foundation & PDM home page that appears is similar to the following:



The first time you access the Windchill home page, you can select one of the links listed under **Available Homes** to make that page your personal home page. Common home pages for the administrator are the **Business Administration** and **System Administration** home pages. The next time you access Windchill, it will open to that page. If, at any time, you want to change your personal home page, click **Options** or **Site Map**, and then click the link to the page you want as your new personal home page.

## Windchill PDMLink Home Page

The Windchill PDMLink home page that appears is similar to the following:



The tabs at the top of the window identify the major areas provided in Windchill PDMLink:

- From the **Home** tab, users manage their daily work.
- From the **Product** tab, users have access to all products to which they are a member. For each product, team members manage all of the information that is relevant to the design, manufacture, and support of a product. When only base data is installed, there are no out-of-the-box products.
- From the **Library** tab, users have access to all libraries to which they are a member. In a library, team members can store and provide access to business information (such as in a document library) or can store and provide access to objects that are not related to a single product (such as in a common parts library). There are no out-of-the-box libraries.

- From the **Site** tab, site administrators configure and manage the Windchill system as a whole. This tab is only visible to site administrators and is the tab from which the initial administration activities, such as creating the organization container, are done.

**Note:** An **Organization** tab also appears if you have loaded the PDMLinkOrgContainer.xml file. This file is part of the base data supplied with your installation. The base data includes data definitions for container templates, workflows, life cycles, preferences, and rules that are required for a functioning Windchill system. For details on loading base data, see the *Windchill Installation and Configuration Guide - Windchill*. The organization container created from the PDMLinkOrgContainer.xml file is defined as follows:

- The organization name for the container and its corresponding object is the organization name entered in the Windchill Services installation.
- The organization object created (type WTOrganization) is the owning organization for the container.

**Note:** This organization object is also the owning organization for activities done in the Site container. Meaning, the same organization object owns both the Site container and this organization container. Since soft types are created in the context of the WTOrganization object, soft types created within this organization container become site level soft types. For more information, see [Associating Soft Types with the Site and Organization Containers](#) in the Using Types and the Type Manager chapter.

- The description entered is:

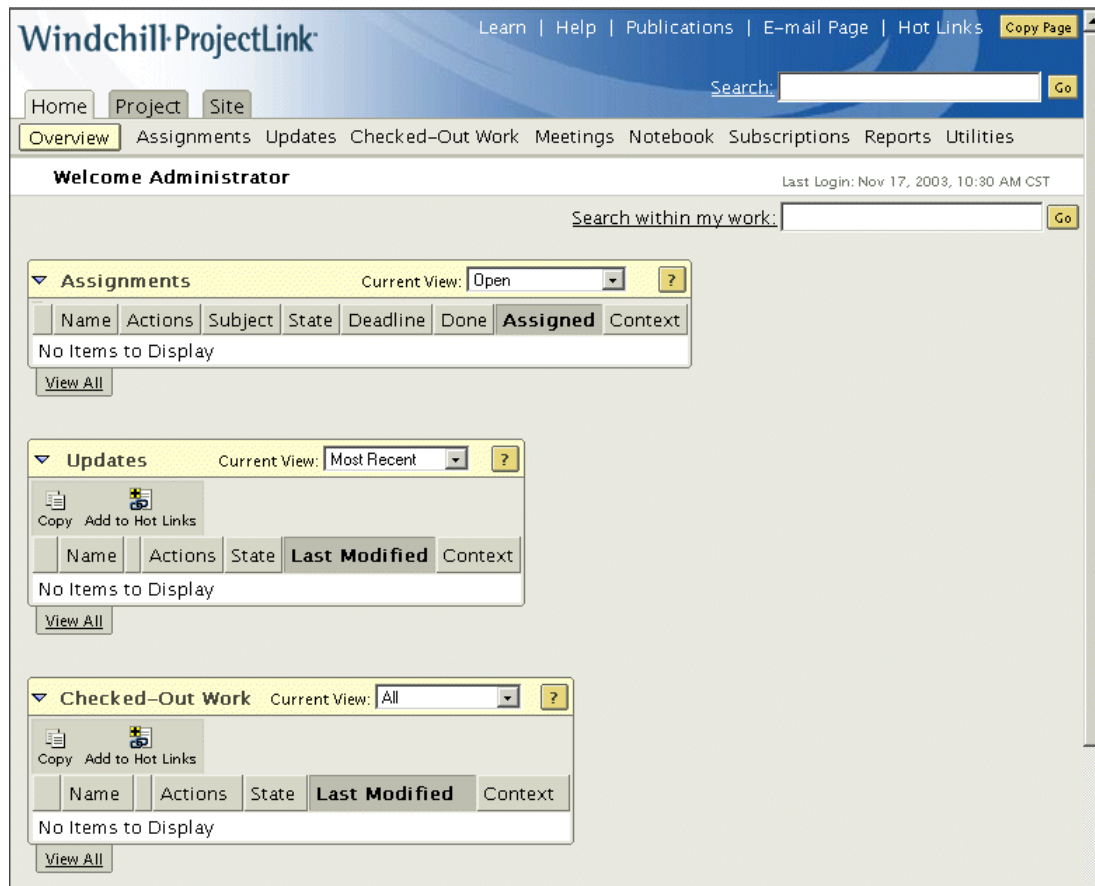
```
Context for the organization hosting this Windchill
installation
```

- The General (PDM) template is used to create the organization container. For details on what is set by this template, see [Out-of-the-Box Organization Templates](#).
- The organization can host products and libraries.
- Organization members can search for all users, not just those in their organization (if users from other organizations have been defined through additional directory services).

If the PDMLinkOrgContainer.xml file was not loaded, the **Organization** tab appears after you create an organization container. Creating an organization container is described in the next section, [Creating the Default Organization Container](#).

## Windchill ProjectLink Home Page

The Windchill ProjectLink home page that appears is similar to the following:



The tabs at the top of the window identify the major areas provided in Windchill ProjectLink:

- From the **Home** tab, users manage their daily work.
- From the **Project** tab, users have access to all projects to which they are a member. For each project, team members have access to the project information, the project schedule, resources, and plan details. There are no out-of-the-box projects.

- From the **Site** tab, site administrators configure and manage the Windchill system as a whole. This tab is only visible to site administrators and is the tab from which the initial administration activities, such as creating the organization container, are done.

**Note:** An **Organization** tab appears after you create an organization container (as described in the next section).

## Creating the Default Organization Container

Your Windchill environment consists of a set of containers that hold all of the administrative areas (known as *domains*), rules, and data that make up the context from which Windchill users work.

The containers are set up in a hierarchy so that the rules and data created in one container can be available to child containers and the domains in one container can have child domains in a child container. The child domains inherit information from ancestor domains.

Initially, every Windchill solution has an installed top-level Site container. In Windchill PDMLink and Windchill ProjectLink, Site container activities are performed from the **Site** tab. In Windchill Foundation & PDM, Site container activities are performed from the **System Administration** page.

Additionally for Windchill Foundation & PDM, the installation creates an organization container which is named during the installation process.

**Note:** In Windchill PDMLink, you can create an organization container by either:

- Loading the PDMLinkOrgContainer.xml file, which is part of the base data. For details on the organization container created, see [Windchill PDMLink Home Page](#) earlier in this chapter.
- Creating an organization container, as described later in this section.

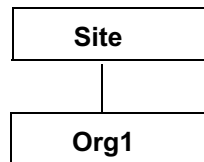
**Tip:** Selecting to load all of the base data loads the PDMLinkOrgContainer.xml file, thus creating an organization container that has the name of the organization that was specified when Windchill Services was installed.

**Note:** In Windchill ProjectLink, the base data installation does not create an organization container; you create the organization container, as described later in this section.



The organization container is a child of the Site container. For Windchill Foundation & PDM and Windchill PDMLink (if the PDMLinkOrgContainer.xml file is loaded), the container name is the organization name that was entered during the Windchill Services installation. For Windchill PDMLink (if the PDMLinkOrgContainer.xml file is not loaded) and Windchill ProjectLink, you enter the organization name when you create the container.

For example, assume that the organization name is Org1, then the following container hierarchy is established:




In this example, the Org1 organization container inherits rules and data from the Site container.

In Windchill PDMLink and Windchill ProjectLink, you must create a default organization container and set up other administrators before other users can perform activities in the Windchill solution.

**Note:** With Windchill PDMLink, the installer can install the golf cart demo data. As part of the golf cart installation, the Demo organization container is created and the Demo user is created as the organization administrator. As is the case for all organization containers, the Demo organization container inherits rules and data from the Site container.

To create the default organization container, complete the following steps:

1. Log on as the site administrator, as described in [Logging On as the Administrator](#).
2. Navigate to the **Site** tab and click **Create Organization** .

The following **Create Organization** window opens:

Create Organization - Microsoft Internet Explorer provided by PTC

## Create Organization ?

\*Organization Name:  Search...

Description:

Postal Address:

Web Site:

\*Template: Demo Template ▼

☒ Subscriber – organization can host libraries and products

☐ Allow entire user and group directory selection

Conferencing ID:

Conferencing URL:

\* Indicates required fields.

OK Apply Cancel

Done Internet

For detailed information on these fields, see the help available from the window.

- For the **Organization Name**, click **Search**.

The search results contains the name of the default organization created during the installation.

4. Select the name of the default organization and click **OK**.
5. Enter information in the **Description**, **Postal Address**, and **Web Site** fields.
6. Select a template from the **Template** drop-down list, as follows:
  - For Windchill PDMLink, select **General (PDMLink)**.
  - For Windchill ProjectLink, select **General** unless you are creating an enterprise, supplier, or customer organization container.
7. Select the check boxes that correspond to the organization options you want, as follows:
  - By default, the **Subscriber** check box is selected. This means the organization can host products and libraries (for Windchill PDMLink standalone), projects (for Windchill ProjectLink standalone), or all three for Windchill PDMLink and Windchill ProjectLink sites.
  - For Windchill ProjectLink, the **Automatically add new members to project creators group** check box shows on the window (not shown in this figure) and is selected by default. Keep this selection for Windchill ProjectLink.
  - By default, you are restricted to seeing only users and groups that belong to the organization. Select **Allow entire user and group directory selection** to provide the ability to search for all users. Windchill PDMLink recommends you select this check box.
8. For Meeting Center access, set the conferencing ID and URL in the corresponding fields. Additional set up is required after the container is created and is described in the *Windchill System Administrator's Guide*.
9. Click **OK** to create the default organization container.

The Windchill solution updates to include the **Organization** tab.

**Note:** Next, you must set the organization administrator for the container. How to do this is described in the next section:

- For Windchill PDMLink, see [Windchill PDMLink Administrators](#).
- For Windchill ProjectLink, see [Windchill ProjectLink Administrators](#).

For more information about organization containers, see [Administering Organizations](#).

## Establishing Administrators

Before you attempt to use your Windchill solution, you should establish a minimum set of administrators for the solution. The types of administrators that are available are determined by the Windchill solutions that are installed.

Every installed Windchill solution defines the Administrators group of which, by default, the administrative user entered during the installation (for example, wadmin) is initially the only member. Out of the box, the members of the Administrators group have full control over all Windchill objects and are commonly called the *system* or *site administrators*.

The Windchill PDMLink and Windchill ProjectLink solutions provide additional types of administrators:

- An organization administrator manages a specific organization.
- A product manager manages a specific product (Windchill PDMLink only).
- A library manager manages a specific library (Windchill PDMLink only).
- A project manager manages a specific project (Windchill ProjectLink only).

For e-mail messages to be sent, some mail servers require that a valid e-mail address be specified in the From header of the e-mail message. Windchill solutions use the wt.mail.from and wt.notify.notificationSenderEmail properties in the wt.properties file to populate the From header when sending e-mail messages. Setting these properties to a valid user should have been done as a manual step after installing your solution. If this was not done, see the Completing Configurations - Manual Steps chapter of the *Windchill Installation and Configuration Guide - Windchill* for details on how to set these properties to the e-mail address of the Windchill administrator or some other authorized user.

The following sections provide additional information about the administrators in each Windchill solution and describe how to set the administrators.

### Windchill Foundation & PDM Administrators

In Windchill Foundation & PDM, the site administrator (as defined by the Administrators group) is the main administrator. This group of users can access all of the links that are on the **Business Administration** and **System Administration** pages. The site administrator is also known as the system administrator.

The site administrator is the only administrator initially needed for administering Windchill Foundation & PDM. If you want other users to be able to administer specific areas in Windchill Foundation & PDM, you add those users to the administrative groups that are created. For additional information, see [Additional Administrative Groups](#) in the Administration Overview chapter.

## Windchill PDMLink Administrators

In Windchill PDMLink, the site administrator (as defined by the Administrators group) can use the links that are on the **Site** and **Organization** tabs. The site administrator is also known as the system administrator.

Clicking the **Administrators** link on the **Site** tab displays the **Site Administrators** table. From this table, you can add users to the Administrators group.

Additional Windchill PDMLink administrators include the following:

- An organization administrator for managing a specific organization. Initially, there is no organization administrator defined for an organization container.

After you create the default organization container, click the **Administrators** link on the **Organization** tab to add one or more users as organization administrators to the default organization. For additional information, see [Creating Users to Select as Administrators](#).

- A product manager for managing a specific product.

Initially, the product manager is the user who creates the product. Additional users can be added to the **Product Manager** role from the **Team** page of the product.

- A library manager for managing a specific library.

Initially, the library manager is the user who creates the library. Additional users can be added to the **Library Manager** role from the **Team** page of the library.

## Creating a Product or Library

To create a product or library, a user must be one of the following:

- The site administrator, but only if the administrator is a member of the organization. The site administrator can become a member of the organization by setting in the organization attribute ("o", by default) for Administrator user in the directory service user entry or by using the usersOrganizationName property. For more information on using this property, see the Configuring Windchill to Use an Enterprise Directory chapter of the *Windchill Installation and Configuration Guide - Windchill*.
- The organization administrator.
- A member of the product creators group or the library creators group. These groups are maintained through the **Creators** link on the **Organization** tab. From the **Organization** tab, click the **Creators** link and then choose the type of creator from the **Current View** drop-down list on the **Creators** table.

## Windchill ProjectLink Administrators

In Windchill ProjectLink, the site administrator (as defined by the Administrators group) can use the links that are on the **Site** and **Organization** tabs. The site administrator is also known as the system administrator.

Clicking the **Administrators** link on the **Site** tab displays the **Site Administrators** table. From this table, you can add users to the Administrators group.

Additional Windchill ProjectLink administrators include the following:

- An organization administrator for managing a specific organization. Initially, there is no organization administrator defined for an organization container.

After you create an organization, click the **Administrators** link from the **Organization** tab to add one or more users as organization administrators to the current organization. For additional information, see [Creating Users to Select as Administrators](#).

- A project manager for managing a specific project.

Initially, the project manager is the user who creates the project. Additional users can be added to the **Project Manager** role from the **Team** page of the project.

## Creating a Project

To create a project, a user must be one of the following:

- The site administrator, but only if the administrator is a member of the organization. The site administrator can become a member of the organization by setting in the organization attribute ("o", by default) for Administrator user in the directory service user entry, or by using the usersOrganizationName property. For more information on using this property, see the Configuring Windchill to Use an Enterprise Directory chapter of the *Windchill Installation and Configuration Guide - Windchill*.
- The organization administrator.
- A member of the project creators group.

Depending on the setup options selected when an organization is created, all members of the organization who have logged in may be automatically added to the project creators group. Additionally, both the system administrator and the organization administrator of the current organization can add users to the project creators group. From the **Organization** tab, click the **Creators** link. From the **Project Creators** table that appears, you can add users.

## Creating Users to Select as Administrators

**Note:** Only members of an organization can be organization administrators, product creators, library creators, or project creators.

In a production environment, users are usually defined in an enterprise directory that is set up during installation. If you are setting up a test system and do not have a set of users from which to select administrators, you can create a set of test users in the default directory service using the Principal Administrator. In Windchill PDMLink and Windchill ProjectLink, the Principal Administrator is available from the **Site** tab. Click the **Utilities** link and then click the **Principal Administrator** link to open the Principal Administrator. In Windchill Foundation & PDM, the link is on the **Business Administration** home page.

The Administrator user that is created during the installation (for example, wadmin) is not associated to a specific organization; this user does not have the organization attribute (usually "o") set. Therefore, this user cannot be added as an organization administrator unless Administrator is updated to include the organization attribute. You can use the Principal Administrator to update the Administrator user.

**Note:** If your site does not use the organization attribute in the directory service entry, users can be assigned to an organization using the usersOrganizationName property. For more information on using this property, see the Configuring Windchill to Use an Enterprise Directory chapter of the *Windchill Installation and Configuration Guide - Windchill*.

At a minimum, you need to either update the Administrator user so the user is a member of the default organization or create another user who can be the organization administrator. This user can then administer the organization and create products, libraries, and projects.

## The Next Steps

After your administrators are established, individual administrators can log on and perform their administrative duties. For example, they can create products, libraries, and projects.

To understand what those duties might entail, refer to the following table:

Type of Administrator	Recommended Chapters to Read
Site administrator for Windchill PDMLink and Windchill ProjectLink	<a href="#">Administration Overview</a> - for general information <a href="#">Administering Containers</a> - for major container concepts <a href="#">Administering the Site</a> - for details using on the Windchill PDMLink and Windchill ProjectLink <b>Site</b> tab
Site administrator for Windchill Foundation & PDM	<a href="#">Administration Overview</a> - for general information <a href="#">Administering Containers</a> - for major container concepts <a href="#">Administering Windchill PDM Library</a> - for the duties surrounding administering Windchill Foundation & PDM
Organization administrator	<a href="#">Administration Overview</a> - for general information <a href="#">Administering Containers</a> - for major container concepts <a href="#">Administering Organizations</a> - for details on using the Windchill PDMLink and Windchill ProjectLink <b>Organization</b> tab and creating organization containers
Product manager	<a href="#">Administering Products and Libraries</a> - for details on administering Windchill PDMLink products
Library manager	<a href="#">Administering Products and Libraries</a> - for details on administering Windchill PDMLink libraries
Project manager	<a href="#">Administering Projects</a> - for details on administering Windchill ProjectLink projects



# 2

## Administration Overview

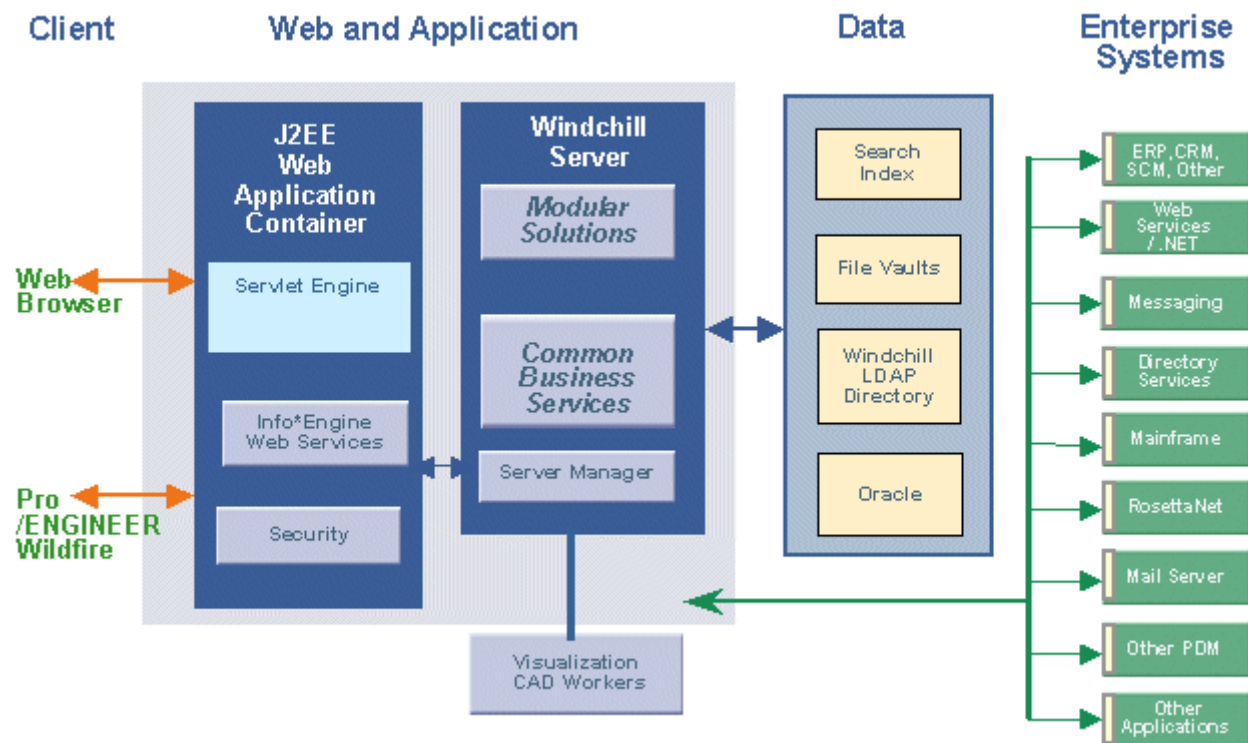
This chapter provides a general overview of your installed Windchill runtime architecture and Windchill environment. It also introduces you to the main Windchill administration areas and gives some basic information about how to manage your Windchill solution.

<b>Topic</b>	<b>Page</b>
Your Installed Windchill Runtime Architecture .....	2-2
Managing Your Windchill System.....	2-3
Your Installed Windchill Environment.....	2-3
Managing User Access to Data .....	2-6
Managing Users.....	2-14
Managing Data .....	2-16
Managing Windchill Processes .....	2-20
Managing User Collaboration .....	2-21
Additional Administrative Groups .....	2-21
Post-Installation Activities .....	2-22
About the windchill Command .....	2-23
About the windchill shell .....	2-25
About the xconfmanager Utility.....	2-26

## Your Installed Windchill Runtime Architecture

After a base Windchill solution is installed, the Windchill runtime architecture consists of the following:

- Client applications that allow users access to Windchill. The clients can include the Windchill client pages, the visualization clients, the Pro/ENGINEER Wildfire client, and, possibly, workgroup manager clients.
- A Web server that includes a servlet engine, Windchill Info\*Engine Web services, and security modules.
- The Windchill server that includes the Windchill solutions and common business services.
- Data storage that includes the Aphelion LDAP directory service and an Oracle database.
- Possibly, connections to other enterprise systems, such as an enterprise directory service, ERP, CRM, SCM, or other PDM systems.



For details about what is installed, see the *Windchill Installation and Configuration Guide - Windchill*.

# Managing Your Windchill System

The *Windchill System Administrator's Guide* describes how to perform system operations that change and improve the out-of-the-box system that is installed. The following topics are covered in the *Windchill System Administrator's Guide*:

- Runtime Services, including starting and stopping Windchill, defining user preferences, using the xconfmanager utility, setting up Meeting Center, setting up internal organizations.
- The Bootstrap Client
- External File Vaults
- Content Replication
- Windchill Import and Export
- Background Queues
- Pro/ENGINEER Wildfire
- Customizing Online Help
- Customizing Online Tutorials

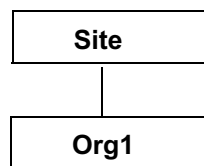
## Your Installed Windchill Environment

Your installed Windchill environment consists of a set of containers that hold all of the administrative areas (known as domains), rules, and data that make up the context from which Windchill users work.

The containers are set up in a hierarchy so that the rules and data created in one container can be available to child containers and the domains in one container can have child domains in a child container. The child domains inherit information from ancestor domains.

Every Windchill solution has an installed top-level Site container.

Additionally for Windchill Foundation & PDM, the installation creates an organization container which is named during the installation process and, for Windchill PDMLink, loading the base data creates an organization container which is named during the installation process. This container is a child of the Site container. The container name is the organization name that was entered during the installation. For example, assume that the organization name entered during the installation is Org1, then the following containers are created in Windchill Foundation & PDM:



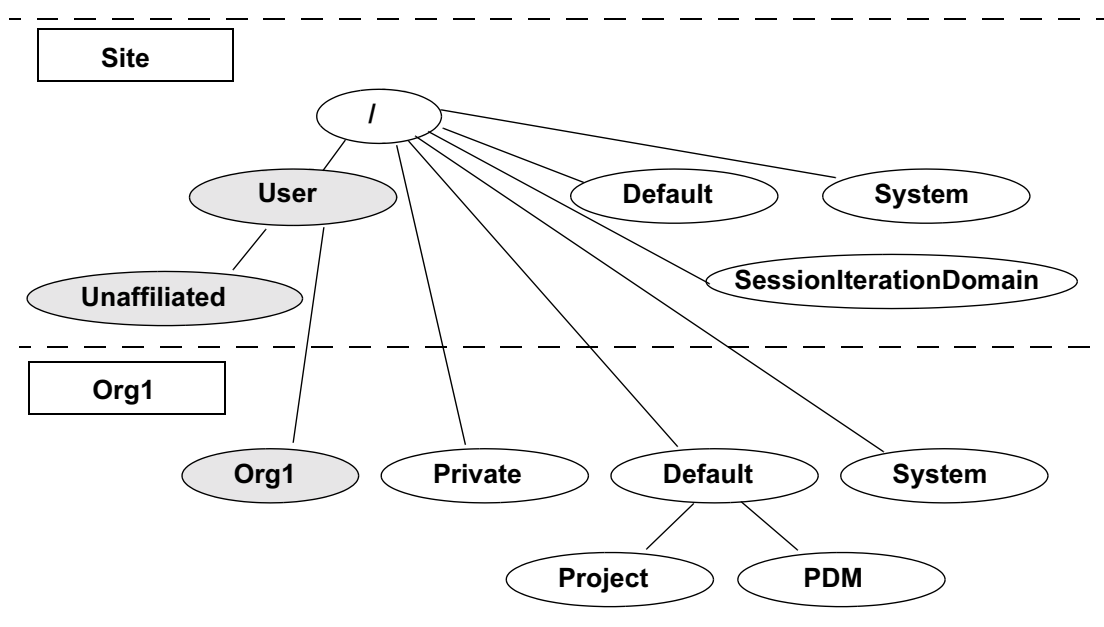
In this example, the Org1 organization container inherits rules and data from the Site container.

**Note:** For Windchill ProjectLink, the base data installation does not create an organization; you create the organization container after the installation is complete, as described in [Creating the Default Organization Container](#).

Regardless of how organization containers are created, all organization containers are children of the Site container, and they inherit rules and data from the Site container.

With Windchill PDMLink, the installer can install the golf cart demo data. As part of the golf cart installation, the Demo organization container is created. As is the case for all organization containers, the Demo organization container inherits rules and data from the Site container.

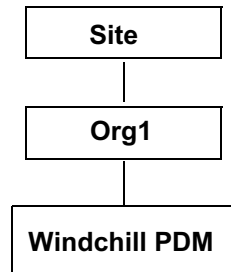
Within the installed containers, a set of domains are loaded during the installation process. For example, the Windchill Foundation & PDM Site and Org1 containers have the domains shown in following diagram:



In the diagram, the dashed line shows the container boundaries and the domain inheritance is shown by the lines connecting the domains. The top-level domain is labeled / (root) and is in the Site container. The shaded domains are the domains associated with Windchill principals (users, groups, and organizations).

The Windchill Foundation & PDM installation also creates a third container, called the Windchill PDM library container, which is the one and only application container in Windchill Foundation & PDM and is a child of the organization

container. For example, the following diagram shows the Site, Org1, and Windchill PDM containers:



The only additional domain loaded during the Windchill Foundation & PDM installation, is the ChangeItems domain. It inherits from the / (root) domain in the Site container.

In Windchill Foundation & PDM, the administrator cannot create additional containers.

After containers are created and users become team members, the framework established is called the *context* from which the users work. In many instances, the context includes the contents of a specific container and the domains, rules, and data available from ancestor containers. For example, if a user entering Windchill ProjectLink navigates to a folder within the Bike Design project and creates a new document, that document is managed in the context of the Bike Design project. Persons with access to the Bike Design project may automatically have the right to see and modify the new document.

Depending on how container rules are set up, users may also be able to share data across containers. When this is the case, the user context can include data from multiple containers. You can think of the context as providing the framework from which user actions are executed. This framework is defined by a container, but can include data from multiple containers. For example, parts defined in one container can be used in an assembly structure that is saved in a different container.

Each context provides the following:

- The context structure, which includes the default domains and folders, forum topics, reference notebook folders, and user notebook folders (if used).
- Context participation for Windchill PDMLink and Windchill ProjectLink, which includes the available roles, teams, and groups.
- Default access policies.
- Data types and object initialization rules.
- Default life cycle, workflow, context, team, and report templates.

The base data that is loaded during the installation process creates the out-of-the-box templates for containers, workflows, life cycles, teams, and reports, and associates them with the System domain that is in the Site container. These templates are then available to descendent containers where appropriate.

The Administrator user and the Administrators group are created during the base installation and are also associated with the System domain. Out of the box, these administrators have full access control over all Windchill objects.

One important type of data that is loaded is the context template data. Context template data files are XML files that define what is initially in a container when it is created. The file contains the types of items that are similar to the type of data, rules, and domains that are loaded during the Windchill solution installation. When creating additional containers, the administrator selects the context template data file to use to establish the context. For more information on context templates see, [Creating Containers](#) in the Administering Containers chapter.

## Managing User Access to Data

All Windchill installations establish a default host organization. The Windchill Foundation & PDM installation creates an organization container that has the same name as the host organization. For Windchill PDMLink and Windchill ProjectLink, the site administrator creates the organization container for the default host organization as a post installation step (as described in [Creating the Default Organization Container](#)). To belong to the default organization, each user must have an entry in the user directory service that was established as part of the installation process and the organization attribute of the entry (usually "o") must be set to the organization name. By being a member of the host organization, users usually have access to the data stored in the organization container or its child containers (depending on the access control rules that are in place). Users who are not members of the host organization usually do not have access to the data stored in the host organization container or its child containers (unless they are invited to participate by someone in the organization). Again, this is dependent on the access control rules that are in place.

**Note:** If your site does not use the organization attribute in the directory service entry, users can be assigned to an organization using the `usersOrganizationName` property. For more information on using this property, see the *Configuring Windchill to Use an Enterprise Directory* chapter of the *Windchill Installation and Configuration Guide - Windchill*.

After analyzing the users who need access to data, the system administrator determines whether additional organization containers are needed in Windchill PDMLink or Windchill ProjectLink. The organization container can limit data access to members of an organization. For example, if your Windchill solution will be used by multiple companies where each company has a different set of data and rules that will be used from within the solution, then setting up an organization container for each company would be the best approach. However, if only one company will actively use the solution and other companies will just

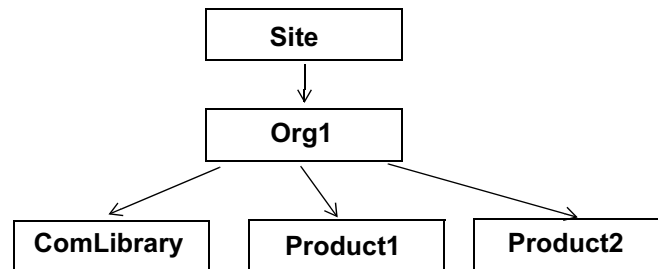
provide data that is managed by the host organization, then one organization container is sufficient.

Additionally, Windchill PDMLink administrators can create product and library application containers under an organization, and Windchill ProjectLink users can create project application containers under an organization. Using application containers further separates the access of data. In each container, unique policy rules are set. For example:

- The permissions for parts stored in a product container are usually set so only members of the product have write access to the parts.
- The permissions for documents stored in a library container are usually set so only members of the library have write access to the documents.
- The permissions for all objects stored in a project container are usually set so only members of the project have write access to the objects.

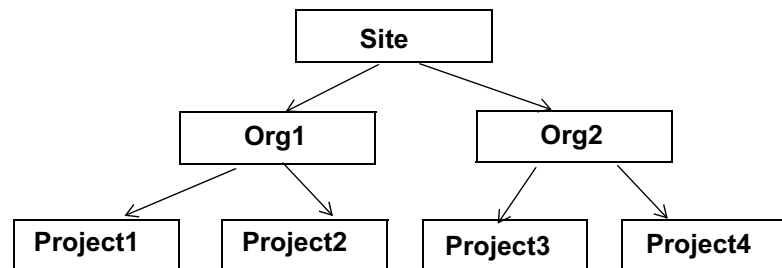
## Windchill PDMLink Container Hierarchy

The following diagram shows the container hierarchy of a Windchill PDMLink solution where administrators have created one library container and two product containers under the Org1 container:



## Windchill ProjectLink Container Hierarchy

The following diagram shows the container hierarchy of a Windchill ProjectLink solution where there are two organizations (Org1 and Org2) and administrators have created two project containers under each organization:

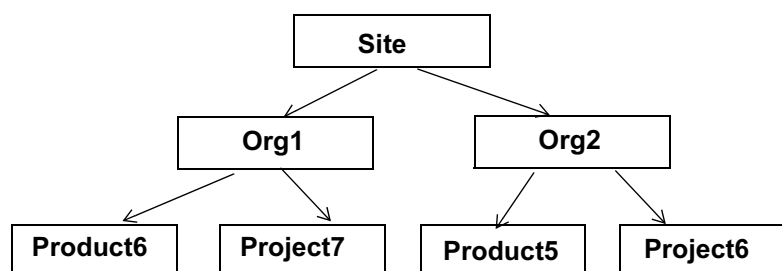


## Container Hierarchy for Integral Windchill Solutions

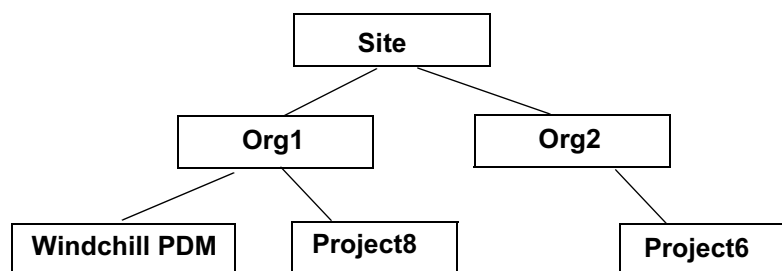
The following solutions can be installed in the same codebase:

- Windchill PDMLink and Windchill ProjectLink
- Windchill Foundation & PDM and Windchill ProjectLink

The following diagram shows the integral Windchill PDMLink and Windchill ProjectLink container hierarchy where there are two organizations (Org1 and Org2) and administrators have created a product and project container under each organization:



The following diagram shows the integral Windchill Foundation & PDM and Windchill ProjectLink container hierarchy where there are two organizations (Org1 and Org2) and Windchill ProjectLink administrators have created two project containers, one under each organization:



## Managing Access to Data through Access Control Rules

In each container, a set of access control rules can be set when the container is created. These rules can be defined programatically as well as defined in the template that is used to create the container. Additionally, an administrator can define access control rules for the data that is in the container and in its child containers, thereby further controlling the access to data. Generally, each access control policy rule does the following:

- Identifies a type of data stored in a specific administrative area (domain) to which access permissions are applied.



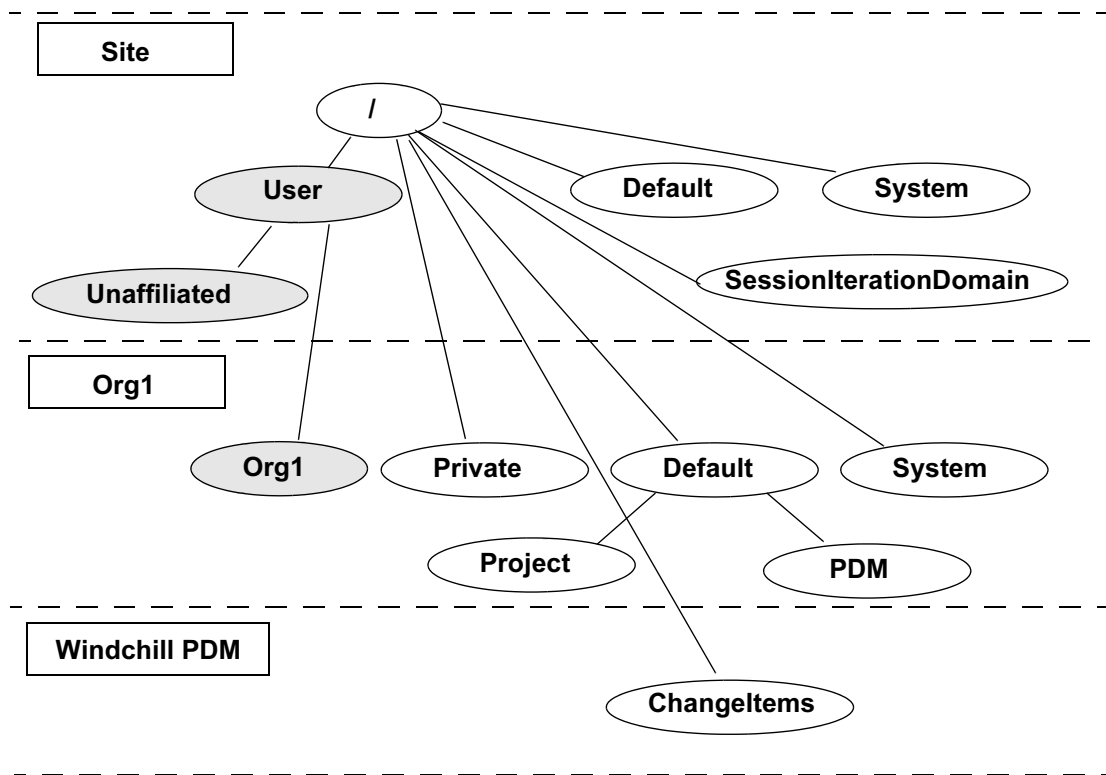
- Identifies the specific state (or all states) of an object to which access permissions are applied.
- Identifies users (either as individual users, groups of users, or entire organizations) for whom access is either granted or denied.
- Specifies the specific access permissions (such as read, write, and modify) given to the users for the data type in the specific domain.

Access control policy rules have a hierarchy based on the domain hierarchy. Dependant domains inherit rules from ancestor domains and cannot override those rules.

The following sections introduce the domains and container structure that are available in each Windchill solution.

## Windchill Foundation & PDM Library Container

The Windchill Foundation & PDM library container is loaded as part of the installation. It inherits from the organization and Site containers that are also loaded. The initial set of domains setup with an organization named Org1 are shown in the following diagram:



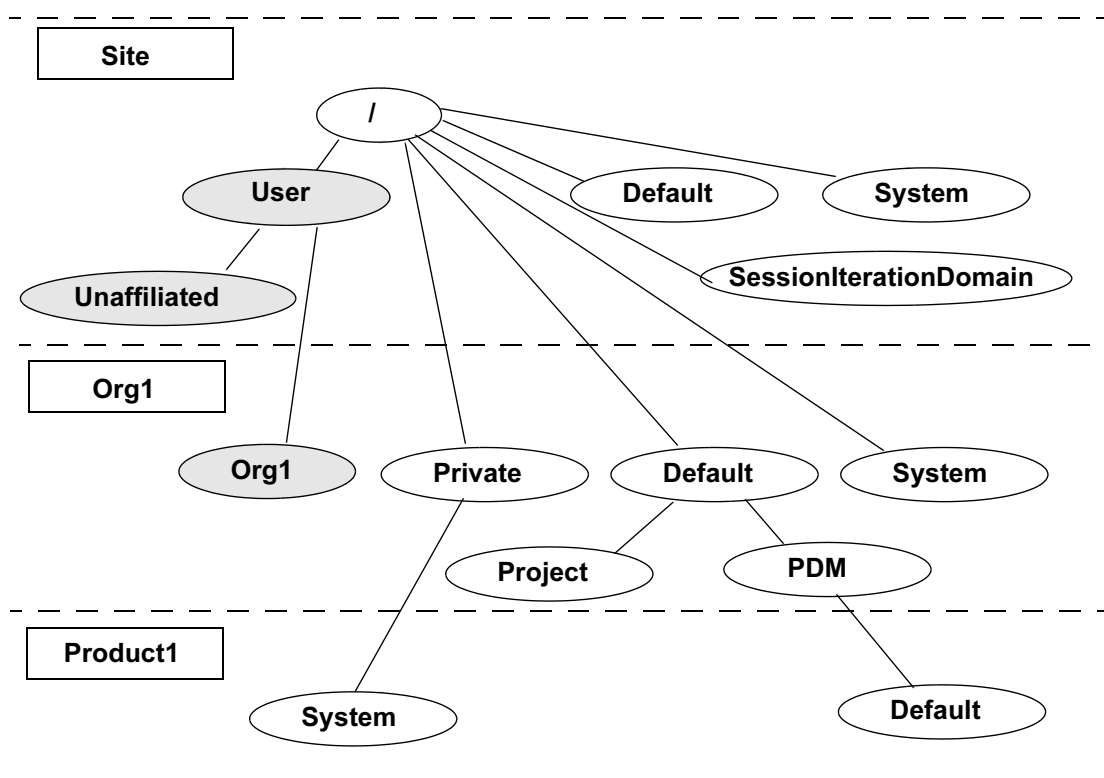
**Note:** If you load demo data, there will be additional domains created.

Use the Principal Administrator to manage users, groups, and organizations. For details on managing users, groups, and organizations, see [Administering Principals](#).

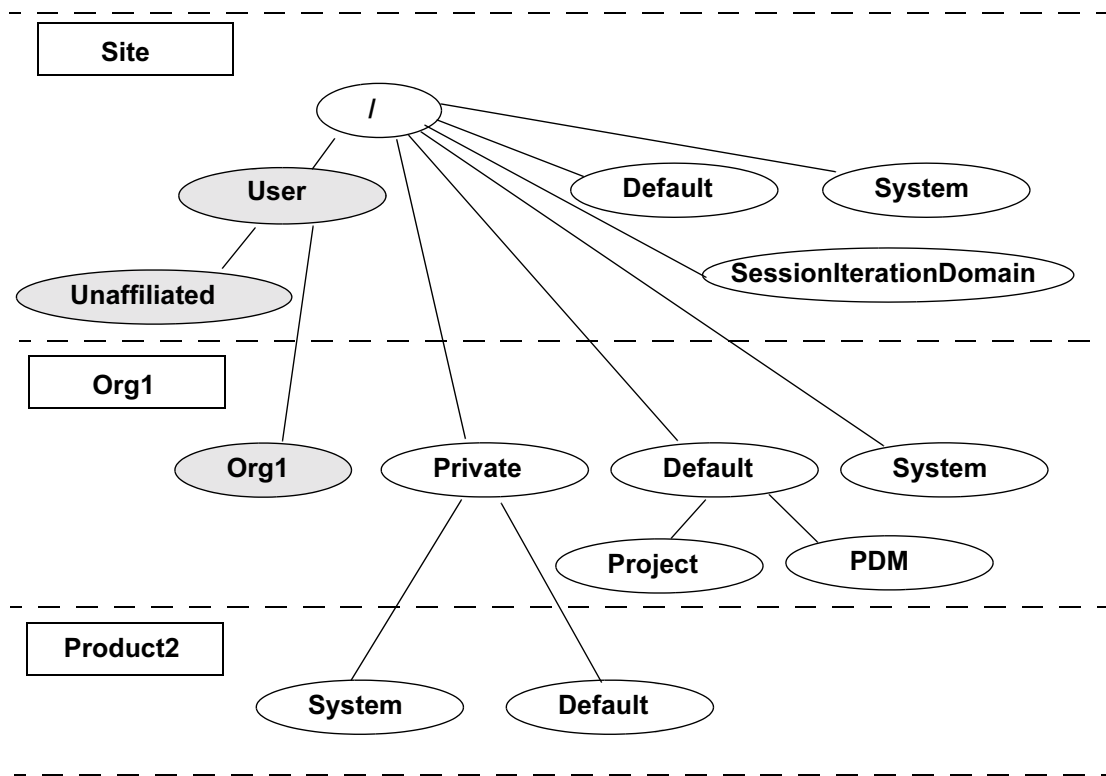
Use the Policy Administrator to create domains and corresponding policy rules. For details on domain inheritance and setting policy rules, see [Administering Domains and Policies](#) in the Administering Containers chapter.

## Windchill PDMLink Application Containers

Each Windchill PDMLink application container that is created usually includes a Default domain that inherits from the /Default/PDM domain in the organization context and a System domain that inherits from the /Private domain in the organization context, as shown in the following diagram. (Depending on the template you use, other domains can also be created in the product or library container.) The diagram shows the Site, Org1, and Product1 containers and domains:



However, private application containers can also be created that do not inherit from the /Default/PDM domain in the organization container. The resulting hierarchy for a Windchill PDMLink private Product2 container is as follows:



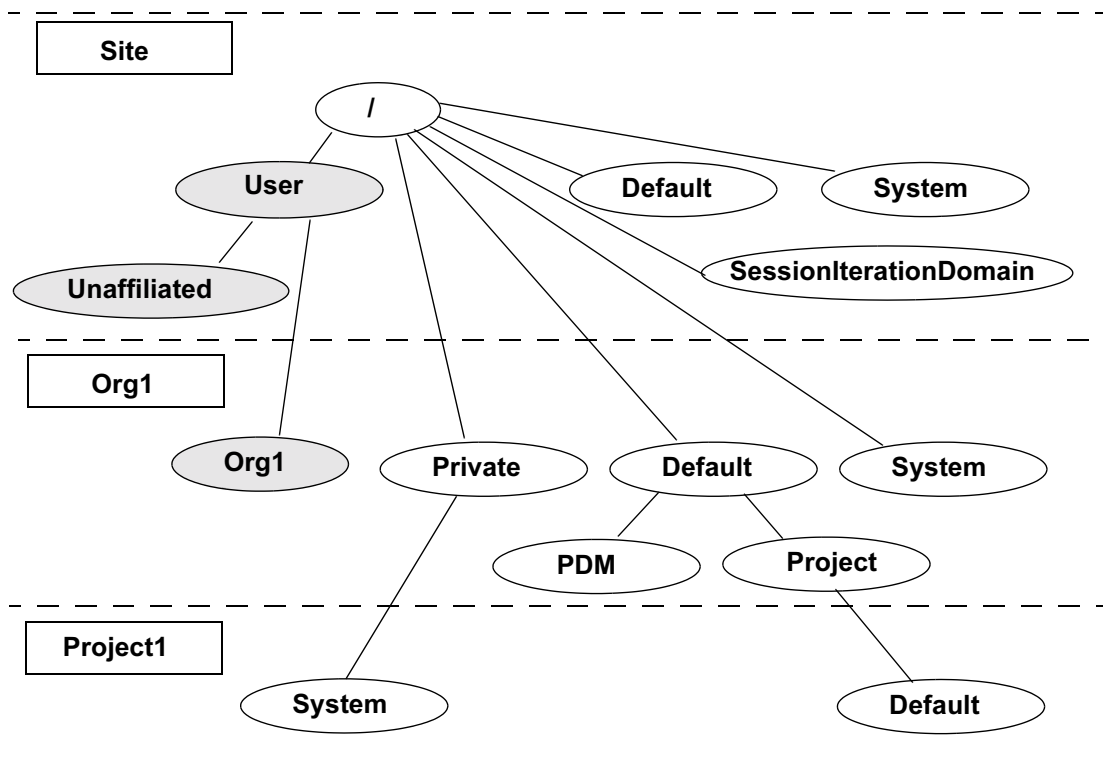
The **Team** link from the **Product** or **Library** tab in Windchill PDMLink allows you to set up the role and role memberships that can be used as the groups against which the access control rules are set, as described in [Managing Access to Data through Team Memberships](#).

Use the Principal Administrator to update users that have changed in your user directory service or create and update groups at the organization level that can be used as team members. For additional information about managing users, groups, and organizations, see [Administering Principals](#).

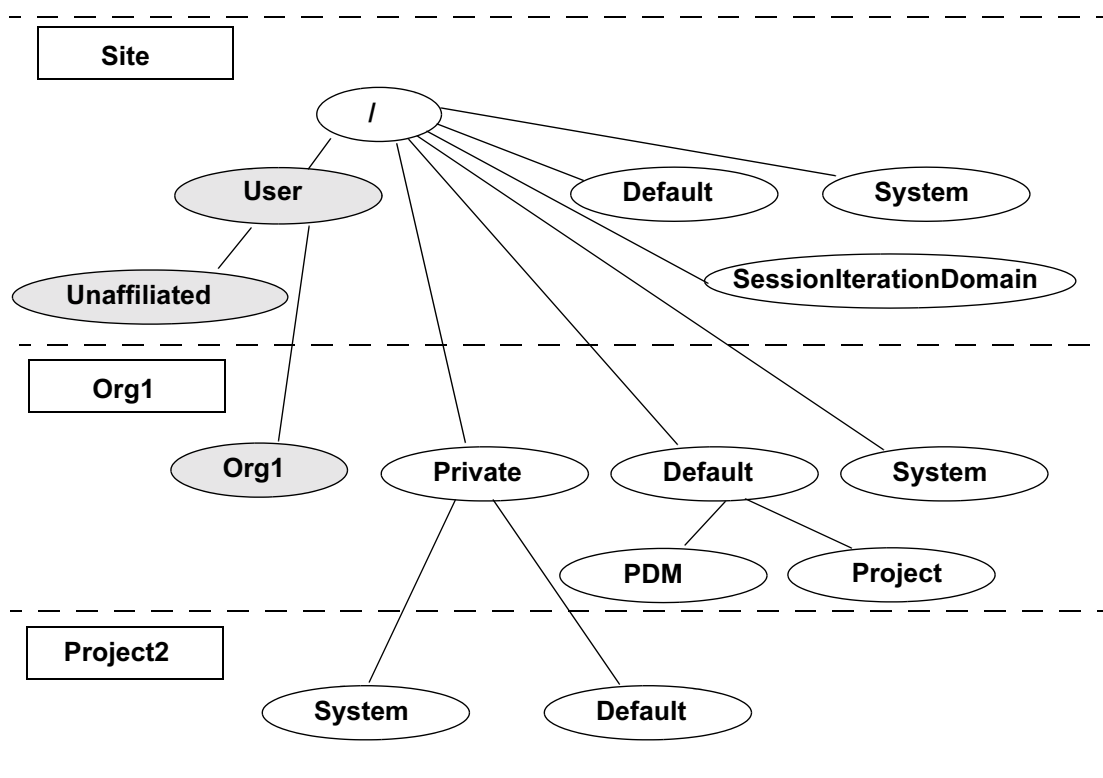
Use the Policy Administrator to create policy rules. For details on domain inheritance and setting policy rules, see [Administering Domains and Policies](#) in the Administering Containers chapter.

## Windchill ProjectLink Project Containers

Each Windchill ProjectLink project container that is created usually includes a default Domain that inherits from the /Default/Project domain in the organization context and a System domain that inherits from the /Private domain in the organization context, as shown in the following diagram. (Depending on the template you use, other domains can also be created in the organization and project containers.) The diagram shows the Site, Org1, and Project1 containers and domains:



However, private project containers can also be created that do not inherit from the /Default/Project domain in the organization container. The resulting hierarchy for a Windchill ProjectLink private Project2 container is as follows:



The **Team** link from the **Project** tab in Windchill ProjectLink allows you to set up the role and role memberships that can be used as the groups against which the access control rules are set, as described in [Managing Access to Data through Team Memberships](#).

Use the Principal Administrator to update users that have changed in your user directory service or create and update groups at the organization level that can be used as team members. For additional information about managing users, groups, and organizations, see [Administering Principals](#).

Use the Policy Administrator to create policy rules. For details on domain inheritance and setting policy rules, see [Administering Domains and Policies](#) in the Administering Containers chapter.

## Managing Access to Data through Team Memberships

In Windchill PDMLink and Windchill ProjectLink, another aspect of managing user access to data can be found in managing who becomes a member of an application container. The context team associated with a product, library, or project establishes which users are members of a specific application container. The team members are added to the team according to their role in the product, library, or project. The initial roles that are available in a specific application container are determined when the container is created; however, additional roles can be added.

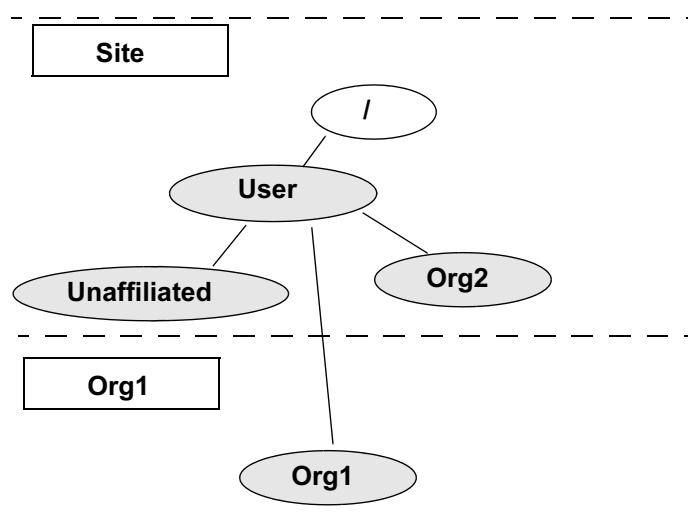
For each role used in a context team there is a corresponding internal group created that administrators can use to create access control rules for the members assigned to the role. The **Team** link from the **Product**, **Library**, or **Project** tab in your Windchill solution provides access to the interface for managing teams. Use the Policy Administrator to create access control rules.

Your Windchill solution also uses roles and corresponding users defined in life cycle templates and team templates (if they are defined for an object). For additional information about teams, see [Administering Teams and Roles](#).

For additional information about access control rules, see [Administering Domains and Policies](#) in the Administering Containers chapter.

## Managing Users

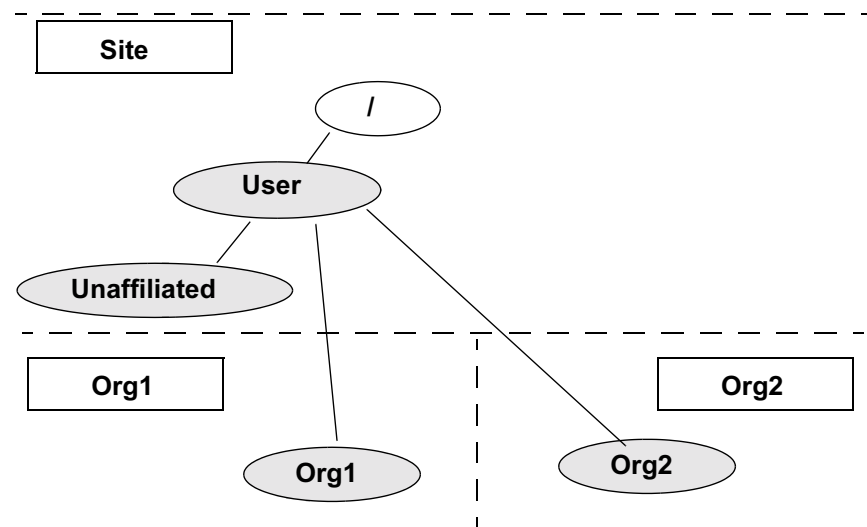
As mentioned in earlier sections, the default domains associated with users are the User domain in the Site container or one of its child domains. For example, assume that the Org1 organization container has been created and users from both the Org1 and Org2 organizations and users that have no organization affiliation (the organization attribute is not set for the user) have accessed your Windchill solution, then the following domains are automatically created:



In this example the user and domain associations are as follows:

- Users from the Org1 organization (their organization attribute is set to Org1) are by default associated with the Org1 domain. This domain is in the Org1 container.
- Users from the Org2 organization (their organization attribute is set to Org2) are by default associated with the Org2 domain. This domain is in the Site container.
- Users who have no organization affiliation are by default associated with the Unaffiliated domain. This domain is in the Site container.

In the previous example, assume that the Org2 organization container is now created using either Windchill PDMLink or Windchill ProjectLink. Then the Org2 domain moves from the Site container to the Org2 container, as show in the following diagram:



By using the default domains for users, users are automatically grouped by organization and access policy rules for each organization are initially set using the organization context template used to create the organization. Rules for users not affiliated with any organization can be set using the Unaffiliated domain.

When your Windchill solution creates user objects, a personal cabinet is also created for each user. By default, the personal cabinet for a user is put in the same domain as the user. Using this approach allows the access control rules for personal cabinets to be in the same domain as the access control rules for the users.

In the previous examples, the organizations have short names. If the organization names you are using are longer than 193 characters, then the names are truncated when creating corresponding domain names. For more information, see [Creating Domains](#) in the next chapter.

## Managing Data

The Windchill installation establishes a set of business object types that are available in the Site container and then can be inherited by organization containers and then by application containers. The out-of-the-box business object types are described in the following table:

Object Type	Description
wt.doc.WTDocument	General documents such as text files or Microsoft Word documents
wt.change2.WTAnalysisActivity	Change data that assigns an analysis task to a user
wt.change2.WTChangeActivity2	Change data that assigns product development work to a user
wt.change2.WTChangeInvestigation	Change data that collects information about the root cause of a product problem
wt.change2.WTChangeIssue	Change data that reports a potential product problem
wt.change2.WTChangeOrder2	Change data that collects all tasks required to implement a product change
wt.change2.WTChangeProposal	Change data that proposes a solution to a product problem
wt.change2.WTChangeRequest2	Change data that collects all change impact data required for a decision
wt.epm.EPMDocument	<p>This type is the parent type of soft types for the following business objects:</p> <ul style="list-style-type: none"><li>• CAD documents (type: DefaultEPMDocument)</li><li>• dynamic documents (type: DynamicDocument)</li><li>• burst configuration files (type: BurstConfiguration)</li></ul>
wt.part.WTPart	A database object that represents a component of assembly in a product structure



Object Type	Description
wt.part.WTProductConfiguration	A snapshot of a product structure that captures the bill of materials to which a product is assembled
wt.part.WTProductInstance2	A serialized copy of a product built according to a specific configuration
wt.vc.baseline.ManagedBaseline	A collection of specific part and document iterations

**Note:** Not all Windchill solutions use all business object types and many additional object types are used by your Windchill solution to help manage administrative processes, such as updating user preferences, life cycles, and workflows. Additional information about the use of specific object types can be found throughout this guide.

## Data Types

All data stored in a Windchill solution are stored as objects of specific types. The type is identified when the object is created or imported. By typing data, you establish patterns for handling the data within the Windchill solution. For example, you can decide if part data is numbered automatically or manually and decide who has access to part data. Separate decisions can be made for each type of data by setting object initialization rules. A set of these rules is established when each container is created through the context template that is used. Additional object initialization rules can be set using the Object Initialization Rules administrator, as follows:

- In Windchill PDMLink and Windchill ProjectLink, you access the Object Initialization Rules administrator from the **Utilities** page of the context where you want the rule stored.
- In Windchill Foundation & PDM, the administrator is on the **Business Administration** page.

## Soft Types

In addition to the modeled object types that are provided out of the box, object types can have subtypes (known as *soft types*). Windchill ProjectLink and Windchill PDMLink provide some document soft types as part of its installation. Additionally, Windchill PDMLink provides soft types for dynamic documents and bursting configuration files that are used when authoring document content through the Windchill connection to Arbortext Editor.

If your site needs additional types, you can create specific soft types in Windchill PDMLink and Windchill ProjectLink from within the Site or an organization

context using the **Types** link. In Windchill Foundation & PDM, the Type Manager is available from the **Business Administration** page.

## Visualization Data

Windchill Visualization Services integrates Windchill with ProductView, a PTC data visualization tool that allows you to view, annotate, analyze, measure, and animate CAD documents and parts. You may also view, mark up, and print the content of documents in ProductView. All information viewed or annotated in ProductView is saved and stored back into the Windchill database.

ProductView supports watermarking of 3D, drawings, images, and documents. Watermarks are defined in INI files created and edited using the ProductView watermark editor. The administrator that manages watermarks manually transfers the INI files from the watermarks directory into the Windchill server.

Windchill Visualization Services can also be used to integrate Windchill with the Arbortext Publishing Engine for creating representations from dynamic documents. Supported formats include PDF and HTML. This is done through the WVS publisher.

For information on Windchill Visualization Services, see the *Windchill Installation and Configuration Guide -Visualization Services*.

For more information about administering visualization data, see [Administering Visualization Services](#).

## CAD Data

Data files created by CAD applications and saved to Windchill using a workgroup manager or Pro/ENGINEER Wildfire are managed in Windchill as CAD documents.

CAD documents are associated with a soft type of the wt.epm.EPMDocument modeled object type and can be used to manage many different types of CAD files including parts, assemblies, drawings, layouts, and manufacturing files. In addition, a CAD document can have secondary content. The secondary content is typically used for additional supporting information or derived files such as STEP, IGES, JPEG, and others.

The Workgroup Manager CAD Document soft type defined as *<prefix>.DefaultEPMDocument* is established during the installation process, where *<prefix>* is generated by reversing the internet domain specified as part of the organization properties during the installation. For example, if the internet domain specified is acme.com, then the *<prefix>* used in the soft type name is com.acme.

Administrators can create and update CAD document templates. These templates can be used to create CAD documents.

## Dynamic Document Data

Windchill PDMLink supports data files that are created by the Arbortext Editor. These files are saved to Windchill using the Windchill connection for the editor and are managed in Windchill PDMLink as dynamic documents. The data files can then be published using the WVS publisher if the publisher has been configured for the Arbortext Authoring Application.

The Arbortext Editor and the Arbortext Publishing Engine are optional Windchill products that can be purchased through PTC.

Dynamic documents are associated with the `<prefix>.DynamicDocument` soft type or are possibly associated with a soft type that is a subtype of `<prefix>.DynamicDocument` (where `<prefix>` is generated by reversing the internet domain specified as part of the organization properties during the installation). For example, if the internet domain specified is `acme.com`, then the soft type for dynamic documents is named `com.acme.DynamicDocument`.

Using additional soft types helps categorize the types of dynamic documents and is often required when deploying a complex content authoring system that connects to the Arbortext Editor.

Understanding how Windchill PDMLink and Arbortext Editor work together and the basics on setting up your Windchill PDMLink system to work with Arbortext Editor are covered in the [Getting Started with Windchill PDMLink and Arbortext Editor](#) guide.

For information on installing Windchill PDMLink and the Arbortext products, see the *Windchill Maintenance Installation and Configuration Guide*.

For general guidelines on creating product and library containers to use with dynamic documents, see [Configuring Product or Library Containers for Dynamic Documents](#).

For information on creating soft type and soft attributes for use with Arbortext Editor and the WVS publisher, see [Managing Types and Attributes for Dynamic Documents](#).

For information on setting up the WVS publisher for publishing dynamic documents, see the *Windchill Installation and Configuration Guide - Visualization Services*.

## Document Data

General documents (such as Microsoft Word files and text files) stored in the Windchill database have the `wt.doc.WTDocument` object type or are possibly associated with a soft type that is a subtype of `wt.doc.WTDocument`. Using soft types helps categorize the types of documents.

Administrators can create and update document templates. These templates can be used to create documents.

## Part Data

Parts stored in the Windchill database have the wt.part.WTPart object type.

Parts created in Windchill Foundation & PDM and Windchill PDMLink can be associated with a view. A view identifies what the part is used for. The Windchill installation provides two out-of-the-box views: design and manufacturing. Before allowing users to create parts, you can rename these views and add other views that make sense at your site by using the LoadViews command line utility. For more information, see [Administering Views and View Associations](#).

## Audit Reports

Auditing reports are designed to record and report user and system events for auditing and traceability purposes.

## Managing Windchill Processes

Windchill provides you with the following types of Windchill data processes that can be used with the business objects stored in Windchill:

- A life cycle process defines a set of phases and gates that manage the life of an object as it progresses from conception to obsolescence.
- A workflow process defines rules which allow workflow participants to view and pass along information, tasks, and documents.
- A change process establishes how to get changes made to parts and many other data types including documents, document soft types, and product instances.

These three processes work together to help you manage data. Workflows are often used to drive the life cycle. That is, the workflow process is used to transition from one life cycle state to another. In most cases, a workflow process is initiated from a life cycle. In any case, life cycle-managed objects obtain a life cycle when they are created. In obtaining a life cycle, an object can have a workflow process instance created as well. Change objects are life cycle-managed objects. Each change object starts a change process when it obtains its life cycle.

Use the Life Cycle Administrator to manage the life cycle templates that can be used. For details on managing life cycle templates, see [Administering Life Cycles](#).

Use the Workflow Administrator to manage the workflow templates that can be used. For details on managing workflow templates, see [Administering Workflow Processes](#).

## Managing User Collaboration

User collaboration can be done using a specific Windchill solution and other PTC or third party products or using multiple co-installed Windchill solutions.

Windchill provides the following tools for collaboration:

- Windchill projects that are created using Windchill ProjectLink provide participants with a place in which they can share information. This information can include data that resides in either a Windchill Foundation & PDM or Windchill PDMLink solution.
- Pro/ENGINEER Wildfire provides users with the ability to share CAD drawings and other design-related information. For administration information, see the Administering Pro/ENGINEER Wildfire chapter in the *Windchill System Administrator's Guide*.

## Additional Administrative Groups

The following additional administrative groups are automatically created for all Windchill solutions to help define users for specific administrative activities in your solution:

Attribute Administrators  
LifeCycleAdministrators  
Type Administrators  
WorkflowAdministrators

For example, those users in the Attribute Administrators group can manage the metadata for attributes. Those users in the LifeCycleAdministrators group become participants in the Default life cycle template, when that template is used.

Additionally, the Windchill Foundation & PDM installation creates the following administration groups:

Classification Administrators  
Navigation Administrators

And the Windchill PDMLink installation creates the following administration group:

Business Entity Authors

The Business Entity Authors group is an option group that can be used when working with the optional Windchill PartsLink product.

If Windchill Supplier Management is installed, the installation creates the following administration group:

Supplier Administrators

For information on Supplier Management activities, see the *Windchill Supplier Management Administrator's and User's Guide*.

By using the Principal Administrator, you can add users to any of the administration groups.

The Windchill PDMLink and Windchill ProjectLink solutions provide additional types of administrators:

- An organization administrator manages a specific organization.
- A product manager manages a specific product.
- A library manager manages a specific library.
- A project manager manages a specific project.

For additional information about administrators, see [Establishing Administrators](#).

## Post-Installation Activities

Before allowing users to access the Windchill solution, be sure to do the following activities:

- Complete the activities described in the [Getting Started](#) chapter.
- For Windchill PDMLink and Windchill ProjectLink, determine if there are additional organization containers that you want to create and create them. See [Administering Organizations](#).
- Determine if additional organization principals are needed and whether or not to allow these organizations to own parts and documents.

By default, all parts and documents are owned by the organization from which they are created. For information on how to set up the ability to choose which organization owns a part or document, see Administering Organizations in the *Windchill System Administrator's Guide*.

- Determine whether you want audit reports enabled. See [Administering Audit Reports](#).

## About the windchill Command

PTC has provided a command, `windchill`, to invoke Windchill actions. For example, the command can be used to stop and start Windchill, check the status of the Windchill server, and create a new shell and set the environment variables. It can also be used as a Java wrapper. In that regard, it can accept a Class file as an argument, just like Java, and execute it without a predefined environment (Windchill classes in CLASSPATH, Java in PATH, and so on).

The `windchill` command should be used to execute any server-side Windchill Java code. This will ensure that the environment that the command is executed in is properly setup. The environment that actions are executed within, including the `windchill` shell action, is defined by the `wt.env` properties in the `wt.properties` file. For example, the `wt.env.CLASSPATH` property will set the CLASSPATH environment variable for the action that is being invoked.

The `windchill` command is a Perl script that has also been compiled into a Windows binary executable. For UNIX systems, Perl 5.0 or greater must be installed. The `windchill` script assumes that Perl is installed in the standard install location of `/usr/bin/perl`. If Perl is not installed at this location, you can either create a symbolic link (recommended method) to the Perl install location or edit the `windchill` script to reference the Perl install location. To modify the `windchill` script, edit the `<Windchill>/bin/windchill` file. Locate the `#!` entry (for example, `#!/usr/bin/perl -w`) and change the Perl directory to the location where Perl is installed.

The `windchill` command is located in the `<Windchill>/bin` directory. If you receive a command not found message when you execute the `windchill` command, add the `<Windchill>/bin` directory to your PATH environment variable. The syntax of the `windchill` command is:

```
windchill [args] action
```

You can display the help for the `windchill` command by executing `windchill` with the `-h` argument or with no argument.

The following tables list some of the arguments and actions applicable to the `windchill` command. To see a complete list of the arguments, use the report generated from the help (argument).

### windchill Arguments:

Arguments (optional)	Description
<code>-h, --help</code>	Displays help and exits.
<code>-v, --[no]verbose</code>	Explains what is being done when a command is executed. Default is <code>noverbose</code> .

Arguments (optional)	Description
-w, --wthome=DIR	<p>Sets the Windchill home directory. Default is the parent directory containing the windchill script.</p> <p><b>Note:</b> On UNIX systems where you have multiple instances of Windchill installed under the same user account, settings made to WT_HOME and SQLPATH environment variables by using this -w option are overridden by any settings to these same variables in the user's .cshrc, .login, and .profile shell initialization files.</p>
--java=JAVA_EXE	<p>The Java executable. Default is the wt.java.cmd variable value specified in the \$WT_HOME/code-base/wt.properties file.</p>
-cp, --classpath=PATH	<p>Java classpath. Default is the wt.java.classpath variable value specified in the \$WT_HOME/code-base/wt.properties file.</p>
--javaargs=JAVAARGS	<p>Java command line arguments.</p>



## windchill Actions

Action	Description
shell	Sets up a Windchill environment in a new instance of the currently running shell.
start	Starts the Windchill server.
stop	Stops the Windchill server.
status	Retrieves the status of the Windchill server.
version	Displays the Windchill installation version.
properties <resource> [,...][?key[&key2]...]	<p>Displays the properties as seen by Windchill for the given resource with substitution and other actions executed. It can be limited to a given set of keys.</p> <p>For example:</p> <p>windchill properties wt.properties — lists all wt.properties</p> <p>windchill properties wt.properties?wt.server.codebase — lists server codebase</p> <p>windchill properties wt.properties?wt.env.* — lists all the environment variables use by windchill shell</p> <p>windchill properties — with no arguments generates the help report</p>
CLASS [CLASS_ARGS]	<p>Run a Windchill class with optional class arguments. For example:</p> <p>windchill wt.load.Developer -UAOps</p>

## About the windchill shell

The windchill shell brings up a new command shell, from the parent shell that is setup for the Windchill environment. This includes setting all environment variables defined in wt.env property in the wt.properties file.

To execute the windchill shell, at the command prompt enter the following command:

```
windchill shell
```

When you are finished using the windchill shell, you can exit the shell and return to the parent shell.

PTC recommends running all server-side Windchill applications, tools, and utilities from the windchill shell. Also, you can use the windchill shell to set up your development environment to use javac or Java directly.

## About the xconfmanager Utility

The xconfmanager is a command-line utility that is used to add, remove, and modify the properties in the Windchill property files. You should use the xconfmanager (or the System Configurator) to manipulate properties; you should not manually edit property files.

There are property files that should not be modified using the xconfmanager. The following registry files are managed by Windchill Information Modeler and they also should not be edited manually or using the xconfmanager:

```
associationRegistry.properties
classRegistry.properties
descendentRegistry.properties
modelRegistry.properties
moduleRegistry.properties
moduleDir.properties
debug.properties
```

The xconfmanager utility saves your changes in the site.xconf file and provides an option to generate updated property files using those changes. The site.xconf file contains changes made to Windchill property files, starting with installation and continuing with each use of the xconfmanager utility or the System Configurator. The xconfmanager utility is located in the <Windchill>/bin directory.

This chapter describes only the information and instructions necessary to modify specific Windchill properties. A full description of the xconfmanager utility and management of the Windchill property files is documented in the *Windchill System Administrator's Guide* in the Administering Runtime Services chapter.

Anyone with write access to the XCONF files and the property files under the Windchill installation directory can successfully run the xconfmanager utility. The xconfmanager is executed from the command line from within a windchill shell. See the [About the windchill Command](#) chapter for more information about the windchill shell.

The syntax of xconfmanager command with only commonly used parameters is as follows:

```
xconfmanager {-h} {-r <product_root>} {-s <property_pair>}
{-t <property_file>} {--add <property_pair>}
{--remove <property_pair>} {--reset <property_names>}
{--undefine <property_names>} {-d <property_names>} {-p}
```

For the purposes of modifying Windchill properties, you will primarily use the -s, -t, and -p parameters as follows:

- Use the -s (--set) parameter to identify the relevant property and specify the new property value. See the [Formatting Property Value Guidelines](#) section (below) for information about formatting the <property\_pair> value.
- Use the -t (--targetfile) parameter to specify the directory location of the property file. If the file name or path contains spaces, you must enclose the

*<property\_file>* value in double quotes (" "). It is recommended to use a fully qualified file name to ensure an accurate reference to the file is made.

- Use the **-p** (**--propagate**) parameter to propagate the changes made to the XCONF files into the property files being modified in order to keep the XCONF and the property files in sync with one another.
- Use the **-h** (**--help**) parameter to view the help for xconfmanager. The help describes all xconfmanager parameters.

Additionally, you can add and remove property values from properties that are declared as a multi-valued properties using the following parameters:

- Use the **--add** parameter to add the value specified at the end of the set of ordered values already defined in the property.
- Use the **--remove** parameter to remove the value specified from the set of ordered values already defined in the property.

**Tip:** If you are unsure as to whether a property is multi-valued, you can display the current set of values using the **-d** parameter. The output from this parameter lists the multivalue separator when the property is multi-valued.

Some examples of using the xconfmanager utility are as follows:

- xconfmanager is run from the windchill shell. To open a windchill shell, execute the following command at a command prompt:

```
windchill shell
```

- To display xconfmanager help, execute the following command from the windchill shell:

```
xconfmanager -h
```

- To display the current settings for a property, execute the following command from the windchill shell:

```
xconfmanager -d <property_names>
```

*<property\_names>* is a comma-separated list of property names. This means that you can display the current settings for multiple properties by executing one command.

- To change a property value, execute the following command from the windchill shell:

```
xconfmanager -s <property_name>=<property_value>  
-t <property_file> -p
```

**Tip:** Use the fully qualified name of the property file to ensure an accurate reference. If you are sure that there is only one property file that is known to xconfmanager containing the property, you can omit the **-t** parameter. When setting a value for a new property not in a property file, you must include the **-t** parameter to name the property file to which the property is added.

- To add a new classpath entry to the Windchill classpath specified in the `wt.java.classpath` property, execute the following command from the windchill shell:

```
xconfmanager --add wt.java.classpath=d:\MyLibraries\somelibrary.jar -p
```

The value `d:\MyLibraries\somelibrary.jar` will be added to the end of the ordered set. You do not have to specify the delimiter `$(path.sep)` as this will be added to the property value automatically by the `xconfmanager`.

**Tip:** The previous example command did not include the target file in the `-t` parameter since the property is known to be in only `wt.properties`.

## Formatting Property Value Guidelines

The following guidelines will help ensure that you set properties correctly on the command line when using `xconfmanager`:

- To specify a property whose value contains characters that might be interpreted by your shell (such as spaces and special characters), escape them using the appropriate technique for the shell you are using.

On a Windows system, you can include spaces in a value by enclosing the argument with double quotes or you can escape the space character with `^`. For example, use either of the following:

```
-s "wt.inf.container.SiteOrganization.name=ACME Corporation"
-s wt.inf.container.SiteOrganization.name=ACME^ Corporation
```

On a UNIX system, you can use double quotes or you can escape the space character with `\`. For example, use either of the following:

```
-s "wt.inf.container.SiteOrganization.name=ACME Corporation"
-s wt.inf.container.SiteOrganization.name=ACME\ Corporation
```

- In many UNIX shells, the use of a backward slash (`\`) escapes the following character as a literal. In most cases, using forward slashes (`/`) in file paths is a simple way to specify a path without having to know the intricacies of your shell's command line escaping rules.
- On UNIX, dollar signs are usually interpreted by shells as variable prefixes. To set a property value that has a dollar symbol in it, use single quotes around the argument so that the shell does not interpret it or use backslash to escape the dollar symbols. For example, use either of the following:

```
-s 'wt.homepage.jsp=$(wt.server.codebase)/wtcore/jsp/wt/portal/index.jsp'
```

or

```
-s wt.homepage.jsp=
\$(wt.server.codebase)/wtcore/jsp/wt/portal/index.jsp
```

Other than escaping arguments so that the command-line shell does not misinterpret them, you should not need to escape other values to be compatible with XML or property file syntaxes. The xconfmanager escapes property names and values automatically if necessary.



# 3

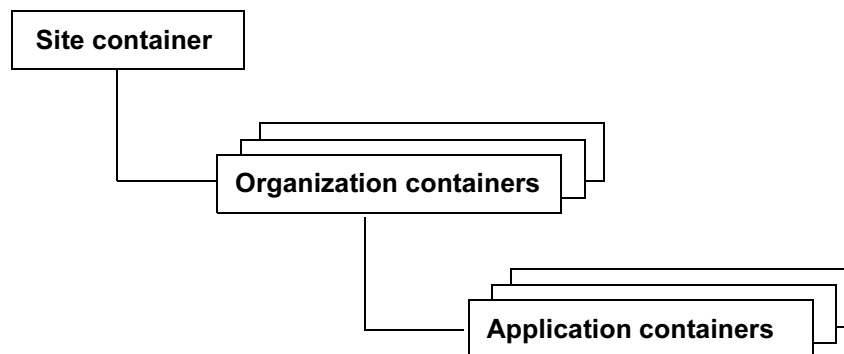
## Administering Containers

This chapter provides the overall details relating to working with containers. Later chapters assume that you have read the information in this chapter.

Topic	Page
Distributed and Hierarchical Administration .....	3-2
Container Administrative Items .....	3-4
Creating Containers .....	3-21
Administering Domains and Policies .....	3-28
Administering Object Initialization Rules.....	3-40
Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects.....	3-48
Administering the Versioning of Parts, Documents, and CAD Documents .....	3-50
Searching for Principals .....	3-63

## Distributed and Hierarchical Administration

Windchill containers provide the framework for collecting and finding related information. The set of containers in your Windchill solution have a hierarchical relationship. The following depicts the basic container hierarchy:



The Site container can have one or more child organization containers. An organization container can have one or more child application containers.

Application containers include:

- Projects (provided by Windchill ProjectLink)
- Products (provided by Windchill PDMLink)
- Libraries (provided by Windchill PDMLink - formerly known as repositories)
- The Windchill PDM library (provided by Windchill Foundation & PDM)

Data, such as template files, can be distributed among the containers. For example, you can define general document templates, such as those used for presentations or memos, at the top level of the hierarchy (in the Site container). The document templates are available to all containers. Then you can define progressively more specific templates at each layer in the hierarchy, such as in an organization container or a library container. In a child container, you can also define templates with the same name as those templates in an ancestor container so that the templates in the child container can override and be used in place of templates in an ancestor container.

With distributed administration, container administrators are responsible for their own administrative tasks. So, for example, each project, product, and library can have its own administrators (called project, product, and library managers). To support distributed administration, the administrative clients are container aware. For example, opening the Policy Administrator in the context of a library initially displays the domains that are in the library context and the domains that are ancestors of the library context domains. Making clients container aware allows for the delegation of administrative duties to users who are recognized as container administrators.



With hierarchical administration, containers inherit administrative items from parent containers (or, in the case of policies, from ancestor domains). Administration performed at the level of an ancestor container is applicable to all of its descendent containers. Except for policies, child containers can choose to override the administrative items of its parent containers. For policies, rules from all ancestor domains are merged with the rules in a current domain to form the policy for the current domain.

In general, always think of performing administration duties in the context of a container, as follows:

- Site administrators can create, modify, delete, and view administrative items in the Site container.
- Organization administrators can create, modify, delete, and view administrative items in the given organization container. They can view and override administrative items defined in the Site container.
- Application container administrators can create, modify, delete, and view administrative items in the given application container. They can view and override administrative items defined in the parent organization container and the Site container.
- Site and organization administrators can administer child containers; however, they do so by going into the context of the child container and performing their administrative duties there.

**Note:** Windchill users generally perform their activities in the context of a container. Most often, these activities occur in application containers rather than in an organization or Site container.

Non-administrative Windchill users generally do not create, modify, or delete administrative items; however, they have visibility to, and are affected by, the following administrative items:

- The administrative items that are defined in a given application container.
- The administrative items that are defined in the parent organization and Site containers, but not overridden in the application container.

## Container Administrative Items

Each container can be populated with the following types of administrative items:

- Container configuration
- Container structure
- Container participation
- Container policies
- Container data types and attributes
- Container templates, including process templates and data templates
- Container object initialization rules

After you have created a container, you can update many of the administrative items that are associated with the container. The containers you can update include the Site container, as well as organization and application containers.

Use the administrative utilities and specific user actions to update the administrative items in a container. These utilities and actions are usually available to the administrator of the container. For the details on Windchill administrators, see [Additional Administrative Groups](#) in the Administration Overview chapter.

Each type of container has a slightly different set of administrative items that can be updated. For example, you can update the set of life cycle and workflow templates provided in organization, product, and library containers. Also, you can only update the set of product, library, and project templates that are provided in an organization or the Site container. This is because product, library, and project templates are not included in application containers.

The following sections provide descriptions of administrative items, information on what is installed in the Site container for each item, and how to update the items.

## Container Configuration

Configuration items identify the type of container and other miscellaneous information about the container.

There are three general types of containers:

- Site – The Site container is the top-level container. There can be only one Site container.
- Organization – Organization containers are always children of the Site container. There is always at least one organization container required to have an operational Windchill solution.

**Note:** Although Windchill Foundation & PDM has an organization container (created as part of the installation), the container is not a main part of the solution architecture. It is provided for compatibility with the other solutions.

- Application – Application containers are always children of an organization container. There are three types of application containers:
  - Product (provided by Windchill PDMLink)
  - Library (provided by Windchill Foundation & PDM and Windchill PDMLink)
  - Project (provided by Windchill ProjectLink)

The container configuration can include the following additional items:

- For Windchill PDMLink and Windchill ProjectLink application containers, you decide whether the container is public or private. Private containers, by default, inherit access control rules from the /Private domain of the organization. Public containers, by default, inherit access control rules from a solution-dependent public domain within the organization container. For Windchill ProjectLink, the public domain is the /Default/Project domain. For Windchill PDMLink, the public domain is the /Default/PDM domain. This configuration allows an administrator to:
  - Create policies in the organization container /Default domain that apply to all public child containers.
  - Create policies in the /Default/Project and /Default/PDM domains that apply to solution-specific child containers.
  - Create policies in the /Private domain that apply to all private child containers.
- For project containers, you decide whether or not data can be shared to other containers.
- For organization containers in Windchill ProjectLink, you decide whether the creators group is auto-populated with members of the organization. The creators group in Windchill ProjectLink determines who can create projects.

The configuration of the container is set when the container is created based on the options chosen through the user interface or through data loading, and are not set in a template.

## Updating the Container Configuration

For Windchill PDMLink and Windchill ProjectLink, the configuration of an organization or application container is set when the container is created based on

the options chosen through the user interface. Only a few of the options can be updated:

- For an organization container, update the container using the **Update Organization** icon or the **Update** action on the organization details page. Site administrators, can navigate to the **Organizations** page from the **Site** tab and select the **Update** action for the organization they want to update.

For details on how to update and what can be updated, click the help available from the **Organization** tab.

- For an application container, update the container using the **Update** icon or the **Update** action from the container details page. First, navigate to the container by selecting the container tab and then selecting the product, library, or project you want to update.

For details on how to update and what can be updated, click the help available from the container tab.

**Note:** In Windchill Foundation & PDM, you cannot update the container configuration.

## Container Structure

Structure items identify the domains, cabinets, folders, notebook folders, forum topics, and reference folders for notebooks that are in the container and the domain inheritance scheme that is in place. Containers define a structure in which information related to a context is organized. This structure can be represented by a folder hierarchy, product structure hierarchy, collection of discussion topics, or predefined milestones in a schedule. The structure defined by the container enforces consistency and improves efficiency.

The use of the container structure is very apparent when you look at how domains can help organize rules for users. For example, the Administrators group and Administrator user are associated with the /System domain and, therefore, are segregated from other users by default. The other users are associated with child domains of the /User domain. Both the /System and /User domains are in the Site container, and the child domains of the /User domain can be in the Site container or in an organization container.

## Installed Site Container Structure

The Site container has the top-level / (root) domain, the /System, /Default, /User, and /SessionIterationDomain domains that are children of the root domain, and the /Unaffiliated domain that is a child of the /User domain. (For an explanation of these domains, see [Administering Domains and Policies](#).)

## Windchill PDMLink

For the Windchill PDMLink, there are no folders installed in the Site container.

## Windchill ProjectLink

For Windchill ProjectLink, the following folders, which are associated with /Default domain in the Site container, are installed:

- Change Log
- General
- Policies

Additionally, the following user notebook folders are installed in the Site container:

- My Hot Links
- My General Links

## Windchill Foundation & PDM

For Windchill Foundation & PDM, the following cabinets are installed in the Site container:

- Default
- System
- Type Definitions

For Windchill Foundation & PDM, the following folders are installed in the Site container:

- /System/Workflows
- /System/Reports

## Updating Container Structure

The container structure is set either by the load files used (in the case of the Site container and the Windchill Foundation & PDM organization and library containers) or by the template selected when the container is created. The template (and its underlying code) sets the domains, cabinets, folders, notebook folders, forum topics, and reference folders that are in the container and the domain inheritance scheme, which identifies parent and child domain relationships.

Although you can modify the set of domains in a container using the Policy Administrator, Windchill PDMLink and Windchill ProjectLink administrators should refrain from doing so unless the purpose of the domain is clearly defined. For example, a new domain may be needed to provide a unique set of access control policy rules for a specific folder.

Windchill PDMLink and Windchill ProjectLink solutions do not expose the creation of cabinets through their user interface. Use the default cabinets defined. For Windchill PDMLink and Windchill ProjectLink, administrators can update the set of folders available in a container by navigating to the specific product, library, project, or organization and then clicking the **Folders** link. Similarly, forum topics can be created in a product, library, or project by clicking the **Forum** link that is available from the product, library, or project.

Windchill Foundation & PDM administrators can use the Windchill Explorer to work with cabinets and folders.

## Container Participation

Participation items establish the following:

- Roles that are automatically available in a container.
- Groups automatically created in a container.

For Windchill PDMLink and Windchill ProjectLink, product, library, and project containers are associated with teams of users and groups. Any user can display all products, libraries, or projects in which the user is identified as a team member. The users and groups in a team are associated with roles that identify the responsibilities and permissions of the team members. For example, the Product Manager role (in Windchill PDMLink) establishes who is in charge of the product and the Project Manager role (in Windchill ProjectLink) establishes who is in charge of a project.

The set of roles and groups that are automatically available in a context consist of the roles and groups established in ancestor containers, as well as those defined specifically for the container.

## Installed Site Container Participation

Through the Site container, roles and groups that are useful throughout your Windchill solution are made available to the site.

### Roles

The roles that are available to the entire site are defined in the `<Windchill>\src\wt\project\RoleRB.rbin` resource bundle, which is translated to all supported languages and made available through the application clients.

For Windchill PDMLink and Windchill ProjectLink, all of these role names are available to all child organizations and then to the product, library, and project containers created. To see the roles, you can click the **Roles** link from the **Organization** tab. For products, libraries, and projects, click the **Team** link from the corresponding tab.

For Windchill Foundation & PDM, you can view the roles from the Life Cycle Administrator. Update a life cycle and then click the **Roles** tab to view the roles.

## Groups

The following group is defined in the Site container System domain:

Administrators (known as the system or site administrators)

**Note:** This group defines the administrators for your entire site as described in [Distributed and Hierarchical Administration](#).

The following groups are defined in the Site container /User/Unaffiliated domain:

Attribute Administrators  
Business Entity Authors (only Windchill PDMLink)  
Classification Administrators (only Windchill Foundation & PDM)  
LifeCycleAdministrators  
Navigation Administrators (only Windchill Foundation & PDM)  
Type Administrators  
Unrestricted Organizations  
WorkflowAdministrators

These groups are used during the normal operation of your Windchill solution and should not be removed. Domain-based access control rules are automatically loaded in the Site container System domain granting permissions to members of these groups, except the Unrestricted Organizations group. The rules for the members of the Unrestricted Organizations group are loaded in the Site container User domain.

## Updating Container Participation

For Windchill PDMLink and Windchill ProjectLink, container participation is set by the template selected when the container is created. The roles and groups defined in the template can be modified in the organization context as well as in an application container context. Those roles and groups defined in the organization are inherited by the children of the organization.

From within an organization, use the **Groups** link to add or update groups and use the **Roles** link to add or remove roles from the organization.

From within a product, library, or project, use the **Team** link to add or remove roles. From the **Team** table, you also add and remove members from the roles. For additional information, see [Administering Teams and Roles](#).

**Note:** In Windchill Foundation & PDM, you cannot update the container participation except by changing life cycle roles in a life cycle template or team template, or by setting access control rules against users, groups, and organizations.

## Container Policies

Container policy items can include the following:

- Domain-based access control rules that establish the access control against specific principal, object type, life cycle state, and domain combinations.

For example, there can be an access control rule for objects with the wt.doc.WTDocument data type in the /Default domain that gives read permission to the Engineers group.

- Indexing rules that define which collections an object is included in when the object of a specified object type and domain combination moves to a specific life cycle state. Collections are used to create indexing lists which help improve performance when searching for objects.
- Notification rules that define which principals get notified when a specified event occurs for an object type and domain combination.

Containers provide a natural context for controlling access to the contained information. In application containers, access is controlled through container team membership, policy rules, and, for Windchill ProjectLink, ad hoc rules applied to folders and roles. Container contents can be restricted so that access is limited to the members of a container team or the container information can be made more broadly available to the enterprise through policies that grant specified groups in an organization access to specific object types.

The policies that are in effect in a context are determined by the policies set in the domains that are in the current context, as well as those set in the ancestor domains. For details, see [Administering Domains and Policies](#).

### Installed Site Container Policies

The Site container policies that are set consist of the domain-based access control rules and one indexing rule. No notification rules are set in the Site container domains, and no policy rules are set in the Site container /SessionIterationDomain domain.

**Note:** Your solution may vary from the following description, as the name of the Administrators group, and the names of the some initial domains are configurable from the wt.properties.xconf file prior to the installation.

The following sections describe the rules that are set in the Site container.



### Access Control Rules for / (Root) Domain

The following domain-based access control rules are programmatically set when the data is loaded during the installation. The rules are in the Site container / (root) domain for all life cycle states.

**Note:** These rules ensure that users can operate within the Windchill solution and should not be changed without fully understanding the reason for the change.

Object Type	Principal	Permissions	Comment
AccessPolicyRule	ALL	Read	Allows organization and application container administrators to see inherited access rules.
AdministrativeDomain	ALL	Read	Allows all users to view domains.
EPMDocConfigSpec	ALL	Full Control (All)	For Windchill Foundation & PDM, allows all users to set and configure personal configuration specification settings. For Windchill PDMLink and Windchill ProjectLink, allows all users to perform configuration management of business objects within their workspace when using Pro/ENGINEER or other workgroup managers.
ExchangeContainer	ALL	Read	Allows all users to complete a variety of general actions.
ObjectSubscription	ALL	Read and Create	Allows all users to create and read object subscriptions.
WTDocumentConfigSpec	ALL	Full Control (All)	For Windchill Foundation & PDM, allows all users to set and configure personal configuration specification settings.
WTMarkup	ALL	Read and Create	Allows all users to create and read markups. These permissions are required because view markups are not life-cycle managed.
WTMarkup	OWNER	Modify and Delete	Allows the owner of a markup the ability to modify and delete it.
WTObject	Administrators	Full Control (All)	Grants full control to all site administrators.

Object Type	Principal	Permissions	Comment
WTPartConfigSpec	ALL	Full Control (All)	For Windchill Foundation & PDM, allows all users to set and configure personal configuration specification settings. For Windchill PDMLink and Windchill ProjectLink, allows all users to perform configuration management of business objects within their workspace when using Pro/ENGINEER or other workgroup managers.

### Access Control Rules for /User Domain

The following domain-based access control rules are set in the Site container /User domain for all life cycle states:

Object Type	Principal	Permissions	Comment
DBPrefEntry	All	Read	Allows users to read preferences.
WTGroup	Unrestricted Organizations	Read	Allows read access to groups for the organizations that are in this group.
WTOject	OWNER	Full Control (All)	Grants full control to the owner of an object.
WTPartConfigSpec	Authors	Read Modify Create	This rule uses an example group named Authors to show how to set up a group to manage business objects. It allows read, modify, and create permission to users in the Authors group for part configuration specifications.
WTUser	Unrestricted Organizations	Read	Allows read access to users for the organizations that are in this group.

### Access Control Rule for /User/Unaffiliated Domain

The following domain-based access control rule is set in the Site container /Unaffiliated domain (which is a child of the /User domain) for all life cycle states:

Object Type	Principal	Permissions	Comment
WTPrincipal	All	Read	Allows read access to principals for all users.

### Access Control Rules for /Default Domain

The following domain-based access control rules are set in the Site container /Default domain for all life cycle states:

Object Type	Principal	Permissions	Comment
Meeting	All	Read	Allows read access to meetings for all users. This rule is used to provide visibility of WebEx meetings for all users. For information about setting up meetings with WebEx, see the <i>Windchill System Administrator's Guide</i> .

Object Type	Principal	Permissions	Comment
RelationshipMap	All	Read	This object type is used for collecting dependent objects when archiving objects. For details on archiving, see the <i>Windchill Archive Administrator's Guide</i> .

### Access Control Rules for /System Domain

The domain-based access control rules set in the Site container /System domain are used to control access to administrative objects. For normal operations, you should not modify these rules.

To view the rules, use the Policy Administrator from the Site context. For information about accessing the Policy Administrator, see [Using the Policy Administrator](#).

### Indexing Rule for / (Root) Domain

The following indexing rule is set in the Site container / (root) domain for all life cycle states:

Object Type	State	Collections	Comment
WTOBJECT	All	Windchill_Business_Collection	Indexes all object in all states and puts the indexes into the Windchill_Business_Collection collection.

### Updating Container Policies

Use the Policy Administrator to update container policies.

For information about using the Policy Administrator, see [Administering Domains and Policies](#).

### Container Data Types and Attributes

Data type and attribute items identify data types (defined as soft types) and attributes that are in addition to the Windchill modeled classes.

The data types available in a context are determined by the data types defined in the organization context, as well as those defined in the Site context.

For example, if your organization context has defined Plan and Report document soft types, then these soft types are available in any project, product, or library context created from your organization context.

## Installed Site Container Data Types and Attributes

All data types that are modeled in your Windchill solution are available for administrator use from the Site container.

If Windchill ProjectLink is installed, then the following document soft types are also available from the Site container:

- Agenda
- General
- Minutes
- Plan
- Presentation

## Updating Container Data Types and Attributes

For Windchill PDMLink and Windchill ProjectLink, you can use the **Types** link from the **Site** and **Organization** tabs to view and update data types:

- To create and update site-wide types, navigate to the **Site** tab and click the **Types** link.
- For organization-specific types, access an organization and click the **Types** link.

To associate attributes with a type, the attributes must already be defined at the site level. Use the Attribute Manager to define attributes and use the Type Manager to add attributes to data types. Only the site administrator can create and update attributes.

**Note:** The internet domain defined on an organization principal is important when creating new soft types. The internet domain is used to distinguish which organization owns the type. The organization principal created during the installation process owns the Site container; any types owned by this organization principal are available to all organizations. Soft types created from an organization container that has its own internet domain set on its associated organization principal are only available to the owning organization.

For additional information about updating data types and attributes, see [Using Types and the Type Manager](#).

## Container Templates

Template items identify the templates that provide users with the required information used when creating and processing Windchill objects.

Windchill provides the following types of templates (not all types of templates are available from all Windchill solutions):

- Context templates provide the required and optional administrative items that are used to create containers. There are four types of context templates:
  - Product templates define default values and other information, such as team roles and access policies, which are used when an administrator creates a product using Windchill PDMLink.
  - Library templates define default values and other information, such as team roles and access policies, which are used when an administrator creates a library using Windchill PDMLink.
  - Project templates define default values and other information, such as folder structure and team roles, which are used when an administrator creates a project using Windchill ProjectLink.
  - Organization templates define default values and other information, such as folder structure and groups, which are used when an administrator creates an organization context from either Windchill PDMLink or Windchill ProjectLink. Each solution provides a unique set of organization templates.
- Document templates provide content files and default values, which are used when users create different types of documents. For example, content might include skeleton documents for memos or reports.
- CAD document templates define the content and default values, which are used when users create CAD documents from either Windchill Foundation & PDM or Windchill PDMLink.
- Life cycle templates define the phases and gates associated with various business objects when the objects are initialized.
- Report templates define default values, which are used when users run reports for change management.
- Team templates define default roles, which are used when users populate teams associated with life cycles.
- Workflow templates define default values, which are used when users initiate a workflow.

## Installed Site Container Templates

The templates loaded in the Site container are associated with the /System domain.

The following sections describe the templates that are loaded.

### Organization Context Templates

**Note:** Windchill Foundation & PDM does not load or use organization context templates.

When Windchill PDMLink is installed, the following organization context template is loaded in the Site container:

- General (PDM)

When Windchill ProjectLink is installed, the following organization context templates are loaded in the Site container:

- General
- Enterprise
- Supplier
- Customer

For additional information about organization context templates, see [Out-of-the-Box Organization Templates](#) in the Administering Organizations chapter. For specific information about creating organization context templates, see [Creating Context Templates](#), later in this chapter.

### Life Cycle Templates

For the details on out-of-the-box life cycle templates, see [Out-of-the-Box Life Cycle Templates](#) section in the Administering Life Cycles chapter.

### Workflow Templates

The following workflow templates are loaded in the Site container for all Windchill solutions:

- Submit
- Review
- ReplicationPublish
- ReplicationReceive
- ReplicationSender

For Windchill PDMLink and Windchill Foundation & PDM when the Closed-loop Process option is installed, the following workflow templates are loaded in the Site container:

- Problem Report Workflow
- Change Request Workflow

Change Notice Workflow  
Change Activity Workflow

For Windchill ProjectLink, the following workflow templates are loaded in the Site container:

Approval Process  
Notify Process  
Release Process  
Review Process

## Team Templates

Team templates are used with life cycle templates.

For all Windchill solutions, the following team template is loaded in the Site container:

Default

For Windchill PDMLink and for Windchill Foundation & PDM with the Closed-loop Process option installed, the following team templates are loaded in the Site container:

CA Team  
ECN Team  
ECR Team  
Problem Report Team

## Document Templates

For Windchill ProjectLink, the following document templates are loaded in the Site container:

Agenda Template  
Memo Template  
Minutes Template  
MS Project Plan Template  
Presentation Template  
Requirements Template

No document templates are loaded for Windchill Foundation & PDM and Windchill PDMLink.

## Updating Container Templates

For Windchill PDMLink and Windchill ProjectLink, templates are available from the **Templates** link in each context. To update templates, navigate to the **Templates** table by clicking the **Templates** link within a specific context and then select the type of template from the **Current View** drop-down list. The list of templates that are available from the context appears in the table. Each type of template has its own method of updating the templates. Generally, updating can



be initiated either by clicking an icon at the top of the **Templates** table or selecting an action from a row in the table.

For details on templates and how to update them, see the help that is available from the **Templates** table. For specific information about updating context templates, see [Creating Context Templates](#) later in this chapter.

For Windchill Foundation & PDM, you update each template using the administrator utility that is associated with the template. The following links, located under Business Administration, give you access to the utilities:

- Process Administrator (for life cycle, team, and workflow templates)
- Document Template Administrator (for document templates)

## Container Object Initialization Rules

Container object initialization rule items identify rules that establish specific default attribute values that are used when instances of objects of a specific type are created.

Windchill solutions provide algorithms for setting default values for the following object attributes:

- Folder paths
- Life cycle template attributes
- Team template attributes
- Default numbering scheme
- Default versioning scheme
- Strings
- Enumerated types

For additional information about object initialization rules, see [Administering Object Initialization Rules](#).

## Installed Site Container Object Initialization Rules

For all Windchill solutions, the out-of-the-box numbering and versioning schemes are used for WTPart, WTDocument, and EPMDocument in the Site container. These schemes are simple Oracle sequences that have been loaded into the Windchill database. Each starts at 1 and increments by 1.

For change objects (WTChangeActivity2, WTChangeIssue, WTChangeOrder2, WTChangeProposal, and WTChangeRequest2), the out-of-the-box numbering scheme is also set.

For additional information about the out-of-the-box numbering and versioning schemes, see [Administering the Numbering of Parts, Documents, CAD](#)

[Documents, and Change Objects](#) and [Administering the Versioning of Parts, Documents, and CAD Documents](#).

No additional object initialization rules are set in the Site container for any solution. However, for Windchill PDMLink, a generic initialization rule that is coded for all WObject types when no default life cycle template is defined is to use the Basic life cycle template. Similarly, when no team template is defined, the rule is to use the Default team template.

For Windchill Foundation & PDM, the value Default is used as the default life cycle template and Default is used as the default team template for all life cycle-managed objects.

**Note:** In Windchill Foundation & PDM, the application clients set values for folder paths, life cycle templates, and team templates. Default values are never used in these cases.

**Note:** In Windchill ProjectLink, default values set for folder paths are not used.

**Note:** PTC recommends that you do not modify the out-of-the-box object initialization rules that are set in the Site container without understanding what the change affects. Windchill PDMLink provides a demonstration template named Product Design that uses a different versioning scheme and two life cycle templates that you can use in evaluating what object initialization rules should be set for your site. For more information on the Product Design template, see [Out-of-the-box Product and Library Context Templates](#) in the Administering Products and Libraries chapter.

## Updating Container Object Initialization Rules

Use the Object Initialization Rules Administrator to update object initialization rules.

For information about using the Object Initialization Rules Administrator, see [Administering Object Initialization Rules](#).

# Creating Containers

**Note:** Only Windchill PDMLink and Windchill ProjectLink provide the capabilities for creating organization and application containers. You cannot create containers in Windchill Foundation & PDM.

There is only one Site container, which is created when your Windchill solution is installed. No additional Site containers can be created.

**Note:** In Windchill PDMLink, the base data installation can create an organization container. If one was not created by loading the base data, the site administrator creates the organization container from the **Site** tab. In Windchill ProjectLink, the base data installation does not create an organization container; after the installation is complete, the site administrator creates the organization container from the **Site** tab.

Application container administrators can create application containers. You become an application container administrator for products, libraries, or projects by being in the creators group for products, libraries, or projects. The creators groups are maintained from the **Creators** link on the **Organization** tab. When you create an organization from Windchill ProjectLink, you can choose to automatically add those organization members who log on to the project creators group. In Windchill PDMLink, you manually add users to the product and library creators groups.

As part of the process of creating a container, you select a context template. Context templates provide the administrative items (as described in [Container Administrative Items](#)) that you want set in the container, thus establishing the initial context framework for users. For the details on creating context templates, see [Creating Context Templates](#), later in this chapter.

For information about creating organization containers, see [Creating an Organization](#) in the Administering Organizations chapter.

For information about creating product and library containers, see [Administering Products and Libraries](#).

For information about creating project containers, see [Administering Projects](#).

## Removing Templates

To remove templates that have been loaded, click the **Site** tab and then click the **Templates** link to display templates. Select the type of template from the **Current View** drop-down list to display the templates in the table. Use the delete row action to remove individual templates. For additional help on removing templates, click the help icon on the **Templates** table to display the templates help.

## Using Out-of-the-box Context Templates

Your Windchill solution provides a set of out-of-the-box context templates that you can use when creating containers.

The text in the templates has been translated, and translated files are provided in the loadFiles and loadXMLFiles directories where your Windchill solution is installed. As part of the installation process, the installer can change the load set so that the base data loaded includes files translated into the language of choice at your site. The base data that is loaded determines which language set of context templates is available.

For details on setting which language files are used when loading base data, see the Database Initializing and Data Loading chapter in the *Windchill Installation and Configuration Guide - Windchill*.

Use one of the out-of-the-box organization templates when creating organization containers, or you can create a customized organization template, as described in the next section.

For application containers, you can use one of the out-of-the-box product, library, or project templates, or you can create additional application container templates, as described in the next section.

The details of what is in each out-of-the-box context template can be found in the following chapters:

- Organization templates are described in [Administering Organizations](#).
- Product and library templates are described in [Administering Products and Libraries](#).
- Project templates are described in [Administering Projects](#).

**Note:** The role names used in Windchill solution context templates are included as references to the <Windchill>\src\wt\project\RoleRB.rbinf resource bundle so that when they are used, they appear in the language specified by the browser language. This means that the translated templates do not contain translated role names.

## Creating Context Templates

You can create additional context templates in the following ways:

- By saving the contents of an existing container as a new context template. This option is available for products, libraries, and projects.
- By exporting an existing application container as a template, updating the template contents (if needed), and then importing the new template. This option is available for products, libraries, and projects.
- By creating a context template input file and then using the **Create Template** icon from the **Templates** table to create the template. This option is available for organizations, products, libraries, and projects.

The following sections describe the format and contents of the files you can use as input when you create context templates.

**Note:** The descriptions in the following sections assume you are familiar with XML. The XML files that are used must be coded against the standardX05.dtd. You can download the standardX05.dtd by downloading any of the context templates. The DTD is contained in the ZIP that is downloaded.

## Organization Templates

The only option available for creating an organization template is to create the input file manually and then, from the **Site** tab, use the **Create Template** icon from the **Templates** table that shows organization templates to create the template.

The input file used when creating an organization template can be just the business XML file if no additional content is needed or a ZIP (or JAR) file that has both the business XML file and other content files named in the XML file.

You can include the following administrative items in a new organization template:

- Additional domains, folders, groups, and access control rules for the organization
- Additional product, library, and project templates that are then available from the organization
- Object initialization rules
- Project roles (if the organization is used with Windchill ProjectLink)

For details on the XML elements required in the business XML file for organization templates, see [Creating Business XML Files for Organization Templates](#) in the Creating Business XML Files for Templates appendix.

## Windchill PDMLink Context Templates

When using any of the options for creating templates, you can include the following administrative items in a new Windchill PDMLink product or library template:

- Domains
- Folder structure (not including the folder contents)
- Team roles
- Team members
- Object initialization rules
- Access control policy rules

## Windchill ProjectLink Context Templates

When using any of the options for creating templates, you can include the following administrative items in a new Windchill ProjectLink project template:

- Domains
- Folder structure (not including the folder contents)
- Folder links and structure
- Project plan
- Deliverables
- Team roles (including role access to actions)
- Team members
- Forum specification
- Forum template
- Object initialization rules
- Access control ad hoc rules
- Shared objects (if any have been created in an existing project that is exported)

Additionally in Windchill ProjectLink, you can save an existing project as a new project. Using this option creates a duplicate of a project under a new name.

## Access to Creating Additional Context Templates

In both Windchill PDMLink and Windchill ProjectLink, you can save a product, library, or project container as a new template from the details page of the container, or you can export an existing product, library, or project container from the details page of the container and then import the exported file in another

container. The following actions are available from the details page for site and organization administrators:

- The **Save As <context> Template** action provides a way to create a new template from the current context. In the action, <context> is either **Product**, **Library**, or **Project**, depending on the context you are viewing.

A newly created product, library, or project template is saved in the application container's organization context and appears in the **Templates** table of that context when the **Current View** is <context> **Templates**.

- The **Export As Template** action allows you to export the current product, library, or project context as a template, creating a ZIP file on your system.

You can then import the exported template into Windchill PDMLink or Windchill ProjectLink by navigating to either the Site or the organization under which you want the template imported and clicking the **Templates** link. From the **Templates** table that appears, select the type of template from the **Current View** and click **Import Template**. This action creates a template with the same name as the exported template.

If you want to specify a different name for the template, you can do either of the following:

- Choose **Create Template** instead of **Import Template** to import the exported template. On the window that displays, you specify the name and description of the template. This information is used in place of the name and description in the ZIP file.
- Modify the contents of the TAG-0.xml file that is in the exported ZIP file to change the name and description of the template and then use **Import Template** to import the exported template. The following section describes the contents of the exported ZIP file.

**Note:** The ZIP file created by selecting the **Download** action from a context template row on the **Templates** table cannot be used as the template data file for creating a context template using the **Import Template** action; it does not contain all of the required information that is needed to create a new context template. The downloaded ZIP file can be used to update an existing context template or create a new template using **Create Template**.

You can use **Create Template** to create organization, product, library, or project templates. The input file you provide can be one that you manually create; however, PTC recommends that you always start with a business XML file that you have either downloaded or located in the codebase. The import and creation functions do not validate that the input XML conforms to the DTD; you must do this before doing the import or create function. The [Creating Business XML Files for Templates](#) appendix describes the elements that you can use in each business XML that you can use as input.

## ZIP File Contents of an Exported Context Template

The ZIP file created on your system from an **Export As Template** action contains the framework options you selected. Depending on your system configuration, you may be prompted to save the ZIP file or the file may be saved on your local drive with the name `<context_name>_generated_template<integer>.zip`, where `<context_name>` is the name of the product, library, or project being exported and `<integer>` is an integer that ensures that the file has a unique name when the download occurs. The name used is not significant and can be changed.

The ZIP file contains the following:

- The `importList.txt` file, which is not currently used during an import and can be ignored.
- The `TAG-ContainerTemplate_<integer>.xml` file, which contains information about container template itself, such as name and container class of the template. Here also, `<integer>` is used to ensure that the files are unique.
- The `TAG-0.xml` file (which is in the `CONTENTS` directory) lists the contents of the exported template. This file is also known as the business XML file and contains information that is used when creating a container from this template (for example, folder structure, access control policy rules, and so on).
- Additional files that make up the contents of the exported template. If unzipped, these files are put in the `CONTENTS` directory. For example, the ZIP file of a project template could contain documents and document templates that then belong in the `CONTENTS` directory.

## Required Contents of ZIP File Used for Importing a Context Template

The ZIP file you use to import a context template must have the same basic structure as the ZIP created when exporting a context template (as described in the previous section). The ZIP file must contain the following:

- An XML file at the top level of the ZIP that contains the definition of the context template itself, such as name and container class of the template. This corresponds to the `TAG-ContainerTemplate_<integer>.xml` file that is generated in an export. For details on the content of this file, see [Contents of Top-level XML File for Imported Templates](#).
- The business XML file (which is defined as coming from the `CONTENTS` directory) that lists the contents of the template you want to import and contains information that is used when creating a container from this template (for example, folder structure, access control policy rules, and so on). This corresponds with the `TAG-0.xml` file that is generated on export. The name of this file must match the file named in the `zipEntry` element in the top-level XML file. For details on the content of the business XML file, see the [Creating Business XML Files](#) appendix.



- Additional files to include in the contents of the template you want to import. These files are named in the business XML to be added as content items through the use of the `contentItem` element. Before creating the ZIP file, put these files in the CONTENTS directory. Examples of files that can be in this directory are documents and document templates that you want to import as part of a Windchill ProjectLink project template.

**Note:** The file used could also be a JAR file rather than a ZIP file.

## Contents of Top-level XML File for Imported Templates

The top-level XML file is used to define the template that you are importing. The file must begin with the following lines:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE ContainerTemplate SYSTEM "standardX05.dtd">
```

The content of the top-level XML file is defined by the following DTD element:

```
<!ELEMENT ContainerTemplate (name,
                             description?,
                             containerClassName,
                             defaultValuesRule?,
                             locale?,
                             enabled?,
                             (xmlText | loadPath | zipEntry)?)>
```

The following example illustrates how to use the `ContainerTemplate` element to define the template you want to import. The example is for a Windchill PDMLink product template as signified by the use of `wt.pdmlink.PDMLinkProduct` in the `containerClassName` element.

In the example, the name of the template is DEMO Product Template and, when the template is imported, it is enabled. The business XML file that contains the content of the template is contained in the CONTENTS/DEMOProductTemplate.xml file that is in the imported ZIP file:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE ContainerTemplate SYSTEM "standardX05.dtd">
<ContainerTemplate>
  <name>DEMO Product Template</name>
  <description>A description of the DEMO template</description>
  <containerClassName>wt.pdmlink.PDMLinkProduct</containerClassName>
  <enabled>true</enabled>
  <zipEntry>CONTENTS/DEMOProductTemplate.xml</zipEntry>
</ContainerTemplate>
```

**Note:** Although the DTD element shows that you can specify the business XML file in multiple ways (through `xmlText | loadPath | zipEntry`), only the use of the `zipEntry` element is supported when you are importing a template.

To specify a project template instead of a product template, use `wt.projmgmt.admin.Project2` in the `containerClassName` element. For a library template, use `wt.inf.library.WTLibrary`.

## Creating Business XML Files

Business XML files are used as input for the **Import Template** and **Create Template** actions provided in both Windchill PDMLink and Windchill ProjectLink (see [Access to Creating Additional Context Templates](#)):

- The **Import Template** requires that the business XML file and additional files (as described in [Required Contents of ZIP File Used for Importing a Context Template](#)) be packaged in a ZIP or JAR file.
- For the **Create Template** action, the business XML file can be used alone as the input when no additional content files are referenced in the business XML file, or it can be used in a ZIP or JAR file when additional content files are referenced in the XML file.

All business XML files must start with the following lines:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE context_type SYSTEM "standardX05.dtd">
```

where *context\_type* is one of the following:

- **OrganizationConfig** for organization templates
- **ProductConfig** for product templates
- **LibraryConfig** for library templates
- **ProjectConfig** for project templates

For the details on the XML elements supported in the business XML files, see the [Creating Business XML Files for Templates](#) appendix.

## Administering Domains and Policies

This section describes how domains and policies are defined. A *domain* is an administrative area that defines a set of administrative policies, such as access control, indexing, and notification. Objects associated with a domain are subject to its policies.

A *policy* is a collection of rules designed for various types of objects associated with a domain. For example, an indexing policy consists of rules that determine the types of objects for which metadata should be entered into specified collections, when the objects belong to the domain.

Before creating a container in Windchill PDMLink and Windchill ProjectLink, you should determine which domains are needed and use a template that creates the domains and the access control policies for those domains when you create the container. For details on what domains and policies are included in the out-of-the-box templates, see the following chapters:

- Organization templates are described in [Administering Organizations](#).
- Product and library templates are described in [Administering Products and Libraries](#).
- Project templates are described in [Administering Projects](#).

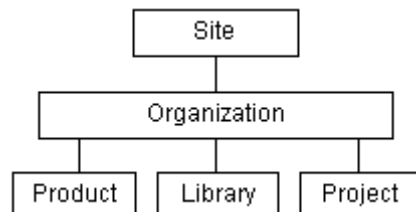
For information about creating additional templates, see [Creating Context Templates](#).

After a container is created, you can use the Policy Administrator to administer the domains in the container.

The following sections provide information about installed domains and how to use the Policy Administrator to administer the domains.

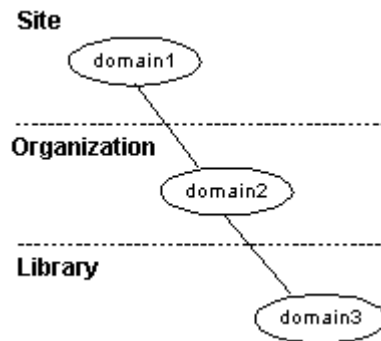
## Container and Domain Hierarchy Overview

Container types have the following established hierarchy:



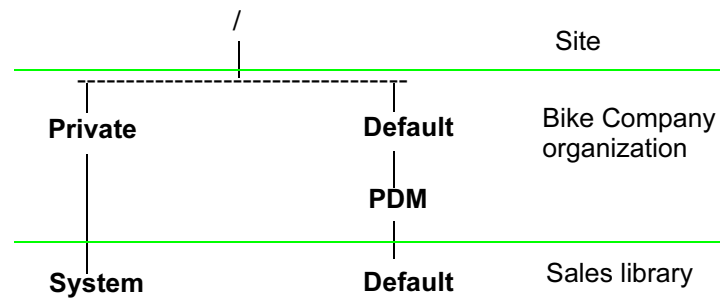
When each container is created, a set of domains is also created for use within the container context. Generally, the domain hierarchy is established using the container hierarchy. A parent domain is either in the same container as its child domain or in the parent container of its child domain's container (Windchill Foundation & PDM being the exception to this rule). In the following diagram, the site, organization, and library containers are shown with one domain defined in each container. The domain hierarchy shows domain3 (in the library container)

as a child of domain2 (in the organization context) and domain2 as a child of domain1 (in the site container):



To illustrate this rule, consider the following examples:

- Assume that the Windchill PDMLink installation creates the Site container and has a child organization container that is named Bike Company, and that an administrator (using an out-of-the-box library template) creates a public library container named Sales that is a child of the Bike Company container. Then the domain hierarchy for the Windchill PDMLink Sales library is as follows:

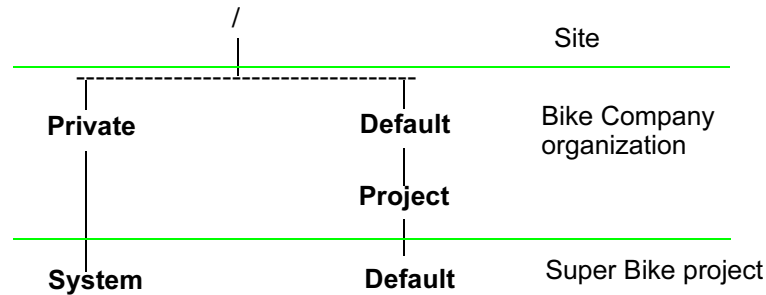


From within the Policy Administrator, the hierarchy is shown as follows:

```

/ (Site)
|-Default (Organization Bike Company)
  |-PDM (Organization Bike Company)
    |-Default (Library Sales)
  |-Private (Organization Bike Company)
    |-System (Library Sales)
  
```

- Assume that the Windchill ProjectLink installation creates the Site container and has a child organization container that is named Bike Company, and that an administrator (using an out-of-the-box project template) creates a public project container named Super Bike that is a child of the Bike Company container. Then the domain hierarchy for the Windchill ProjectLink Super Bike project is as follows:



From within the Policy Administrator, the hierarchy is shown as follows:

```

/ (Site)
|-Default (Organization Bike Company)
  |-Project (Organization Bike Company)
    |-Default (Project Super Bike)
  |-Private (Organization Bike Company)
    |-System (Project Super Bike)
  
```

- Assume that the Windchill Foundation & PDM installation loads the Windchill Foundation & PDM library container as a child of Bike Company organization container and the organization container is a child of the Site (root) container. When only the out-of-the-box load files are used, the domain hierarchy for the Windchill Foundation & PDM library does not include domains from the organization container. Instead, the ChangeItems domain in the Windchill PDM library container inherits directly from root (/) as shown in the Policy Administrator:

```

/ (Site)
|-ChangeItems (Library Windchill PDM)
  
```

## Domains in the Site Container

When a Windchill solution is installed, the following domains are initially defined in the Site container:

Domain Names	Description
/ (root )	Sets rules that govern the entire enterprise.
/System	Serves as the default domain for most system administration objects, including queues and life cycle definitions. /System is the default domain of the Administrators group.  The /System domain is not intended for general user business objects.
/User	Serves as the default domain for organization objects and as the parent domain for the /Unaffiliated domain and for organization container domains that hold the users (and their personal cabinets) for that organization.
/Default	Serves as the default domain for the initialization of business information objects; however, this domain is not usually the domain set when an object is persisted.
/SessionIterationDomain	Serves as the default domain for internal session processes. Do not modify the policies set for this domain.
/Unaffiliated	Serves as the default domain for users who are not associated with any organization. These users do not have the organization attribute set in their directory service entry. The /Unaffiliated domain is a child domain of the /User domain.

The root, /System, /User, and /Default domains cannot be moved or deleted. Also the domain names of these domains and of the Unaffiliated domain are configured through the wt.properties file.

The /System, /User, /Default, and /SessionIterationDomain domains belong to the / (root) domain. Additional domains can be defined and associated with the root domain, or domains can be nested by defining them as children of another domain. This allows you to define a hierarchy of domains and then define policies at each level in the hierarchy.

## Creating Domains

Windchill automatically creates domains as follows:

- During installation, as described earlier in this chapter.
- When you create containers. Each container template defines a set of domains that are created.
- Each time Windchill identifies a user that is affiliated with an organization that does not have an associated domain, Windchill creates the corresponding domain for the organization as a child domain of the User domain.

In addition, administrators can manually create domains.

The name of a domain can be a maximum of 200 characters. When Windchill automatically creates the domains associated with organizations, the domain name is usually the same as the organization name. However, organization names can be up to 200 characters. Therefore to name the domain, Windchill truncates the organization name to 193 characters. This allows Windchill to add a maximum of seven more characters to the domain name to identify a unique name when two or more organizations have the same beginning 193 characters in their names. When the truncated domain name is not unique, Windchill appends [1] to the truncated name. If appending [1] to the name does not make it unique, then Windchill appends [2] instead of [1]. If appending [2] does not make the name unique, then [3] is appended and so on until a unique name is found or until the maximum number of attempts to find a unique name is achieved. The maximum number of attempts is defined in the following property in the wt.properties file:

`wt.inf.container.WTContainerServerHelper.maxDomainCreationAttempts`

The default value for this property is 25.

If the organization name contains a forward slash ( / ), Windchill automatically converts the forward slash to a dash ( - ) in the domain name. For example, if the organization name is "ABC/Main", the domain name created would be "ABC-Main".

**Note:** After Windchill creates a domain that is associated with an organization, an administrator can change the domain name to something more meaningful; Windchill does not rely on the organization domain name matching the organization name.

## Defining Domain-based Policies

Using the Policy Administrator, you can create additional domains and define or change the rules set for each domain. For example, you can define the following:

- An *access control policy*, which determines user, group, and organization permissions to access objects associated with the domain.
- An *indexing policy*, which determines the collections into which metadata for an object is entered when the object is in a specific life cycle state.
- A *notification policy*, which determines who is informed when an event of interest occurs to an object in the domain.

Load files can include sets of domain-based policies for access control and indexing. Templates that are used to create containers can only include sets of domain-based policies for access control.

**Note:** Policy rules apply to object types. Access to an instance of an object type can be governed by ad hoc access control rules in addition to policy rules. For more information, see [Administering Access Control](#).

For objects to be subject to policies, they must reside in a domain. The policies that apply to an object are those policies defined for the domain to which the object belongs, as well as ancestor domains. To exist within a domain, the object must implement the `wt.admin.DomainAdminister` interface. Not all objects exist within a domain. For example, objects of type `WTPartMaster`, `WTDocumentMaster`, and `WorkItem` do not exist within a domain. For details on the `wt.admin.DomainAdminister` interface, see the associated javadoc.

## Using the Policy Administrator

The Policy Administrator operates within the confines of the context from which it is started.

How you access the Policy Administrator is determined by your Windchill solution:

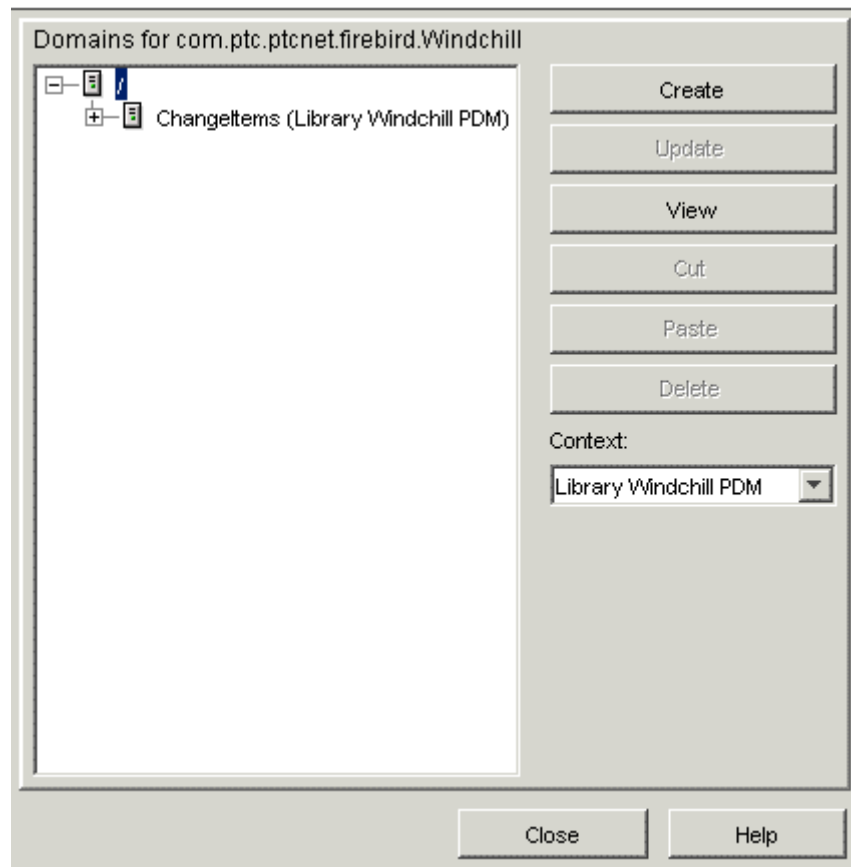
- From Windchill Foundation & PDM, you can access the Policy Administrator by clicking the **Policy Administrator** link that is on the **Business Administration** home page.
- From Windchill PDMLink and Windchill ProjectLink, you can access the Policy Administrator from the **Utilities** pages that are under the **Site**, **Organization**, **Product**, **Library**, and **Project** tabs.

You begin policy administration on the Policy Administrator window. From this window, you have access to the following domains:

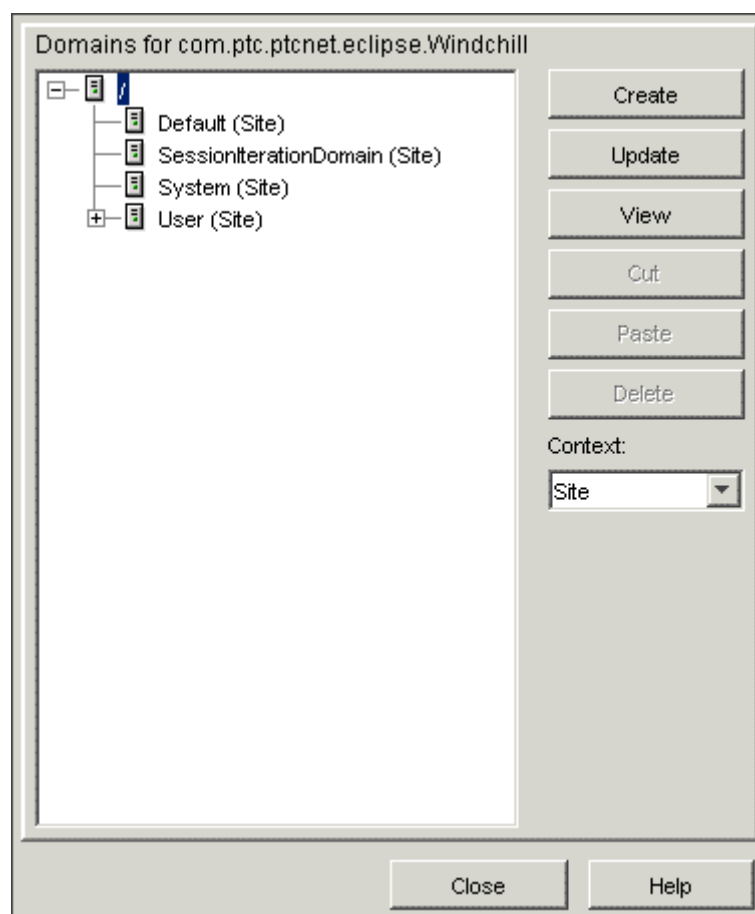
- The domains that are in the current context.
- The ancestor domains of the domains in the current context.



For example in Windchill Foundation & PDM, the initial window is similar to the following:



When accessing Policy Administrator from the **Site Utilities** page in either Windchill PDMLink or Windchill ProjectLink, the initial window is similar to the following:



The domains display in a tree view which is open to the domains at the top of the domain hierarchy within the current context. The current context appears selected in the **Context** drop-down list.

The format of each domain in the list includes the domain name followed by the following information in parentheses:

- The type of context (for organization and application contexts)
- The name of the context where the domain resides

For example, if a domain in the list is shown as **Default (Library Common Parts)**, then the domain name is Default and the domain resides in the library context named Common Parts. If the Default domain is in the Site context, then you will see **Default (Site)**. In this case, the context type is not displayed.

You can select any of the domains listed in the tree, or you can expand the tree to show descendent domains within the current context and then select a domain. You can also change the contents of the domain tree by using the **Context** drop-down list, as follows:

- Selecting an ancestor context from the **Context** drop-down list updates the domain tree so that the domains in the selected context and direct ancestor domains are available.

The contexts listed in the **Context** drop-down list are formatted as the context type followed by the context name.

- Selecting **All Contexts** from the **Context** drop-down list updates the domain tree so that all domains from all contexts are available.

**Note:** The buttons that are enabled when you select a domain are determined by the Policy Administrator. Buttons are disabled if the action cannot be performed on the domain based on current selections or if the action can never be performed on the domain; buttons are not disabled based on user permissions. For example, if you select a domain that can never be deleted (such as the Default (Site) domain), the **Delete** button is disabled. If a button is enabled and a user who does not have permission to execute the action clicks the button, the user receives an error.

## Specifying Policy Rules in a Context Template

For Windchill PDMLink, the recommended method for including a specific set of policy rules in a context template is to set the rules in an existing container using the Policy Administrator. Then, either save the current context as a template or export the context to a ZIP file on your system. Be sure to select the **Access Policy Rules** option when you save or export an existing template. Using this method provides access control policy rules for domains within a container and allows only groups and organizations as principals. Also the principal names must be unique within the container search scope. For information about the search scope, see [Searching for Principals](#).

After you have a ZIP file containing the access control policy rules (along with any other administrative items you want in the template), import the template using the ZIP file as input.

For Windchill ProjectLink, you cannot save the policy rules set in the current context when creating a template from the current context or when exporting the context.

For information on creating context templates, see [Creating Context Templates](#).

## Assigning Domains to Folders in Windchill PDMLink or Windchill Foundation & PDM

When creating folders, the default behavior is that the domain associated with the new folder is the same domain that is associated with the parent cabinet or folder.

The ability to create folders is controlled through the access control policy rules that are set for folders. Out-of-the-box, only product or library managers and organization administrators (in Windchill PDMLink) and the site administrators (in both Windchill PDMLink and Windchill Foundation & PDM) can create folders (or update the domains in existing folders). To allow others to create folders, you can add access control policy rules that grant Create and Modify permission for the SubFolder object type to those principals who you want to have this capability.

To allow those who can create folders the ability to associate a domain other than the domain that is associated with the parent folder, you must set the following preference:

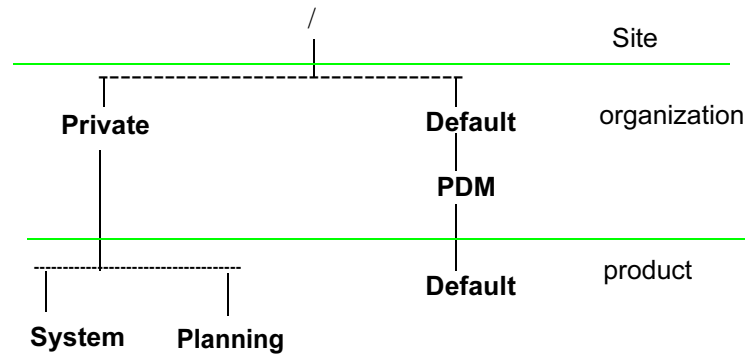
```
/wt/admin/displayDomains=true
```

Setting this preference to true reveals domain information on the dialogs used when creating or updating folders and on folder details pages. Providing the domain information allows users who have the appropriate access control rights to assign different domains to folders and to see which domain is currently assigned to a folder. In Windchill PDMLink, the interface allows a user to select a domain from the domains established in the container other than the /System domain. For Windchill Foundation & PDM, users can select any domain in the container and domains can be assigned to cabinets as well as folders (see [Folded Information](#) in the Administering Access Control chapter).

You can set this preference at any Windchill PDMLink or Windchill Foundation & PDM container level through the Preference Administrator as described in the following:

- In Windchill PDMLink, the Preference Administrator is available from the product, library, organization, and Site **Utilities** pages.
- In Windchill Foundation & PDM, the Preference Administrator is available from System Administration home page.

You can use the ability to create folders that are associated with a domain other than the domain that is associated with the parent folder in conjunction with the ability to create top-level domains in an application container. For example, assume you have created the /Planning domain in following domain structure for a product:



The access control rules in the /Planning domain only inherit the rules from the Site / (root) domain and the organization /Private domain and not the rules from the organization /Default and /Default/PDM domains. When creating folders in a product, the domain associated, by default, is the /Default domain in the product. By allowing the user to select a different domain, the user could select the /Planning domain instead of the /Default domain. Creating a folder associated with the /Planning domain provides a different access policy for objects in the folder since the access control rules for objects associated with the /Planning domain can be different than the rules associated with the product /Default domain and a different set of rules are inherited from the organization.

# Administering Object Initialization Rules

The Object Initialization Rules administrator provides a way to specify default values for the attributes of a specific object type. The default values are then used when the Windchill solution creates objects of that type. These specifications are called rules. Each rule can contain default values for one object type.

**Note:** The object initialization rules that are set only apply when the Windchill solution that is used to create an object does not set a corresponding value. Therefore, if the user interface sets a value (such as a folder), the default value set in an object initialization rule will never be used.

**Note:** In Windchill Foundation & PDM, the application clients set values for folder paths, life cycle templates, and team templates. Default values are never used in these cases.

**Note:** In Windchill ProjectLink, default values set for folder paths are not used.

The following sections provide information on rules that are loaded during installation, how to add rules, and how rules work.

## Loading Object Initialization Rules

A set of rules is loaded during installation and additional sets are included in the context templates that are loaded (if either Windchill PDMLink or Windchill ProjectLink is installed). The following rules are set:

- For all Windchill solutions, a rules load file sets the initial numbering and versioning rules for WTDocument and WTPart.
- For Windchill Foundation & PDM, a rules load file sets the rules for the change objects that include setting values for the folder, life cycle, and numbering scheme.
- For Windchill Foundation & PDM and the Closed-loop Process option, a rules load file sets the rules for the change objects. These rules include the setting of the team template and the setting of different life cycle values.
- For Windchill PDMLink, a rules load file sets the rules for WTDocument and WTPart attributes in addition to the numbering and versioning rules set for all solutions, as well as numbering rules for the change objects.
- For Windchill ProjectLink, a rules load file sets the rules for document soft types that are defined for WTDocument.

The rules that are loaded are named according to the object type to which the rule applies and appear with a site context. For additional information, see [Installed Site Container Object Initialization Rules](#).

**Note:** PTC recommends that you do not modify the out-of-the-box object initialization rules that are set through the load files.

## Adding Object Initialization Rules

You can add rules for the creation of an object, as follows:

- By using the Object Initialization Rules administrator.
- By creating additional context templates from within Windchill PDMLink or Windchill ProjectLink or updating existing templates. For information about context templates, see [Container Templates](#).

## Accessing the Object Initialization Rules Administrator

You can access the Object Initialization Rules administrator, as follows:

- In Windchill Foundation & PDM, it is available from the **Business Administration** page.
- In Windchill ProjectLink, it is available from the project, organization, and Site **Utilities** pages.
- In Windchill PDMLink, it is available from the product, library, organization, and Site **Utilities** pages.

If you have access to the **Object Initialization Rules Administrator** link, then you can use it. Clicking the link displays the **Object Initialization Rules** table where all rules that have been added in the current context or ancestor contexts are listed.

## How Object Initialization Rules Work

Where a rule is created determines which objects are affected by the rule:

- Rules set during installation affect all objects of the types specified in the load file.
- Rules set from the **Business Administration** page affect all objects of the types specified that are initialized in the Windchill PDM context after the rule is set.
- Rules set from the site **Utilities** page affect all objects of the types specified that are initialized in the site context after the rule is set.
- Rules set from the organization **Utilities** page affect only those objects of the types specified that are initialized by a user from the organization under which the rule was set.
- Similar restrictions are in place for those rules set for project, products, and libraries, where the rules only apply to the project, product, or library under which the rules are set.

Rules for an object type that are set at one context do not replace other rules that are set at a higher context, but all rules are merged to create a composite rule. For example, a rule for WTPart numbering and versioning can be set at the Site

context and a rule for WTPart folders can be set at the product or organization context. Then the composite rule for WTPart objects created under the product or organization includes both the setting for numbering and versioning, and the setting for folders. If the product rule had included setting the numbering scheme, then this rule setting would take precedence over the setting made at the Site context.

If a default value is not set for an object attribute in the composite rule that is in place and the user creating the object does not specify a value for the attribute, then one of the following occurs:

- If the underlying code provides a default, then it is used. For example, if the rule does not set the default life cycle state, then the Life Cycle service would use its property value to set a default state.
- If there is no underlying code that provides a default, then the attribute value is set to NULL.

## Determining the Composite Rule

To determine what object initialization rule is in effect for an object type in a specific context, you can generate the composite rule for the object type.

Use the following steps to generate and display the composite rule:

1. Access the Object Initialization Rules Administrator from the context in which you want the composite rule generated, as described in [Accessing the Object Initialization Rules Administrator](#).

2. Click **Download Composite** .

The **Download Composite Object Initialization Rule** window opens.

3. Enter a valid object type. Valid object types are of any of the object types listed in the **Object Type** column of the **Object Initialization Rules** table.
4. Click **OK** to initiate the download process.

OR

Click **Cancel** to cancel the process.

How the download occurs depends on how your system is configured. Your system may be set up to display the XML in an XML editor or a browser. From the display, you can save the XML; otherwise, you may be prompted to save the XML in a file.



The following XML shows a sample composite rule for the wt.doc.WTDocument object type (formatted to fit on this page):

```
- <AttributeValues objType="wt.doc.WTDocument">
- <!-- set the folder -->
- <AttrValue id="folder.id" algorithm="com.ptc.core.foundation.folder.server.impl.
FolderPathAttributeAlgorithm" ignore="false" force="false" final="false">
  <Arg>/Default</Arg>
</AttrValue>
- <!-- set the lifecycle -->
- <AttrValue id="lifeCycle.id"
algorithm="com.ptc.core.foundation.lifecycle.server.impl.
LifeCycleTemplateAttributeAlgorithm" ignore="false" force="false" final="false">
  <Arg>Basic</Arg>
</AttrValue>
- <!-- set the team template -->
- <AttrValue id="teamTemplate.id"
algorithm="com.ptc.core.foundation.team.server.impl.
TeamTemplateAttributeAlgorithm" ignore="false" force="false" final="false">
  <Arg>Default</Arg>
</AttrValue>
- <!-- set the number to a generated number -->
- <AttrValue id="number"
algorithm="com.ptc.windchill.enterprise.revisionControlled.server.impl.
NumberGenerator" ignore="false" force="false" final="false">
  <Arg>{GEN:wt.enterprise.SequenceGenerator:WTDOCUMENTID_seq:10:0}</Arg>
</AttrValue>
- <!-- set the version info to a generated version info -->
- <AttrValue id="MBA|versionInfo"
algorithm="com.ptc.core.foundation.vc.server.impl.VersionInfoGenerator"
ignore="false" force="false" final="false">
  <Arg>wt.series.HarvardSeries</Arg>
</AttrValue>
</AttributeValues>
```

In this rule, the following wt.doc.WTDocument attribute default values are set:

- The folder.id default value is set to Default.
- The lifeCycle.id default value is set to Basic.
- The teamTemplate.id default value is set to Default.
- The number default value (which sets the numbering scheme) is set to {GEN:wt.enterprise.SequenceGenerator:WTDOCUMENTID\_seq:10:0}.

By default, the sequence is 10 characters long and the pad character used is 0 as indicated by seq:10:0.


- The MBA|versionInfo default value (which sets the versioning scheme) is set to wt.series.HarvardSeries.

For an explanation of what the numbering scheme and versioning scheme default values mean, refer to the Object Initialization Rules help. To access the help, click the help icon on the **Object Initialization Rules** table.

## Specifying Rules in the Object Initialization Rules Administrator

An object initialization rule is specified in an XML document that is formatted according to the object initialization rules DTD.

You can download this DTD from the **Object Initialization Rules** table, as follows:

1. Access the Object Initialization Rules administrator from the context in which you want to create the rule, as described in [Accessing the Object Initialization Rules Administrator](#).
2. From the **Object Initialization Rules** table, click **Download DTD** .

Your system may be set up to display the DTD in an XML editor or a browser. From the display, you can save the DTD; otherwise, you may be prompted to save the DTD in a file.

To specify rules through the Object Initialization Rules Administrator, complete the following steps:

1. Create the XML document for the rule you want to create, validating it against the DTD you can download, as described in the earlier procedure.

The content of the XML document is described in the next section.

2. Access the Object Initialization Rules administrator from the context in which you want to create the rule, as described in [Accessing the Object Initialization Rules Administrator](#).

3. Click **Create Rule** .

The **Create Object Initialization Rule** window opens.

4. Enter values in the fields provided.

For help on what to enter in the fields, click the help icon; browse to the XML document you created in Step 1 for the value to specify in **XML File** field.

5. Click **OK** to create the rule.

## Content of Object Initialization Rules XML Documents

This section describes the content of XML documents used when specifying rules through the Object Initialization Rules Administrator, as described in the previous section.

Object initialization rules can also be set through the use of templates; however, although the format of the rules in an XML document that can be imported to establish a template is the same, additional elements are needed before and after the rule designation to define import information that is needed. For information on the elements required in XML documents for importing templates, see [Specifying Object Initialization Rules in a Context Template](#) later in this chapter.

For each object initialization rule, the XML document must contain the following:

- The type of object for which the default attribute values are being set. For example, use the following XML for documents, where the object type is wt.doc.WTDocument:

```
<AttributeValues objType="wt.doc.WTDocument">
</AttributeValues>
```

Use the logical identifier for the object type you want to specify. For information on logical identifiers, see [Specifying Windchill Types](#).

- The attributes for which default values are being set. The attributes can include any hard or soft attributes that have been defined for the object type.

An attribute must be named by its logical ID.

For information on attribute logical IDs, see the Type Manager online help that is available from the Type Manager utility.

For example, use the following XML to identify the document folder path:

```
<AttrValue id="folder.id">
</AttrValue>
```

- The algorithms that are used to set the default values; you specify an algorithm for each attribute. Out of the box, Windchill provides the following algorithms (names are shown on multiple lines in the table; enter the name of the algorithm on one line):

Algorithm	Description
com.ptc.core.foundation.folder.server.impl. FolderPathAttributeAlgorithm	Converts folder path into a type instance identifier of the specified folder.
com.ptc.core.foundation.lifecycle.server.impl. LifeCycleTemplateAttributeAlgorithm	Converts the specified life cycle name into a type instance identifier of the life cycle.
com.ptc.core.foundation.team.server.impl. TeamTemplateAttributeAlgorithm	Converts the specified team template name into a type instance identifier of the team template.
com.ptc.windchill.enterprise.revisionControlled. server.impl.NumberGenerator	Implements the numbering scheme identified in the generator function that is specified in the argument.
com.ptc.core.foundation.vc.server.impl. VersionInfoGenerator	Implements the versioning scheme that is specified in the argument.
wt.rule.algorithm.StringConstant	Converts the specified value into a string. This algorithm is not currently used, but is provided as a convenience for those doing customizations.
wt.rule.algorithm.EnumTypeConstant	Converts the specified value into an enumerated value. This algorithm is not currently used, but is provided as a convenience for those doing customizations.

For example, use the following XML to specify the document folder path algorithm:

```
algorithm="com.ptc.core.foundation.folder.server.impl.FolderPathAttributeAlgorithm"
```

- Any additional arguments that are needed to set the default values. Each out-of-the-box algorithm requires one argument that is used to identify the default value.

For example, use the following XML to specify the default document folder path as Default:

```
<Arg>/Default</Arg>
```

The number attribute requires that you specify a generator function as the argument. The function is used to generate the numbers. For details on numbering, see Object Initialization Rules help by clicking the help icon on the **Object Initialization Rules** table.

Optionally, the following AttrValue arguments can also be included:

- Using final="true" as an AttrValue attribute identifies the object attribute value specified for the AttrValue id as the absolute value to use in the composite rule that is formed by merging rules from the associated contexts.
- Using ignore="true" as an AttrValue attribute indicates that when an object of the type specified in the rule is created, no value is assigned to the attribute specified in the AttrValue element.

For details on using these attributes, see Object Initialization Rules help by clicking the help icon on the **Object Initialization Rules** table.

## Example XML Document

The complete example XML document for specifying the default document folder path for a document in the Object Initialization Rules Administrator is as follows:

```
<AttributeValues objType="wt.doc.WTDocument">
  <AttrValue id="folder.id" algorithm=
    "com.ptc.core.foundation.folder.server.impl.FolderPathAttributeAlgorithm">
    <Arg>/Default</Arg>
  </AttrValue>
</AttributeValues>
```

## Specifying Object Initialization Rules in a Context Template

The recommended method for including a specific set of object initialization rules in a context template is to set the rules in an existing container using the Object Initialization Administrator. Then, either save the current context as a template or export the context to a ZIP file on your system. Be sure to select the **Object Initialization Rules** option when you save or export an existing template. After you have a ZIP file containing the object initialization rules (along with any other administrative items you want in the template), import the template using the ZIP file as input. For the details on how to create a context template, see [Creating Context Templates](#).

The XML file containing the object initialization rules in a context template includes the specification XML elements described in the previous sections as well as additional elements that define what is being set through the template. For details on the XML file and elements needed in the input file used when importing a template, see the [Creating Business XML Files for Templates](#) appendix.

# Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects

Numbering schemes control how numbers are generated for parts, documents, CAD documents, and change objects. Each object class has its own numbering scheme.

For all Windchill solutions, a rules load file sets the initial numbering rules for documents, CAD documents, parts, and change objects. These rules affect the following:

- Documents, CAD documents, parts, and change objects created in Windchill Foundation & PDM (if it is installed)
- Documents, CAD documents, parts, and change objects created from within a product, library, organization, or site context (if Windchill PDMLink is installed)

**Note:** Autonumbering must be turned on for the Dynamic Document soft type and its subtypes. See [Configuring Product or Library Containers for Dynamic Documents](#).

- Documents, CAD documents, and parts created from within a project, organization, or site context (if Windchill ProjectLink is installed)



**Caution:** Ensure that the numbering scheme that you want used is in place before allowing users to create documents, CAD documents, parts, and change objects. Changing the numbering scheme after objects have been created can cause problems if the new scheme attempts to reproduce a number that has already been used.

You can view and update the rules that are being used from the Object Initialization Rules table using the Object Initialization Rules Administrator. For details on assessing the Object Initialization Rules Administrator, see [Administering Object Initialization Rules](#).

## Out-of-the-box Numbering Schemes

The loaded numbering schemes for parts, documents, CAD documents, and change objects are simple Oracle sequences that have been loaded into the Windchill database. Each starts at 1 and increments by 1. This means that the numbers for all parts, documents, CAD documents, and change objects are autogenerated according to the numbering scheme that is in place.

## Changing the Numbering of Parts, Documents, CAD Documents, and Change Objects

You can view and update the rules that are being used from the Object Initialization Rules table using the Object Initialization Rules Administrator. For

details on assessing the Object Initialization Rules Administrator, see [Administering Object Initialization Rules](#).

Removing a numbering scheme rule in a specific context causes the numbering scheme rule at the parent context to be used. If you remove the numbering scheme rules in all contexts in the hierarchy, then the following items are true:

- If the context is Windchill PDM, a product, or a library, then manual numbering occurs.
- If the context is the site, an organization, or a project context, then manual numbering occurs for CAD documents and sequences that generate a single-digit integer, starting with 1, are used for documents and parts.

The number of parts and documents that are created in site, organization, and project contexts is always autogenerated. However for parts, users can override the generated number by editing the **Part Number** field in the part creation wizard. You can remove the ability to override the autogenerated part number by adding the following preference using the Preference Manager from either the site or organization container:

```
/wt/part/PJLDisableAutoNumberOverride=true
```

By setting this preference in the Site container, the **Part Number** field is no longer editable when the part creation wizard is run from the site, all organization, and all project containers. By setting the preference in a specific organization container, only the organization and all projects created under the organization are affected by the preference. Deleting the preference or setting it to False allows the user to override the part number. If your site uses the capability to move parts between projects or between a project and a product or library and wants to keep the number assigned to the part, then you should consider setting this preference to True.

## Additional Information about Numbering

See **Object Numbering** in the Object Initialization Rules Administrator help for additional information about numbering parts, documents, CAD documents, and change objects including the following:

- The details on numbering schemes.

**Note:** By default, the sequence is 10 characters long and the pad character used is 0. Do not change the pad character to another numeric value as doing so would change the sequence ordering of the numbers.

- How to create additional numbering schemes.
- Numbering scheme examples.

The **Numbering Schemes Administration** link takes you directly to this help and can be found on the project, product, library, organization, and site **Utilities** page.

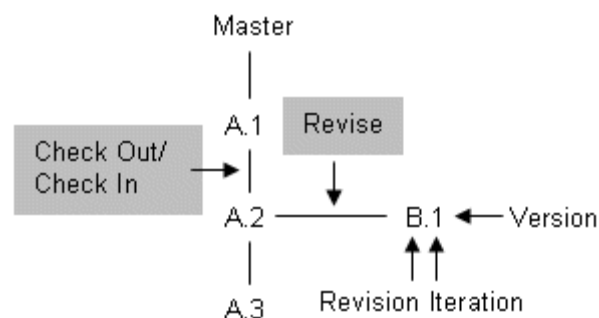
Additionally, you can view the help from the Object Initialization Rules Administrator.

## Administering the Versioning of Parts, Documents, and CAD Documents

Versioning schemes define the labels, or identifiers, that are automatically applied as object versions are created in a Windchill solution.

### Understanding the Use of Versioning Schemes

The following diagram and the sections that follow provide some context on how versioning schemes are used:



#### Master

When users create a document or part, your Windchill solution creates a master as well as an initial version, which is identified in the previous diagram as revision A, iteration 1 (A.1). The master consists of common attributes for all versions, such as part number or document name.

#### Version

The version of a part or document is identified by the combination of revision and iteration.

#### Revision

When users select the **Revise** action, your Windchill solution creates a new version as the start of a new revision of the part or document. For example, if the previous version was A.2, the new revised version would be B.1.

#### Iteration

When users select the **Check Out** action, modify the part or document, and select the **Check In** action, your Windchill solution creates a new version of the part



or document as the next iteration of the current revision upon check in. For example, if the previous version was A.2, the new iteration would be version A.3.

## Initial Versioning Rules

For all Windchill solutions, load files set the initial versioning rules for parts, documents, and CAD documents in the site container. These rules affect the following:

- Parts, documents, and CAD documents created in the Windchill Foundation & PDM Windchill PDM library (if Windchill Foundation & PDM is installed). The versioning scheme used is set when site container rules are established. The scheme is a simple Oracle sequence that has been loaded into the Windchill database. It starts at 1 and increments by 1.
- Parts, documents, and CAD documents created from within a product or library (if Windchill PDMLink is installed). The default versioning scheme used is set when site container rules are established unless you choose to set them in an organization, product, or library container. The out-of-the-box demonstration template named Product Design sets versioning for a product container. For details on the Product Design template, see [Out-of-the-box Product and Library Context Templates](#) in the Administering Products and Libraries chapter.

**Note:** PTC recommends that you establish one versioning scheme for all of your containers and that you set the versioning scheme in the site container. However, you can override the default versioning scheme for a specific organization, product, or library container.

**Note:** Documents, CAD documents, and parts that are created in the following ways are not versioned objects:

- From within a project context (if Windchill ProjectLink is installed)
- From within an organization context or the site context (if either Windchill ProjectLink or Windchill PDMLink is installed)

To change an out-of-the-box versioning scheme, you can change to a scheme that is defined for you or you can define your own scheme. You specify the versioning scheme in an object initialization rule for a particular object type, such as wt.doc.WTDocument.

**Note:** If you make any changes to a versioning scheme, ensure that the revision labels are unique within the entire versioning scheme. Specifying the same label multiple times will lead to corrupt data.



**Caution:** Ensure that the versioning scheme that you want used is in place before allowing users to create documents, CAD documents, and parts. Changing the versioning scheme after objects have been created can cause problems. If the scheme is changed such a way that when revising an existing object, the current

revision is not in the new versioning scheme, then an error occurs. For example, assume that an alphabetic scheme is in place when some objects are created and then the scheme is changed to an integer scheme. When the user tries to revise an object that is at version A, the operation fails because the new scheme does not have the alphabetic character A.

Object initialization rules can be set in context templates or by using the Object Initialization Rule Administrator from within a specific context. For a versioning rule, you must include the following:

- MBA|versionInfo as the AttrValue id
- com.ptc.core.foundation.vc.server.impl.VersionInfoGenerator as the algorithm
- The version series name as the argument to the algorithm

For the details on creating object initialization rules, see [Container Object Initialization Rules](#).

For details on creating templates, see [Creating Context Templates](#).

The following sections describe the Harvard versioning schemes that are available and how to set them up for use within a container.

## Harvard Series Versioning Schemes

There are four Harvard versioning schemes available for your use:

- A simple alphabetic scheme
- A standard integer scheme
- A file-based scheme
- A state-based scheme

A Harvard series is made up of one or more subseries and is typically depicted as a stream of values delimited by a period (for example, 1.1, A.B.A, 1.2.1.1, and so on).

The format of the version identifier in a Harvard series is defined through following properties that are in the wt.properties file:

Property	Description	Default
wt.series.HarvardSeries.delimiter	Specifies the character that is used to separate version identifiers.	. (period)
wt.series.HarvardSeries.depth	Specifies the number of levels supported by the series.	16

Additionally, when using either the simple alphabetic scheme or the standard integer scheme (or a combination of the two), you can specify the scheme to use at each level in the following set of properties:

```
wt.series.HarvardSeries.level.1
wt.series.HarvardSeries.level.2
wt.series.HarvardSeries.level.3
:
wt.series.HarvardSeries.level.<max level number>
```

Where *<max level number>* is the depth of the series specified in the `wt.series.HarvardSeries.depth` property.

The value you can use for each level is either of the following:

Level Value	Description
<code>wt.series.MulticharacterSeries</code>	Uses the <a href="#">Simple Alphabetic Scheme</a> to increment the version identifiers.
<code>wt.series.IntegerSeries</code>	Uses the <a href="#">Standard Integer Scheme</a> to increment the version identifiers.

For example, the following line is the default setting and it indicates that the first level uses a multicharacter series:

```
wt.series.HarvardSeries.level.1=wt.series.MulticharacterSeries
```

If you do not specify a series for each level up to the number specified in the depth entry, the levels you have specified repeat.

**Note:** The file-based and state-based schemes do not use the level properties; all levels use the series as defined in the XML file that is loaded. See [File-based Scheme](#) later in this chapter.

The following sections provide more details on the versioning schemes that are defined.

## Simple Alphabetic Scheme

When using the simple alphabetic scheme, version identifiers are first assigned as a single letter. After all letters from A through Z have been assigned, the following series are used:

AA–ZZ

- AA, AB, AC, ..., AZ
- BA, BB, BC, ..., BZ
- (and so on)

AAA–ZZZ

- AAB, AAC, ..., AAZ
- BAA, BAB, BAC, ..., BAZ
- (and so on)

This simple alphabetic scheme is defined in wt.properties using the following default property settings:

```
wt.series.MulticharacterSeries.min=A  
wt.series.MulticharacterSeries.max=Z  
wt.series.MulticharacterSeries.delta=1  
wt.series.MulticharacterSeries.length=3
```

When a new view version of a part, document, or CAD document is created, by default a second identifier is added using a period as a delimiter. In subsequent view versions, the second identifier is incremented according to the simple alphabetic versioning scheme. For example:

- A new view version of version B is assigned the identifier B.A
- A new view version of version B.A is assigned the identifier B.B
- A new view version of version Z.Z is assigned the identifier Z.AA

As described in the previous section, the use of the simple alphabetic versioning scheme is defined by specifying wt.series.MulticharacterSeries in the wt.series.HarvardSeries.level.<level number> properties. For example:

```
wt.series.HarvardSeries.level.1=wt.series.MulticharacterSeries
```

The out-of-the-box Harvard series that is defined in wt.properties is named wt.series.HarvardSeries and uses the property settings described in this section. The wt.series.HarvardSeries series is used as the default versioning series in the object initialization rules loaded for Windchill Foundation & PDM and Windchill PDMLink.

## Standard Integer Scheme

When using the simple integer scheme, version identifiers are assigned starting with 1, incremented by 1, and can reach a maximum of 2147483647.

This standard integer scheme is defined in wt.properties using the following default property settings:

```
wt.series.IntegerSeries.min=1  
wt.series.IntegerSeries.delta=1  
wt.series.IntegerSeries.max=2147483647
```

When a new view version of a part, document, or CAD document is created using the integer series, by default a second identifier is added using a period as a delimiter. In subsequent view versions, the second identifier is incremented according to the standard integer versioning scheme. For example:

- A new view version of version 2 is assigned the identifier 2.1
- A new view version of version 2.1 is assigned the identifier 2.2
- A new view version of version 235.567 is assigned the identifier 235.568

As described in the previous section, the use of the standard integer versioning scheme is defined by specifying `wt.series.IntegerSeries` in the `wt.series.HarvardSeries.level.<level number>` properties. For example:

```
wt.series.HarvardSeries.level.1=wt.series.IntegerSeries
```

There are no additional properties needed for using the integer series.

To change the out-of-the-box Harvard series (named `wt.series.HarvardSeries`) that is described in the previous section to use the integer series, you need only change the `wt.series.HarvardSeries.level.<level number>` property values to `wt.series.IntegerSeries`. Or you can create an alternate Harvard series that only uses integers or uses a combination of integers and alphabetic characters. For details on how to create an alternate series, see **Object Versioning** in the Object Initialization Rules Administrator help. The **Versioning Schemes Administration** link takes you directly to this help and can be found on the project, product, library, organization, and site **Utilities** page. Additionally, you can view the help from the Object Initialization Rules Administrator.

## File-based Scheme

When using a file-based scheme for versioning, version identifiers are assigned from an XML file that is loaded into the database. The values in the XML file can establish the version identifiers that are used when a part, document, or CAD document is revised.

The XML file that is loaded must have the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<scheme>
  <series name = "name1">
    <value>P1</value>
    <value>P2</value>
    <value>P3</value>
  </series>
</scheme>
```

In the XML file, the series name can contain any alphanumeric characters other than the period (.) or the "at" sign (@). The name you specify in the XML file is the name also used to name the scheme in the versioning rule that is described later in this section.

The values contained in the <value> elements can be any set of values. There is no limit to the number of characters that can be specified in each value; however, if you use lengthy values, the complete object name can be longer than the UI field in which it is displayed. Also, a value cannot contain the series delimiter that is defined in the wt.properties wt.series.HarvardSeries.delimiter property. The default series delimiter is the period (.).

The order of the values in the file determines the order in which the values are used as the version designator when the object is revised.

You can only load one XML file. However, you can include multiple series in the one file. When there are multiple series in the file, each series name must be unique. For example, the following XML file has two series defined:

```
<?xml version="1.0" encoding="UTF-8"?>
<scheme>
  <series name = "name1">
    <value>P1</value>
    <value>P2</value>
    <value>P3</value>
  </series>
  <series name = "name2">
    <value>A_1</value>
    <value>A_3</value>
    <value>A_5</value>
  </series>
</scheme>
```

There are no out-of-the-box object initialization rules loaded that set versioning to file-based versioning, as defined in this section. You can create a file-based Harvard series that uses the file that you load and name the series in one or more versioning object initialization rules.

In the XML file, the series name can contain any alphanumeric characters other than the period (.) or the at symbol (@). The series name you specify in the XML file is the name you must also append to wt.series.HarvardSeries to form the name used in the argument for the versioning rule. For example, if the series name is "name1", then the argument to include is "wt.series.HarvardSeries.name1".

The values contained in the <value> elements can be any set of unique values with the following qualifications:

- There is no upper limit to the number of characters that can be specified in each value; however, if you use lengthy values, the complete object name can be longer than the UI field in which it is displayed.
- A value cannot be empty; it must contain at least one non-white space character.
- A value cannot contain the series delimiter that is defined in the wt.properties wt.series.HarvardSeries.delimiter property. The default series delimiter is the period (.).
- Any white space before or after the value is removed before the value is used.

- The values you specify for the revision labels must be unique within the entire series and specifying the same value multiple times is not allowed; however, you can use the values from one series in another series.

The order of the values in the file determines the order in which the values are used as the version designator when the object is revised.

**Note:** If you make any changes to a versioning scheme, ensure that the values you specify for the revision labels are unique within the entire versioning scheme and that you do not remove any values from a series that are already in use. Specifying the same value multiple times is not allowed. Also, removing a series name or a value from a series after the series has been in use causes problems for any existing objects that use the series or series value. For example, those objects could not be revised and the object iteration history could not be displayed.

For details on how to create this series, see [Setting Up a File-based or State-based Versioning Scheme](#).

## State-based Scheme

The state-based versioning scheme is an extension of file-based versioning. It involves the use of <seed name> elements within the file-based versioning XML file.

In the Windchill PDMLink out-of-the-box XML file (StateBasedVersioning.xml), the seed names correspond to the items in the Life Cycle Administrator **Version Series** drop-down list and are used in the One Phase Development and Two Phase Development out-of-the-box life cycle templates. With this setup, the values are used for versioning when the object is revised in a particular state. If an object moves forward in the state phases (such as from Prototype to Released), then the versioning series being used changes as defined in the life cycle. If a revision or set state action moves the object back to an earlier state (such as from Released back to Prototype), then the versioning series used continues to be the series in use in the most forward state. A new version of an object always uses a higher revision value than its predecessor, even if it is moved to an earlier lifecycle state.

The XML file must have the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<scheme>
  <series name = "name1">
    <seed name = "seed_name1">
      <value>P1</value>
      <value>P2</value>
      <value>P3</value>
    </seed>
  </series>
</scheme>
```

The same general information listed under [File-based Scheme](#) holds for a state-based scheme. In addition, when using the out-of-the-box Life Cycle Administrator **Version Series** drop-down list and using the Two Phase

Development and One Phase Development life cycle templates (which are the default life cycle templates defined in the Windchill PDMLink Product Design template), the seed names you can include are limited to the following:

NUMERIC

MILSTD

For Windchill PDMLink, the loaded XML file that is used in the Product Design template contains the seed names and corresponding values, as follows:

Seed Name	Values in the Seed
NUMERIC	The integers 1 through 99.
MILSTD	The alphabetic characters A through Y, but not including I, O, Q, S, and X.

The loaded XML versioning file is, as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<scheme>
  <series name = "StateBased">
    <seed name = "NUMERIC">
      <value>1</value>
      <value>2</value>
      <value>3</value>
      :
      <value>98</value>
      <value>99</value>
    </seed>
    <seed name = "MILSTD">
      <value>A</value>
      <value>B</value>
      <value>C</value>
      :
      <value>W</value>
      <value>Y</value>
    </seed>
  </series>
</scheme>
```

The Windchill PDMLink load file that loads the Product Design template loads the XML versioning file described earlier in this section and sets the versioning for parts, documents, and CAD documents to use the values in the file. For details on the out-of-the-box versioning in the Product Design template, see [Out-of-the-box Product and Library Context Templates](#) in the [Administering Products and Libraries](#) chapter.



**Caution:** You can modify the state-based versioning values that are established through the XML versioning file. However, all changes should be made before the versioning is used. Changing values that have already been used can cause unpredictable results when versions are being updated. Also, removing a series name after the series has been in use causes problems for any existing objects that



use the series. For example, those objects could not be revised and the object iteration history could not be displayed.

**Note:** You can create additional state-based versioning schemes by adding additional seed names and values to the out-of-the-box file loaded into the database for Windchill PDMLink. However, the out-of-the-box Life Cycle Administrator **Version Series** drop-down list and the One Phase Development and Two Phase Development life cycle templates are not automatically updated to use the additional seed names. To use the additional seed names, you must customize the Life Cycle Administrator and modify the life cycle template you want to use.

Also, the Life Cycle Administrator is not set up to use a series name other than the series name provided in the out-of-the-box XML file that is loaded in Windchill PDMLink. The series name that must be used is **StateBased** unless you customize Windchill PDMLink by updating the wt.series.SeriesRangeSelectorRB.rbInfo file.

**Note:** If you make any changes to a versioning scheme, ensure that the values you specify for the revision labels are unique within the entire versioning scheme and that you do not remove any values from a series that are already in use. Specifying the same value multiple times is not allowed.

**Note:** For Windchill Foundation & PDM, there are no out-of-the-box object initialization rules loaded that set versioning to state-based versioning, as defined in this section. You can create a state-based Harvard series that uses the file that you load and name the series in one or more versioning object initialization rules. However, for the file you load to be used with the Life Cycle Administrator, it must use the same series name and seed names as the Windchill PDMLink file that is described earlier in this section.

The details on how to create or modify a state-based series are the same as those for a file-base series. See the following section for details.

## Setting Up a File-based or State-based Versioning Scheme

To set up a file-based or state-based versioning scheme:

1. Create the XML file containing the series name and the version identifier values for each file revision.
2. Load the XML file into the database.
3. Set the versioning for an object type to the versioning you have set up.

For details on each step, see the following sections.

## Creating the XML File-based or State-based Versioning File

Use an XML editor to create the XML file. For the format of the file, see either of the following:

[File-based Scheme](#)

[State-based Scheme](#)

The file you create can have any name and be located in any accessible directory.

If you have installed Windchill PDMLink, the out-of-the-box XML file that is loaded can be retrieved by entering the following from within a windchill shell:

```
java wt.series.LoadFileBasedSeries <XML_file> -read
```

where *<XML\_file>* is the absolute file path of the file in which the loaded content is placed.

If you have installed Windchill Foundation & PDM, there is no out-of-the-box versioning file that is loaded.

## Loading the XML Versioning File

To load the file-based or state-based versioning XML file, enter the following from within a windchill shell:

```
java wt.series.LoadFileBasedSeries <XML_file> -load
```

where *<XML\_file>* is the absolute file path to the existing XML file.

**Note:** Only one file can be loaded at a time. Loading a second file overwrites the first file.

To save what is currently loaded in an XML file, enter the following:

```
java wt.series.LoadFileBasedSeries <XML_file> -read
```

where *<XML\_file>* is the absolute file path of the file in which the loaded content is placed.

## Setting File-based or State-based Versioning for Objects

For Windchill PDMLink, the default Harvard series versioning scheme is set at the site level and then used at the organization, product, and library levels unless it is overwritten. One context template that is loaded (the demo Product Design template) sets the versioning for parts, documents, and CAD documents to state-based versioning. Additionally, new context templates that are created can set file-based or state-based versioning; however, PTC recommends that you set the same versioning scheme for all organizations, products, and libraries by setting the scheme at the site level.

In a context template XML file, you can specify the versioning you want used for each object type as an element in the corresponding *<TypeBasedRule>* tag for the object type. For example, in the Product Design template XML file, the following

sample <AttrValue> tag sets the versioning for parts to StateBased (which is the series set in the out-of-the-box versioning XML file that is loaded):

```
<TypeBasedRule>
  <EngineRule isDefault="false" enabled="true">
    <ruleName>Part</ruleName>
    <ruleSpecification><![CDATA[
      <AttributeValues objType="wt.part.WTPart">
        :
        :
      <!-- set the version info to a generated version info -->
        <AttrValue id="MBA|versionInfo"
          algorithm="com.ptc.core.foundation.vc.server.impl.VersionInfoGenerator">
          <Arg>wt.series.HarvardSeries.StateBased</Arg>
        </AttrValue>

      </AttributeValues>
    ]]></ruleSpecification>
    <ruleType type="INIT"/>
  </EngineRule>
  <className>wt.part.WTPart</className>
</TypeBasedRule>
```

In this example, versioning for parts is set to wt.series.HarvardSeries.StateBased. The versioning code checks for wt.series.HarvardSeries.StateBased properties and when they are not found, it looks in the loaded versioning XML file for a series named StateBased. If you have named the series you want to use something other than StateBased, then use that name in place of StateBased in the argument.

Additionally in Windchill PDMLink, you can change versioning to file-based or state-based versioning by using the Object Initialization Rules Administrator from within the context where you want the change to take place. However, all changes should be made before a product or library is used. Changing values that have already been used can cause unpredictable results when versions are being updated.

In Windchill Foundation & PDM, the default Harvard series versioning scheme is set by default. You can change the versioning for the Windchill Foundation & PDM to file-based or state-based versioning by using the Object Initialization Rules Administrator. However, all changes should be made before the product is used. Changing values that have already been used can cause unpredictable results when versions are being updated.

The format of the <attrValue> tag that is used for setting file-based or state-based versioning in an object initialization rule is the same as the format shown previously in the template file example. For additional information about using the Object Initialization Rules Administrator, see [Administering Object Initialization Rules](#).

## Updating Versioning Schemes to Conform to the ASME Y 14.35M Standard

PTC supports the use of the ASME Y 14.35M revision standard for revisions; however, none of the out-of-the-box versioning schemes fully implements this standard.

To use a versioning scheme that conforms to the ASME Y 14.35M standard, you can modify either of the following:

- Modify the `wt.series.MulticharacterSeries.MilSpec.skipList` property that is part of the MilSpec multicharacter series and then use this series for versioning as described in [Setting up the MilSpec Multicharacter Series to Conform to the ASME Y 14.35M Revision Standard](#).
- Modify the MILSTD seed values in the state-based versioning scheme XML file and then use this scheme as described in [Setting up the State-based Versioning Scheme to Conform to the ASME Y 14.35M Revision Standard](#).

### Setting up the MilSpec Multicharacter Series to Conform to the ASME Y 14.35M Revision Standard

The MilSpec multicharacter series is defined through the following properties:

```
wt.series.MulticharacterSeries.seriesNames=MilSpec
wt.series.MulticharacterSeries.MilSpec.min=A
wt.series.MulticharacterSeries.MilSpec.max=Y
wt.series.MulticharacterSeries.MilSpec.seed=-
wt.series.MulticharacterSeries.MilSpec.skipList=IOQS
wt.series.MulticharacterSeries.MilSpec.delta=1
wt.series.MulticharacterSeries.MilSpec.length=3
```

To conform to the ASME Y 14.35M revision standard, use the `xconfmanager` to change the following property:

```
wt.series.MulticharacterSeries.MilSpec.skipList=IOQXS
```

To use this multicharacter series:

- Use the Harvard series that is defined through the following properties:

```
wt.series.HarvardSeries.seriesNames=MilSpec
wt.series.HarvardSeries.MilSpec.delimiter=.
wt.series.HarvardSeries.MilSpec.depth=16
wt.series.HarvardSeries.MilSpec.level.1=wt.series.MulticharacterSeries.MilSpec
```

- Define your object initialization rules so the MilSpec versioning scheme is used. For example, to use MilSpec as the versioning scheme for WTPart, set the following rule:

```
<AttributeValues objType="wt.part.WTPart">
<!-- set the version using the alternate MilSpec HarvardSeries -->
  <AttrValue id="MBA|versionInfo"
algorithm="com.ptc.core.foundation.vc.server.impl.VersionInfoGenerator">
    <arg>wt.series.HarvardSeries.MilSpec</arg>
  </AttrValue>
</AttributeValues>
```

## Setting up the State-based Versioning Scheme to Conform to the ASME Y 14.35M Revision Standard

Setting up the state-based versioning scheme to conform to the ASME Y 14.35M revision standard involves editing the PTC-supplied file named `StateBasedVersioning.xml` and then loading the updated file.

The changes to make to the `StateBasedVersioning.xml` file are as follows:

- Include the hyphen character (-) as the first value in the MILSTD seed.
- After Y, add AA, AB, ... AY (skipping AI, AO, AQ, AS, and AX) as values in the MILSTD seed. At the same time, you can also add BA, BB, ... BY (skipping BI, BO, BQ, BS, and BX) if you want to make these values available in the series.

For details on obtaining a copy of the `StateBasedVersioning.xml` file and loading the updated file, see [Setting Up a File-based or State-based Versioning Scheme](#), earlier in this chapter.

## Searching for Principals

The following administrative clients use a common interface when searching for principals (users, groups, and organizations):

- Policy Administrator -- When working with policy rules, you select principals against which the rules are applied.
- Life Cycle Administrator -- When defining a life cycle template, you can select principals as participants for any of the roles defined for that life cycle template.
- Team Template Administrator -- When defining teams, you can select principals as participants for roles.
- Workflow Administrator -- When defining activities, you can select principals to complete each assigned activity.

The method used to display users, groups, or organizations within an administrative client participants window is determined by the setting of the `wt.org.client.applet.ActiveSearch` property. By default, the property is set to false and the interface initially displays no users, groups, or organizations. When this is the case, the user enters values to restrict the search and clicks **Find** to display a subset of users, groups, or organizations. By setting `wt.org.client.applet.ActiveSearch=true` in the `wt.properties` file, you can change the behavior of the interface so that when the participants window appears, it is automatically populated with all groups or organizations that are available.

**Note:** If you have a large number of groups or organizations, then automatically populating the participants window with them may take a long period of time. PTC recommends that you use the default setting for

wt.org.client.applet.ActiveSearch unless you have a small number of groups and organizations.

In these administrative clients, the ability to locate users, groups, and organizations is determined by the administrator's access permissions and by the search scope used within each of the services searched. The administrator must have Read permission for a principal to see that principal in a search result.

The service's default search scope is used to locate users and organizations (that is, the search scope configured in the service's JNDI adapter).

In the administrative clients, the search scope used to locate groups is determined by the context, as follows:

- Site context:
  - Searches the bundled Aphelion directory service at the site container's base node, limiting the search to the current directory hierarchy level.
  - In other services, searches with the default search scope.
- Organization context:
  - Searches the bundled Aphelion directory service, as follows:
    - Searches the public node, limiting the search to the current directory hierarchy level. The public node is where the groups created under the organization context are located.
    - Searches the Site container's base node, limiting the search to the current directory hierarchy level.
  - In other services, searches with the default search scope.
- Windchill Foundation & PDM library context:
  - Searches the bundled Aphelion directory service, as follows:
    - Searches the Windchill Foundation & PDM Library container's base node, limiting the search to the current directory hierarchy level.
    - Searches the parent organization's public node, limiting the search to the current directory hierarchy level.
    - Searches the site container's base node, limiting the search to the current directory hierarchy level.
  - Searches all other services with their default search scope.
- Windchill PDMLink product and library context:
  - Searches the bundled Aphelion directory service, as follows:
    - Searches the applications container's node for internal access groups with subtree scope. This scope starts the search at the current level

and searches all levels of the complete LDAP hierarchy below the current level.

- Searches the parent organization's public node, limiting the search to the current directory hierarchy level.
  - Searches the site container's base node, limiting the search to the current directory hierarchy level.
- In other services, searches with the default search scope.

The bundled Aphelion directory service is set up during installation and uses the bundled JNDI adapter. For additional information about setting up directory services and JNDI adapters, see the Configure Windchill to Use an Enterprise Directory chapter in the *Windchill Installation and Configuration Guide - Windchill*.





# 4

## Administering the Site

This chapter provides an overview for administering the site and describes the typical duties that a site administrator performs. It also provides additional information about some of the main administrative tasks for sites.

Topic	Page
Overview .....	4-2
Typical Duties of Site Administrators.....	4-2
Business Object Uniqueness Considerations .....	4-11
Out-of-the-Box Site Container Configuration.....	4-15
About the Site Tab.....	4-16
Best Practices .....	4-16

## Overview

The site administrator manages the organizations within the site and is responsible for the common information and rules inherited by all containers in the site.

The site administrator (for example, wadmin) created at installation time may be a temporary administrator specified only to get the system up and running for the customer. If that is true, create a user who is associated with the hosting organization and add that person to the Administrators group. That person will then take over as the site administrator once the quick start program is completed.

For general information about container contents and how to create containers, see [Administering Containers](#).

**Note:** The general information in this chapter pertains to all Windchill systems, but is more specific to Windchill PDMLink and Windchill ProjectLink. The site administrator for Windchill PDMLink and Windchill ProjectLink performs these actions from the **Site** tab of the user interface.

## Typical Duties of Site Administrators

Site administrators are responsible for the configuration and management of the Windchill system as a whole. They create organizations representing business units of a hosting company and organizations representing partners and suppliers. In an exchange environment, organizations represent those companies willing to pay to use the service. By making an organization a subscriber, members of that organization have the ability to create libraries, products, and projects. Site administrators also control how authorized users are added to the system. They also define the information that is common across all organizations and their products, libraries, and projects.

Responsibilities of the site administrator include the following:

- Create and update organizations participating in the site.
- Manage overall exchange access (control users that have access to the exchange service).
- Manage a group of users with site administration privileges.
- Manage site-level folders, documents, and links.
- Manage site-level access policies and rules.
- Manage site-level types and type-specific attributes that are inherited by all contexts in the site.
- Manage the rules governing item creation.
- Manage site-level templates that are inherited by child contexts.
- Perform security audits to track access to specific products, projects, documents, and parts.

- Manage site configuration (such as vaulting, replication, calendar, property options, and preferences).
- Manage processes (such as workflow, CAD viewable publishing, and replication).
- Export and import site-level information.
- Define and manage reports.
- Determine searchable object types.
- Manage saved searches.

## Creating and Managing Organizations

Windchill Foundation & PDM installations create an organization object and associate the object to an organization container. In Windchill Foundation & PDM, you cannot create additional organization containers.

Windchill PDMLink installations create an organization object that is associated with the site container and the default data loading process creates the organization container. If the data loading process was modified so that the organization container is not created, you must create an organization container before users access the solution.

Windchill ProjectLink installations create an organization object that is associated with the site container, but do not create the organization container. Before users access the solution, create an organization container. You can associate it with the organization object that was created during the installation, or you can create a new object.

Users who have an organization attribute (the "o" attribute, by default) that matches the organization name automatically become members of the organization. If your site does not use the organization attribute in the directory service entry, users can be assigned to an organization using the `usersOrganizationName` property. For more information on using this property, see the *Configuring Windchill to Use an Enterprise Directory* chapter of the *Windchill Installation and Configuration Guide - Windchill*.

**Note:** You can create organization containers from either the **Site** tab or the **Organization** tab. There is no difference in the type of organization created; it is a convenience to be able to create an organization from the **Organization** tab.

When you create an organization, you can grant privileges so the organization can create its own projects, products, or libraries, or you can restrict participation to only products, projects, and libraries created by a parent company that is hosting the organization.

**Caution:** To avoid system stability issues, it is important that the site administrator user, such as `wcadmin`, not be used to create contexts (projects,

products, programs, or libraries) within organizations, nor should they create objects within those contexts.

When you create an organization, you can grant its members full access to a corporate directory with which the system is configured. The purpose is to allow employees of a company that is hosting the Windchill system the ability to search their entire directory for users and groups.

You can review the list of organizations in the site and navigate to update each of the organizations. A company will probably choose to administer the organizations representing their partners or customers. In this way, you become the organization administrator on behalf of any number of organizations defined in the site.

When a Windchill solution is installed and an organization container is created, then this organization automatically owns the parts and documents that are created under the organization context. In all Windchill solutions, you can create additional organization objects (using the Principal Administrator) and allow users to select the organization that owns parts and documents. To change the out-of-the-box functionality so that a user who creates a part or document can specify which organization owns the part or document by specifying the organization ID, see Administering Runtime Services in the *Windchill System Administrator's Guide*.

## **Adding and Updating Members**

You control how members are added to the site to enable them to log on and view and author information in the site. You can use several options. One option is for you to add each member to the site using the Principal Administrator utility. Another option is to establish a workflow process for requesting and approving members. You can also define the users who have site administration privileges so that several individuals in the company can play this role.

For Windchill ProjectLink sites, you can allow users to register to the site through a registration interface and then automatically add the users as members when they complete the registration.

For more information about using the Principal Administrator utility, see [Using the Principal Administrator](#) in Administering Principals later in this manual.

## Creating and Managing Site Folders and Documents

You can define documents, folders, and links within the site context. The site folders are designed to hold any documents that are important for administering the site.

Types of documents that administrators might define at the site include the following:

- System configuration documentation
- System change log that captures a record of changes made to the system
- Operation rules and procedures (such as shutdown, backup and restart procedures)
- System administrator responsibilities document
- Key contact list for system administrators
- Deployment schedule and plans (this might also be defined in a project but referenced by providing a link in the site folder)
- Documents describing site-level document, life cycle, and workflow templates

By default, Windchill PDMLink requires that you select a file containing the primary content of a document when you create, update, or check in a document.

You can change this requirement by setting the following property in `wt.properties`:

```
wt.doc.primaryContentRequired=false
```

Use the `xconfmanager` to set the property and then restart the method server. Users can then create, update, or check in documents without selecting any primary file content. The asterisk that indicates a value is required is no longer displayed and the form submission validation allows the **File** field to be empty.

## Changing Default Configuration Options for Windchill PDMLink and Windchill ProjectLink

In both Windchill PDMLink and Windchill ProjectLink, you can use properties and preferences to change the configuration options that are set by default.

For example, if the folder browser performance is too slow, consider changing the folder configuration options. You can change these options through the following properties located in the `wt.properties` file:

```
com.ptc.netmarkets.folder.showDiscussIconPJJL
```

- Determines whether the project Discuss icon appears or is hidden
- Default: true; show the Discuss icon

com.ptc.netmarkets.folder.showDeliverableIconPJJL

- Determines whether the project Deliverable icon appears or is hidden
- Default: true; show the Deliverable icon

com.ptc.netmarkets.folder.showRoutingIconPJJL

- Determines whether the project Routing icon appears or is hidden
- Default: true; show the Routing icon

com.ptc.netmarkets.folder.showDiscussIconPDM

- Determines whether the product and library Discuss icons appear or are hidden
- Default: false; hide the Discuss icons

Additionally, you can change the cache size used when displaying content in a Windchill solution through the following properties:

wt.folder.ResultsLimit

- Limits the number of objects that can be displayed in the **Folders** table. If the limit is exceeded, a message informs the user to use search or go to a folder details page to access the object they are looking for. A maximum number of objects that can be displayed is set so that the server does not become unresponsive when a user tries to retrieve a large product, library, or project. When the limit is reached, the user should pick a subfolder to reduce the size of the page.
- Default: 15000 objects

com.ptc.netmarkets.serverCacheEnabled

- Determines if caching is used.
- Default: true; use caching

You can find the details for these properties and others that you may want to change for performance reasons in the properties.html file located in your codebase. When changing properties, always use the xconfmanager utility (or System Configurator) so that your site changes are maintained over maintenance releases.

The Preference Administrator helps you maintain configuration preferences as well as many other types of preferences. For example, the following preferences provide configuration options for folders:

/wt/admin/displayDomains

- Determines the visibility of the domain a folder belongs to when creating, updating, and displaying the information about a folder. For details on

using this preference, see [Assigning Domains to Folders in Windchill PDMLink or Windchill Foundation & PDM](#).

- Default: false; do not show the domain

/ProjectLink/PDMLinkFoldersAndContentsView

- Determines whether **Folders and Contents** appears in the **Current View** drop-down list for the **Product Folders** table and the **Library Folders** table in Windchill PDMLink. If the preference is set to true, the list includes **Folders and Contents**.
- Default: false; do not show **Folders and Contents**.

For details on using the Preference Administrator, see the *Windchill System Administrator's Guide*.

## Managing Site-level Types and Type-specific Attributes

You can define object types and type-specific attributes to make available to all organizations in the system. For example, a company might define a change impact report document type with specific attributes for each of several categories, including replacement cost, production tooling cost, and so forth.

You can associate document types with life cycles that identify the various states of maturity of the document.

**Note:** If you create a type and associate it with a default routing life cycle, there may be problems when users create documents using templates created with the type. The default routing life cycles start automatically, so when the user tries to create from a template using one of these life cycles and selects Checkout and download, the checkout and set state (life cycle) gets caught in a race condition. This can cause a couple of outcomes:

- The set state fails; the user has a checked-out document that is In Work, and the life cycle does not happen.
- The set state succeeds and the user has a checked-out, newly-created document that has been immediately routed for review.

PTC recommends you create your own routing life cycle that would contain a specific Submit task later in the process, not an automatic Submit upon creation.

## Managing Site-level Templates

A company may choose to define a number of document, life cycle, and workflow templates that they want all their business units and partners to use. For example, the site may want to define a document template and associated life cycle and workflow process for capturing enhancement requests for the Windchill system. Or a company may want to make a corporate presentation template available to all its business units collaborating in the exchange.

The site can define project, product, library, and organization templates. Organization templates are required to create organizations.

You can configure how objects of various types are created and their attributes initialized using a rules administration utility. This utility defines how the initial values for the object are established and can determine basic relationships, such as life cycle association, when an item is created. The rules established by the parent site are inherited by each organization by default, but they can be overridden by an organization. The rules are defined in an XML format, which the site administrator can view and edit.

## **Auditing System Information**

For security or other reasons, you may need to examine the audit logs for specific members, or to determine who has accessed or modified a particular document, part, project, or product. If there is concern that a user obtained access to projects or products to which that user should not have been invited or granted access to, an audit history record for the user can be reviewed to determine which projects, products, and documents the user viewed or updated.

You must enable auditing to ensure the audit events of interest are recorded by the system. See [Enabling Auditing](#) in the Administering Audit Reports chapter later in this manual for more information. Administrators should audit only the events that are necessary because the audit record consumes a significant amount of database table space.

## **Creating and Managing Access Control Policies**

You can define policies that control the level of access to information to the system. For example, as site administrator, you may want to provide read access to all documents of type Engineering Specification to an engineering group. You need to first define an Engineering Group and populate it with the appropriate members, then define a document type of Engineering Specification at the site level, and then use the Policy Administrator at the site level to define the access policy based on the document type, the group or groups provided access and the access level.

You should create only those site-level policies that provide broad access for types of information defined at the site level for all organizations in the system.

For more information about access control, see [Administering Access Control](#) later in this manual.

## **Configuring External Vaults or Replication Sites to Optimize Performance**

You can configure external file vaults so that document and part content is stored on a file system rather than in the database. This type of configuration can provide significant upload performance improvements and is appropriate when the site is frequently used to exchange large files (such as CAD model files).



By default, external storage rules are based on individual domains in each container. You can launch the External Storage Administrator client from the **Utilities** page in the context of a product, a library, or the site. (The client is not available from the context of a project or organization.) When launched from a product or library context, product and library managers can set only the vaulting rules. The additional functionality within the External Storage Administrator is available only to site administrators. The client allows you to create vaulting rules for the domains - System and Default - associated with the context in which the client is launched.

**Note:** The rules needed for setting up an external file vault for a product or library cannot be inherited from the site.

If the increased number of vaults and file vault rules becomes unmanageable, you can force vaulting to be accomplished through a single vault by setting the `wt.fv.forceContentToVault` property to true. For how to set external file vault rules or set up a single vault, see Administering External File Vaults in the *Windchill System Administrator's Guide*.

You can also configure replication sites so the document and part content files are replicated at a remote site where local users have only very low bandwidth connections to the Windchill server. You can launch the Replication Administrator client from the **Utilities** page in the context of a product, a library, a project, or the site. (The client is not available from the context of an organization). As a site administrator, you would typically define only the replica site and the replication schedule. The product, library, or project managers would configure the replication rules for a particular site. When launched from a product, library, or project context, only the creation of replication rules is available to product, library, or project managers. The additional functionality within the Replication Administrator is available only to site administrators. For how to set replication rules, see Administering Content Replication in the *Windchill System Administrator's Guide*.

## Configuring Numbering and Versioning Schemes and Units of Measure

You can configure the number and version scheme used to uniquely identify parts, documents, and other items in the system. The set-level schemes and units of measure are inherited by the organizations, but each organization can optionally define its own schemes. In general, when a company is hosting Windchill for its internal use, the numbering, versioning, and units of measure should all be defined at the site level and should not be overridden by organizational units. This approach ensures consistence of basic identification and units across the company.

For additional information on numbering and versioning, see [Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects](#) and [Administering the Versioning of Parts, Documents, and CAD Documents](#) in the [Administering Containers](#) chapter.

For more information on units of measure, see the online help for the Measurement System Manager. For Windchill PDMLink and Windchill ProjectLink, the Measurement System Manager is accessible on the **Utilities** page of the **Site** tab. For Windchill Foundation & PDM, the Measurement System Manager is accessible in the **Attribute Administrator**, found in the **Business Administration** section of the **Site Map**.

## Configuring and Managing CAD Publishing Utilities

You can configure CAD workers that publish viewables for CAD models that can be accessed by participants that do not have native CAD authoring tools. The site includes utilities to configure the CAD workers and monitor and manage the publishing schedules and queues.

For additional information, see [Administering Visualization Services](#)

## Creating, Updating, and Managing Custom Reports

You can create and update custom reports against the objects and attributes in the system as a whole.

## Importing and Exporting Information Among Systems

You can exchange information between a staging server and production server, between servers, or between a server and a file system using the Windchill import/export utilities. These utilities read and write system information in an XML format. For information on exporting and importing XML template files, see [Creating Containers](#).

The workflow and life cycle administration utilities accessible to the site and organization administrator integrate the import and export functions with administering workflows and life cycles. For life cycle import and export information, see [Import and Export](#) in the Administering Life Cycles chapter. For workflow import and export information, see [Import and Export](#) in the Administering Workflows chapter.

## Managing Overall System Configuration and Preferences

There are many system configuration settings you can view, set, and update remotely using the system preferences and system configuration utilities. Use these utilities to view and set system properties, view and manage queues, and view server status and logs.

For more information, see the *Windchill System Administrator's Guide*.

## Monitoring Enterprise Systems Transactions Log

If you configured Windchill to exchange information with an Enterprise Resource Planning (ERP) system, you can monitor the transactions with the ERP system through the Enterprise Systems Transactions Log.

For more information, see the *Windchill Enterprise Systems Integration Administrator's Guide*.

## Purging, Archiving, and Restoring Jobs

You can create and manage purge jobs with the Purge Manager. If you have the applicable functionality installed, you can create purge jobs to permanently remove data from the Windchill system or create an archive of the data to restore at a later time.

## Managing Searches

You determine the searchable object types for the site. Out-of-the-box, the default is to search name and number; however, you can determine other fields to be searchable. For more information, see the Adding a New Type to the Search User Interface section in the *Windchill Customizer's Guide*.

You can search for and delete searches created by other users. This can be useful if a user or other administrator has created saved searches, but has since left the company.

To delete saved searches created by another user, click **Administrative Delete** on the **Saved Searches** table to open the **Find Searches For Delete** window. Specify your search criteria and click **Search**. You can use wildcards to broaden your search. Select the searches to be deleted from the search results, and click **Delete**. For more information, see the help available from the **Saved Searches** table.

You can determine whether a search from the **Search** page includes only the latest version of an object, rather than all versions. To search for only the latest version of an object, set the following preference to TRUE using the Preference Manager on the **Site** tab **Utilities** page:

```
/com/ptc/windchill/enterprise/search/latestVersionDefaultSearch
```

## Business Object Uniqueness Considerations

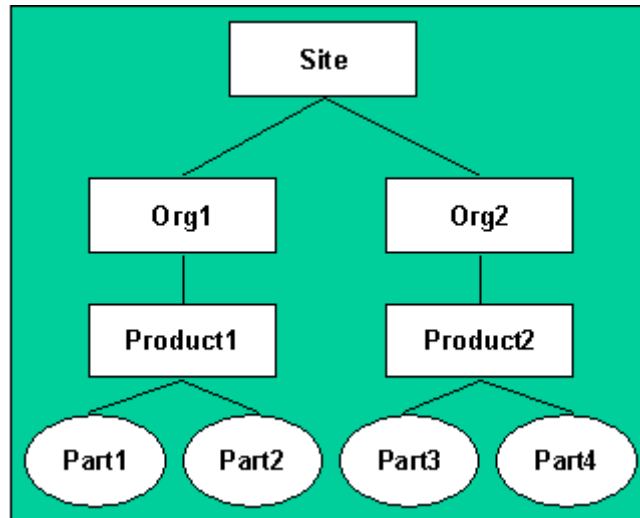
When your Windchill PDMLink system was installed and configured, choices were made regarding the uniqueness of business objects, such as parts and documents, when you have multiple organization contexts within your site: whether business objects must be unique across all organizations, or whether they can be unique within each organization.



**Caution:** The configuration of a Windchill system to allow business object uniqueness within each organization must occur as a part of the installation and

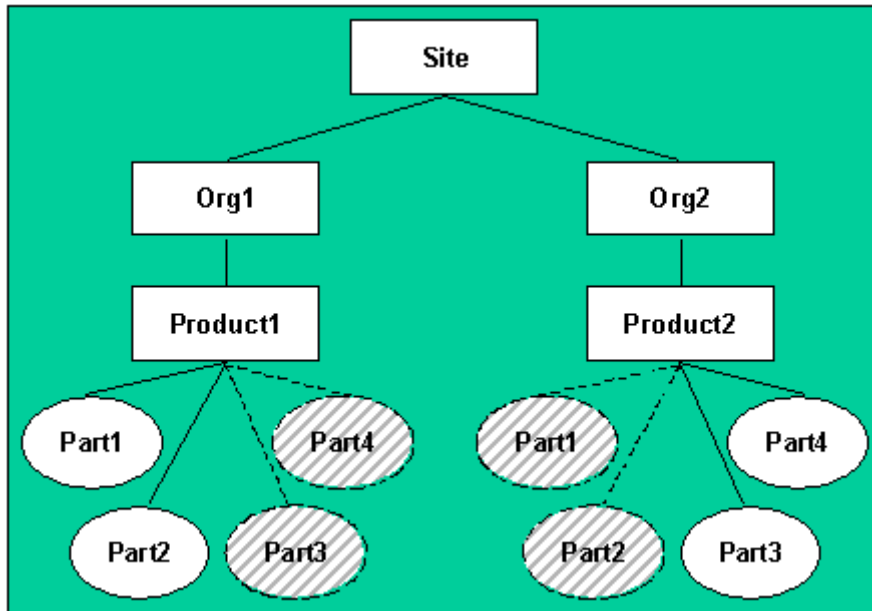
configuration process, before any data is loaded. Changing this configuration after data has been loaded will result in data corruption and should not be attempted.

In a default configuration with multiple organizations, business objects such as parts and documents must be unique across all organizations within the site, as shown in the diagram below. The colored area indicates the space within which each business object must be unique, also called the *business namespace*.



Since each business object must be unique across all organizations within the site, there cannot be a Part1 or Part2 created in Org2, because Part1 and Part2 already

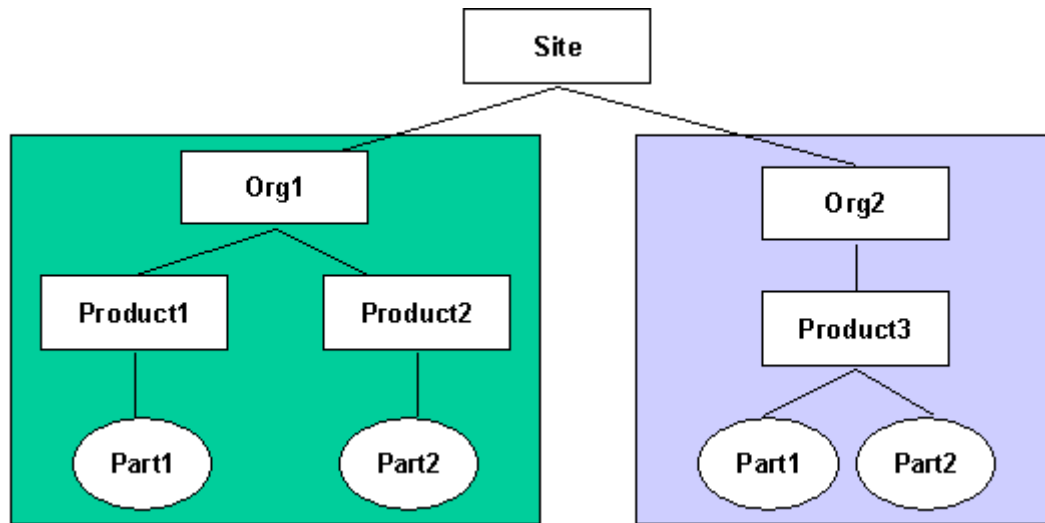
exist within Org1; similarly, since Part3 and Part4 already exist in Org2, a user could not create a Part3 or a Part4 in Org1, as shown in the diagram below.



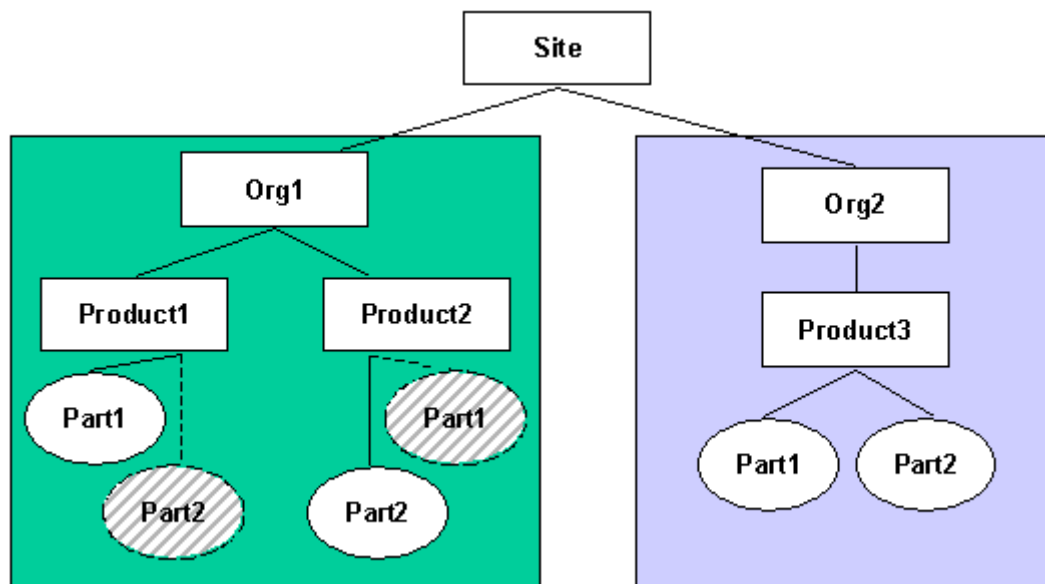
**Note:** CAD documents must be unique across all organizations within the site, regardless of the business object uniqueness configuration. If CADDoc1 exists within Org1, a user cannot create a CADDoc1 in any other organization within your site.

If your system was configured to allow business objects uniqueness by organization, however, the organization becomes part of the uniqueness constraint for each business object. As a result, the uniqueness of a business object is constrained within each organization, rather than within the entire site. This allows for a Part1 to be created in Org2, regardless of whether a Part1 exists in

Org1. In the following diagram, the colored areas again indicate the space within which business objects must be unique.



A part named Part1 can exist within both Org1 and Org2 because it is unique within each organization. Within Org1, however, a Part1 cannot be created in Product2 because there is already a Part1 in Product1, as shown in the diagram below.



Regardless of the configuration, users can search for all objects across all organizations in the site, limited only by their access controls and other site configurations. If your site is configured for business object uniqueness within

each organization, however, business objects from one organization cannot be selected from the search results and used within a different organization. For example, a part from one organization cannot be added as a child part to a parent part that is from a different organization.

Even though Windchill ProjectLink can be installed in combination with Windchill PDMLink and configured with business objects uniqueness within each organization, the uniqueness of objects within projects is not affected. Objects created within a project context are always unique within that project.

If you do not know whether your site was configured with business object uniqueness by organization, do one of the following:

- check with the person who installed your Windchill system.
- see the *Windchill Installation and Configuration Guide - Windchill* for the business object uniqueness configuration details, and determine if they have been implemented.

## Out-of-the-Box Site Container Configuration

When Windchill PDMLink or Windchill ProjectLink is installed, the following are defined for the site:

- Container structure; for more information, see [Installed Site Container Structure](#).
- Container participation; for more information, see [Installed Site Container Participation](#).
- Container policies; for more information, see [Installed Site Container Policies](#).
- Container data; for more information, see [Installed Site Container Data Types and Attributes](#).
- Container templates; for more information, see [Installed Site Container Templates](#).
- Container rules; for more information, see [Installed Site Container Object Initialization Rules](#).

## About the Site Tab

For Windchill PDMLink and Windchill ProjectLink, the **Site** tab provides access to the following pages:

Page	Description
Organizations	Allows you to create, view, update, or subscribe to organizations.
Folders	Allows you to view information about and add information to folders. This is where the site administrator can store information needed for administering the site.
Administrators	Allows you to view, add, or remove users from the site administrators group.
Types	Allows you to create and update document types. This page also provides access to the Life Cycle Administrator, Attribute Manager, and Type Manager.
Templates	Allows you to view, create, or update templates.
Audit Reports	Allows you to create and update custom reports against the objects and attributes in the system. The Audit Reports link appears only if the feature has been enabled. See <a href="#">Enabling Auditing</a> for information about enabling Audit Reports.
Utilities	Allows access to utilities available to your site

## Best Practices

### For Windchill PDMLink

#### Setting Object Initialization Rules

The object initialization rules include the ability to set the following for each object type:

- Folder paths
- Default life cycle template
- Default team template
- Default numbering scheme
- Default version scheme



Although these rules can be set for individual product and library containers, PTC recommends that you set them at either the organization or site level so that an entire organization or site is using the same set of rules.

## Setting Up Enhanced Life Cycle Templates

The out-of-the-box Product Design template includes object initialization rules that set different default life cycle templates and versioning schemes for parts, documents, and CAD documents than is set out-of-the-box at the site level. This template demonstrates the use of simplified life cycles and state-based versioning. Unless you have overriding business practices that require unique rules, PTC recommends that if you want to use these features, you modify the site object initialization rules and remove them from the Product Design template.

## Setting Delimiters for Subscription Name Field

When subscribing to parts and documents, a comma ( , ) is used as the delimiter for entering multiple user names in the Name field of the Subscription window. If your site uses commas as part of your user name (for example: Smith, John), you can choose a different character to use as a delimiter for this field, such as " | " or " & ".

To set a different character as the delimiter, create a new site preference using the Preference Manager from the Site tab Utilities page:

```
PDMLink/subscriptionUserDelimiter
```

Clear the **Override Allowed** check box. For further details on using the Preference Manager, see the *Windchill System Administrator's Guide*.

## Overriding and Reassigning Life Cycle and Team Templates

The life cycle and team templates for objects such as parts and documents can be manually selected during object creation. For existing objects, life cycles can be reassigned and team templates can be reset.

For more information, including the preferences used to enable this functionality, see [Overriding and Reassigning Life Cycle and Team Templates](#).

## For Windchill ProjectLink

### Setting Default Preferences in Windchill ProjectLink to Collapse Tables

In order to cut down the time it takes to load tables, you can create a site preference to collapse tables for document and part iteration history pages.

The following can be set to collapse tables by default.

For a document iteration history table:

```
name: "history$doc_list$::ccomp" and value: "1"
```

For a part iteration history table:

```
name: "history$part_list$::ccomp" and value: "1"
```

For details on creating a preference, see the *Windchill System Administrator's Guide*.

You can use the xconfmanager utility to add the following wt.property to make all components collapsed:

```
com.ptc.netmarkets.defaultTablesCollapsed=true
```

For more information about using the xconfmanager, see the [Administration Overview](#).

# 5

## Administering Organizations

This chapter provides an overview for administering organizations and describes the typical duties that an organization administrator performs. It also provides additional information about some of the main administrative tasks for organizations.

Topic	Page
Overview .....	5-2
Typical Duties of Organization Administrators .....	5-3
Out-of-the-Box Organization Templates.....	5-7
Creating an Organization.....	5-13
Using the Organization Utilities Page.....	5-15
Changing an Established Internet Domain .....	5-16
Using the OrganizationSync Utility for User Organization Changes .....	5-16
Best Practices .....	5-18

## Overview

Organization administrators are responsible for the configuration and management of an organization within the Windchill system. The organization may represent a business unit of the parent company hosting the Windchill system or it may represent a supplier or partner to the parent company. In an exchange environment, an organization represents those companies paying for the ability to create projects.

Windchill solutions use organization principals (made up of an object of type WTOrganization and a directory service) and organization containers when administering organization information.

The development of products and the subsequent management of product information throughout their entire life cycle is truly a collaborative process involving a number of organizations, including suppliers, contract manufacturers, and design partners. The Windchill solutions use organization containers as follows:

- To define your digital product value-chain.
- To define data ownership responsibilities.
- To define the level of engagement that organizations have within your system and business processes.

All Windchill solutions, when configured, contain a host organization. This organization represents your enterprise and is associated with an organization container through the organization principal that was selected or created when the container was created. By using the organization container, the users in the host organization either author product information or in some way are consumers of this information.

Windchill Foundation & PDM has only one organization container (and corresponding organization principal) that is created during installation. Additional organization principals can be created using the Principal Administrator, but no additional organization containers can be created.

In Windchill PDMLink and Windchill ProjectLink, organization containers (and corresponding organization principals) can be created for each of the business organizations and or business units that are collaborating together through the Windchill solutions. Each organization inherits templates (document, workflow, and life cycle templates) and groups defined in the parent site container and then defines its own organization-specific templates, groups, types and roles. A separate group of administrators is associated with each organization to manage the organization templates, groups, and policies. The organization administrator can control who is allowed to create application containers (products, libraries, and projects) within their organization.

Windchill PDMLink and Windchill ProjectLink provide client user interfaces for doing most activities that are related to administering organizations. Organization administrators define the information that is common across all products, libraries, and projects hosted within the context of their organization.

This chapter contains information that an organization administrator needs to know, as well as information that a site administrator needs in order to get the organization functional.

## Typical Duties of Organization Administrators

Responsibilities of the organization administrator include the following:

- Managing organization, groups, and roles
- Creating, updating, and managing organization folders and documents
- Managing organization-level types and type-specific attributes
- Managing organization templates and object creation rules
- Auditing activities within the organization
- Creating and managing access control policies
- Configuring numbering and versioning schemes
- Monitoring and managing viewable publishing
- Viewing project reports
- Importing and exporting information
- Purging, archiving, and restoring jobs

## Managing Organization Members, Groups, and Roles

You control the users that can administer the organization (organization administrators) and those that can create and administer products, libraries, and projects. If Windchill ProjectLink is installed, the organization has a project creators group. If Windchill PDMLink is installed, the organization has product and library creators groups. Only members of the organization can be added to the project, product, or library creators groups. Organization administrators and site administrators who are members of the organization can create projects, products, and libraries; otherwise, you must be a member of the project creators group to create projects, a member of the product creators group to create products, or a member of the library creators group to create libraries in the organization.

In Windchill ProjectLink, all members of the organization, by default, are allowed to create projects and administer the projects they create. However, if the organization is set up so that members are not automatically added to the project creators group and empowered to create projects, you must manually add members to the project creators group. Projects are considered the least formal of

the application contexts (product, project, and library), so it is generally appropriate to allow all users to create and administer a project.

You can create groups at the organization level that can be available when members create product, library, and project teams or when access policies are defined. For example, an organization may want to define groups for each of the functional teams with membership in the organization. An organization might define a sales and marketing group, an engineering group, a publications group, and a quality control group. These groups can then be invited to a product or project team without adding each member individually. Furthermore, when groups are updated, the updates can be refreshed to update all the teams referencing the groups without the need for each project or project manager to update their team membership. For additional information about updating team membership, see [Best Practices for Windchill PDMLink and Windchill ProjectLink](#) in the [Administering Teams and Roles](#) chapter.

An organization inherits the roles defined at the site (as defined in the system roles resource bundle). All the roles from the RoleRB.rbInfo file are displayed on the Organization Roles page. Organization administrators explicitly remove the roles that they do not want their organization to use.

## **Creating, Updating, and Managing Organization Folders and Documents**

You can define documents, folders, and links within your organization. The organization folders are designed to hold any documents that are important for administering the organization. The following are examples of the types of documents that administrators may define at the organization level:

- Organization configuration documentation
- Organization environment change log that captures a record of changes to the organization
- Organization administration rules and procedures
- Internal training information for organization administrators
- Key contact list for organization administrators
- Documents describing organization-level types, document templates, life cycle templates, and workflow process templates.

## **Managing Organization-level Types and Type-specific Attributes**

Organization administrators can create types; however, they cannot create attributes. You can use existing attributes created by the site administrator and link them to types for your organization.

For more information about types and attributes, see [Using Types and the Type Manager](#).

## Managing Organization Templates and Object Creation Rules

An organization can define a number of document, life cycle, and workflow process templates for all its members to use. To ensure consistency and maximize efficiency, an organization may want to define templates for specifications, presentations, reports, proposals, meeting minutes, and so forth. Life cycle and workflow process templates may be associated with each of the defined templates. Each organization inherits the templates defined in the site and can either use these site-defined templates or override them by defining organization-specific templates with the same name.

The organization can inherit the project, product, and library templates defined at the site, or you can override these templates by defining templates for your organization. If Windchill ProjectLink is installed, you can create project templates. If Windchill PDMLink is installed, you can create product and library templates.

You can define a set of default values to configure how objects of various types are created and their attributes initialized using the Object Initialization Rules Administrator utility. With this utility, you can define how the initial values for the object are established and determine basic relationships, such as life cycle association, when an object is created in the organization. The rules established by the parent site are inherited by each organization by default, but they can be overridden by the organization. The rules are defined in XML format; you can view and update them.

For more information, see [Container Object Initialization Rules](#).

## Auditing Activities Within the Organization

You may need to examine the audit logs for specific users or determine who has accessed or modified a particular document, part, project, or product. If there is concern that a user obtained access to projects or products to which that user should not have been invited, you can review the audit history record for the user to determine which projects, products, and documents the user visited and viewed or updated. In order to create the audit record, the auditing capability must be enabled at the site level.

For more information, see [Administering Audit Reports](#).

## Creating and Managing Access Control Policies

You can define policies that control the level of access to information by its members. For example, you may want to provide read access to all documents of type Engineering Specification to an engineering group in your organization. In this case, you need to first define an Engineering group and populate it with the appropriate members, then define a document type of Engineering Specification in your organization, and then use the Policy Administrator to define the access policy based on the document type, the group or groups provided access, and the access level.

You can create organization-level policies that apply to the entire organization, from the **Utilities** page of the **Organization** tab. To establish site-level policies, create those policies that apply to all organizations in the system from the **Utilities** page of the **Site** tab.

For more information about access control, see [Administering Access Control](#).

## Configuring Numbering and Versioning Schemes

You can configure the number and versioning schemes used to uniquely identify parts, documents, and other objects in the organization. The organization inherits site-level schemes, but can optionally define its own schemes. In general, when a company is hosting Windchill for its internal users, the numbering and versioning schemes should be defined at the site level; you should not override the site-level schemes by defining schemes at the organization level.

For more information about numbering and versioning schemes, see [Administering Containers](#).

## Monitoring and Managing Viewable Publishing

You can monitor and manage the publishing of viewable files that are optionally generated when CAD models are checked into products, libraries, and projects. You can also configure and update the watermarks used by the Windchill ProductView visualization tool when viewing document and part content.

For more information, see [Administering Visualization Services](#).

## Viewing Project Reports

You can access several predefined reports for project information. One of these reports lists projects deleted by members in the organization. You can restore projects that were inadvertently deleted by a user or empty projects to free up database space.

## Importing and Exporting Information

You can exchange information with another server, or between servers, or between a server and a file system using the Windchill import/export utilities. These utilities read and write system information in an XML format. The workflow and life cycle administration utilities accessible to you integrate the import/export functions. See [Import and Export](#) in the Administering Life Cycles chapter and [Import and Export](#) in the Administering Workflow Processes chapter.



## Purging, Archiving, and Restoring Jobs

You can create and manage purge jobs with the Purge Manager. If you have the applicable functionality installed, you can create purge jobs to permanently remove data from the Windchill system or create an archive of the data to restore at a later time. This would apply to viewing, restoring and deleting the contents of projects.

## Out-of-the-Box Organization Templates

At installation, the following organization templates are loaded:

Template Name	Description
General (PDMLink)	A sample template that can be used to create an organization for Windchill PDMLink.
General	A sample template that can be used to create a general organization for Windchill ProjectLink.
Enterprise	A sample template that can be used to create an enterprise organization.
Supplier	A sample template that can be used to create a supplier organization.
Customer	A sample template that can be used to create a customer organization.

The organization templates can define the same basic information that is discussed in the [Container Administrative Items](#) section of the Administering Containers chapter. The out-of-the-box organization templates define the following:

- Container structure
- Container participation
- Container access control policies
- Container data

The following sections describe the items that are defined in the templates.

### Container Structure

The organization templates define the following folder structure: Change Log, General, and Policies.

Some organization templates define groups that are automatically included in the organization. You can add users to the groups.

## Container Participation

The following groups are automatically created when an organization is created:

- Organization administrators
- Product creators (Windchill PDMLink)
- Library creators (Windchill PDMLink)
- Project creators (Windchill ProjectLink)

## Container Access Control Policies

Users who are members of the site Administrators group are granted Full Control to all object types at the site root domain. Users who are members of the organization Administrators group are granted Full Control to all object types at the root domains of the Organization (Default, System, Private, and User with the same name as the organization).

During the creation of an Organization container, additional domain-based access control rules are automatically created as follows:

- In the organization's System domain, the organization's All Participating Members group is granted read access to templates (such as document templates, life cycle templates, and workflow templates), objects, and initialization rules. For the complete list, see [System Domain Rules](#).
- In the organization's User domain, the organization's All Participating Members group is granted read access to the organization container and organization.

Additional domain-based access control rules can be defined within an organization template. The following is a list of the access control rules defined in the out-of-the-box organization templates:

- In the General (PDMLink) organization template, organization members (all users) are granted read access to all Released objects in the organization's /Default/PDM domain. Only product and library containers are affected by this rule.
- In the Enterprise organization template, project type groups are granted read access to projects contained in the project type domains. For example, a project of type Engineering has a corresponding group and organization domain. An access control rule is defined granting read access to the Engineering group in the Engineering domain. When a project of type Engineering is created, the project is put in the Engineering domain. A user who is added to the Engineering group is able to see all projects of type Engineering.

PTC recommends that you do not modify or delete the default set of access control rules automatically created during the creation of an organization, product, library, or project containers. It is acceptable to modify access control rules created from a template.

To adjust access control rules, use the Policy Administrator. To launch the Policy Administrator in the context of the organization, navigate to the **Utilities** page under the **Organization** tab, and click **Policy Administrator**. By launching the Policy Administrator from the **Organization** tab, the context is set to the organization context. In this context, only the domains and subdomains of the organization plus any ancestor domains from the site are visible. Members of the organization's Administrators group can create and modify rules within the organization's domains. Below is a list of some of the automatically created organization domains with some basic rules:

- /Default – Rules created at this level are inherited by the default domains of all public products, libraries, and projects contained within the organization. Typically, only business objects belong to this domain.
- /Default/PDM – Rules created at this level are inherited by the default domains of all public products and libraries contained within the organization. Typically, only business objects belong to this domain.
- /Default/Project – Rules created at this level are inherited by the default domains of all public projects contained within the organization. Typically, only business objects belong to this domain.
- /Private – Rules created at this level are inherited by the system domains of products, libraries, and projects contained within the organization. The default domain of private products, libraries, and projects also inherit these rules. PTC recommends that no additional access control rules be created within this domain.
- /System – Typically, only administrative objects (such as document templates, team templates, and life cycle templates) are in this domain; define policies for those types.

To update the access control rules for an organization domain within the Policy Administrator, select a domain and click **Update**. From the **Administrator Domain** window, click the **Access Control** tab. This provides the list of existing access control rules for this domain. From this tab, you can modify or delete existing rules and create new rules. When creating or updating rules, the list of groups available from the **Groups** tab include groups defined at the site and organization levels. For more information, see [Searching for Principals](#) in the Administering Containers chapter.

## Access Control Rules

Out-of-the-box access control rules are described in the following sections.

### Default Domain Rules

No permissions are created by default in this domain.

### System Domain Rules

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the organization.

Object Type	State	Permissions	Principal
WTObject	All	Full Control (All)	ORG ADMIN
DefaultWTContainerTemplate	All	Read	LIBRARY CREATOR (for Windchill PDMLink only)
DefaultWTContainerTemplate	All	Read	PRODUCT CREATOR (for Windchill PDMLink only)
DefaultWTContainerTemplate	All	Read	PROJECT CREATOR (for Windchill ProjectLink only)
WTDocument	All	Read	All Participating Members
WTPart	All	Read	All Participating Members
Rule	All	Read	All Participating Members
FilteredDynamicEnumSet	All	Read	All Participating Members
Notebook Template	All	Read	All Participating Members
DefaultWTContainerTemplate	All	Read	All Participating Members
Cabinet	All	Read	All Participating Members
SubFolder	All	Read	All Participating Members
TeamTemplate	All	Read	All Participating Members
WfTemplateObject	All	Read	All Participating Members
LifeCycleTemplate	All	Read	All Participating Members
DefaultWTContainerTemplate	All	Read	<i>organization principal</i> (for Windchill PDMLink and Windchill ProjectLink only)

Object Type	State	Permissions	Principal
Notebook Template	All	Read	<i>organization principal</i>

**Note:** The rules for all object types are defined programmatically when an organization container is created; they are not defined through the template that is used.

#### Organization User Domain (same name as the organization) Rules

The following table lists the combination of object type, life cycle state, and granted permissions that the organization user domain out-of-the-box rules define for the organization.

Object Type	State	Permissions	Principal
WTOBJECT	All	Full Control (All)	All (for Windchill Foundation & PDM)
OrgContainer	All	Read	All (for Windchill Foundation & PDM)
WTOBJECT	All	Full Control (All)	ORG ADMIN
OrgContainer	All	Read	All Participating Members
WTOrganization	All	Read	All Participating Members
OrgContainer	All	Read	<i>organization principal</i>
WTGroup	All	Read	<i>organization principal</i>
WTUser	All	Read	<i>organization principal</i>

**Note:** The rules for all object types are defined programmatically when an organization container is created; they are not defined through the template that is used.

### /Default/PDM Domain Rules for General (PDMLink) Template

The following table lists the combination of object type, life cycle state, and granted permissions that the out-of-the-box rules define for the General (PDMLink) organization template:.

Object Type	State	Permissions	Principal
WTOBJECT	Released	Read	<i>organization principal</i>

**Note:** The rules for the WTOBJECT object type is defined through the template.

### Default/PDM Domain Rules

The following table lists the combination of object type, life cycle state, and granted permissions that the Default/PDM domain out-of-the-box rules define for the organization:.

Object Type	State	Permissions	Principal
WTLibrary	All	Create	LIBRARY CREATOR
PDMLinkProduct	All	Create	PRODUCT CREATOR

**Note:** The rules are set programmatically when an organization container is created.

### Default/Project Domain Rules

The following table lists the combination of object type, life cycle state, and granted permissions that the Default/Project domain out-of-the-box rules define for the organization:

Object Type	State	Permissions	Principal
Project2	All	Create	PROJECT CREATOR

**Note:** The rules are set programmatically when an organization container is created.

## Container Data

For organizations created using the General or Enterprise organization templates, document types are inherited from the site.

For organizations created using the Customer or Supplier organization templates, document types include those defined in the template as well as the ones inherited from the site. The Customer and Supplier organization templates define the following document types: Analysis, Contract, Design, Drawing, Issues, Memo, Proposal, Report, Requirements, Schedule, Specification, and Statement of Work.

For all organizations, document attributes, part attributes, and project types are inherited from the site.

## Creating an Organization

**Note:** This section is for Windchill PDMLink and Windchill ProjectLink.


There are two types of organizations, an organization principal (WTOrganization type) and an organization container (also known as context). The organization principal represents a group of users. Each organization principal can be associated with and manage an organization container that allows creation of products, libraries, and projects within that organization.

For Windchill PDMLink, members of the organization principal that is associated with the organization container, and are product or library creators, can create products or libraries within the organization. For Windchill ProjectLink, all members of the organization, by default, can create projects if the organization is a subscriber. Members of other organization principals can participate as team members in these products, libraries, and projects.

Not every organization principal should have a corresponding organization container. The overhead of managing multiple organization containers for a single company can be difficult and is discouraged. Only create an organization container if the organization principal has a need to manage its own products, libraries, and projects.

**Note:** Only site administrators have the permissions to create organization containers.

Site administrators can create organization containers (contexts) from two locations:

- From the **Site** tab, on the **Organizations** page, click **Create Organization**  at the top of the **Organizations** table.

- From the **Organization** tab, the icon is located in the white bar below the tabs. The **Organization** tab is available only if an organization has been selected from the **Organization** table under the **Site** tab.

To associate an existing organization principal to the organization container, click **Search**, next to the **Organization Name** field, to search for existing organization principals. You can also type in a new organization name (one that does not match an existing organization principal); however, a new organization principal is created along with the organization container.

**Note:** The internet domain defined on the organization principal is important when creating new soft types. The internet domain is used to distinguish which organization owns the type. During the installation process, a default organization principal is created that contains an internet domain. The default organization principal (site administrator) is associated with the site container; any types defined at the site context are available to all organizations. In a non-exchange environment, create the first organization container and associate it with the default organization principal. In a multiorganization environment, PTC strongly



recommends that the internet domains of organizations be distinct from the internet domain for the site.

By default, the **Subscriber** check box is selected. This means the organization can host products and libraries (for Windchill PDMLink) and projects (for Windchill ProjectLink).

For Windchill ProjectLink, the **Automatically add new members to project creators group** check box shows on the window (not shown in this figure) and is selected by default.

By default, you are restricted to seeing only users that belong to the organization. Select the **Allow entire user and group directory selection** check box to provide the ability to search for all users. For groups, this setting allows you to see group entries contained in the current LDAP directory hierarchy level of the bundled directory (which shows the groups defined in the current organization context) and each enterprise directory setup through a JNDI adapter (search scope set to ONELEVEL). If you have created groups in a different organization context, you cannot see those groups in this organization context. Windchill PDMLink recommends you select this check box.

For additional information, see the help available from the **Create Organization** window.

For a description of the contents of the templates, see [Out-of-the-Box Organization Templates](#).

## Using the Organization Utilities Page

**Note:** This section is for Windchill PDMLink and Windchill ProjectLink.

The utilities listed on the **Utilities** page, which is accessible by clicking the **Utilities** link from the **Organization** tab, allow you to perform administrative actions at an organization level. Some of these utilities appear on other tabs. The difference is the context from which the utility is launched.

The utilities are grouped according to whether they are system administration utilities or business administration utilities. Many of links provided on the page give you access to the utilities that you need to use to perform the duties described in [Typical Duties of Organization Administrators](#).

To explore the use of each utility, click the corresponding link on the page and then click the help icon in the window that opens.

## Changing an Established Internet Domain

If the internet domain associated with an organization container is changed by modifying the organization principal using the Principal Administrator, the previous internet domain is no longer recognized as being associated with the organization container. Any types associated with the organization container's previous internet domain are no longer recognized as mapping to the organization. This can be corrected by adding the previous internet domain mapping to the internetDomain.properties file in the following format:

```
<internet_domain>=/wt.inf.container.OrgContainer=<org_name>
```

where *<internet\_domain>* is the previously defined internet domain for the organization principal, and *<org\_name>* is the name of the organization principal.

For example, the World Wide Tractors organization principal has an internet domain of worldwidetractors.com. A soft type of Engine created in that organization has the fully qualified name of com.worldwidetractors.Engine.

If the internet domain is later changed using the Principal Administrator to wwtractor.com, then the com.WorldWideTractors.Engine soft type is no longer associated with the organization's internet domain. By adding the following mapping to the internetDomain.properties file:

```
worldwidetractors.com=/wt.inf.container.OrgContainer=World Wide Tractors
```

any soft types associated with the old internet domain now map to the updated organization.

The internetDomain.properties file is also useful for historical tracking of previous internet domains associated with an organization principal.

**Note:** Updating the internetDomain.properties file constitutes a customization, and you must save a copy of the updated internetDomain.properties file using your site defined practices for customization. For more information, see the *Windchill Customizer's Guide*.

## Using the OrganizationSync Utility for User Organization Changes

When a user is moved from one organization to another by using a third party tool that allows your site to modify user entries in your directory service, the domains of the user and of the user's personal cabinet are not automatically changed. As the site administrator, you can use the OrganizationSync command-line utility to update organization-related information for users who have changed organizations. After changing the user's organization through a third party tool, run the OrganizationSync utility to change the other organization-related information for the user. Using the utility ensures that properties of a user or set of users are in sync with their current organization.

**Note:** If you use the Principal Administrator (and not a third party tool) to update user information in your directory service, then the OrganizationSync utility is automatically run as a result of changing a user's organization from the **Update User** page.

The OrganizationSync utility does the following:

- Assigns a set of user-related business objects to the administrative domain of the user's current organization. If a user does not have an organization, these objects are assigned to the Unaffiliated domain. The objects reassigned include the WTUser object, the user's personal cabinet and content of that personal cabinet, and the user's notebook.
- Removes the user from the administrators and creators groups of their previous organization container, if necessary. The program determines the previous container by extracting the container of the user's previous administrative domain.
- Updates the organization groups of any teams, in which the user was a member, to reflect the user's current organizational membership.

**Note:** It is safe to run the OrganizationSync utility on a user that was not actually updated.

The syntax of the OrganizationSync command is as follows:

```
java wt.inf.container.OrganizationSync [--users <user_list>] [--domains  
<domain_list>] [--domain-hierarchies <domain_list>] [--organizations  
<organization_list>]
```

The brackets ([]) in the syntax indicate optional parameters and indicate parameters that you specify together. Parameter names are case-sensitive; enter the names using the case shown in the syntax and the following table.

Parameter Value	Description
<user_list>	A space-separated list of user names.
<domain_list>	A space-separated list of domain paths.
<organization_list>	A space-separated list of organization names.

When <domain\_list> is used with the --domains parameter, all users in the domain are synchronized. When <domain\_list> is used with the --domain-hierarchies parameter, all the users in each domain and in all descendent domains are synchronized.

**Note:** Always run the OrganizationSync utility from a windchill shell. For details on using the windchill shell, see [About the windchill shell](#).

### Examples:

```
//Update the foo and bar users
java wt.inf.container.OrganizationSync --users foo bar

// Update all users in the site-level Unaffiliated domain
// and the "PTC Inc" domain in the "PTC Inc" org container
java wt.inf.container.OrganizationSync --domains
[/]/User/Unaffiliated "[/wt.inf.container.OrgContainer=PTC
Inc]/PTC Inc"

// Update all users in the site-level User domain, and in
// all of its child domains
java wt.inf.container.OrganizationSync --domain-hierarchies
[/]/User

// Update all members of the "Acme Inc." and
// "Widget Intl." organizations
java wt.inf.container.OrganizationSync --organizations "Acme
Inc." "Widget Intl."
```

In the above examples, the container path is enclosed in brackets to separate it from the domain path. [/] is the Site (root) container.

## Best Practices

### For Both Windchill PDMLink and Windchill ProjectLink

#### E-mail Addresses

Ensure that users have an e-mail address; many features in Windchill PDMLink and Windchill ProjectLink require that users have an e-mail address. If users do not have the e-mail attribute set in their user directory service entry, they cannot participate in the features that require an e-mail address.

#### Refreshing Teams

The recommended practice for defining groups is to define them at the organization level whenever possible and then use the groups in product, library, and project teams. The **Groups** page on the **Organization** tab can be used for this purpose.

Over the life a product, library, or project, the members of groups referenced in teams may change and you need to put together a plan on how to update the teams on a regular basis. You can use the **Refresh Teams** button on the **Organization** tab to immediately update the teams affected by a group membership change. However, doing this uses a lot of system resources that can slow down system performance during peak hours. Instead, PTC recommends that you schedule daily and weekly team updates using tasks provided by PTC. The tasks can be run in off-peak hours by using a queue. Automating the update of teams is turned off

by default. For details on turning on this feature, see the [Refreshing Groups](#) section of the Administering Teams and Roles chapter.

## For Windchill PDMLink

### Creating Only One Organization Container for Windchill PDMLink

For Windchill PDMLink, PTC recommends creating only one organization container (unless your site was configured as a unique namespace multiorganization environment when it was installed; for more information see [Business Object Uniqueness Considerations](#)). The default organization principal created during installation should be associated with the organization container. When creating the container, use **Search** to locate the installed organization principal.

To provide the ability to search for all members in the system, select the **Allow entire user and group directory selection** check box.

### Setting Object Initialization Rules

The object initialization rules include the ability to set the following for each object type:

- Folder paths
- Default life cycle template
- Default team template
- Default numbering scheme
- Default version scheme

Although these rules can be set for individual product and library containers, PTC recommends that you set them at either the organization or site level so that an entire organization or site is using the same set of rules.

### Setting Up Enhanced Life Cycle Templates

The out-of-the-box Product Design template includes object initialization rules that set different default life cycle templates and versioning schemes for parts, documents, and CAD documents than is set out-of-the-box at the site level. This template demonstrates the use of simplified life cycles and state-based versioning. Unless you have overriding business practices that require unique rules, PTC recommends that if you want to use these features, you modify the site object initialization rules and remove them from the Product Design template.

## For Windchill ProjectLink

### Setting Default Preferences in Windchill ProjectLink to Collapse Tables

In order to cut down the time it takes to load tables, you can create a preference to collapse tables for document and part iteration history pages.

The following can be set, using the Preference Manager on the **Utilities** page of the **Organization** tab, to collapse tables by default.

For a document iteration history table:

```
name: "history$doc_list$::ccomp" and value: "1"
```

For a part iteration history table:

```
name: "history$part_list$::ccomp" and value: "1"
```

You can use the xconfmanager utility to add the following wt.property to make all components collapsed:

```
com.ptc.netmarkets.defaultTablesCollapsed=true
```

For more information about using the xconfmanager, see [Administration Overview](#).

# 6

## Administering Products and Libraries

This chapter provides an overview for administering product and libraries in Windchill PDMLink and describes the typical duties that an administrator does. It also provides additional information about some of the main administrative tasks for products and libraries.

Topic	Page
Overview .....	6-2
Typical Duties of Product and Library Administrators.....	6-2
Out-of-the-box Product and Library Context Templates .....	6-8
Creating a Product .....	6-40
Creating a Library .....	6-41
Administering Teams .....	6-42
Creating Custom Change Monitor Reports .....	6-44
Using the Product and Library Utilities Page.....	6-45
Configuring Product or Library Containers for Dynamic Documents.....	6-45
Configuring Part and Document Relationships.....	6-47
Allowing URLs as Content for Documents .....	6-52
Best Practices For Object Initialization Rules.....	6-52

## Overview

Product and library administrators (also known as product and library managers) are responsible for managing product and library containers in Windchill PDMLink. The capabilities of product and library administrators are nearly identical.

Product and library administrators control the product and library configuration, and the membership in their product and library teams within the confines of a specific product or library application container. They control access to product and library information, and they define the specific life cycles, templates and processes, and monitor and manage the product and library activities.

Product application containers are used to define new product models or instances and collect all the information associated with the product. Product containers are defined by product creators who are authorized by the parent organization under which a product is created. Products inherit templates, roles, groups, and policies from their parent organization container. In addition, the administrator can define product-specific templates, roles, and policies.

Library application containers are used to manage standard parts and documents that are used across products and projects in an organization. Library containers are defined and managed by authorized library creators in the parent organization under which a library is created. Libraries inherit templates, roles, groups, and policies from their parent organization container. In addition, the administrator can define library-specific templates, groups, roles, and policies.

If Arbortext products are installed in your environment, product and library containers can be used to manage dynamic documents and the published documents created from dynamic documents.

For general information about container contents and how to create containers, see [Administering Containers](#).

## Typical Duties of Product and Library Administrators

Product and library administrators are responsible for managing the content of products and libraries. The capabilities of product and library administrators are nearly identical.

Product and library administrators are responsible for administering the system and business aspects of products and libraries.

The system administration aspects include the following:

- Importing and exporting data
- Managing preferences
- Managing replication and library rules



The business administration aspects include managing the following:

- Document and CAD document templates
- Life cycle templates
- Workflow templates
- Team templates
- Report templates
- Object initialization rules (including numbering and versioning)
- Policy access rules
- Custom reports
- Related reports
- ProductView and visualization

The following sections describe some of the duties in more detail.

## **Managing Team Members and Roles**

You define the team members and roles for the products and libraries that you manage. Each product team has a product manager group and each library team has a library manager group. Any individual that is a member of one of these managers groups has the rights to administer the product or library after it has been created. The creator of a library or product is automatically defined as a member of the product or library managers group, and is identified as the product or library owner, by default.

A product or library inherits the roles defined by its parent organization and the site. Additionally, roles can be defined in the context template used to create the product or library. You can then use these roles in the product or library, or you can add product or library-specific roles to product and library teams.

As time goes on, team membership in a product or library may change or the current members can take on different responsibilities. To accommodate these types of changes, product and library managers have the ability to reassign tasks from one member to another.

For additional information, see [Administering Teams](#).

## Managing Folders

You can define folders and links within products and libraries.

By default, only product or library managers can define folders and subfolders in a product or library. This is typically a good policy because it prevents members from adopting a multitude of folder organization models, thereby creating folder chaos.

By default, a folder inherits the domain of the parent folder. You can change this behavior as described in the [Assigning Domains to Folders in Windchill PDMLink or Windchill Foundation & PDM](#) section of the [Administering Containers](#) chapter.

## Managing Templates

You can define the document, CAD document, life cycle, team, report, and workflow templates that you want used in the context of a product or library. Each product or library inherits the templates defined by its parent organization and the site. Additionally, you can create new product or library-specific templates. If the name you specify is the name of an inherited template, then the new template overrides the inherited template that has the same name.

For additional information about templates, see [Container Templates](#).

## Managing Object Initialization Rules

Object initialization rules establish specific default attribute values that are used when instances of objects of a specific type are created.

In Windchill PDMLink, you can set default values for the following out-of-the-box object attributes:

- Folder paths
- Life cycle attributes
- Team template attributes
- Default numbering scheme
- Default versioning scheme

By default, the object initialization rules established by the site are inherited by each organization and then inherited by the product or library. However, they can be overridden by an organization or overridden in a product or library. The rules are defined in an XML format and can be viewed and edited by a product or library manager. For PTC recommendations in this area, see [Best Practices For Object Initialization Rules](#), later in this chapter.

For additional information about object initialization rules, see [Administering Object Initialization Rules](#).

## Managing Access Policies

You can define policies that control the level of access to information in a product or library. When defining a policy, the object types and groups defined in the parent organization may be used as well as the groups representing the team roles in a product or library. For example, you could create a policy that provides read access to all documents of type “Quality Assessment” to the product team role/group called “Testers”. You can also choose to extend read access to this document type to an organizational group with the name “Quality Assurance” (if such a policy is not already granted at the parent organization level).

For general information about domains and policies, see [Administering Domains and Policies](#).

For additional information about creating or updating access control rule policies, see [Administering Access Control](#).

## Configuring Numbering and Versioning Schemes

You can configure the number and versioning scheme used to uniquely identify parts, documents and other objects in the product or library. The numbering and versioning schemes defined at the site and organization level are inherited by products and libraries by default, but each product or library can optionally define its own schemes. Use the Object Initialization Rules Administrator to configure the number and versioning schemes.

In general, when a company is hosting Windchill for its internal use, the numbering and versioning schemes should all be defined in the Site container and should not be overridden by an organization or in a product or library. The Product Design template demonstrates how to set up versioning (and other required functionality) for use in a basic life cycle management environment, where multiple life cycle templates are in use and state-based versioning is set up. If you choose to use this type of environment, consider defining the versioning (and other initialization rules) in the Site container rather than in individual product containers so that all of your products and libraries use the same set of rules.

For additional information, see [Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects](#) and [Administering the Versioning of Parts, Documents, and CAD Documents](#).

## Managing the Life Cycle of Parts, Documents, CAD Documents, and Dynamic Documents

Within your Windchill PDMLink solution, you have the choice of using a variety of out-of-the-box life cycle templates to help manage the life cycle of parts, documents, CAD documents, and dynamic documents. You can also create your own life cycle template or modify one of the existing templates. The options available to you range from ability to fully automate the process of promoting a business object through maturity levels with the use of workflow processes to

providing manual transition process where users in specified roles can promote objects to new states explicitly through the Promotion action provided in Windchill PDMLink. Default life cycle templates are established for object types through the use of object initialization rules and a default set of Site rules is loaded during the installation process.

For additional information, see [Administering Object Initialization Rules](#) and [Administering Life Cycles](#).

## Managing Viewable Publishing

You can monitor and manage the publishing of viewable files that are optionally generated when CAD models are checked into products and libraries. You can also configure and update the watermarks used by the ProductView visualization tool when viewing document and part content from the product or library.

For additional information, see [Administering Visualization Services](#).

## Managing Custom Reports

Windchill PDMLink provides the ability to create custom reports for:

- Auditing your system
- Monitoring the change process

In both of these cases, you create the reports using the report generation utility, Query Builder. You access this utility through the **Report Manager** link on the **Utilities** page of product, library, organization, and Site containers. The reporting utility is designed to be used by those who have a working knowledge of the Windchill data object model.

For Auditing reports, you can create and update custom reports against the objects and attributes defined within the product or library. For additional audit report information, see [Administering Audit Reports](#) in this chapter.

For Change Monitor reports, you can create custom reports that then can be accessed from a product or library **Change Monitor** link. Windchill PDMLink provides several out-of-the-box reports which are listed under **All Special Reports** on the Change Monitor page. The custom reports you create will also appear in this list.

For details on creating Change Monitor reports, see [Creating Custom Change Monitor Reports](#) in this chapter.

## Managing Related Reports

Windchill PDMLink provides a number of related reports that you can use when viewing a product structure. These reports include:

- Single Level BOM
- Indented BOM
- Multilevel Where Used
- Multilevel BOM Compare

If you want part masters in the same level with the same quantity to be considered identical in the Multilevel BOM Compare report, you need to set the following property to true:

```
wt.part.identicalProductMastersAreSame
```

For more information on the related reports, see the *Windchill PDMLink User's Guide*.

## Importing and Exporting Information

You can import information into a product or library, and export information to a local file system. The import/export facilities support information exchange with another Windchill server or non-Windchill system. These utilities read and write system information in an XML format. For general import and export information, see the Windchill Import and Export chapter in the *Windchill System Administrator's Guide*.

The workflow and life cycle administration utilities accessible to the site and organization administrator integrate the import and export functions with administering workflows and life cycles.

For life cycle import and export information, see [Import and Export](#) in the Administering Life Cycles chapter.

For workflow import and export information, see [Import and Export](#) in the Administering Workflows chapter.

## Configuring External Vaults or Replication Sites to Optimize Performance

You can configure the vaulting and replication rules for the external file and replica sites established by the site administrator.

When external vaults are configured, document and part content is stored on a file system rather than in the database. This configuration can provide significant upload performance improvements and is appropriate when the site is frequently used to exchange large files (such as CAD model files).

By default, external storage rules are based on individual domains in each container. You can launch the External Storage Administrator client from the

**Utilities** page in the context of a product, a library, an organization, or the site. The client allows you to create vaulting rules for the domains - /System and /Default - associated with the container in which the client is launched.

**Note:** The rules needed for setting up an external file vault for a product or library are not inherited from the organization or site.

If the increased number of vaults and file vault rules becomes unmanageable, the site administrator can force vaulting to be accomplished through a single vault by setting the `wt.fv.forceContentToVault` property to true. For how to set external file vault rules or set up a single vault, see the Administering External File Vaults chapter in the *Windchill System Administrator's Guide*.

Site administrators typically define the replica site and the replication schedule, and the product or library managers configure the replication rules for a particular product or library. You can access the administrator utilities, but can only work on setting the replication and file vaulting rules. For how to set replication rules, see the Administering Content Replication chapter in the *Windchill System Administrator's Guide*.

## Out-of-the-box Product and Library Context Templates

When Windchill PDMLink is installed, the following product and library templates are loaded:

- **Product Design** -- This template provides a demonstration of how to setup a product container for use with the One Phase Development and Two Phase Development life cycle templates and state-based versioning. As described in detail later in this section, the default set of roles defined are Members, Promotion Reviewers, Promotion Approvers, Marketing, Procurement Engineer, Designer, Quality Engineer, Manufacturing Engineer, Design Team Leader, Change Review Board, Change Admin I, Change Admin II, and Change Admin III. Also, the Guests and Product Manager roles are created automatically. Some basic information about these roles is as follows:
  - The Members role is used as a basic role to grant container team membership. A set of access rules are defined for confirmed members of the container team.
  - The Designer and Design Team Leader roles are unique to this template and identify the people who are responsible for creating and maintaining the business objects used to design a product.
  - Any user added to a role on the team, except the Guests role, will be added to the CONFIRMED group. The access control rules for this group limits the group to a subset of the capabilities available to the Designer and Design Team Leader.
  - Additionally, there are marketing and engineering roles that are unique to this template that can be used to delineate those functions in the design process.

- Access policies defined for Change Objects have been setup to work in conjunction with the closed-loop change process.
- Members of the Guest role are granted Read access to all objects.
- General Product and General Library -- These templates provide examples of how to setup basic access control for a general product or library container. As described in detail later in this section, the default set of roles defined are Members, Promotion Reviewers, Promotion Approvers, Change Review Board, Change Admin I, Change Admin II, and Change Admin III. Also, the Guests and Product/Library Manager roles are created automatically. Some basic information about these roles is as follows:
  - The Members role is used as a basic role to grant container team membership. A set of access rules are defined for confirmed members of the container team.
  - Any user added to a role on the team, except the Guests role, will be added to the CONFIRMED group. The general theme for the CONFIRMED access rules in the General Product and General Library templates is to grant Create, Modify and Delete permission in the initial state (In Work) and to grant Read permission for all states (In Work, Released, and Canceled). These rules are generated for all business objects.
  - Access policies defined for Change Objects have been setup to work in conjunction with the closed-loop change process.
  - Members of the Guest role are granted Read access to all objects.
- Part Library -- This template provides an example of how you can define a parts library. The default set of roles defined are Members, Change Admin I, Change Admin II, and Change Admin III. The same theme for CONFIRMED access rules defined in the general templates are used here except rules are only defined for WTParts, CAD documents and change objects. This restricts the use of this library for the management of parts only. Members of the Guest role are granted Read access to all objects.
- Document Library -- This template provides an example of how you can define a document library. The default set of roles defined are Members, Change Admin I, Change Admin II, and Change Admin III. The same theme for CONFIRMED access rules defined in the general templates are used here except only rules are only defined for WTDocuments and change objects. This would restrict the use of this library for the management of documents only. Members of the Guest role are granted Read access to all objects.

The product and library templates can define the same basic information that is discussed in the [Container Administrative Items](#) section of the Administering Containers chapter.

When users are working primarily with dynamic documents, you can start with the context template that is closest to your business model and modify the product

or library that you create to meet your needs. For some general guidelines on administrator activities that you should consider doing, see [Configuring Product or Library Containers for Dynamic Documents](#).

## Product Design Template

The Product Design out-of-the-box template defines the following container administrative items:

- Promotion Requests subfolder in the Default domain
- Container participation
- Container access control policies
- Object initialization rules for the following objects:
  - wt.maturity.PromotionNotice
  - wt.part.WTPart
  - wt.epm.EPMDocument
  - wt.doc.WTDocument

Through the object initialization rules, the following life cycle templates are set as defaults in the Design Product template:

- Two Phase Development life cycle template - default template for parts and functional specifications. It has the following states:



- One Phase Development life cycle template - default template for documents and CAD documents. It has the following states:

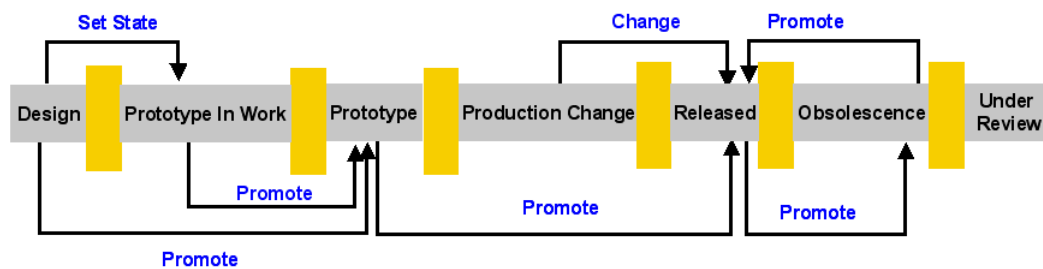
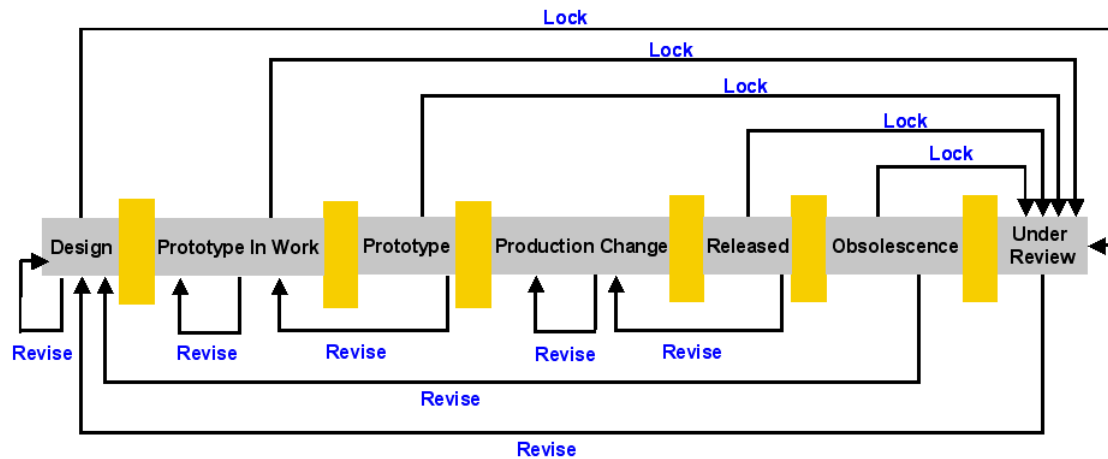


**Note:** Although you can use multiple life cycle templates within a product container as is done with the demonstration Product Design template, PTC recommends that you become familiar with the functionality available through these life cycle templates and select one of the out-of-the-box templates (or create your own template) to use for all life-cycle managed objects in a product container.

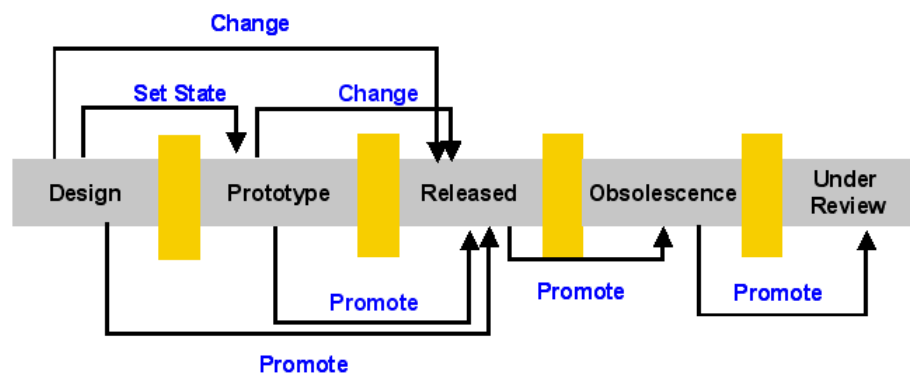
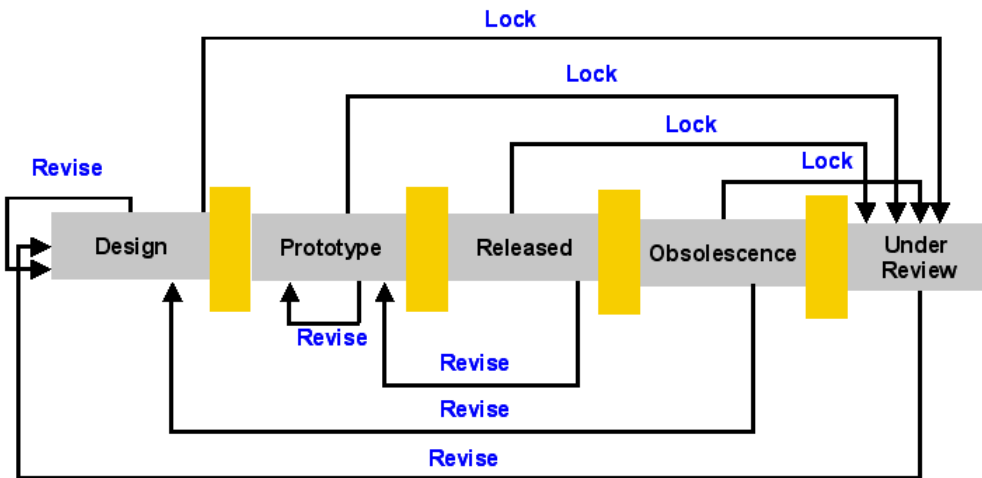


The Two Phase Development and One Phase Development life cycle templates have the following transitions defined: Lock, Revise, Promote, Change, and Set State. Each template establishes different transition rules out of the box.

The following diagrams show the out-of-the-box transition rules for the Two Phase Development life cycle:



The following diagrams show the out-of-the-box transition rules for the One Phase Development life cycle:



For additional information about modifying or creating life cycle templates, see the [Administering Life Cycles](#) chapter. The out-of-the-box workflows that are used with the default life cycle templates are:

- Promotion Request Approval Process
- Promotion Request Review Process

For additional information about these workflows, see [Promotion Process](#) in the Administering Life Cycles chapter.

The following sections describe the items that are defined in the Product Design template.

## Out-of-the-box Subfolder for wt.maturity.PromotionNotice Objects

In the Default cabinet and Default domain, the Product Design template creates the Promotion Requests subfolder. This folder is then defined as the default folder for the wt.maturity.PromotionNotice object type. Creating and using this folder is not required, but provides a useful folder for promotion requests.

## Out-of-the-box Container Participation

The Product Design template defines roles that are automatically included in product teams. These roles are the roles that automatically appear on the **Teams** table. Additionally, roles available through the organization context can be added to the team.

The following roles are automatically included in the team:

- Product Manager
- Guests, which is the role to use for users who you do not want as a member of the team, but you do want them to have limited access to the product. Users who are guests will not see the product on their Product List. They would need to search for the product to locate it. Out of the box, guests have only read access to product data.
- Guests, which is the role to use for users who you do not want as a member of the team, but you do want them to have limited access to the product or library. Users who are guests will not see the product or library on their Product List or Library List. They would need to search for the product or library to locate it. Out of the box, guests have only read access to product or library data.

You can allow guests to see products and libraries on their product and library lists by adding the following property and value to the wt.properties file:

```
wt.inf.team.displayContainerListForGuest=true
```

**Note:** Use the -s parameter on the xconfmanager command to add Windchill properties. See [About the xconfmanager Utility](#) for details. This property controls the guest visibility on project lists as well as product and library lists.

The out-of-the-box Product Design template adds following roles to the team:

- Members, which is a role that can be used for team members who do not belong in other roles on the team.
- Change Admin I, which is the change administrator I change management role
- Change Admin II, which is the change administrator II change management role
- Change Admin III, which is the change administrator III change management role

- Promotion Reviewers, which are the team members for reviewing Promotion Requests
- Promotion Approvers, which are the team members for approving Promotion Requests
- Change Review Board, which are the team members of the Change Review Board
- Design Team Leader, which is the Design Team Leader role. This role is unique to this template.
- Designer, which is the role used to create, edit and manage business objects within the system. This role is unique to this template.
- Manufacturing Engineer, which is the Workflow Manufacturing Engineer role. This role is unique to this template.
- Marketing, which is the Marketing role. This role is unique to this template.
- Procurement Engineer, which is the Procurement Engineer role. This role is unique to this template.
- Quality Engineer, which is the Workflow Quality Engineer role. This role is unique to this template.

Additionally, the following groups are created:

- The CONFIRMED group contains all users and groups who are added as a member of a role in the team except those added to the Guests role.
- The INVITED group is created, but not used for products.

## Out-of-the-box Container Access Control Policies

When you create a product container using the out-of-the-box Product Design template, domain-based access control rules are defined in the System and Default domains of a product container. The rules that are created reference the groups that correspond to the out-of-the-box roles established for the team.

The following section lists the roles and their corresponding groups. Additional sections provide the out-of-the-box rules generated that reference the groups listed in the next section. Some rules are generated as a result of program code that is executed when the container is created. While other rules are generated as a result of administrative policy items that are in the out-of-the-box template. Each of the tables in the following sections includes the Generation Method column that indicates how the rule is generated. When the method of generation is a result of executing program code, then the column contains the word "program". When the method of generation is a result of administrative policy items that are in the out-of-the-box template, then the column contains the word "template".

**Caution:** PTC recommends that you do not modify the access control rules that are programmatically set when containers are created. Doing so can cause problems with access to data and administrative functionality.

**Note:** There are no out-of-the-box rules set for the Members or Change Admin III roles.

## Team Roles and Groups

The following table lists the out-of-the-box Product Design template team roles and groups, and identifies which group that is associated with each role:

Team Role	Corresponding Group
Change Admin I	CHANGE ADMINISTRATOR I
Change Admin II	CHANGE ADMINISTRATOR II
Change Admin III	CHANGE ADMINISTRATOR III
Change Review Board	CHANGE REQUEST REVIEW BOARD
Promotion Approvers	PROMOTION APPROVERS
Promotion Reviewers	PROMOTION REVIEWERS
Design Team Leader	DESIGN TEAM LEADER
Designer	DESIGNER
Manufacturing Engineer	MANUFACTURING ENGINEER
Marketing	MARKETING
Procurement Engineer	PROCUREMENT ENGINEER
Quality Engineer	QUALITY ENGINEER
Guests	GUEST
Members	MEMBERS
Product Manager	PRODUCT MANAGER
	CONFIRMED
	INVITED

**Note:** The CONFIRMED group is not associated with a specific role. All users and groups added to any role, except the Guests role, are automatically added to the CONFIRMED group.

**Note:** Although the INVITED group is created for all application containers, this group is not used for products and libraries.

### Rules for the CONFIRMED Group

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the CONFIRMED group are described in the following sections.

#### Default Domain Rules for CONFIRMED Group

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the CONFIRMED group:

Object Type	State	Permissions	Generation Method
Cabinet	All	Read Modify	program
EPMDocument	Prototype	Read	template
EPMDocument	Released	Read	template
EPMDocument	Obsolescence	Read	template
EPMDocument	Under Review	Read	template
ManagedBaseline	All	Read	template
ManagedBaseline	In Work	Modify Create Delete	template
PromotionNotice	All	Read	template
PromotionNotice	Open	Read Modify Create	template
StructuredAnnotationSet	All	Read	template
StructuredAnnotationSet	In Work	Modify Create Delete	template
SubFolder	All	Read Modify	program
WTChangeActivity2	All	Read	template
WTChangeActivity2	Open	Modify	template
WTChangeActivity2	Implementation	Modify	template
WTChangeIssue	All	Read	template
WTChangeIssue	Open	Modify Create	template
WTChangeOrder2	All	Read	template
WTChangeOrder2	Implementation	Modify	template

Object Type	State	Permissions	Generation Method
WTChangeOrder2	Open	Modify	template
WTChangeProposal	Open	Modify Create	template
WTChangeRequest2	All	Read	template
WTChangeRequest2	Open	Modify Create	template
WTDocument	Prototype	Read	template
WTDocument	Released	Read	template
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template
WTPart	Prototype	Read	template
WTPart	Released	Read	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template
WTPartSubstituteLink	All	Read Create Modify Delete	template
WTPartAlternateLink	All	Read Create Modify Delete	template
WTProductConfiguration	All	Read	template
WTProductConfiguration	In Work	Modify Create Delete	template
WTProductInstance2	All	Read	template
WTProductInstance2	In Work	Modify Create Delete	template
WTProductInstance2	Released	Revise New View Version	template

**Note:** As shown in the Generation Method column, the rules for Cabinet and SubFolder object types are defined programmatically when a product container is created; they are not defined through the template that is used. All other rules are defined through the template.

### System Domain Rules for CONFIRMED Group

The following table lists the combination of object type, life cycle state, and granted permissions that the System domain out-of-the-box rules define for the CONFIRMED group:

Object Type	State	Permissions	Generation Method
WTExecutionObject	All	Modify Create Read	program
WTOBJECT	All	Read	program

**Note:** As shown in the Generation Method column, these rules are defined programmatically when a product container is created; they are not defined through the template that is used.

### Rules for the GUEST Group

Out-of-the-box Product Design access control rules for the GUEST group (Guests role) are described in the following sections.

#### Default Domain Rules for the GUEST Group

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for guests of a product:

Object Type	State	Permissions	Generation Method
Cabinet	All	Read	program
SubFolder	All	Read	program
WTOBJECT	All	Read	template

**Note:** As shown in the Generation Method column, the rules for Cabinet and SubFolder object types are set programmatically when a product container is created; they are not defined through the template that is used. The rule for the WTOBJECT object type is defined through the template.



### System Domain Rules for the GUEST Group

The following table lists the combination of object type, life cycle state, and granted permissions that the System domain out-of-the-box rules define for the guests of a product or library:

Object Type	State	Permissions	Generation Method
WTObject	All	Read	program

**Note:** As shown in the Generation Method column, this rule is defined programmatically when a product container is created; it is not defined through the template that is used.

### Rules in Default Domain for the MARKETING Group

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the MARKETING group are described in the following sections. The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the MARKETING group:

Object Type	State	Permissions	Generation Method
EPMDocument	Prototype In Work	Read	template
EPMDocument	Prototype	Read	template
EPMDocument	Production Change	Read	template
EPMDocument	Released	Read	template
EPMDocument	Obsolescence	Read	template
EPMDocument	Under Review	Read	template
WTDocument	Design	Read Modify Create Delete Revise	template
WTDocument	Prototype In Work	Read Modify Delete Revise	template
WTDocument	Prototype	Read Modify Delete Revise	template
WTDocument	Production Change	Read Modify Delete Revise	template
WTDocument	Released	Permission List Revise	template

Object Type	State	Permissions	Generation Method
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template
WTPart	Prototype In Work	Read	template
WTPart	Prototype	Read	template
WTPart	Production Change	Read	template
WTPart	Released	Read	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template

#### Rules in Default Domain for the PROCUREMENT ENGINEER Group

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the PROCUREMENT ENGINEER group are described in the following sections. The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the PROCUREMENT ENGINEER group:

Object Type	State	Permissions	Generation Method
EPMDocument	Prototype In Work	Read	template
EPMDocument	Prototype	Read	template
EPMDocument	Production Change	Read	template
EPMDocument	Released	Read	template
EPMDocument	Obsolescence	Read	template
EPMDocument	Under Review	Read	template
WTDocument	Design	Read Modify Create Delete Revise	template
WTDocument	Prototype In Work	Read Modify Delete Revise	template
WTDocument	Prototype	Read Modify Delete Revise	template

Object Type	State	Permissions	Generation Method
WTDocument	Production Change	Read Modify Delete Revise	template
WTDocument	Released	Read Revise	template
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template
WTPart	Prototype In Work	Read	template
WTPart	Prototype	Read	template
WTPart	Production Change	Read	template
WTPart	Released	Read	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template

#### Rules in Default Domain for the QUALITY ENGINEER Group

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the QUALITY ENGINEER group are described in the following sections. The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the QUALITY ENGINEER group:

Object Type	State	Permissions	Generation Method
EPMDocument	Prototype In Work	Read	template
EPMDocument	Prototype	Read	template
EPMDocument	Production Change	Read	template
EPMDocument	Released	Read	template
EPMDocument	Obsolescence	Read	template
EPMDocument	Under Review	Read	template
WTDocument	Design	Read Modify Create Delete Revise	template

Object Type	State	Permissions	Generation Method
WTDocument	Prototype In Work	Read Modify Delete Revise	template
WTDocument	Prototype	Read Modify Delete Revise	template
WTDocument	Production Change	Read Modify Delete Revise	template
WTDocument	Released	Read Revise	template
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template
WTPart	Prototype In Work	Read	template
WTPart	Prototype	Read	template
WTPart	Production Change	Read	template
WTPart	Released	Read	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template

#### Rules in Default Domain for the DESIGNER Group

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the DESIGNER group are described in the following sections. The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the DESIGNER group:

Object Type	State	Permissions	Generation Method
EPMDocument	Design	Read Modify Create Delete Revise	template
EPMDocument	Prototype In Work	Read Modify Create Delete Revise	template
EPMDocument	Prototype	Read Modify Create Delete Revise	template

<b>Object Type</b>	<b>State</b>	<b>Permissions</b>	<b>Generation Method</b>
EPMDocument	Production Change	Read Modify Create Delete Revise	template
EPMDocument	Released	Read Revise	template
EPMDocument	Obsolescence	Read	template
EPMDocument	Under Review	Read	template
WTDocument	Design	Read Modify Create Delete Revise	template
WTDocument	Prototype In Work	Read Modify Create Delete Revise	template
WTDocument	Prototype	Read Modify Create Delete Revise	template
WTDocument	Production Change	Read Modify Create Delete Revise	template
WTDocument	Released	Read Revise	template
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template
WTPart	Design	Read Modify Create Delete Revise New View Version	template
WTPart	Prototype In Work	Read Modify Create Delete Revise New View Version	template
WTPart	Prototype	Read Modify Create Delete Revise New View Version	template
WTPart	Production Change	Read Modify Create Delete Revise New View Version	template
WTPart	Released	Read Revise New View Version	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template

**Note:** As shown in the Generation Method column, the rules for Cabinet and SubFolder object types are defined programmatically when a product container is created; they are not defined through the template that is used. All other rules are defined through the template.

#### Rules in Default Domain for the MANUFACTURING ENGINEER Group

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the MANUFACTURING ENGINEER group are described in the following sections. The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the MANUFACTURING ENGINEER group:

Object Type	State	Permissions	Generation Method
EPMDocument	Design	Read Modify Create Delete Revise	template
EPMDocument	Prototype In Work	Read Modify Delete Revise	template
EPMDocument	Prototype	Read Modify Delete Revise	template
EPMDocument	Production Change	Read Modify Delete Revise	template
EPMDocument	Released	Read Revise	template
EPMDocument	Obsolescence	Read	template
EPMDocument	Under Review	Read	template
WTDocument	Design	Read Modify Create Delete Revise	template
WTDocument	Prototype In Work	Read Modify Delete Revise	template
WTDocument	Prototype	Read Modify Delete Revise	template
WTDocument	Production Change	Read Modify Delete Revise	template
WTDocument	Released	Read Revise	template
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template

<b>Object Type</b>	<b>State</b>	<b>Permissions</b>	<b>Generation Method</b>
WTPart	Design	Read Modify Create Delete Revise New View Version	template
WTPart	Prototype In Work	Read Modify Create Delete Revise New View Version	template
WTPart	Prototype	Read Modify Create Delete Revise New View Version	template
WTPart	Production Change	Read Modify Create Delete Revise New View Version	template
WTPart	Released	Read Revise New View Version	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template

#### **Rules in Default Domain for the DESIGN TEAM LEADER Group**

Out-of-the-box Product Design access control rules that are defined in the Default and System domains for the DESIGN TEAM LEADER group are described in the following sections. The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the DESIGN TEAM LEADER group:

<b>Object Type</b>	<b>State</b>	<b>Permissions</b>	<b>Generation Method</b>
EPMDocument	Design	Read Modify Create Delete Revise	template
EPMDocument	Prototype In Work	Read Modify Create Delete Revise	template
EPMDocument	Prototype	Read Modify Create Delete Revise	template
EPMDocument	Production Change	Read Modify Create Delete Revise	template
EPMDocument	Released	Read Revise	template
EPMDocument	Obsolescence	Read	template

<b>Object Type</b>	<b>State</b>	<b>Permissions</b>	<b>Generation Method</b>
EPMDocument	Under Review	Read	template
WTDocument	Design	Read Modify Create Delete Revise	template
WTDocument	Prototype In Work	Read Modify Create Delete Revise	template
WTDocument	Prototype	Read Modify Create Delete Revise	template
WTDocument	Production Change	Read Modify Create Delete Revise	template
WTDocument	Released	Read Revise	template
WTDocument	Obsolescence	Read	template
WTDocument	Under Review	Read	template
WTPart	Design	Read Modify Create Delete Revise New View Version	template
WTPart	Prototype In Work	Read Modify Create Delete Revise New View Version	template
WTPart	Prototype	Read Modify Create Delete Revise New View Version	template
WTPart	Production Change	Read Modify Create Delete Revise New View Version	template
WTPart	Released	Read Revise New View Version	template
WTPart	Obsolescence	Read	template
WTPart	Under Review	Read	template



### Rules in Default Domain for PROMOTION REVIEWERS Group

Out-of-the-box Product Design access control rule for the PROMOTION REVIEWERS group is defined through the template and is described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
PromotionNotice	Under Review	Read	template

### Rules in Default Domain for CHANGE REQUEST REVIEW BOARD Group

Out-of-the-box Product Design access control rule for the CHANGE REQUEST REVIEW BOARD group are defined through the template and are described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
WTChangeRequest2	Under Review	Read	template
WTChangeOrder2	Under Review	Read	template

**Note:** No rules are defined in the System domain for the CHANGE REQUEST REVIEW BOARD group.

### Rules in Default Domain for PROMOTION APPROVERS Group

Out-of-the-box Product Design access control rule for the PROMOTION APPROVERS group is defined through the template and is described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
PromotionNotice	Under Review	Read	template

**Note:** No rules are defined in the System domain for the PROMOTION APPROVERS group.

### Rules for PRODUCT MANAGER Group

Out-of-the-box Product Design access control rules for the PRODUCT MANAGER group (Product Manager role) are described in the following sections.

#### Default Domain Rule for PRODUCT MANAGER Group

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the Product Manager.:

Object Type	State	Permissions	Generation Method
WTOBJECT	All	Full control (All)	program

**Note:** As shown in the Generation Method column, this rule is defined programmatically when a product or library container is created; it is not defined through the template that is used.

#### System Domain Rule for PRODUCT MANAGER Group

The following table lists the combination of object type, life cycle state, and granted permissions that the System domain out-of-the-box rules define for the Product Manager:

Object Type	State	Permissions	Generation Method
WTOBJECT	All	Full control (All)	program

**Note:** As shown in the Generation Method column, this rule is defined programmatically when a product or library container is created; it is not defined through the template that is used.

#### Rules in Default Domain for CHANGE ADMINISTRATOR I Group

Out-of-the-box Product Design access control rules for the CHANGE ADMINISTRATOR I group (Change Admin I role) are defined through the template and are described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
WTChangeActivity2	All	Full Control (All)	template
WTChangeIssue	All	Full Control (All)	template
WTChangeOrder2	All	Full Control (All)	template
WTChangeProposal	All	Full Control (All)	template
WTChangeRequest2	All	Full Control (All)	template

**Note:** No rules are defined in the System domain for the CHANGE ADMINISTRATOR I group.

### Rules in Default Domain for CHANGE ADMINISTRATOR II Group

Out-of-the-box Product Design access control rules for the CHANGE ADMINISTRATOR II group (Change Admin II role) are defined through the template and are described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
WTChangeActivity2	All	Full Control (All)	template
WTChangeOrder2	All	Full Control (All)	template

**Note:** No rules are defined in the System domain for the CHANGE ADMINISTRATOR II group.

### Out-of-the-box Object Initialization Rules

The Product Design template sets object initialization rules for the following object types:

- wt.maturity.PromotionNotice
- wt.part.WTPart
- wt.epm.EPMDocument
- wt.doc.WTDocument

The following table lists the object types, the attributes, and the default values set by the object initialization rules that are in the template.:

Object Type	Attribute	Default Value
PromotionNotice	folder.id	/Default/Promotion Requests
WTPart	folder.id	/Default
WTPart	lifeCycle.id	Two Phase Development
WTPart	teamTemaplate.id	Default
WTPart	MBA versionInfo	StateBased
EPMDocument	folder.id	/Default
EPMDocument	lifeCycle.id	One Phase Development
EPMDocument	teamTemaplate.id	Default
EPMDocument	MBA versionInfo	StateBased
WTDocument	folder.id	/Default
WTDocument	lifeCycle.id	One Phase Development
WTDocument	teamTemaplate.id	Default
WTDocument	MBA versionInfo	StateBased

**Note:** Setting object initialization rules in a product template means that the rules are different than those set for the organization or site. This is the case with this template and the out-of-the-box object initialization rules that are set in the site container. PTC recommends that you only use this template as a demonstration template and consider setting those rules that you want to use for parts, documents, and CAD documents in either the organization or site container.

For additional information about object initialization rules, see [Administering Object Initialization Rules](#).

For additional information about state-based versioning, see [Administering the Versioning of Parts, Documents, and CAD Documents](#).

## General Product and General Library Templates

The General Product and General Library out-of-the-box templates define only the following container administrative items:

- Container participation
- Container access control policies

The following sections describe the items that are defined in these templates.

### Out-of-the-box Container Participation

The General Product and General Library templates define roles that are automatically included in product and library teams. These roles are the roles that automatically appear on the **Teams** table. Additionally, roles available through the organization context can be added to the team.

The following roles are automatically included in the team:

- Product Manager (for products) and Library Manager (for libraries)
- Guests, which is the role to use for users who you do not want as a member of the team, but you do want them to have limited access to the product or library. Users who are guests will not see the product or library on their Product List or Library List. They would need to search for the product or library to locate it. Out of the box, guests have only read access to product or library data.

The out-of-the-box General Product and General Library templates add the following roles to the team:

- Members, which is a role that can be used for team members who do not belong in other roles on the team.
- Change Admin I, which is the change administrator I change management role
- Change Admin II, which is the change administrator II change management role
- Change Admin III, which is the change administrator III change management role
- Promotion Reviewers, which are the team members for reviewing Promotion Requests
- Promotion Approvers, which are the team members for approving Promotion Requests
- Change Review Board, which are the team members of the Change Review Board

Additionally, the following groups are created:

- The CONFIRMED group contains all users and groups who are added as a member of a role in the team except those added to the Guests role.
- The INVITED group is created, but not used for products or libraries.

## Out-of-the-box Container Access Control Policies

When you create a product or library container using the out-of-the-box General Product and General Library templates, domain-based access control rules are defined in the System and Default domains of a product or library container. The rules that are created reference the groups that correspond to the out-of-the-box roles established for the team.

The following section lists the roles and their corresponding groups. Additional sections provide the out-of-the-box rules generated that reference the groups listed in the next section. Some rules are generated as a result of program code that is executed when the container is created. While other rules are generated as a result of administrative policy items that are in the out-of-the-box templates. Each of the tables in the following sections includes the Generation Method column that indicates how the rule is generated. When the method of generation is a result of executing program code, then the column contains the word "program". When the method of generation is a result of administrative policy items that are in the out-of-the-box templates, then the column contains the word "template".

**Caution:** PTC recommends that you do not modify the access control rules that are programmatically set when containers are created. Doing so can cause problems with access to data and administrative functionality.

**Note:** There are no out-of-the-box rules set for the Members or Change Admin III roles.

## Team Roles and Groups

The following table lists the out-of-the-box General Product and General Library team roles and groups, and identifies which group that is associated with each role:

Team Role	Corresponding Group
Change Admin I	CHANGE ADMINISTRATOR I
Change Admin II	CHANGE ADMINISTRATOR II
Change Admin III	CHANGE ADMINISTRATOR III
Change Review Board	CHANGE REQUEST REVIEW BOARD
Promotion Approvers	PROMOTION APPROVERS

Team Role	Corresponding Group
Promotion Reviewers	PROMOTION REVIEWERS
Guests	GUEST
Members	MEMBERS
Product Manager (products only)	PRODUCT MANAGER
Library Manager (libraries only)	LIBRARY MANAGER
	CONFIRMED
	INVITED

**Note:** The CONFIRMED group is not associated with a specific role. All users and groups added to any role, except the Guests role, are automatically added to the CONFIRMED group.

**Note:** Although the INVITED group is created for all application containers, this group is not used for products and libraries.

#### Rules for the CONFIRMED Group

Out-of-the-box General Product and General Library access control rules that are defined in the Default and System domains for the CONFIRMED group are described in the following sections.

#### Default Domain Rules for CONFIRMED Group

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the CONFIRMED group:

Object Type	State	Permissions	Generation Method
Cabinet	All	Read Modify	program
EPMDocument	All	Read	template
EPMDocument	In Work	Modify Create Delete	template
EPMDocument	Released	Revise	template
ManagedBaseline	All	Read	template
ManagedBaseline	In Work	Modify Create Delete	template
StructuredAnnotationSet	All	Read	template
StructuredAnnotationSet	In Work	Modify Create Delete	template

<b>Object Type</b>	<b>State</b>	<b>Permissions</b>	<b>Generation Method</b>
SubFolder	All	Read Modify	program
WTChangeActivity2	All	Read	template
WTChangeActivity2	Implementation	Modify	template
WTChangeActivity2	Open	Modify	template
WTChangeIssue	All	Read	template
WTChangeIssue	Open	Modify Create	template
WTChangeOrder2	All	Read	template
WTChangeOrder2	Implementation	Modify	template
WTChangeOrder2	Open	Modify	template
WTChangeProposal	Open	Modify Create	template
WTChangeRequest2	All	Read	template
WTChangeRequest2	Open	Modify Create	template
WTDocument	All	Read	template
WTDocument	In Work	Modify Create Delete	template
WTDocument	Released	Revise	template
WTPart	All	Read	template
WTPart	In Work	Modify Create Delete	template
WTPart	Released	Revise New View Version	template
WTPartAlternateLink	All	Read Modify Create Delete	template
WTPartSubstituteLink	All	Read Modify Create Delete	template
WTProductConfiguration	All	Read	template
WTProductConfiguration	In Work	Modify Create Delete	template
WTProductInstance2	All	Read	template
WTProductInstance2	In Work	Modify Create Delete	template



<b>Object Type</b>	<b>State</b>	<b>Permissions</b>	<b>Generation Method</b>
SubFolder	All	Read Modify	program
WTChangeActivity2	All	Read	template
WTChangeActivity2	Implementation	Modify	template
WTChangeActivity2	Open	Modify	template
WTChangeIssue	All	Read	template
WTChangeIssue	Open	Modify Create	template
WTChangeOrder2	All	Read	template
WTChangeOrder2	Implementation	Modify	template
WTChangeOrder2	Open	Modify	template
WTChangeProposal	Open	Modify Create	template
WTChangeRequest2	All	Read	template
WTChangeRequest2	Open	Modify Create	template
WTDocument	All	Read	template
WTDocument	In Work	Modify Create Delete	template
WTDocument	Released	Revise	template
WTPart	All	Read	template
WTPart	In Work	Modify Create Delete	template
WTPart	Released	Revise New View Version	template
WTPartAlternateLink	All	Read Modify Create Delete	template
WTPartSubstituteLink	All	Read Modify Create Delete	template
WTProductConfiguration	All	Read	template
WTProductConfiguration	In Work	Modify Create Delete	template
WTProductInstance2	All	Read	template
WTProductInstance2	In Work	Modify Create Delete	template

Object Type	State	Permissions	Generation Method
WTProductInstance2	Released	Revise New View Version	template

**Note:** As shown in the Generation Method column, the rules for Cabinet and SubFolder object types are defined programmatically when a product or library container is created; they are not defined through the template that is used. All other rules are defined through the template.

#### System Domain Rules for CONFIRMED Group

The following table lists the combination of object type, life cycle state, and granted permissions that the System domain out-of-the-box rules define for the CONFIRMED group:

Object Type	State	Permissions	Generation Method
WTExecutionObject	All	Modify Create Read	program
WTObject	All	Read	program

**Note:** As shown in the Generation Method column, these rules are defined programmatically when a product or library container is created; they are not defined through the template that is used.

#### Rules for the GUEST Group

Out-of-the-box General Product and General Library access control rules for the GUEST group (Guests role) are described in the following sections.

#### Default Domain Rules for the GUEST Group

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for guests of a product or library:

Object Type	State	Permissions	Generation Method
Cabinet	All	Read	program
SubFolder	All	Read	program
WTObject	All	Read	template

**Note:** As shown in the Generation Method column, the rules for Cabinet and SubFolder object types are set programmatically when a product or library container is created; they are not defined through the template that is used. The rule for the WTObject object type is defined through the template.

### System Domain Rules for the GUEST Group

The following table lists the combination of object type, life cycle state, and granted permissions that the System domain out-of-the-box rules define for the guests of a product or library:

Object Type	State	Permissions	Generation Method
WTOBJECT	All	Read	program

**Note:** As shown in the Generation Method column, this rule is defined programmatically when a product or library container is created; it is not defined through the template that is used.

### Rules in Default Domain for CHANGE REQUEST REVIEW BOARD Group

Out-of-the-box General Product and General Library access control rules for the CHANGE REQUEST REVIEW BOARD group are defined through the template and are described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
WTChangeRequest2	Under Review	Read	template
WTChangeOrder2	Under Review	Read	template

**Note:** No rules are defined in the System domain for the CHANGE REQUEST REVIEW BOARD group.

### Rules in Default Domain for PROMOTION APPROVERS Group

Out-of-the-box General Product and General Library access control rule for the PROMOTION APPROVERS group is defined through the template and is described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
PromotionNotice	Under Review	Read	template

**Note:** No rules are defined in the System domain for the PROMOTION APPROVERS group.

### Rules in Default Domain for PROMOTION REVIEWERS Group

Out-of-the-box General Product and General Library access control rule for the PROMOTION REVIEWERS group is defined through the template and is described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
PromotionNotice	Under Review	Read	template

**Note:** No rules are defined in the System domain for the PROMOTION REVIEWERS group.

### Rules for PRODUCT MANAGER and LIBRARY MANAGER Groups

Out-of-the-box General Product and General Library access control rules for the PRODUCT MANAGER group (Product Manager role) and LIBRARY MANAGER group (Library Manager role) are described in the following sections.

#### Default Domain Rule for PRODUCT MANAGER and LIBRARY MANAGER Groups

The following table lists the combination of object type, life cycle state, and granted permissions that the Default domain out-of-the-box rules define for the Product Manager or Library Manager:

Object Type	State	Permissions	Generation Method
WTObject	All	Full control (All)	program

**Note:** As shown in the Generation Method column, this rule is defined programmatically when a product or library container is created; it is not defined through the template that is used.

#### System Domain Rule for PRODUCT MANAGER and LIBRARY MANAGER Groups

The following table lists the combination of object type, life cycle state, and granted permissions that the System domain out-of-the-box rules define for the Product Manager or Library Manager:

Object Type	State	Permissions	Generation Method
WTObject	All	Full control (All)	program

**Note:** As shown in the Generation Method column, this rule is defined programmatically when a product or library container is created; it is not defined through the template that is used.

### Rules in Default Domain for CHANGE ADMINISTRATOR I Group

Out-of-the-box General Product and General Library access control rules for the CHANGE ADMINISTRATOR I group (Change Admin I role) are defined through the template and are described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
WTChangeActivity2	All	Full Control (All)	template
WTChangeIssue	All	Full Control (All)	template
WTChangeOrder2	All	Full Control (All)	template
WTChangeProposal	All	Full Control (All)	template
WTChangeRequest2	All	Full Control (All)	template

**Note:** No rules are defined in the System domain for the CHANGE ADMINISTRATOR I group.

### Rules in Default Domain for CHANGE ADMINISTRATOR II Group

Out-of-the-box General Product and General Library access control rules for the CHANGE ADMINISTRATOR II group (Change Admin II role) are defined through the template and are described in the following table. The table lists the combination of object type, life cycle state, and granted permissions that are defined in the Default domain out-of-the-box rules:

Object Type	State	Permissions	Generation Method
WTChangeActivity2	All	Full Control (All)	template
WTChangeOrder2	All	Full Control (All)	template

**Note:** No rules are defined in the System domain for the CHANGE ADMINISTRATOR II group.

## Updating Access Control Rules

**Caution:** PTC recommends that you do not modify the access control rules that are programmatically set when containers are created. Doing so can cause problems with access to data and administrative functionality.

To adjust the access control rules that are defined, you can use the Policy Administrator. Navigate to the product or library and click the **Policy Administrator** link from the **Utilities** page. Select the /Default domain and click **Update**. From the **Administrative Domain** window, click the **Access Control** tab to view the access control rules for the domain. From this tab, you can create new rules and update or delete existing rules. When creating or updating rules, the **Groups** tab shows the groups available from in the current context. These groups include the groups that map to the roles being used in the product or library team.

For additional information about creating or updating access control rules, see [Administering Access Control](#).

## Creating a Product


A Windchill PDMLink product container provides the context under which a team of people can create and manage all of the information that is relevant to the design, manufacture, and support of a customer product. This information includes the following:

- A defined data storage area for the business objects associated with a customer product.
- A set of rules that control the access to the product and optionally set the numbering scheme, versioning scheme, life cycles, and workflows that are used with the objects that are associated with a product.
- The team of users who have access to the product.

From within a product context, the product team produces an end item. The end item is the top-level assembly in the Bill of Materials for the product that is delivered to the customer. Also, if your company's product is made up of modular assemblies which can be sold separately, additional end items can be created within an existing product context. These additional end items can be children of the top-level end item.

**Note:** Product containers are created under an organization by members of the product creators group that is defined in the organization or by the organization administrator. Additionally, if the site administrator is a member of the organization (has the organization set in the organization attribute for Administrator user directory service entry), then the site administrator can also create products. The organization administrator can add users who are members of the organization to the product creators group by accessing the **Creators** link from the **Organization** tab.

The following procedure summarizes the steps needed to create a product (for detailed instructions, see the help available from the **Create Product** window):

1. Navigate to the **Product** tab and click **Create Product** .

**Note:** The icon appears only if you can create a product container.

2. Select the context template and fill in the other fields in the **Create Product** window.

**Note:** The slash (/) character is not allowed in a product name.

For a description of the contents of the templates, see [Out-of-the-box Product and Library Context Templates](#).

3. Determine whether you want the product to be accessible through the normal domain-based access control rules or accessible only to the product team members.

This selection is made through the **Private Access** check box. If the check box is selected, then the product domain structure inherits from the Private organization domain rather than the normal PDM domain. This means that access is restricted to use only the access control policies that are defined within the product context being created; access control policies are not inherited from the parent PDM context. For additional information on the domain structure, see [Managing Access to Data through Access Control Rules](#).

4. Click **OK** to create the product and close the window.

## Creating a Library


A Windchill PDMLink library container provides the context under which you can store and provide access to business information. For example, all documents owned by a department can be stored in a department library.

Libraries can also hold objects that are not related to a single product. For example, parts that are related to more than one product could be stored in a common parts library (such as a Commodity Parts or Engineered Parts library), from which you allow multiple product teams access to those parts.

In previous releases, a library was known as a repository.

**Note:** Library containers are created under an organization by members of the library creators group or by the organization administrator. Additionally, if the site administrator is a member of the organization (has the organization set in the organization attribute for Administrator user directory service entry), then the site administrator can also create libraries. The organization administrator can add users who are members of the organization to the library creators group by accessing the **Creators** link that is on the **Organization** tab.

The following procedure summarizes the steps needed to create a library (for detailed instructions, see the help available from the **Create Library** window):

1. Navigate to the **Library** tab and click **Create Library** .

**Note:** The icon appears only if you are a member of the library creators group for your organization.

2. Select the context template and fill in the other fields in the **Create Library** window.

**Note:** The slash (/) character is not allowed in a library name.

For a description of the contents of the templates, see [Out-of-the-box Product and Library Context Templates](#).

3. Determine whether you want the library to be accessible through the normal domain-based access control rules or accessible only to the library team members.

This selection is made through the **Private Access** check box. If the check box is selected, then the library domain structure inherits from the Private organization domain rather than the normal PDM domain. This means that access is restricted to use only the access control policies that are defined within the library context being created; access control policies are not inherited from the parent PDM context. For additional information on the domain structure, see [Managing Access to Data through Access Control Rules](#).

4. Click **OK** to create the library and close the window.

## Administering Teams

One of the main activities that a product or library manager has is to administer the team associated with the product or library. When a product or library is first created, a base set of roles for the team is established from the context template that is used in the creation. For information about these roles and the access control rules that are set for the roles that are in the out-of-the-box templates, see [Out-of-the-box Container Access Control Policies](#).

To administer a product or library team, perform the following activities:

- Establish the roles that you want used in the team. You can add, remove, or create new roles.

The base set of roles that are established through the out-of-the-box templates is the minimal set of roles that you should have for a team. (See [Out-of-the-box Container Participation](#).) Before removing any of these roles, consider the consequences of the removal. For example in a product team, you should not remove the Product Manager role because this role defines who can administer the product.



The roles you can add are roles inherited from your organization. For additional information about these roles, see [Installed Site Container Participation](#).

You can also create roles specifically for your product or library team. In addition to creating a new role through the **Team** page, you must define the access control rules for the role by using the Policy Administrator. As part of this activity, create the rules against the group that is created for the team role. This group has the same name as the role that you create.

**Note:** Any new roles that you create are not available for life cycle, workflow, or team templates.

- Add users to the team by adding the users in the specific roles.

You can add users by selecting individual users or by selecting groups that are available. For example, if the organization administrator has created groups for the organization, you can select one or more of these groups to be a member of a role.

**Note:** When the organization container is created, the Site administrator has the option of restricting user access. By default, users are restricted so that they see only users that belong to the organization. Selecting the **Allow entire user and group directory selection** check box when the organization container is created provides the ability to search for all users. If this check box was not selected when the container was created, the Site or organization administrator can update the organization container to select it.

- Change task assignments for members of a team.

As the membership of a team changes or members change responsibilities, you can manage the ownership of the tasks assigned to team members

From the **Product** or **Library** tab, click the **Team** link to access the **Team** table. From this table, you can administer a product or library team. For additional information about teams, see the [Administering Teams and Roles](#) chapter. Be sure to read the [Best Practices for Windchill PDMLink and Windchill ProjectLink](#) section in the chapter.

Click the **Assignments** link to access a table of assignments with the following views: **All**, **Open**, **Overdue**, **Assigned to me**, **Completed**, **Uncompleted**. From this table you can manage the tasks assigned to your team.

## Creating Custom Change Monitor Reports

Change Monitor reports are available from the **Change Monitor** link that is on the **Product** and **Library** tabs. The **All Special Reports** list on the **Change Monitor** page includes the out-of-the-box reports loaded when you installed Windchill PDMLink and all custom reports that are available. The availability of a custom report is determined by where the report is created. You use the Report Manager utility to create custom Change Monitor reports. This utility is available from the **Utilities** link that is on the **Site**, **Organization**, **Product**, and **Library** tabs:

- Custom reports created from the **Site** tab are available to all products and libraries.
- Custom reports created from the **Organization** tab are available to all products and libraries whose owning organization is the organization that was active when the custom report was created.
- Custom reports created from the **Product** tab are only available to the product that was active when the custom report was created.
- Custom reports created from the **Library** tab are only available to the library that was active when the custom report was created.

To create a custom Change Monitor report:

1. Navigate to the tab under which you want to create the report.
2. Click **Utilities** and then open Report Manager by clicking its link on the page.
3. Click **New** to create a new report.
4. Set the location under which the report is stored as follows:
  - a. From the **File** menu, click **Set Location**.  
The **Set Location** window opens.
  - b. From the **Look in Cabinet** field, select System (if it is not already in the field)
  - c. Open the System folder by double clicking it in the field displayed under the **Look in Cabinet** field.
  - d. Continue opening folders under the System folder until the path displayed in the **Location** field is:  
`/System/Reports/ChangeMonitor/Custom`
  - e. Click **OK**.
5. Create and save your custom report. For help with Report Manager activities, click the help icon in Report Manager.

To use the custom report, navigate to the applicable **Product** or **Library** tab and click the **Change Monitor** link. On the **Change Monitor** page that opens, the new report appears in the **All Special Reports** list. Click the custom report link from the list to execute the report.

## Using the Product and Library Utilities Page

The utilities listed on the **Utilities** page that is accessible by clicking the **Utilities** link from the **Product** and **Library** tabs allow you to perform administrative actions at a product and library level. The same set of utilities appears on both the **Product** and **Library** tabs. The difference is the context from which each utility is launched:

- Clicking a link from a product **Utilities** page launches the utility within the context of the current product.
- Clicking a link from a library **Utilities** page launches the utility within the context of the current library.

The utilities are grouped according to whether they are system administration utilities or business administration utilities. Many of the links provided on the page give you access to the utilities that you need to use to perform the duties described in [Typical Duties of Product and Library Administrators](#).

To explore the use of each utility, click the corresponding link on the page and then click the help icon in the window that opens.

## Configuring Product or Library Containers for Dynamic Documents

Product and library containers that will be used primarily for managing dynamic documents are generally no different from product and library containers used for managing parts. When setting up a product or library container for this purpose, review all of the activities described in the [Typical Duties of Product and Library Administrators](#) section, keeping in mind that the main business object type being managed is the Dynamic Document object type.

Since the out-of-the-box context templates do not have anything specifically set up for this object type, you may want to look closely at the following administration activities as they relate to managing this type of object:

- Create folders for use in organizing the dynamic documents that are checked into Windchill PDMLink.
- Create additional soft types and attributes for dynamic documents. The soft types and soft attributes correspond with Arbortext document types and attributes that are set up for use with the Arbortext Editor. See [Managing Types and Attributes for Dynamic Documents](#).
- Set up specific teams, life cycle templates, and workflow templates for use with dynamic documents. For details on these topics, see the associated

chapter: [Administering Teams and Roles](#), [Administering Life Cycles](#), and [Administering Workflow Processes](#).

- Review and set object initialization rules for Dynamic Document soft types. For information on setting object initialization rules, see [Administering Object Initialization Rules](#).

**Note:** Autonumbering must be turned on for the Dynamic Document soft type and its subtypes. By default, autonumbering is turned on for the wt.epm.EPMDocument type and all of its subtypes through an object initialization rule in the site container. Since dynamic documents are subtypes of the wt.epm.EPMDocument type, the rule automatically applies dynamic documents unless additional rules have been set up that override the site container rule. For information about numbering objects, see [Administering the Numbering of Parts, Documents, CAD Documents, and Change Objects](#).

- Set access control policies on the Dynamic Document soft types. For information on setting access control policies, see [Administering Domains and Policies](#).

Additionally, there are activities unique to using the Arbortext Editor and the Arbortext Publishing Engine with Windchill PDMLink that need to be performed. These include the following:

- Establish the configuration settings used by the Windchill connection to Arbortext Editor. For details, see the Windchill connection online help accessible from the Arbortext Editor.
- Manage the bursting rules used when a dynamic document is stored in Windchill. For information on bursting rules, see the Arbortext Editor online help.
- Manage additional soft attributes for WTDocument object type that are used by the WVS publisher when publishing dynamic documents. For information on specifying publishing rules, including specifying rules for the resulting documents, see the *Windchill Installation and Configuration Guide -Visualization Services*.

For general information about using Windchill PDMLink with Arbortext products to manage dynamic documents, see the [Getting Started with Windchill PDMLink and Arbortext Editor](#) guide.

# Configuring Part and Document Relationships

The relationships between a part and a document, two parts, or two documents vary depending upon how the two objects are linked. This section briefly describes the different types of links that form these relationships and then describes the configuration options that are available.

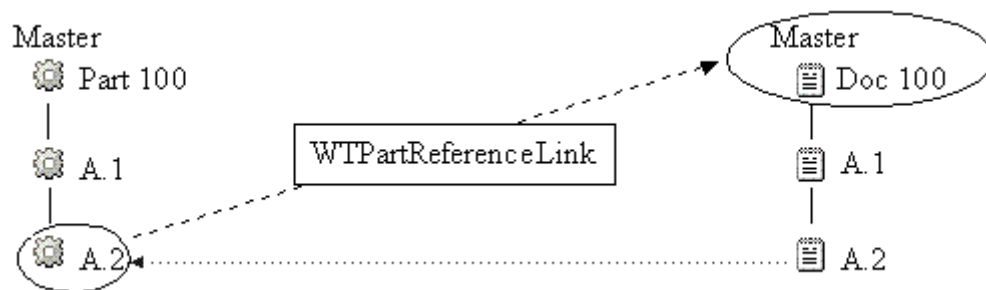
For detailed information about understanding relationships between parts and documents, see the *Windchill PDMLink User's Guide*.

## Part to Document Relationships

The association between a part and a document is created from the part information page, Product Information Manager, Product Structure Explorer, or Product Structure Browser. Associations can be viewed either from the part information page under the **Related Documents** link or from the document information page under the **Related Parts** link.

A user associates a document to a part using the **Associate Document** action. There are two types of links that can be established:

- A WTPartReferenceLink -- Using this type of link always links the part to the master document. Regardless of the version of the document the user selects, the part will always link to the master document when this link type is used. This is true even when the system has been configured to allow the user to select the link type and the document type.



Reference links always display a document version of the master based on life cycle state rules. For example, when a user clicks the reference link for Part 100 version A.1, Windchill PDMLink searches for the latest released version of Doc 100 to display. If no version of the document has been released, it displays the latest working version (in this case, Doc 100 version A.2).

How to configure the life cycle state rules is described later in this section.

- A WTPartDescribeLink -- Using this type of link links the part to the latest iteration of a document and vice versa.



For example, when the user clicks a describe link for Part 100 version A.2, Windchill PDMLink searches for the latest iteration of Doc 100 to display (in this case, Doc 100 version A.2).

You can configure the link types for part to document relationships in the following ways:

- You can have the system determine the type of relationship based on the document soft type. This is the default  
  
If the system determines the link and the document soft type is Reference Document, a WTPartReferenceLink is created; otherwise, a WTPartDescribeLink is created.
- You can allow the user to select the link type regardless of the document soft type.

To allow the user to select the link type regardless of the document soft type, set the following preference using the Preference Manager from the **Site Utilities** page:

```
Key=/wt/part/AssociateDocToPartPrefs=true
```

For details on using the Preference Manager, see the *Windchill System Administrator's Guide*.

Additional configuration options are described in the following sections.

For additional information on part to document relationships, see the *Windchill PDMLink User's Guide*

## Revised or Saved Part to Related Document

When a user revises a part using the **Revise** action or saves the part using the **Save As** action, the new version of the part carries forward the link to the document by default. While the **Revise** action always carries the link forward, you can choose to prevent the link from being carried forward during a **Save As** action by removing the Relationship copy rules related to this operation from wt.properties.

For example, assume the following properties are set in wt.properties:

```
wt.enterprise.copyRuleDelimiter=,  
  
wt.enterprise.copyRulesN=wt.part.WTPart,Relationship,  
wt.part.WTPartReferenceLink-references
```

The first property sets the delimiter for the copy rules to the comma (,).

The wt.enterprise.copyRulesN property is the Relationship copy rule for wt.part.WTPart. This rule copies the references forward when the type of link is WTPartReferenceLink.

If you remove the wt.enterprise.copyRulesN property, then reference links are not carried forward.

**Note:** There can be no gaps in the sequence of copy rules. If you remove a copy rule, you must renumber those rules that follow. For example if there are six copy rules and you remove copyRules5, then you must renumber copyRules6 so that it is copyRules5.

Use the xconfmanager utility when modifying the wt.properties file. For information on using this utility, see [About the xconfmanager Utility](#). For additional information about the properties used for copy rules, see the description of wt.enterprise.copyRules in the properties.html file.

**Caution:** PTC recommends that you do not change the value of the wt.enterprise.copyServiceRules property. The property is used by internal services.

## Document Version Used with Reference Link

As described earlier, reference links (WTPartReferenceLink) link to a document master, but display a document version of the master based on the life cycle state rules for the document.

The default behavior is that Windchill PDMLink searches for the latest released version of the document to display. If no version of the document has been released, it displays the latest working version of the document.

To change the default behavior, change the following property in the wt.properties file using the xconfmanager utility (or System Configurator):

```
WT.part.WTPartReferenceLink.validStateKeys="<state1>,<state2>,...,<stateN>"
```

**Note:** The states must be valid life cycle states. The states are defined as key value pairs in StateRb.rbinf and can be viewed in the life cycle template associated with the object. States are always specified using uppercase characters.

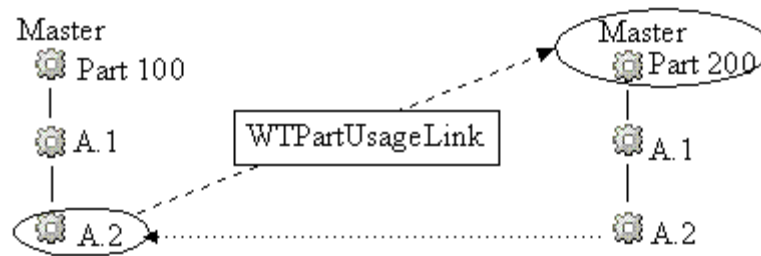
For example, change the search to include the Released, Approved, and then Completed states of a document, enter the following command from a windchill shell:

```
xconfmanager -s
WT.part.WTPartReferenceLink.validStateKeys="RELEASED,APPROVED,COMPLETED"
-t wt.properties -p
```

After this property is set, Windchill PDMLink searches for the latest Released version first. If none is found, it searches for the latest Approved version. If none is found, it searches for the latest Completed version. So, if a part is linked to Reference Document 4 which has three versions (A, B, and C), where A = Released, B = Approved, and C = In Work, based on the state settings in wt.properties, Windchill PDMLink displays the latest iteration of version A which was released and ignores the rest.

## Part to Part Relationships

A user associates one part with another by using the Product Structure Explorer. When a user makes the association, a WTPartUsageLink is created, forming a "uses part" relationship between a part and a part master.



The only configuration option for part to part relationships is described in the next section.

For additional information on part to part relationships, see the *Windchill PDMLink User's Guide*

## Revised or Saved Parent Part to Child Part

When a user revises a parent part using the **Revise** action or saves the part using the **Save As** action, the new version of the part carries forward the usage link.

To prevent a link from being copied forward on a **Revise** or **Save As** action, you must remove copy rules from wt.properties. For example, assume the following properties are set in wt.properties:

```
wt.enterprise.copyRuleDelimiter=,
wt.enterprise.copyRulesN=wt.part.WTPart,Relationship,
wt.part.WTPartUsageLink-uses
```



The first property sets the delimiter for the copy rules to the comma (,).

The `wt.enterprise.copyRulesN` property is the Relationship copy rule for `wt.part.WTPart`. This rule copies the references forward when the type of link is `WTPartUsageLink`.

If you remove the `wt.enterprise.copyRulesN` property, then the usage links are not carried forward.

**Note:** There can be no gaps in the sequence of copy rules. If you remove a copy rule, you must renumber those rules that follow. For example if there are six copy rules and you remove `copyRules4`, then you must renumber `copyRules5` and `copyRules6` so that `copyRules5` becomes `copyRules4` and `copyRules6` becomes `copyRules5`.

Use the `xconfmanager` utility when modifying the `wt.properties` file. For information on using this utility, see [About the xconfmanager Utility](#). For additional information about the properties used for copy rules, see the description of `wt.enterprise.copyRules` in the `properties.html` file.

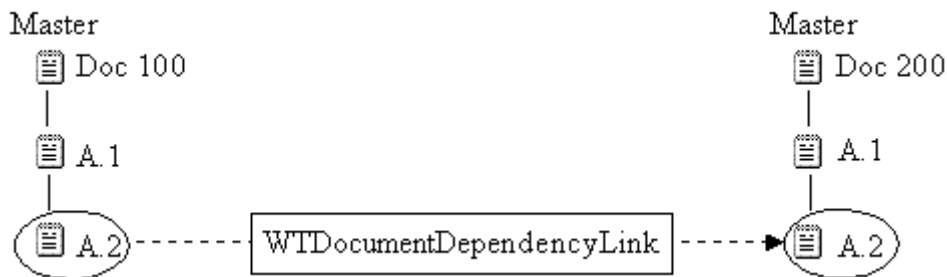
**Caution:** PTC recommends that you do not change the value of the `wt.enterprise.copyServiceRules` property. The property is used by internal services.

## Document to Document Relationships

There are several different types of document to document relationships in Windchill PDMLink:

- A document can reference another document under **Related Documents**.
- A document can reference another document as an attachment.
- A document can serve as the parent to a second child document in the **Document Structure**.

All of these relationships create a link of type `WTDocumentDependencyLink`.



There are no configuration options available for document to document relationships.

For additional information on document to document relationships, see the *Windchill PDMLink User's Guide*.

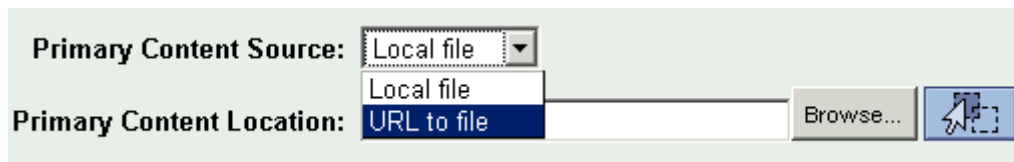
## Allowing URLs as Content for Documents

By setting a preference, you can allow URLs to be used as content for documents. Each URL is then managed using process capabilities such as versioning and change management. As a document evolves, a physical file can eventually replace the URL as the primary document content. Additionally, setting the preference allows URLs to be used as attachments or secondary content.

Create the following preference using the Preference Manager (accessible from the **Site** tab **Utilities** page) to allow URLs to be used as content for Windchill documents:

```
/com/ptc/windchill/doc/exposeUrlContent
```

Specifying this preference as TRUE adds the **URL to file** value to the **Primary Content Source** field in the document create, update, and check-in windows. When **URL to file** is selected from the **Primary Content Source** drop-down list, only the text field is available for the **Primary Content Location** field; the **Browse** button and drag-and-drop target are hidden.



The screenshot shows a form with two main sections. The first section is labeled "Primary Content Source:" and contains a dropdown menu with "Local file" selected. Below this, a second section is labeled "Primary Content Location:". It features a text input field with "URL to file" entered, a "Browse..." button, and a drag-and-drop icon. The "URL to file" option is highlighted in the dropdown menu.

By default, the **URL to file** value is not available. For additional information on setting preferences, see the help available from the Preference Administrator.

## Best Practices For Object Initialization Rules

Although you can set object initialization rules for individual product and library containers, consider the consequences of doing this. The object initialization rules include the ability to set the following for each object type:

- Folder paths
- Default life cycle template
- Default team template
- Default numbering scheme
- Default versioning scheme

For most sites, the recommended approach is to set the rules at either the organization or site level so that an entire organization or site is using the same set of rules.

If you choose to set different rules in a product or library container, be aware that the ability to move objects between containers may be restricted. Specifically, if you create a life cycle template in a specific product or library and the template is not available in other products or libraries, then you cannot move any object that uses the life cycle template. The same is true if the team template is unique to a product or library; objects using the team template cannot be moved. Similarly, the versioning scheme used for an object must match the scheme set in the destination product or library or the object cannot be moved.

**Note:** For demonstration purposes, the Product Design template (available by default when you create a product) includes object initialization rules that set a different default life cycle templates and versioning schemes for parts, documents, and CAD documents than is set out-of-the-box at the site level. This template demonstrates the use of simplified life cycles and state-based versioning. Unless you have overriding business practices that required unique rules, PTC recommends that if you want to use these features, you modify the organization or site object initialization rules and remove them from the Product Design template.



# 7

## Administering Projects

This chapter provides an overview for administering projects in Windchill ProjectLink and describes the typical duties that a project manager performs. It also provides additional information about some of the main administrative tasks for projects.

<b>Topic</b>	<b>Page</b>
Overview .....	7-2
Typical Duties of Project Managers .....	7-2
Out-of-the-box Project Templates .....	7-7
Creating a Project .....	7-10

## Overview

Project managers (also known as project administrators) are responsible for creating and managing projects hosted by a parent organization in the system.

Project managers create and configure the project and control the membership of the project teams within the confines of a project application container. They control access to the project information, and they define the project schedule, resources, and plan details.

Project application containers are used to define new projects and to collect all the information associated with the project. Project containers are defined by project creators who are authorized by the parent organization. Projects inherit templates, roles, groups, and policies from their parent organization container. In addition, the project manager can define project-specific templates, groups, roles, and policies.

For general information about container contents and how to create containers, see [Administering Containers](#).

## Typical Duties of Project Managers

Project managers are responsible for creating and managing projects. Your typical duties include the following:

- Creating and updating project space
- Managing project team members and roles
- Creating, updating, and managing documents and folders
- Creating, updating, and managing activities, deliverables, resources, and action items
- Managing document templates
- Importing and exporting information

The following sections describe some of the duties in more detail.

### Creating and Updating Project Space

You create the project instance in the context of a parent organization. You can choose the project template on which the project is based, define the key project attributes, define the project properties and select the configuration options. You can also update the project attributes as the project progresses. These attributes include the project description, scope, status description, risk value and description, project number, and so forth.

Only members of the project creators group and the organization administrator in an organization are allowed to create projects. By default, organizations allow all members to create projects, but the organization or site administrator may update an organization restricting project creation privileges to identified creators.

For more information about creating a project, see *Beginning a Project in the Windchill ProjectLink User's Guide*.

## Managing Project Team Members and Roles

You have exclusive control over your project team. Only project managers can add team members and roles.

### Establishing Roles

Each project team contains two fixed roles that cannot be removed: Project Manager and Guests. The Guests role is designed to include groups and users that are not active project team members and need only read access to project information.

**Note:** It is possible to hide the Guest role for projects. For more information, see [Hiding the Guests Role](#) below.

Guest members do not receive project invitations; projects in which a user is a guest do not show up on project lists for the user. You can allow guests to see projects on their project lists by adding the following property and value to the wt.properties file:

```
wt.inf.team.displayContainerListForGuest=true
```

**Note:** Use the -s parameter on the xconfmanager command to add Windchill properties. See [About the xconfmanager Utility](#) for details. This property controls the guest visibility on project lists as well as product and library lists.

Additionally, you can select project team roles from the list of roles inherited from the parent organization, or you can create new roles.

The creator of the project is automatically established as both a project manager and as the project owner. The project manager can change the owner and can add members to the project managers group. All members of the project managers group have the same privileges.

You can invite groups to the project roles from the groups defined in the parent organization under which you create the project.

### Controlling the Visibility of Actions By Role

As the project manager, you can control visibility to information and actions in a project based on the roles in a project team. On the project **Team** page, set the **Current View** to **Members By Role** to view the roles currently used in the project. By selecting roles and then clicking the **Configure Roles** icon, you can display the list of actions for which visibility can be controlled. For example, to configure visibility of actions for the members of the Reviewer role, select **Reviewer** and then click **Configure Roles**. From the **Configure Roles** window, select the check boxes below the role name to allow the action to be visible.

**Tip:** When selecting the roles to configure, it is good practice to show the **All Members** role. This is because selecting other roles for a given action clears the selection of the **All Members** role. See the online help for additional information.

**Note:** If a user is a member of multiple roles, the user has access to the union of the actions. For example, if a user can view the **Team** page because visibility is set in one role, the user keeps that ability even if other roles do not grant this visibility. For details on configuring visibility to actions, see the online help available from the **Configure Roles** window.

If you have configured some roles to restrict the visibility of some of the actions in a project, you include those visibility restrictions when you export the project or save the project as a new project or a new project template. To include the restrictions, be sure to select team roles when exporting a project as a template or saving the project as a new project or a new project template. For details on exporting or saving a project as a template, see [Creating Context Templates](#) in the Administering Containers chapter. The XML tags used for exporting and then importing team roles are described in the [ExportedRoleMemberMap Element](#) section of the Creating Business XML Files for Templates appendix.

## Hiding the Guests Role

To hide the Guests role in a project team, a site or organization administrator can set the value of the following property to TRUE using the Preference Manager on the **Utilities** page of the **Site** or **Organization** tab:

```
ProjectLink/hideGuestsGroup
```

Once the preference is set to TRUE, the Guests role is then hidden to users in all existing and new projects. The preference value can be changed at any time, however the change affects all projects within your site or organization, rather than an individual project. Setting this preference does not require a restart of the server, but does require that the user making the change log out and then log back into the system before they can see the change.

**Caution:** If a user is in the Guests role of a project when the above property is set to TRUE, then that user's access control rules are not altered, and that user cannot be removed from the now hidden Guests role until and unless the property is set back to FALSE and an administrator manually removes the user from that role.

## Creating, Updating, and Managing Documents and Folders

You can create documents for the project, and you can define a folder structure. By default, any member of the project can define folders and subfolders in a project.

You have full control over all documents created in the project; no member can prevent you from reading, updating, or deleting any object. You can also modify the access rules on any folder or any document.



You can delete discussion forum topics and postings. Non-project managers can only post to discussion forum topics; they cannot delete posting or topics, even if they created them.

## Creating, Updating, and Managing Activities, Deliverables, Resources, and Action Items

You alone can create activities, milestones, deliverables, and resources in the project. The owner of an activity, milestone, or deliverable can update and complete the item, but only project managers can create plan items.

### Limiting Update Privileges for All Action Items

You can limit the update privileges on all action items, preventing the action item assignee from updating the following fields:

- Name
- Description
- Assignee
- Due Date
- Priority

To limit the update privileges for all action items, set the following preference using the Preference Manager from the **Utilities** page on the either **Site** or **Organization** tab:

```
Key=/wt/meeting/actionitem/restrictUpdateByAssignee=true
```

Assignees will not be allowed to update the fields listed above for any action items.

## Managing Project Templates

You control the document templates available to a project. You can add to the list of document templates inherited from the parent organization or site, can override the inherited template with a project-specific template of the same name or can disable a document template that was created within the context of the project. Disabled templates do not show up in the list of available templates when using the **Create (Document) from Template** action in the project folders page.

## Importing and Exporting Information

You can export information from one project and import it into another project on the same system or a different Windchill system. Generally, you can only import into a system at the same or higher release version as the system from which the data was exported. You can export project information as a template. A site or

organization administrator can then create a template from that information that can be reused by the organization.

You can also save an existing project as a new project as a means of quickly starting a similar project.

You can also import a Microsoft Project plan into the system and export project information from Windchill into the Microsoft Project format. You must install a Microsoft Project plug-in utility to take advantage of this capability.

## Optional Utilities (Optional Access Through Project Utilities Link)

A site or organization administrator has access to a number of utilities through the **Utilities** link under the **Project** tab. The site or organization administrator can set a preference to grant all project managers access to the project utilities functions. By default, only site and organization administrators can access the project utilities and must act on behalf of the project manager. Access is restricted by default because these utilities are complex and require a level of training that is not expected for typical project managers.

The preference that must be set to expose the **Utilities** link to project managers is:

```
\ProjectLink\AllowPMUtilitiesAccess
```

The following utilities are available through the **Utilities** link:

- Numbering Schemes Administration
- Object Initialization Rules Administrator
- ProductView Configuration Administrator
- Publish Monitor
- Publish Scheduler Administrator
- Replication Administrator

### Numbering Schemes Administration

You can update the numbering scheme applied to establish uniqueness for documents and parts in a project.

### Object Initialization Rules Administrator

You can configure how objects of various types are created and their attributes are created and initialized in the project. The Object Initialization Rules Administrator can define how the initial values for the object are established and can determine basic relationships such as life cycle association when an object is created in the organization. The rules established by the parent organization are inherited by each project by default, but they can be overridden by a project. The rules are defined in an XML format; you can view and edit them if a site or organization administrator has given you access to them.

## ProductView Configuration Administrator

If given access to this utility, you can manage the ProductView configurations to set up watermark files that users see when they view document and part content from the project.

## Publish Monitor

You can access the Publish Monitor utility to monitor the publishing of viewable files generated when CAD models are checked into the project.

## Publish Scheduler Administrator

You can manage the scheduling of publishing CAD files that are checked into the project.

## Replication Administrator

You can establish project content replication rules. Replication is useful when sharing large files with remote sites where the remote users have low bandwidth connections to the Windchill server. Replication sites are configured so that document and part content files are replicated at a remote site and made available locally to the remote users through a higher bandwidth connection. Replication can decrease the time required to upload and download files at the remote site. Site administrators must define the replica site and the replication schedule; project managers configure the replication rules for a particular replication site established for the system.

Define replication only for those projects in which remote sites are participating. Replication imposes a substantial performance burden on the Windchill server.

Replication rules can be configured on a project-by-project basis. See the *Windchill System Administrator's Guide* for information about replication.

## Out-of-the-box Project Templates

When Windchill ProjectLink is installed, the following project templates are loaded:

Template Name	Description
General	A sample template that can be used to create a general project.
Design	A sample template that can be used to create a design project.
Manufacturing	A sample template that can be used to create a manufacturing project.

Template Name	Description
Custom	A sample template that can be used to create a custom project.

The project templates can define the same basic information that is discussed in the [Container Administrative Items](#) section of the Administering Containers chapter. However, the out-of-the-box templates define only the following:

- Container structure
- Container environment
- Container participation

The following sections describe the items that are defined in the templates.

Container Structure				
	General	Design	Manufacturing	Custom
Folder structure	General	Analysis Contracts General Designs Plans Prototypes Specifications Standards	Analysis Contracts Designs General Plans Specifications	Changes Contracts Designs General Installation Plans Prototypes Specifications

Container Environment				
	General	Design	Manufacturing	Custom
Forum topics	General	Analysis Design General Manufacturing Specifications	Design General Manufacturing Quality Specifications	Changes Contracts Design General Manufacturing Quality Specifications
Reference folders	Documents General Links Parts	Documents General Links Parts	Documents General Links Parts	Documents General Links Parts

Container Participation				
	General	Design	Manufacturing	Custom
Project team roles	Guests Members Project Manager	Consultant Designer Guests Manufacturer Members Project Manager	Engineer Guests Manufacturer Members Production Planner Project Manager Purchasing Agent Quality Engineer Supplier	Customer Engineer Guests Members Project Manager Quality Engineer Supplier

Template Plan Objects				
	General	Design	Manufacturing	Custom
Milestones	Project Start Project Finish	Project Start Contract Signed Specifications Approved Design Approved Prototype Approved Design Released Project Finish	Project Start Contract Signed Specifications Approved Design Approved Testing Complete Prototype Approved Production Part Approved Production Initiated Project Finish	Project Start Contract Signed Specifications Approved Design Approved Testing Complete Prototype Approved Production Part Approved Production Initiated Project Finish
Deliverables	Scope Post Mortem	Scope Contract Specification Draft Design Prototype Final Design Post Mortem	Scope Contract Specification Draft Design Test Report Prototype Production Part Production Initiated Post Mortem	Scope Contract Specification Draft Design Test Report Prototype Production Part Production Plan Post Mortem

## Creating a Project

**Note:** Members of the project creators group of an organization create containers under that organization. By default, organizations allow all members to create projects, but the organization or site administrator may update the organization restricting project creation privileges to identified creators.

For information about creating projects, see Beginning a Project in the *Windchill ProjectLink User's Guide* or the help available from the **Create Project** window.

For information about project plans, see Managing Project Plans in the *Windchill ProjectLink User's Guide*.

# 8

## Administering Windchill PDM Library

This chapter provides the an overview for administering products in Windchill Foundation & PDM and describes the typical duties that an administrator does. It also provides additional information about some of the main administrative tasks for the Windchill Foundation & PDM library container.

Topic	Page
Overview .....	8-2
Typical Duties of Windchill Foundation & PDM Administrators .....	8-2
Out-of-the-box Windchill Foundation & PDM Library Contents.....	8-5
Creating Groups for Use in Site Domain Policies.....	8-6
Improving Performance on Searches from Windchill Explorer.....	8-7

## Overview

The Windchill Foundation & PDM library container is created during the installation process and the initial Windchill Foundation & PDM administrator (also known as the site or system administrator) is defined. How this container works with the other installed containers and how to set additional administrators is described in the [Getting Started](#) chapter.

The Windchill Foundation & PDM administrators are responsible for administering business and system functions in Windchill Foundation & PDM. They define the specific life cycles, templates and processes, and monitor and manage the product activities.

The Windchill PDM library container is used to define new product models or instances and to collect all the information associated with the product.

For general information about containers and container contents, see [Administration Overview](#) and [Administering Containers](#).

For information about the creating products and navigating throughout the solution, see the *Windchill Foundation & PDM User's Guide*.

## Typical Duties of Windchill Foundation & PDM Administrators

Windchill Foundation & PDM administrators are responsible for administering the system and business aspects of products.

The system administration aspects include the following:

- Importing and exporting data
- Managing external storage
- Managing preferences
- Managing replication

The business administration aspects include managing the following:

- Document and CAD document templates
- Life cycle templates
- Workflow templates
- Team templates
- Report templates
- Object initialization rules (including numbering and versioning)
- Policy access rules
- ProductView and visualization



The following sections describe some of the duties in more detail.

## Managing Groups and Roles

You define groups and roles for use in the products being created.

For additional information on groups, see [Administering Principals](#).

For additional information on roles, see [Administering Teams and Roles](#).

## Managing Templates

You can define a number of document, CAD document, life cycle, team, report, and workflow templates that you want used in Windchill Foundation & PDM library. The library inherits the templates defined by the site. You can add to or just use these templates, or you can override the templates by defining specific templates with the same name as the site templates you want to override.

For additional information about templates, see [Container Templates](#).

## Managing Object Initialization Rules

You can configure how objects of various types are created and their attributes initialized by using the Object Initialization Rules Administrator. With this utility, you can define how the initial values for the object are established and can determine basic relationships such as life cycle association when an object is created in the product.

The rules established by the site are inherited by the Windchill Foundation & PDM library. The rules are defined in an XML format that you can view and edit.

For additional information about object initialization rules, see [Administering Object Initialization Rules](#).

## Managing Access Policies

You can define policies that control the level of access to information in a product. When defining a policy, the object types defined by the site can be used as well as object types defined in Windchill Foundation & PDM library. For example, you could create a policy that provides read access to all documents of type “Quality Assessment” to the product team role/group called “Testers”.

For general information about domains and policies, see [Administering Domains and Policies](#).

For additional information about creating or updating access control rule policies, see [Administering Access Control](#).

## Configuring Numbering and Versioning Schemes

You can configure the number and versioning scheme used to uniquely identify parts, documents and other objects in Windchill Foundation & PDM library. The numbering and versioning schemes defined at the site level is inherited by Windchill Foundation & PDM library by default. Use the Object Initialization Rules Administrator to configure the number and versioning schemes.

For additional information, see [Administering Object Initialization Rules](#).

## Managing Viewable Publishing

You can monitor and manage the publishing of viewable files that are optionally generated when CAD models are checked into products. You can also configure and update the watermarks used by the Windchill ProductView visualization tool when viewing document and part content from the product.

For additional information, see [Administering Visualization Services](#).

## Managing Custom Reports

You can create and update custom reports against the objects and attributes defined within the product or library. These reports are created using a report generation utility. The reporting utility is designed for use by those with a working knowledge of the Windchill data object model.

For additional information, see [Administering Audit Reports](#).

## Importing and Exporting Information

You can import information into a product, and can export information to a local file system. The import/export facilities support information exchange with another Windchill server or non-Windchill system. These utilities read and write system information in an XML format. For general import and export information, see the Windchill Import and Export chapter in the *Windchill System Administrator's Guide*.

The workflow and life cycle administration utilities accessible to the site administrator integrate the import and export functions with administering workflows and life cycles.

For life cycle import and export information, see [Import and Export](#) in the Administering Life Cycles chapter.

For workflow import and export information, see [Import and Export](#) in the Administering Workflows chapter.

## Configuring External Vaults or Replication Sites to Optimize Performance

You can configure the vaulting and replication rules for the external file and replica sites established by the site administrator. When external vaults are configured, document and part content is stored on a file system rather than in the database. This configuration can provide significant upload performance improvements and is appropriate when the site is frequently used to exchange large files (such as CAD model files).

For how to set external file vault rules, see the *Administering External File Vaults* chapter in the *Windchill System Administrator's Guide*.

Replication sites can also be configured so that document and part content files are replicated at a remote site where local users have only very low bandwidth connections to the Windchill server. Site administrators typically define the replica site and the replication schedule.

For how to set replication rules, see the *Administering Content Replication* chapter in the *Windchill System Administrator's Guide*.

## Out-of-the-box Windchill Foundation & PDM Library Contents

When Windchill Foundation & PDM is installed the following Change Management administrative items are loaded into the Windchill PDM library container:

- The ChangeItems administrative domain
- The ChangeItems cabinet
- The following life cycle templates:
  - Change Activity
  - Analysis Activity
  - Change Investigation
  - Change Issue
  - Change Order
  - Change Proposal
  - Change Request
- The following team templates:
  - Change Control Board
  - Change Team

- The following workflow templates:


- Change Activity Process
- Change Analysis Process
- Change Investigation Process
- Change Issue Process
- Change Order Process
- Change Proposal Process
- Change Request Process 1
- Change Request Process 2

**Note:** Windchill Foundation & PDM library container inherits administrative items from the Site container as described in [Distributed and Hierarchical Administration](#) and other sections of the Administering Containers chapter.

## Creating Groups for Use in Site Domain Policies

If you want to create policy rules for domains that are in the Site context and use groups (rather than individual users) as the principals to which the rules apply, then there is a multi-step process that you can follow to create the groups in the Site context.

To create a group in the Site context use the following steps:

1. From Site Map, open Policy Administrator.
2. Select the Site context and then select a domain in the Site context and click View.
3. From Administrative Domain Window of that domain, click  to open the Principal Administrator in the Site context.
4. From the Principal Administrator, create the group in the Site context.

You can then use the group created when setting policy rules for domains in the Site context.

## Improving Performance on Searches from Windchill Explorer

If users conduct a search that contain a large result set, the additional attributes requested on Windchill objects can cause some loss in performance.

If a user is experiencing a long wait for each results, they should stop the search, and enter in more specific search criteria to limit the result set being returned.

If poor search performance is a persistent issue at a site, you can customize the searches by removing attributes from the default Windchill Explorer interface. This is done by modifying `wt.clients.query.QueryPanelOptions.java`. For example, the out-of-the-box set up for a `WTDocument` is shown below (see the `JavaDoc` for `wt.clients.beans.query.WTSchema` for explanation of the different tags):

```
{"All Document Types", "C:wt.doc.WTDocument; K:61; A:number; " +  
"A:name; A:docType; A:versionIdentifier; A:lifeCycleState; " +  
"A:teamTemplateId; K:62; A:cabinet; A:modifyTimestamp; D:organizationReference; " +  
"D:formatName; A:format; D:organizationUniqueIdentifier;"},
```

To remove the `organizationReference` and `organizationUniqueIdentifier` attributes from the search (as these attributes were found to give the largest performance gains), the modified string is as follows:

```
{"All Document Types", "C:wt.doc.WTDocument; K:61; A:number; " +  
"A:name; A:docType; A:versionIdentifier; A:lifeCycleState; " +  
"A:teamTemplateId; K:62; A:cabinet; A:modifyTimestamp; D:formatName; A:format;"},
```



# 9

## Administering Principals

This chapter describes the principals (user, group, and organization objects) that are used in your Windchill solution. It also provides details about managing the principals.

Topic	Page
Overview .....	9-2
Using the Principal Administrator.....	9-4
Understanding Principals .....	9-7
Best Practices for Assigning Domains to Principals.....	9-8
Searching for Users and Groups.....	9-9
Managing Users.....	9-9
Managing Groups .....	9-14
Managing Organizations .....	9-18
Receiving Administrative Notifications.....	9-20
Managing the Principal Cache .....	9-20
Maintaining the Connections between Principal Objects and their Directory Service Entries.....	9-22

## Overview

Windchill uses the term *principal* to mean a user, group, or organization. It uses *principals* to mean any combination of users, groups, or organizations.

As the Windchill system administrator for any Windchill solution, you can create and update Windchill user, group, and organization objects through the Principal Administrator. As an Organization Administrator in Windchill PDMLink and Windchill ProjectLink, you can update the Windchill user, group, and organization objects that are in the organization context that you are administering.

**Note:** When a Windchill solution is installed, the system administrative user (Administrator), the system administrative group (Administrators), and the host organization object are always created. By default, the user Administrator (for example, wadmin) belongs to the Administrators group. This user does not have an organization affiliation (as defined by the organization attribute, which is "o" by default). Members of the Administrators group are granted all the permissions required to accomplish the administration tasks described in this guide.

**Note:** If your site does not use the organization attribute in the directory service entry, users can be assigned to an organization using the `usersOrganizationName` property. For more information on using this property, see the Configuring Windchill to Use an Enterprise Directory chapter of the *Windchill Installation and Configuration Guide - Windchill*.

Windchill uses both the Windchill database and a directory service when creating principals. For each principal, there is an entry in a directory service and a Windchill object stored in the database:

- The directory service entry contains attributes specific to the type of principal. For example, user entries have attributes for the user's full name, e-mail address, and organization.

The Aphelion Directory Service is set up when your Windchill solution is installed. Other directory services can be established by setting up JNDI adapter entries through the Info\*Engine Property Administrator and adding the adapter entries to the `wt.federation.org.directoryServices` property value. For additional information, see the *Windchill Installation and Configuration Guide - Windchill*.

- The Windchill object contains information that is relevant to Windchill (such as the associated domain) and the *Unique Federation Identifier (UFID)* associated with principal.

The UFID contains the distinguished name of the principal and identifies the directory service where the principal entry resides.

The following sections provide additional details about Windchill principals.



## Windchill Users

A Windchill user object identifies a user and is used when establishing group membership and policy rules for that user. It is stored in the Windchill database and holds user information for those users who have access to Windchill. This information includes the user name, the UFID associated with the user, the Windchill domain of the user, and administrative flags that are set if the object needs to be repaired or is disabled.

A Windchill user object is automatically created and persisted in the Windchill database the first time the user is selected from a search or the first time the user logs on to Windchill. In both of these cases, the corresponding directory service entry for the user already exists and is then referenced in the object that is created. As an administrator, you can also create, update, and delete users through the Principal Administrator.

Windchill does not rely on the user object to authenticate users. Rather, users are authenticated when they log on to the Web server. The user's Web server ID is then mapped directly to the user object that has a matching user name.

Windchill users are usually affiliated with an organization that is set through the directory service organization attribute (by default, "o"). If the organization attribute is not set, then the user is an unaffiliated user and cannot create products, libraries, or projects from Windchill PDMLink and Windchill ProjectLink. However, that user can be invited to a team by e-mail and can do the same things within the product, library, or project as any other member.

**Note:** If your site does not use the organization attribute in the directory service entry, users can be assigned to an organization using the `usersOrganizationName` property. For more information on using this property, see the *Configuring Windchill to Use an Enterprise Directory* chapter of the *Windchill Installation and Configuration Guide - Windchill*.

## Windchill Groups

Organizing users into Windchill groups provides you with a more efficient way to apply policies for access control, indexing, and event notification. Each group object identifies selected users, organizations, and possibly other groups, under one name. You can create groups so that you can efficiently apply administrative policies (such as those for access control, indexing, and notification policies) to sets of users, rather than to each user individually.

Groups are associated with the context in which they are created. From the Principal Administrator, you only have access to the groups created through the Principal Administrator in the current context or in ancestor contexts. Some Windchill solutions also create and manage internal groups that are used to manage team role membership. These groups are not accessible from the Principal Administrator.

A Windchill group object holds the group name, the UFID associated with the group, the Windchill domain of the group, and administrative flags that are set if the object needs to be repaired or is disabled. The UFID contains the distinguished name of the group and identifies the directory service where group entry resides.

## Windchill Organizations

Categorizing users by organization provides an additional way in which you can identify a set of principals by name. The Windchill Foundation & PDM solution does not make extensive use of organization objects; however, it does provide an organization ownership feature that can be turned on to identify which organization owns specific Windchill objects (see Administering Organizations in the *Windchill System Administrator's Guide*). Other Windchill solutions make use of organization objects within organization contexts to manage Windchill objects within each organization context.

An organization can be a company, company division, university, or some other list of people. Organization membership is defined by user directory entries that include the organization name defined for the organization object. For example, if the organization name is DIV1, then all users in the configured directory services who have their organization attribute (o, by default) set to DIV1 are members of the DIV1 organization.

**Note:** If your site does not use the organization attribute in the directory service entry, users can be assigned to an organization using the usersOrganizationName property. For more information on using this property, see the Configuring Windchill to Use an Enterprise Directory chapter of the *Windchill Installation and Configuration Guide - Windchill*.


Organizations are associated with the context in which they are created. From the Principal Administrator, you only have access to the organizations created in the current context or in ancestor contexts.

Each Windchill organization object holds the organization name, the UFID associated with the organization, the Windchill domain of the organization, and administrative flags that are set if the object needs to be repaired or is disabled. The UFID also contains the distinguished name of the organization and identifies the directory service where the organization entry resides.

## Using the Principal Administrator

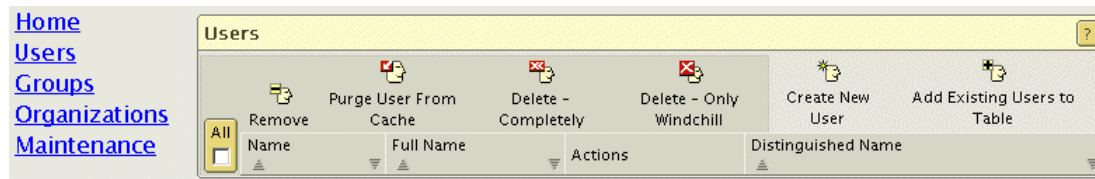
How you access the Principal Administrator is determined by your Windchill solution:

- From Windchill Foundation & PDM, you can access the Principal Administrator by clicking the **Principal Administrator** link that is on the **Business Administration** home page. This link launches the Principal Administrator in the Windchill PDM context and provides you with access to principals that are in the current context or in ancestor contexts.







- From Windchill PDMLink and Windchill ProjectLink, you can access the Principal Administrator from the **Utilities** pages that are under the **Site** and **Organization** tabs:
  - The Principal Administrator link from the **Site** tab provides you (as the system administrator) with access to users and to the groups and organizations created in the Site context.
  - The link from the **Organization** tab provides access to only those principals that are available through the organization context that is active when you launch the Principal Administrator and through the Site context (which is its ancestor context). If the organization was set up so that it allows entire user and group directory selection, then you can see all users and groups (except for the internal groups maintained by your solution). However, access control rules may be set to prohibit you from seeing some users and groups.
- From the Policy Administrator **Administrative Domain** window, you can access the Principal Administrator by clicking . Using this icon allows you to open the Principal Administrator from the current context of the Policy Administrator.

To get to the **Administrative Domain** window, launch Policy Administrator as described in [Using the Policy Administrator](#). Then select a domain and click **Update** or **View**.

The Principal Administrator allows administrators to manage Windchill principal objects using the following links on the **Principal Administrator** main page:



The following table describes the links:

Link	Description
Home	Re-displays the designated home page of the Windchill solution from which you opened the Principal Administrator.
Users	Displays the <b>Users</b> table from which you can manage users. Click <b>Add Existing Users to Table</b>  to add existing users to the table. Click <b>Create New User</b>  to create a new user. For additional information, see <a href="#">Managing Users</a> .
Groups	Displays the <b>Groups</b> table from which you can manage groups that are created in either the current context or ancestor contexts. Click <b>Add Existing Groups to Table</b>  to add existing groups to the table. Click <b>Create New Group</b>  to create a new group. For additional information, see <a href="#">Managing Groups</a> .
Organizations	Displays the <b>Organizations</b> table from which you can manage organizations. Click <b>Add Existing Organizations to Table</b>  to add existing organizations to the table. Click <b>Create New Organization</b>  to create a new organization object. For additional information, see <a href="#">Managing Organizations</a> .
Maintenance	Displays the <b>Disconnected Principals</b> table, which contains principals for which the distinguished name currently associated with the principal is not valid. From this table, you can search for additional principals that have nonexistent distinguished names, update disconnected principals, delete disconnected principals, or purge all principals from cache. For additional information, see <a href="#">Maintaining the Connections between Principal Objects and their Directory Service Entries</a> .

## Best Practices for Windchill PDMLink and Windchill ProjectLink

The Principal Administrator creates only organization objects and not organization contexts. To create both the organization object and its context, use **Create Organization** from either the **Site** or **Organization** tab.

Ensure that users have an e-mail address as many features in Windchill PDMLink and Windchill ProjectLink require users to have an e-mail address. If users do not have the e-mail attribute set in their user directory service entry, they cannot participate in the features that require an e-mail address.

## Best Practices When Maintaining a Directory Service Outside of Windchill

Because your user directory service can be maintained outside of your Windchill solution, you may not be creating users through the Principal Administrator; instead, users are automatically created in the Windchill database when the users become active in the solution. As users are removed or changed in the user directory service through an external tool, you will need to manage the Windchill user objects by doing the following:

- Deleting Windchill user objects that no longer have valid user directory service entries (see [Maintaining the Connections between Principal Objects and their Directory Service Entries](#)).
- Cleaning up after deleted users (see [Deleting Users](#)).
- Managing the principal cache so that changes in a user directory service are available in Windchill (see [Managing the Principal Cache](#)).

## Understanding Principals

Principals are identified across Windchill systems through the use of the principal Unique Federation Identifiers (UFIDs). The syntax for principal UFIDs is as follows:

*<distinguished\_name>:<guid>@<domain>*

where:

*<distinguished\_name>* is the distinguished name of the principal.

*<guid>* is the globally unique identifier of the repository in which the principal was first created or discovered.

*<domain>* is the Internet-style domain name of the repository in which the principal currently resides.

Together, *<guid>@<domain>* identifies the directory service in which the principal resides.

## Best Practices for Assigning Domains to Principals

When principal objects are created, they are associated with default domains as follows:

- Users who are affiliated with an organization (the organization attribute on their directory service entries is set) are associated with the domain that was created for the organization when the organization object was created. This domain usually has the same name as the organization (or a shortened version of the organization name) and the domain is a child of the Site container User domain. If the organization is associated with an organization container, then the domain is in the organization container; otherwise, the domain is in the Site container.
- Users who are not affiliated with an organization (the organization attribute on their directory service entries is not set) are associated with the Site container Unaffiliated domain. This domain is a child of the User domain. One exception to this rule is the Administrator user, which is associated with Site container System domain.
- User personal cabinets are associated with the same domain as the user. The one exception to this rule is that the personal cabinet for the Administrator user is associated with the User domain. in the Site context.
- Groups created in the Site context are associated with the Site container Unaffiliated domain. This domain is a child of the User domain.
- Groups created in an organization context are associated with the domain that was created for the organization when the organization object was created. This domain usually has the same name as the organization (or a shortened version of the organization name) and the domain is a child of the Site container User domain. The domain is in the organization context.
- Organizations are associated with the domain that was created for the organization when the organization object was created. This domain usually has the same name as the organization (or a shortened version of the organization name) and the domain is a child of the Site container User domain. If the organization is associated with an organization container, then the domain is in the organization container; otherwise, the domain is in the Site container.

**Note:** The default domain associations described in the previous list only apply if you have not set up JNDI adapters that configure principal domain assignments as described in previous releases. The domain assignments in a JNDI adapter take precedence over the system defaults.

User objects and personal cabinets are created automatically when users are selected in a search or when users log in. These user objects and personal cabinets are always associated with the default domains described earlier. When you create a user through the Principal Administrator, you can select the domains. But in

most cases, you should use the defaults (which are used when you do not select a domain).


When creating group objects through the Principal Administrator, using the default domain is usually a good choice. However, you may want to choose a different domain if you want policy rules from a domain other than the default to apply to the group object.


When creating organization objects through the Principal Administrator, using the default domain is usually a good choice. However, you may want to choose a different domain if you want policy rules from a domain other than the default to apply to the organization object. This may be the case if you want to allow users to select the owning organization when they create parts and documents. For the details on how to turn on the organization ownership feature, see *Administering Organizations* in the *Windchill System Administrator's Guide*.

## Searching for Users and Groups

By default, you are restricted to seeing only users that belong to the organization. If, when creating or updating an organization, you select the **Allow entire user and group directory selection** check box, you provide the ability to search for all users. For groups, this setting allows you to see group entries contained in the current LDAP directory hierarchy level of the Aphelion directory and each enterprise directory setup through a JNDI adapter (search scope set to ONELEVEL). Windchill PDMLink recommends you select this check box.

## Managing Users

Clicking the **Users** link on the **Principal Administrator** main page displays the **Users** table from which you can manage users. Clicking **Add Existing Users to Table**  allows you to locate existing users and add them to the **Users** table.

Clicking **Create New User**  allows you to create a new user. There are some restrictions for the user name you enter for a new user:

- User names cannot include the slash (/) character. This is because the slash is used as the folder separator in Windchill.
- Because of known problems with the Apache and Sun ONE Web servers, user login names cannot contain extended ASCII characters nor multi-byte characters.

**Note:** Although Windchill Foundation & PDM and the Principal Administrator do not require that users have an e-mail address, many features in Windchill PDMLink and Windchill ProjectLink require that users have an e-mail address.

After a user is added to the table, you can manage the user. In previous releases, users were identified by a user ID. The user ID is now known as the user name.

Managing users includes performing the following activities:

- Creating users, either from scratch or by starting with a similar user  
To create a similar user, access the information page of a current user and select the **Create Similar User** link.
- Searching for users
- Updating and deleting users  
When deleting users, you can delete them from just the Windchill database or delete them from both the database and the user directory service.
- Viewing information about users
- Defining electronic signatures for users.  
For additional information on using electronic signatures, see the [Electronic Signatures](#) section in [Administering Workflow Processes](#).
- Managing personal cabinet names  
From Windchill Foundation & PDM, you can administer personal cabinets from the Windchill Explorer. From Windchill PDMLink and Windchill ProjectLink, you can administer personal cabinets from the **Personal Cabinets Administration** link on the Site **Utilities** page.
- Purging users from the principal cache
- Refreshing the teams' membership using the **Recompute Team Membership** option.

For specific instructions on how to perform these activities, click **Help** from within the Principal Administrator.

The following sections provide information about personal cabinets and deleting users.

## Changing User Passwords

When you change the password for a user, the old password may still be active for a period of time. This is the case if the Web server you are using caches user information, as is true with the Apache Web server. If a user password is changed in LDAP, there may be a short period of time where the new password will not work, or where the old password and the new password will both work due to the cache. This is only temporary and will resolve itself when the Web server cache expires and the Web server refreshes the LDAP information. Generally, the time to live for cache entries is short. For example, the default for Apache is 10 minutes.



## Naming a User's Personal Cabinet

Since a Windchill user name does not need to be unique and all personal cabinet names must be unique, Windchill uses the `wt.folder.personalCabinetNamingAttribute` property in the `wt.properties` file to determine what the initial personal cabinet name should be for a given user. The `wt.folder.personalCabinetNamingAttribute` property contains the following default ordered list of attributes:

`name` -- The cabinet name used is the user's name.

`eMail` --The cabinet name used is the user's e-mail address.

`fullName` --The cabinet name used is the user's full name.

For the cabinet name, Windchill uses the value of the first attribute in the list that produces a unique name. In most cases, the name of the personal cabinet is the user's name. If there is already a personal cabinet with that name, the user's e-mail address is used for the personal cabinet name. If the e-mail address is already being used as a cabinet name, then the full name is used. If the full name is already being used, the object identifier for the user (OID) is used as the cabinet name. The oid is a unique string that identifies each object in Windchill. If the oid is already in use, Windchill appends an underscore and an integer, starting with 1, to the object identifier (`<oid>_1`, `<oid>_2`, and so on) until a unique cabinet name is discovered.

You can change the attributes used in creating the personal cabinet name or the order of these attributes by modifying the list of attributes in the `wt.folder.personalCabinetNamingAttribute` property using the `xconfmanager` utility. Valid values are those attributes used in user directory service entries. For example, to use the full name before using an e-mail address, you could specify the following `xconfmanager` command from a windchill shell:

```
xconfmanager -s wt.folder.personalCabinetNamingAttribute=name,fullName,eMail,oid  
-t <Windchill>/codebase/wt.properties -p
```

Where `<Windchill>` is the location where your Windchill solution is installed.

To use a user's telephone number instead of the e-mail address, you could specify the following property and value pair:

```
wt.folder.personalCabinetNamingAttribute=name,fullName,  
telephoneNumber,oid
```

If the attribute list for `wt.folder.personalCabinetNamingAttribute` has been modified and no personal cabinet name is discovered using the modified list, then Windchill derives the cabinet name from the user's name with the OID appended (as discussed earlier in this section).

For additional information about using the `xconfmanager` utility, see [About the xconfmanager Utility](#).

## Deleting Users

**Caution:** Do not delete a user unless you understand how it affects the system, as described in this section.

There are two actions that result in deleting a user. They are:

- Delete Only from Windchill
- Delete from Windchill and Directory Service

The first action has the effect of deleting the user from the Windchill database. The second action deletes the user from both the Windchill database and the user directory service. To use the second action, you must have the required permissions to be able to delete users from the directory service as well as the database.

**Note:** You cannot delete the Administrator or the Administrator group. Nor can you delete yourself.

The results of deleting a user from the Windchill database are as follows:

- The user is removed from all groups.
- All access control rules that identify the user are removed from the access control policy for a domain.
- The user is removed from all notification lists within notification policy rules and, if deleting the user from the list results in an empty list, then the rule is also deleted.

The following rules govern work items associated with a workflow process when a user is deleted from the Windchill database:

- If a user is deleted after a workflow process has been initiated, but prior to assignment of a work item, the user is removed from the list of participants and no work item is assigned.
- If the user is deleted after a workflow process has been initiated, and a work item has been assigned, that work item must be manually removed. (For more information, see [Administering Workflow Processes](#).)
- A deleted user continues to appear in iteration history, object properties pages, and so on, but the name is not displayed as an e-mail link.
- When a user is deleted, the user is automatically removed from the list of participants in any workflow process template. The user is also removed from any role mappings created as part of a life cycle or team definition.
- If a user is identified as a participant in a workflow template definition and that user is deleted from the system after the workflow has been initiated, any work item that would have been assigned to the user is reassigned to the workflow administrator who created the template.

- If both the template creator and a user identified in a workflow process template are deleted after the workflow process is initiated, the workflow process stops until the work items assigned to the deleted user are manually reassigned.
- Another user can be created with the same user name, but if the original user's personal cabinet was not deleted, the new personal cabinet will not have same name. For more information, see [Naming a User's Personal Cabinet](#).
- If a deleted user is specified as the user of a collection defined in the index properties, a stack trace prints in the Method Server log when an attempt is made to index an object.

The results of deleting a user from both the Windchill database and the directory service include all results described earlier for deleting a user from the Windchill database and additionally include the following:

- A user is not authenticated when attempting to log into Windchill.
- The user's name is not included in search results.

If a user is not removed from the user directory service, a new user object is created in Windchill database when the user tries to log on or when the user is selected from a search. This new user object is not the same object that was deleted, and all of the results of the earlier deletion are still true. For example, the user is no longer a member of the groups to which the user had been a member.

After deleting a user from the Windchill database, you must perform the following clean-up steps:

- Reassign any items in the user's worklist.
- Unlock any objects the user has checked out of the Windchill database.
- Remove the user's personal cabinet and any folders or objects within it. From Windchill Foundation & PDM, select the cabinet from the Windchill Explorer and delete it. From Windchill PDMLink, use the **Personal Cabinets Administration** link from the **Site Utilities** page.

## Changing the Organization to which a User Belongs

By using the Principal Administrator (if you have write permission to the directory service) or by using a 3rd party tool that allows your site to modify user entries in your directory service, you can change the organization attribute for users. Changing the organization attribute by using the Principal Administrator automatically changes the other organization-related information for users. However, changing this attribute by using a 3rd party tool does not change the other organization-related information for users.

After using a 3rd party tool to change a user's organization, you must also change the organization-related information for the user using the OrganizationSync command-line utility. For details on using this utility, see [Using the OrganizationSync Utility for User Organization Changes](#).

## Refreshing Team Membership for Users and Groups

Users and groups can be members of teams. Team membership can be refreshed when creating or editing users and groups, or at a later time by using a schedule. See the [Refreshing Groups](#) section to schedule the team refresh at a later point in time.

Refresh the team membership for users from the **Groups** tab by selecting the **Recompute Team Membership** check box. This option is only available for users when creating a new user, or when creating a similar user.

Refresh the team membership for groups from the **Members** tab by selecting the **Recompute Team Membership** check box. This option is only available when editing a group, or when creating a similar group.

**Note:** Be aware that depending on the number of teams the user or group is part of, and the size of the affected teams, recomputing team memberships now can add considerable processing time during peak usage periods. See the [Refreshing Groups](#) section in the [Administering Teams and Roles](#) chapter.

## Managing Groups

Clicking the **Groups** link on the **Principal Administrator** main page displays the **Groups** table from which you can manage groups. Clicking **Add Existing**

**Groups to Table**  allows you to locate existing groups and add them to the

**Groups** table. Clicking **Create New Group**  allows you to create a new group.

Managing groups from the **Principal Administrator** includes performing the following activities:

- Creating groups, either from scratch or by starting with a similar group

To create a similar group, access the information page of a current group and select the **Create Similar Group** link.

- Searching for groups
- Updating and deleting groups

When deleting groups, you can delete them from just the Windchill database or delete them from both the database and the directory service.

- Viewing information about groups

- Purging groups from the principal cache
- Refreshing the teams' membership using the **Recompute Team Membership** option.

From within an organization, you can also use the **Groups** link to add or update groups.

**Note:** The groups that can be managed from the Principal Administrator do not include internal groups that are created as a result of administrator interaction with Windchill PDMLink and Windchill ProjectLink. For example, the internal groups created for the context team roles can only be managed from the **Teams** link; they are not visible through the Principal Administrator.

For specific instructions on how to perform these activities, click **Help** from within the Principal Administrator.

The following sections provide additional information about working with groups and deleting groups.

## Working with Groups that are Maintained in a Directory Service

Any groups created in a node of a directory service that can be searched by your Windchill solution are automatically added to the Windchill database when the node is searched. The name of each Windchill group object is taken from the cn attribute of the LDAP group entry distinguished name (unless the mapping done in the JNDI adapter specifies a different attribute). This allows you to create groups using the editing tool available through a directory service rather than using the interfaces available in Windchill.

Adding and removing members from a group can also be accomplished using the editing tool available through a directory service. After updating group membership in the directory service, you can refresh the Windchill group membership as described in [Best Practices for Windchill PDMLink and Windchill ProjectLink](#).

When access to Windchill has been limited, you can rename a group using the editing tool available through a directory service while maintaining the association of the directory service group with an existing Windchill group; however, the name of the group in the Windchill database cannot be changed to match the name stored in the directory service unless the user has write permission to both the database and the directory service. PTC recommends that you do not rename groups outside of the Principal Administrator.

Use the following steps to re-establish the connection between an existing Windchill group to an LDAP entry for a group that has been renamed in your directory service:

1. Limit access to Windchill before renaming the group.

**Caution:** If you do not limit access and someone searches and finds the renamed group or updates the user information for any member of the group, then a new Windchill group object is automatically created for the group. After a new Windchill group is associated with the LDAP entry, you cannot reconnect the renamed group to another Windchill group.

2. Rename the group using the editing tool available through a directory service.
3. From the Principal Administrator, click **Maintenance** to display the **Disconnected Principals** table.
4. In the row corresponding to the group that no longer has a related directory service entry, select the **Update** action.

The **Update Principal** window opens from which you can re-establish the connection to an existing distinguished name for the group. The name of the group is updated to match the name in directory service entry.

## Deleting Groups

**Caution:** Do not delete a group unless you understand how it affects the system, as described in this section.

There are two actions that result in deleting a group. They are:

- Delete Only from Windchill
- Delete from Windchill and Directory Service

The first action has the effect of deleting the group from the Windchill database. The second action deletes the group from both the Windchill database and the directory service. To use the second action, you must have the required permissions to be able to delete groups from the directory service as well as the database.

The results of deleting a group from the Windchill database are as follows:

- Users who were members of the group no longer belong to the group.
- All access control rules that identify this group are removed from the access control policy for a domain. If any users had access permissions derived solely from membership in the deleted group, it may be necessary to create new rules to restore the lost permissions.

- The group is removed from all notification lists within notification policy rules and, if deleting the group from the list results in an empty list, then the rule is also deleted.

The following rules govern work items associated with a workflow process when a group is deleted from the Windchill database:

- If a group is deleted after a workflow process has been initiated, but prior to assignment of a work item, the group is removed from the list of participants.

If removing the group leaves no participants for a role, then the role resolution is determined by the settings in the wt.properties file:

- If the wt.workflow.engine.ignoreUnresolvedRole property is set to true and if the ignoreUnresolvedRole event configuration is set for this activity; then there will be no work item created and the WfAssignment object completes so the workflow does not hang.
- If the wt.workflow.engine.ignoreUnresolvedRole property is set to false, one work item is created that goes to the Responsible Role defined in the activity template. The default for the Responsible Role is the process creator. When the workflow process is started through a life cycle, the process creator is the creator of the business object.

For more information, see [Administering Workflow Processes](#).

- If the group is deleted after a workflow process has been initiated and work items are assigned, deleting the group has no effect on the process because the group itself is no longer being referenced. Work items are assigned to the individual users that were in the group.
- When a group is deleted, it is automatically removed from the list of participants in any workflow process template. The group is also removed from any role mappings created as part of a life cycle or team definition.
- If a group is identified as a participant in a workflow template definition, and that group is deleted from the system after the workflow has been initiated, any work item that would have been assigned to the group is reassigned to the workflow administrator who created the template.


The results of deleting a group from both the Windchill database and the directory service include all results described earlier for deleting a group from the Windchill database and additionally include the following:


- The group is not included in search results.

If a group is not removed from the directory service, a new group object is created in Windchill database when the group is selected from a search. This new group object is not the same object that was deleted and all of the results of the earlier deletion are still true. For example, the users who had been members of the group are no longer members.

## Managing Organizations

Clicking the **Organizations** link on the **Principal Administrator** main page displays the **Organizations** table from which you can manage organizations.

Clicking **Add Existing Organizations to Table**  allows you to search for existing organization objects and add them to the **Organizations** table. Clicking

**Create New Organization**  allows you to create a new organization object.

**Note:** To use an organization object (created using the Principal Administrator) as a Windchill PDMLink or Windchill ProjectLink organization context, you must create an organization container for the organization object. You can create an organization container from the **Organization** tab in either of these solutions. When creating an organization container, you can either use an existing organization object that is not associated with a container or create a new organization object.

Organization objects created using the Principal Administrator are considered restricted organizations. This means that no access control rules are automatically added to allow users in one organization to see users and groups from other organizations. When you create an organization container, you can select a check box that allows users in the organization to see all users and groups. This adds the organization to the Unrestricted Organizations group, which has the access control rules set to allow users to see other users and groups.

Organization objects can be used to identify an organization as the owner of specific parts and documents. By default, Windchill solutions are not set up to allow organization ownership selection. The organization under which the parts and documents are created automatically own them. There are multiple steps involved in enabling organization ownership selection, one of which is creating the organization objects. For details on how to enable organization ownership selection, see *Administering Organizations* in the *Windchill System Administrator's Guide*.

Managing organizations includes performing the following activities:

- Creating organizations
- Searching for organizations
- Updating and deleting organizations

When deleting organizations, you can delete them from just the Windchill database or delete them from both the database and the directory service.

- Viewing information about organizations
- Purging organizations from the principal cache

For specific instructions on how to perform these activities, click **Help** from within the Principal Administrator.



**Note:** When specifying the internet domain name of an organization, the name you enter can contain only alphanumeric characters and the hyphen (-) character. Do not enter any other types of characters in the name.

The following section provides additional information about deleting organizations.

## Deleting Organizations

**Caution:** Do not delete an organization object unless you understand how it affects the system, as described in this section.

**Note:** You cannot delete an organization object that is associated with an organization container.

There are two actions that result in deleting an organization. They are:

- Delete Only from Windchill
- Delete from Windchill and Directory Service

The first action has the effect of deleting the organization from the Windchill database. The second action deletes the organization from both the Windchill database and the directory service. To use the second action, you must have the required permissions to be able to delete organizations from the directory service as well as the database.

The results of deleting an organization from the Windchill database are as follows:

- The organization is removed from all notification lists.
- All access control rules that identify this organization are removed from the access control policy for a domain. If any users had access permissions derived solely from membership in the deleted organization, it may be necessary to create new rules to restore the lost permissions.

The results of deleting an organization from both the Windchill database and the directory service include all results described earlier for deleting an organization from the Windchill database and additionally include the following:

- The organization is no longer included in search results.

If an organization is not removed from the directory service, a new organization object is created in Windchill database when the organization is selected from a search. This new organization object is not the same object that was deleted and all of the results of the earlier deletion still hold.

## Receiving Administrative Notifications

When a user is deleted using the Principal Administrator, the administrator receives a notification by e-mail (or possibly through Windchill workflow). The administrator is notified that the principal has been disabled and that any additional manual actions should be taken, such as removing a user's personal cabinet.

Similarly, when Windchill detects that a user, group, or organization needs to be repaired because the object in the Windchill database no longer references an existing directory service entry, possibly because the entry has been removed from or relocated in the directory service, the administrator is notified that repair is needed.

These notifications are initiated by calling the Info\*Engine tasks named `NotifyPrincipalDisabled.xml` and `NotifyPrincipalRepair.xml` in the `<Windchill>/tasks/wt/federation` directory (where `<Windchill>` is the Windchill installation directory). You can customize these tasks to tailor the way in which notification is done.

Also, you can change the e-mail address used for the notifications by setting the `wt.org.principalAdministratorEmail` property in the `wt.properties` file. The default e-mail address used is the e-mail address of the Administrator user (if one is set) or the `postmaster@<server_hostname>` e-mail address, where `<server_hostname>` is the value of the `java.rmi.server.hostname` property in the `wt.properties` file.

## Managing the Principal Cache

To improve the access time required for users, groups, and organizations, Windchill maintains an internal principal cache of user, group, and organization information that has been obtained from the Windchill database and directory services.

**Note:** If user, group, or organization attributes are changed using an administration tool other than the Principal Administrator (for example, using a directory administration tool), then the cache containing those attributes must be purged in order for Windchill to display the changed attributes.

You can manage the principal cache in two ways:

- By setting the maximum time that the information about a principal can remain available in the cache. Then cache entries are automatically purged when users try to access the old entries.
- By selecting actions that purge information from the cache from within the Principal Administrator.

## Automatically Purging Entries from the Principal Cache

You can add the following properties to the wt.properties file to automatically purge principal cache entries:

- The wt.principal.cache.timeToLive property defines the amount of time that any given principal cache entry is available from the cache. Specify the property value in seconds. If the property value is not set or is set to zero or less than zero, then cache entries are not automatically removed from the cache. Out of the box, this property is not set.
- The wt.principal.cache.timeToLiveRandomizer property adds a random amount of time to the time stamp of each cache entry so that a large number of entries do not expire at the same time. Specify the property value in seconds. If the property is not set, the value defaults to 600 seconds (10 minutes). If the property is set to zero or set to less than zero, then no random value is added to the time stamp of each cache entry. Out of the box, this property is not set.

When a valid value is specified, the random amount added to the time stamp varies between 1 second and the property value. For example, if the property value is 600 seconds (10 minutes), a value between 1 second and 600 seconds is added to the time stamp of a cache entry when the entry is added to the cache.

Use the xconfmanager utility to add the properties. For example to add the wt.principal.cache.timeToLive property, specify the following xconfmanager command from a windchill shell:

```
xconfmanager -s wt.principal.cache.timeToLive=600  
-t <Windchill>/codebase/wt.properties -p
```

Where <Windchill> is the location where Windchill is installed.

For information about using the xconfmanager utility, see [About the xconfmanager Utility](#).

## Manually Purging Entries from the Principal Cache

From within the Principal Administrator, you can purge either the entire cache or individual entries from the cache. For specific instructions on how to perform these activities, click **Help** from within the Principal Administrator.

## Maintaining the Connections between Principal Objects and their Directory Service Entries

Sometimes the definitions of principals in the directory service are changed using a directory administration tool other than the Principal Administrator. For example, another tool is sometimes used to move users, groups, or organizations from one directory location to another. When this happens, the distinguished names for the principals that are moved change. During normal operation, Windchill keeps track of the objects it encounters that do not have valid distinguished names.

Clicking the **Maintenance** link on the **Principal Administrator** main page displays the **Disconnected Principals** table, which contains principals for which the distinguished name currently associated with the principal is not valid.

From the **Disconnected Principals** table, you can do the following activities:

- Search for additional principals that have nonexistent distinguished names
- Update disconnected principals
- Delete disconnected principals
- Purge all principals from the cache

For specific instructions on how to perform these activities, click **Help** from within the Principal Administrator.

# 10

## Administering Access Control

Topic	Page
Overview .....	10-2
About Access Control Rules .....	10-2
Creating and Managing Access Control Rules.....	10-4
Setting Permissions .....	10-4
About Access Control Lists.....	10-6
About Predefined Access Control Policy Rules.....	10-12
Managing Access to Enterprise Information.....	10-13
Access Control Strategies for Cabinets in Windchill Foundation & PDM.....	10-23
Access Control Strategies for Life-Cycle Managed Objects.....	10-25
Combining Access Control Strategies for Cabinets and Life-Cycle Managed Objects.....	10-25

## Overview

As a Windchill administrator, you must ensure that only the appropriate principals have access to objects. These decisions are expressed as access control rules governing domains. Defining these rules determines the types of interactions principals can have with objects of a specific object type and a specific life cycle state.

For example, you can create a rule that gives the Publication group permission to modify objects of type WTDocument when they are in the Under Review state of their life cycle. Together, such rules form an access control policy for the domain. Subsequently, access control lists (ACLs) are derived from the policy for a domain and the policies of all its ancestor domains to enforce your access decisions.

When users are viewing the attributes of an object where some of the attributes reference access controlled objects, such as principals, then whether the user sees the value of the attributes is determined by whether the user has Read permission for the referenced objects. Typically, when a user does not have Read permission for a referenced object, the field shows **(Secured information)** instead of the attribute value. For example, assume that a user displays information about a product. On the page displayed, one of the product attribute fields is **Created By** and the value is the name of the user who created the product. If the user displaying the product information does not have Read permission for the user who created the product, then the name of the user will not appear. Instead of the name, the user sees **(Secured information)**.

This chapter discusses Windchill access control concepts, explains the relationship between domain and instance-based access control rules, and presents strategies for developing useful rules.

## About Access Control Rules

One common administration task is specifying policy rules for controlling access to objects governed by a domain. When you create these rules, you customize the domain's access control policy. From this policy, ACLs are generated and associated with an object type. The ACL is the basic mechanism for enforcing access control decisions when a user attempts to interact with an object. ACLs are created upon demand and are cached to maximize system performance.

There are two types of ACLs:

- *policy ACLs*, which apply to an object type
- *ad hoc ACLs*, which apply to a specific instance of an object type.

For more information on ad hoc ACLs, see [Rules Governing Domain-based ACLs and Ad Hoc ACLs](#) in this chapter.

An *access control rule* for a domain is a mapping between an object type and life cycle state, and sets of principals and their associated permissions. For an object type and a specific state, an access control rule specifies rights of principals concerning access to objects of that type, in that state. For example, an access control rule might state that everyone in the Publications group has permission to read all objects of type WTDocument in the Engineering domain when they are in the Under Review state.

An *object type* specifies a category of objects that share the same attributes and functions. For example, WTDocument is an object type, and instances of that type may be found in some of the domains you have created. Since Windchill domains are hierarchical, access control rules defined for a domain are inherited by descendent domains. For example, access control rules defined for the WTDocument object type in all states within the Design domain apply to instances of the type within that domain or any descendent domains. Because Windchill types are also hierarchical, an object inherits rules defined for its ancestor types. Therefore, more than one rule may apply to a given object. For example, a rule that applies to the type WTPart also applies to the type WTSerialNumberedPart. Additionally, there can be access control rules specific to WTSerialNumberedPart.

**Note:** Not all business objects are subject to access control, nor must all object types exist in a domain.

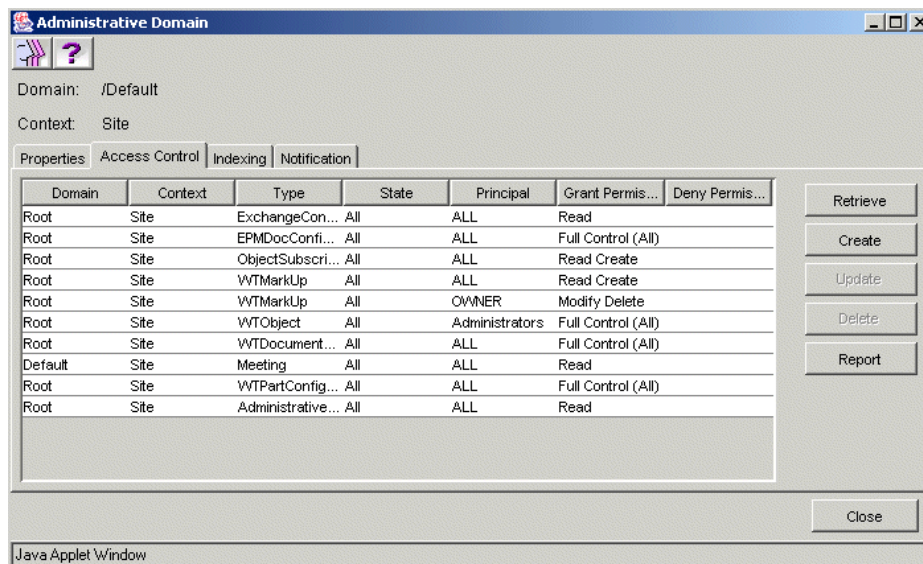
A *principal* is either an individual user, a group, or an organization. Most often, you define access control rules for groups or organizations. Dealing with groups or organizations helps reduce administrative overhead by enabling you to apply rules to more than one user at a time. Sometimes, however, you need to create rules for a specific user. For example, an access control rule can explicitly deny one group member a permission that is granted to the entire group by another rule.

*Permissions* represent operations that apply to an object. Permissions are described in more detail in the next section.

## Creating and Managing Access Control Rules

You can create and manage access control rules by opening the Policy Administrator as described in [Using the Policy Administrator](#). Select a domain and click **Update**. The **Administrative Domain** window opens.

On the **Administrative Domain** window, click the **Access Control** tab to bring it to the front of the window. To create an access control rule, click **Create**. To update a rule, select a row and click **Update**. Click **Help** to display detailed instructions.



## Setting Permissions

Through access control rules, you can establish whether a specific user, group, or organization is granted or denied permissions to the objects of a specified object type in a specified life cycle state. You can grant or deny the following types of permissions on the **Access Control Rule** window:

Permission	Description
Read	Determines the right to know the existence of the object and view it.
Modify	Determines the right to change the attributes of an object, as well as other characteristics that are part of the object definition. (For example, you have the right to modify the description attribute for a group, as well as remove a user from that group.)
Create	Determines the right to create an object.



Permission	Description
Revise	Determines the right to revise an object. Revising creates a new version of the object at the same level as the original in the version tree. For example, you can create Revision B from Revision A.
New View Version	Determines the right to create a version for a specific view.
Delete	Determines the right to delete an object.
Change Permissions	Determines the right to change the ad hoc permissions that others have.  Users, groups, or organizations granted the Change Permissions permission are allowed to change the ad hoc permissions of others to the permissions they themselves have or to a subset of the permissions they have.
Administrative	Determines the right to perform certain administrative tasks. (For example, this gives you the right to break a lock or change an object's owner.)
Full Control (All)	Determines full control.  A user, group, or organization with the Full Control (All) permission has all rights currently defined and any defined in the future. Therefore, when new permission types are defined, you do not have to write rules that specifically grant them to principals with full control.

Selecting certain permissions on the **Access Control Rule** window automatically selects other permissions when granting access to an object type. For example, if a group is given permission to create an object, the group typically should also be able to read and modify the object; however, these automatic selections can be deselected. The following table illustrates which permissions are selected automatically for each permission granted.

Permission	Selects
Read	None
Modify	Read
Create	Modify and Read
Revise	Modify and Read

Permission	Selects
New View Version	Modify and Read
Delete	Modify and Read
Change Permissions	None
Administrative	None

For example, if you select **Grant** for **Modify**, the **Grant** for **Read** button is automatically selected, as illustrated in the following part of the **Access Control Rule** window:

Permission	Grant	Deny	None
Full Control (All)	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Read	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Modify	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Create	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Revise	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
New View Version	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

If you do not want to permit **Read** access, simply click the **None** button for **Read** to clear it. Selecting **None** means that the rule neither grants nor denies the permission.

## About Access Control Lists

The Windchill access control list (ACL) mechanism, and the rules for evaluating ACLs, follow those in the `java.security.acl` package. This section provides a brief description of the way in which ACLs are derived from the access control policies for domains, and describes how they work to enforce access control.

When you create an access control rule, you specify the rule antecedent and the rule consequent:

- The rule antecedent comprises three parts:
  - The domain.
  - The object type, determines which rules within an access policy apply to a specific object.
  - The life cycle state, which identifies the life cycle phase that an object must be in for the permissions to apply. If the object type is not a life cycle managed type, then the state is not applicable and only rules with life cycle state set to ALL apply.

- The rule consequent comprises three parts:
  - A principal, which is a user, a group, or an organization. A user can be a member of more than one group.
  - Associated permissions.
  - Whether permissions are granted or denied.

For access control policy rules, the principal can be the pseudo-user OWNER or the pseudo-group ALL:

- A rule defined for OWNER specifies permissions that apply to the owner of an ownable object.
- A rule defined for ALL specifies permissions that apply to all principals.

You can create access control rules for some or all of the object types within a domain. Together, these rules constitute the access control policy for the domain.

An ACL is created on demand for each rule antecedent. An ACL is derived from the policy for a domain and its ancestor domains, and is composed of multiple ACL entries. Each ACL entry contains a set of permissions associated with a principal. Each ACL entry is either positive (+) or negative (–). If the entry is positive, the permissions are granted to the associated principal. If negative, the permissions are denied.

To improve performance, when ACLs are calculated, they are cached so they can be quickly retrieved the next time a user requests access to a particular object.

## Deriving ACLs from Access Control Policies

ACLs are the mechanism used to enforce access control. This section describes how ACLs are derived from the access control policy for a domain. The next section describes how ACLs work.

An ACL is generated for each object type, state, and domain. A given object is associated with the ACL whose domain, type, and state match that of the object. For example, all WTDocument objects of a given life cycle state, within a given domain, are associated with the same ACL. In addition, this ACL is different from the ACL associated with WTPart objects that belong to the same domain.

An ACL for an object is obtained by combining all rules that apply to the object's type, state, and domain. To make this definition precise, it is necessary to describe how rules are combined, and when a rule applies to a type.

A rule is applicable to a given type when the object type referred to in the access control rule is the type itself or one of its ancestor types. For example, a rule that applies to the WTDocument type also applies to incident reports if IncidentReport is a soft type of WTDocument.

Rules that apply to a type are combined by merging rules that have the same sign (+ or –) and principal. The merge is performed by calculating the union of all permissions within the consequents.

For example, consider the combination of the following rules:

	Domain	Type	State	Principal		Permission
Rule 1:	/ (Site)	WTOBJECT	InWork	Analysts	+	Read
Rule 2:	/Parts (Windchill PDM)	WTOBJECT	InWork	Engineers	+	Read
Rule 3:	/Parts (Windchill PDM)	IncidentReport	InWork	Analysts	+	Modify

The combination of these rules produces the following ACL entries for incident reports in the InWork state in the /Parts domain in the Windchill PDM container:

	Type	State	Principal		Permission(s)
/Parts (Windchill PDM)	IncidentReport	InWork	Analysts	+	Read, Modify
/Parts (Windchill PDM)	IncidentReport	InWork	Engineers	+	Read

## How ACLs Work

Permissions for a principal defined in one access control rule may conflict with other permissions defined in other rules. For example, you could define an access control rule that gives everyone in the Team 1 group delete permission for all incident reports belonging to the /Acme domain when they are in the Under Review state. However, in another access control rule, you could explicitly deny user Audrey.Carmen that permission, despite her membership in Team 1. In such cases, the ACL mechanism calculates the net permissions for a principal.

As defined in the java.security.acl package, the net permissions for a principal are calculated based on the following rules:

- Each principal (user, group, or organization) can have at most one positive ACL entry and one negative ACL entry; that is, multiple positive or negative ACL entries are not allowed for any principal. Each entry specifies the set of permissions that are to be granted (if positive) or denied (if negative).
- If there is no ACL entry for a particular user, group, or organization, that principal has the null permission set. In effect, having a null permission set denies the principal access to the object.
- If there is a positive entry that gives a principal a permission and a negative entry that denies the principal the same permission, the permission is not granted.

- Permissions that are explicitly granted (+) to the pseudo-user OWNER override any permissions denied (–) to the user that is the owner of the ownable object through a negative entry for the individual user or for a group or organization to which the user belongs.
- Permissions that are explicitly granted (+) to the user that is the owner of an ownable object or to a group or organization to which the user belongs override any permissions denied (–) to the pseudo-user OWNER.
- Permissions that are explicitly granted (+) or denied (–) an individual user always override that user's group or organization permissions. For example, user ReneN is a member of Group 1. According to an access control rule for the Acme domain, all members of Group 1 have modify permission to incident reports in the Under Review life cycle state. However, if another access rule explicitly denies ReneN permission to modify incident reports, then, ReneN is denied the modify permission, despite membership in Group 1.
- For a given user, the net group positive permission set is the union of all the positive permissions of each group and organization to which the user belongs. This includes permissions granted for the pseudo-group ALL. For example, if user ReneN belongs to Group 1, Group 2, and Group 3, ReneN's positive group permission set includes all of the permissions granted to those groups.
- For a given user, the net group negative permission set is the union of all the negative permissions of each group and organization to which the user belongs. This includes permissions denied for the pseudo-group ALL. Similarly, the permissions denied to Group 1, Group 2, and Group 3, are also denied to user ReneN, a member.

When the permissions are calculated for the ACL, the difference between the positive and negative group permission sets for user ReneN is used to determine access rights. For example, as a member of Group 1, ReneN is granted read permission to all incident reports in the Under Review life cycle state in the /Acme domain. However, ReneN is also a member of Group 2, which is denied read access to incident reports in that domain. When calculated, ReneN's permission to read incident reports is set to null for the /Acme domain.

The following table provides some further examples of permission calculation. Assume you, the administrator, are creating an access control policy for several domains. One of the users you have identified, Ann, belongs to two groups, G1 and G2. If you assign permissions as shown in the table, Ann's resulting permissions are identified in the last column.

	<b>G1 Permissions</b>	<b>G2 Permissions</b>	<b>Union of G1 and G2</b>	<b>Individual Permissions</b>	<b>Resulting Permissions</b>
+	Modify (M)	Create (C)	(C)+(M)	Delete (D)	(C)+(M)+(D)
-	Null set	Null set	Null set	Null set	
+	(M)	(C)	(C)	(D)	(C), (D)
-	-(D)	-(M)	-(D)	Null set	
+	(M)	(D)	(M)	(C)	(C)
-	-(D)	-(C)	-(C)	-(M)	
+	(M)	(C)	(M)+(C)	(D)	(C)+(D)
-	Null set	Null set	Null set	-(M)	

When you have defined the access control rules for domains, all of the instances of the object type of a particular state, and belonging to the same domain for which you have created rules, share an ACL. This association between the ACL and the object type, state and domain is preserved thereafter. When a principal attempts to access an object (for example, to view it or modify it), the associated ACL is retrieved, and the policy is enforced. Once an ACL is calculated, it is cached so it can be retrieved quickly for the next access request.

For example, assume that within the /Acme domain, user Audrey.Carmen is a member of a group that has read, delete permission for all objects of the type WObject that are in the Closed state. She is also a member of a group that has modify permission for all incident reports within the /Acme/Support domain in the Closed state, where IncidentReport is a soft type of WObject. However, there is another access control rule within the /Acme domain for Audrey.Carmen as an individual user, that explicitly denies her delete permission for incident reports in the Closed state.

The following shows the ACL entry for Audrey.Carmen that is associated with incident reports in the Closed state within the /Acme/Support domain:

+Audrey.Carmen read, modify

When this ACL entry was derived from the access control policies for the /, /Acme, and /Acme/Support domains, Audrey.Carmen was given read and modify permissions. Because IncidentReport is a soft type of WObject, Audrey's read and delete permissions for objects of type WObject also apply to incident reports. However, because there is another access control rule that explicitly denies her delete permission for incident reports in the Closed state, she is not able to delete objects of that type/state combination belonging to the /Acme/Support domain.

## Rules Governing Domain-based ACLs and Ad Hoc ACLs

An access control rule for the domain applies to an object type. An ad hoc ACL applies to a specific instance of that object type. The ad hoc ACL, however, specifies only positive (+) permissions; it cannot be used to deny access to an object. If the ad hoc ACL grants a permission that is denied in the policy ACL, the ad hoc rule supersedes the policy rule, and the access right is granted.

## Distributed Administration of Policy Rules

Distributed administration is the administering of a Windchill solution by different groups of individuals. Each group has responsibility for a particular area of the solution, with enough privileges to fulfill their administrative responsibilities. Domains demark administrative areas in Windchill. Windchill supports distributed administration of access control, indexing, and notification policy rules. General information about setting up administrators can be found in [Establishing Administrators](#).

Access control, indexing, and notification rules are members of the domain to which the rule applies. For example, if you define an access control rule granting Read access to documents belonging to the Publications domain, then the rule itself belongs to the Publications domain. This allows policy rules to be administered by different groups of administrators.

To give a group of administrators the rights they need to manage policy rules for an area of the system, you need to define access control rules granting permissions to the group for the AccessPolicyRule, IndexPolicyRule, and NotificationRule object types, and the domain associated with their area of responsibility. A predefined access control rule for the / (Root) domain in the Site container, grants all permissions to the Administrators group for all objects, so members of this group can manage policy rules for all domains.

For example, consider the following rules:

	Domain	Type	State	Principal	Permission
Rule 1:	/ (Site)	AccessPolicyRule	All	MarketingAdministrators	+Read
Rule 2:	/ (Site)	IndexPolicyRule	All	MarketingAdministrators	+Read
Rule 3:	/ (Site)	NotificationRule	All	MarketingAdministrators	+Read
Rule 4:	Marketing (Windchill PDM)	AccessPolicyRule	All	MarketingAdministrators	+Full Control (All)
Rule 5:	Marketing (Windchill PDM)	IndexPolicyRule	All	MarketingAdministrators	+Full Control (All)
Rule 6:	Marketing (Windchill PDM)	NotificationRule	All	MarketingAdministrators	+Full Control (All)

These rules grant all permissions to the MarketingAdministrators group for the policy rule object types in the Marketing domain of the Windchill PDM library container. They allow members of the MarketingAdministrators group to view, create, update, and delete rules in the Marketing domain or any of its descendent domains, but not to manage rules in any ancestor domains. The rules granting read permissions to the MarketingAdministrators group for the policy rule object types in the Root domain allows members of the MarketingAdministrators group to see the rules inherited from ancestor domains.

## About Predefined Access Control Policy Rules

When a Windchill solution is installed, the set of access control policy rules that are described in the [Installed Site Container Policies](#) section of the Administering Containers chapter is created for the initial domains in the Site container. Similarly, additional access control rules are created when an organization container or an application container is created.

For the details on the organization rules, see [Container Access Control Policies](#).

For the details on the product and library rules, see the [Out-of-the-box Container Access Control Policies](#) section in the Administering Products and Libraries chapter.

**Caution:** The access control rules set for the domains in the Site container should not be modified without considering the full consequences of the modification. For example, changing the rule that grants Administrators Full Control (All) on the WObject object type in All states should not be modified. If this rule is removed by mistake, you cannot administer your Windchill solution.



**Caution:** Creating rules that deny permissions for the pseudo-group ALL is also discouraged. Denying access to ALL includes denying access to users in the Administrators group unless there is a rule granting access to an individual user that is in that group.

To repair the removal of the Administrators rule described above or to remove rules such as a rule that denies access to all principals, complete the following steps:

1. Using the xconfmanager from within the windchill shell, set the wt.access.enforce property in the wt.properties file to false:

```
xconfmanager -s wt.access.enforce=false  
-t <Windchill>/codebase/wt.properties -p
```

**Note:** Setting this property to false turns off access control. This means that none of the access control rules are enforced.

2. Restart Windchill so that the new property value is used.
3. Recreate the rule that was deleted using the Policy Administrator.
4. Set the wt.access.enforce property back to true and restart Windchill.

For additional information about using the xconfmanager utility, see [About the xconfmanager Utility](#) in the Administration Overview chapter.

## Managing Access to Enterprise Information

This section describes access control issues and strategies related to managing access to your enterprise information. The following characteristics of objects impact what access control rules need to be defined to manage access to your enterprise information and to define strategies for managing the information:

- Domain administered information -- includes any object that belongs to a domain.
- Policy and ad hoc access controlled information -- includes any object to which domain-based access control rules or ad hoc access control rules can be applied.
- Content holder information -- includes any object to which files and URLs can be attached.
- Foldered information -- includes any object that is contained within a folder. Foldered objects are manifested in the interface as cabinets and subfolders.
- Life-cycle managed information -- includes any object that is life-cycle managed. Each life-cycle managed object has associated life cycle state.
- CAD document information -- includes the objects that are used when creating and working with CAD documents. For additional information, see

the workgroup manager guide that describes how to administer CAD documents in your Windchill solution.

**Note:** A specific object type can have one or more of the characteristics and thus needs to have access control rules set in multiple ways. The rules and strategies you set up must take all characteristics into account.

The following sections describe each characteristic and identify the access control requirements necessary for operating on an object that has the characteristic.

## Domain Administered Information

Access control decisions for an object that is a member of a domain are based on the following criteria:

### Object Type

Determines which rules within an access policy apply to an object.

### Domain

The domain determines which access control policies apply to an object.

An object's domain and type determine which policy ACLs are associated with the object. The policy ACL, in turn, specifies which principals have permissions for objects that share the same domain and type. The set of permissions is described in [Setting Permissions](#).

## Required Rules for Domain Administered Information

The following rule governs the movement of objects among domains:

- Moving an object from one domain to another requires Delete permission for the object in the source domain and Create permission in the destination domain.

For example, to change the domain of a cabinet in Windchill Foundation & PDM, the user must have the rights to delete the cabinet in its current domain and create the cabinet in the target domain.

## Policy and Ad Hoc Access Controlled Information

Objects that are policy or ad hoc controlled are subject to access control. The permissions that can be set on these objects are described in [Setting Permissions](#). For example, to create an object of a specific type in a domain, a user must have Create permission for the object in that domain.

## Content Holder Information

A number of Windchill objects, including all document types and change objects (change requests, change orders, and change activities), are modeled as content holders. A content holder is an object to which files and URLs can be attached. For example, after a document is created and saved to the Windchill database, the user who created it can add a number of files and URLs to it, which are then uploaded to the database. When the document is later checked out, the user has the option to download one or more of the content files, which can be replaced with new or updated content when the document is checked in. Users can also request that read-only copies of one or more content objects be downloaded to the local file system. That is, users can access content files without checking the content holder out of the database.

The following are several access control implications for content holders:

- If a content holder object is not workable (that is, it is not eligible for check in and check out), Modify permission is required to update the object.
- If a content holder is workable (that is, it is eligible to be checked in and out), content can be added only to the working copy of the object, which is located in the user's Checked Out folder by default.
- There is no way to create separate access control rules for content associated with a content holder. Rules applied to the object govern access to its content as well.
- Through current Windchill solutions, the content of a document or change object is only visible when a user has access to the document or change object. Therefore, if you are able to check-out the object, you are able to modify content.
- Access control rules for the content holder are explicitly checked before any content is uploaded or downloaded. If you have Modify permission for the content holder, you can upload content, and if you have Read permission for the content holder, you can download content.
- Workable objects (those objects that must be checked out and checked in) need Modify permission to check in an object.

## Foldered Information

The Windchill conceptual model for information storage is based on the organization provided by an operating system. The components of this model include the following:

- *Cabinets* -- a type of folder that is the top-level organizing mechanism in the Windchill solution. A cabinet is analogous to a disk drive in the Windows operating system. Cabinets are exposed in Windchill Foundation & PDM, but not exposed in the Windchill PDMLink and Windchill ProjectLink user interface. However, Windchill PDMLink and Windchill ProjectLink do use cabinets internally.

- *Subfolders* -- a type of folder that holds objects and resides in cabinets or other subfolders.

The top-level foldered object exposed through the Windchill PDMLink and Windchill ProjectLink user interface is called a **Folder** and has the subfolder object type. Within a solution container, all subfolders are members of either the Default or System container cabinet.

- *Folder members* (also called *foldered objects*) -- objects that must be stored in a folder.

The following sections provide more information about cabinets, subfolders and access control rules related to foldered information.

## Cabinets

In addition to being a folder object, a cabinet is a domain administered object. When a cabinet is created, it is associated with a domain. The associated domain determines its policy rules and administrative policies. In Windchill Foundation & PDM, if no domain is selected when a cabinet is created through the Windchill Explorer, / (root) domain in the Site context is associated with the cabinet.

A cabinet can contain folder members, which include subfolders and shortcuts to other folder members. Cabinets cannot contain other cabinets.

In Windchill Foundation & PDM, the /wt/admin/displayDomains preference determines the visibility of the domain a cabinet or subfolder belongs to on properties pages, and on dialogs for creating and updating cabinets and subfolders within Windchill Explorer. If the value is true, the cabinet or subfolder's domain is displayed. For subfolders, inheritance of the domain from a parent cabinet or subfolder is also displayed. If the value is false, the domain information is not displayed.

A cabinet may have a primary owner. By default, the owner of a cabinet is also the owner of all information stored in that cabinet. In general, cabinets provide an organizational root for information.

To facilitate organization and control of information, the system provides the following two types of cabinets:

- *personal*: A personal cabinet is associated with a single user, who is considered its owner. In other words, you are the owner of your personal cabinet and all of the information it contains. In Windchill Foundation & PDM, access to personal cabinets is through the Windchill Explorer and the personal cabinet icon. In Windchill PDMLink and Windchill ProjectLink, access to personal cabinets is through a user's work list items; there is no direct access to personal cabinets.
- *shared*: A shared cabinet is not associated with a single user and does not have an owner. Like a common filing cabinet, a shared cabinet contains information intended to be shared among users and groups. A shared cabinet has no owner, and the information stored in that cabinet also has no owner.

The administrative rules determine who has access to the shared cabinet and its objects. The shared cabinets within the system can also be thought of as a vault for storing information. However, the access control rules applied to the domain associated with the cabinet determine the level of security the cabinet provides.

## Subfolders

Similar to that of an operating system, in which a root directory contains both subdirectories and files a subfolder is used to hold Foldered objects. Subfolders reside in cabinets or other subfolders.

## Foldered Objects

A folder member (or foldered object) must always reside in a folder, whether a cabinet or a subfolder. A folder member can be located in only one folder at a time, and its identity must be unique within that folder. However, you can use a shortcut to make a foldered object that resides in one folder also appear to be in another folder.

By default, the owner of a folder member is the owner of the folder in which it is located.

## Domain Inheritance for Foldered Objects in Windchill Foundation & PDM

The domain association for a foldered object (other than a cabinet) is inherited from its parent, unless a domain has been explicitly associated with the foldered object.

For example, assume that the following structure exists:

- A cabinet named Publications belongs to the /Pubs domain.
- A subfolder named Technical Documents (located in the Publications cabinet), belongs to the /TechDocs domain.
- A subfolder named Installation Guides (located in the Technical Documents subfolder), inherits its domain from its parent subfolder.

In this example, both subfolders are governed by the policies defined in the /TechDocs domain.

The following rules govern the movement of subfolders:

- If the subfolder inherits its domain, the subfolder and all its folder members that also inherit their domain are governed by the administrative policies in the domain associated with the subfolder's new parent, when it is moved. For example, if the Installation Guides subfolder was moved to another cabinet or subfolder, this subfolder and the folder members of the subfolder that inherit their domain from the subfolder would then be associated with the domain of the new parent.

- If the subfolder belongs explicitly to a domain, it continues to belong to that domain, when it is moved. For example, if the Technical Documents subfolder is moved to another cabinet or subfolder, its domain remains unchanged and both the Technical Documents and Installation Guides subfolders and the folder members of the subfolders that inherit their domain from the subfolders are still governed by the administrative policies of the TechDocs domain.
- When you change a subfolder to inherit its domain, the domain of the subfolder, and all its folder members that inherit their domain, are changed accordingly. For example, if you change the Technical Documents subfolder to inherit its domain, then both the Technical Documents and Installation Guides subfolders are associated with the Pubs domain.

If you want to ensure that a subfolder retains the same domain association as its current parent, even if it is moved, you must explicitly associate the subfolder with the domain of its current folder.

To change the domain of a subfolder, the user must have the right to delete subfolders in their source domain and to create subfolders in the target domain.

## Required Permissions for Foldered Object Activities

The following rules govern foldered objects:

- To change the contents of a folder (for example, to create or delete foldered objects or to move objects from one folder to another), the user must have Modify rights for the folders involved in the change. To navigate to an object that is to be deleted or moved, the user must also have the Read rights to the subfolders in the path to the object.
- If you are moving objects between the folders that are not in the same domain, the rules listed under [Required Rules for Domain Administered Information](#) also apply. If the folders are in the same domain, no additional permissions are needed.

If the rules above are met, a foldered object in Windchill Foundation & PDM can be moved as follows:

- From one shared cabinet to another
- From one personal cabinet to another
- From a personal cabinet to a shared cabinet

An object cannot be moved from a shared cabinet to a personal cabinet.

**Note:** In Windchill Foundation & PDM, parts must be created in the user's personal cabinet. They can then be moved to a shared cabinet, so others can access them.

## Default Access Control Rules for Foldered Objects

By default, the Administrators group has Full Control (All) rights to a cabinet and all of its folder members. This is the case because of the predefined access control rule created for the Root domain granting Full Control (All) permissions to members of this group for objects of type WObject and all of its soft types.

Whenever you create a new Windchill user object, a personal cabinet is created for the user. Since a Windchill user name does not need to be unique and all personal cabinet names must be unique, Windchill uses the `wt.folder.personalCabinetNamingAttribute` property in the `wt.properties` file to determine what the initial personal cabinet name should be for a given user. In most cases, the name of the personal cabinet is the user's name. For additional information about naming personal cabinets, see [Naming a User's Personal Cabinet](#).

By default, the personal cabinet is associated with a child domain of the /User domain. The name of the child domain is usually the name of the user's organization unless the user is not affiliated with a domain (see [Managing Users](#) for details). The user is the owner of the personal cabinet and any objects he or she creates within this personal cabinet or any of its subfolders. Therefore, the user has all rights to those objects. This is the case because of the predefined access control rule created for the User domain that is in the Site container. The rule grants Full Control (All) permissions to the OWNER of objects of type WObject and all of its soft types. However, the domain with which the user cabinet is associated can be changed, and appropriate rules granting owner rights would need to be defined for any other domains associated with personal cabinets.

## Life-Cycle Managed Information

A very important characteristic of an object is whether it is life cycle-managed. Objects that are life cycle managed are also domain administered. Therefore, the criteria for the objects includes the domain-administered criteria.

Access control decisions for life cycle-managed information are based on the following criteria:

### Team

Users can participate in a role for an object. The team stores the current team membership by role. The team is resolved using the team template, life cycle template and, for Windchill PDMLink and Windchill ProjectLink, the context team. For additional information on teams, see [Administering Teams and Roles](#).

### Life Cycle

Determines an object's initial life cycle state by associating the object with a life cycle template. Life cycle state influences both policy and ad hoc ACLs. The ad hoc ACL is computed by binding life cycle and team roles to principals. The ad hoc ACL is stored within the object.

## Life Cycle State

Indicates the phase of the life cycle, which was used to compute the ad hoc ACL. It is used during execution to determine the policy ACL that is appropriate for an object.

An object's domain, type, and life cycle state determine which policy ACL is associated with the object. The policy ACL, in turn, specifies which principals have which permissions on objects that share the same domain, type and life cycle state.

When an object is associated with a life cycle or workflow activity, access to that specific instance of the object type can be governed by an ad hoc ACL, in addition to the policy ACL associated with the object based on its domain, type, and state. The life cycle or workflow activity can include permissions for roles associated with each life cycle phase or workflow activity. For example, principals who fulfill a life cycle or workflow role by submitting, reviewing, or promoting the object to the next life cycle phase are given access rights. Ad hoc ACLs for a life cycle phase or workflow activity are in effect for the duration of that phase or activity.

## Required Permissions for Life-Cycle Managed Object Activities

In Windchill Foundation & PDM, Modify rights for an object is required to set its life cycle state. However, if the activity is done through the Windchill Explorer, the user must also have Administrative rights to the object.

In Windchill PDMLink, the ability to set the life cycle state of a life-cycle managed object is controlled by either of following conditions:

- The user has Administrative rights to the object.
- A Set State transition is defined for the current life cycle state of the object and the user has create privileges for the Promotion Request in the container where the object resides. For information on the Set State transition and the Promotion Request, see [Promotion Process](#) in the Administering Life Cycles chapter.

In both Windchill Foundation & PDM and Windchill PDMLink, Read rights to a life cycle template and team template is required in order to select them when creating a life-cycle managed object.

## Example of Using Life Cycle Roles

When an object is created, the user is asked to select a life cycle and a team for it. Therefore, life cycle roles can be resolved by mapping them to team roles, which are then mapped to actual users.

For example, assume the following:

- For the Under Review phase of the Development life cycle, the life cycle role Promoter is mapped to the team role Team Leader.



- In the Prototype team, Team Leader is mapped to Amanda Smith.

Then, user Pat Johnson chooses the Development life cycle and selects Prototype as the team when he creates a design document in his personal cabinet. Subsequently, Pat moves his design document to a shared cabinet.

Later, when the design document is promoted to the Under Review phase in its life cycle, Amanda Smith becomes the Promoter. Although the policy ACL does not grant Amanda modify rights to design documents, she does have that access permission for Pat's document as long as the document is in Under Review phase (that is, until she submits it for promotion to its next life cycle phase).

## **Example Permissions Needed for Moving a Document**

The following example illustrates the permissions that are needed for the move operation as a result of the characteristics of a document. Although the permissions can be granted by either policy or ad hoc access control rules, this example describes the use of policy rules.

Moving a document from one folder to another requires the permissions described in the list below. For example, consider moving an object of type WTDocument (which is a foldered object) from one folder (which is either the SubFolder or Cabinet object type) to another.

- Requires Read permission for the container that the document resides in because the document is contained.
- Requires Read permission for the document in the domain it belongs to (that is, the domain of the source folder) in order to select it for moving, because the document is access controlled.
- Requires Read permission for the document in the domain it belongs to after the move (that is, the domain of the destination folder) in order to view it once it has been moved, because the document is access controlled.
- Requires Modify permission on the source and destination folders because the document is foldered and the folder content is being changed (removing the document from the source folder and adding it to the destination folder).
- If the source and destination folders are in different domains, then the domain of the document will change when it is moved, since it is domain administered and inherits its domain from the folder it resides in. Changing the domain requires Delete permission for the document in the domain of the source folder and Create permission for the document in the domain of the destination folder.

## Windchill Foundation & PDM Example Permissions Needed for Creating a Part in a Shared Cabinet

The following Windchill Foundation & PDM example illustrates the permissions that are needed to create a part in a shared cabinet or folder. The part can be created in the user's personal cabinet, and then can be checked in or moved to a shared cabinet. The permissions can be granted by either policy or ad hoc access control rules.

Creating a part in the user's personal cabinet requires the following permissions:

- Requires Read permission for the container that the part is being created in, because the part is contained.
- Requires Create permission for the part in the domain it is being created in (that is, the domain of the personal cabinet), because the part is access controlled.
- Requires Modify permission on the personal cabinet, because the part is foldered and the folder content is being changed (adding the part to the personal cabinet).

In addition to these permissions, the user may also need permissions to other objects related to the part creation. For example, to select a view, a life cycle, or a team for the part, the user must have Read permission for the view, life cycle template, or team to be selected.

Moving the part from the user's personal cabinet to a shared cabinet requires the same permissions as moving a document from one folder to another. The permissions are as follows:

- Requires Read permission for the container that the part resides in because the part is contained.
- Requires Read permission for the part in the domain it belongs to (that is, the domain of the personal cabinet) in order to select it for moving, because the part is access controlled.
- Requires Read permission for the part in the domain it belongs to after the move (that is, the domain of the shared cabinet) in order to view it, because the part is access controlled.
- Requires Modify permission on the source and destination folders (personal and shared cabinets) because the part is foldered and the folder content is being changed (removing the part from the source folder and adding it to the destination folder).
- If the source and destination folders are in different domains, then the domain of the part will change when it is moved, since it is domain administered and inherits its domain from the folder it resides in. Changing the domain requires Delete permission for the part in the domain of the source folder (personal cabinet) and Create permission for the part in the domain of the destination

folder (shared cabinet). Typically personal cabinets will be in a different domain than shared cabinets.

If the part is checked in to the shared cabinet, Modify permission on the part in the personal cabinet is also needed. This is because attributes on the part are changed as a result of a checkin.

## Access Control Strategies for Cabinets in Windchill Foundation & PDM

This section describes some strategies for developing access control policies for cabinets in Windchill Foundation & PDM.

### Access to Cabinets

Access control rules that apply to the cabinet (based on the domain it belongs to) do not extend to the objects located within that cabinet and its folders. For example, assume that user Bill Smith has Read permission for Cabinet type in domain X. However, having Read permission to cabinets does not give Bill Smith read access to documents in a cabinet. There must be an additional access control rule defined for documents that provides the Read permission.

Also, consider the following example:

- User Bill Smith *does not* have Read permission for the Cabinet type in domain X.
- He *does* have Read permission for the Requirements type, a soft type of WTDocument, in domain X.
- Consequently, Bill Smith can search for and find Requirements documents that reside in domain X. However, because he does not have read access to the cabinet itself, he cannot see the cabinet or any of its contents through the Windchill Explorer.

As illustrated by this example, it is important that you create logical rules that provide users with the access they need.

The following sections describe several strategies for creating rules to effectively manage both cabinets and the objects within them. These strategies assume that foldered objects inherit their domain from the parent folder, rather than explicitly assigning a domain.

### Restrictive Rules for Cabinets

One way to manage access control for Windchill objects is to define rules granting limited access to the cabinet (applied to its ancestor type WObject), and then to add more rules that grant some principals additional permissions for specific foldered object types. In general, this strategy uses the potential of shared cabinets for use as an information storage vault.

For example, you may decide that all users within the system should be allowed to see the shared Development cabinet and its contents when they are navigating in Windchill Explorer. Also assume that only the Engineering and Design groups should be allowed to check out and modify Specification (which is a soft type of WTDocument) documents stored within that cabinet. The following rules (defined for the domain to which the shared Development cabinet belongs or for an ancestor of that domain), support this strategy:

Rule 1:

WTObject	+	ALL	Read
----------	---	-----	------

Rule 2:

Specification	+	Engineering, Design	Modify
---------------	---	---------------------	--------

All folder members must reside within a cabinet or a subfolder. In addition to permissions such as those above, the required rules described in [Foldered Information](#) apply.

## Open Rules for Cabinets

Another strategy for applying access control to objects is to define relatively open rules for access to the cabinet (applied to its ancestor type WTObject), and then to add more rules that deny access to certain object types.

For example, you may decide to grant all principals read, modify, create, and delete permissions to objects that belong to the domain associated with the shared Development cabinet. However, you may want to deny the Publications group the right to create, modify, or delete Specification documents, and deny the Marketing group the right to create, modify, or delete Requirements documents.

The following rules (defined for the domain to which the shared Development cabinet belongs or for an ancestor of that domain), would support this strategy:

Rule 1: WTObject + ALL: read, modify, create, delete

Rule 2: Specification – Publications: create, modify, delete

Rule 3: Requirements – Marketing: create, modify, delete

## Access Control Strategies for Life-Cycle Managed Objects

Consider starting with more restrictive access control rules and then using activity- or life cycle-based rules to open up access. Also consider placing rules that grant wide access to information or access to information in its final state in a policy ACL, as access can easily be extended or restricted. A policy rule can change access to many objects, while changing access control permissions in ad hoc ACLs requires action on each individual object instance.

As described earlier, you can establish complementary access control rules for domains and the objects that are associated with them. Similarly, you can implement an access control strategy by balancing the use of policy and ad hoc ACLs.

Conversely, you can create a domain policy that provides for more open access to cabinets and their contents. In this case, you would define few (if any), access control rules within the life cycle.

## Combining Access Control Strategies for Cabinets and Life-Cycle Managed Objects

In some cases, you may decide to create restrictive domain policies, which provide only the minimum access to most users. Specifically, you can grant users read permission to one or more shared cabinets, so they can view the cabinet while withholding additional permissions for objects residing in the cabinet and its folders. Then, based on the life cycle and team associations for the objects within each cabinet, you can use ad hoc ACLs to grant certain principals the access permissions they need to fulfill their roles for a life cycle phase or workflow activity.

Life cycle roles can be mapped to team template roles when a life cycle is created. For example, the life cycle role Promoter can be mapped to the team role Team Leader. When a team is defined, roles are mapped directly to specific principals or to actor roles (of which there is only the Creator actor role currently defined). In addition for Windchill PDMLink and Windchill ProjectLink, the context team roles and members are used. For additional information about teams, see [Administering Teams and Roles](#).

Additionally, life cycles can contain access control rules for specific phases and roles. For example, assume that the Development life cycle includes an Under Review phase. The access control rules for this phase specify that for the duration of the phase, the Promoter role has modify permission for the object.



# 11

## Using Types and the Type Manager

This chapter discusses the basic concepts of types and the runtime typing capability. It describes the Type Manager utility and how to use it to define new subtypes, attributes, and constraints. The chapter also has information about managing types and attributes for parts, documents, CAD documents, dynamic documents, and change objects.

<b>Topic</b>	<b>Page</b>
Overview of Types and the Runtime Typing Capability .....	11-2
The Type Manager .....	11-10
Specifying Windchill Types .....	11-24
Managing Types and Attributes for Parts, Documents, and Change Objects ..	11-25
Managing Types and Attributes for CAD Documents .....	11-29
Managing Types and Attributes for Dynamic Documents .....	11-31
Best Practices .....	11-34

## Overview of Types and the Runtime Typing Capability

Each Windchill out-of-the-box modeled class appears as a node in the Type Manager. These modeled classes define the out-of-the-box *types* that are available to you.

There are two ways to extend the set of out-of-the-box types:

- By modeling additional classes through Rational Rose, you can add subclasses to Windchill business objects.

After code is recompiled, any modeled classes that extend the basic Windchill business objects automatically appear as subtypes under the Type Manager node of types for the classes that were extended. The Type Manager supports inheritance; therefore the newly created subtype has both the modeled and non-modeled attributes of the parent type. For example, assume that your company creates a subclass in Rational Rose named `wt.doc.SiteDocument` which extends the Windchill base class `wt.doc.WTDocument`. Then `wt.doc.SiteDocument` appears as a subtype of `wt.doc.WTDocument` in the Type Manager and automatically has the attributes from `wt.doc.WTDocument`. From the Type Manager, you then can update the type to change the display name (for example, use `Site Document`), add an associated icon, add additional attributes, and further refine constraints, or you can create new subtypes of `wt.doc.SiteDocument` with its corresponding attributes and constraints.

For additional information on why you may want to use modeled classes for subtypes, see [When to Use Typing and When to Use Modeling](#) in this chapter.

- By using the runtime typing capability available through the Type Manager, you can augment Windchill out-of-the-box business object types without changing the object model and writing code.

Runtime typing allows you to fine-tune the system and address changing needs without recompiling, rebooting, or stopping operations. Using the Type Manager to create a type that does not have an underlying modeled class creates a *soft type* as a subtype of the parent type. For example, you can create a soft type with the **Site Document** display name that is a subtype of `wt.doc.WTDocument` (instead of customizing code as described earlier). The **Create Type** window allows you to enter the initial information such as name, description and display name. The new soft type subtype inherits both the modeled and non-modeled attributes of the parent type. Additionally, you can update the soft type to add an associated icon, add additional attributes, and further refine constraints, or you can create new subtypes under the soft type.

You can use runtime typing to:

- Augment a Windchill part or document by adding additional attributes, or adding new types with different attribute sets (implementation).



- Quickly show your end users how Windchill could be used to solve their business problem (prototyping environment).
- Distribute Windchill to multiple divisions, when each division wants to modify slightly the site-specific modeled classes to enhance the part and document definitions for their own division (deployment.)

The Type Manager user interface, described later in this chapter, allows you to perform the following actions:

- Update the existing Windchill out-of-the box business object types.
- Create new soft types of the existing Windchill out-of-the-box part, document, and change object types, and any modeled extensions to these types created at your site.
- Create new attributes for any existing type.
- Define constraints for any existing attribute.
- Update types, attributes, and constraints.
- Delete types and attributes.
- Duplicate types (by copying and pasting) and move types (by cutting and pasting).
- Add a logical identifier for an existing type. For information on logical identifiers, see [Specifying Windchill Types](#).

**Note:** You can provide display values for types, attributes, and valid value lists in multiple locales. For details, see [Localizing New Type Definitions](#) in this chapter.

To add attributes to your type, select attributes from the global attribute set. The attributes you add can be optional or required, and constraints can be supplied for any of the new attributes. Constraints specify characteristics, such as range restrictions or a valid value set. You can also adjust or provide constraints on inherited attributes.

For more information on creating attributes in the global attribute set with the Attribute Definition Manager, see the *Windchill Classification Tools Installation Guide*.

## Effects of Deploying a New Type

When you deploy a new type definition, the type is recognized and used by the following Windchill functionalities:

- Access control policies
- Indexing policies
- Notification policies

- Life cycle template definitions
- Base loader
- Reporting
- Windchill adapter

You cannot define external file vault policies or replication policies specifically for a new type. However, the type inherits the rules that apply to the modeled class on which the type definition is based.

## Using Typing in Conjunction with Classification

The Windchill classification capabilities complement the Windchill typing capabilities. Typing allows you to define the business process characteristics of an object that can be used for processing the object through its product life cycle. Classification allows you to define the attributes that describe the object's form, fit, and function, used to classify and organize a product master database to promote consolidation of suppliers and reuse of design components.

## Overview of Classification

Windchill's classification capabilities enable Windchill to become a searchable repository for part and supplier data. This allows your organization to consolidate suppliers, standardize parts, manage multiple part number schemes, and promote the reuse of design and component knowledge across the manufacturing organization.

Sourcing administrators and design engineers can search the Windchill repository by navigating a textual hierarchy or image matrix, or by initiating parametric searches against attributes that describe the form, fit, and function of the business object. Advanced searching capability is available to help find second sources and functional equivalents for a selected part.

If classification is implemented at your site, you can develop navigation structures to help end users in their searches. A navigation structure is a hierarchical set of navigation nodes, each with a textual representation and a graphical representation. Navigation structures present product data in a way that will help different users uniquely. For example, a design engineer may want to view a detailed mechanical hierarchy of parts with a large set of attributes, while a dealer is interested only in locating the springs used on a particular tractor model. While the user navigates a Windows-like folder structure, in the background, the system "navigates" an object type, such as parts or suppliers. Queries from this navigation structure are executed against objects of that specific type.

Classification structures are a special case of navigation hierarchy, where a child in the hierarchy is always a type of its parent. Windchill classification structures provide templates for Classified Windchill objects, for example, parts and suppliers. Each node of a classification structure has an associated set of attributes that describe the part's form, fit, and function, as well as attribute value

constraints and a representative graphical image. Classification node attributes and constraints are used as a template for assigning form, fit, and function attributes and constraints to newly classified parts and suppliers. The template is not strictly enforced, and the end user may optionally change the template definition depending on the form, fit, and function data available for any instance of the object being classified. A single instance of a part or supplier can be classified multiple times to reflect different views of the same business object.

## Comparison of Classification and Typing

Windchill's typing capability allows you to extend the Windchill data model without using the modeling and system generation capabilities of Windchill's Information Modeler. The typing capability allows you to add attributes and further refine constraints for existing Windchill business objects. Typing also allows you to create subtypes of existing Windchill business objects and add attributes and constraints to those subtypes.

The attributes assigned through either modeling or typing are directly related to the product and process information necessary to manage the business object over its complete product life cycle. Product and process information is used to support business processes such as: the development of a Bill of Materials (BOM), support of where-used queries, and the development of a complete and accurate change history. Many of the attributes assigned to the definition of an object are inherited from fundamental Windchill base classes to support the management of the product through its life cycle. Other attributes are inherited through your organization's addition of new attributes using either modeling or the Type Manager.

Classification allows the end user to omit defined attributes from a particular instance of a business object; however, modeling and typing strictly enforce their definitions of the attributes expected for each instance. Business objects can have multiple classifications, but only one type.

## When to Use Typing and When to Use Modeling

The typing capability complements Windchill's model-driven customization approach of modeling with Rational Rose, system generation, and Java coding. The modeling and Java coding-based approach should be used to provide fundamental new types of information in the system. For example, a plant object would be a new type of information, where new functional capabilities are required and new behavior must be added, through new methods or implementation of plug-and-play interfaces.

You should use modeling in the following situations:

- You need to add behavior to an extension of a Windchill business class with new methods or plug-and-play interfaces. The modeled class could subsequently be augmented with additional attributes, added through typing, to reflect changing business needs at some time in the future.

- You want to develop granular policy rules for administrative policies that recognize typing. For example, if you want to develop a replication rule that would replicate only a particular type of part, you would need to create the part type by extending the part through modeling. This is because replication policies are based on classes.
- You want to make your tailored business objects visible to the end user and the rest of the Windchill system; however, the customization effort is less than the effort required to make the object visible through typing.

## Localizing New Type Definitions

Types and their attributes can be localized in the same files and in the same manner as modeled classes and attributes; however, entries in these files are not generated automatically.

Soft types can be localized just like modeled types. For example, if one wanted to localize a soft-subtype of WTPart, the partModelRB.rbInfo would be appended with entries like:

```
# Entry Format (values equal to default value are not included)
# <key>.value=
# <key>.category=
# <key>.comment=
# <key>.argComment<n>=
# <key>.constant=
# <key>.customizable=
# <key>.deprecated=
# <key>.abbreviatedDisplay=
# <key>.fullDisplay=
# <key>.shortDescription=
# <key>.longDescription=
```

where *<key>* is the external form for the soft type  
 "WCTYPE|wt.part.WTPart|com.myco.MySoftPart"

**Note:** Do not use multibyte characters in the soft type name. Multibyte characters are not supported.

Attributes can be handled in a similar way, but the corresponding file is com/ptc/core/meta/common/DefinitionResource.rbInfo:

```
# Entry Format (values equal to default value are not included)
# <key>.value=
# <key>.display=
# <key>.abbreviatedDisplay=
# <key>.fullDisplay=
# <key>.shortDescription=
# <key>.longDescription=
# <key>.dataType=
```

where *<key>* is the external form for the instance based attribute  
 "IBA|mySoftAttribute"

## Migrating Existing Type Instances to a New Type Definition

When a type definition is changed to add additional soft attributes, existing type instances are updated, using a strategy called *lazy migration*. This means the affected instances do not need to be changed immediately (that is, as soon as the new definition is deployed), but only when an individual instance is updated.

For example, if you add an attribute to a type, it will subsequently appear on forms to create new instances of the type and will be added to a new object instance if the user assigns it a value. It will not be automatically added to existing instances of the type; an end user must update the instance and give the attribute a value to add it to the instance.

## Defining Additional Attributes

In Windchill Foundation & PDM and Windchill PDMLink, you can define *soft attributes* in order to augment the attributes of out-of-the-box business objects. Soft attributes are non-modeled; they are created at runtime and do not require recompiling or interrupting operations.

You can define attributes for the following object types:

- Documents and document subtypes
- Products
- Parts
- Parts usage links
- Serial numbered parts
- Problem reports
- Change activities
- Change requests
- Change notices
- CAD documents
- Dynamic documents



**Caution:** See [Best Practices](#) for information about restrictions on creating soft types and soft attributes in Windchill PDMLink and Windchill ProjectLink.

## Adding Attributes

To create or add an attribute in Windchill Foundation & PDM, click **Attribute Administrator** on the **Business Administrator** page. From the **Attribute Administrator window**, click **Attribute Definition Manager**.

To add attributes in Windchill PDMLink or Windchill ProjectLink, click **Attribute Definition Manager** from the **Utilities** page of the **Site** tab. Only site administrators can access the Attribute Definition Manager.

Using the Attribute Definition Manager, you can name and set the type of a new attribute. After the attribute has been created, you can update it to specify a description and display name. A Logical Identifier must be defined for the soft attribute to make it searchable.

**Note:** When naming a new attribute, the name should not contain the following characters; "\" or "/". The suggested internet domain should not contain underscores or additional periods. Also, the name should not contain any lower case characters if it needs to be communicated with a Pro/ENGINEER designated parameter. For more detailed information on creating soft attributes, see the *Attribute Definition Manager* online help.

To add an existing attribute to one of the supported object types, start the Type Manager as described in the [Starting the Type Manager](#) section. The left pane of the resulting **Type Manager** window shows object types. You can add an attribute to any of the supported types listed earlier in this section.

**Note:** The names shown for change objects in the **Type Manager** window differ from those shown in Windchill. The following table explains how the change object names correspond:

Type Manager (Windchill Foundation & PDM) Change Object	Windchill PDMLink Change Object or Field
Analysis Activity	N/A
Change Activity	Change Notice Implementation Task
Change Investigation	N/A
Change Issue	Problem Report
Change Order	Change Notice
Change Proposal	Change Request Proposed Solution
Change Request	Change Request

Products and serial numbered parts are subtypes of Part.

For more information about adding attributes, see the [Using the Type Manager](#) section, the online help available from the Attribute Definition Manager, and the online help available from the Type Manager.

## Client Changes

When you add attributes to objects, Windchill clients are affected in the following ways:

- On create and update HTML pages, additional attributes appear following the standard fields. Certain types of attributes cannot be set. See [Types Not Supported in the HTML and Desktop Integration Clients](#).
- In Product Structure Explorer (PSE), to modify additional attributes in the create part and update part windows, click **Edit Attributes**. A dialog box opens to allow you to modify all additional attributes that have been added through the Type Manager. PSE supports all available attributes, including the attributes not supported by the HTML client. The default attribute values, defined in the **Type Manager**, are added to the part automatically if the user does not open this dialog during the creation of a part.
- On details pages, additional attributes (referred to as *properties* in the user interface) are displayed in one of the following ways:
  - For parts, end items, and documents, attributes are displayed in a separate table that can be viewed by clicking the **General** link on the left-side navigation bar of the object details page
  - For change objects, the attributes are displayed in the main attributes section of the object details page if the appropriate preference below is set to false in the Preference Manager:

DisplayChangeTaskCustomAttributesInTable

DisplayECNCustomAttributesInTable

DisplayECRCustomAttributesInTable

DisplayPRCustomAttributesInTable

If the preference is set to true, the attributes are displayed in a separate table in the **Additional Properties** section of the details page. By default the preference values are false.

**Note:** If a discrete set constraints has been defined for a String, Long, Floating Point, or Date/Time soft attribute, then the set of valid values for that attribute is presented in a drop-down list in HTML windows.

### Data Types Supported in the Attribute Definition Manager

The following types are supported:

- Boolean
- Date/Time stamp
- Floating Point Number
- Floating Point Number with Units

- Integer
- Reference
- String
- URL

### Types Not Supported in the HTML and Desktop Integration Clients

There are types listed in the Attribute Definition Manager that are not supported in the HTML and Desktop Integration clients. They are the following:

- Floating Point Number With Units
- Reference
- URL

Multiple values for attributes are not supported in the client. Only one value per attribute is accepted.

## The Type Manager

This section describes how to start the Type Manager, provides examples of using the Type Manager to create and update a soft type, and provides information about setting constraints.

### Starting the Type Manager

**Note:** In order to create or modify types, a user must either be a site administrator or an organization administrator that is a member of Type Administrators group.

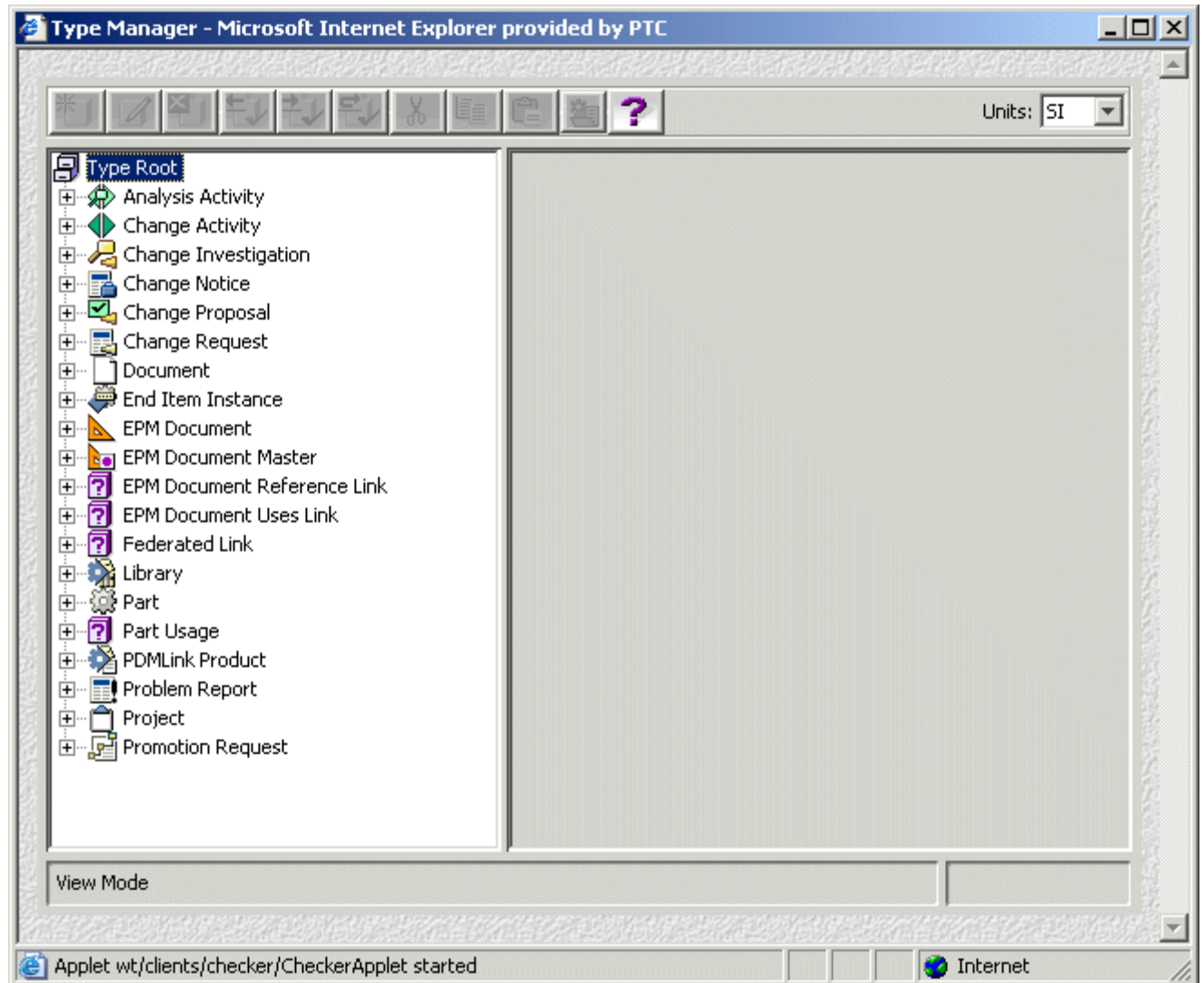
How you access the Type Manager is determined by your Windchill application:

- From Windchill Foundation & PDM, you can access the Type Manager by clicking the **Type Manager** link that is on the **Business Administration** home page.
- From Windchill PDMLink and Windchill ProjectLink, you can access the Type Manager from the **Utilities** pages that are under the **Site** and **Organization** tabs. The Type Manager link from the **Site** tab provides you (as the system administrator) with unrestricted access to all types. The link from the **Organization** tab provides access to only those types that are in the organization context that is active when you launch the Type Manager.

**Note:** You can also create, update, and delete document types from the **Types** pages that are under the **Site** and **Organization** tabs. You cannot create a document type from the **Types** page of the **Organization** tab when the organization has the same name as the site.



A window similar to the following appears:



The left pane displays the type hierarchy in an expandable and collapsible tree structure. The right pane is used for viewing, creating, and updating types, attributes, their values, and their constraints.

## Using the Type Manager

The following section displays two major examples: creating a new part soft type and updating a soft type.

The examples are based on a multinational company that has deployed a customized version of Windchill to a division. This division wants to further tailor one of the customized types.

The customized version of Windchill contains a new soft type named Part Type, which is extended from Part. This division wants to create a new soft type, based on Part Type, which will be named Unit 1 Part. The new soft type will contain the following new attributes:

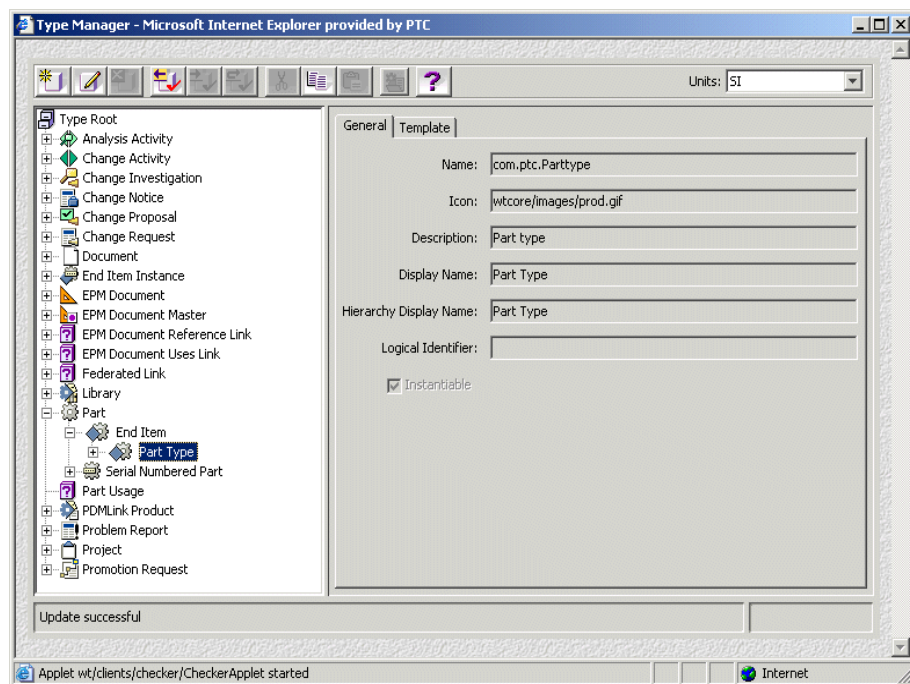
- *cageCode*, an integer value identifying the vendor.
- *procurementLeadtime*, a range of dates specified using timestamp values.
- *system*, a string type identifying whether the part is in the electrical, hydraulic, or pneumatic system.

After the new part type is deployed, it will be updated to add an attribute named spare, which will be a Boolean value.

## Creating a Type Definition

The following procedure creates the type and attributes described above.

1. Start the Type Manager from the **Site Utilities** page, as described earlier in this chapter, and expand the desired node.
2. Select the type **Part Type**, as shown in the following figure:



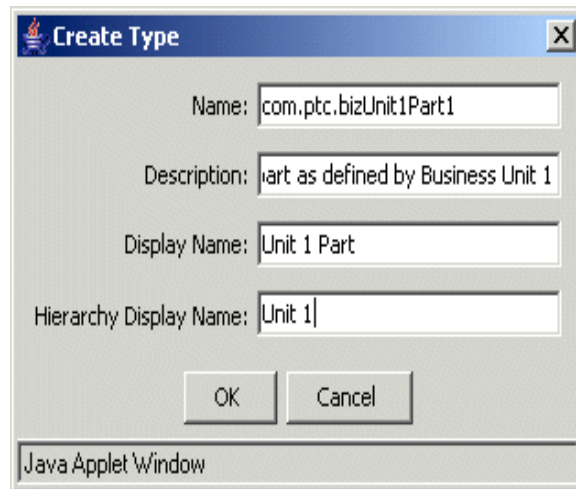
3. Click the **Create** icon.

The **Create Type** window appears.

In the window, the **Name** field contains a suggested prefix for the type (in this case, **com.ptc**). Including the suggested prefix ensures that the soft type being created is unique and is associated with the context from which you launched the Type Manager. For additional details on including a prefix in the type name, see [Associating Soft Types with the Site and Organization Containers](#).

The name should be unique among siblings and can have a maximum of 200 alphanumeric characters and underscores. Hyphens are not allowed in type names and underscores are not allowed in the internet domain.

4. Fill in the appropriate fields for the new soft type, bizUnit1Part1, as shown in the following figure.

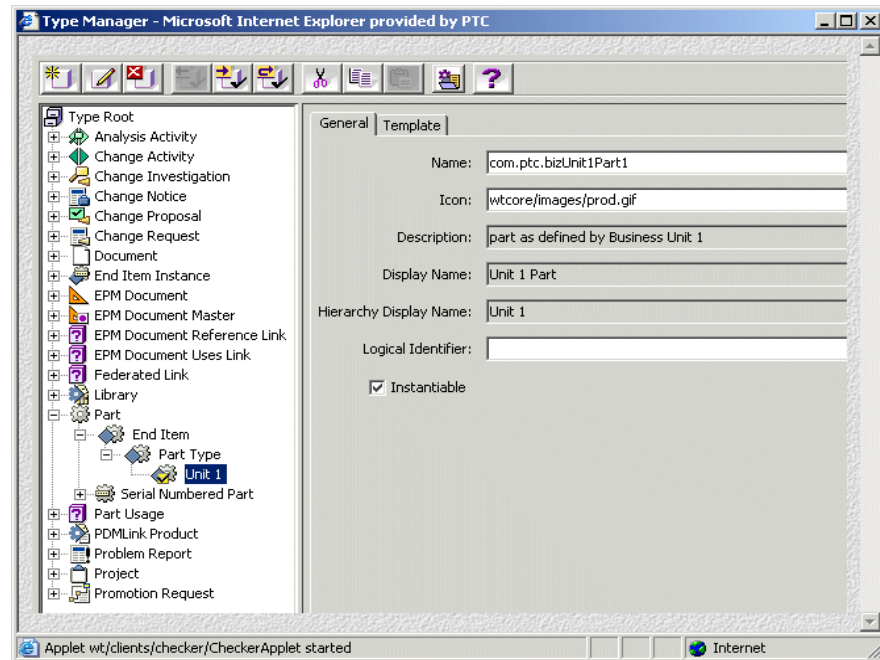


The screenshot shows a 'Create Type' dialog box with the following fields and values:

- Name: com.ptc.bizUnit1Part1
- Description: part as defined by Business Unit 1
- Display Name: Unit 1 Part
- Hierarchy Display Name: Unit 1

At the bottom of the dialog are 'OK' and 'Cancel' buttons. The window title bar reads 'Create Type' and the status bar at the very bottom reads 'Java Applet Window'.

5. Click **OK**. The new Unit 1 part type appears in the type hierarchy, as shown in the following figure. The new type is checked out and unavailable for use in the system. The Type Manager remains in update mode.



**Note:** By default, the **Instantiable** check box is selected. When the type is instantiable, the type appears in the list of available types when you are creating an object such as a part or document. To remove the type from the available types, remove the check mark and save your change.

6. At this point, you can add the new attributes: cageCode, procurementLeadtime, and system. To be accessible from the Type Manager, the attributes must be available from the global attribute set that is defined through the Attribute Definition Manager.

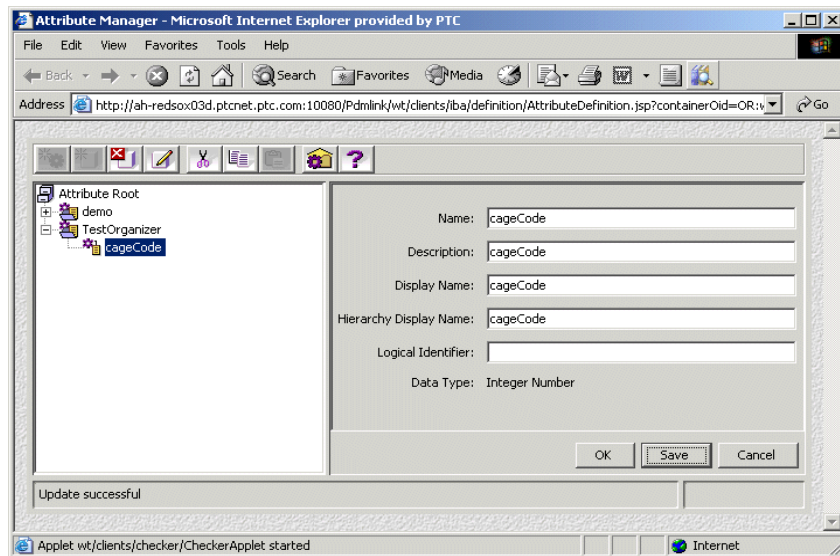
- a. Start the Attribute Definition Manager as follows:

- From Windchill Foundation & PDM, go to the **Business Administration** window and click **Attribute Administrator**.

**Note:** You can create new attributes in the Attribute Administrator at any time, even before you start the Type Manager session. But, if you are already using the Type Manager, neither it nor the method server need be restarted for the new attributes to be available for use.

From the **Attribute Administrator** window, click **Attribute Definition Manager**.

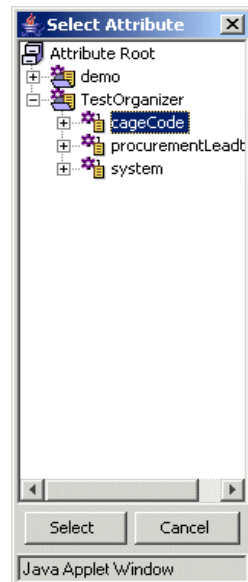
- From Windchill PDMLink or Windchill ProjectLink, click **Attribute Definition Manager** from the **Utilities** page of the **Site** tab. Only site administrators or users in the Attribute Administrators group can access the Attribute Definition Manager.
- b. In the resulting window, click **Create Organizer** to create an organizer named TestOrganizer. Then select **TestOrganizer** and click the create attribute icon. Enter the name **cageCode** in the **Name** field, and select **Integer Number** from the **Data Type** drop-down list. Click **OK** to create the new attribute. **cageCode** now appears as an available attribute, as shown in the following figure:



- c. Add the remaining attributes, procurementLeadtime and system, using the information provided at the beginning of this section and then return to the Type Manager.
- 7. Click the **Template** tab, which displays the types' attributes and their default values. Initially, no attributes appear. (Although they do not appear on this tab, all modeled attributes of the parent type are inherited by the new type.)
- 8. Click **Attribute Root**, and click **Add Attribute**. This opens the **Select Attribute** window.

**Note:** All actions for the buttons at the bottom of the **Template** page are also available through a right-click pop-up menu.

9. Expand the attributes available under TestOrganizer and select the kind of attribute desired, in this case, cageCode.

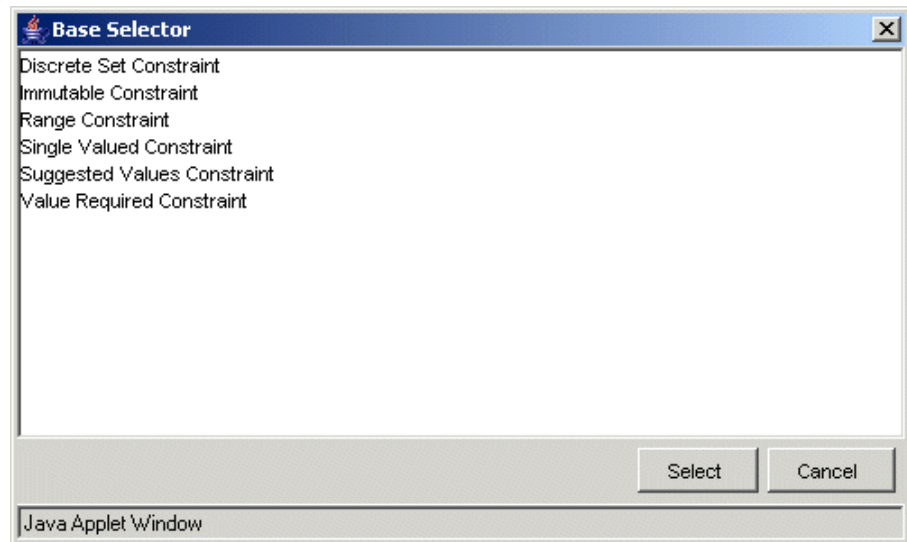


10. Click **Select**. This window closes, and you are returned to the **Type Manager** window, which now lists the new attribute cageCode and a default value of 0.
11. If you want to change the default value, click in the **Value** field corresponding to the new attribute and enter a different value.

**Note:** A default value is required for each attribute you add. The value is required to ensure there is at least one value for the attribute that is capable of satisfying the constraints you may define. (You are not allowed to check in any changes to types that violate their own constraints.)

12. Add the remaining attributes in the same manner.
13. To put constraints on any of these attributes, click the desired attribute and click **Show Constraint**. The **Constraint Editor** window opens.

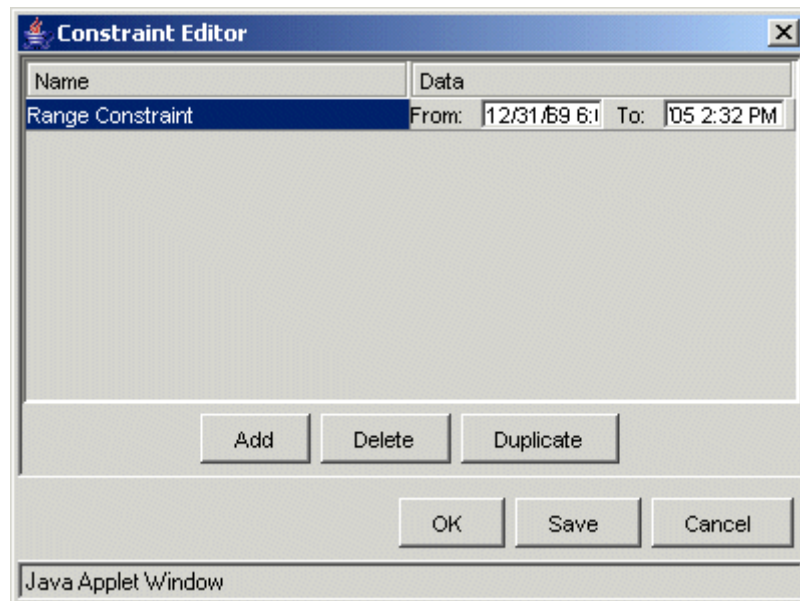
14. Click **Add**. The **Base Selector** window opens, as shown in the following figure.



15. Select the desired constraint. See [Setting Constraints](#) for more information. For example, to set a range for the `procurementLeadtime` attribute, select the **RangeConstraint** constraint, and click **Select**.

The **Base Selector** window closes and you are returned to the Constraint Editor, which now shows the attribute's constraint.

16. In the case of a range constraint, clicking the **Data** fields allows you to modify the two values for the range, as shown in the following figure.

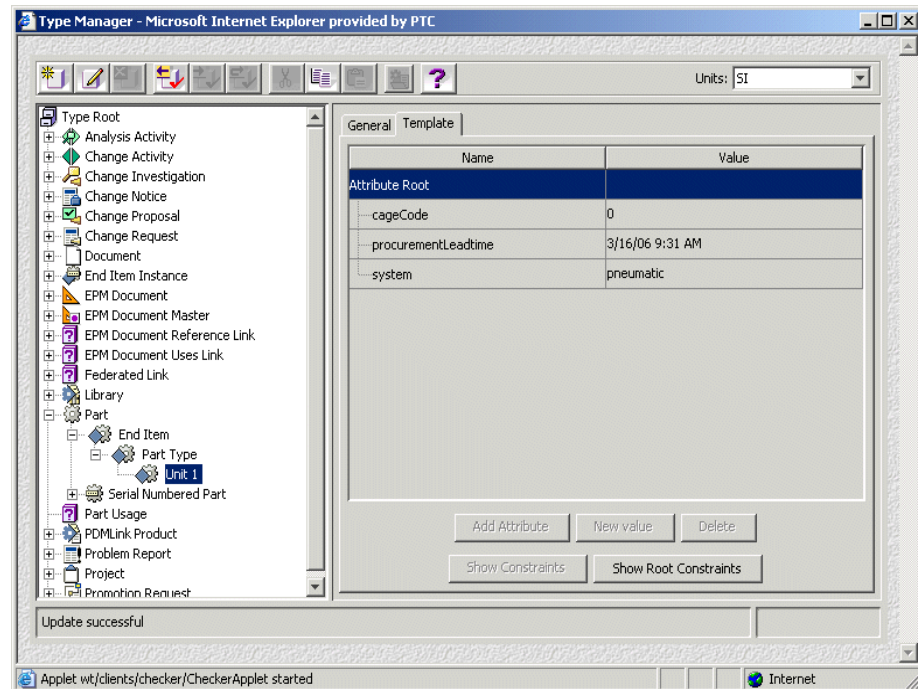


When you are ready, click **OK** to close the **Constraint Editor** window and return to the **Type Manager** window.

17. To specify allowable values for the attribute named **system**, a string type, select that attribute and return to the Constraint Editor's **Base Selector** window (that is, select the system attribute, click **Show Constraint**, click anywhere in the **Constraint Editor** window, and select **Add**).

In the **Base Selector** window, select the **DiscreteSetConstraint** and click **Select**. When you return to the **Constraint Editor** window, enter the possible values for system (electrical, hydraulic, and pneumatic) in the **Data** field of the constraint, separating each value with a vertical bar (|). Click **OK** to close the window and return to the **Type Manager** window. When the constraint is saved, the Constraint Editor checks for a vertical bar at the end of the list and adds one if it is not already there.

18. At this point, the new attributes are created, and have associated constraints and default values.



Click **OK** to save the attributes and return to view-only mode.

19. Check in the new type by selecting it and clicking the **Check in** icon. The type is now available for use in the system.



## Updating a Type Definition

The following procedure updates the type created in the preceding example and adds a boolean attribute named spare. Before an attribute can be added using the Type Manager, it must exist in the Global Attribute Set. You can add it using the same procedure given in the earlier example, or you can do it before or during your Type Manager session. The following example assumes the spare attribute has already been created:

1. In the **Type Manager** window, select the type Part Type and click the **Update** icon.

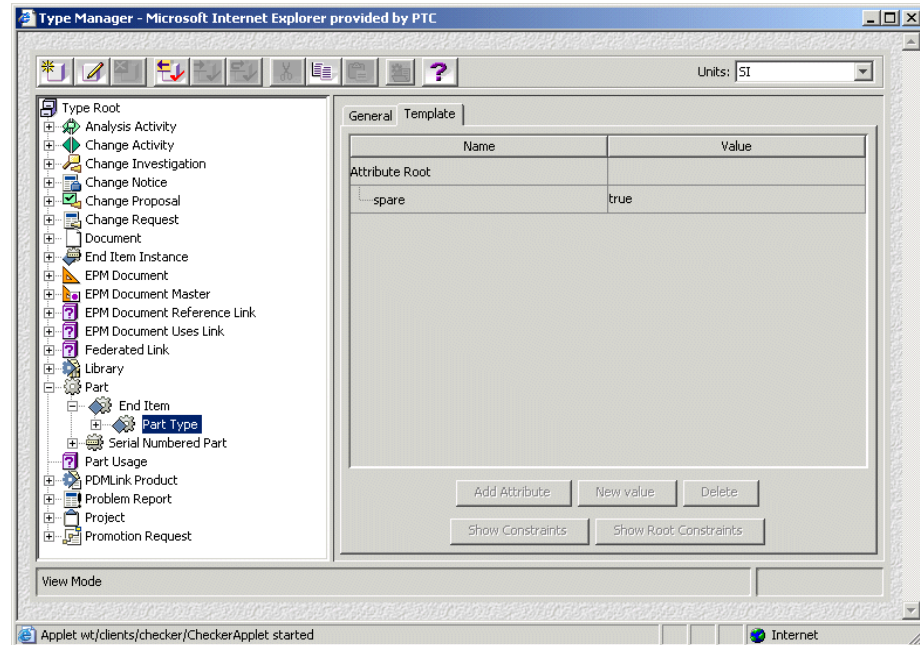
The type is now checked out automatically and no longer available for use in the system.

2. Follow a procedure below to add an attribute and set a default value:
  - a. Click the **Template** tab.
  - b. Click the **Attribute Root** field and click **Add Attribute**.
  - c. In the **Select Attribute** window, expand the TestOrganizer node and select the spare attribute. Then click **Select**. This attribute has a boolean type.

The spare attribute is added with a value of **false**.

- d. If you want the default value to be true, select **true** from the drop-down list in the **Value** field.
3. Click **OK** to save the new attribute.

4. Click the **Check in** icon to check in the updated type and make it available for use in the system.



New instances of the Part Type type now have the spare attribute. Existing instances can wait until the first time they are accessed to be updated. At this time, the spare attribute, and its default value, are added to the instance. This is known as lazy migration and prevents excessive activity in the database when a new attribute is added or deleted.

## Setting Constraints

The following table provides information about constraint types.

Type of Constraint	Applies to Attributes of Data Type	Can Apply to Container	Description	Example of Constraint Value
Discrete Set Constraint	All	No	The attribute value must be the same as one of the specified constraint values.	String data type: constraint value set: abc cde efg Legal strings can be abc, cde, or efg Integer data type: constraint value set: 1 2 3 Legal integer value can be 1, 2, or 3 <b>Note:</b> ' ' is the delimiter for the string values. Currently, ' ' is the reserved character.
Immutable Constraint	All	Yes	The user cannot change, add, or delete the value of the attribute except during initial object creation. <b>Notes:</b> To add this constraint, the user has to save the attribute value first. After this constraint is imposed to the container, the container is frozen. It cannot be modified any further. In Windchill PDMLink and Windchill ProjectLink, attributes with immutable constraints cannot be modified after the object is created; when you update an object, immutable attributes are present but are grayed out.	No constraint value.

Type of Constraint	Applies to Attributes of Data Type	Can Apply to Container	Description	Example of Constraint Value
Range Constraint	Can be applied to All data types, but will not be effective on boolean data type	No	<p>The actual value of the attribute must be greater than or equal to the maximum values specified (the range is inclusive)</p> <p><b>Note:</b> "From:" specifies the minimum value. "To:" specifies the maximum value.</p>	<p>From: A To: Z.</p> <p>Legal strings can be Abc, BCDE, ...</p>
Single Valued Constraint	All	Yes	<p>No more than one value is allowed for a specific attribute. When applied to Container, no attribute can have more than one value.</p> <p><b>Note:</b> This constraint is not applicable in Windchill PDMLink and Windchill ProjectLink; multiple values are not allowed for attributes.</p>	No constraint value.
String Format Constraint	Can be applied to All data type, but only effective on String data type.	No	<p>Provide a set of basic masking to regulate the format of a string. The constraint value is a set of strings defining positional formats for the string content. C, L, and D in constraint value are reserved characters and should not be used as delimiters; all the other characters are considered delimiters.</p> <p>C means one letter or one digit. L means one letter. D means one digit.</p> <p>The definitions of letter and digit can be found in Java.lang.Character Class.</p>	<p>1. SSN Formatting value: DDD-DD-DDDD. Legal strings can be 123-45-6789, 452-98-4444, ...</p> <p>2. Telephone number Formatting value: (DDD)DDD-DDDD  DDD-DDD-DDDD   D-DDD-DDD-DDDD. Legal string can be (555)454-6789, 555-198-3247, 1-800-436-7869, 1-800-CAN-HELP,...</p> <p>3. Air flight seat number Formatting value: DL D-L. Legal strings can be 1A, 3-E, ...</p>

Type of Constraint	Applies to Attributes of Data Type	Can Apply to Container	Description	Example of Constraint Value
String Length Constraint	Can be applied to All data type, but only effective on String data type.	No	The length of the string value must be greater than or equal to the minimum, and less than or equal to the maximum values specified (the range is inclusive).  <b>Note:</b> "From:" specifies the minimum length. "To:" specifies the maximum length.	From: 3 To: 200. 3 <= length of legal string <= 200.
Suggested Values Constraint	All	No	Provide a set of suggested values to the Constraining.  <b>Note:</b> In Windchill PDMLink and Windchill ProjectLink, these values are presented in a tool tip on HTML object creation and update forms.	String data type: constraint value set: abc cde efg. Legal strings can be abc, cde, or efg. Integer data type: constraint value set: 1 2 3. Legal integer value can be 1, 2, or 3.  <b>Note:</b> ' ' is the delimiter for the string values. Currently, ' ' is the reserved character.
UpperCase Constraint	Can be applied to Add data type, but only effective on String data type.	No	The String values are converted to Uppercase when saved.	No constraint value.
Value Required Constraint	All	Yes	The attribute must have at least one value. When applied to Container, each attribute must have at least one value.	No constraint value.

Type of Constraint	Applies to Attributes of Data Type	Can Apply to Container	Description	Example of Constraint Value
Wildcard Constraint	Can be applied to All data type, but only effect on String data.	No	<p>The String attribute value must match the Wildcard pattern of constraint value specified.</p> <p>Contains: Contains the value specified.</p> <p>Begins With: Begins with the value specified.</p> <p>Ends With: Ends with the value specified.</p> <p>Exact: Exactly the same as the value specified.</p>	<p>'Contains' abc.</p> <p>Legal strings can be Ababc, abcZ, AabcZ,...</p> <p>'Ends With' er.</p> <p>Legal strings can be Aer, Developer, ...</p>

## Specifying Windchill Types

When you are working with the types that are available from the Type Manager, you need to be aware of how to identify the types. For example, you can specify types in object initialization rules, properties, preferences, publishing to Arbortext, and custom code.

When specifying an object type, use the logical identifier of the type.

The logical identifier of a type can be set when out-of-the box types are loaded or can be set when you create (or update) a type through the Type Manager.

The logical identifier must be unique across all types in the system and is limited to 50 characters, with the following restriction:

Only letters, numbers, periods, and underscores are supported.

Often, the logical identifier is the same as the type name. If you are unsure of whether a type has a logical identifier, you can view the information from the **General** tab of the Type Manager.

**Tip:** If the type does not have a logical identifier, you can add one by updating the type using the Type Manager.



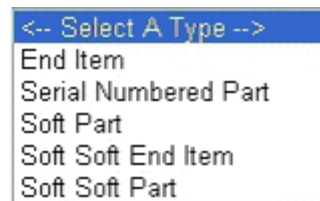
**Caution:** After a logical identifier has been set and used in your solution, do not change the identifier. Changing the identifier after it is being used can cause problems.

# Managing Types and Attributes for Parts, Documents, and Change Objects

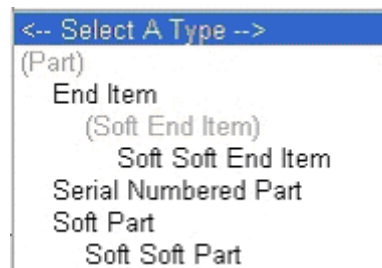
The create action for parts, end items, documents, and change objects includes a **Type** drop-down list from which the user can select the type of object to be created. You can determine if type lists display as flat lists or hierarchical lists, define defaults for type lists, and exclude types from appearing in a type list. This section also provides some information that can help you understand how soft attributes are displayed.

## Displaying Type Lists

The following is an example of a flat list of types. By default, type lists are displayed as flat lists.



The following is an example of a hierarchical list of the same types.



In the hierarchical list, types that have been marked as uninstantiable in the Type Manager or to which the user does not have create privileges are shown in parenthesis, and depending on the browser, may also be grayed out. These types cannot be selected.

The following table describes the preferences affecting the type drop-down list; the table includes preferences for both the display and the default value that appears in the drop-down list:

Preference	Description
/com/ptc/core/foundation/type/common/DisplayHierarchy	Determines whether the type lists are flat (the default) or hierarchical. Specify TRUE for a hierarchical list.

Preference	Description
/com/ptc/windchill/doc/DefaultType	Specifies the default type for documents in products or libraries.
/com/ptc/windchill/netmarkets/doc/DefaultType	Specifies the default type for documents in projects, sites, and organizations.
/com/ptc/windchill/part/DefaultType	Specifies the default type for parts in products or libraries.
/com/ptc/windchill/netmarkets/part/DefaultType	Specifies the default type for parts in projects, sites, and organizations.
/com/ptc/windchill/enditem/DefaultType	Specifies the default type for end items.
/wt/change2/DefaultType/ wt.change2.WTChangeActivity2	Specifies the default type for Change Activities.
/wt/change2/DefaultType/ wt.change2.WTChangeIssue	Specifies the default type for Problem Reports.
/wt/change2/DefaultType/ wt.change2.WTChangeOrder2	Specifies the default type for Change Notices.
/wt/change2/DefaultType/ wt.change2.WTChangeRequest2	Specifies the default type for Change Requests.

The possible values for the DefaultType preferences are as follows:

To display the following as a default value:	Specify the following for the preference value:
Nothing; a blank.	The string "blank".
A string of your choosing. For example, "Select A Type".	<p>A string indicating an entry in an RBINFO file that contains the text you want to display. The string should be in the format:</p> <p style="text-align: center;"><i>filepath:constant</i></p> <p>where <i>filepath</i> is the file path for the RBINFO file, and <i>constant</i> is the constant for the RBINFO file entry.</p> <p>For example:</p> <pre>com.ptc.core.foundation.type.common.common Resource:SELECT_A_TYPE</pre>



To display the following as a default value:	Specify the following for the preference value:
One of the modeled types in the list.	The logical identifier of the type. See <a href="#">Specifying Windchill Types</a> .
One of the soft types in the list.	The logical identifier of the type. See <a href="#">Specifying Windchill Types</a> .

## Excluding Types from a Types List

By default, the **Type** drop-down list displayed in the create windows includes all creatable and instantiable types, starting with the root type (WTPart or WTDocument), and includes all subtypes, both soft and modeled.

To exclude types from the list, use the following properties in the wt-properties file. These properties contain a delimited list of types to be excluded from the **Type** drop-down list. If the property already exists, you can add types to the list. If the property does not exist in your wt.property file, you can add it.

Object	Property
Part	wt.typepickerexclude/wt.part.WTPart
Document	wt.typepickerexclude/wt.doc.WTDocument
Change Issue	wt.typepickerexclude/wt.change2.WTChangeIssue
Change Request	wt.typepickerexclude/wt.change2.WTChangeRequest2
Change Notice	wt.typepickerexclude/wt.change2.WTChangeOrder2
Change Activity	wt.typepickerexclude/wt.change2.WTChangeActivity2

Use the logical identifier of the type when setting the properties. For example, if your site has a custom modeled subtype of WTPart called ext.part.MyPart (with a logical identifier of the same name) and a soft type of MyPart called com.mycompany.foo.MySoftPart (with a logical identifier of the same name), and you want to exclude these types from the **Type** drop-down list, then you can set the following property:

```
wt.typepickerexclude/wt.part.WTPart=ext.part.MyPart,com.mycompany.foo.MySoftPart
```

By default, a comma is used to separate types in the list. However, that default can be overridden by modifying the wt.property:

```
wt.typepickerexcludelimiter=,
```

**Note:** Modify the wt.properties file using the xconfmanager utility. For details on using the xconfmanager utility, see [About the xconfmanager Utility](#). For information on specifying the logical identifier of the type, see [Specifying Windchill Types](#).

## Understanding Soft Attribute Behavior on Object Create and Update Windows, and on Information Pages

The following list describes the soft attribute behavior on object create and update windows:

- Soft attributes are not marked as required on create and update windows unless they are given a **Value Required** constraint in the Type Manager. If a nonrequired attribute is not given a value on a create or update window, it is displayed without a value on the object information page.
- The default values given for attributes in the Type Manager are not prepopulated in the attribute input fields on the create and update windows. If the attribute has not yet been set for the object, and the attribute has a default value, a **Use Default** button is displayed to the right of the input field. Clicking this button sets the attribute to the default value. Input fields for custom modeled attributes also have a **Use Default** button if a default value was specified for the attribute.

If your site does not want to display a **Use Default** button for a given attribute, the button display can be turned off by customizing using the AllClients.xml configuration file. For customization details, see the *Windchill Customizer's Guide*.

- Multi-line input fields are displayed for string type soft attributes if the maximum permitted length of the string exceeds the value of the StringLengthThresholdForMultilineInput preference. The maximum length of string soft attributes is 500 characters by default, but can be set to a smaller value using the **String Length** constraint in the Type Manager.

A multi-line input field is defined as a field in which multiple lines of text can be entered. It is not necessarily more than one line in height.

You can override this behavior by specifying that a single-line or multi-line input field is to be used for a given attribute in the AllClients.xml configuration file.

- On create and update windows required modeled attributes are listed before nonrequired modeled attributes; each group of attributes is sorted alphabetically by display name.
- On information pages, the attributes are sorted alphabetically by display name with no differentiation of required and nonrequired attributes.

# Managing Types and Attributes for CAD Documents

Data files created by CAD applications and saved to Windchill using a workgroup manager or Pro/ENGINEER Wildfire are managed in Windchill as CAD documents.

There is one main out-of-the-box soft type that is used for CAD documents. This soft type is defined as a subtype under the wt.epm.EPMDocument modeled type. The soft type is named *<prefix>.DefaultEPMDocument*, where *<prefix>* is generated by reversing the internet domain specified for the default organization during the installation. For example, if the internet domain specified is acme.com, then the *<prefix>* used in the soft type name is com.acme. This same prefix is used for all out-of-the-box soft types that are automatically loaded.

The soft type used for CAD documents cannot be deleted nor can you create subtypes under this subtype.

You can perform the following actions with regard to the CAD document soft type:

- Add new attributes for the soft type.
- Define constraints for any existing attribute.
- Update the type, attributes, and constraints.
- Delete attributes.

To assign attributes to CAD documents, add the attributes to the default CAD document soft type. All attributes are defined globally using the Attribute Definition Manager.

The attributes you add can be optional or required, and constraints can be supplied for any of the new attributes. Constraints specify characteristics, such as range restrictions or a valid value set. You can also adjust or provide constraints on inherited attributes. If there is an Immutable Constraint applied to an attribute, the attribute cannot be removed from the type definition. For additional information about setting constraints, see [Setting Constraints](#).

**Tip:** To avoid difficulties based on differences in how Windchill and various CAD editing applications assign significant figures to restricted attribute values, PTC recommends that you use the **Significant Figures Set Constraint**, rather than **Discrete Set Constraint**, for constraining floating point IBA values. This allows the system to accept values within a reasonable range of precision on either side of the desired value.

## Renaming the CAD Document Soft Type

Pro/ENGINEER Wildfire assumes that the out-of-the-box CAD document soft type has not been renamed. PTC recommends that you do not rename the out-of-the-box CAD document soft type other than changing the **Display Name**, which is visible in the client.

You can rename the soft type; however, the soft type needs to be recognized by the system in order to assign the proper default soft type. Therefore, if you must change the name, you also need to record the change by setting the following site level preference. Otherwise, the system will fail to recognize the soft type as the default type for CAD documents.

```
/wt/type/defaultType/wt.epm.EPMDocument
```

Set this preference to logical identifier of the soft type using the Preference Manager. For details on the required type format, see the [Specifying Windchill Types](#) section.

## Additional Soft Types Related to CAD Documents

Windchill provides the Workgroup Manager CAD Document soft type as the subtype for adding attributes that can have different values for each iteration of the item. Although this is probably the most commonly used type for CAD documents, you should be aware of the other soft types that can be used. In all of the following soft types, *<prefix>* is generated when the type is loaded by reversing the internet domain specified for the default organization during the installation:

- Workgroup Manager CAD Document Master soft type (*<prefix>.DefaultEPMDocumentMaster*) is a subtype of the EPM Document Master type (*wt.epm.EPMDocumentMaster*).

Attributes that are added to this soft type have only one value for all iterations. Therefore, changing the value of an attribute on a Workgroup Manager CAD Document Master soft type changes that value for all iterations.

Attributes added on this type are the Windchill equivalent of Pro/INTRALINK non-versioned attributes.

- Workgroup Manager CAD Document Uses Link (*<prefix>.DefaultEPMMemberLink*) soft type is a subtype of the EPM Document Uses Link (*wt.epm.structure.EPMMemberLink*) type.

Attributes that are added to this soft type are specific to the use of an iteration of an item. For example, if there are four bolts of the same type (*bolt.prt*) in an assembly and each bolt needs to be tightened to a specific torque, you can add torque to this Uses Link soft type and then apply a different value to each occurrence of the bolt in the assembly. In contrast, if you instead add this attribute to the CAD Document soft type, then all bolts in all assemblies would have the same torque wherever they are used.

- Workgroup Manager CAD Document Reference Link (*<prefix>.DefaultEPMReferenceLink*) soft type is a subtype of the EPM Document Reference Link (*wt.epm.structure.EPMReferenceLink*) type.

Attributes that are added to this soft type apply to reference links (not to the CAD document, itself).

**Tip:** Do not add attributes with the same name to different soft types. By using different names you can avoid confusion for users when they are adding and displaying the attribute values.

## Managing Types and Attributes for Dynamic Documents

Data files created by the Arbortext Editor and saved to Windchill using the Windchill connection to the editor are managed in Windchill as dynamic documents.

There are two out-of-the-box soft types defined for use with the Arbortext Editor. Both reside under the wt.epm.EPMDocument modeled type and provide the initial set of soft types defined for dynamic documents.

The soft type named `<prefix>.DynamicDocument` (where `<prefix>` is generated by reversing the internet domain specified for the default organization during the installation) is a subtype of EPM Document and is used to identify XML files authored in the Arbortext Editor and to identify graphic files referenced in the XML files.

The soft type named `<prefix>.BurstConfiguration` is a subtype of the Dynamic Document soft type and is used to identify burst configuration files stored in Windchill. Burst configuration files provide the details of how to split up a document into individual dynamic documents based on user-defined rules. For example, a document could contain multiple chapters and the burst configuration file might have a rule that says to store each chapter as a separate dynamic document in Windchill.

## Renaming Dynamic Document and Burst Configuration Soft Types

The current Windchill PDMLink implementation with Arbortext assumes that the loaded soft types are named as described in the previous section. PTC recommends that you do not rename the out-of-the-box soft types other than changing the **Display Name**, which is visible in the client.

If you rename the soft types, the soft types need to be recognized by the system in order to assign the proper default soft types. Therefore, if you must change the names, you also need to record the changes by setting the following site level preference. Otherwise, the system will fail to recognize the soft types as default types.

The following preference controls the default type established for dynamic documents:

```
/wt/type/defaultType/DynamicDocument/wt.epm.EPMDocument
```

The following preference controls the default type established for bursting configuration files:

```
/wt/type/defaultType/BurstConfiguration/wt.epm.EPMDocument
```

Set each of these preferences to logical identifier of the soft type using the Preference Manager. For details on the required type format, see the [Specifying Windchill Types](#) section.

## Creating Additional Soft Types for Use with Arbortext Editor and the Arbortext Publishing Engine

PTC recommends that you define a different soft type for each XML element you want to associate with bursting rules (for example, chapters and sections). In the bursting configuration file that is managed from the Arbortext Editor, those setting up Arbortext must specify the soft types that are applied at the time that documents are burst into Windchill. For information on bursting, see the Arbortext Editor online help.

The soft types defined are also used in the PublishRules file for configuring representations and are available to users when they browse Windchill for dynamic documents.

After the following activities are complete, users can select a specific soft type to narrow searches for objects and the soft types are available for use when bursting dynamic documents and when creating representations (such as PDFs) from dynamic documents:

- Windchill connection has been installed and configured for the Arbortext Editor. For details, see the readme file delivered with the Windchill connection and the Arbortext Editor online help.
- Arbortext Publishing Engine is installed and configured. For details, see the *Windchill Installation and Configuration Guide -Visualization Services*.
- Related soft types have been created.

The following is an example that shows the mapping of Arbortext XML elements and Windchill Dynamic Document soft types.

Assume the following Arbortext XML elements are the levels to be burst:

book  
chapter  
sect1

To support these bursting levels in Windchill PDMLink, create three soft types. Each soft type is used for one of the Arbortext XML elements. For example, soft types with the following display names could be created:

Book  
Chapter  
Section

For details on creating soft types, see the online help available from the Type Manager. For examples of creating and updating types, see the [The Type Manager](#) section.

**Note:** The type names specified in the Type Manager include a prefix that identifies the owning organization for the type. A suggested prefix is always presented when you are creating a new subtype and is generated based on the internet domain associated with the owning organization principal of the context from which the type is being created. For additional information, see the [Associating Soft Types with the Site and Organization Containers](#) section.

After the soft types are created, communicate with those setting up the Arbortext bursting configuration file and the PublishRules file to tell them which soft type they should associate with each Arbortext XML element. The following section describes the format of the type name that is required in Arbortext.

## Specifying the Logical Identifier for Types Used with Arbortext

Setting up document bursting in Windchill for use with the Arbortext files requires the use of logical identifiers for all types associated with dynamic documents. This is because the types are referenced in files within the Arbortext Editor and must be unique across the entire system.

To determine the logical identifier for a type, view the type from the Type Manager. For details on using the Type Manager, see [The Type Manager](#).

On the **General** tab, there is a **Logical Identifier** field that contains the value that should be used for the type in Arbortext files. If the field is empty, you can update the type to add its logical identifier. For details on the logical identifier format, see [Specifying Windchill Types](#).

## Defining Windchill Soft Attributes for Attributes Used with Arbortext

As is the case with other types, you can create soft attributes in Windchill PDMLink that can be stored and used with objects that are associated with the Dynamic Document soft type or subtypes of Dynamic Document. Arbortext bursting rules can be used to populate the soft attributes from XML attributes when storing objects in Windchill. For example, if the author enters a vendor's name in an XML document, bursting rules can be set up so that the vendor's name is populated in a Windchill soft attribute, such as supplier. The benefit to setting up soft attributes for this purpose is that the author only enters the information in one location, but the information is accessible through the XML document and through the Windchill soft attribute.

Additionally, you may want to create a specific set of attributes that are associated with the representables created when publishing dynamic documents through the WVS publisher. These attributes would be associated with the WTDocument type or a subtype of WTDocument. For information about using the WVS publisher for publishing dynamic documents, see the *Windchill Installation and Configuration Guide -Visualization Services*.

For information about creating soft attributes, see [Defining Additional Attributes](#).

## Best Practices

The following sections provide best practices for soft types and attributes in Windchill PDMLink and Windchill ProjectLink, for CAD documents in Windchill Foundation & PDM, and for searchable types.

### For Windchill PDMLink and Windchill ProjectLink

The following sections provide information on using soft types and soft attributes in Windchill PDMLink and Windchill ProjectLink.

#### Associating Soft Types with the Site and Organization Containers

When creating soft types from the Type Manager, the interface always provides a suggested prefix for the new type. The Type Manager determines what prefix to suggest based on the context from which the Type Manager was launched. The suggested prefix is the reverse internet style domain derived from the internet domain defined on the organization principal owning the container.

**Note:** The internet domain defined on the organization principal is important when creating new soft types. During the installation process, a default organization principal is created that contains an internet domain associated with the organization principal. The internet style domain prefix for soft types (created by reversing the internet domain) is used to identify which organization owns the type. The default organization principal is associated with the site container; any types owned by this organization principal are available from all organization containers. Additionally, any soft types created with internet style domain prefixes that do not map to known organization principals are considered to be site types and are available from all organization containers. In an environment where there is only one organization used, you can create the organization container using the default organization principal (as is the case when creating the organization container using the PDMLinkOrgContainer.xml described in the [Creating the Default Organization Container](#) section of Getting Started chapter). In this case, the soft types created from the site container and from the organization container have the same internet domain. If you want to create soft types that are only available from a specific organization container, the organization principal used when creating the organization container must be different from the organization principal associated with the site container. In an exchange environment where there are multiple organizations, you can restrict access to the soft types created in an organization container to those members in the container. This is done by setting the internet domain defined in the organization principal associated with the organization container to a value different from the site internet domain. For information on changing the internet domain, see [Changing an Established Internet Domain](#).

#### Restricting the Use of Soft Types and Soft Attributes

If you are not customizing the Windchill PDMLink or Windchill ProjectLink interface, restrict the use of soft types and soft attributes as described in the



following paragraphs. If you do not restrict their use, client customizations are necessary to expose the new soft types and soft attributes.

From the **Site** and **Organization** tabs, administrators who can access the Type Manager (see [Starting the Type Manager](#)) can define soft types for the following object types without requiring any client customizations:

- wt.part.WTPart
- wt.part.WTProduct (End Item)
- wt.part.WTSerialNumberedPart
- wt.doc.WTDocument
- wt.change2.WTChangeActivity2 (Change Activities)
- wt.change2.WTChangeIssue (Problem Report)
- wt.change2.WTChangeOrder2 (Change Notice)
- wt.change2.WTChangeRequest2 (Change Request)

For information on defining soft types for use with dynamic documents in Windchill PDMLink, see [Managing Types and Attributes for Dynamic Documents](#).

All soft attributes are defined in the Site context through the Attribute Definition Manager. From the **Site** and **Organization** tabs, administrators who have access to the Type Manager can include certain types of soft attributes on the following object types without requiring client customizations:

- subtypes of wt.part.WTPart
- subtypes of wt.part.WTProduct
- subtypes of wt.part.WTSerialNumberedPart
- wt.part.WTPartUsageLink
- wt.doc.WTDocument and its subtypes
- wt.change2.WTChangeActivity2 (Change Activities) and its subtypes
- wt.change2.WTChangeIssue (Problem Report) and its subtypes
- wt.change2.WTChangeOrder2 (Change Notice) and its subtypes
- wt.change2.WTChangeRequest2 (Change Request) and its subtypes

For information on defining soft attributes for use with dynamic documents in Windchill PDMLink, see [Managing Types and Attributes for Dynamic Documents](#).

**Note:** For details on the types of soft attributes that require customizations, see [Client Changes](#).

In a Windchill ProjectLink exchange environment, individual organization administrators can define WTDocument subtypes for the project contexts the organization is hosting.

In a Windchill PDMLink environment (standalone or combined with Windchill ProjectLink), type management activities can occur in the site and organization context.

In a Windchill Foundation & PDM environment (standalone or combined with Windchill ProjectLink), type management activities should occur globally or only in the site context.

Implications for Windchill users are as follows:

- Business objects created in the site context must be instances of modeled classes or instances of soft types defined in the site context.
- Business objects created in an organization context must be instances of modeled classes or instances of soft types defined in the site or given organization context.
- Business objects created in an application context must be instances of modeled classes or instances of soft types defined in the site or parent organization context.

## Enhancing Performance on Systems with Many Soft Types

For Windchill PDMLink (both standalone and in combination with Windchill ProjectLink) the following preference enhances the performance of systems that contain many soft types. Pages that display create actions, such as the **Folders** page, take less time to load with the following preference specified using the Preference Manager:

Preference Root: /com/ptc/windchill/enterprise/templateutil

Preference Key: create\_valid\_check

Preference Values:

- Bypass (default) - All create actions will show up by default, bypassing any access checks as a trade-off for higher performance.
- Base - The create action availability is determined by checking access against the base hard-type (i.e. WTPart, WTDocument, etc.) and does not do the extra processing to determine the specific soft/sub-types that a user has create access for. This is the medium performance option.
- Full - The availability of all create actions are processed using the base-type and all soft/sub-types, thus only showing the action when a user can truly perform a create. This option has the highest overhead and has the poorest performance.

## For CAD Documents

The following steps describe the creation of Windchill Foundation & PDM version and non-version attributes for CAD documents. Similar steps pertain to the creation of attributes for inclusion in templates for Windchill Foundation & PDM member links and reference links.

1. Create definitions for CAD document attributes through the **Attribute Definition Manager**. You create both version and non-version CAD document attributes in this window, as well as link attributes. Instance base attributes can have any name you wish.
2. Open the **Attribute Root** folder. Click the **Create Attribute** icon. The **Create Attribute** window appears.
3. In the **Create Attribute** window, type the attribute name, data type, and any other data required.
4. Click **OK** in the **Create Attribute** window; click **Save** in the **Attribute Definition Manager** window.
5. Open the **Type Manager** to begin adding attributes to the CAD document soft type definition or to a CAD document. You will check out different objects, discussed in the following steps, depending on whether you are creating version or non-version attributes.

The type of checked-out object to which you add an attribute restricts the mapping of data to the attribute.

6. Check out an object by selecting it and clicking the **Checkout** icon. Initiate the update of the object by clicking the **Update** icon.
  - For non-version attributes check out the WorkGoup Manager CAD Document Master.
  - For version attributes, check out the WorkGoup Manager CAD Document.
7. Click the **Template** tab. Click **Add Attribute**. Click on the attribute root to add previously created attributes. You can select and add multiple attributes simultaneously.
8. Define constraints in the **Constraint Editor** and **Base Selector**.
9. Click **OK** when the selection of attributes is complete. Click in the row for the attribute to add a value for the attribute. Click **Save** in the lower right corner of the window.
10. Check in the object when you have completed the work with attributes by clicking the **Check In** icon.
11. From the **Windchill Foundation & PDM** home page, choose **Data Mapping > CAD Attribute Mapping** to specify mapping of Windchill PDMLink system attributes to Windchill attributes. This operation specifies the

one-to-one mapping of attribute data into Windchill Foundation & PDM attributes.

**Note:** Mapping is case sensitive: Windchill PDMLink system attribute names must be in upper case.

The type of checked out object to which you added an attribute restricts the mapping of data to the attribute. Version attributes appear in the Windchill Foundation & PDM attribute list when you click **Versioned**.

To create a mapping, select an attribute and a Windchill Foundation & PDM attribute and click **Link**. You can mark mappings for deletion by selecting check boxes in the rows that represent them.

12. Click **Save** to create the mappings that you specified and to delete the mappings that you marked for deletion.

When you examine CAD document structure, the version or non-version character of Windchill Foundation & PDM attributes is shown in the right-hand **Applicable to** column. Version attributes are marked while **This Iteration Only** while non-version attributes are marked **All Iterations**.

## Soft Type Attribute Definitions, Template, and Attribute Types

The type of checked out object to which you added a Windchill Foundation & PDM attribute defines the type of Windchill Foundation & PDM object that can include that attribute. The following table shows the four types of Windchill Foundation & PDM soft type definitions to which you can add attributes and corresponding options in mapping CAD document attributes to the Windchill Foundation & PDM attributes. This scheme is a result of the soft typing that enforces an orderly set of data transitions. The left-hand column in the table shows the checked out Windchill Foundation & PDM object while the right-hand column shows the option that you select under the **CAD Attribute Mapping**.

Checked out Windchill Foundation & PDM Object	Object and Attribute
CAD document	Non-versioned in inheriting CAD documents
CAD document master	Versioned in all CAD documents
Reference link master	Versioned in Reference links
Member link master	Versioned in Membership links

Each attribute mapping defines an exclusive relationship between a CAD document attribute and a Windchill Foundation & PDM attribute, and after they are mapped neither can participate in any other mapping.

Windchill Foundation & PDM determines whether a dependency link is a reference link or a member link by examining the objects linked. For most users, PTC recommends handling reference link and member link definitions identically with a bias to include all or most of the available data in both.

## Effects of Data Changes

As the soft type definitions change by the addition or deletion of attributes and the alteration of constraints on values, the documents that inherit structure from these definitions also change. This can result in mismatches between CAD data and the data expected in Windchill Foundation & PDM.

During CAD document attribute mapping, the software ensures that value constraints in Windchill Foundation & PDM are no more restrictive than value constraints on the mapped CAD data.

## For Searchable Types

By default, the types that are searchable are maintained in the `/com/ptc/windchill/enterprise/search/allSearchTypes` preference. This preference can be updated at the site and organization levels using the Preference Administrator. For information on specifying types in preferences, see [Specifying Windchill Types](#).

If you add a type through the Type Manager, you can also update the preference to make the type searchable. For additional information, see the *Windchill Customizer's Guide*.



# 12

## Administering Indexing

Topic	Page
Overview .....	12-2
About Indexing Rules.....	12-2
Creating and Managing Indexing Rules .....	12-3
About Indexing Policy.....	12-3
About Indexing Lists .....	12-4
Defining a Collection .....	12-5

## Overview

*Indexing* is the process of extracting text strings of attribute names and attribute values from Windchill objects and sending them to a search engine that builds indices optimized for searching. This enables users to efficiently search for data stored in a Windchill database, without having to know anything about the internal object model.

Windchill solutions provide the option of installing RetrievalWare to help with indexing. For additional information about RetrievalWare and its indexing capabilities, see the *Windchill Installation and Configuration Guide -RetrievalWare*.

## About Indexing Rules

Creating an indexing rule from within the Policy Administrator requires you to specify the rule antecedent and the rule consequent. The rule antecedent comprises the following parts:

- The *domain*.
- The *object type*-- determines which rules within an indexing policy apply to a specific object.
- The *life cycle state*, identifies the life cycle phase that an object must be in for a rule to apply.
- The *collections* into which objects are to be entered, when the objects belong to the domain, are of the type, and are in the life cycle state specified by the rule.

For example, you can define a rule specifying that a general document object is to be placed in a Released collection when the object's state becomes Released. Together, the domain indexing rules form the indexing policy for a domain.

The rule consequent is a list of one or more collections.

A *collection* represents a group of related objects that can be searched. It includes indices optimized for searching, as well as references to the actual object locations. Every indexable object carries a list of collections into which it is indexed. The first such list is assigned when the object is created. When the object is deleted, it must be withdrawn from every collection in which it is indexed. Between creation and deletion, the collections in which the object is indexed can change, based on the object's life cycle state and which domain it belongs to. When you create indexing rules, you customize the indexing policy for a domain by specifying which collections an object should move into (or be removed from) when the object moves into a specified life cycle state. From this policy, indexing lists are generated and associated with an object type. To improve performance, indexing lists are cached after they are created.

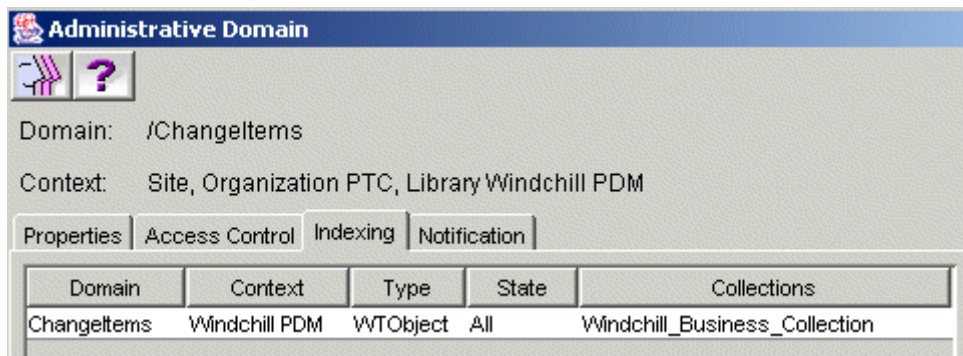


An *indexing rule* identifies a life cycle state for a particular object type and the collections into which the object should be entered, based on the state that it is in. There can be only one state and one object type specified within a single rule. However, each rule can identify multiple collections.

An *object type* specifies a category of objects that share the same attributes and functions. For example, WTDocument is an object type, and instances of that type may be found in some of the domains you have created. Since Windchill domains are hierarchical, indexing rules defined for a domain are inherited by descendent domains. For example, indexing rules defined for the WTDocument object type in all states within the Design domain apply to instances of the type within that domain or any descendent domains. Because Windchill types are also hierarchical, an object inherits rules defined for its ancestor types. Therefore, more than one rule may apply to a given object. For example, a rule that applies to the type WTPart also applies to the type WTSerIALIZEDPart. Additionally, there can be indexing rules specific to WTSerIALIZEDPart.

## Creating and Managing Indexing Rules

Use the Policy Administrator to create and manage indexing rules. Indexing rules are created and managed for a domain that is within a specific context as described in the [Administering Domains and Policies](#) section of the Administering Containers chapter. Open the Policy Administrator from the context of the domain where you want the indexing rules to apply. Select the domain and click **Update**. On the **Administrative Domain** window, click the **Indexing** tab to bring it forward.



Click the help icon located in the upper left hand corner of the window for specific instructions on retrieving, creating, updating, deleting, and reporting on indexing rules.

## About Indexing Policy

This section describes the Windchill implementation of indexing policies.

## Considerations for Establishing Indexing Rules

As you create an indexing policy, you may find it helpful to answer the following questions:

- Which life cycle states are associated with the most changes in business objects? You could decide to create rules that affect objects in the In Work state. Or, you might decide to index objects in the Under Review or the Released state.
- Do certain domains contain objects that are more dynamic than others or objects that users are more likely to search for?
- How do different types of objects undergo change in the system?
- How might rules differ for object types that are versioned (for example, document types) and object types that are not versioned (for example, change objects)?

## About Indexing Lists

Indexing lists are generated for each object type, state, and domain. Objects are associated with the indexing list of the domain, type, and state to which they belong. For example, all WTPart objects in a given state and domain are associated with the same indexing list. The indexing list for WTPart objects is different from the list associated with other types of objects (for example, WTDocument objects) that are in the same state and belong to the same domain.

An indexing list for an object is obtained by combining all rules that apply to that object based on its type, state, and the domain to which it belongs.

A rule is applicable to a given type when the object type referred to in the indexing rule is the type itself or one of its ancestor types. For example, if IncidentReport is a soft type of the type WObject, then a rule that applies to WObject also applies to IncidentReport. In addition, rules are inherited from ancestor domains.

As described above, this type and domain hierarchy means that more than one collection may have to be updated when a specific event occurs. Consider the following example:

	Domain	Context	Type	State	Collections
Rule 1:	/	Site	IncidentReport	ALL	Current, Assignments
Rule 2:	/Publications	Windchill PDM	WObject	ALL	Assignments
Rule 2:	/Publications	Windchill PDM	IncidentReport	ALL	CustomerX

The combination of these rules produces the following index list entry for IncidentReport in /Publications:

Domain	Context	Type	State	Collection(s)
/Publications	Windchill PDM	IncidentReport	ALL	Current, Assignments, CustomerX

As this list entry specifies, the Current, Assignments, and CustomerX collections must be updated whenever an incident report is created or modified that is associated with /Publications domain, which is in the Windchill PDM context, regardless of the life cycle state it enters.

When you have defined the indexing rules for your domains, all of the objects with the same domain, type, and state combination for which you have created a rule share an indexing list.

This association between the indexing list and the object is preserved.

## Defining a Collection

*Indexing* is the process of extracting text strings of attribute names and attribute values from Windchill objects and sending them to a search engine that builds index *collections* optimized for searching. This enables users to efficiently search for data stored in a Windchill database without having to know anything about the internal object model.

Windchill collections are defined in the wt.properties file. Each collection has properties that define the collection. For more information, see the *Windchill Installation and Configuration Guide - RetrievalWare*.



# 13

## Administering Notifications

Topic	Page
Overview .....	13-2
About Notification Rules.....	13-2
Creating and Managing Notification Rules.....	13-3
Notification Lists.....	13-4
Folder Subscription Notifications .....	13-5
About Notification.....	13-6

## Overview

A notification policy determines who is notified when events of interest happen in the system.

**Note:** Notifications are also generated as part of workflow processing. Workflow notifications can be more task specific. For more information, see the chapter entitled [Administering Workflow Processes](#).

## About Notification Rules

When you create notification rules from within the Policy Administrator, you specify who is to be informed when a given system event occurs within the context of a specific object. You can construct a rule for a domain, an object type, and an event posted by a manager type (ownership, locking, versioning, life cycle, and so on). The set of notification rules for a domain constitutes the notification policy for that domain.

Actual notification is accomplished by sending an e-mail message to the users on the notification list. This message identifies the event and the object.

**Note:** A user must have Read permission to an object to receive notification of an event applied to that object.

Creating a notification rule requires you to specify the rule antecedent and the rule consequent. The rule antecedent comprises the following parts:

- The domain
- The object type, which determines which rules within a notification policy apply to a specific object
- The system event type (for example, Checkout).

The rule consequent is a list of one or more principals.

A *notification rule* for a domain identifies a system event of interest for a particular object type and determines which users, groups, and organizations should be notified when one of those events occurs, for example, checkout. There can be only one event type and one object type specified within a single rule. However, each rule can identify multiple principals.

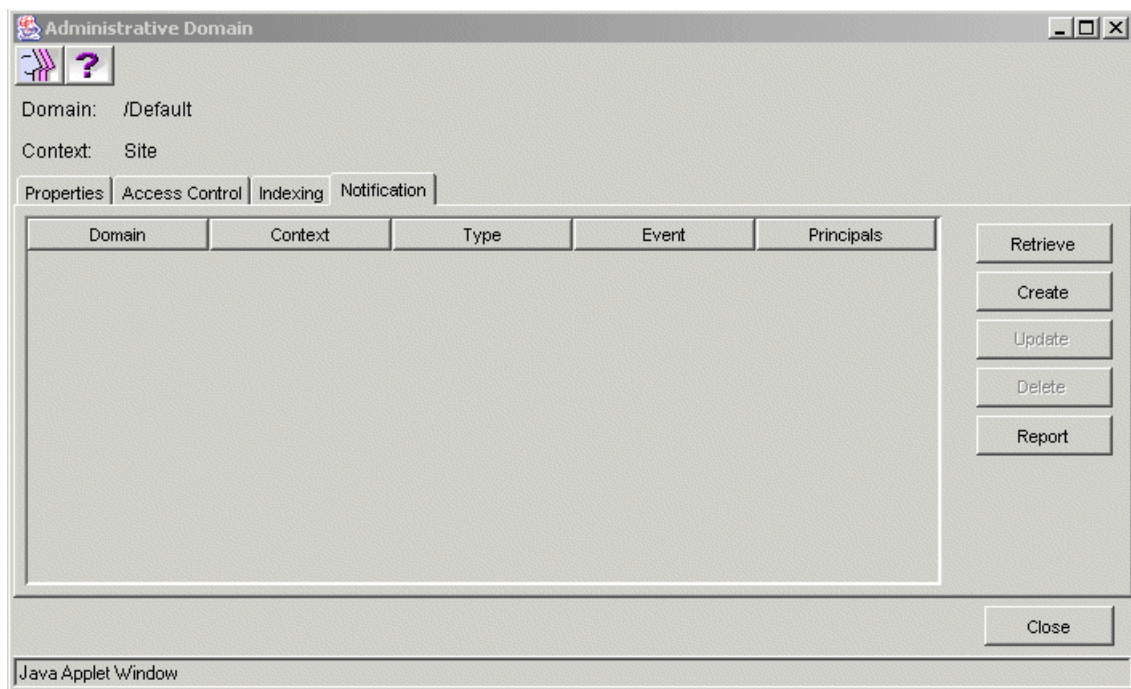
An *object type* specifies a category of objects that share the same attributes and functions. For example, WTDocument is an object type, and instances of that type may be found in some of the domains you have created. Since Windchill domains are hierarchical, notification rules defined for a domain are inherited by descendent domains. For example, notification rules defined for the WTDocument object type in all states within the Design domain apply to instances of the type within that domain or any descendent domains.

Because Windchill types are also hierarchical, an object inherits rules defined for its ancestor types. Therefore, more than one rule may apply to a given object. For example, a rule that applies to the type WTPart also applies to the type WTSerIALIZEDPart. Additionally, there can be notification rules specific to WTSerIALIZEDPart.

A *principal* is either an individual user, a group, or an organization. Typically, you should define notification rules for groups. Dealing with groups helps reduce administrative overhead by making it possible to apply rules to multiple users at the same time, enabling mass mailings for notification. Sometimes, however, you need to create rules specific to an individual user or to an entire organization.

## Creating and Managing Notification Rules

Use the Policy Administrator to create and manage notification rules. Notification rules are created and managed for a domain within a specific context as described in the [Administering Domains and Policies](#) section of the Administering Containers chapter. Open the Policy Administrator from the context of the domain where you want the notification rules to apply. Select the domain and click **Update**. In the **Administrative Domain** window, click the **Notification** tab to bring it forward.



Click the help icon located in the upper right hand corner of the window for specific instructions on retrieving, creating, updating, deleting, and reporting on notification rules.

## Notification Lists

Notification lists are generated from the notification rules for a domain and its ancestor domains. These lists are the basic mechanism for initiating user notification when an event occurs within the context of a specific object within the domain. For performance reasons, once lists are constructed, they are kept in a cache.

Notification lists are generated for every combination of type, event, and domain. A notification list for an object that is the target of an event is obtained by combining all rules that apply to the event, the domain to which the object belongs, and the type of object. For example, the same notification list applies to the Create event for all WTDocument objects within a given domain. In addition, this notification list can be different from the list associated with WTPart objects, even if the objects belong to the same domain.

To make this definition precise, it is necessary to describe how rules are combined and when a rule applies to a type.

A rule is applicable to a given type when the object type referred to in the notification rule is the type itself or one of its ancestor types. For example, a rule that applies to the WLObject type also applies to the IncidentReport type if IncidentReport is a soft type of WLObject.

This type of hierarchy, in addition to the domain hierarchy, means that more than one set of principals may need to be notified when a specific event is applied to an object.

For example, consider the combination of the following rules:

	Domain	Context	Type	Event	Principal(s)
Rule 1:	/	Site	WLObject	Create	Marketing, Engineers
Rule 2:	/Publications	Windchill PDM	WLObject	ALL	Amanda.Smith
Rule 3:	/Publications	Windchill PDM	IncidentReport	Create	Support

The combination of these rules produces the following notification list entry for IncidentReports created in the /Publications domain:

Domain	Context	Type	Event	Principal(s)
/Publications	Windchill PDM	IncidentReport	Create	Marketing, Engineers, Support, Amanda.Smith



As this list entry specifies, all members of the Marketing, Engineers, and Support groups, and also user Amanda Smith, must be notified whenever an incident report is created in the /Publications domain. This result is based on the following application of the notification rules:

1. Members of the Support group are to be notified when an incident report is created in the /Publications domain.
2. Members of the Marketing and Engineers groups are to be notified when an object of type WObject is created in the Root domain. IncidentReport is a soft type of WObject, and the Root domain is an ancestor of /Publications, so this rule also applies to the incident report created in the /Publications domain.
3. User Amanda.Smith is to be notified of all events that occur to an object of type WObject in the /Publications domain, which includes any event that occurs to an object of the soft type IncidentReport in the /Publications domain.

When you have defined the notification rules for your domains, a single notification list is shared by all events with the same event type, target object type, and target object domain. When an event occurs to an object, the notification list associated with the object is retrieved, and the appropriate principals are notified of the event.

## Folder Subscription Notifications

Windchill PDMLink and Windchill ProjectLink provide the capability for users to subscribe to folders. In conjunction with this capability, there is a set of out-of-the-box notification templates that are used.

You can customize the content of the folder subscription notifications by modifying the following files:

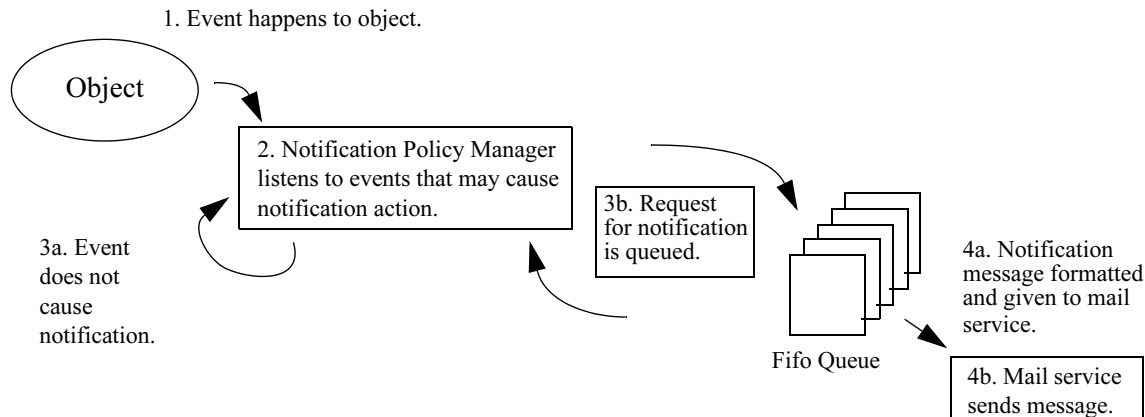
checkInInFolderNotification.html  
createInFolderNotification.html  
multiCheckInInFolderNotification.html  
multiCreateInFolderNotification.html  
multiStateChangeInFolderNotification.html  
stateChangeInFolderNotification.html

These files are available under `<Windchill>/codebase/templates/folder/`, where `<Windchill>` represents the Windchill installation directory.

**Note:** Modifying files is a customization; be sure to follow PTC recommendations on modifying files supplied by PTC. For details on customizing HTML template files and best practices when modifying files, see the *Windchill Customizer's Guide*.

## About Notification

Before the notification process can begin, the Notification Policy Manager subscribes to all system events that are specified in the `notify.properties` file. The following figure represents an overview of the notification process:



- An event happens to an object (step 1). If it is an event for which the Notification Policy Manager is listening, the event and the object are posted to the manager (step 2).
- The Notification Policy Manager checks to determine whether the event triggers a notification action. In many cases it does not, and can be ignored (step 3a). (For example, if the object does not belong to a type subject to notification rules, or there is no list for the domain/type/event.)

In some cases, there is a notification list for the domain/type/event. When a list exists, the manager does not try to send the notification immediately. Rather, it queues the notification request for deferred processing in a FIFO queue (step 3b).

- Later, the queued requests are asynchronously executed. These requests translate into calls, so that the notification message is formatted and then mailed using the mail service (steps 4a and 4b).

For more information about background processing queues and their maintenance, see the chapter titled *Configuring and Administering Background Queues* in the *Windchill System Administrator's Guide*.

# 14

## Administering Life Cycles

This chapter provides information about life cycles and the Life Cycle Administrator.

Topic	Page
Overview .....	14-2
The Windchill Life Cycle Model .....	14-2
Managing Life Cycle Processes .....	14-4
Promotion Process .....	14-9
Basic and Advanced Life Cycles.....	14-14
Out-of-the-Box Life Cycle Templates .....	14-15
Life Cycle Iteration .....	14-18
Viewing Iteration History.....	14-19
Creating a Life Cycle Template .....	14-19
Overriding and Reassigning Life Cycle and Team Templates.....	14-20
Life Cycle Properties.....	14-22
Defining Life Cycle Phases.....	14-24
Predefined Life Cycle States .....	14-42
Import and Export .....	14-42
Access to Life Cycle Administration .....	14-45
Best Practices for Windchill PDMLink and Windchill ProjectLink .....	14-45

## Overview

Business information and business objects generally become more mature and reliable over time. *Life cycles* define the way in which these business objects mature, providing a model for a product's commercialization process.

How you access the Life Cycle Administrator is determined by your Windchill solution:

- From Windchill Foundation & PDM, you can access the Life Cycle Administrator by clicking the **Process Administrator** link that is on the **Business Administration** home page. Then click **Life Cycle Administrator**.
- From Windchill PDMLink, you can access the Life Cycle Administrator by clicking the **Life Cycle Administrator** link on the **Utilities** pages that are under the **Site**, **Organization**, **Product**, and **Library** tabs. The **Life Cycle Administrator** link from the **Site** tab provides you (as the site administrator) with unrestricted access to all life cycles. The link from the **Organization** tab provides access to only those templates that are in the organization context that is active when you launch the Life Cycle Administrator. The link from the **Product** and **Library** tabs provides access to only those templates that are in the context that is active when you launch the Life Cycle Administrator.

From the **Product** and **Library** tabs, the Life Cycle Administrator shows all life cycle templates from that context, the organization context, and the site context. From the **Organization** tab, you see life cycle templates from the organization context and the site context. From the **Site** tab, you see only site-level life cycle templates.

- From Windchill ProjectLink, you can access the Life Cycle Administrator from the **Utilities** pages that are under the **Site** and **Organization** tabs.

This chapter describes how to define a life cycle using the Life Cycle Administrator.

## The Windchill Life Cycle Model

A Windchill life cycle is an automated, graphical model, employing phases and gates, used to manage business objects as they progress from conceptualization through obsolescence.

While an item is in a specific life cycle phase, certain business rules apply, such as access control rules defined for that phase.

When created, an object modeled to be life cycle-managed enters a life cycle phase, where it is assigned an initial state, and is then associated with the initial phase of its life cycle.

## Windchill Solutions

In Windchill PDMLink and Windchill ProjectLink, object initialization rules are used to designate to which life cycle an object type is assigned when a user creates a new object of that type. Workflows associated with an object, life cycle manage the maturing processes from state to state through workflow activity templates. When new versions of an object are created through the Revise action, the team and the life cycle is assigned to the object following the same rules as when the object was created, and the life cycle state is set to the first state of the life cycle.

Depending on several business environment considerations (impact of change, development team, maturity of object, type of object, and so forth), it may be important to tailor the life cycle to include certain business operations and to define how these operations are executed. Transition rules tailor how Change Management, Revise, Promotion, and Set State operations should behave and which workflow processes you should use to manage these processes.

## Windchill Foundation & PDM

The Windchill life cycle implementation also provides interface points for specification of workflow processes at each phase and gate. As a result of the automatic initiation of workflow processes, workflow tasks are assigned to participating users and are listed in the worklist of those users. A default workflow process sends a notification message to the Submitter role player, and is associated with entry into each life cycle phase. Another default workflow initiates Review and Promote activities when an item enters a life cycle gate. When an item enters a life cycle gate, it is awaiting promotion.

When an item is considered ready for promotion to its next life cycle phase, it reaches a decision point (gate) for the phase. If the item is found ready to progress, the Promoter role player moves it to a new phase through an explicit promote action, executed on the task for promoting the item in the worklist/assignment table or a set state robot can automatically do it. Promote assigns a new state to the item and associates it with its next phase, where new business rules may apply.

The end user can promote items by using the Promote action. For more information, see the online help on the Promotion Request window.

## Additional Life Cycle Capabilities

In Windchill, the life cycle is the core capability to manage the maturity of an item (referred to as a life cycle state) as well as access control policies to manage or view the item for a set of roles in the system. The sophistication of business processes to manage an item depends upon the maturity of the item. State is an enterprise object, and its meaning is applied regardless of the life cycle by which a given item was processed. For example, if an access control rule applies to a Requirements item in the Under Review state, the rule is applicable to all Requirements items in that state, even if they arrived at the state through different life cycles. However, each phase of a life cycle must be associated with a life cycle state chosen from among all states defined in the system.

**Note:** If the domain policy denies access to a user and if an ad hoc policy grants the user access, then the ad-hoc policy is used, that is, the user is granted access. Ad hoc policies overrule and can provide access to a user where the domain policy denied it.

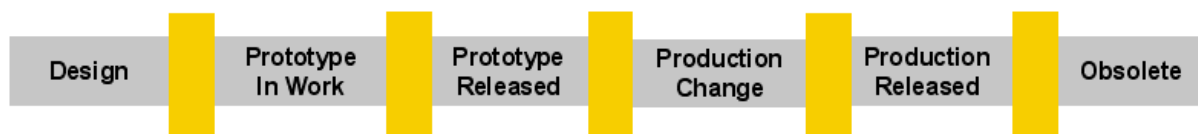
As a life cycle administrator, you can create a variety of life cycles. These life cycles, which are stored in the System cabinet (for Windchill Foundation & PDM) or System folder (for Windchill PDMLink and Windchill ProjectLink), define the phases and gates associated with various business objects. For the life cycle of each item, you can define the transitions through which the item must move, and the behavior associated with the item while it is in a specific state.

Windchill provides a Default life cycle, with many predefined states, such as In Work, Under Review, and Released.

Before you begin creating life cycles, you should understand life cycle iteration and life cycle roles, as described in this guide.

## Managing Life Cycle Processes

In Windchill, the life cycle is the core capability to manage an item's maturity (referred to as a life cycle state) as well as access control policies to manage or view the item for a set of roles in the system. Each item type in a Windchill system (such as a part, specification, drawing, and so forth) can have unique sets of life cycle states and access control policies. Object initiation rules are used to designate which life cycle an item type is assigned when a user creates a new item of that type. Workflows (associated with an item) life cycle manage the maturing processes from state to state through workflow activity templates. By default, when new versions of an item are created through the Revise action, the life cycle state is set to the first state of the life cycle. A life cycle administrator can specify the target state of the Revise action.

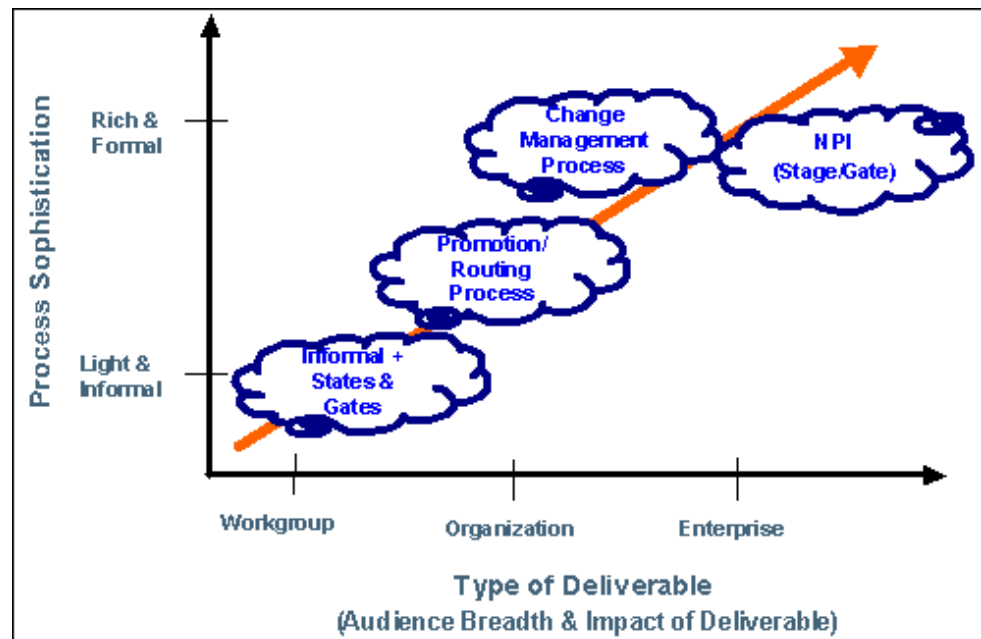


Depending on several business environment considerations (impact of change, development team, maturity of item, type of item, and so forth), it may be important to tailor the life cycle to include certain business operations and to define how these operations are executed.

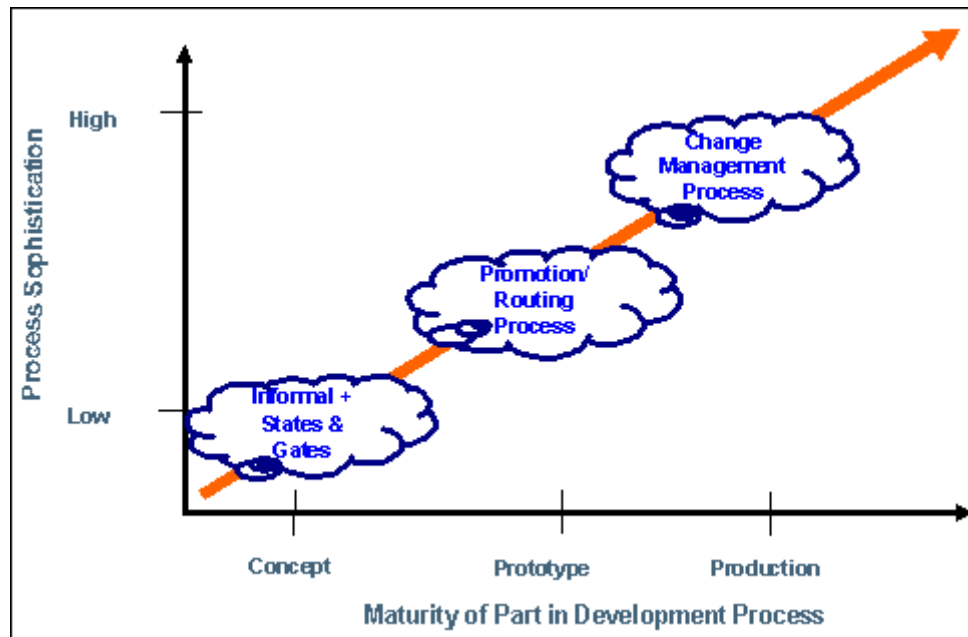
For example, a development team in the automotive division of a company can have different review processes for releasing a design specification than the industrial products division. The automotive division may require a prototype to be created for a new product before the first production versions of the product are released. Formal change processes are required to update these prototype designs. After a part has been initially released to production, new versions of the design

are designated as Production Change, and they must undergo a formal change process.

The following figure is an example of the variety of business processes that can be used with a Windchill system to mature different types of product development deliverables. The development team processes for the review and release of a design specification can differ from those required for the release of a new configuration of the product.



The next figure illustrates a case when a combination of the business processes is required for some product development deliverable items, such as parts, as they move from a concept, through prototyping, and into production.



Windchill PDMLink supports the following business processes that you can use to manage the life cycle of a part, document, or CAD document. You can use transition rules to control when in the life cycle this operation is available. For example:

- Set State - Enables you to informally set the life cycle state of an item.
- Promote - Enables you to set the state of one or more items to a new life cycle state as part of a review process.
- Change - Enables you to execute a change order for a product development item.
- Revise - Enables you to create a new version of the item.

For more information on transition rules, see [Transition Rules](#).

## Multiple Selection of Processes

Windchill PDMLink has two out-of-the-box workflows from the Promote transition. See [Promotion Process](#) for a brief description of these workflows.

These two workflow processes are presented for every promotion request. However preferences may be configured to provide zero or more workflows to reflect business practices.



All of the transitions preferences are located under the same node path of:

```
/transitions/PROMOTE
```

Within this preferences node, there are a series of keys that are the constant name for the life cycle state that is the desired terminal state. For example, if a transition is created for the state Changed, its constant is CHANGED. The fully-qualified preference key is:

```
/transitions/PROMOTE/CHANGED
```

The value of the preference is a delimited list of workflow process template names. The delimiting character is ~ (tilda).

Here is an example of the above preference:

```
/transitions/PROMOTE/CHANGED=Review with Approval~Fast Promote
```

In this case, two processes are defined, Review with Approval and Fast Promote, delimited by the ~ character.

Specifying a preference value of "" (empty), indicates that the state transition does not have any process (similar to the set state action); the promotion targets are automatically promoted when a promotion request is created.

The preference searches the current container (Product or Library) defined by the promotion request and also searches the owning organization container and site container. A preference value defined at a lower-level container overrides the higher level container, provided that the higher-level container preference Override Allowed=true.

Out-of-the-box, a preference is loaded into the site container that specifies the two default transition processes. Initially, this is set to two processes; however, you can modify this to change or add additional processes, as needed.

Container: Site

Pref Node: /transitions/PROMOTE

Pref Key: \$DEFAULT

Pref Value: Promotion Request Approval Process Workflow and Promotion Request Review Process Workflow.

Override Allowed: true

```
/transitions/PROMOTE/$DEFAULT= Promotion Request Approval Process~Promotion Request Review Process
```

The default preference is added to the load-set for Windchill PDMLink with the following definition:

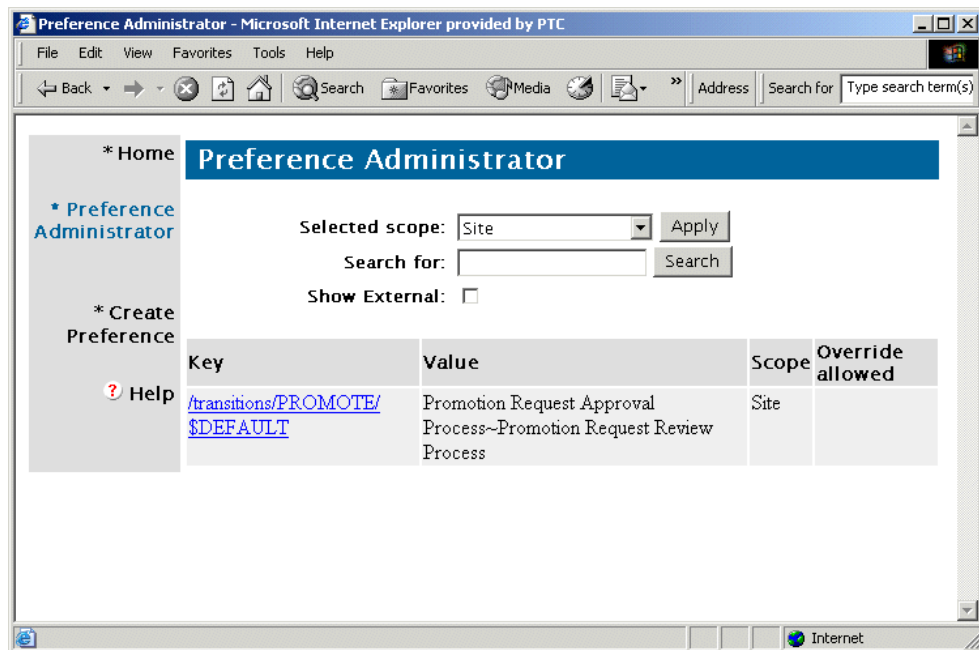
```
<csvPrefEntry handler="wt.prefs.LoadPrefEntry.createPrefEntry">
  <csvname>$DEFAULT</csvname>
  <csvvalue> Promotion Request Approval Process~Promotion Request
Review Process </csvvalue>
  <csvnode>/transitions/PROMOTE</csvnode>
  <csvuser></csvuser>
  <csvcontext>@CONTAINER:/</csvcontext>
</csvPrefEntry>
```

The preferences can be created in following ways:

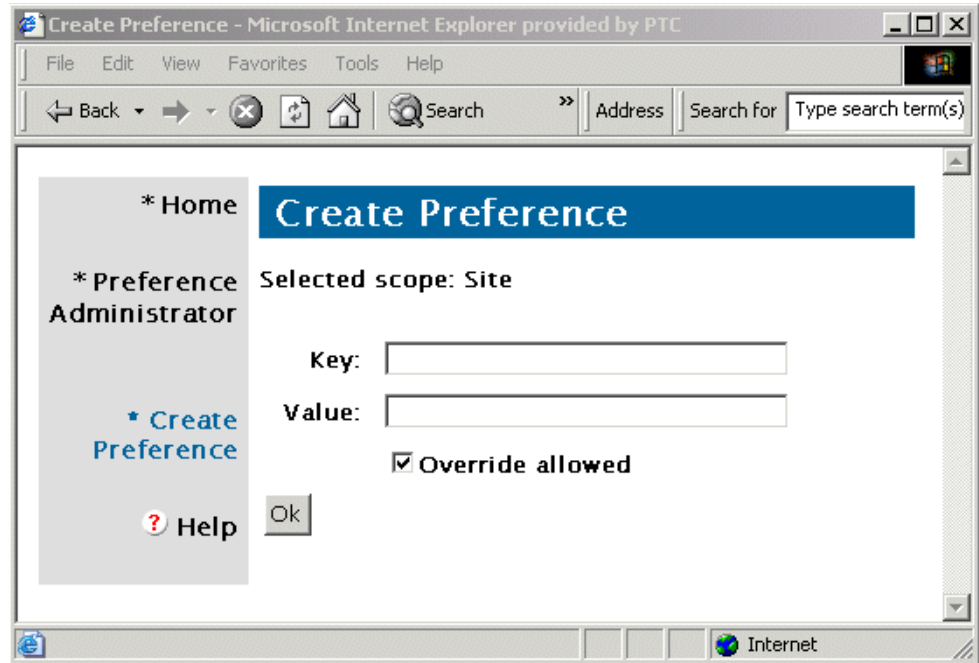
- a loadfile (this is how the default preference is loaded)
- the Preference Administrator

The loadfile format is documented as part of the loader formats, however the default entry shown above illustrates the format.

You can use the Preference Administrator to create the preferences for your transitions. Clear the **Show External** check box and type **transitions** in the **Search for** field. Your search should find one preference, the default.



You can create a preference by clicking the **\*Create Preference** link on the left of your window. For example, if you wanted to provide a workflow process for items that are targeted for promotion to the state obsolete, you would establish a preference for the life cycle state constant, OBSOLETE:



For **Key**, enter

/transitions/PROMOTE/OBSOLETE

for **Value**, enter the name of the process you want to use for promotion to obsolete. For example, maybe you want to promote to obsolete using the Promotion Request Approval Process. If you do not want a process (that is, you want to auto-promote) then simply do not fill in a value. If you have multiple processes, they are separated with a ~. The state you are promoting to should be the constant state representation (OBSOLETE not Obsolete, since this latter is a localized display name only).

The **Override allowed** check box means that this transition process could be redefined at a lower level (such as, the organization level or a product level).

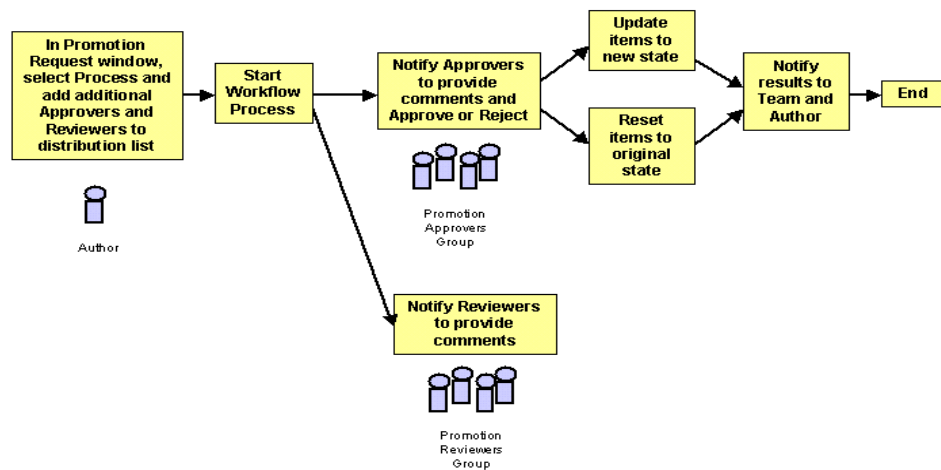
## Promotion Process

For the promotion process, you can use two standard review and approve workflow processes. With the enhancements to life cycle management, these two processes can be used to manage the development of CAD documents, parts, and documents.

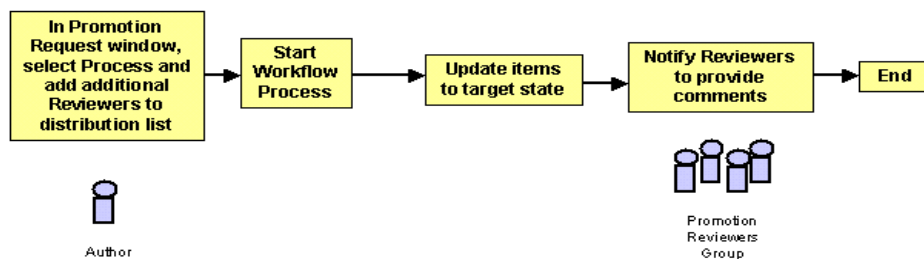
The promotion request window guides a user requesting a promotion of one or more CAD documents, parts, and documents through the promotion process. The user can add dependents of parts and CAD documents to the request through this process. All items requested for promotion must be set to the same target life cycle state. If the items requested for promotion do not share any life cycle states, no promotion can be completed.

Two standard workflow processes for promotion are as follows:

- **Promotion Request Approval Process** - A process to notify a set of Approvers to approve or reject the Promotion Request and invite a list of Reviewers to provide comments to the Promotion Request. If all Approvers approve the request, the items in the promotion request are updated to the target life cycle state. Otherwise, these items are returned to their original life cycle state.



- **Promotion Request Review Process** - A process to automatically set the state of promotion request items to the target life cycle state and invite a list of Reviewers to provide comments on the promotion request.



A list of Approvers who must approve the promotion request is defined by the Promotion Approvers group for a product or library. A list of Reviewers, who are invited to provide comments to the items under review, but do not approve the request, is defined by the Promotion Reviewers group for that product or library. The user submitting the promotion request can also add (in the last step of the **Promotion Request** window) Approvers or Reviewers for each specific review through a list of users and groups for the product and library.

When a promotion request is created for one or more CAD documents, parts, and documents, the list of specific iterations of these items is maintained in the baseline for this promotion request. Additionally, for a set of CAD documents on a promotion request, a set of viewables is generated for this baseline set, and it can be viewed directly from the promotion request task or from the promotion request form.

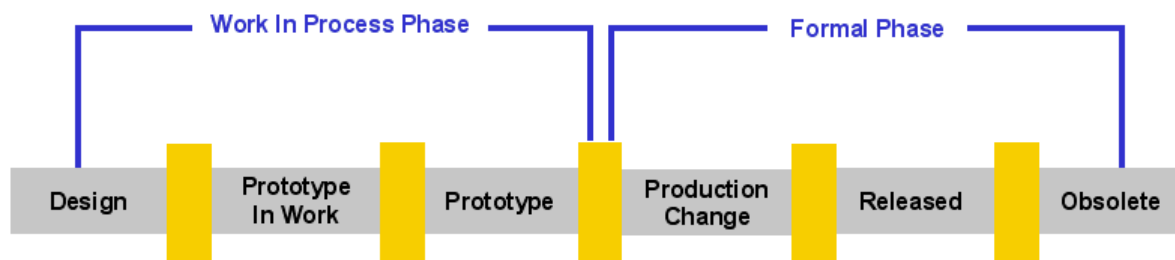
**Note:** If you want to disable viewables from being generated in the promotion process, set the following property to false; `publish.promote.enabled=false`.

## Transition Rules

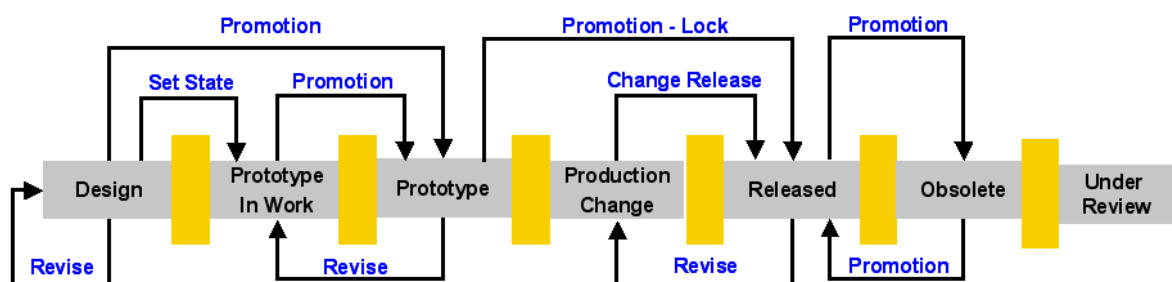
Life cycle transition rules provide capabilities for defining some behaviors of an item. These rules apply to the following actions in Windchill PDMLink:

- **Change** - Enables you to execute a change order for a product development item. In particular, the Change transition identifies the state to which an item is released from a given state.
- **Lock** - Enables you to identify a temporary state in which an item resides only while the promotion process is being executed: at the end of the process, the item either promoted to a specified state (see description of Promote transition) or returned to its original state. Typically, the state identified by a Lock transition specifies restricted access for modification. Lock transitions are always optional.
- **Promote** - Enables you to set the state of one or more items to a new life cycle state as part of a review process. Transition rules can be used to control when in the life cycle this operation is available.
- **Revise** - Enables you to specify the initial life cycle state of a new version of the item. Transition rules can be set up to control the life cycle state of the new version based on the state of the version being revised. Additionally, rules can be set up for a life cycle state to use a new version sequence when the Revise action occurs.
- **Set State** - Enables you to informally set the life cycle state of an item. Set state is an extension of the administrative set state function to users without administrative privilege. The user must have create permission for the promotion request in the context in question.

Using a set of transition rules, you can define appropriate sets of business processes for the stage of development for that item. For example, the processes for developing a part can be informal in the early stages of development, but will require more formal processes if the part is introduced in production.



Using transition rules, you can set up a variety of business processes to be available for each type of item in a Windchill PDMLink system. The next diagram is an example of how these rules can be applied to a life cycle.



Some business practices require more formal processes, such as review and approval cycles. The Promotion and Change transition rules have associated workflows for that purpose. By default, the promotion transition rule has two workflow processes defined. Depending on your business practices, zero or more workflow processes can be configured for use with a specific promotion process.

For some promotion review processes, it is important to freeze or lock the promotion items, so that no changes are made during the review process. For the example life cycle above, the lock transition can be added to set the items to the Under Review state during a promotion process.

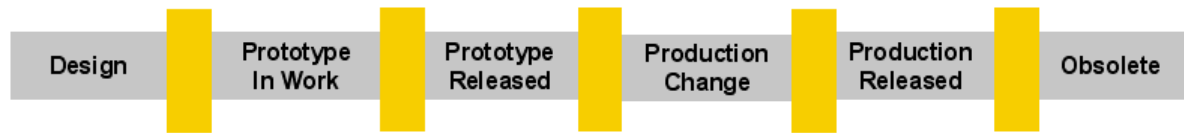
For change management processes, the new life cycle state for approved change notice is now defined by the Change transition and is no longer fixed to one specific life cycle state.

These transition rules are defined for each life cycle.

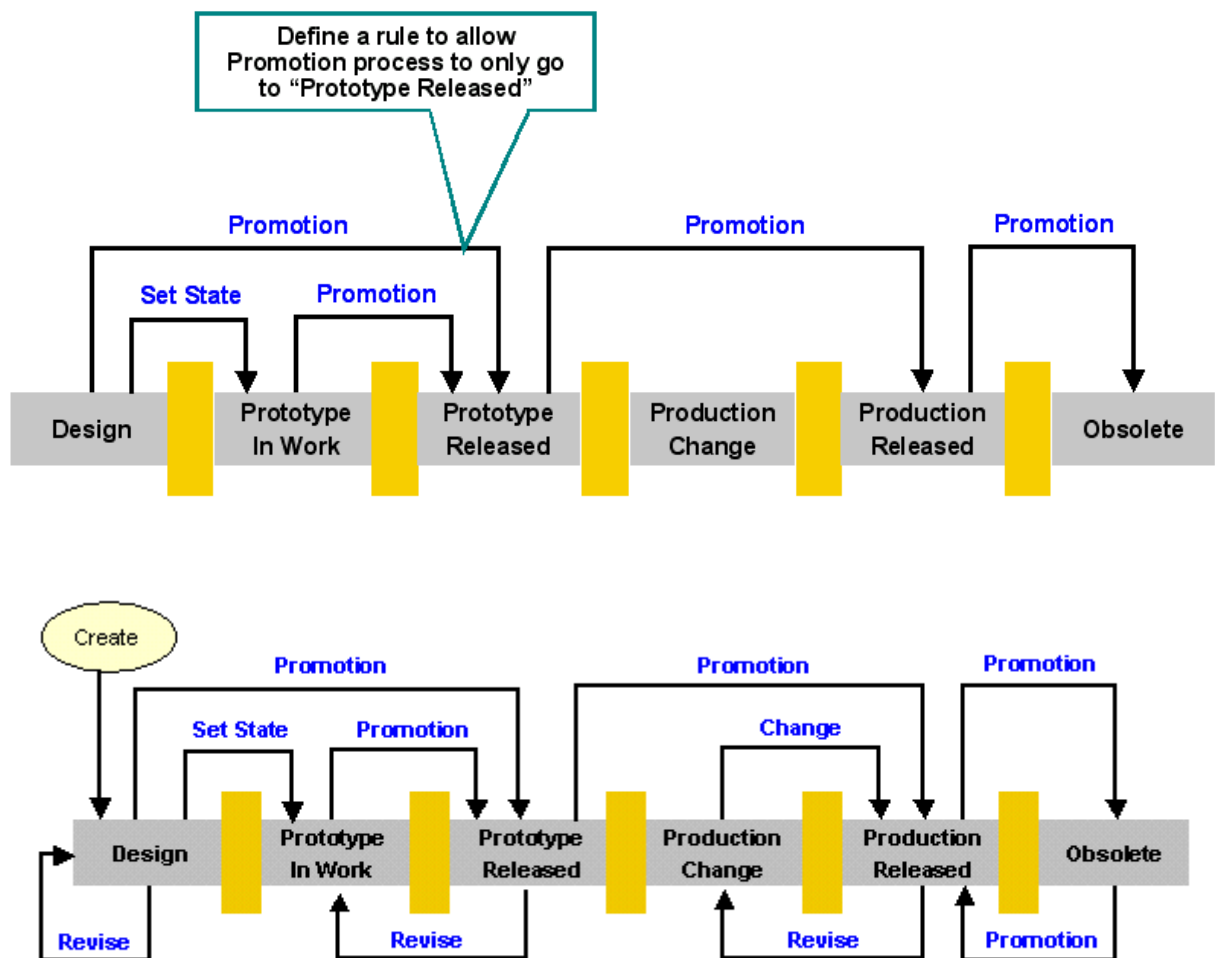
Transition rules may also identify whether the items are locked during Promotion reviews. The item resides temporarily in the locked state while the promotion

process executes. If a lock transition is defined for a particular state and an item using the life cycle is designated for promotion on a promotion request, the workflow of the promotion request sets the state of the item into the state identified with the lock transition.

Suppose your life cycle model looked like this:



You determine that you want to establish transition rules:



## Basic and Advanced Life Cycles

You can designate whether the life cycle includes certain aspects of a life cycle definition. This is especially useful for team definitions and the use of workflow processes in Windchill ProjectLink and Windchill PDMLink.

Two life cycle types are supported: basic and advanced.

**Note:** PTC recommends that you use the basic life cycle type and their associated concepts for managing workflow processes, teams, promotion criteria and access control policies in Windchill ProjectLink and Windchill PDMLink. Basic life cycles address performance and scalability requirements.

Any particular container could have a mix of basic and advanced life cycles associated with objects. Ideally, you associate an object with an advanced life cycle, just in time when needed, such as when you route an object in Windchill ProjectLink.

Any time you need to manage access via life cycle team roles and to drive an object through its life cycle states by a workflow process, you need an advanced life cycle.

The basic life cycle type is designed to use the context team defined in a Windchill ProjectLink project, a Windchill PDMLink product, or Windchill PDMLink library. A basic life cycle does not include workflow process definitions and relies on the workflow process definitions with change management, routing, or promotion processes. Use this type of life cycle for most Windchill ProjectLink or Windchill PDMLink objects, because the basic life cycle provides improved performance during the creation and revision of objects.

When you create an item, such as a part or document, and select a team template and a basic life cycle; the selection of the team has no effect. Basic life cycles do not utilize teams or workflows.

The advanced life cycle type provides the ability for a separate team to be assigned for instances of this object type and to include workflow process definitions within the life cycle states to manage the maturing of the object type. Use this type for managing objects that have company-specific business processes, special access control requirements, or additional flexibility in managing these types of objects.

**Note:** There are certain major user-visible differences between basic or advanced life cycle. The main differences are in the setup of the team definitions for an item, in the workflow processes for managed items, and in the system performance when creating and revising items. Using basic life cycles provides improved performance for core creation and revise operations.



# Out-of-the-Box Life Cycle Templates

The following out-of-the-box life cycle templates are identified as basic or advanced.

## Windchill Foundation & PDM

All out-of-the-box life cycle templates for Windchill Foundation & PDM are advanced life cycle templates:

- Default
- Analysis Activity
- Change Activity
- Change Investigation
- Change Issue
- Change Proposal
- Change Request
- Change Order
- ReplicationPublish
- ReplicationReceive
- ReplicationSender
- ReplicationBaseline

## Windchill PDMLink

The following Windchill PDMLink out-of-the-box life cycle template are basic life cycles:

- Basic
- Library Development



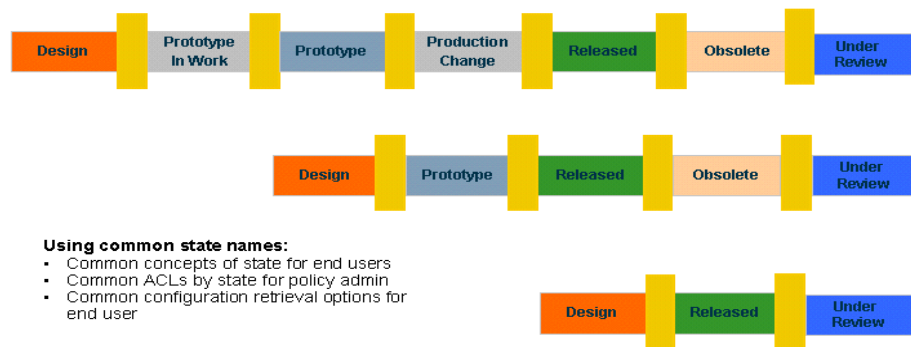
- One Phase Development



- Two Phase Development



The following figure shows the states that the three life cycle templates have in common.



The following Windchill PDMLink life cycle templates are advanced:

- Change Activity Life Cycle
- Change Proposal Life Cycle
- Change Notice Life Cycle
- Change Request Life Cycle
- Problem Report Life Cycle
- Promotion Request

## Using the Product Design Template

The out-of-the-box Product Design template that is available in Windchill PDMLink establishes life cycle templates that are complete with transition rules and workflow processes for business objects; it uses the One Phase Development and Two Phase Development templates. Remember, the Product Design template should be used as an example. Your site should determine how to use transition rules and workflow processes, and set up default life cycles for the site, not just at a particular product level. For additional details on the Product Design template, see [Product Design Template](#) in the Administering Products and Libraries chapter.

### Access Control for Parts Established Through the Product Design Template

The Product Design template also establishes access control policy rules for business objects. The following table shows the access control lists (ACLs) for parts as established in the Product Design template.

Group	Design	Prototype In Work	Prototype	Production Change	Released	Obsolete	Under Review
Confirmed Members	No Access	No Access	Read	No Access	Read	Read	Read
Marketing	No Access	Read	Read	Read	Read	Read	Read
Procurement Engineer	No Access	Read	Read	Read	Read	Read	Read
Quality Engineer	No Access	Read	Read	Read	Read	Read	Read
Designer	Read Create Modify Delete Revise NVV <sup>1</sup>	Read Modify Delete Revise NVV	Read Modify Delete Revise NVV	Read Modify Delete Revise NVV	Read Revise NVV	Read	Read
Manufacturing Engineer	Read	Read	Read	Read	Read	Read	Read
Design Team Leader	Read Create Modify Delete Revise NVV	Read Modify Delete Revise NVV	Read Modify Delete Revise NVV	Read Modify Delete Revise NVV	Read Revise NVV	Read	Read
Product Manager	Full	Full	Full	Full	Full	Full	Full

1. NVV is New View Version.

For additional details about the access control settings in the Product Design template, see [Product Design Template](#) in the Administering Products and Libraries chapter.

## Windchill ProjectLink

The following out-of-the-box Windchill ProjectLink life cycle templates are basic:

- Approval
- Basic
- Notify
- Release
- Review

The following Windchill ProjectLink life cycle templates are advanced:

- Approval Routing
- Notify Routing
- Release Routing
- Review Routing
- Two Level Approval Routing

## Life Cycle Iteration

Working with life cycles is an iterative process.

Like version-controlled objects, *iterated objects* are checked in to and out of shared locations; however, unlike version-controlled objects, they cannot be revised. Instead, any change to an object creates a new and separate iteration when it is checked in. Earlier iterations, which may still be in use, are unchanged and unaffected by the new iteration. Only the latest iteration is available for new uses.

To make changes to a life cycle template, you must check out a copy. (Click **Update** on the **Life Cycle Administrator** page to check out a copy of the selected life cycle.) While it is checked out, no one else can check out a copy, but the original can be viewed or selected to manage an item. When you have completed changes to the checked-out copy, you must save it and check it in, so it is available to others. It then becomes the latest iteration. Items that are being managed by an earlier iteration continue to be managed by that iteration. They are not affected by the newer iteration.

## Testing an Updated Life Cycle

Under normal circumstances, when you select a life cycle to manage an item, the latest iteration in the shared location is used. However, if you have the life cycle checked out and stored in your personal cabinet, the checked-out copy is used. This makes it possible to test a life cycle before checking it in.

To update and test a life cycle, follow this procedure:

1. Check out a copy of the life cycle.
2. Update the copy, and save it to your personal cabinet.
3. Create a life cycle-managed item that uses that life cycle.

The updated copy of the life cycle in your personal cabinet is used, rather than the current, checked-out iteration in the shared location.

Before you can check in the updated life cycle or undo the check out, you must delete the life cycle-managed item.

If you attempt to check in the updated life cycle, or undo the checkout while the item is still managed by the updated life cycle, an error message, similar to the following, is displayed:

```
The iteration of the <life cycle name> is used in <number> places.  
The following uses must be removed before completing the checkin...
```

## Viewing Iteration History

To view the iteration history, select the item, then **Iteration History** on the **Life Cycle Administrator** navigation panel. A list of all the life cycle iterations appear with the date and time of last modification, and the name of the modifier. Select any iteration, and click **View** to view the life cycle.

## Creating a Life Cycle Template

The **Life Cycle Administrator** page displays a list of existing life cycle templates and their locations. Using the buttons on this page you can create, update, view, and delete life cycles. You can import and export life cycles among other functions.

**Note:** To sort the list of life cycle templates in a descending order, hold the shift key down when you click on the column header. To sort ascending, click on the column header without holding the shift key down.

To create a life cycle, click **Create** to open the **Create Life Cycle** window. To update a life cycle, click **Update** to open the **Update Life Cycle** window. Update uses essentially the same procedures as create, but you change information, rather than creating it. When you update a life cycle, it is automatically checked out.

You must have the necessary access permissions to create or update a life cycle. If you do not have the required permissions, the **Create** and **Update** buttons are enabled, but you get an error message if you try the operation.

Click **Help** on the **Life Cycle Administrator** page for a description of the buttons and their functions.

When you create a life cycle, you define the following:

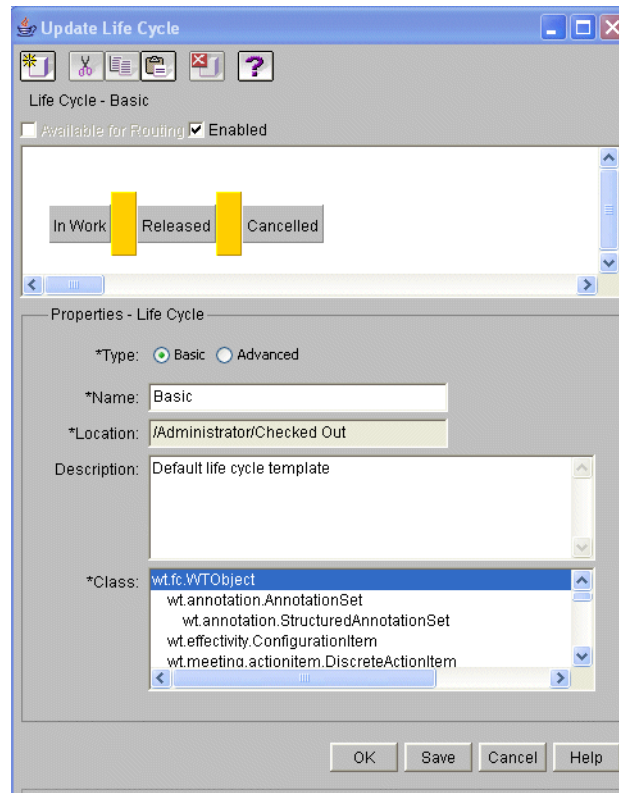
- The properties of the life cycle, including name, location, the object types to which the life cycle applies, and whether the life cycle is enabled. (See [Life Cycle Properties](#).)
- Phases and gates defining the life cycle. (See [Defining Life Cycle Phases](#).)
- Roles, such as Submitter or Promoter, for each life cycle phase. These roles are mapped to users, groups, organizations, actors, or other roles. (See [Selecting Life Cycle Roles](#).)
- Access permissions for the roles associated with each life cycle phase. (See [Defining Life Cycle Access Control Rules](#).)
- Workflow processes to be associated with each phase and gate. (See [Associating a Workflow Process with Phases and Gates](#).)
- Promotion criteria to help determine whether or not an item is ready to move to the next phase in its life cycle. (See [Defining Promotion Criteria](#).)
- Transition rules to help determine the path from the selected state to any other state within the life cycle template. (See [Defining Transitions](#).)

Use the toolbar buttons to create a graphical representation of the life cycle you are defining. For information on the buttons, see the online help.

## Overriding and Reassigning Life Cycle and Team Templates

In PDMLink, during the creation of a part and a document, the life cycle template and the team template selection can be manually selected. It is also possible to reassign a life cycle and reset a team template on an existing part or document. In most cases, the manual selection can be done from either a drop-down list or a Search function.

The life cycle drop-down list is automatically generated based on the object type being created and the class attribute value that is assigned in a life cycle template. A life cycle template contains an attribute that specifies for which class the life cycle is valid. The class attribute can be set to a value of an abstract class, a parent class, or a leaf node class. See the Class field in the graphic below.



Based on the object type being created, all enabled life cycle templates which have a class attribute value matching the selected object type being created, or a parent type, or an abstract class, will be included in the drop-down list.

The Search function allows the user to search for any enabled life cycle template that the user has access to, independent of the life cycle templates class attribute value.

## Preference for Manual Selection of Life Cycle and Team Templates

The action to manually select the life cycle and team template during the creation of a part and a document is available to both administrative and non-administrative users. The action is enabled by setting the following preference to true at the site level:

PDMLink/displayLifeCycleTeamPickersOnCreate

Preferences are set on the Site tab from the Utilities page, using the Preference Manager. For more information, see the *Windchill System Administrator's Guide*.

## Preference for Reassigning Life Cycles and Resetting Team Templates

The action to reassign life cycle templates and reset team templates on existing parts and documents is only available to users with administrative privileges.

The action is enabled by setting a preference at the site level and giving individual users administrative permissions. The user must be added to the Administrators group to display the drop-down lists. For more information on adding a user to a group, see the [Administering Principals](#) chapter.

Set the following preference at the site level to true:

PDMLink/displayReassignLifeCycleAndResetTeam

Preferences are set on the Site tab from the Utilities page, using the Preference Manager. For more information, see the *Windchill System Administrator's Guide*.

## Life Cycle Properties

In the **Create** (or **Update**) **Life Cycle** window, the **Properties** panel (the lower part of the window) displays the properties of the life cycle itself. When you select a phase icon, this panel reflects the properties of that phase.

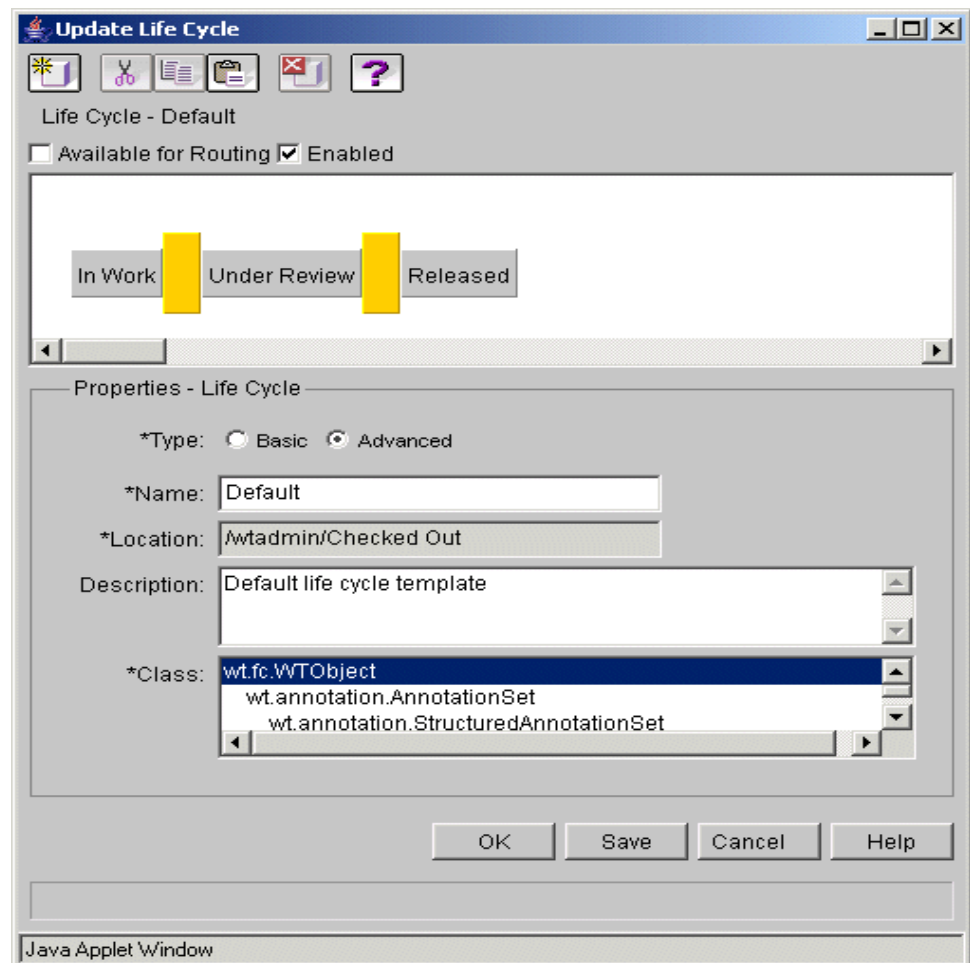
A life cycle has the following properties:

Life Cycle Property	Description
Type	Specifies whether the life cycle is basic or advanced. The basic life cycle contains only the life cycle template, state, and life cycle history. The advanced life cycle includes extra capabilities of phase information, team, related objects, and so forth.
Name	Specifies the name of the life cycle. Life cycle names must be unique. If you enter a name already in use, an error message appears. When you update a life cycle, you cannot change its name. This is a required property.
Location	Specifies the cabinet and folder in which this life cycle is stored. The System cabinet is the default location. This is a required property for Windchill Foundation & PDM.  <b>Note:</b> Only Windchill Foundation & PDM allows the user to select the location. Windchill PDMLink and Windchill ProjectLink automatically check in the template to the System folder for the context in which it was created.
Description	Specifies optional text describing this life cycle.
Class	Specifies the object type to which this life cycle applies. This is a required property.



Life Cycle Property	Description
Enabled	Indicates whether the life cycle-managed item is enabled or disabled. Select the check box to enable a life cycle-managed item when the life cycle is created. Typically, you clear the <b>Enabled</b> check box only when you plan to delete the life cycle in the future, when it is no longer being used by a life cycle-managed item.
Available for Routing	Used only by Windchill ProjectLink, this property specifies the life cycle templates can be used for routings.  <b>Note:</b> Only life cycle templates that have one workflow associated to the first phase of the life cycle support routing.

The following figure displays the **Update Life Cycle** window, with the Default life cycle and its properties displayed.



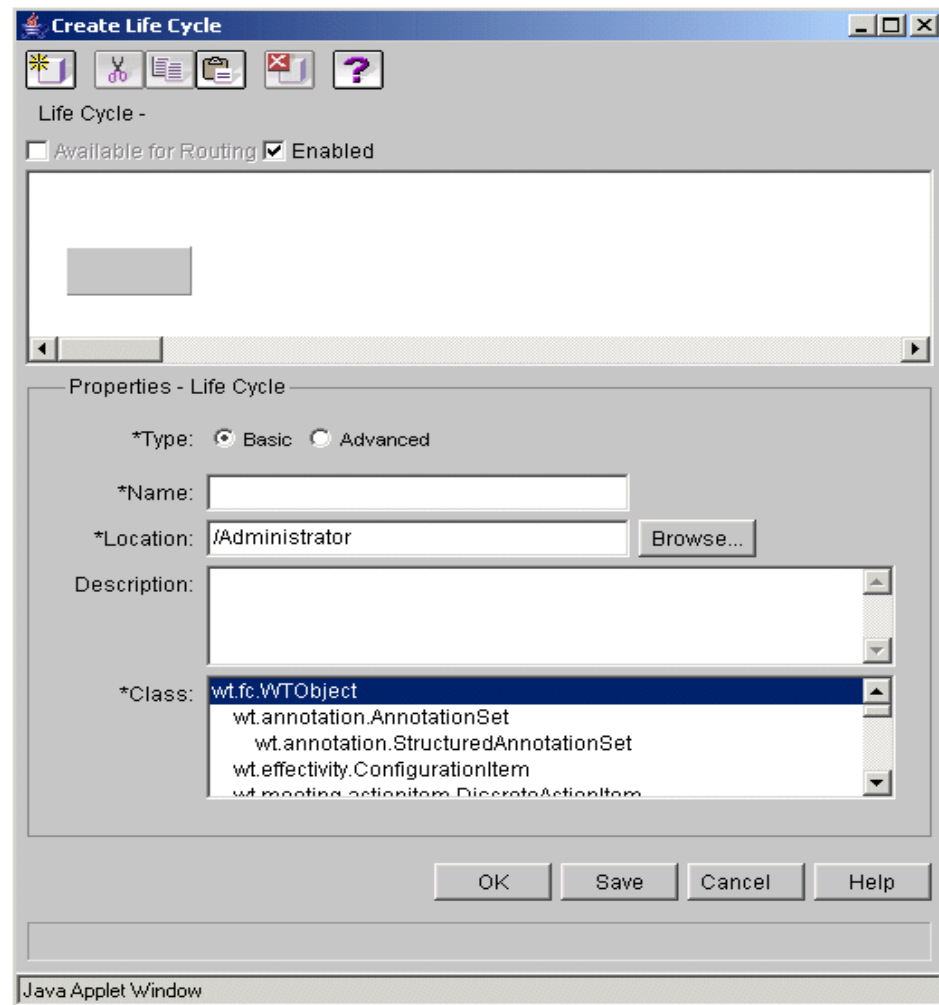
The **Class** display in the panel above provides a tree view of all object types subject to life cycle management. You must choose the type to which this life cycle applies. Because Windchill types are hierarchical, the life cycle is applicable to the selected type and all of its subtypes. A type can inherit more than one life cycle; you can directly associate a life cycle with a given subtype. For example, you could associate a life cycle with the type WObject, and all its subtypes would also be associated with that life cycle. You could also associate those subtypes (for example a change request, WTChangeRequest2) with other life cycles.

When users create items subject to a life cycle, they must choose an appropriate life cycle as part of the creation process. All of the life cycles in the current context, or inherited by ancestor contexts to which you have read access, that are associated with the type of the item appear in the drop-down list. If you want to use a life cycle defined for another object type, click **Search** to list all life cycles, irrespective of object type.

## Defining Life Cycle Phases

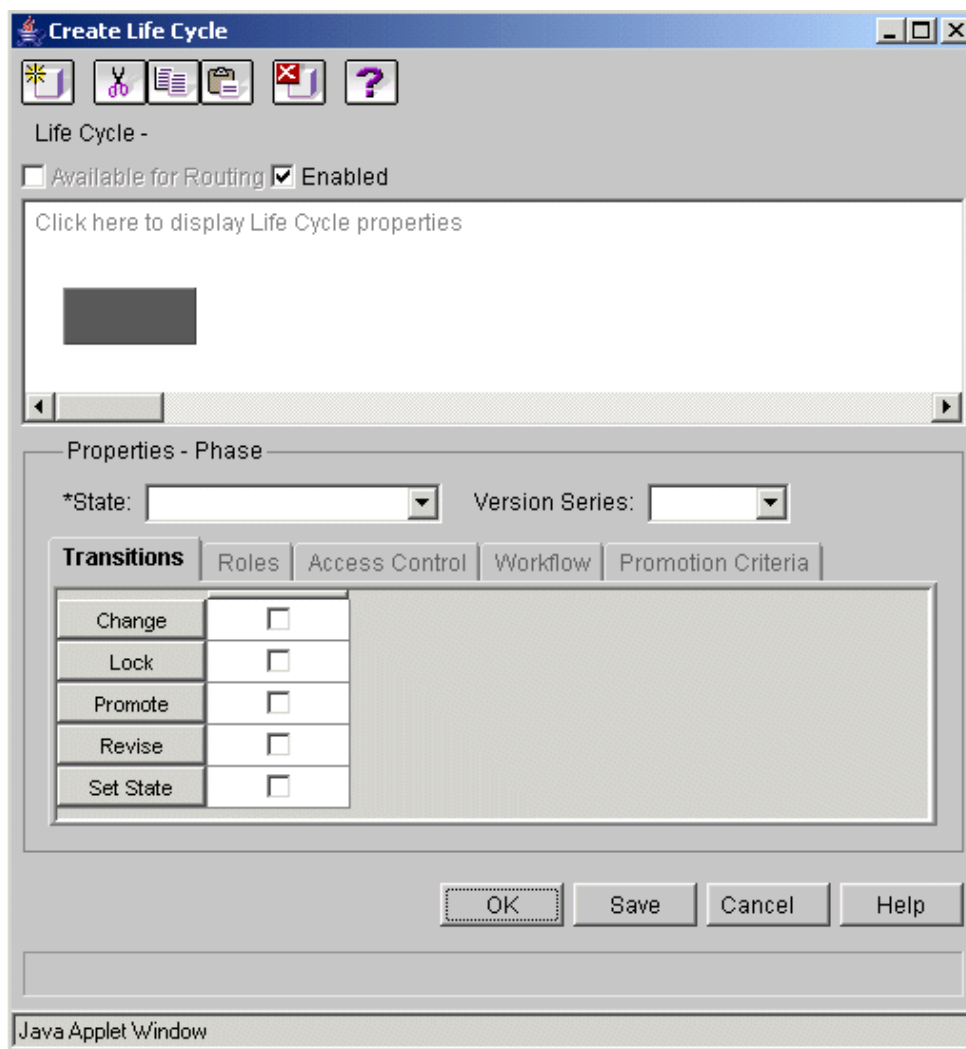
The following graphic displays the **Create Life Cycle** window. When you first create a life cycle, select the phase icon to see a single undefined phase, along

with the window where you can define the phase properties. Here you can also use the tabs to define the characteristics of the life cycle phases and gates.

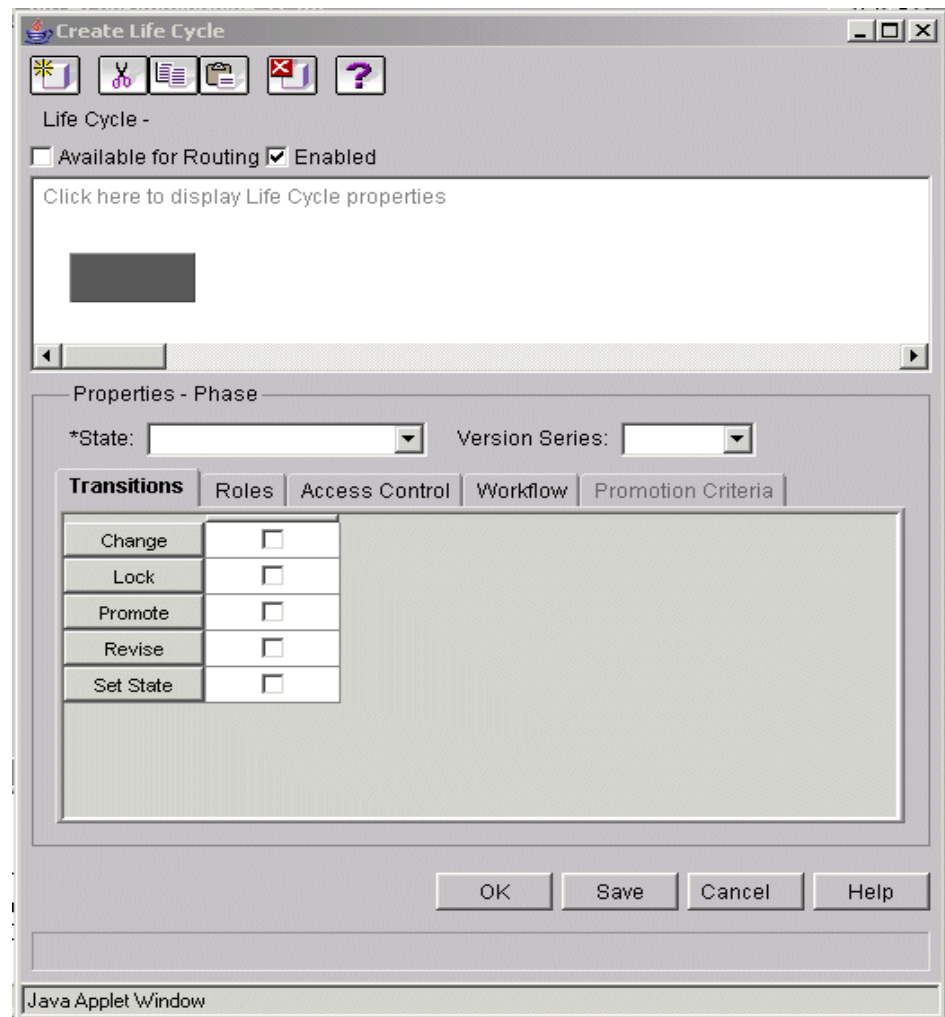


**Note:** To view the properties of the life cycle itself, click anywhere on the life cycle diagram background to open the **Properties–Life Cycle** panel. The **Properties–Life Cycle** panel also opens when you delete a phase.

To view a **Properties–Phase** panel, click on a phase of the life cycle diagram. The figure that follows displays the **Properties–Phase** panel of a basic life cycle.



The figure that follows displays the **Properties–Phase** panel of an advanced life cycle.



Note the difference between the two panels. With a basic life cycle, the only tab available is **Transitions**. With an advanced life cycle, **Transitions**, **Roles**, **Access Control**, and **Workflow** are available.

The following table provides a brief description of the phase properties.

Phase Property	Description
State	<p>When you add a phase icon to the life cycle diagram, you must choose the state it represents from the drop-down list, which is populated with all available states.</p> <p>Windchill provides predefined states (for example, In Work and Under Review). You can define additional states by adding them to the StateRB.rbInfo resource file. For additional information, see, <a href="#">Predefined Life Cycle States</a>.</p> <p>When you select a state, its name appears on the phase icon. The other phase properties you add define the behavior associated with an item while it is in this state.</p>
Version Series	<p>For each state, you can define a version series. The version series determines the versioning scheme for items in the state. The choices in the Version Series drop-down list include mil std and numeric. The chosen series is an attempt to select the correct versioning scheme; various runtime circumstances determine whether the specified series selection is actually employed.</p> <p>For more information, see <a href="#">State-based Revision Sequences by Life Cycle State</a>.</p>
Transitions	<p>For each life cycle phase, you can define the transition path from the selected state to any other state within the life cycle template. If you do nothing from the <b>Transitions</b> tab, the template uses the default state transitions. You can define additional transitions by adding them to the TransitionRB.rbInfo resource file as described in the <i>Windchill Customizer's Guide</i>. This resource file is located in the &lt;Windchill&gt;/src/wt/lifecycle directory.</p>

Phase Property	Description
Roles	<p>For each life cycle phase, you can select roles (for example, Reviewer or Workflow Assignee).</p> <p>When a separate workflow is used for each phase, or each phase and gate pair in the life cycle, a Submitter role and a Promoter role must be added to the appropriate phase and gate, and the proper users will need to be assigned to those roles in the Team so that proper role resolution takes place at runtime.</p> <p>These roles are mapped to role players. A role player can be specified as a user, a user group, an organization, an actor, or another role.</p>
Access Control	<p>You can also define access control rules that will be in effect for this phase. These rules, which specify permissions for each role, will be added to those already in effect for the item, based on the domain's access policy.</p>
Workflow	<p>You can choose a workflow process to be associated with this life cycle phase and with the gate representing promotion to the next phase.</p>
Promotion Criteria	<p>You can define the criteria for promotion of an item from this phase to the next phase in the life cycle.</p>

An item must be approved and explicitly promoted in order to move forward in its life cycle. To illustrate this, promotion gate icons divide the phases in the life cycle diagram.

## State-based Revision Sequences by Life Cycle State

The versioning schemes for Windchill PDMLink are identified based on the object type through object initialization rules.

For a specific object type, such as a part or document subtype, each object type can use a specific version scheme referenced by a scheme name.

These version schemes are then identified in an XML file. For example the out-of-the-box XML file loaded with Windchill PDMLink is:

```
StateBasedVersioning.xml
```

This file contains the sequences and seeds (for example, NUMERIC and MILSTD).

**Note:** To change version schemes, do not manually modify the XML file; instead, use the steps described in the [Setting Up a File-based or State-based](#)

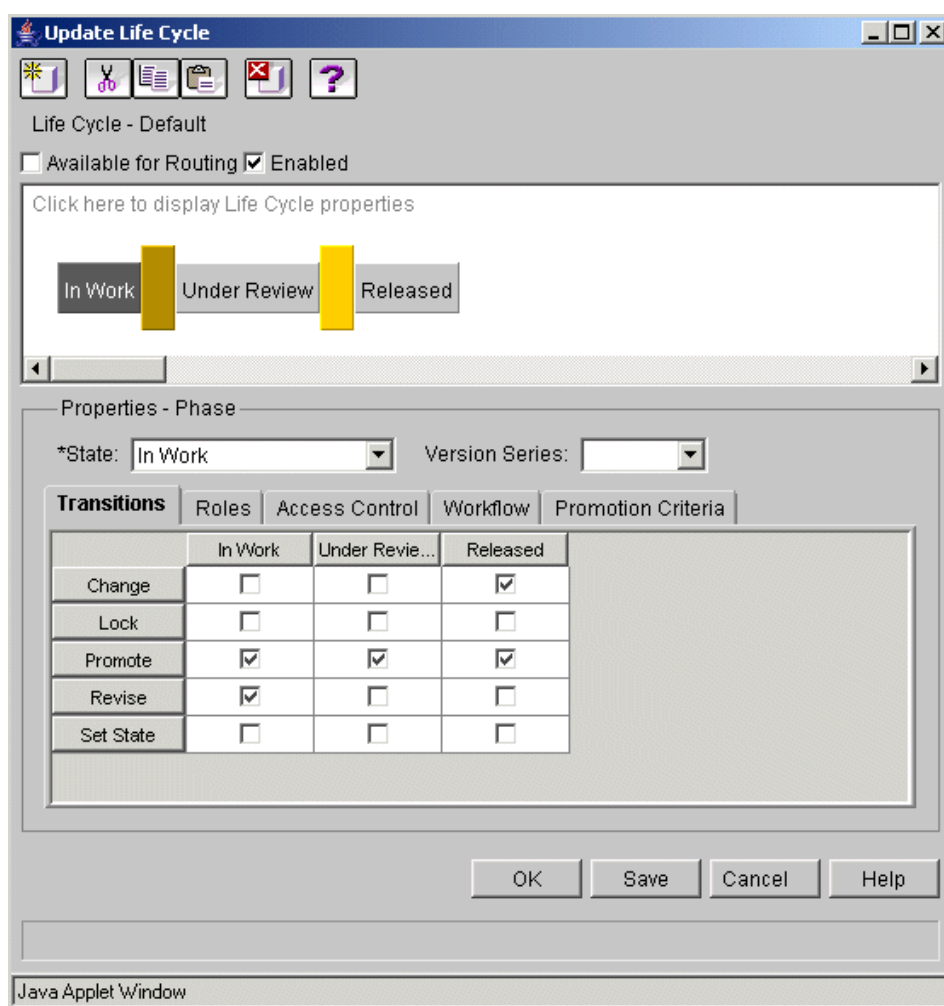
[Versioning Scheme](#) section of the Administering Containers chapter to load an updated XML file.

For information on setting up the state-based versioning scheme that corresponds with the **Version Series** drop-down list that is available for each state, see [Administering the Versioning of Parts, Documents, and CAD Documents](#) in the Administering Containers chapter.

## Defining Transitions

Click the **Transitions** tab of the **Properties–Phase** panel to select the transitions to be associated with this life cycle phase. When you create, update, or view a life cycle, the Transitions tab displays.

The figure that follows shows the Default life cycle on the **Update Life Cycle** window, the **Transitions** tab active:





In the figure above, the **In Work** state is selected in the life cycle panel. This means that information on the tabs below, including the **Transitions** tab correspond to the In Work phase of the life cycle being updated. A transition is a named path from a source phase to a destination phase in a given life cycle. In this figure, **In Work** is the source phase. The table of check boxes indicates to which destination states there are particular transitions.

For example, the check in the first column of the third row means that there is a Promote transition from **In Work** to **In Work** (this is an example of a reflexive transition, where the source and destination phases are the same).

Similarly, the check in the third column of the first row means that there is a Change transition from **In Work** to **Released**.

So, the defined transitions shown in this figure are:

In Work – Change to Released

In Work – Promote to In Work

In Work – Promote to Under Review

In Work – Promote to Released

In Work – Revise to In Work

These are not all the transitions for the entire life cycle, but only for those with **In Work** as a source phases. Other transitions are viewed by activating the **Transitions** tab when other source phases are selected.

## Transition Defaults

Life cycle transition defaults are applied whenever a life cycle is saved (either interactively or via a bulk load) or is upgraded and does not otherwise contain explicit transition information.

Default transitions are initialized according to a specification recorded in the *wt.properties* file. The specification is encoded using a small language, whose grammar is given (beginning with a valid *wt.properties* entry):

```
wt.lifecycle.transitions.default.<transition_name>=<expression_list>

transition_name = { key element of wt.lifecycle.Transition }
expression_list = expression ["," expression_list]
expression = term ["|" expression]
term = "ALL" | "FIRST" | "LAST" | state_name
state_name = "\"" { key element of wt.lifecycle.State } "\""
```

Each property entry specifies a rule for creating default instances of a particular transition type in each life cycle. Each rule is applied to each state in each life cycle. Each expression in each *expression\_list* is evaluated to create transitions to

the specified states. However, each *term* in each *expression* is evaluated in order of appearance only until one is encountered that validly specifies one or more states; the remaining terms in the expression are ignored.

The out-of-the-box default transition expressions are:

```
wt.lifecycle.transitions.default.PROMOTE=ALL
```

```
wt.lifecycle.transitions.default.CHANGE="RELEASED"
```

```
wt.lifecycle.transitions.default.REVISE=FIRST
```

These rules can be interpreted as follows:

- **PROMOTE:** create a PROMOTE transition from each state to every state in each life cycle.
- **CHANGE:** create a CHANGE transition from each state to the RELEASED state in each life cycle. If the life cycle does not have a RELEASED state, no transition is defined.
- **REVISE:** create a REVISE transition from each state to the first state in each life cycle.

## Additional Advanced Life Cycle Management Options

The following sections apply only to advanced life cycle templates:

- Selecting Life Cycle Roles
- Defining Life Cycle Access Control Rules
- Associating a Workflow Process with Phases and Gates
- Defining Promotion Criteria

**Note:** These options are legacy options primarily available for existing Windchill Foundation & PDM and Windchill PDMLink installations. For Windchill PDMLink and Windchill ProjectLink, PTC recommends using the basic life cycle type and their associated concepts for managing workflow processes, teams, promotion criteria and access control policies.

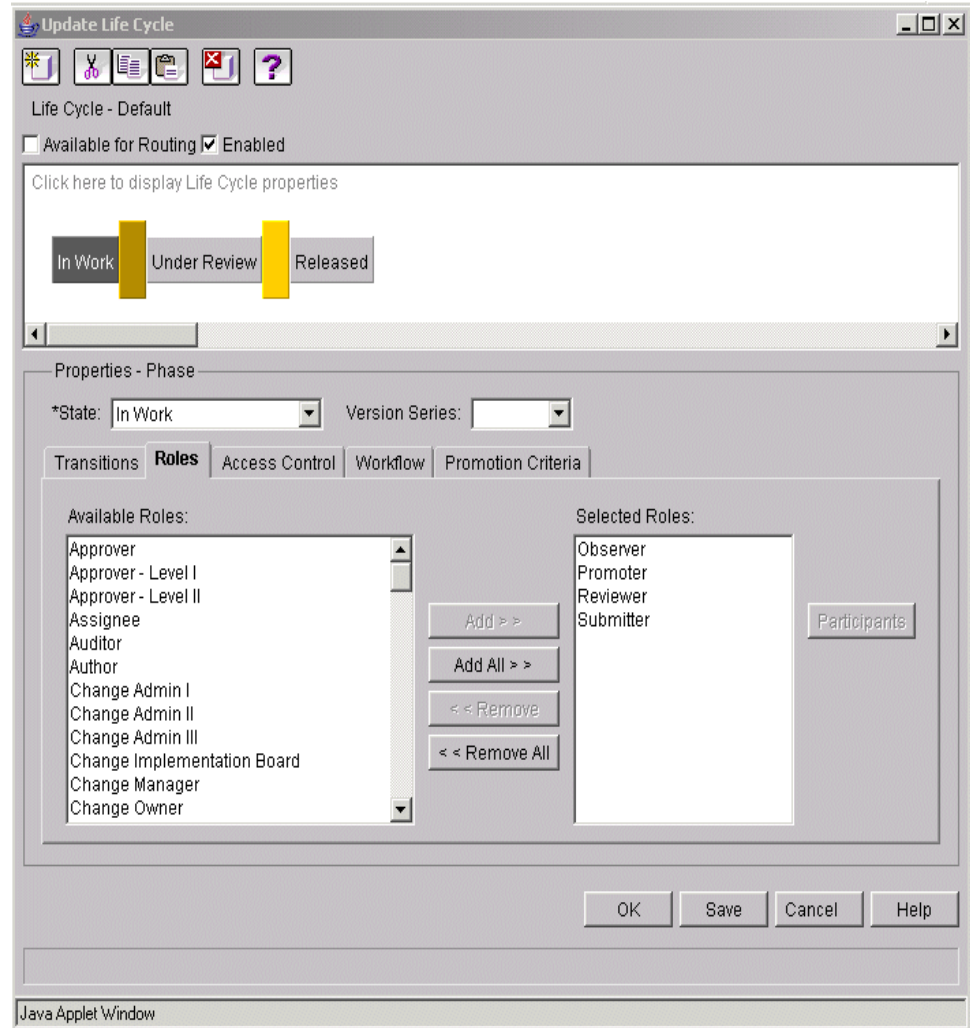
### Selecting Life Cycle Roles

**Note:** This section applies only to advanced life cycle templates.

Click the **Roles** tab of the **Properties–Phase** panel to select the roles to be associated with this life cycle phase. When an item is promoted to this phase in its life cycle, these roles are resolved to principals (users, groups, or organizations) who perform one of the roles in the **Available Roles** list.

The Submitter is responsible for submitting the item for promotion to the next phase in its life cycle. Submitter is a required role for a final life cycle phase only if another role (for example, Reviewer or Observer) has been added to that phase.

The following figure is an example of the **Roles** tab panel:



To add a role to the phase, select the role and click **Add** to move it to the **Selected Roles** list. You can also click **Add All** to move all the displayed roles to the **Selected Roles** list. Click **Remove** or **Remove All** to delete roles from the list.

**Note:** The **Available Roles** list is populated with predefined roles. You can define additional roles by adding them to the RoleRB.rinfo resource file. Defined roles are added to this list when you recompile RoleRB.info and deploy the class file to your production environment. For additional information, see the enumerated types information in the *Windchill Customizer's Guide*.

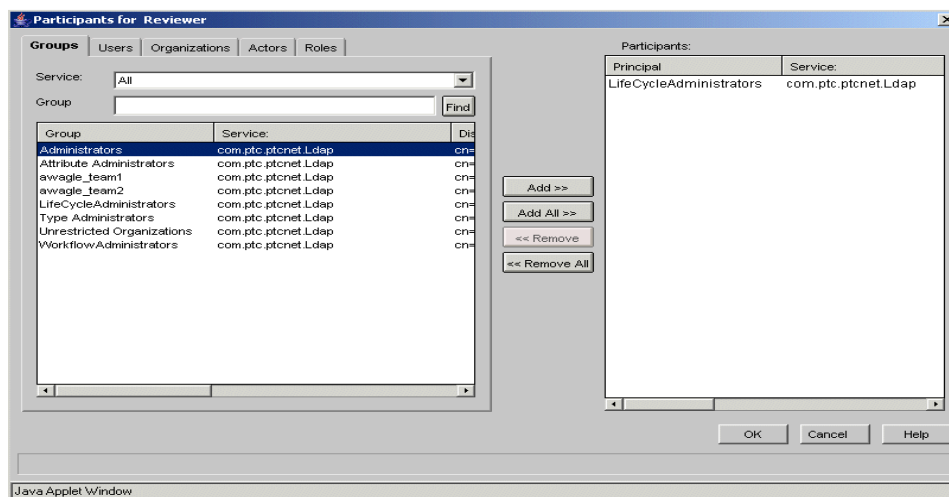
## Role Mappings

A role mapping is resolved in one of several ways:

- You can directly map life cycle roles to users, groups, organizations, or other roles. However, since organizations generally want to define only a small number of life cycles, it is not often practical to map life cycle roles directly to principals. Using life cycles and teams together allows role participants to be identified at runtime, rather than making this mapping an explicit part of the life cycle definition.
- You can map life cycle roles to team roles. At runtime, the role is resolved according to the team role mapping. (For example, the life cycle role Promoter could be mapped to the team role Team Leader, and the life cycle role Promoter would be resolved at runtime according to the Team Leader role, as mapped in the team.)
- You can map a life cycle role to an actor. That is, you can map a role to someone who performs a specific action within the context of the business object. At runtime, this role is then resolved to the principal who created the object with which the life cycle is associated. For example, you could assign the Creator actor to the Submitter role for a given life cycle phase. For that phase, the user who created the item would be assigned the Submitter role at runtime. If the Submitter role is defined in the team, it resolves to the team's Submitter role.

## Selecting Participants for Roles

To add participants to a specific life cycle role, select a role in the **Selected Roles** window. Then click **Participants** on the **Roles** tab, to choose participants for the selected role. The **Participants for** window opens (as shown in the following figure), allowing you to choose users, groups, organizations, an actor, or other roles to be mapped to this role.



- Click the **Users** tab to choose from a list of users defined in the context-specific list of users filtered by service (source) and access control. From this tab:

- Select **All** or select a specific directory service, from the **Service** drop-down list. The default service displayed is the service associated with the bundled Aphelion Directory. If users are contained in an enterprise directory, be sure to select the service for that directory.
- To limit the users displayed in the table, enter the name of a user in the **Full Name** or **User Name** field, and click **Find**. You can use the \* or % wild card characters in the name, where \* represents one or more characters and % represents exactly one character.

The specified services are searched and the matching results display in the table. When your system has a lot of users, this is the recommended method of finding a specific user.

If you leave the **Users** field blank and click **Find**, all users from the selected service display in the table. If your system has a lot of users, displaying all users may take a long time.

- Similarly, click the **Groups** tab to choose from a list of groups or click the **Organizations** tab to choose from a list of organizations.
- Click the **Actors** tab and choose an actor to base your selection on a particular user action. Creator is the only actor defined. The Creator is resolved at run time to the user who created the selected item.
- Click the **Roles** tab and choose a role to resolve the life cycle role.

To add a principal to a role, select it and click **Add** to move it to the **Participants** list. You can also click **Add All** to move all the displayed users, groups, or organizations to the **Participants** list. Click **Remove** or **Remove All** to delete participants from the list.

Click **Help** to view detailed instructions for selecting participants.

## Defining Life Cycle Access Control Rules

**Note:** This section applies only to advanced life cycle templates.

Access to a specific item (for example, a document or a part) is controlled by the access policy for the domain in which the item is located. In many contexts, policy access control lists (ACLs) are sufficient for controlling access to items; however, when an item is part of a life cycle, there are often many different principals who must participate in moving an item through its phases.

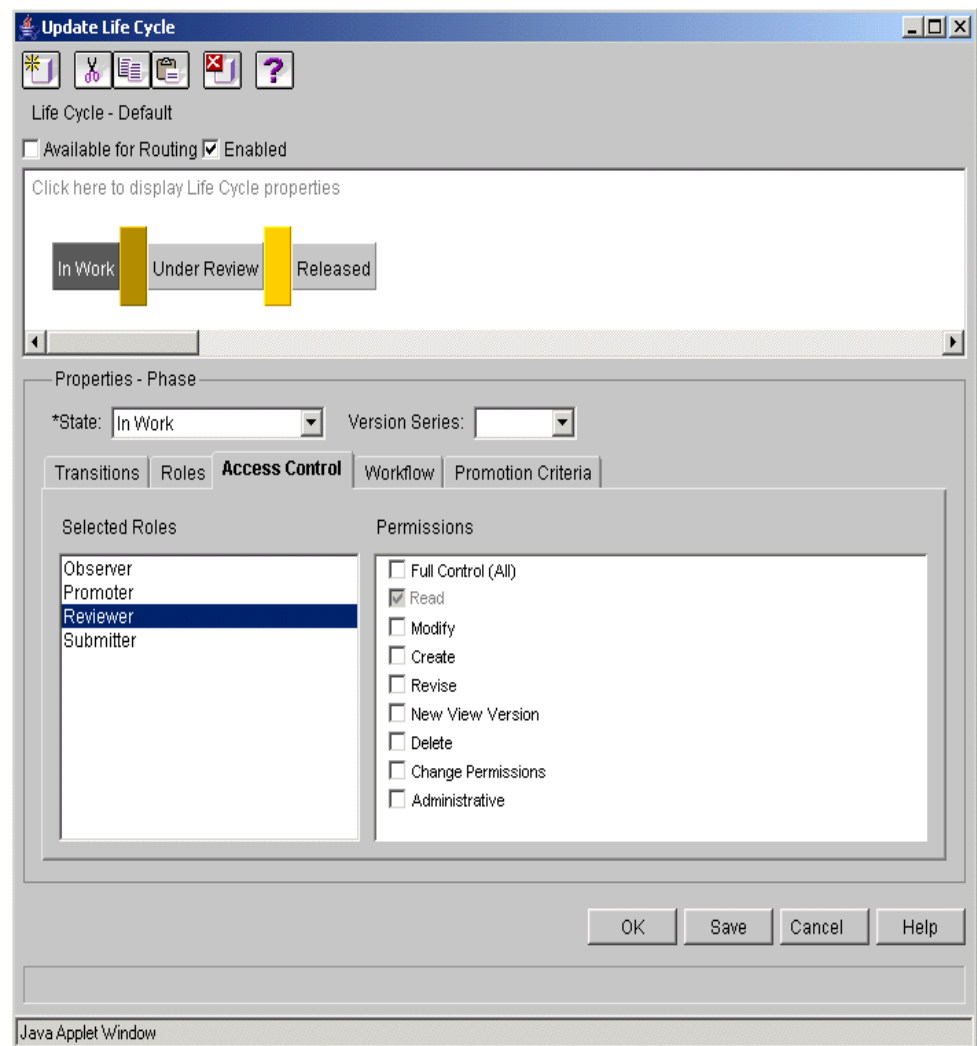
Depending on your security needs, you may not want to create domain policy rules that provide all of the necessary permissions. If this is the case, access to an item can be controlled by an *ad hoc ACL* that is part of its life cycle. In general, policy ACLs apply to an object type within a domain, and ad hoc ACLs apply to an instance of the type while that item remains in a specific life cycle phase.

Rules in an ad hoc ACL are added to the rules in the policy ACL for a given object. The ad hoc rules exist for the duration of a specific life cycle phase. These rules grant roles additional access to an item during the life cycle phase. This access is then revoked when the item moves to a new phase and the participant no longer needs the ad hoc permissions.

Ad hoc ACLs can only grant permissions. They cannot be used to deny access to an item.

To create ad hoc rules, you must select a life cycle phase on the **Create/Update Life Cycle** window. Click the **Access Control** tab, select a role, and choose the appropriate access permissions.

The following figure is an example of the **Access Control** tab panel:



All roles are automatically given Read permission, so the associated principals can access their tasks and view the item. By default, submitters are automatically given Modify permission so they can submit the item for promotion as part of updating it; however, you can change this at your site. For each role, you can also select one or more of the permissions described in the [Administering Access Control](#) earlier in this guide. You can also learn more about domain access control and the relationship between policy and ad hoc ACLs.

## Associating a Workflow Process with Phases and Gates

**Note:** This section applies only to advanced life cycle templates.

By default, all advanced life cycles have predefined workflow processes associated with the phases and gates.

Workflow process to moves items through their states is separate from the life cycle of item. You can manage one or more items with a workflow process.

To change the workflow process that is associated with a phase or a gate, modify the following properties in the wt.properties file:

- wt.lifecycle.defaultPhaseProcess
- wt.lifecycle.defaultGateProcess

As shown in the figure that follows, the Submit process is automatically associated with the In Work phase. The Review process is associated with the gate by which an item moves from In Work to its next phase.



**Note: Use Latest Iteration** is selected, so the most recent iteration of the workflow process template is used at instantiation. If this check box is cleared, the specific iteration selected is used, even if it is not the most recent.

**Update Life Cycle**

Life Cycle - Default

☐ Available for Routing ☒ Enabled

Click here to display Life Cycle properties

In Work Under Review Released

Properties - Phase

\*State: In Work Version Series: 1.0.0

Transitions Roles Access Control **Workflow** Promotion Criteria

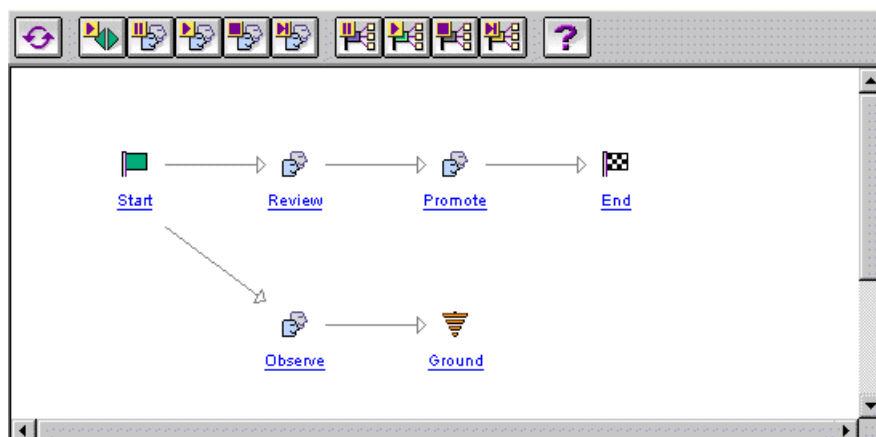
Phase Process: [Submit](#) Browse... ☒ Use Latest Iteration Clear

Gate Process: [Review](#) Browse... ☒ Use Latest Iteration Clear

OK Save Cancel Help

Java Applet Window

As a result, the Review workflow, which is shown in the following Workflow Process Manager figure, defines the process and activities that are part of moving an item forward from the In Work phase.



This workflow process has three defined activities: Review, Observe, and Promote. You can view the properties of each link and activity within the process on the Workflow Process Manager. For example, for the Review workflow, the participant to be assigned the Review task is the Reviewer role.

Therefore, when an item is submitted for promotion from the In Work phase and the Review workflow process is started, the Reviewer role is mapped to an actual user based on role mappings in either the life cycle or a team. In Windchill Foundation & PDM, the Review task is added to the Windchill worklist for that user. In Windchill PDMLink and Windchill ProjectLink, the Review task is added to the Assignments table for the user.

The Submit and Review workflow processes are predefined and available for your use when Windchill is installed; however, your organization may have a number of additional workflow processes in place. To associate a specific workflow process with a phase or gate, click **Browse** to locate and select a process from the shared location.

## Defining Promotion Criteria

**Note:** This section applies only to Advance life cycle templates.

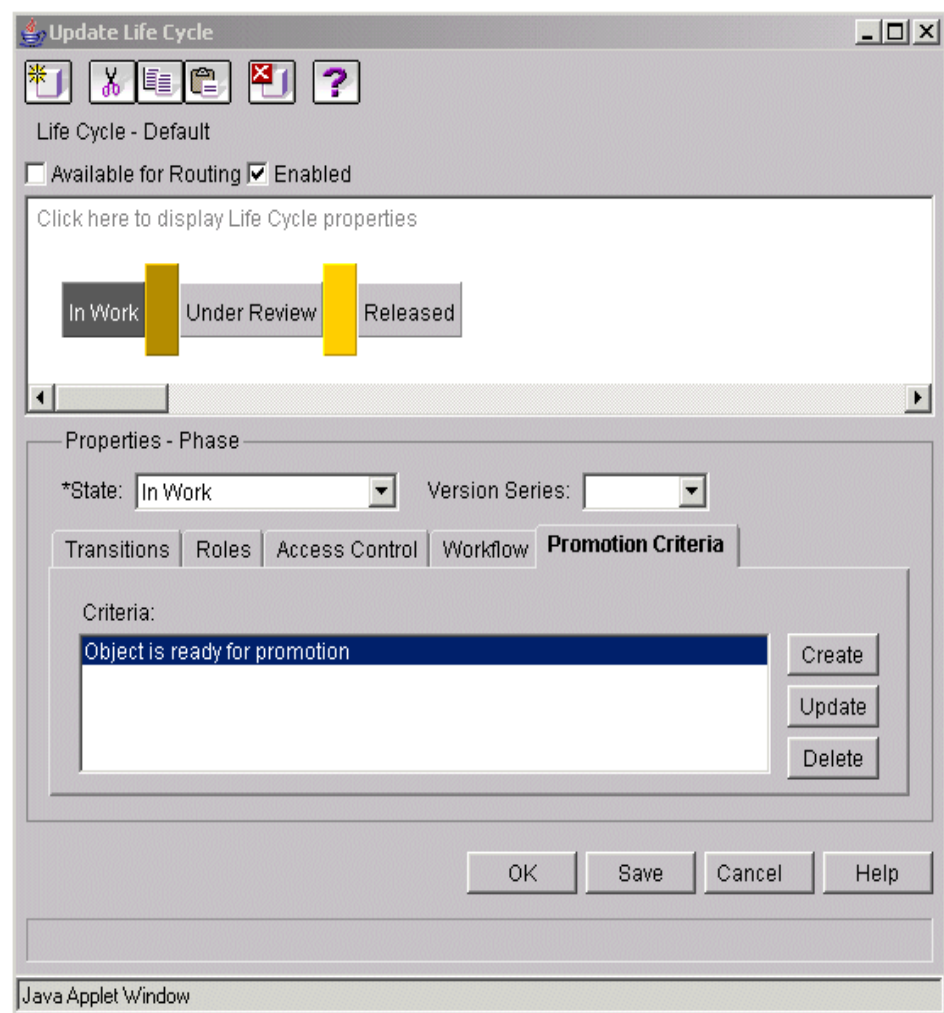
Life Cycle Management does not enforce the satisfaction of promotion criteria. For example, reviewers' votes are not tabulated, and any reviewer is allowed to check off one or more promotion criteria. The promoter can choose to promote the item to its next life cycle phase, regardless of whether all reviewers have voted or all promotion criteria have been satisfied.

However, the following specifies promotion criteria guidelines you can use to help reviewers and promoters make appropriate decisions, based on your site's processes:

- Click the **Promotion Criteria** tab to view the existing criteria set.
- Click **Create** or **Update** to create or modify the criteria.
- Click **Delete** to remove a criterion.

The online help contains detailed instructions to help you perform these actions.

The following example displays the **Promotion Criteria** tab panel:



When you click **Create** or **Update**, the **Create Criteria** window opens. Type a criterion statement in the field provided. For example, you could enter the following statement for a given phase of a life cycle:

```
All reviewers have voted for promotion.
```

This statement would then serve as a criterion for promoting an item from the current life cycle phase to the next.

## Predefined Life Cycle States

Windchill includes many predefined life cycle states and roles. You can define additional states and roles by customizing the `StateRB.rbInfo` and `RoleRB.rbInfo` resource files as described in the *Windchill Customizer's Guide*. These resource files are located in subdirectories under the `<Windchill>/src` directory. Newly defined states and roles are added to the existing states and roles when you recompile the resource files and deploy the class files to your production environment. For additional information, see the enumerated types information in the *Windchill Customizer's Guide*.



**Caution:** Removing a value you previously added to an enumerated type (for example, removing a state in the `StateRB.rbInfo` resource file), could result in serious runtime error. Do not remove a state unless you are certain there is no reference to it within the system.

## Import and Export

### Preparing to Import or Export Life Cycles

Before you begin, you should be familiar with the following information regarding importing and exporting:

- Upgrade to the latest maintenance only release (MOR) as it becomes available, to ensure you have the latest enhancements to the import and export functionality.
- You can import a life cycle into a later version of Windchill; but not to an earlier version. That is, importing and exporting are not backwardly compatible. To import from versions prior to Windchill 7.0, you must convert the CSV files to XML format. Use the `CSV2XML` utility to convert CSV files to XML files. The syntax is as follows:

```
windchill wt.load.util.CSV2XML -input <input file or directory>  
-output <output directory> -root <root directory> -help
```

- Importing or exporting life cycles creates objects in a JAR or ZIP file format. (This is the same format that the load.Installer functionality uses.)

Use a zip utility like WinZip to create a ZIP file consisting of XML files. To create a JAR file, use the following syntax:

```
jar -cf <jar file name> <xml files to be included>
```

or

```
jar -cvf (for verbose output to print each file name as it is added)
```

- Importing a JAR or ZIP file with one or more XML files creates one or more life cycle templates (depending on how many templates were defined in the XML files).
- There is no limit to the number of life cycles that you can export. You can export multiple life cycles into a single JAR or ZIP file. Select them (on the **Life Cycle Administrator** page) and click **Export**. All the selected life cycles are exported to the same JAR or ZIP file.
- Errors can occur, especially, when importing life cycles. Some errors result in messages displayed; others cause a loss of data. Check the method server log for error information.
- When you export a life cycle, only the life cycle itself is exported. This includes references to underlying objects, such as principals, roles, and actor roles. However, the underlying objects themselves are not exported. If the export file is used to import the life cycle into another system, the underlying objects must first exist in the system, or the import fails and errors appear in the method server log. This can occur, especially when importing the object into a different system. Be certain that all underlying objects referenced in the XML representation of the life cycle exist.
- If a life cycle is imported and a life cycle with the same name already exists in the Import directory, the results depend on the Iteration On Import setting in the wt.properties file. If it is set to true, the imported life cycle is appended to the existing life cycle as a new iteration. If it is set to false, the imported file causes a method server exception, stating that there is a duplicate name, and the life cycle is not imported.

## Importing and Exporting Life Cycles

To import or export life cycles, use the **Import** and **Export** buttons in the **Life Cycle Administrator** window.

To import one or more life cycles from a JAR or ZIP file in the life cycles export directory, select a file from the **Import** dialog box, and click **Import**.

To export one or more life cycles into an XML representation of the life cycles in the life cycles export directory, use the following procedure:

1. Select a template from the **Life Cycle Administrator** window, and click **Export**. (The **Export** button is disabled if you do not have a life cycle selected.)
2. A grant permission window may appear asking for permission to access the local file system and to write a file on that system. If you select the remember selection check box, permission need only be granted once. Once permission is granted, a **Browse for file** picker opens, defaulting to the system **temp** folder.
3. You can pick a file that exists or type in a new name. If the file name exists, you are asked to confirm to overwrite the file. You must click **Yes** to continue the export. If the file name does not exist, a new file is created.

There is no confirmation that the export is completed. When the progress bar and the hourglass on the life cycle administrator applet disappear, the export is complete.

If you want to use a template in another context (such as between two organizations, two solution contexts, or a solution and organization context), export the template from the source context and import it into the target context.

You can update a template to make it specific to an organization or solution context by copying them (using **Save As**) from parent contexts.

## Access to Life Cycle Administration

As described in this chapter and [Administering Teams and Roles](#), administrators create life cycles and teams.

The access control rules listed in the following table provide life cycle administrators with permissions needed to manage life cycles and teams, and to move them after they have been created. These rules need to be defined for the domains (or ancestor domains) associated with folders in which life cycle templates and teams will reside.

Object Type	Permissions Required
AdministrativeDomain	Read
LifeCycleTemplate	Read, Modify, Create, Delete
Team Template	Read, Modify, Create, Delete
Cabinet	Read, Modify
SubFolder	Read, Modify
WTContainer	Read

## Best Practices for Windchill PDMLink and Windchill ProjectLink

Site, organization, and solution administrators manage life cycle templates.

- Site administrators create, modify, delete, and view life cycle templates in the site context.
- Organization administrators create, modify, delete, and view life cycle templates in the given organization context. Organization administrators can view life cycle templates from the site text.
- Solution administrator create, modify, delete, and view life cycle templates in the given solution context. They can view life cycle templates from the parent organization context and the site context. This includes administrators of Windchill PDM library contexts.

**Note:** The Life Cycle Administrator is not available to administrators of project contexts.

The Life Cycle Administrator client displays a table that lists all life cycle templates belonging to the given context, plus those belonging to its parent contexts. A column in the table identifies the context owning each life cycle template.

When you create a life cycle template, the system saves the new life cycle template in the System cabinet or folder of the context in which it is created. Consequently, the **Create** dialog for life cycle templates disables the location field since it is the system, not the user, that decides where the new life cycle template is to be located.

**Note:** When assigning a workflow template to a life cycle template, you see a list of valid workflows. The list of valid workflow templates includes the ones defined in the given solution context, plus those defined in the parent organization and the site contexts. Workflow templates defined in a sub-context override and filter out the workflow templates defined in parent contexts having the same name.

The search scope used to locate groups is determined by the type of administrator doing the search. For more information about the search scope, see [Searching for Principals](#) in the Administering Containers chapter.

## Life Cycle Support in Windchill ProjectLink

The following table shows the life cycle support in Windchill ProjectLink.

Feature	Support Level	Notes
Life cycle template definition	Full	Organization administrators may create, update, and delete life cycles for their organization.
Life cycle-managed object support	Partial	Only document and part items are supported.
Phases	Full	
Gates	Full	
Default life cycles	Partial	The following default life cycles are supported out of the box: Default, Basic, Approval, Review, and Release.
Workflow association	Full	Workflow templates must be defined in the organization, and are available for reference at a life cycle phase or gate.
Roles	Partial	Roles must be defined in the project instance.



Feature	Support Level	Notes
Role-based access control	None	Access rules defined for objects in the project instance take precedence. By default, members of the project manager group have full access to all objects in the project. The life cycle access rules by role for each phase and gate are ADDITIVE. This means access rights can be added for each role only at each phase and gate.
Criteria	Full	
Life cycle template import export	Partial	There is no organization association with the imported or exported template so the life cycle import/export folder should be cleared between uses.
Access to current life cycle state	Partial	The life cycle state is provided on the Details page for the item.
Set life cycle state	Full	Item owners and project managers can set the life cycle state for a document or part with the <b>Set State</b> action.
Access to life cycle history	None	The life cycle history for a part or document is not available in the user interface.
Queries based on life cycle state	None	No user interface is provided to support life cycle-based queries, but this may be supported in a future release.
Access to running process through life cycle	None	

## Best Practices for Managing Promotion Processes in Windchill PDMLink

The decision to promote an item from one state to another often pertains to a business decision. As such, it is important to make sure that the Products and Libraries are set up correctly to allow for the proper approvals to be obtained prior to promotion. See [Administering Products and Libraries](#) for detailed information. For any process template used in the promotion process, users assigned to the particular tasks are obtained from the following:

- the process template itself (the process has a member defined for a specific task)
- the state of the life cycle of the promotion request
- the team instance of the promotion request

If possible, use the team instance to define users who will receive workflow tasks. The out-of-the-box templates provided with Windchill PDMLink do not define users for the roles defined in the process templates or life cycle templates used by the promotion request.

The team template provided for the promotion request defines the roles Approver and Reviewer; however, no default users are assigned to these roles. The Approver and Reviewer roles should be populated by the container team, as well as in **Step 6: Define Participants** step of the **Promotion Request** window.

If using a process template for the promotion process that defines roles in addition to Approver and Reviewer, add the roles to the team template used for promotion request creation. You can modify team templates through the Team Administrator found on the Utilities page of each tab. Create a copy of the team template (using the Save As functionality), update the new template with any new roles, and then modify the appropriate object initialization rule for the promotion request to point to the new team template in the container of interest.

The best primary method for allocating users to roles is by using the container team. You can find the team on the **Team** page of the **Product** or **Library** tab. Both out-of-the-box product templates define two container roles, Promotion Approvers and Promotion Reviewers, to work with the promotion process. These container team roles, along with role mapping copies any users/groups from these roles into the team instance of the promotion request at creation. When a new product is created, the Promotion Approvers role is empty; any approval tasks are sent to the promotion request creator unless this role (or another mapping) is properly defined. For each product, resolve at least one user into the promotion request team instance for the Approver role to ensure that a proper approval is obtained prior to promotion. Out-of-the-box, this can be done in the following ways:

- adding the approver to the Promotion Approvers role of the container team
- modifying the team template to have a default participant for the Approver role
- modifying the preference mapping to add additional roles for the Approver role, ensuring that at least one user is assigned to at least one of the roles in the container team.

## Restrictions on Moving Items Between Containers

If you create a life cycle template in a specific product or library and the template is not available in other products or libraries, then you cannot move any item that uses the life cycle template.

**Note:** For demonstration purposes, the Product Design template (available by default when you create a product) includes object initialization rules that set different default life cycle templates and versioning schemes for parts, documents, and CAD documents than is set out-of-the-box at the site level. This template demonstrates the use of simplified life cycles and state-based versioning. Unless you have overriding business practices that require unique rules, PTC recommends that if you want to use these features, you modify the organization or site object initialization rules and remove them from the Product Design template.



# 15

## Administering Teams and Roles

This chapter provides information about teams, team templates, and roles.

<b>Topic</b>	<b>Page</b>
Overview .....	15-2
Out-of-the-Box Team Templates .....	15-3
Defining Team Properties and Roles.....	15-17
Predefined Roles.....	15-19
Assigning Participants to Team Roles.....	15-20
Best Practices for Windchill PDMLink and Windchill ProjectLink.....	15-21

## Overview

Teams and team templates are used throughout Windchill.

To manage teams, you need to be aware of the different ways that teams are used in Windchill.

- When business objects (such as parts and documents) are created, a unique team is created as its own object and is associated with the business object. This *team* contains all the roles consolidated from the team, life cycle, and workflow templates. The roles get mapped to end users; ad hoc access permissions are defined for the participants in the life cycle and workflow templates.
- At the container level in Windchill ProjectLink and Windchill PDMLink, the manager can create a context team specific to a particular project, library, or product. Each *context team* identifies the members of the specific project, library, or product and assigns the members to roles that have been established in the container.

In Windchill PDMLink, the context teams are also used when creating the team that is associated with a specific business object. For additional information, see [Administering Teams](#) in the Administering Products and Libraries chapter.

The *team template* is an object managed by Windchill Foundation & PDM and Windchill PDMLink administrative users. This object can map participants and actor roles to roles. The team template can be assigned to a life cycle or workflow-managed business object, when it is created, to use as a template for roles resolution for the team.

**Note:** Windchill ProjectLink does not use team templates.

To understand this section, you should be familiar with the following terms:

- A *principal* is an individual user, group, or organization.
- A *role* is a function that can be fulfilled by some principal. A role is mapped to participants. A list of predefined roles is available when you define a team.
- An *actor* represents a user who performs a specific action within the context of a specific business object. Currently, Creator is the only actor defined.
- A *participant* is a principal or an actor, which has been mapped to a specific role in a team.

This chapter describes the following:

- Team Roles Resolution
- Defining Team Properties and Roles
- Assigning Participants to Team Roles

Teams and team templates make it possible to define a smaller number of life cycles, since the life cycle roles can be mapped to team roles, rather than to specific users and groups. For more information about life cycles, see [Administering Life Cycles](#).

The differences and relationships between teams and team templates are summarized in the following list:

- Team templates are used only in the creation of teams. When a business object that requires a team is created, the necessary team is automatically created.
- Updates to team templates do not affect existing teams that were created from the template. Team template changes are reflected in new teams only.
- Each team created for a specific business object is distinct from other teams created for objects of the same type.

For example, two problem reports in the same library may have associated teams that initially appear to be identical, consisting of the same roles and participants; however, changes made to one of the problem report teams do not affect the other problem report team.

- Teams cannot be manually created from team templates. Team creation is automatic, based on associations to business objects and workflow requirements.

**Note:** For more information about how teams are automatically created based on team templates and object associations, see [Team Templates and Object Types](#) later in this section.

## Out-of-the-Box Team Templates

Windchill Foundation & PDM and Windchill PDMLink provide out-of-the-box team templates. The following sections summarize the team templates that are provided.

### Windchill PDMLink Out-of-the-Box Team Templates

This section summarizes the team templates that are provided with Windchill PDMLink. These team templates are designed to work with the Windchill Change Management functions.

In order for the Windchill Change Management functions to operate properly, each of the Change Management objects listed in this section must have an associated team template, and each of the associated team templates must contain certain roles. The following table indicates which team template roles administrators must define, and which are designed to remain empty. (A "Yes" entry in the **Modify?** column means an administrator should define the role.)

Team Name	Role Name	Modify?	Notes
Problem Report Team	Change Admin I	Yes	Assign this role to the user serving as Change Administrator I for problem reports. This user is the initial reviewer of problem reports.
	Problem Report Author	No	This role is used for workflow notifications, and is automatically set to the creator of the problem report. It should be left blank in the team template.
Change Activity Team	Assignee	No	This role is set through the user interface. It should be left blank in the team template.
	Reviewer	No	This role is set through the user interface. It should be left blank in the team template.
Change Request Team	Change Admin I	Yes	Assign this role to the user serving as Change Administrator I for change requests. This user is the initial reviewer of change requests.
	Change Admin II	Yes	Assign this role to the user serving as Change Administrator II. This user is responsible for creating change notices.
	Change Review Board	Yes	Assign this role to the users who are responsible for approving full-track change requests.
	Change Request Author	No	This role is used for workflow notifications, and is automatically set to the creator of the change request. It should be left blank in the team template.
	Problem Report Author	No	This role is used for workflow notifications, and is automatically set. It should be left blank in the team template.



Team Name	Role Name	Modify?	Notes
Change Notice Team	Change Admin III	Yes	Assign this role to the user serving as Change Administrator III. This user is responsible for auditing a change notice after it has been completed and before the product data is released.
	Change Admin II	No	This role is used for workflow notifications, and is automatically set. It should be left blank in the team template.
	Change Admin I	No	This role is used for workflow notifications, and is automatically set. It should be left blank in the team template.
	Change Request Author	No	This role is used for workflow notifications, and is automatically set. It should be left blank in the team template.
	Problem Report Author	No	This role is used for workflow notifications, and is automatically set. It should be left blank in the team template.
	Change Implementation Board	No	This role is used for workflow notifications, and is automatically set. It should be left blank in the team template.

To associate a team template with a particular object type, use the Object Initialization Rules Administrator.

## Windchill Foundation & PDM Out-of-the-Box Team Templates

The out-of-the-box life cycle, team, and workflow templates that are provided with Windchill Foundation & PDM are intended as examples of what a site might set up. The three templates include:

- Change Team
- Change Control Board
- Default

### Change Team

The Change Team example team template is a companion to the following out-of-the-box example workflow processes:

- Change Activity Process
- Change Order Process
- Change Request Process 1

- Change Request Process 2
- Change Analysis Process
- Change Investigation Process
- Change Issue Process
- Change Proposal Process

These workflow processes contain workflow task assignments and notification robots that refer to the roles defined in the Change Team template. The workflow templates are provided as an example and are not intended to be used without modification to suit your particular purpose.

To learn from this example team template, study the workflow processes listed above. Look at each workflow task and observe the role assigned to that task. Each task has a description and instructions to help you understand the purpose of the task. You will note that sometimes a single role is assigned multiple tasks, sometimes a role is assigned only a single task, and sometimes the role exists only to send someone a notification.

<b>Role Participant</b>	<b>Principal or Actor Assigned</b>
Product Manager	Administrators
Engineer	Administrators
Purchasing Agent	Administrators
Manufacturing Engineer	Administrators
Change Manager	Administrators
Change Owner	Administrators
Submitter	Creator
ESI Administrator	Administrators
Quality Engineer	Administrators
Production Planner	Administrators
Design Engineer	Administrators
Change Admin III	Administrators

## Change Control Board

The Change Control Board is only an example team template. There are no corresponding out-of-the-box workflow process templates that use the specific set of roles on the team.

Role Participant	Principal or Actor Assigned
Product Manager	Administrators
Change Manager	Administrators
Manufacturing Manager	Administrators
Change Owner	Administrators
Production Planner	Administrators

## Default

The Default team template is empty and has no roles defined. It exists as a default team template that the system automatically chooses if no other team template is designated.

## Team Templates and Object Types

This section describes how team templates are associated with Windchill object types. Object associations determine how teams are automatically created from team templates.

### Out-of-the-Box Associations for Windchill PDMLink

The Windchill PDMLink out-of-the-box team templates are associated with the following object types:

Team Template	Object Types
Change Activity Team	wt.change2.ChangeActivity2
Change Notice Team	wt.change2.ChangeOrder2
Change Request Team	wt.change2.ChangeRequest2 wt.change2.WTChangeProposal
Problem Report Team	wt.change2.ChangeIssue
Default Team	All other objects

With the exception of the Default team, the out-of-the-box team templates are stored in the PDMLink cabinet. The Default team is stored in the System cabinet.

**Note:** The Default team template, which is stored in the System cabinet, does not contain any roles. If your site uses workflows to manage objects (such as documents and parts) other than change objects, you must add roles to the Default team template and any other team templates you create for use with non-change objects.

The change objects listed in the preceding table are the object names used in Windchill. The following table illustrates the correspondence between Windchill Change Objects as displayed in the Type Manager and as displayed in the user interface. ("NA" indicates that the object is not applicable.)

Type Manager (Windchill) Change Object	Windchill Change Object or Field
WTAnalysisActivity	NA
WTChangeActivity2	Change Notice Implementation Task
WTChangeInvestigation	NA
WTChangeIssue	Problem Report (PR)
WTChangeOrder2	Change Notice
WTChangeProposal	Change Request Proposed Solution
WTChangeRequest2	Change Request

## Team Association

To associate a team template with a particular object type, use the Object Initialization Rules Administrator. For additional information, see [Administering Object Initialization Rules](#) in the Administering Containers chapter.

In the Site container, the value Default is set as the default team template for the following object types:

- ManagedBaseline
- WTProductInstance2
- WTProductConfiguration

You can also create organization-, product-, and library-specific team associations by creating a team template in the corresponding context.

## Team Association Rules

When a workflow activity begins, an appropriate team is automatically created, according to the following rules:

1. If you have established a default team template in the product or library context, that team template is used as the basis for any new team.
2. If no corresponding team template exists in the product or library context, the default team template defined in the organization context is used as the basis for the new team.
3. If no corresponding team template exists in the organization context, the team template in the site context is used as the basis for the new team.
4. The team is given a name containing the name of the object for which it is created.

**Note:** The System cabinet/domain is used to store document templates. For each product and library, a context domain is automatically created for storing document templates. For more information about domains, see [Administering Domains and Policies](#) in the Administering Containers chapter.

After a team has been created, users with the necessary permissions can update the team members by clicking the team name. On the **Team** page, click **Update Team** to make changes.

For example, the out-of-the-box team template association for problem report objects is the Problem Report Team. If you were to create a problem report titled My Problem Report in the library called My Library, the team would be created according to the following rules:

1. If a team template called Problem Report Team exists in the My Library context, that team template is used as the basis for the new team.
2. If no Problem Report Team exists in the My Library context, the Problem Report Team in the organization context is used as the basis for the new team.
3. If no Problem Report Team exists in the organization context, the Problem Report Team in the site context is used as the basis for the new team.
4. The team will be named My Problem Report.

Users with the necessary permissions can then modify the My Problem Report team by accessing the **My Problem Report** details page and clicking the team name. On the **Team** page, click **Update Team** to make changes.

## Team Roles Resolution

The primary task in defining a team is selecting roles and mapping them to participants. To understand the concept and purpose of teams you should understand the relationship between teams and life cycles. (For additional information about life cycles, see [Administering Life Cycles](#) in this guide.)

Business objects are associated with life cycles and teams, and roles are selected within these. The primary purpose of a team is to determine who is assigned the roles that are selected in a life cycle, that is, how life cycle roles are *resolved* to *principals* at runtime. Life cycles are more complex than teams, and they require more resources to create and maintain. Therefore, it is generally more efficient to create a relatively small number of life cycles with abstract roles and a larger number of teams that map roles to specific principals, which may change over time. Your site should have a policy regarding your use of teams with life cycles.

As team administrator, you should be familiar with existing life cycles that might be associated with the team you are about to create. You should then select a role for the team for every role that exists in the relevant life cycles.

### Initial Team Creation

#### Windchill ProjectLink

If the object is assigned to a life cycle containing a phase workflow in the initial phase, the object uses the context team as the template. The role-participants of the context team are copied into the team. If the object is assigned to a life cycle that does not contain a workflow process in the phase of the initial state, it is assigned to a default team where the role-participants are meaningless. The team used for workflow and life cycle is created when the object is routed.

#### Windchill PDMLink

If the object is assigned to a team template, the team template is resolved into the team. Roles-principals are added and roles-actor roles are resolved to the users playing the role for the object and added to the team.

### Role Resolution Rules

The property `wt.team.re-resolveRoles` is present in `wt.properties`. You can specify whether or not to have the life cycle roles resolve again when entering a new phase. If you wish to have the roles resolved, set `wt.team.re-resolveRoles=true`. The default behavior does not resolve roles at the new phase. If you do not wish to resolve roles, you do not have to add this property. More information about the role resolution and this property are as follows.

## Default Behavior

For the default, the following list illustrates the order in which Windchill tries to resolve each role in a life cycle:

1. If the life cycle role exists in the team, the life cycle role is resolved to principals (or the actor for the role), *as defined in the team*. All life cycle mapping for that role is overridden by the team values.
2. If the life cycle role *does not* exist in the team (that is, rule 1 does not apply), but the life cycle role is mapped to an existing team role, then the life cycle role is added to the team and resolved to principals, as defined in the team role.
3. If the life cycle role does not exist in the team and is not mapped to a role that exists in the team (that is, rules 1 and 2 do not apply), then the life cycle role is added to the team and resolved to principals, as defined in the life cycle.
4. If the object's context team contains the role, any participants who play the role in the context team that are not already members of the role in the team are added to the team.

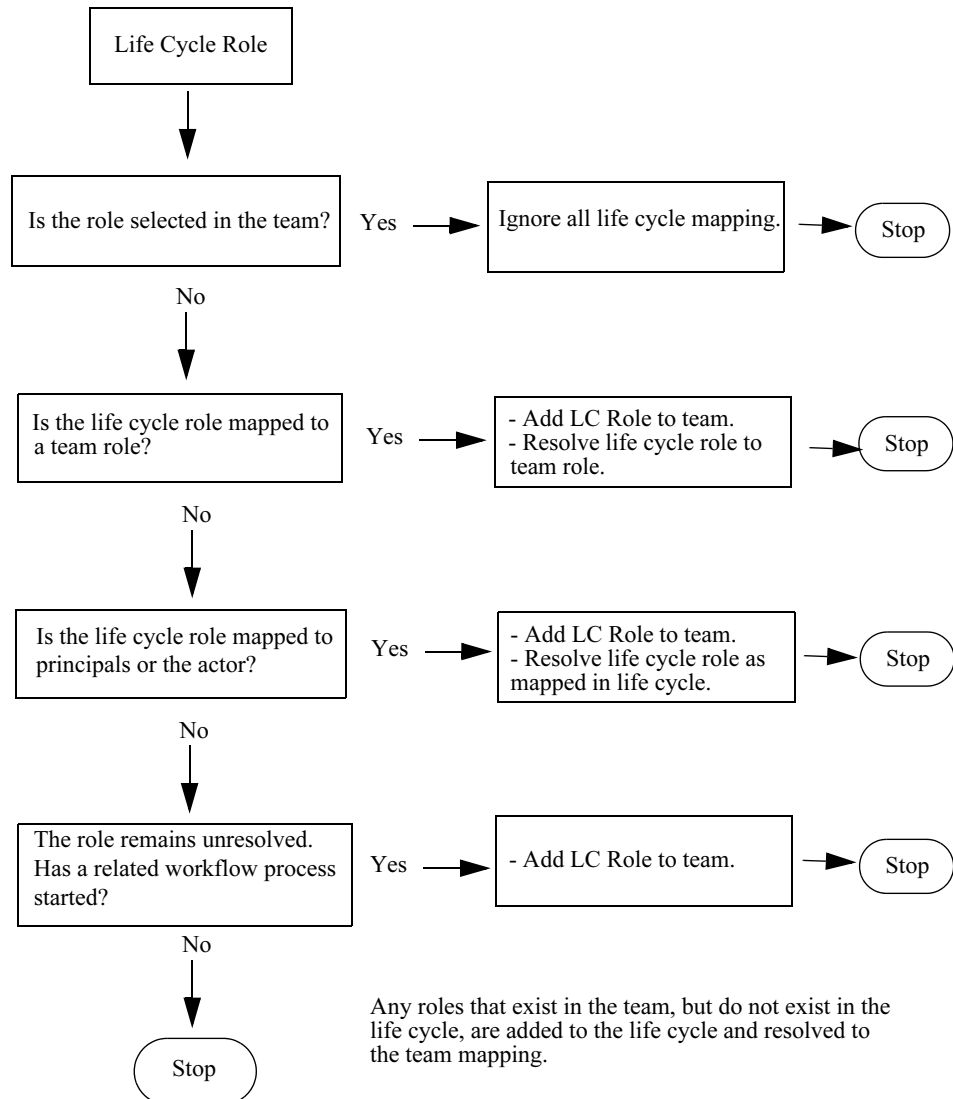
**Note:** Windchill Foundation & PDM does not use context teams.

5. All roles that are not defined in the team, but are used in a related workflow process, are added to the team when the workflow process starts.

If all life cycle roles also exist in the team, they are resolved directly, as defined in the team, without regard to life cycle mapping. This makes it convenient to define relatively few life cycles with abstract roles, which will be resolved to principals that are defined in teams.

The following flow chart illustrates the Windchill business rules for resolving life cycle roles:

**Note:** Although it is possible to define a team that does not map roles to principals, or even to define a team with no role mapping, with typical usage, such a team would be useless.





## Property Set to True

If the property `wt.team.re-resolveRoles=true`, the only difference is the re-resolution of the roles when the object enters a new phase. Otherwise, role resolution will happen for only first phase of first state and not for consequent states.

1. If the life cycle role is mapped to an existing role in the team template, the role is resolved to the members in the team template.
2. If the life cycle role is not mapped to an existing role in the team template, the life cycle roles participants are resolved and added to the team in the role.

Example 1: The life cycle contains Role A and is assigned to Role B. The team contains Role B with member user x. The team template does not contain Role A. Role A is added to the team with user x as the participant.

Example 2: The life cycle contains Role A and is assigned to Role B. The team contains Role B with member user x. The team template contains Role A with user y. Role A is added to the team with user y as the participant.

3. If the object's context team contains the role, any participants who play the role in the context team that are not already members of the role in the team are added to the team.

**Note:** Windchill Foundation & PDM does not use context teams.

4. All roles that are not defined in the team, but are used in a related workflow process, are added to the team when the workflow process starts.

## Role Resolution Example

This section contains an example of how roles might be resolved for an document object that is associated with a team and a life cycle.

The life cycle template contains the following roles:

	1st Phase	2nd Phase	3rd Phase	4th Phase
Submitter	creator	Not in this phase	Not in this phase	Not in this phase
Promoter	Not in this phase	Not in this phase	Not in this phase	Not in this phase
Reviewer	Design Engineer	Project Manager Prod Marketing	QA Engineer Pubs	Jane Design Engineer Project Manager QA Engineer
Observer	Not in this phase	Team Leader	Not in this phase	Not in this phase

The team template contains the following roles/participants:

- Design Engineer--Kristin
- Project Manager--Dave, John
- QA Engineer--Sean, Sachin
- Team Leader--Tom, Beth

The context team contains the following roles/participants:

- Design Engineer--Kristin, Flavio, Bill, Galen
- Project Manager--Dave, John
- Prod Marketing--Chris
- QA Engineer--Sean, Sachin, Iyrena
- Pubs--April, Diane, Muriel
- Team Leader--Tom, Beth

## First Phase

An object is created by Jeff and assigned to the life cycle template and team template above. The context team is list above. The team resolves to the following participants for the first phase:

	<b>Default (property set to false)</b>	<b>Property set to true</b>
Design Engineer	Kristin, <i>Flavio, Bill, Galen</i>	Kristin, <i>Flavio, Bill, Galen</i>
Project Manager	Dave, John	Dave, John
QA Engineer	Sean, Sachin, <i>Iyrena</i>	Sean, Sachin, <i>Iyrena</i>
Team Leader	Tom, Beth	Tom, Beth
Reviewer	Kristin	Kristin
Submitter	Jeff	Jeff
Promoter		

Observations:

- Participants are added during the team template/life cycle role resolution.
- Participants (Flavio, Bill, and Galen) were added to the design engineer role from the container team.

- Roles from the container team that do not exist in the team are not added (in this case, Prod Marketing and Pubs).
- The life cycle Reviewer role is mapped to Design Engineer, but since the container team roles are not added until after the team template/life cycle role resolution is completed, the people in the Design Engineer role are not members of the Reviewer role.

## Second Phase

The object is promoted to the second phase.

	<b>Default (property set to false)</b>	<b>Property set to true</b>
Design Engineer	Kristin, <i>Flavio</i> , <i>Bill</i> , <i>Galen</i>	Kristin, <i>Flavio</i> , <i>Bill</i> , <i>Galen</i>
Project Manager	Dave, John	Dave, John
QA Engineer	Sean, Sachin, Iyrena	Sean, Sachin, Iyrena
Team Leader	Tom, Beth	Tom, Beth
Reviewer	Kristin	Dave, John
Submitter	Jeff	
Promoter		
Observer	Tom, Beth	Tom, Beth

Observations are the same as for the first phase.

## Third Phase

The team templates and container teams are modified.

The team template contains the following roles/participants:

- Design Engineer--Kristin, Flavio
- Project Manager--Dave, John
- QA Engineer--Sean, Sachin, Dan
- Integration--Mark

The context team contains the following roles/participants:

- Design Engineer--Kristin, Flavio, Jeff, Michelle
- Project Manager--Dave, John
- Product Marketing--Chris

- QA Engineer--Sean, Sachin, Iyrena
- Pubs--April, Diane, Muriel
- Team Leader--Tom, Beth
- Observer--Jane, Lynn

	<b>Default (property set to false)</b>	<b>Property set to true</b>
Design Engineer	Kristin, Flavio, Bill, Galen, <i>Jeff, Michelle</i>	Kristin, Flavio, Bill, Galen, <i>Jeff, Michelle</i>
Project Manager	Dave, John	Dave, John
QA Engineer	Sean, Sachin, Iyrena	Sean, Sachin, Iyrena
Reviewer	Kristin	Sean, Sachin, Dan
Submitter	Jeff	
Promoter		
Observer	Tom, Beth	Tom, Beth
Team Leader	Tom, Beth	Tom, Beth

#### Fourth Phase

A set state sets the object to the fourth phase.

	<b>default (property set to false)</b>	<b>property set to true</b>
Design Engineer	Kristin, Flavio, Bill, Galen, Jeff, Michelle	Kristin, Flavio, Bill, Galen, Jeff, Michelle
Project Manager	Dave, John	Dave, John
QA Engineer	Sean, Sachin, Iyrena	Sean, Sachin, Iyrena
Reviewer	Kristin	Jane, Kristin, Flavio, Bill, Galen, Jeff, Michelle, Dave, John, Sean, Sachin, Iyrena
Submitter	Jeff	
Promoter		
Observer	Tom, Beth	Tom, Beth

	default (property set to false)	property set to true
Team Leader	Tom, Beth	Tom, Beth

Observations:

The Design Engineer and Observer do not exist in the life cycle template for the fourth phase, but they do exist in the team and container team. New members in the container team are added, but existing members are never removed.

## Defining Team Properties and Roles

**Note:** The Team Administrator is not accessible from Windchill ProjectLink.

To begin working with team templates in Windchill Foundation & PDM, from the **Business Administrator** page, click **Process Administrator > Team Administrator**.

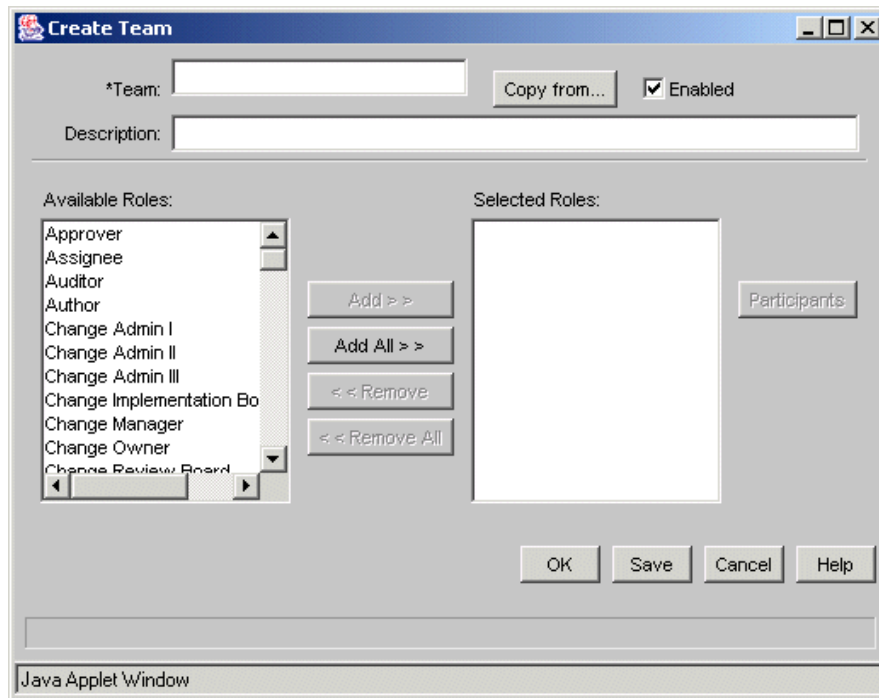
For Windchill PDMLink, you can access the **Team Administrator** page in one of the following ways:

- From the **Product**, **Library**, **Organization**, and **Site** tabs, click **Utilities**. Click **Team Administrator** to access the Team Administrator.
- From the **Product** and **Library** tabs, click **Templates**. On the **Templates** table, select **Team Templates** from the **Current View** drop-down list. Click **Administer Team Templates**.

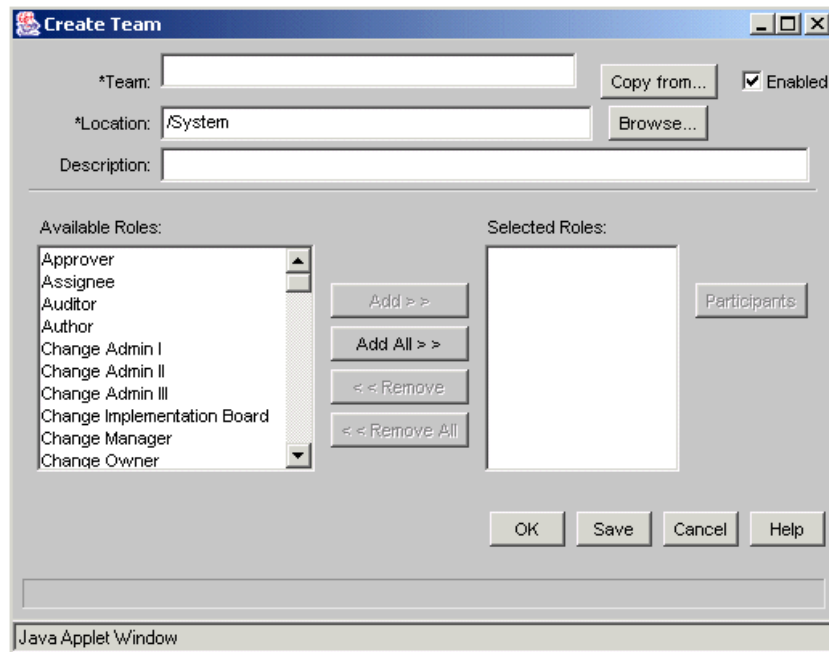
The **Team Administrator** page displays a list of existing teams. The Team Administrator refers to team templates as teams. Use the buttons on this page to create, update, view, delete, rename, and save as a new team. Click **Help** on the Team Administrator page for a description of the buttons and their functions.

Click **Create** or **Update** to access the **Create/Update Team** window.

The following figure shows the **Create Team** window for Windchill PDMLink, with properties entered:



The following figure shows the **Create Team** window for Windchill Foundation & PDM:



Notice there is a **Location** field that was not in the **Create Team** dialog box for Windchill PDMLink.

Teams can be stored in or moved to any cabinet or folder for which you have the required access permissions. Team names must be unique within the context associated with the cabinet or folder in which the team is stored.

The fundamental task in defining a team is selecting the roles that compose the team. The **Available Roles** list is populated with the names of all roles available in the system.

To add a role to a team, select it and click **Add** to move it to the **Selected Roles** list. You can also click **Add All** to move all the available roles to the **Selected Roles** list. Click **Remove** or **Remove All** to delete roles from the team.

Leave the **Enabled** check box selected to make the team selectable. Typically, a team is disabled only when you want to remove it. A disabled team remains in effect for all current usages, but cannot be selected for new objects.

## Predefined Roles

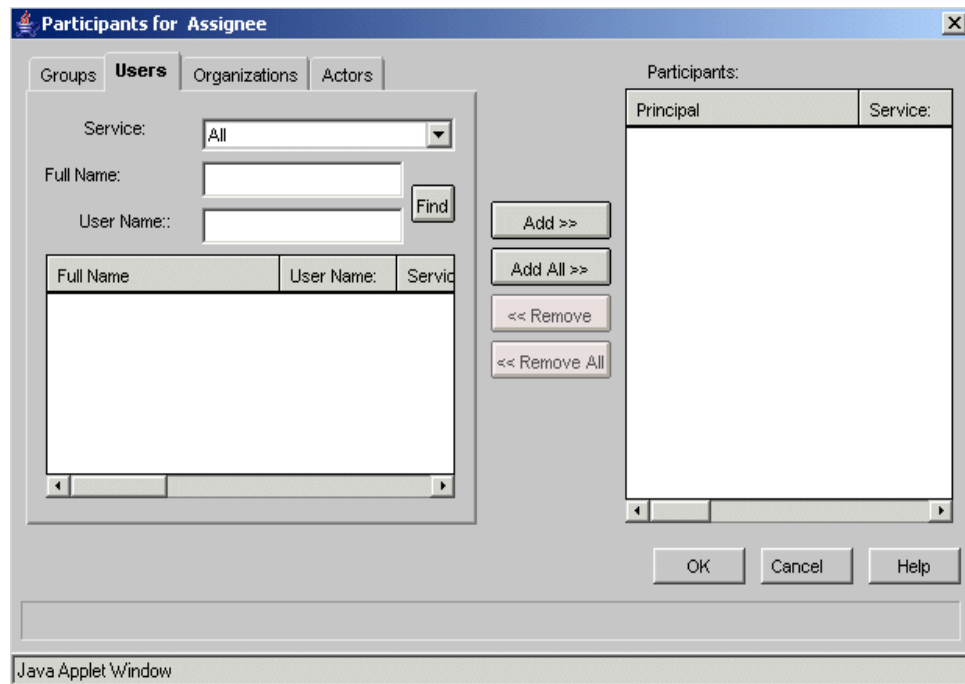
Windchill includes many predefined life cycle states and roles. You can define additional states and roles by adding them to the RoleRB.rinfo resource files. Defined states and roles are added to this list when you recompile the resource files and deploy the class files to your production environment. For additional information, see the enumerated types information in the *Windchill Customizer's Guide*.



**Caution:** When you add a value to an enumerated type (for example, by adding a role in the RoleRB.info resource file), removing that value can result in a serious runtime error. Do not remove a role unless you are certain there is no reference to it.

## Assigning Participants to Team Roles

To associate principals (users, groups, organizations) or actors with a team role, select a role from the **Selected Roles** list in the **Update Team** window, and click **Participants**. This opens the **Participants** selection window, shown in the following figure, where you can map users, groups, organization, or actors to the selected role:



Use the following information to search for groups, users, organizations, and actors:

- To search for groups, click the **Groups** tab. Select **All** or a directory service from the **Service** drop-down list, to narrow your search. Enter the name of the group in the **Group** field, and click **Find**. You can use the \* wild card to search for groups with similar names. The specified services are searched and the matching results display in the table.
- To search for users, click the **Users** tab. Select **All** or a directory service from the **Service** drop-down list to search the entire system. To search for a specific user, enter information in the **Full Name** or **User Name** fields, and click **Find**. You can use the \* wild card to search for users with similar names. The specified services are searched and the matching results display in the table.
- To search for organizations, click the **Organizations** tab. Select **All** or a directory service from the **Service** drop-down list to search the entire system. Enter the name of the organization in the **Organization** field, and click **Find**. You can use the \* wild card to search for organizations with similar names.



The specified services are searched and the matching results display in the table.

- To search for actors, click the **Actors** tab. To assign a role, select an actor. Currently, Creator is the only actor defined. The Creator is resolved at runtime to the user who created the selected object.

To map a participant to a role, select it and click **Add** to move it to the **Participants** list. You can also click **Add All** to move all the displayed users, groups, organizations, or actors to the **Participants** list. Click **Remove** or **Remove All** to delete participants from the list.

For detailed instructions for selecting participants, see the online help by clicking **Help**.

## Best Practices for Windchill PDMLink and Windchill ProjectLink

Site, organization, and product and library administrators manage team templates.

- Site administrators create, modify, delete, and view team templates in the site context.
- Organization administrators create, modify, delete, and view team templates in the given organization context. Organization administrators can view team templates from the site context.
- Product and library managers create, modify, delete, and view team templates in the given application context. They can view team templates from the parent organization context and the site context. This includes administrators of Windchill PDM library contexts.

**Note:** The Team Administrator is not available to administrators of project contexts.

The Team Administrator client displays a table that lists all team templates belonging to the given context plus those belonging to its ancestor contexts. A column in the table identifies the context owning each team template.

For Windchill Foundation & PDM, when you create a team template, you can choose the location of the team template as any of the cabinets or folders available in the Windchill PDM library. In Windchill PDMLink, the system decides the location.

**Note:** Although you are allowed to create team templates using users, it is best to create them with groups.

## Refreshing Groups

When a group is added as a participant of a context team role, the group's members (users) are added individually to the role and tracked independently as participants. This means that if the group membership is changed or the group is deleted, the context teams must be refreshed so that the role participants properly reflect the current members of the group.

In order to effectively refresh the groups that are members of teams, you should consider doing the following:

- Define groups that are common to products, libraries, and projects at the organization level, rather than at the product, library, or project instance level.
- Reference the groups in the definition of each of the context teams, where appropriate.
- When groups are referenced in context teams, refresh the teams periodically to reflect the dynamic group membership.

A site or organization administrator can refresh groups in one of the following ways:

- Manually refresh groups by:
  - Accessing the **Team** page from every project, product or library in which the group is participating, and clicking the **Refresh Groups** action.

This is the only refresh method available if groups are defined in the Site context, and it can also be used to refresh groups defined in an organization context.

The **Refresh Groups** action is only available to site and organization administrators, which means that at least one site or organization administrator must be a participant in every context team. When the membership in a group is changed, an administrator then goes to each context team and performs the **Refresh Groups** action.

- Accessing the **Groups** page of the organization and clicking the **Refresh Teams** action for the updated group.

This refresh method is only available for groups which are defined within an organization context.

- Schedule the automatic update of teams through the use of a queue, where teams for which the group membership has changed can be updated on a daily basis in an off-peak time of your choice. Additionally, a schedule can be set up to recompute the membership of all context teams once a week.

Schedule the refreshing of groups by:

- Enabling the scheduled refresh functionality.

- Setting up a schedule for refreshing the group memberships in context teams.

Using the xconfmanager utility or System Configurator, you can set the following properties in the wt.properties file to turn on the scheduling of team updates:

#### `wt.inf.team.useScheduledRefreshGroups`

Specifies whether or not to use a queue to refresh context teams that contain groups that have been modified. By default, this property is set to false and daily automatic team updates do not happen. Set this property to true to turn on this feature.

When this feature is turned on, all teams containing a modified group get marked as needing to be refreshed and then, at the designated time each day, only those teams are refreshed. By default, the designated time of day is 1 AM (formatted as 01:00:AM). You can use the `wt.inf.team.refreshGroupsDailyQueueTime` property to change the time of day.

#### `wt.inf.team.useScheduledRecompute`

Specifies whether or not to use a queue to recompute the membership of all container teams. By default, this property is set to false and weekly team updates do not happen. Set this property to true to turn on this feature.

When this feature is turned on, all teams are updated at a specified day and time each week. By default, the designated day is Sunday (indicated by 1) and the time of day is 1 AM (formatted as 01:00:AM). You can use the `wt.inf.team.recomputeMembershipDayOfWeek` property to change the day and the `wt.inf.team.recomputeMembershipWeeklyQueueTime` property to change the time of day.

For additional information about the properties, see the property descriptions in the System Configurator or the `properties.html` file.

**Note:** Because recomputing team membership uses a lot of system resources that can slow down system performance, scheduling team updates is the recommended method to use.



# 16

## Administering Workflow Processes

This chapter provides information about workflow processes and the Workflow Administrator.

Topic	Page
Overview .....	16-2
Workflow Iteration .....	16-3
Using the Workflow Process Editor .....	16-3
Import and Export .....	16-30
Process Manager Toolbar Access Control .....	16-32
Viewing Workflow History .....	16-32
Configuring Worklist Fields .....	16-38
Workflow Instance States .....	16-40
Domain-based Workflows .....	16-41
Default Workflow Templates .....	16-42
Workflow Template Execution Flags .....	16-48
Electronic Signatures .....	16-58
Best Practices .....	16-62

## Overview

A *workflow system* gives you the ability to automate procedures in which information, tasks, and documents are passed among participants. This procedure is based on a process composed of well-defined rules, designed to efficiently accomplish your business goals.

This chapter provides information on creating and updating workflow templates, importing and exporting templates, and viewing workflow history. See the end of this chapter for detailed information concerning configuring worklist fields and object subscription code.

The Windchill workflow system, implemented by the Windchill Workflow application, consists of four components:

- The Workflow Process Editor, which allows you to define a workflow process and save your definition as a process template. This graphical editor and its use are the focus of this chapter.
- The workflow runtime system, which executes a defined workflow process within the context of a specific business object (for example, a part or a document). Process execution includes delivering work items to users participating in the process, opening applications (for example, automatically interacting with the Windchill Explorer to check a business object out of the database), initiating subprocesses, and so on. The way in which the end user interacts with a running workflow process is described in detail in the *Windchill User's Guide*.
- The Workflow Process Manager, a graphical tool for monitoring and reporting on workflow processes. (For access control information, see [Process Manager Toolbar Access Control](#), in this chapter.)
- The Workflow History Viewer, which provides a simple ASCII interface used to access recorded workflow events, such as state changes, data transfers, or process start. The information in [Viewing Workflow History](#) can assist you in optimizing or streamlining a workflow process.

## Workflow Iteration

Administering workflow templates is an iterative process. Like version-controlled objects, iterated objects are checked in and out of shared locations. However, unlike version-controlled objects, they cannot be revised. Instead, any change to an object creates a new and separate iteration when it is checked in. Earlier iterations, which may still be in use, are unchanged and unaffected by the new iteration. Only the latest iteration is available for new uses.

To make changes to a workflow template, you must check out a copy. (Clicking **Update** on the Workflow Administrator page automatically checks out a copy of the selected template.) While it is checked out, no one else can check out a copy, but processes can still be initiated based on the current template. When you have completed changes to the checked-out copy, you must save it and check it in to make it available to others. It then becomes the latest iteration. Running processes that use an earlier iteration continue to run, unaffected by the newer iteration.

## Testing an Updated Workflow Process Template

Under normal circumstances, when you initiate a process template, the latest iteration in the shared location is used; however, if you have the template checked out and stored in your personal cabinet, the working copy is used. This makes it possible to test a workflow process template before checking it in.

To update and test a workflow process template, follow this procedure:

1. Check out a copy of a workflow template.
2. Update the copy, and save it to your personal cabinet.
3. Initiate a workflow process, based on that template.

The updated copy of the template in your personal cabinet is then used, rather than the current, checked-out iteration in the System cabinet.

You must either complete the running process, or stop and delete it before you can check in the updated workflow process template or undo the check out.

If you attempt to check in the updated process template or undo the checkout while the process instance is running, the following error message is displayed:

```
Can't modify or delete template, <template name>; there are open instances.
```

## Using the Workflow Process Editor

The Workflow Process Editor is a graphical interface for defining workflow processes that range from the simple to the highly complex. It features a large set of predefined activity nodes that you can place and connect. The Process Editor supports nested processes, branching, merging, loops for iterative activities, and defining assigned activities.

How you access the Workflow Administrator is determined by your Windchill solution:

- From Windchill Foundation & PDM, you can access the Workflow Administrator by clicking **Process Administrator** on the **Business Administration** home page, then click **Workflow Administrator**.
- From Windchill PDMLink, you can access the Workflow Administrator by clicking the **Workflow Administrator** link on the **Utilities** pages that are under the **Site**, **Organization**, **Product**, and **Library** tabs. The **Workflow Administrator** link from the **Site** tab provides you (as the system administrator) with unrestricted access to all workflow templates from the site context. The link from the **Organization** tab provides access to the organization-level and site-level templates. The link from the **Product** and **Library** tabs provides access to those templates that are in the context that is active, plus access to the organization-level and site-level workflow templates.
- From Windchill ProjectLink, you can access the Workflow Administrator from the **Utilities** pages that are under the **Site and Organization** tabs.

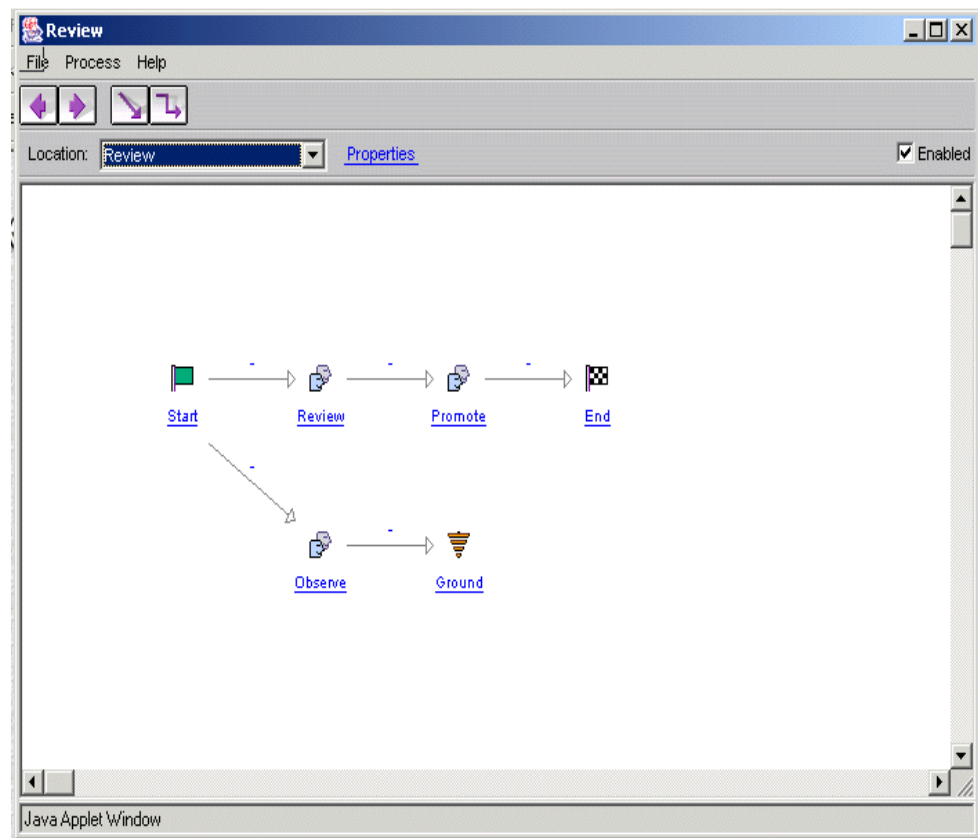
## Working with Workflow Templates

The **Workflow Administrator** page opens, displaying a list of existing workflow templates, with their locations enabled status, and context. Using the buttons on this page, you can create, update, view, and delete templates, as well other activities, including importing and exporting workflows. Click **Create** or **Update** to access the Workflow Process Editor. When you update a workflow process, it is automatically checked out if it is in a vault.

For detailed help on using the **Workflow Administrator**, click **Help**.

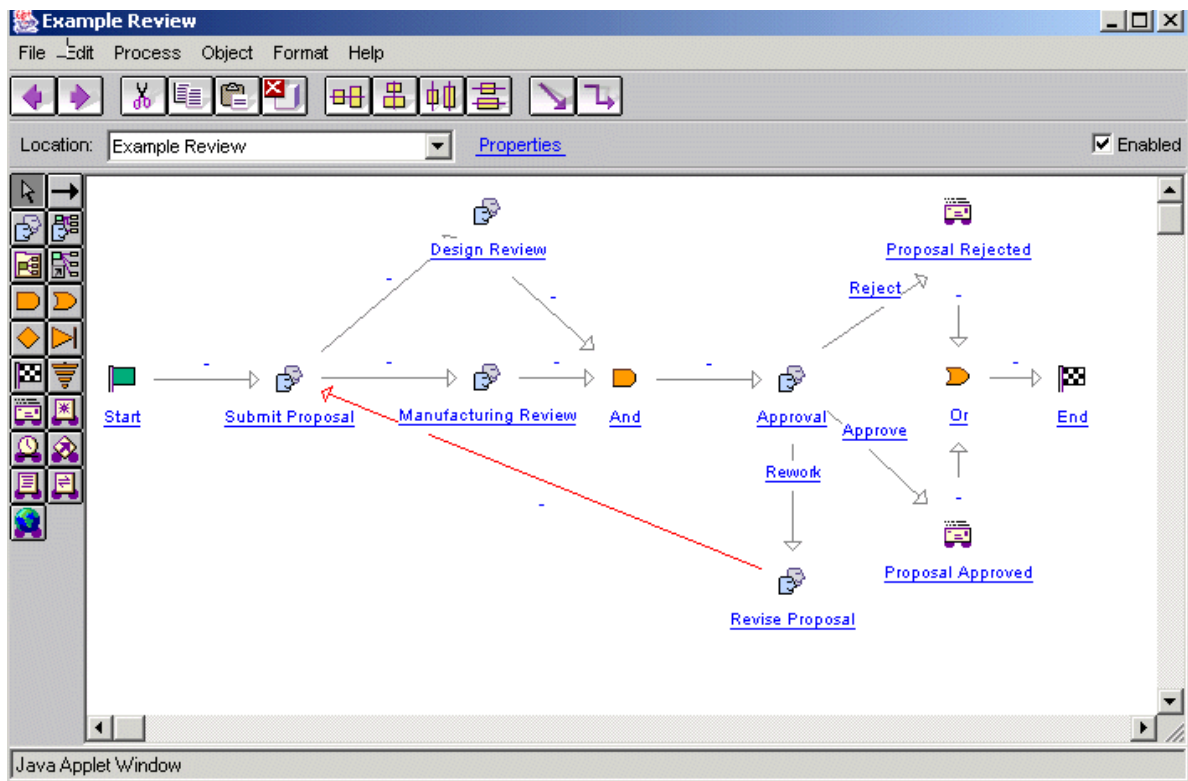


The following figure shows the Workflow Process Editor displaying a Review process, (one of the default workflow templates included in an out-of-the-box system):

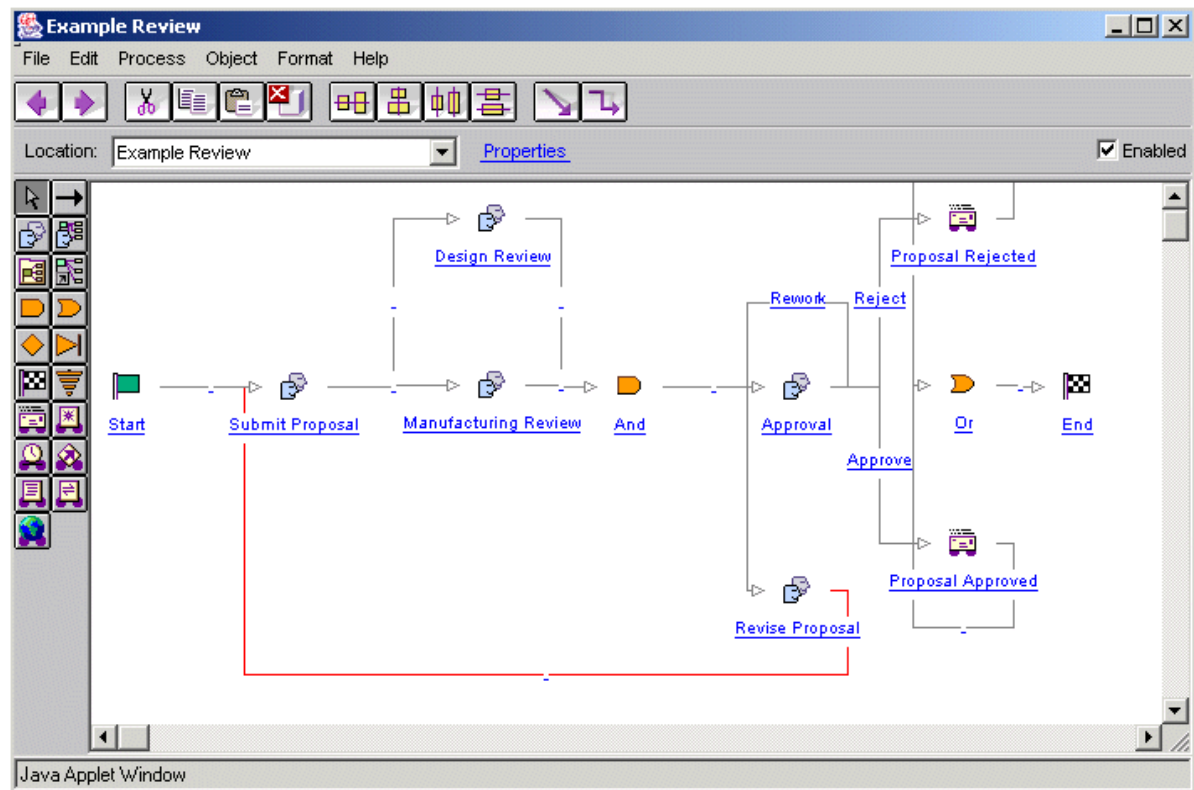


If you click **Create**, the Workflow Process Editor displays only the Start node.

You can display the workflow process with *straight lines* at any angle or with *square lines* – horizontal and vertical lines and right angles. Click **Straight Lines** to use straight lines at any angle, as in the following figure:



Click **Square Lines** to use horizontal and vertical lines at right angles, as in the following figure:



To verify that your process definition is correct, select **Validate All** from the **Process** menu. The **Validate** window either confirms the process definition or identifies dangling activities or malformed processes.

A Java compiler is integrated with the Process Editor to support expressions of arbitrary complexity. Workflow routing functionality includes links and event triggers. Events that cause a link to *fire* are displayed on the link itself, so anyone viewing the process definition can easily understand and verify the process behavior. For example, the Approve and Revise events in the previous straight line and square line examples are events that cause a link to fire.

When you have completed your process definition, it is saved in your personal folder. To change a process definition that has been saved in a vault, it must be first checked out, and then checked in.

Updating workflow processes is an iterative process. A new iteration is created when an update is checked in. You can view iterations on the **Workflow Administrator** page.

The following sections describe the tools and components available to help you define a workflow process in Windchill.

## Navigating a Process Diagram

The Workflow Process Editor is designed for easy navigation of processes and their subprocesses, using common Web navigation techniques. For example, you can edit a subprocess diagram, by clicking a subprocess hyperlink.

To navigate between a parent process and a subprocess, use either the **Back** and **Forward** buttons, or the drop-down list in the **Location** field.

The title bar of the Workflow Process Editor displays the name of the process or subprocess you are currently editing.

Hyperlinks display the properties of each activity type and link, for example:

- Click an activity node hyperlink to open the properties window for that activity. You can then create, update, or view the properties that define the node's behavior.
- Click the Properties hyperlink (at the right of the **Location** drop-down menu) to view and edit the properties of the process itself.
- Double-click the link that connects a node to open a window to map events that are *broadcast* (or *emitted*) from the preceding activity to actions in the succeeding activity. By default, the completion event for a given task triggers the start of all successor tasks.

## Placing Process Nodes

You can build a process definition by adding, selecting, and linking nodes that are represented by icons at the left of the workflow process editor.

- To add nodes to the process definition, select the appropriate icon and then click on open space within the process diagram.
- To select a node and display its properties, click the node hyperlink.
- To link two nodes, click the **Action** icon (a right arrow). Click and drag from the first node to the second node and release the mouse button. (A line with an arrow appears linking the nodes. The first node does not move to the second.)

The following list describes the process nodes that can be added to your process definition. The list is displayed in order of each icon's appearance on the Process Editor.

- The *Assigned Activity* is an activity assigned to one or more users or user groups or an actor to perform.
- The *Ad Hoc Activity* is assigned to a user to define a group of activities at runtime. The group of activities is similar to a simple block.
- The *Block* represents a group of activities, connectors, or robots. You can reduce the complexity of a process by creating blocks of activities that can be expanded when needed.

- The *Proxy Process* is a subprocess embedded within the main parent process, which can be nested to reduce complexity and provide reuse.
- The *And Connector* fires when all the predecessor links have fired, but not before.
- The *Or Connector* fires when any one of the predecessor links has fired. Preceding activities are terminated if **Terminate Open Predecessor Activities when Fired** has been selected.
- The *Conditional Router* allows you to branch a process based on a conditional expression.
- The *Threshold Connector* fires when a user-defined number of predecessor links have fired. Preceding activities are terminated if **Terminate Open Predecessor Activities when Fired** has been selected.
- The *End* stops the process. All process activities should eventually be connected to an end.
- The *Ground* stops a parallel branch of activities within the process, but it does not stop the process.
- The *Notification Robot* notifies the appropriate user with a user-defined e-mail. You can use braces to delimit variables created for the process or node, for example, {varname}. Use back slashes to escape the delimiter, for example, \{{varname}}\.
- The *Method Robot* represents one of several single actions performed when adding the robot to the process. No other configuration is required. The following table lists the robot actions:

Robot	Description
Checkin	Checks in the primary business object to the Windchill database.
Checkout	Checks out a business object to the specified user. For example, you can use the Checkout robot to automatically check a part out to the engineer assigned the task of applying changes after a design review cycle is complete.
Demote	Causes the primary business object to transition to a predecessor phase, with an associated state change, and the application of new business rules (such as those for access control).
Deny	Removes the primary business object from the gate and returns it to the submitter.

Robot	Description
Drop	Causes an object to be removed from its current life cycle and sets its state to <i>dropped</i> . For example, you could have a process branch in which two vendors submit bids for review. These bids could be entered into the database as Windchill documents, which would move through a review and approval process by application of a process definition. In this case, your process may require that, when one bid is approved, its document object is automatically promoted to its next life cycle phase, while the document containing the rejected bid is dropped from its life cycle and goes no further.
Promote	Causes the primary business object to transition to a successor phase, with an associated state change and the application of new business rules, such as those for access control. For example, you could define a process in which an object is automatically promoted to the next phase in its life cycle, if a specific user approves the object. In this case, you could add the Promote robot to your process definition to execute all of the actions associated with an object's promotion.
Set State	Sets a Life Cycle–managed object to an ordinal state or a specific state. The ordinal state is entered as any non-zero integer. The specific state is selected from those defined in the <code>wt.lifecycle.StateRB</code> enumerated type.
Submit	Moves the business object associated with this process to the gate for its current life cycle phase. After a submit, an object awaits promotion to the next life cycle phase. For example, you could add the Submit robot to a process definition to indicate that, when a user creates a change request, it is automatically submitted for promotion to the Open state.

- The *Timer Robot* delays the start of an activity by a specified amount of time, based on the time it is fired or the time the parent process is begun.
- The *Launch Application Robot* executes system commands on the server. These commands are executed using the `Java runtime.exe` command. The execution can be either synchronous or asynchronous.
- The *Execute Expression Robot* enters a synchronous Java expression to be executed in a workflow. By default, the expression returns `true`. A return of `false` indicates a problem during execution, and an exception is thrown on the server.

- The *Synchronization Robot* synchronizes the start of an activity or process with events that are not time related. You can set the robot to start an activity when certain generic external or Windchill-keyed events occur.

If synchronization does not work, make sure the Windchill time is set to the correct time zone. See the Managing User Preferences section of the *Windchill System Administrator's Guide* for more information.

- The *URL Robot* executes a URL to communicate with another server, for such purposes as initiating various Info\*Engine tasks or providing information necessary to complete workflow tasks. It can initiate an operation or retrieve status information to be collected in a string variable. HTML links to binary objects, such as graphics, can be retrieved, although the objects, themselves, cannot.

You can specify the results of a failure by the robot to execute the URL. The following list of error codes may be helpful. For instructions see the help file.

- *400 Bad Request*: Request has not been recognized by the server, because of incorrect syntax. The client should not repeat the request.
- *401 Unauthorized*: Request requires user authentication. Under normal usage, the URL robot does not support authentication. Request should not be repeated.
- *403 Forbidden*: Request has been recognized but the server has refused to honor it. Authentication was not the reason. The request should not be repeated.
- *404 Not Found*: Server has found no match for the Request-URI. May be temporary or permanent. Repeat of request may be appropriate.
- *500 Server Error*: Server has encountered an unexpected condition, which prevents it from fulfilling the request. Repeat of request may be appropriate.
- *501 Not Implemented/Internal Error*: Server does not support the functionality required to fulfill the request. Request should not be repeated.
- *503 Service Unavailable*: Server is temporarily unable to handle the request. Repeat of request may be appropriate.
- *504 Gateway Timeout*: Server has not received a timely response from the upstream server specified by the URI. Repeat of request may be appropriate.

For more information on error messages, see the Internet standards at W3C HTTP RFC (<http://www.w3.org>).

## Declaring Variables

When you define a process, variables can be used within transition condition or automatic routing expressions. Variables can be either global (applicable to the process itself) or local (applicable to an assigned activity or a subprocess).

Variables can be declared as any Java type or as any Windchill class. The only restriction is that the variable must be Serializable. If the variable is typed as a Windchill business object, attributes of that object can be referenced through standard getter APIs. Variables can be declared as follows:

- Visible or invisible
- Required or optional
- Read only or read/write
- Resettable or Static

Variable values can be initialized from parent process variables when an activity or subprocess is started and can also be copied into parent process variables when the activity or subprocess is complete.

**Note:** Although you can declare a workflow variable to be any Java type or Windchill class (provided it is Serializable), PTC encourages you to consider the following guidelines:

- Workflow variables are composited within a ProcessData object, which in turn is persisted in the Oracle database in a BLOB column. If the persistent signature of a BLOB object changes after it is stored in the database, an error may be thrown at the time it is retrieved from the database and de-serialized; therefore, PTC recommends that Workflow Variable types be restricted to the following:

- java primitives
- Serializables with stable persistent signatures
- Evolvables
- Persistables

Evolvables provide methods to handle changes in their persistent signatures; and Persistables are stored in workflow variables as ObjectReferences (not the fully inflated object).

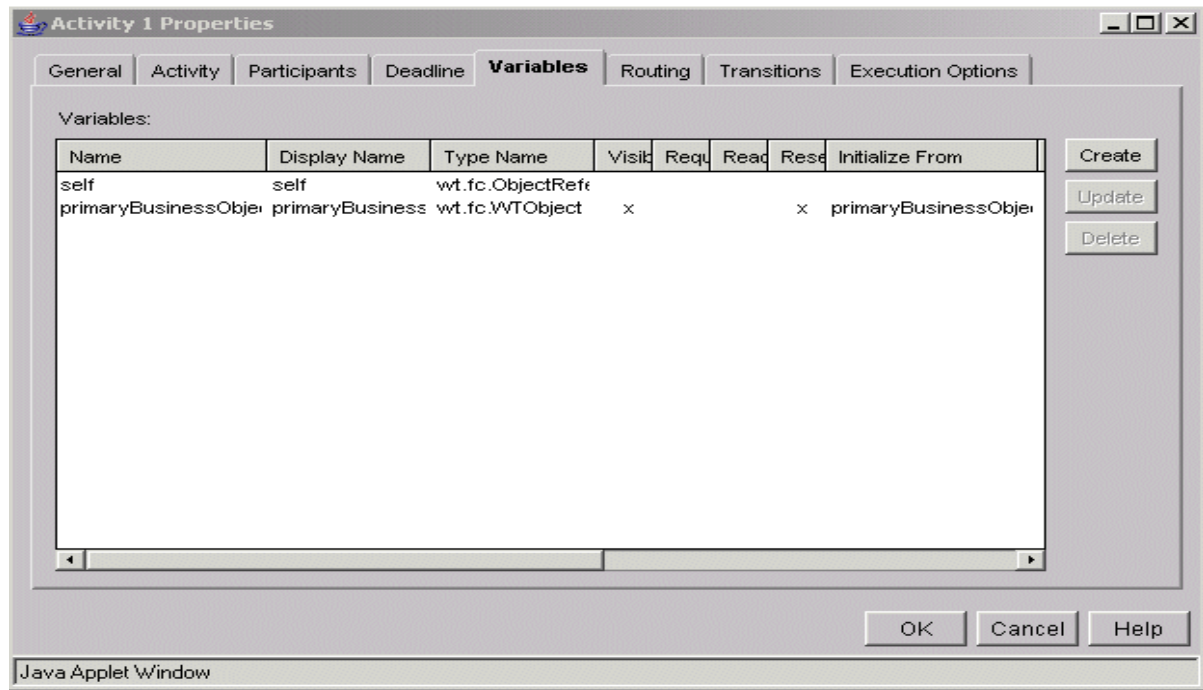
- Persistables should not be composited within another class as this may lead to data inconsistency or corruption. Doing this will also lead to a `wt.pds.NotEvolvableException` during migration.
- Persistables may be an element in a `java.util.Vector` (this is because the `Vector` class receives special handling by the `WfVariable` class), but this should be avoided if possible. Use of a `CachedObjectReference` or `ObjectReference` pointing to a `Persistable` is the preferred practice.



- A Workflow Variable can contain, composite, or otherwise encapsulate a ObjectReference to a Persistable as necessary.

For information on the Windchill Evolvable classes and interfaces, see the *Windchill Customizer's Guide*.

The following figure shows the **Variables** tab panel on the properties window for an assigned activity named Activity 1.



To define additional activity variables, click **Create**. The **Create Variables** window opens:

The screenshot shows the 'Create Variable' dialog box. It includes a 'Name' text field, a 'Type Name' dropdown menu (currently showing 'int'), and a 'Define Display Name' button. Below these are checkboxes for 'Visible' (checked), 'Required' (unchecked), 'Read Only' (unchecked), and 'Resetable' (checked). There are also 'Initialize From' and 'Copy Into' dropdown menus, and a 'Default Value' text field. A 'Permissions' section contains a list of checkboxes: 'Full Control (All)', 'Read', 'Modify', 'Create', 'Revise', 'New View Version', 'Delete', 'Change Permissions', and 'Administrative'. At the bottom right are 'OK', 'Cancel', and 'Help' buttons. The status bar at the bottom indicates 'Java Applet Window'.

To define a display name to be displayed for the variable, click **Define Display Name**.

The online help file for this window provides detailed instructions for variable declaration.

## Defining an Assigned Activity

An assigned activity is assigned to a specific user or group of users when an instance of this process definition is running. When you define an assigned activity, you specify a task that the selected user is to perform as part of a workflow process.

In Windchill Foundation & PDM, this task is added to the user's worklist. When the assigned activity is executed. See the *Windchill Foundation & PDM User's Guide* for information about worklists. In Windchill PDMLink and Windchill ProjectLink, the task is added to the user's Assignments table. See the appropriate user's guide for more information about Assignments tables.

The following figure shows the properties window for an assigned activity, with the **General** tab panel forward:

The screenshot shows a Java Applet Window titled "Activity 1 Properties". It has a tabbed interface with the following tabs: General, Activity, Participants, Deadline, Variables, Routing, Transitions, and Execution Options. The "General" tab is selected. It contains the following fields:

- \*Name: A text box containing "Activity 1".
- Category: A drop-down menu showing "Default".
- Responsible Role: A drop-down menu showing "Process Initiator".
- Description: A large text area.

At the bottom right are buttons for "OK", "Cancel", and "Help". At the bottom left, it says "Java Applet Window".

The assigned activity properties are described in the sections that follow.

## General

On the **General** tab panel, shown above, you can specify a name, a category, a responsible role, and a description for the assigned activity. Name is the only required property.

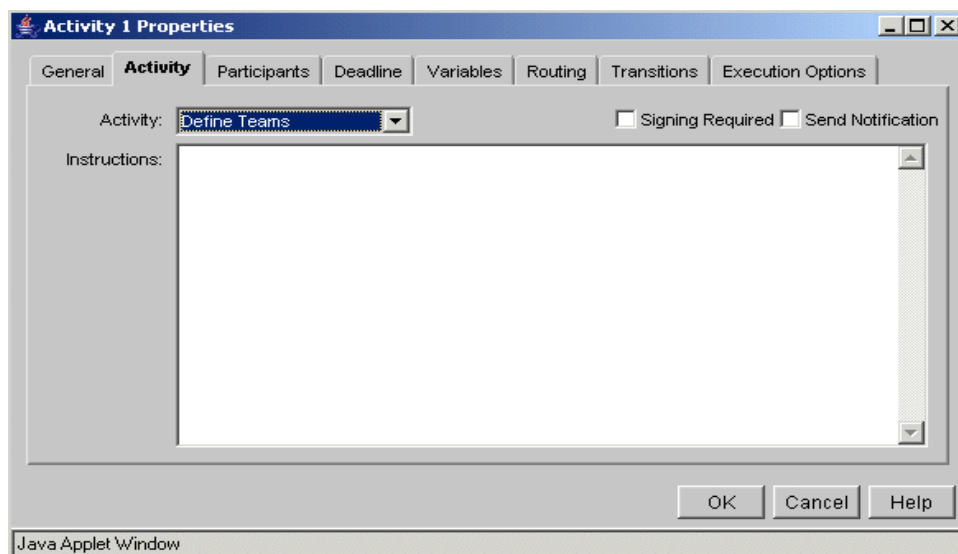
From the **Category** drop-down list you can categorize the activity. For example, you can have activity categories that reflect a team or product type. Windchill provides several predefined categories. Categories are an enumerated type, and you can define additional categories in the `wt.workflow.definer.WfTemplateCategory` file. For more information about enumerated types, see the *Windchill Customizer's Guide*.

## Activity

On the **Activity** tab panel, shown below, you can specify a task for a user or group to perform. Select a task from the drop-down list and, if desired, provide instructions in the **Instructions** text area. You can use braces to delimit variables, for example, `{varname}`. Use back slashes to escape the delimiter, as shown in the following example:

```
\{{varname}}\
```

To create a Define Teams activity, select **Define Teams** from the **Activity** drop-down list. The workflow activity must also have a wt.team.Team variable created for use by this activity. Generally, you can do this by creating a process level wt.team.Team variable and copy to assign the process level variable to the wt.team.Team variable defined in the activity.



When the workflow is assigned, you can create a new instance of a team that gets assigned to the wt.team.Team variable you created. You must determine the usage of the variable when it is assigned the instance of the Team object.

If you want users to be notified by e-mail of the task assignment, check **Send Notification**. E-mail notification is optional, as tasks are always added to the appropriate user's worklist (for Windchill Foundation & PDM) or Assignment table (for Windchill PDMLink or Windchill ProjectLink).

If the **Signing Required** check box is displayed in the **Activity** tab, select the check box if you want an electronic signature required for the activity to be complete. For more information about electronic signatures, see [Requiring Electronic Signatures in a Workflow](#).

## Participants

From the **Participants** panel, shown below, you can assign users, user groups, roles, actors, teams, or variables, to complete the assigned activity:

The screenshot shows the 'Activity 1 Properties' dialog box with the 'Participants' tab selected. The 'Groups' list on the left includes 'Attribute Administrators', which is highlighted. The 'Assignee' table in the center lists 'Attribute Administrator' as a 'Group' with a checked 'Required' checkbox. The 'Required' dropdown on the right is set to 'Any'.

Assignee	Type	Required
Creator	Actor	<input type="checkbox"/>
Attribute Administrator	Group	<input checked="" type="checkbox"/>

Although you have the option to assign an activity task to actual users or user groups as part of the process definition, you may find it more useful to select actors, roles, or teams as participants, which are then mapped to users or groups when an instance of the process definition is instantiated. Assigning participants in this way provides more flexibility and promotes reuse of a process definition in a variety of contexts.

If you choose a role, such as Submitter, as the participant to whom the task will be assigned, the role is resolved at runtime by mapping it to a participant specified in a life cycle or a team (usually a team). For example, if the workflow process is applied to a specific document, the Submitter role in the relevant phase of the document's life cycle can be mapped to a Team Leader role in the team to which the document is assigned. The actual user to whom Team Leader is mapped then receives the task in his or her worklist (for Windchill Foundation & PDM) or Assignment table (for Windchill PDMLink or Windchill ProjectLink) when this activity is fired. For more information on role mapping, with an extensive example, see [Administering Teams and Roles](#).

The selected assignees are displayed on a table, which specifies the participant types. You can designate whether a specific participant is required to complete the task or what conditions must be met to meet a requirement. (That is, any participant, all participants, or a specified number, must complete the activity.)

The **Required** drop-down list is not displayed for users or actors or if the assignee is not required to complete the activity.

If a task is assigned to a group, the task appears in the worklists (or Assignment table) of all group members. (However, one group member can choose to accept the task on behalf of the group.) By default, the person creating the assigned activity is defined as the user who is assigned the task.

To find a specific user, you can search the entire system, or narrow your search to your local system or a specific federated services or user group.

If you want to map a role to a team other than the one associated with the process, you can select the team from the drop-down list at the bottom of the panel. This exception applies only to this specific activity.

You can also map a task role to an actor, that is, someone who performs a specific action within the context of the business object. Currently, Creator is the only actor defined.

**Note:** Click **Help** on the **Participants** tab panel for a help file, which documents specific procedures for assigning participants.

## Deadline

On the **Deadline** tab panel, shown below, you can set the time that the activity is due. You can set the deadline in relation to the start of the activity or the start of the parent process. If you define them both, the earliest deadline applies. You can also designate the consequences and be notified if the deadline is overdue.

**Activity 1 Properties**

General | Activity | Participants | **Deadline** | Variables | Routing | Transitions | Execution Options

Set deadline to [ ] days, [ ] hours, and [ ] minutes from the start of the activity, or  
[ ] days, [ ] hours, and [ ] minutes from the start of the parent process.

Overdue Consequences

☐ Skip ☐ Mark complete ☐ Reassign to the responsible role

Overdue Notifications

☐ Notify the selected role ☐ Notify the responsible role:

Approver  
Approver - Lev  
Approver - Lev  
Auditor  
Author  
Assignee  
Change Manag  
Change Owner

[ ] days, [ ] hours, and [ ] minutes before the deadline

☐ Notify the responsible role:

[ ] days, [ ] hours, and [ ] minutes after the deadline

OK Cancel Help

Java Applet Window

To designate the consequences of a missed deadline, select one of the following check boxes:

- Skip
- Mark complete
- Reassign to the responsible role

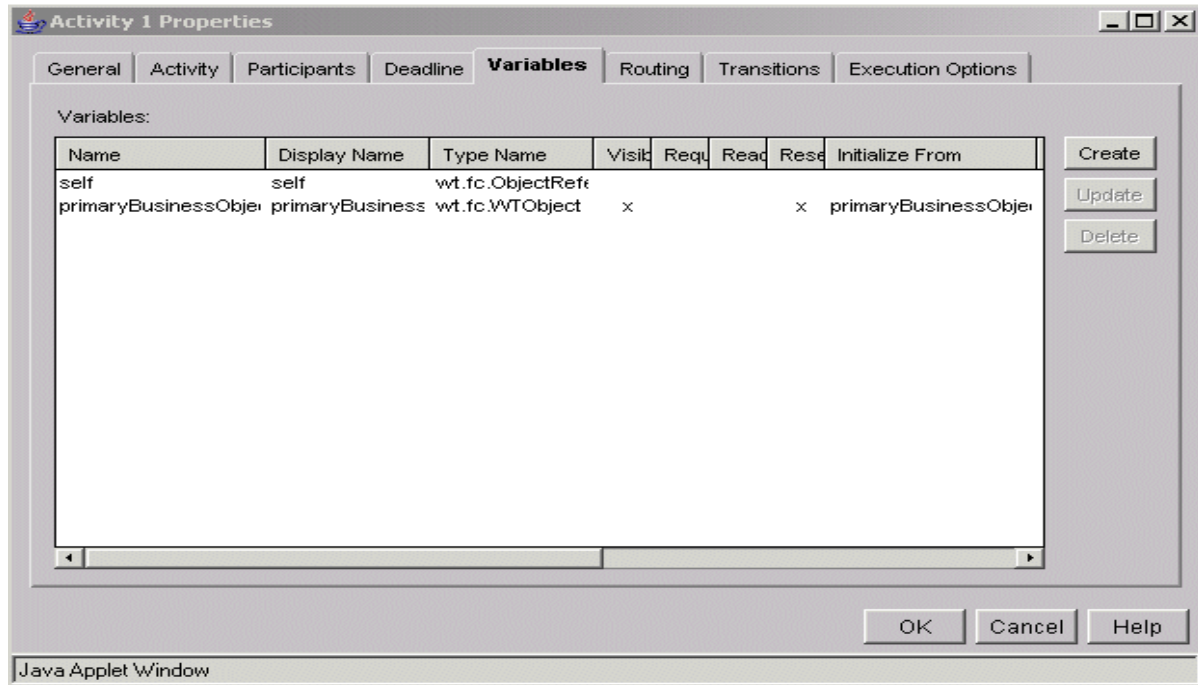
To designate who is notified and when, select any number of the following check boxes:

- To notify the user assigned to a specific role (in addition to the responsible role) when the activity is overdue, select the check box labeled **Notify the selected roles**. Then select any number of roles.
- To notify the responsible role before the activity deadline, select the upper check box labeled **Notify the responsible role** and fill in the amount of days, hours, and/or minutes before the deadline that you want notification sent.
- To notify the responsible role after the activity deadline, select the lower check box labeled **Notify the responsible role** and fill in the amount of days, hours, and/or minutes after the deadline that you want notification sent.

If you do not designate the time, notification is sent to the responsible role when the deadline is reached, even if you do not select a check box. If you select both responsible role notification check boxes and fill in times, notification is sent both before and after the deadline.

## Variables

On the **Variables** tab panel, shown below, you can declare additional variables, or you can update or delete existing variables. The variables defined for this assigned activity are listed in the **Variables** text area.



There are two default variables for assigned activities: *self* and *primaryBusinessObject*. The variable *self* refers to the assigned activity object at runtime. The variable *PrimaryBusinessObject* holds the business object associated with the workflow process at runtime.

See [Declaring Variables](#) for information about adding other variables in this tab panel.

## Routing

On the **Routing** tab panel, you can specify custom routing events. These events are used to control the process flow by mapping one of the events you define to an action that will be performed in a successor activity, via a link.

**Note:** Windchill ProjectLink workflows present routing choices (either manual or exclusive) only to required participants. Non-required participants can see the task and all its information, but the task does not have check boxes or radio buttons or any of the associated text with the routing selection. The task complete button is available for all assignees.

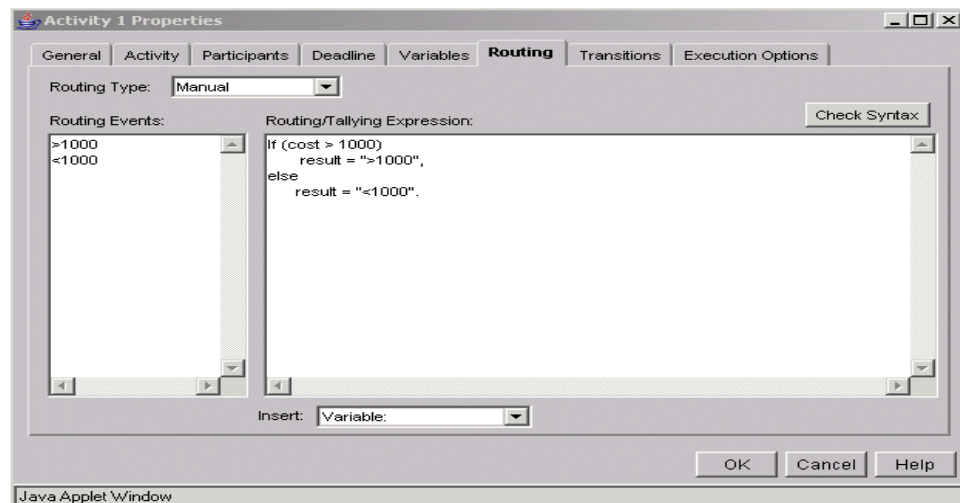


Routing events are defined in the **Routing Events** text area, where you enter the name of the events (one event per line). Click **Check Syntax** to verify that the Java code you have entered is correct.

You can select any of the following routing types from the drop-down list:

- **None** for no routing. The **Routing Events** and **Routing Expression** text areas are not available.
- **Conditional** to automatically trigger the specified event.
- **Manual** to allow a user to select one or more of the routing events specified on this tab.
- **Manual exclusive** to allow a user to select one and only one of the routing events specified on this tab.

Conditional events require a firing expression to automatically fire the event. As shown in the following example, this expression is a fragment of Java code that sets a variable *result* to one of the custom routing events. The *result* variable is required and must return a string matching one of the event names. The expression can reference any variable defined in the **Variables** tab panel.



If this information on the example were entered in the **Routing Events** panel, the >1000 event would be emitted if the result variable for the cost of the object associated with the process (for example, a change order) has a value greater than \$1000. Likewise, the <=1000 event would be emitted if the *result* variable has a value less than or equal to \$1000. One link coming from the activity can be configured to start another activity that would be assigned to a user who would review costs when the >1000 event was emitted (that is, when a change order required an expenditure greater than \$1000). Another link can be configured to simply continue with the process when the <=1000 event is emitted.

If you do not specify an automatic event firing expression, the user who is assigned the task chooses a custom routing events. The activity emits the event that the user chooses.

**Note:** The online help file, accessed from the **Routing** tab panel, contains an extensive set of tally examples and a link to an HTML file that contains additional code samples.

## Transitions

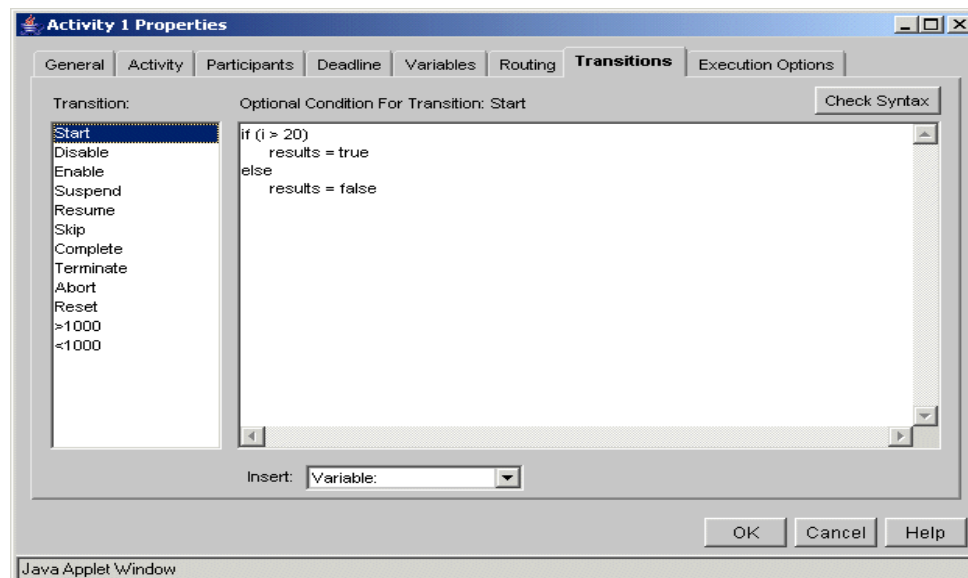
On the **Transitions** tab panel, shown below, you can define the conditions necessary for moving from one internal state to another within a workflow process. Each assigned activity defines transitions.

For example, initiation of a particular assigned activity represents a transition. A state transition can result from a routing decision made by the workflow process while it is running.

Each transition can have an associated condition. If this condition is TRUE, the transition succeeds. Otherwise, it does not. These conditions are defined in the **Transitions** tab panel.

To add a condition to a transition, select it from the **Transition** list, and type an expression in the condition text area. The condition is a standard Java expression that sets the `result` variable to TRUE if the transition should proceed, or to FALSE if it should not. Click **Check Syntax** to verify that the Java code you have entered is correct.

For example, you might want the process to start an assigned activity only if a variable (`i`) is set to a certain value. Therefore, you would select **Start** from the **Transition** list and enter the expression shown as the condition:



## Execution Options

On the **Execution Options** tab, which follows, you can specify the consequences of an error, including who is notified.

**Activity 1 Properties**

General Activity Participants Deadline Variables Routing Transitions **Execution Options**

Execution Properties

☐ Notify the responsible role if there is an error

☐ Notify the responsible role on abort

☐ Abort if there is an error

☐ Abort the parent process (if any) on abort

☐ Record variable changes

☐ Record task reassignment

☐ Record voting

☒ Ignore unresolved role

OK Cancel Help

Java Applet Window

The following table describes the check boxes you may want to select:


Check box	Description
<b>Notify the responsible role if there is an error</b>	Sends a notification to the responsible role identified for the activity in the case in which an unexpected error occurs in the activity instance. This value is checked by default and the initiator of the workflow process is defined as the user associated with the responsible role if no other role is identified.

Check box	Description
<b>Notify the responsible role on abort</b>	<p>Sends a notification to the responsible role identified for the activity in the case in which an unexpected error occurs in the activity instance. This value is checked by default and the initiator of the workflow process is defined as the user associated with the responsible role if no other role is identified.</p> <p>If the activity is aborted, and there are following activities in the process that depend on the completion of this preceding activity, the following activities linked to this preceding activity will NOT start.</p>
<b>Abort if there is an error</b>	<p>Automatically aborts the activity if an unexpected error occurs when the activity is executing. If the activity is aborted, and there are following activities in the process that depend on the completion of this preceding activity, the following activities linked to this preceding activity will NOT start. This option is checked by default.</p>
<b>Abort the parent process on abort</b>	<p>Automatically aborts the parent process for an activity if an unexpected error occurs when the activity is executing. This option is checked by default.</p> <p>If the activity is aborted, and there are following activities in the process that depend on the completion of this preceding activity, the following activities linked to this preceding activity will NOT start.</p>

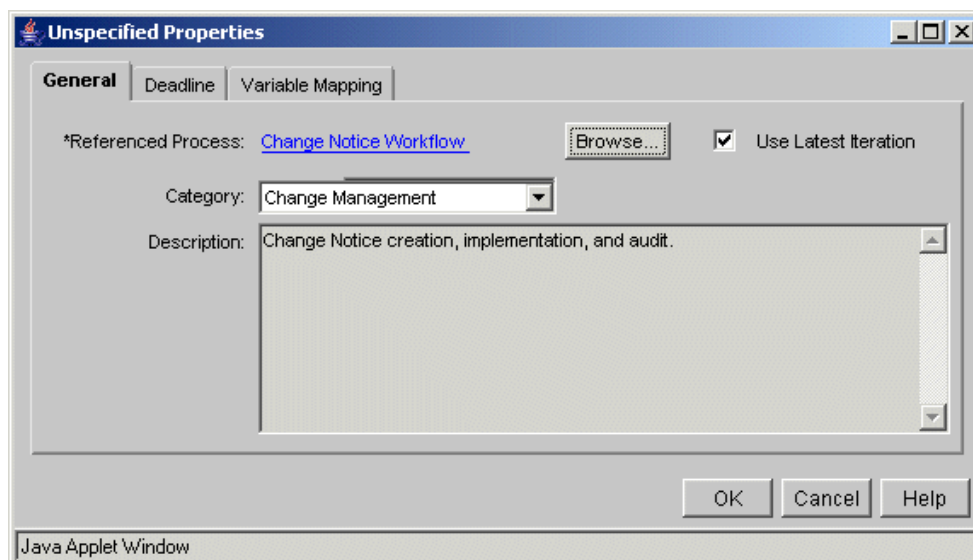
Check box	Description
<b>Record variable changes</b>	Saves the variable values entered by an individual assignee in the activity. This applies only to variables that are displayed in an activity and are editable by an assignee for the activity. These values are captured during workflow execution for each activity instance and for every time the activity is executed.
<b>Record task reassignment</b>	Keeps a record of task reassignments associated with each task. When you select this option for an activity, a record of the original assignee, the new assignee, the reason for reassignment, and the reassignment date is saved. These values are captured during workflow execution for each activity instance and for every time the activity is executed.
<b>Record voting</b>	Keeps a record of routing votes made by each participant in workflow activity with manual routing choices. These values are captured during workflow execution for each activity instance and for every time the activity is executed.
<b>Ignore unresolved role</b>	Ignores non-required role participants if no users are mapped to that role in the associated team. If no users are mapped from the team, for a required role, that role participant is automatically mapped to the responsible role (by default, the user that created the business object to which the workflow process, via the life cycle, is associated).

## Defining a Subprocess

A subprocess, or proxy process, can be included as a node of another process, the parent process.

To add a proxy process to the process definition, drag the Proxy Process icon () onto the process diagram. The properties window opens. You can designate a category, enter a description, and browse to select an existing process, which then becomes the proxy process. To ensure that the proxy process is updated when the original process is changed, select **Use Latest Iteration**. The name of the process appears as a link as the **Referenced Process**.

If you do not select a process, you cannot return to the process diagram. Instead, you must click **Cancel**; the proxy process node disappears from the process diagram.



To set a deadline for the proxy process, click the **Deadline** tab.

To map variables, click the **Variable Mapping** tab. See [Variables](#) for more information.

**Note:** Ad hoc activities and blocks are similar to proxy process, in that they are composed of a group of activities. A block is a way of simplifying the graphical representation of the process, by combining a number of activities under one icon. An ad hoc activity is a group of activities defined at runtime.

## Defining Connectors

The Workflow Process Editor supports the following connector types:

Connector Type	Description
Start	The Start connector represents the starting point in a process. Each process has only one Start connector, which cannot be removed or duplicated.
And	An And connector does not fire until all predecessor links have fired. That is, it waits for all preceding activities to complete before allowing the process to continue. For example, a Promote activity can be connected to multiple review activities by an And connector.
Or	An Or connector fires if any of its predecessor links fire. That is, it allows the process to continue if any of the preceding activities have completed. For example, a Revise activity can be linked to multiple review activities by an Or connector.
Threshold	<p>A Threshold connector fires if a user-defined number of predecessor links fire. That is, it allows the process to continue only when the user-defined number of preceding activities have completed. To set the number of activities that must complete before a Threshold connector fires, enter a number in the <b>Firing Threshold</b> text box, on the <b>Threshold Properties</b> window.</p> <p>To set a dynamic threshold, in which the firing threshold is set to 0 at runtime and is reset to the number of started predecessor activities, select <b>Add One</b> from the drop-down list in the <b>Action</b> list on the preceding <b>Link Properties</b> window.</p>
Conditional Router	A Conditional Router fires user-defined events based on an automatic event firing expression. Because you can define user events and fire them with an expression for all connectors, the Conditional Router is essentially an Or connector identified by a special icon.

You specify custom routing events, which can be used to control process flow, on the **Routing** tab panel on the Connector Properties window. See [Routing](#) for more information.

**Note:** Because the Or connector and the Threshold connector do not require all activities to be completed before allowing the process to continue, unnecessary activities can remain open. To terminate these activities, click **Terminate Open Predecessor Activities When Fired** on the **Or Properties** window, or the **Threshold Properties** window.

## Defining Links

Links define the flow of control among nodes within a process definition. They also determine which actions are performed in an activity when a predecessor activity broadcasts (or emits) events. For example, when a user completes a review task (indicating completion by clicking a button on that task page), you can specify that the completion event will cause a link to the next activity to fire.

Click the hyperlink associated with a link representation, or double-click the rule between nodes in your diagram, to access the **Link Properties** window, which allows you to map events to actions.

On the **Link Properties** window, you can map events to actions.

To map an event (in the left column), select a successor action from the drop-down list in the right column. You can specify more than one event to cause the same action to occur. To indicate that an event will be ignored, leave the field in the **Action** column blank. If an event is ignored, no action is performed when that event is emitted.

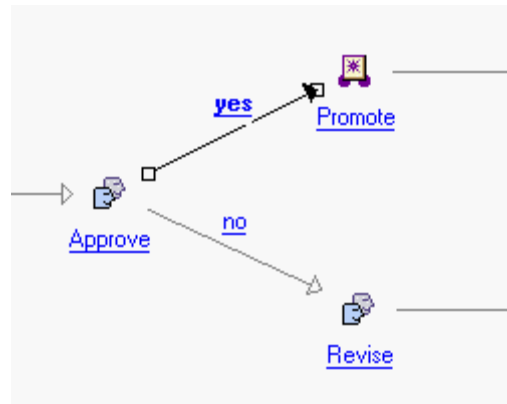
To reset all connectors in an events path when it is fired, select the **Loop Link** check box. Selecting the Loop Link check box ensures that connectors that are in the loop get reset prior to the workflow loop getting traversed again. The connectors that have already fired are reset, and they can be fired again. Loop links are designed to reset all connectors in the path of the workflow activity that is the target of the loop link. A connector is considered in the path of this activity if it is reached by going forward from the activity AND going backwards from the activity. Loop links appear in red.

You can define custom routing events for most activities and processes. When you do so, these events are also displayed in the **Link Properties** window, and you can map them to actions in a successor activity. For example, you could include in a process definition the following assigned activities:

- Approve
- Revise
- Promote

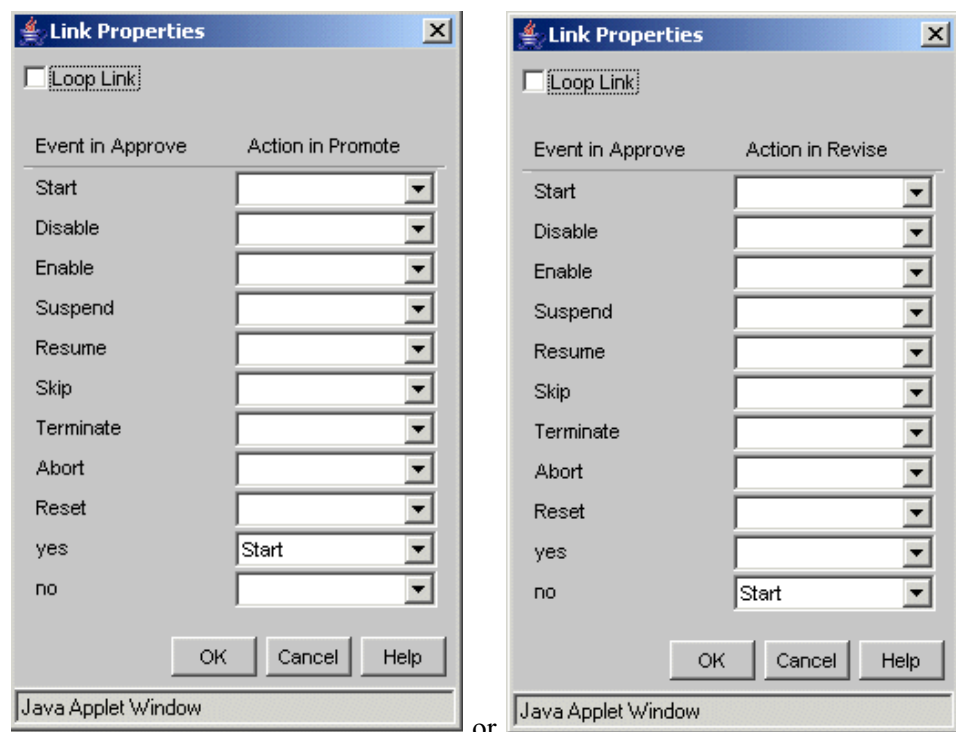


See the following figure:



The Approve activity defines two custom routing events, yes and no. This activity has two links: one link connects to the Revise activity, and another connects to a Promote robot. The link to the Revise activity can be configured to perform the Start action in Revise when the no event is emitted from the Approve activity. The other link can be configured to perform the Start action in the Promote robot when the yes event is emitted from Approve. In this way, the flow of control to the Revise or Promote activity is controlled by the event emitted from the Approve activity.

See the figure that follows for possible results:



# Import and Export

## Preparing to Import or Export Workflow Templates

Before you begin, you should be familiar with the following information regarding importing and exporting:

- Be sure to upgrade to the latest maintenance only release (MOR), as it becomes available, to ensure you have the latest enhancements to the import and export functionality.
- You can import templates into a later version of Windchill; importing to an earlier version may not work. The data you are importing must comply with the database integrity and structure of the older version. To import templates from versions prior to Windchill 7.0, you must convert the CSV to XML format. Use the CSV2XML utility to convert CSV files to XML files. The syntax is as follows:

```
windchill wt.load.util.CSV2XML -input <input file or directory>  
-output <output directory> -root <root directory> -help
```

- Importing or exporting workflow templates creates objects in a JAR or ZIP file format. (This is the same format that the load.Installer functionality uses.) Use a zip utility like WinZip to create a ZIP file consisting of XML files. To create a JAR file, use the following syntax:

```
jar -cf <jar file name> <xml files to be included>
```

or

```
jar -cvf (for verbose output to print each file name as it is  
added)
```

- You can import any number of templates at a time. You can have many XML files, each representing one workflow template, in one JAR or ZIP file, or you can have one XML file representing many workflow templates. The order of the templates is important. For example, if template A has a subprocess template B, then B must be placed before A in the XML.

There is no limit to the number of workflow templates you can export. To export multiple templates into a single JAR or ZIP file, hold down the CTRL key, select the templates you want to export, and click **Export**. All the selected templates are exported to the same JAR or ZIP file.

- Iterations of a workflow template are not imported or exported. When you export a workflow that contains a fixed reference to a specific iteration that is not the latest iteration, the import fails. PTC recommends that you change all fixed references to floating references (that is, references to the latest iteration) before you export the workflow template.
- PTC recommends that you not export data from one version of Windchill and import it into a different version. Instead, migrate the data in your database and then export the templates.

- Errors may occur, especially when importing workflow templates. Some errors result in messages displayed; others may be fatal. Check the method server log for error information.
- When you export a workflow template, only the template itself is exported. This includes references to underlying objects, such as principals, roles, and actor roles. The underlying objects themselves are not exported. If the export file is used to import the template into another database, the underlying objects must exist in the database, or the import fails and errors appear in the method server log. This can occur especially when importing the object into a different system. Be certain that all underlying objects referenced in the XML representation of the template exist.
- If a workflow template is imported and a template with the same name already exists in the Import directory, the results depend on the Iteration On Import setting in the wt.properties file. If wt.workflow.iterateOnImport=true, the imported workflow template is appended to the existing template as a new iteration. If it is set to FALSE, the imported file causes a method server exception, stating that there is a duplicate name, and the template is not imported.
- If you want to reuse an existing template that resides in another context, export the template from source context to your local client file system, then import to your target context.

## Importing and Exporting Workflow Templates

Use the **Import** and **Export** buttons on the **Workflow Administrator** page to import or export workflow templates.

To import one or more templates from a JAR or ZIP file, select the file from the local file system in the **Import** window, and click **Import**.

To export a workflow template into an XML representation of the template, use the following procedure:

1. Select a template from the **Workflow Administrator** page, and click **Export**. (The **Export** button is disabled if you do not have a template selected.)
2. A grant permission window may appear asking for permission to access the local file system and to write a file on that system. If you select the remember selection check box, permission need only be granted once. Once permission is granted, a **Browse for file** picker opens, defaulting to the system **temp** folder.
3. You can pick a file that exists or type in a new name. If the file name exists, you are asked to confirm to overwrite the file. You must click **Yes** to continue the export. If the file name does not exist, a new file is created.

There is no confirmation that the export is completed. When the progress bar

and the hourglass on the workflow administrator applet disappear, the export is complete.

## Process Manager Toolbar Access Control

Process Manager toolbar buttons give authorized users the ability to change workflow processes. An authorized user can make vital changes to an activity or a process, such as terminating or completing it. Access to Process Manager buttons is defined in the following file:

```
<install directory>/codebase/wt/clients/workflow/manager/process-manager.properties
```

As shipped, full access is given only to the system administrator and the workflow administrator, as defined in the following properties:

```
activityrestartAccessControl=Administrators
activitysuspendAccessControl=Administrators
activityresumeAccessControl=Administrators
activityterminateAccessControl=Administrators
activitycompleteAccessControl=Administrators
processsuspendAccessControl=Administrators
processresumeAccessControl=Administrators
processterminateAccessControl=Administrators
processcompleteAccessControl=Administrators
activityrestartAccessControl=WorkflowAdministrators
activitysuspendAccessControl=WorkflowAdministrators
activityresumeAccessControl=WorkflowAdministrators
activityterminateAccessControl=WorkflowAdministrators
activitycompleteAccessControl=WorkflowAdministrators
processsuspendAccessControl=WorkflowAdministrators
processresumeAccessControl=WorkflowAdministrators
processterminateAccessControl=WorkflowAdministrators
processcompleteAccessControl=WorkflowAdministrators
```

To add other groups, add additional AccessControl lines to the file, in the format:

```
<action>AccessControl=<group name>
```

When you have updated the properties file the, you must recreate the JAR file (wtWorkFCS.jar).

## Viewing Workflow History

To optimize and streamline your workflow process definitions, you should capture and record events, such as state transitions and variable updates that occur during process execution. The ordered sequence of these events is called a workflow history, and allows you to capture significant events.

In addition, you can specify that a keyed event be emitted, which allows synchronization of other Windchill managers and servers with process events. As described in this section, you can specify the following:

- Which events are to be ignored
- Which are to be recorded

- Which are to be emitted as keyed events
- Which are to be both recorded and emitted

**Note:** When there is a loop link in a workflow with a conditional route, only the last vote is recorded.

## Selecting Events

The following events can be generated during execution of a workflow process:

- Process creation, which occurs when a Start command is issued for an existing, enabled process definition.
- Change of state, which occurs when an execution object (a process or an activity) changes states (for example, from running to completed).
- Change of data value, which occurs when values are read into a process activity or written to process variables.
- Change of assignee, which occurs when assignment of a workflow task changes from one user to another.
- Error event, which occurs when an exception is thrown during process execution.

You can identify the events to be recorded in the workflow history or emitted as keyed events by setting the default event configuration in the `wt.properties` file. Please note that workflows created before changes to the `wt.properties` entries are applied are not affected. When one or more of the following properties is set to true, the corresponding event is recorded or emitted:

```
wt.workflow.engine.recordProcessStateChange
wt.workflow.engine.recordProcessDataChange
wt.workflow.engine.recordActivityStateChange
wt.workflow.engine.recordActivityDataChange
wt.workflow.engine.recordProcessCreation
wt.workflow.engine.recordAssigneeChange
wt.workflow.engine.recordException
wt.workflow.engine.recordVoting
wt.workflow.engine.emitProcessStateChange
wt.workflow.engine.emitProcessDataChange
wt.workflow.engine.emitActivityStateChange
wt.workflow.engine.emitActivityDataChange
```

```
wt.workflow.engine.emitProcessCreation
```

```
wt.workflow.engine.emitAssigneeChange
```

```
wt.workflow.engine.emitException
```

All running workflow processes record and emit events based on this default configuration. As described in the next section, however, you can use the Workflow History Viewer to change the events recorded or emitted by a given process.

## Using the Workflow History Viewer

You can view workflow-generated events on the Workflow History Viewer, a simple ASCII utility. Issue the following command to start the Workflow History Viewer:

```
windchill wt.workflow.engine.WfMonitor
```

The following is the main menu of the Workflow History Viewer:

```
Workflow Monitor - Main Menu
```

```
Existing Workflow Processes
```

```
-- -- -- -- --
```

```
>> 1. Simplest, key = 3347 (Running)
```

```
Audit events
```

```
-- -- -- -- --
```

```
No event retrieved
```

1. Select process.
2. Show selected process.
3. Delete selected process.
4. Refresh processes.
5. Change event configuration for selected process.
6. Show selected events for selected process.
7. Show all events for selected process.
8. Show selected events for all processes.
9. Show all events for all processes.
10. Select event.
11. Show event.
12. Show event source.
13. Refresh events.
14. Delete events for selected process.
15. Delete events for all processes.
16. Exit

```
>>> Choose an option:
```

This menu includes a list of all existing processes and a list of all retrieved events. The list of retrieved events is initially empty.

In addition to these lists, the main menu provides 16 options. A description of each follows:

#### Select Process

Select this option to select a process, so associated events can be queried. The selected process is preceded by the characters >>.

#### Show Selected Process

Select this option to display the following information about the selected process, as well as all of its activities and subprocesses that have been started:

- The process status (for example, running).
- The process creator.
- The team with which the process is associated.
- An indication of whether or not the process or activity is overdue (that is, whether it has exceeded its specified duration).
- The time at which the process or activity started.
- The time at which the process or activity ended (not shown if the process or activity has not been closed).
- The deadline for the process or activity, if one exists.
- The event configuration for the process.
- Suspend and alert times, in milliseconds.

The following shows a sample process display:

```
Simplest - Running
Creator: Administrator
Team: Default
Times: is overdue = false
      start time = 1998-11-20 14:08:53.0
      suspend time = 0, alert time = 0
Event configuration:
Process (RECORD/EMIT): Creation: R, State change: RE
Activity (RECORD/EMIT): Data change: E
Exception (RECORD/EMIT): Exception: R
Activities:
*** act_simplest (wt.workflow.engine.WfTestActivity) -
Running
Times: is overdue = false
      start time = 1998-11-20 14:08:56.0
      suspend time = 0, alert time = 0
```

#### Delete Selected Process

Select this option to delete the selected process.

## Refresh Processes

Select this option to refresh all processes. The Workflow History Viewer does not automatically update the status of a displayed process. You must explicitly refresh it in order to see new processes and changes of state since the last refresh.

## Change Event Configuration

Select this option to open an event configuration editor. With the editor you can select the events that should be emitted, recorded, or ignored, for the selected process.

### Edit event configuration

#### Current configuration

Process (RECORD/EMIT): Creation: R, State change: RE

Activity (RECORD/EMIT): Data change: E

Exception (RECORD/EMIT): Exception: R

1. Create process - record: true
2. Create process - emit: false
3. Change process state - record: true
4. Change process state - emit: true
5. Change process data - record: false
6. Change process data - emit: false
7. Change activity state - record: false
8. Change activity state - emit: false
9. Change activity data - record: false
10. Change activity data - emit: true
11. Change assignment - record: false
12. Change assignment - emit: false
13. Execution error - record: true
14. Execution error - emit: false
15. Save configuration
16. Save configuration and return
17. Return (looses changes since last save)

>>> Choose an option:



As previously described, all processes are governed by the event configuration specified in the wt.properties file, unless this submenu is used to change that configuration for the selected process. Selecting an option on this submenu toggles the current setting. For example, if you select option 1, you will change the setting of **Create process** from true to false. Consequently, process creation would no longer be a recorded event.

#### Show Selected Events for Selected Process

Select this option to open the submenu shown in the following figure. You can select an option from this submenu to specify the type of event in which you are interested. (In the example, option 8 is selected, indicating that the user is interested in all events associated with the selected process.)

```
Types of events to show
1. Process creation
2. Process state change
3. Process data change
4. Activity state change
5. Activity data change
6. Assignment change
7. Execution error
8. All
9. None (return)
```

```
>>> Choose an option: 8
```

When you have select an option from this submenu, the requested events are retrieved, and the list of events is refreshed. For each event, the list identifies the event type and the execution object associated with it:

```
Audit events
```

```
>> 1. PROCESS_STATE_CHANGED, process = Simplest
```

#### Show All Events for Selected Process

Select this option to represent a shortcut to selecting **All**, (option 8) from the submenu of event types described in the preceding section. If this main menu option is selected, all events associated with the selected process are shown.

#### Show Selected Events for All Processes

Select this option to display all events of the selected type for all processes.

#### Show All Events for All Processes

Select this option to display all stored events for all processes. This list can be very large.

#### Select Event

Select this option to select another event. When events are retrieved, the first event in the list is automatically selected.

### Show Event

Select this option to display detailed event information for a selected event, as shown in the following figure:

```
event type = PROCESS_STATE_CHANGED
timestamp = 1998-11-20 14:08:53.0
activity key = 0 activity name = null
process key = 3347
process name = Simplest
process template name = simplest
old state = Not started
new state = Running
```

### Show Event Source

Select this option to display the execution object (process or activity) that is the source of the selected event.

### Refresh Events

Select this option to refresh all events corresponding to the selected process and event type. This is necessary to view events generated since the last refresh.

### Delete Events for Selected Process

Select this option to delete all the events of the last selected type associated with the selected process.

### Delete Events for All Processes

Select this option to delete all events of the last selected type associated with all processes.

### Exit

Select this option to terminate the Workflow History Viewer session.

## Configuring Worklist Fields

In Windchill Foundation & PDM, users can create and save layouts for their worklist on the **Manage Layouts** page (this option is not available for Windchill PDMLink and Windchill ProjectLink). Users can determine the attributes displayed by selecting attributes from drop-down lists under each column. The drop-down lists for the columns are identical. The attributes displayed in these lists are determined in the wt.properties file.

**Note:** This option is not available for Windchill PDMLink or Windchill ProjectLink.

The following is an example of the section of wt.properties that determines what appears in these lists:

# -- Workflow Worklist fields

# -- Note: The SelectWorkItemCheckbox is essential for the Reassign, Accept

& Update DueDate functions

```
wt.workflow.worklist.column.1=wt.workflow.worklist.SelectWorkItemCheckbox
wt.workflow.worklist.column.2=wt.workflow.worklist.Task
wt.workflow.worklist.column.3=wt.workflow.worklist.Priority
wt.workflow.worklist.column.4=wt.workflow.worklist.Role
wt.workflow.worklist.column.5=wt.workflow.worklist.PrimaryBusinessObject
wt.workflow.worklist.column.6=wt.workflow.worklist.ActivityDeadline
wt.workflow.worklist.column.7=wt.workflow.worklist.WorkflowProcessName
wt.workflow.worklist.column.8=wt.workflow.worklist.Team
wt.workflow.worklist.column.9=wt.workflow.worklist.Owner
wt.workflow.worklist.column.10=wt.workflow.worklist.AssignmentState
wt.workflow.worklist.column.11=wt.workflow.worklist.ActivityStart
wt.workflow.worklist.column.12=wt.workflow.worklist.WorkflowProcessDeadline
wt.workflow.worklist.column.13=wt.workflow.worklist.ActivityDescription
wt.workflow.worklist.column.14=wt.workflow.worklist.WorkflowProcessStart
wt.workflow.worklist.column.15=wt.workflow.worklist.WorkflowProcessDescription
wt.workflow.worklist.column.16=wt.workflow.worklist.Required
```

The attributes are defined by the key: wt.workflow.worklist.column.#. The entry to the right of the key is a class displaying a work item attribute. The order of columns displayed in the worklist table is determined by the numbers in the fifth level (1–n).

In the example above, the drop-down list would display the first ten attributes (WorkItemCheckbox through AssignmentState) in the order they appear.

To change the order in which the attributes are displayed, edit the numbers in the fifth level. For example, for the following line, if you change the 8, it will change the order.

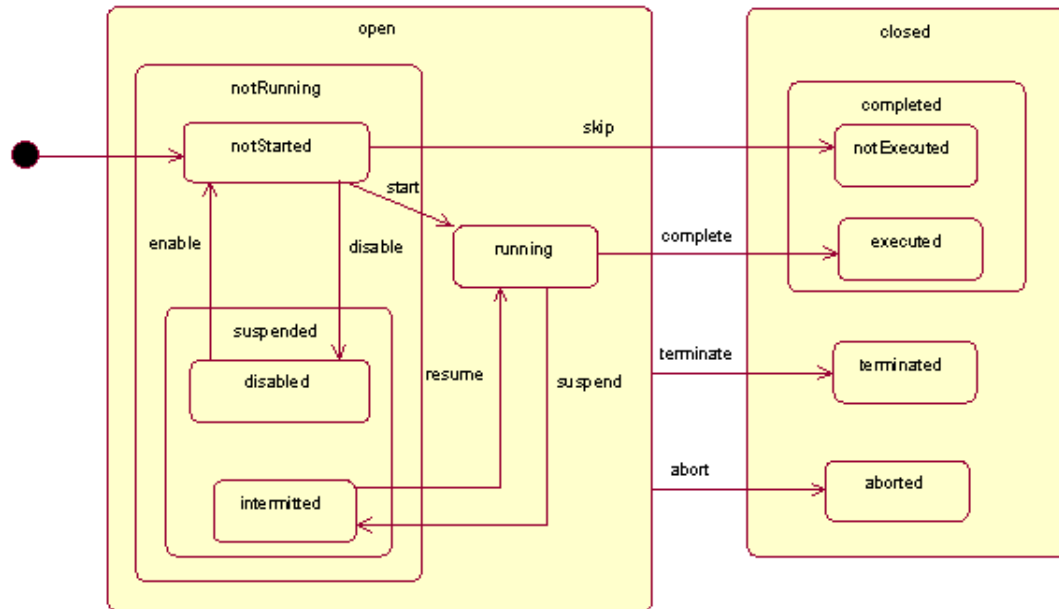
```
wt.workflow.worklist.column.8=wt.workflow.worklist.Team
```

To remove an attribute from the list, insert # at the beginning of the line to comment out the line.

**Note:** The columns must be numbered consecutively, beginning with 1. Therefore, if you comment out a line or edit the order, you may have to renumber the remaining entries. In the example above, if you commented out line 4 (Role), you would have to renumber lines 5 (PrimaryBusinessObject) through 10 (AssignmentState), to be 4 through 9.

## Workflow Instance States

The following diagram shows the states that an execution object (activity or process) can be in, and the transitions from state to state.



Rounded boxes represent states; arrows represent transitions. The actual states are always the innermost ones. The others are super-states, indicating a collection of substates. For example, a query for closed processes returns the processes that have completed successfully and also the processes that have terminated or were aborted.

The initial state for all execution objects is the not started state. The following is the normal sequence of states for an execution object is:

1. Not Started
2. Running
3. Executed

The final state can be reached with the following two transitions:

- Start
- Complete

Some transitions apply to more than one state. This is indicated by an arrow that begins in a super-state. For example, terminate transitions from any open state to the terminated state. An additional transition, reset, is not shown in the diagram. The reset transition brings any object back to the not started state.

Both Open.NotRunning.Suspended.Disabled and Open.NotRunning.Suspended.Intermitted are labeled suspended in the GUI; however, the Open.NotRunning.Suspended.Disabled state is currently not used, so there is no chance of confusion.

The following is the model for connectors:

- Enabled
- Disabled

## Domain-based Workflows

During initialization, loading administration data into the Windchill database creates an access control policy rule that grants all users read, modify, and create permissions for objects of type wt.workflow.engine.WfExecution Object in the /System domain. The WfProcess class extends the WfExecutionObject class, so this rule applies to workflow process instances. All Windchill users can view process information by using Local Search. This is the default out-of-the-box behavior.

In order to provide better control over access to workflow process instances, you can set a property to enable domain-based workflows. The workflow processes are created and run within the same administrative domain as the primary business objects associated with the processes. You can specify access control policies for the domains associated with the workflows, granting different rights to WfProcess objects for each of the domains.

The property in the wt.properties file that toggles domain-based functionality is wt.workflow.engine.domainBasedWorkflow. The default setting is FALSE. If this property is set to true, the domain-based functionality takes effect following method server restart. WfProcesses do not immediately change to their new domain; however, upon the next workflow state change (for example, from not started to running), the WfProcess changes domains, as appropriate.

The user can use Local Search to find workflow processes in domains for which they have read access to WfProcess objects. If access control policy rules do not grant a user access to the primary business object, the user can be granted ad hoc access to the object. These rights are explicit to the primary business objects and are not applied to the parent folder.

**Note:** Windchill Explorer is not an option for Windchill PDMLink or Windchill ProjectLink.

There is a property in the wt.properties file called wt.workflow.definer.checkAdminAccess. The default is false. That means you can access the workflow clients through the URL alias. If you want to restrict access to the Workflow Administrator, perform the following steps:

1. Set wt.workflow.definer.checkAdminAccess to true.
2. Update the httpd.conf file to require authentication when accessing Web pages under /windchill/wt/clients/workflow (see below).
3. Restart the method server.

Any user attempting to access the **Workflow Administrator** page is then required to authenticate as a member of the Workflow Administrators group. If the authentication fails, an error page indicates that they are not authorized to access that page.

An example of setting the web server (Apache) to authenticate access to workflow clients is as follows. This change is made to the apache/conf/httpd.conf file.

```
# Admin level authentication
<Location /windchill/wt/clients/workflow/>
    AuthName Windchill
    AuthType Basic
    <IfModule auth_ldap.c>
        AuthLDAPAuthoritative off
        AuthLDAPURL
            ldap://ldap.company.com:389/ou=People,ou=wc62,l=city,o=company
    </IfModule>
    AuthUserFile "C:/Program Files/Apache Group/Apache/conf/wtpasswd"
    require valid-user
</Location>
```

Italicized items are installation-dependent.

## Default Workflow Templates

The out-of-the-box workflow templates in Windchill allow you to automate procedures in which tasks and documents are passed among participants. For general information about workflows, refer to the Workflow Administrator online help.

PTC Global Services consultants can help you determine how the out-of-the-box workflows correspond to your business processes. Although you can modify workflows if necessary, you will get better results if you map the existing workflows to your processes, rather than create new workflows based on your processes.

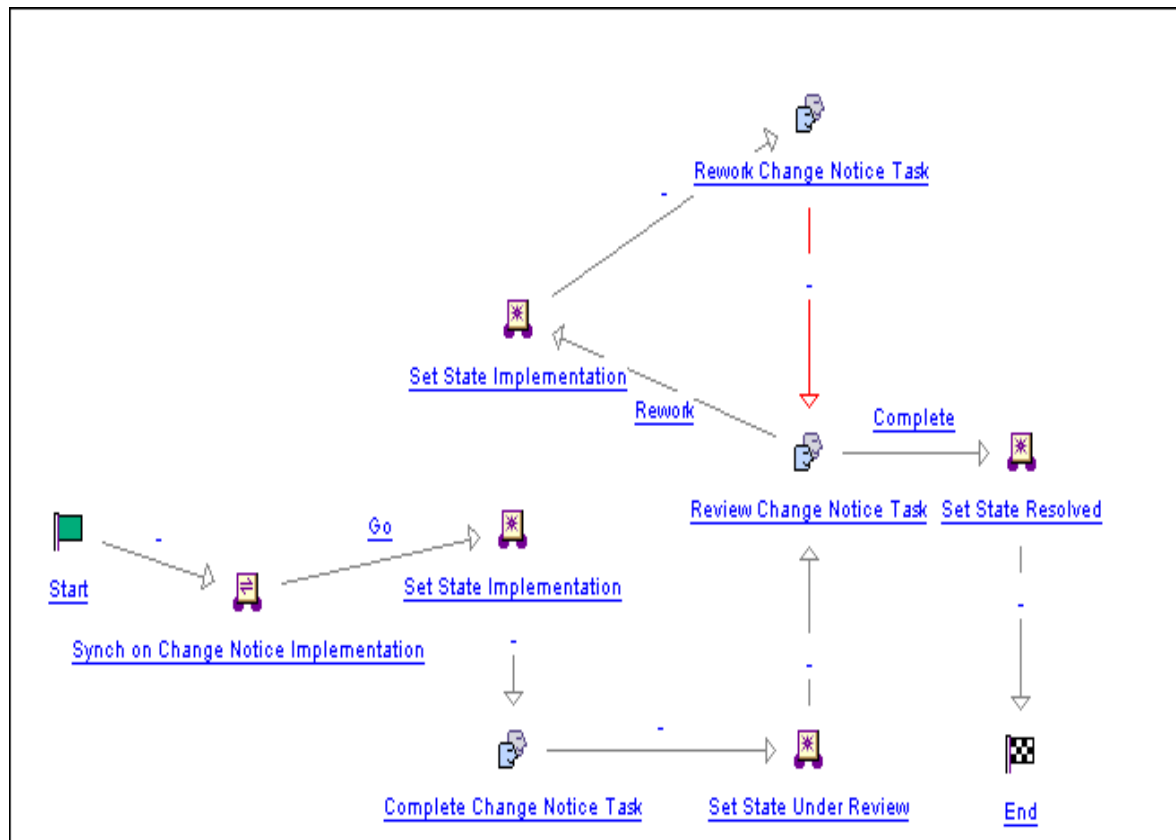


**Caution:** Changes you make to workflows may not be compatible with future Windchill release and service packs, and may require additional support from PTC Global Services consultants.

## Windchill PDMLink Default Workflows

This section summarizes the steps in each Windchill PDMLink out-of-the-box workflow template.

### Change Activity Workflow



1. Complete change notice task

Review the assigned change notice task, specify updates to the affected data, and submit the completed task.

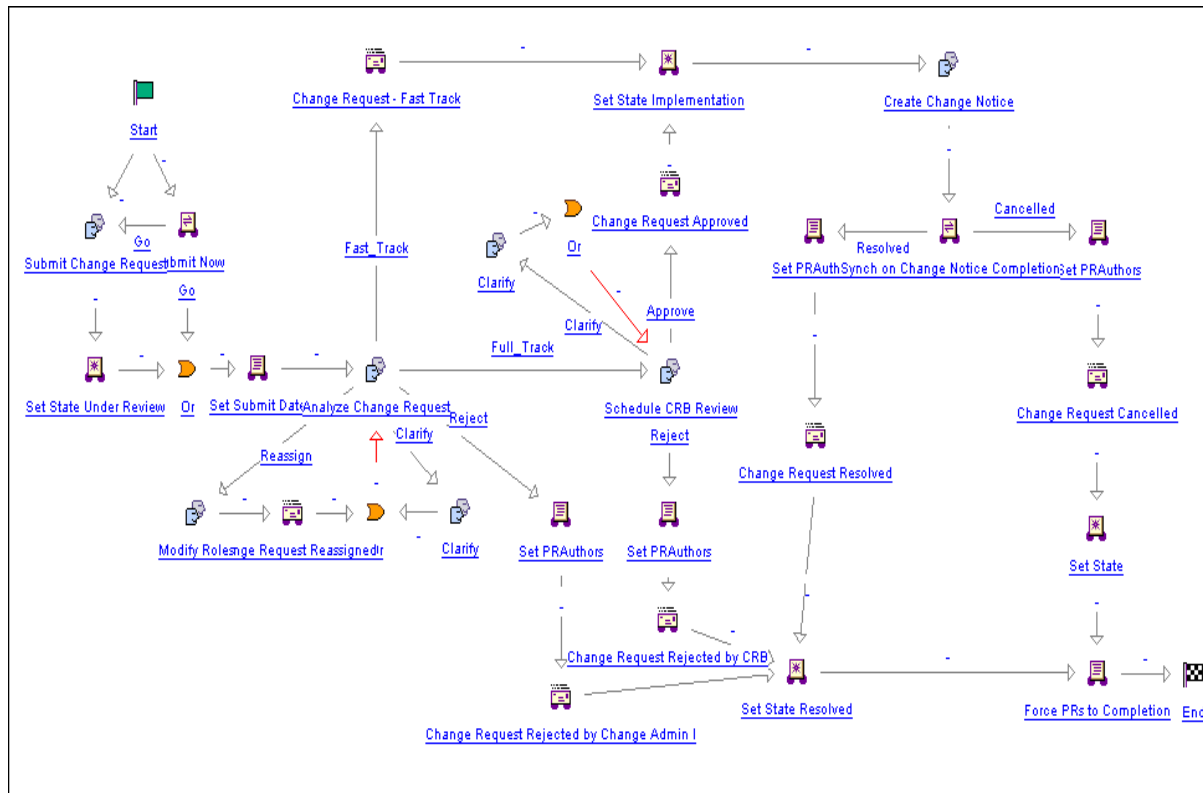
2. Review change notice task

Review the data modifications resulting from the preceding step and approve or reject the modifications.

3. Rework change notice task

According to the reviewer's instructions from the preceding step, perform the required modifications and resubmit them to Change Administrator III for audit.

## Change Request Workflow



### 1. Submit change request

Submit the change request for review by Change Administrator I.

### 2. Analyze change request

Validate the change request, coordinate the technical analysis and generation of a recommended solution, associate relevant analysis information, and determine the disposition of the ECR.

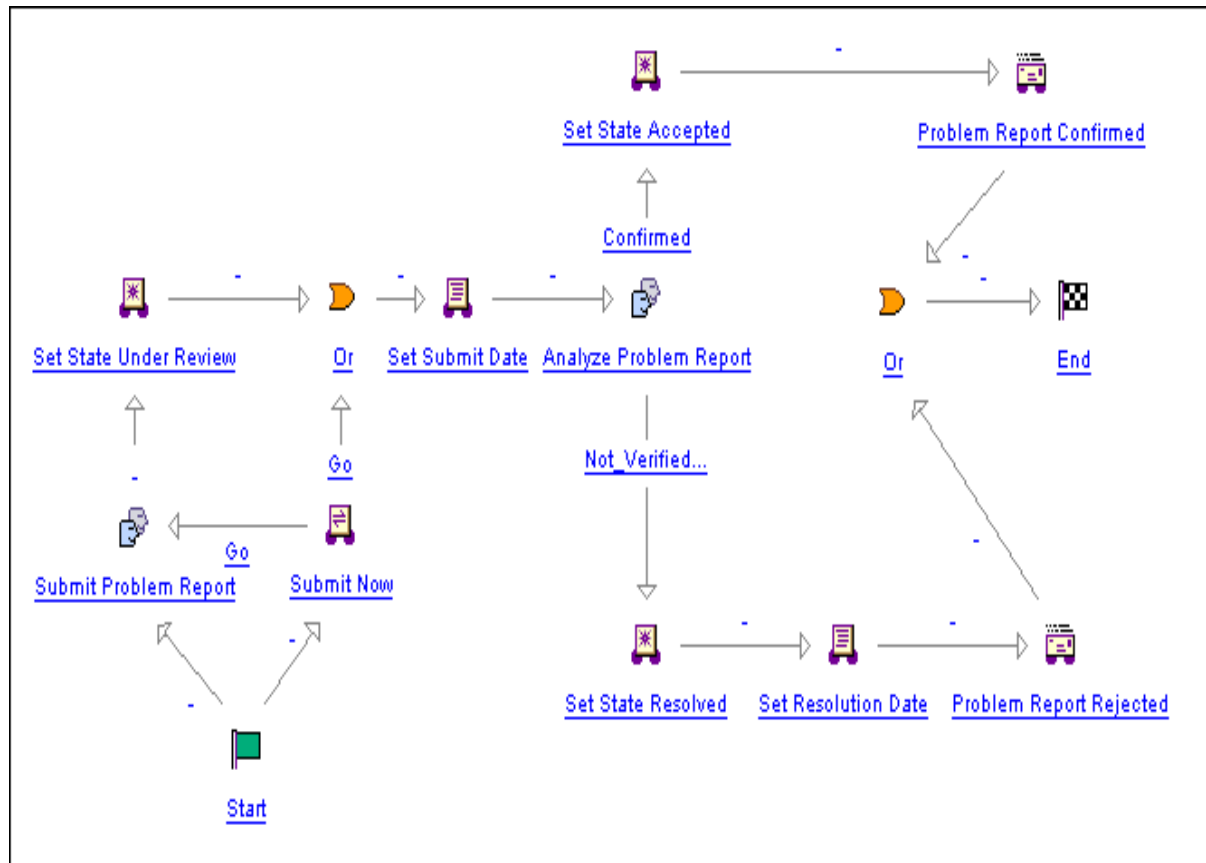
### 3. Clarify change request

The change request is sent back to the author for the addition of further information.



- a. Schedule and facilitate (Change Review Board) CRB review and record CRB decision.
- b. Create change notice

## Problem Report Workflow



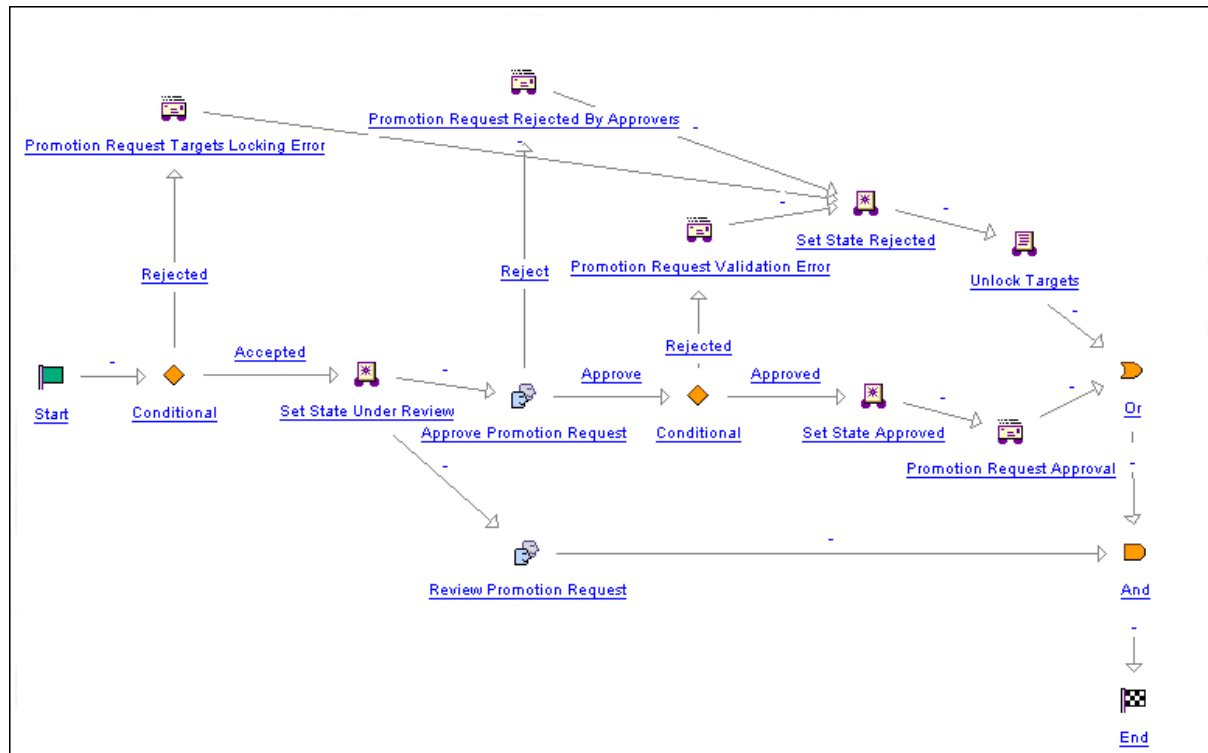
### 1. Submit problem report

Submit the problem report for review by Change Administrator I.

### 2. Analyze problem report

Analyze the problem report, associate the relevant analysis information, and determine the disposition of the problem report.

## Promotion Request Approval Process Workflow



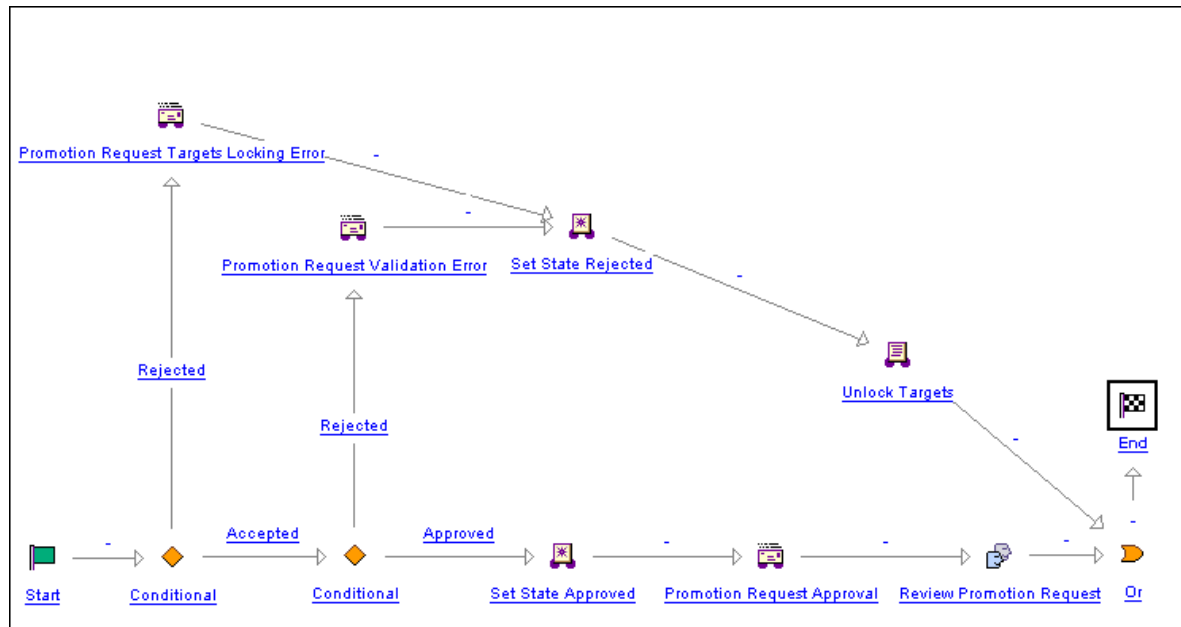
### 1. Review Promotion Request

Invite Reviewers from the Promotion Reviewers group, as well as participants added by the requesting author, to provide comment on these items

### 2. Approve Promotion Request

Invite Approvers from the Promotion Approvers group to provide comments and approve or reject this request. All Approvers must approve the request in order for the items to be promoted to the target state.

## Promotion Request Review Process Workflow



### 1. Review Promotion Request

Route a promotion request to a group of people to review and comment on the items. All items on the promotion request are automatically promoted to the requested life cycle state.

## Workflow Template Execution Flags

The workflow engine uses several flags to control behavior during the execution of the workflow. These flags default to wt.property values. Execution flags fall into three categories:

- Process - These flags affect the entire workflow process.
- Activity - These flags affect individual workflow activities.
- Both process and activity - These flags are meaningful for both workflow processes and workflow activities.

## Process Flags

These flags are meaningful only to workflow processes.

Flag Name	Associated wt.property	Short Description
Create process - record	wt.workflow.engine.recordProcessCreation	Determines if the workflow process creation events are recorded as part of the workflow history. You can change the value of this property for a specific process by altering its execution characteristics.
Create process - emit	wt.workflow.engine.emitProcessCreation	Determines if workflow process creation events should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific process by altering its execution characteristics.
Change process state - record	wt.workflow.engine.recordProcessStateChange	Determines if the workflow change of state events should be recorded as part of the workflow history. You can change the value of this property for a specific execution object by altering its execution characteristics.
Change process state - emit	wt.workflow.engine.emitProcessStateChange	Determines if the workflow state change events that occur with processes should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific process by altering its execution characteristics.
Change process data - record	wt.workflow.engine.recordProcessDataChange	Determines if the workflow change of data events should be recorded as part of the workflow history. You can change the value of this property for a specific execution object by altering its execution characteristics.

<b>Flag Name</b>	<b>Associated wt.property</b>	<b>Short Description</b>
Change process data - emit	wt.workflow.engine.emitProcessDataChange	Determines if the workflow state change events that occur with processes should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific process by altering its execution characteristics.
Execution error - record	workflow.engine.recordException	Determines if the workflow execution exceptions should be recorded as part of the workflow history. You can change the value of this property for a specific execution object by altering its execution characteristics.
Execution error - emit	workflow.engine.emitException	Determines if workflow execution exception should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific execution object by altering its execution characteristics.
Notify on exception	wt.workflow.engine.notifyOnException	This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific process by altering its execution characteristics.
Abort process on exception	wt.workflow.engine.abortProcessOnException	Determines whether to abort the workflow when an exception occurs. If true, the process instance is aborted if a contained activity aborts.
Notify on overdue	wt.workflow.engine.notifyOnOverdue	Determines whether a notification should be sent to the principals in the overdue notification list when the process or activity is overdue. You can change the value of this property for a specific process by altering its execution characteristics.

<b>Flag Name</b>	<b>Associated wt.property</b>	<b>Short Description</b>
Notify on complete	wt.workflow.engine.notifyOnComplete	Determines whether a notification should be sent to the process responsible when the execution object completes. You can change the value of this property for a specific process by altering its execution characteristics.
Notify on past deadline	wt.workflow.engine.notifyOnPastDeadline	Determines whether a notification should be sent to the process responsible after some time has elapsed since the execution object completed. The time is defined in an attribute if the object is past the deadline.
Notify on approaching deadline	wt.workflow.engine.notifyOnApproachingDeadline	Determines whether a notification should be sent to the process responsible when the time to deadline has been reached. You can change the value of this property for a specific process by altering its execution characteristics.
Asynchronous execution	wt.workflow.engine.asynchronousExecution	Determines if the execution of an activity is asynchronous, that is, not terminated when the host process or block completes. If an asynchronous activity occurs in a loop, one may have several instances of the same activity template executing concurrently.

Flag Name	Associated wt.property	Short Description
Has dedicated queue	wt.workflow.engine.dedicatedQueueMode	<p>Determines whether the workflow processes have dedicated queues for event propagation and user work (robot expressions, synchronization, and so forth). A value of true causes the processes to have dedicated queues. Users may want a dedicated queue for a process when they want to prevent its execution from being impacted by the execution of other workflows.</p> <p>The default value is false, which means that these queues are normally shared (as are the other workflow queues).</p> <p>This attribute is used when the dedicatedQueueMode is one of propagation, userWork, or both.</p> <p>You can control this property in an individual process base by setting the appropriate execution flag in the process template.</p>
Select Latest Template Iteration	wt.workflow.engine.useLatestTemplateIteration	<p>Determines whether the latest iteration of a specific version of an object is used the first time the subprocess is encountered in the workflow. If the subprocess template is iterated and then the process loops back, the initial version (not the iterated version) continues to be used since the subprocess referenced that version.</p> <p>The default value is true, which means that the latest iteration is used the first time the subprocess is encountered.</p>



## Activity Flags

These flags are meaningful only to workflow activities.

Flag Name	Associated wt.property	Short Description
Change activity state - record	wt.workflow.engine.recordActivityStateChange	Determines if the workflow state change events that occur with activities are recorded as part of the workflow history. You can change the value of this property for a specific process by altering its execution characteristics.
Change activity state - emit	wt.workflow.engine.emitActivityStateChange	Determines if workflow state change events that occur with activities should be emitted as a keyed event. This event can then be listened to and used for synchronization and other purposes. You can change the value of this property for a specific activity by altering its execution characteristics.
Change activity data - record	wt.workflow.engine.recordActivityDataChange	Determines if the workflow change of data events that occur with activities are recorded as part of the workflow history. You can change the value of this property for a specific process by altering its execution characteristics.
Change activity data - emit	wt.workflow.engine.emitActivityDataChange	Determines if workflow change of data events that occur with activities should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific activity by altering its execution characteristics.
Change assignment - record	wt.workflow.engine.recordAssigneeChange	Determines if workflow change of assignment events should be recorded as a as part of the workflow history. The value of this property for a specific activity can be changed by altering its execution characteristics.

<b>Flag Name</b>	<b>Associated wt.property</b>	<b>Short Description</b>
Change assignment - emit	wt.workflow.engine.emitAssigneeChange	Determines if workflow change of assignment events should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific activity by altering its execution characteristics.
Ignore unresolved Role	wt.workflow.engine.ignoreUnresolvedRoles	Determines whether unresolved roles should be ignored rather than automatically assigned to the responsible role. If set to true and all roles in an activity are unresolved, no work item is generated without an indication that something went wrong. This is the global installation default behavior.

## Both Process and Activity Flags

These flags are meaningful to both workflow processes and activities.

Flag Name	Associated wt.property	Short Description
Execution error - record	wt.workflow.engine.recordException	Determines if the workflow execution exceptions should be recorded as part of the workflow history. You can change the value of this property for a specific execution object by altering its execution characteristics.
Execution error - emit	wt.workflow.engine.emitException	Determines if workflow execution exception should be emitted as a keyed event. This event can be listened to and used for synchronization and other purposes. You can change the value of this property for a specific execution object by altering its execution characteristics.
Notify on abort	wt.workflow.engine.notifyOnAbort	Determines whether a notification should be sent to the process or activity responsible when the process or activity aborts. You can change the value of this property for a specific process by altering its execution characteristics.
Abort on exception	wt.workflow.engine.abortOnException	Determines whether to abort the activity or process if an exception occurs during its execution.
Reassign on overdue	wt.workflow.engine.reassignOnOverdue	Automatically reassigns the process or activity when the process or activity is overdue. The new assignee is read from a variable named reassignPrincipal. You can change the value of this property for a specific activity by altering its execution characteristics.
Complete on overdue	wt.workflow.engine.completeOnOverdue	Determines whether an activity or process is automatically completed once it is overdue.

Flag Name	Associated wt.property	Short Description
Skip on overdue	wt.workflow.engine.skipOnOverdue	Automatically skips the process or activity when the process or activity is overdue. Note that an execution object can be skipped only if it has not been started. Also this transition needs to be as the event of an outgoing link for it to be propagated. You can change the value of this property for a specific activity or process by altering its execution characteristics.

## Modifying Execution Flags

To modify the execution flags, use the SetConfiguration tool provided.

## Running SetConfiguration

To run SetConfiguration, issue the following command:

```
java wt.clients.workflow.definer.SetConfiguration
Y:\Windchill>java wt.clients.workflow.definer.SetConfiguration
Introspection Runtime Mode = true
Windchill DataStore: Oracle

Workflow process templates
>> 1. Submit - (10003aaa)
      2. Review - (10003aaa)
      3. Problem Report Workflow - (10003aaa)
      4. Change Request Workflow - (10003aaa)
      5. Change Notice Workflow - (10003aaa)
      6. Change Activity Workflow - (10003aaa)
      7. Promotion Request Approval Process - (10003aaa)
      8. Promotion Request Review Process - (10003aaa)
      9. Approval Process - (1001faaa)
     10. Notify Process - (1005faaa)
     11. Release Process - (1005faaa)
     12. Review Process - (1005faaa)
```

13. Two Level Approval Process - (1001faaa)

14. Change Notice Workflow - (10003aaa)

Activity templates of selected process:

>> 1. Submit - (10003aaa)

Set template configuration

-----

1. Select process template
2. Show/edit process template configuration
3. Select activity template
4. Show/edit activity template configuration
5. Refresh templates
6. Exit

>>> Choose an option:

This tool shows a list of workflow templates, with the active template selected with the >> symbol. The next list is a list of the activities of the selected process templates. The active activity is shown with the >> symbol. In our example above, the Submit process template is active and the Submit activity is the active activity within the Submit process. To change the active process template, choose option 1. To change the active activity, select option 3.

To modify an exception flag, choose option 2. To modify an activity flag choose option 4. A list of flags is shown. For example if you choose 4:

```
Y:\Windchill>java wt.clients.workflow.definer.SetConfiguration
```

```
Introspection Runtime Mode = true
```

```
Windchill DataStore: Oracle
```

Workflow process templates

- >> 1. Submit - (10003aaa)
2. Review - (10003aaa)
  3. Problem Report Workflow - (10003aaa)
  4. Change Request Workflow - (10003aaa)
  5. Change Notice Workflow - (10003aaa)
  6. Change Activity Workflow - (10003aaa)
  7. Promotion Request Approval Process - (10003aaa)
  8. Promotion Request Review Process - (10003aaa)
  9. Approval Process - (1001faaa)
  10. Notify Process - (1005faaa)

- 11. Release Process - (1005faaa)
- 12. Review Process - (1005faaa)
- 13. Two Level Approval Process - (1001faaa)
- 14. Change Notice Workflow - (10003aaa)

Activity templates of selected process:

>> 1. Submit - (10003aaa)

Set template configuration

-----

- 1. Select process template
- 2. Show/edit process template configuration
- 3. Select activity template
- 4. Show/edit activity template configuration
- 5. Refresh templates
- 6. Exit

>>> Choose an option:

Selecting an option toggles the value to either true or false.

## Saving Your Work

To save the configuration, select either option 30 or 31. Make sure that the relevant workflow process template is checked out. Also, make sure that all existing instances of the relevant process template are terminated. If one of these conditions is not met, an exception is thrown.

New workflow processes will use the new behavior defined by these execution flags.

## Electronic Signatures

The Windchill products allow you to require electronic signatures on workflow activities to authenticate the activity.

### Setting Up for Electronic Signatures

If the eSignatures are in an LDAP directory that has already been established for Windchill, there is no additional setup required. If the LDAP directory is not used with Windchill, see the *Configure Windchill to Use an Enterprise Directory* chapter in the *Windchill Installation and Configuration Guide - Windchill* for information about creating a JNDI adapter entry and repository description. The `wt.org.electronicIdentity.authenticationService` property must identify the directory.

From a windchill shell, execute the following commands:

- To display the current value of the property:

```
xconfmanager -d
wt.org.electronicIdentity.authenticationService
```

- Specify the existing and new value (append new value to the existing property value). You can specify one or more JNDI adapter Service Names. Use a comma to separate the adapter names. See the xconfmanager guidelines for specifying multiple property and property value combinations:

```
xconfmanager -s
wt.org.electronicIdentity.authenticationService=<JNDIadapter
service name(s)> -t <Windchill>/codebase/wt.properties -p
```

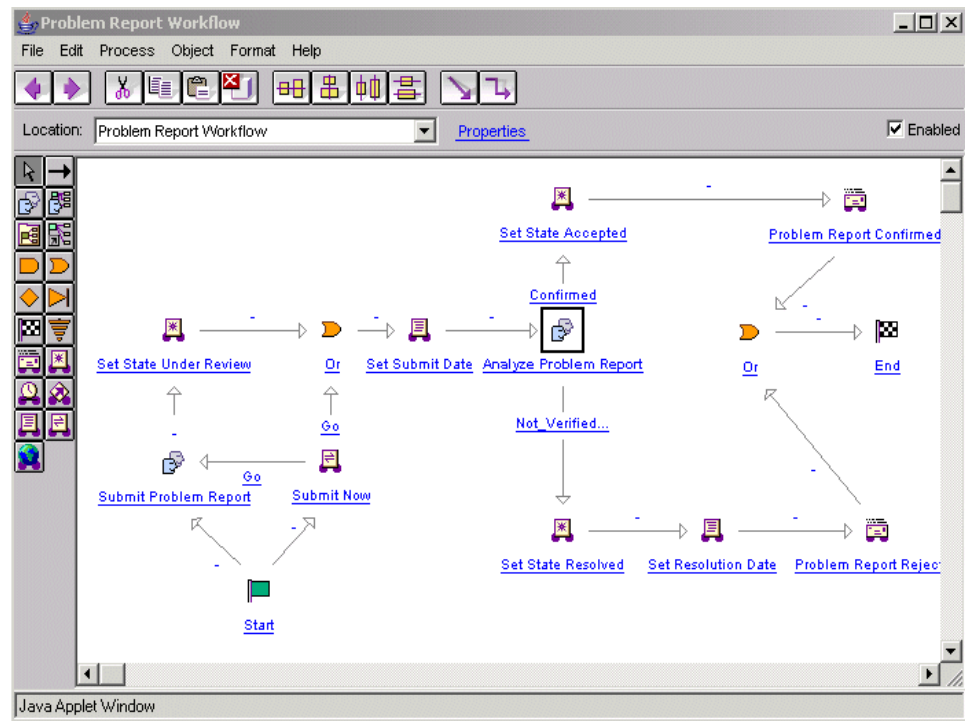
Where <Windchill> is the location where Windchill is installed.

- The parameter wt.org.electronicIdentification.class can be set to determine the authentication required. The parameter can be set as follows:
  - wt.org.electronicIdentity.engines.LDAPPasswordSignatureEngine – this signature engine only requires a password to be supplied. It verifies that the password belongs to the currently logged in user.
  - wt.org.electronicIdentity.engines.LDAPFDACompliantSignatureEngine – this signature engine requires both a user name and a password field to be entered, in compliance with FDA rule 21 CFR part 11 section 11.2. It verifies that the user name and password belong to the currently logged in user.

## Requiring Electronic Signatures in a Workflow

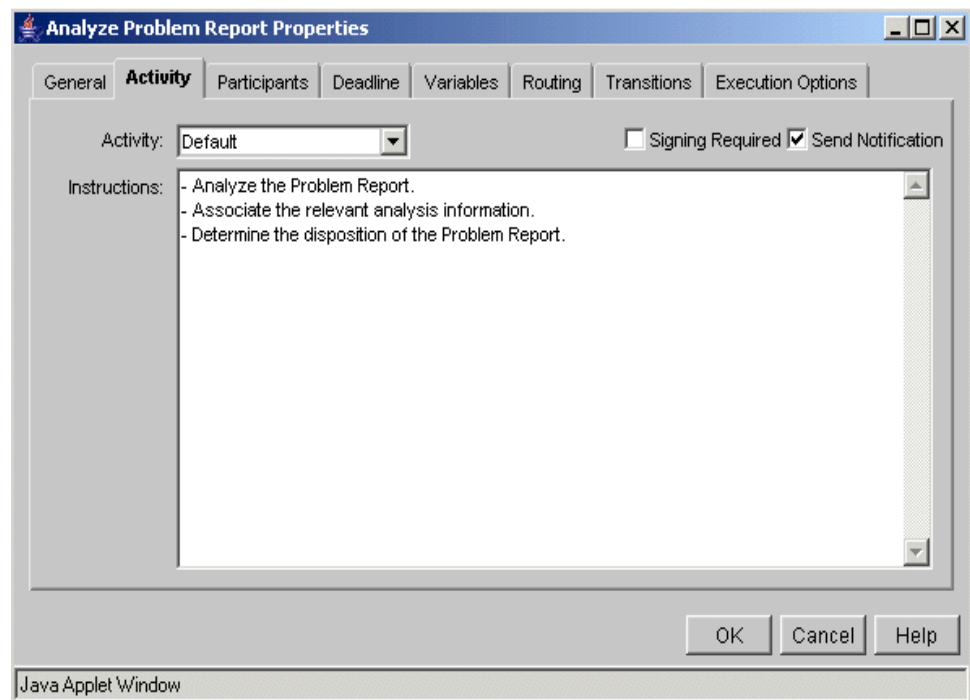
To require electronic signatures, perform the following steps:

1. Use Workflow Administrator to make an electronic signature required for a particular activity. When you create or update a process template, the process window appears.

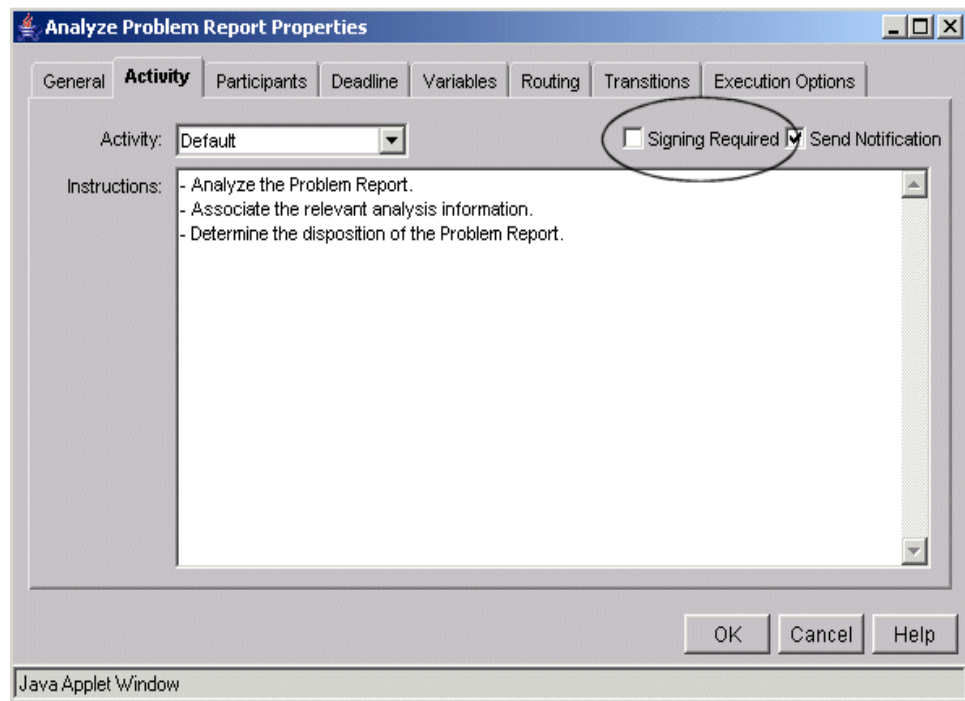




2. If you double-click the activity or right-click the activity and select **Properties**, a properties window appears. In this window, select the **Activity** tab.



3. Select the **Signing Required** check box to require an electronic signature at this point in the process.



**Note:** If you select **Signing Required**, you are prompted for authentication in order to start the activity.

## Best Practices

### Access Control

Access control is generally defined, using the Policy Administrator, on the domain to which the object belongs. However, it is possible to specifically add additional access policies to a given Windchill object (such as a specific WTPart or WTDocument). For example, you can use ad hoc access policies inside Windchill workflow, or you can define object-level access policies in Windchill ProjectLink.

If you are likely to have an extremely high number of running workflow activities referencing the same object, make sure that you grant ad hoc access only as needed. Use Policy (static) access control where appropriate. For more information, see [Administering Access Control](#).

## Using a Single Workflow in a Life Cycle Having Multiple State

Duplicate tasks get started when a set state robot is used along with a loop link to the task related to the new state.

In order to avoid this situation, create an initial dummy life cycle phase that is associated with the workflow and that represents a state that is never returned to in a loop or in a set state robot.

## For Windchill PDMLink and Windchill ProjectLink

Site, organization, and solution administrators manage workflow templates.

- Site administrators create, modify, delete, and view workflow templates in the site context.
- Organization administrators create, modify, delete, and view workflow templates in the given organization context. Organization administrators can view workflow templates from the site text.
- Product and library administrators create, modify, delete, and view workflow templates in their respective context. They can view organization templates from the parent organization context and the site context. This includes administrators of Windchill PDM library contexts.

**Note:** The Workflow Administrator is not available to administrators of project contexts.

The Workflow Administrator client displays a table that lists all workflow templates belonging to the given context, plus those belonging to its parent contexts. A column in the table identifies the context owning each workflow template.

When you create a workflow process in Windchill PDMLink, the system saves the new workflow process in the System cabinet of the context in which it is created. The system, not the user, decides where the new workflow template is to be located.

**Note:** When you assign a workflow template to a life cycle template, you see a list of valid workflows. The list of valid workflow templates includes the ones defined in the given solution context, plus those defined in the ancestor organization and the site contexts. Workflow templates defined in a sub-context override and filter out the workflow templates defined in parent contexts having the same name.

## Workflow Process Support in Windchill ProjectLink

The following table shows the workflow process support in Windchill ProjectLink.

Feature	Support Level	Notes
Use of workflow administrator's interface	Partial	Users can see all templates in the context from which they launched the workflow administrator and up the hierarchy. For example, if you launch the workflow administrator from Windchill Foundation & PDM, you see only templates from the site context. If you launch from library, you will see templates from the library and site contexts. The site templates are only available to view or save. In Windchill ProjectLink, you can launch the workflow administrator applet from the Utilities page under Organization tab or the utilities under the Site tab. The templates shown on the applet depends on the tab from which it is launched.
Use of workflow process editor interface	Full	The full capabilities of the process editor are exposed. There is no bootstrap capability with Windchill ProjectLink; so a high bandwidth connection to the server is necessary to stream the workflow process editor applet to the client.
Default task choices	Partial	Life cycle and default task types are supported. Setup participant task types and change tasks are not supported.
E-mail task delivery	Full	All tasks can be delivered by e-mail, in addition to being added to the user's assignment table.
Ad hoc task support	None	There is no support for ad hoc workflow tasks.
Custom task form used and definition	None	Windchill ProjectLink task forms are rendered as JSP pages. There is currently no capability to define or select custom task forms. A future release may support custom task form definition.
Electronic signature support	Full	Windchill ProjectLink requires additional authentication, as specified by the signature engine, for tasks generated from activities that require signing.
Activity-based access control	Partial	Activity-based access control may be used for the primary business object only. Access right may only be added to the rights currently defined for the object. Access rights cannot be removed.

<b>Feature</b>	<b>Support Level</b>	<b>Notes</b>
Variable support	Partial	All standard variables are supported. Variables the represent Windchill classed classes can be defined and used, but will not be rendered in the task for user interface.
Routing	Full	Exclusive and nonexclusive manual routing is supported. Automatic routing is supported via expressions. Routing selections are displayed only for required roles.
Robots	Full	Notification, set state, timer, execution, and expression robots are all supported.
Connectors	Full	Start, stop, AND, OR, and threshold connectors are all supported.
Process references	Full	Only processes defined for the organization associated with the logged-in user may be referenced.
Blocks	Full	
Transition conditions	Full	
Roles	Partial	Roles that are referenced must be satisfied in the project instance associated with the process instance.
Voting	Full	
Variable substitution	Full	Variable support is limited. Variables representing Windchill classes can be defined and used, but will not be displayed in the task pages.
Import/export	Partial	Process templates can be exported to XML format. There is no organization association with the imported or exported template so the workflow import/export folder should be cleared between uses.
Process initiation	Partial	A workflow process may be instantiated indirectly only by creating a life cycle managed object or by initiating a routing action. A workflow may also be started programmatically. There is no user interface to explicitly start a workflow process in Windchill ProjectLink.

<b>Feature</b>	<b>Support Level</b>	<b>Notes</b>
Participation identification	Partial	Participation in a downstream task can be accomplished through the use of user and group variables that are satisfied by an assignee in a task form. You can add or remove assignees from a running process by updating the route participants. The change applies to unstarted started activities, it will not affect tasks that are currently running. There is no support for the setup participants capability.
Content holder access	None	There is no workflow content holder exposed in a workflow task.
Ad hoc tasks	None	Ad hoc tasks cannot be created from a parent task.
Access to primary business object	Partial	A link to the subject is provided.
Adding comments	Partial	A link to the subject's discussion forum is available in the task form. Comments can also be added to a string variable.
Task actions	Partial	A user can open and reassign task items, but there is no capability to delegate them or to access a user calendar. When tasks are reassigned, the assignee pool is the project team.
Assignments customization	None	The Assignments tables support multiple predefined views There is no capability to further define custom views.
Project assignments view	Full	There is a specific view of all assignments in a project.
Access to the Workflow Process Manager	Partial	There is no access to the process manager for a general workflow user.
Access to workflow participation or execution history	Partial	Windchill ProjectLink exposes no explicit access to workflow participation or history information for non-administrators. However, individual task forms can accumulate information regarding participation and disposition. Comment history can be captured in a discussion forum associated with the primary business object.
Local search support for process information	None	There is no support for searching running or completed processes.

Feature	Support Level	Notes
Workflow process management	Partial	





# 17

## Administering Views and View Associations

Topic	Page
Overview .....	17-2
Views and View Associations .....	17-2
Managing Views and View Associations .....	17-3

## Overview

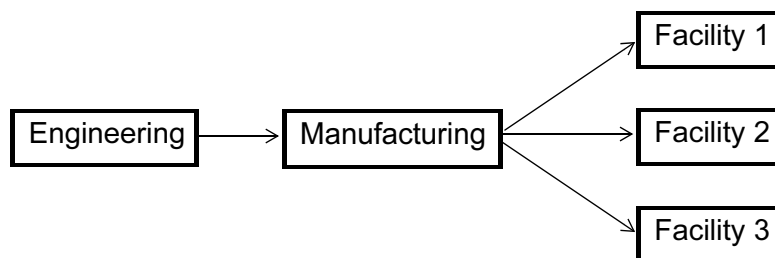
Within Windchill, a part is assigned to a view when it is created. A new view version can be created with the **Selected > New View Version** menu selection of the Product Structure Explorer.

A part may be a *view-dependent* object, if several versions of the part may be required to address the needs of the various organizations working on it. For example, the Engineering and Manufacturing departments may want to work on different versions of a part, with each version representing a view specific to that department's needs.

## Views and View Associations

Before a part can be assigned to a view, you must set up views and view associations for your Windchill system. Each view you configure must have a unique name. A view can have many child views, but only one parent view.

Only the first view in the structure has no parent view. You can define only one such *root view* for your site. The following diagram provides an example of a typical view setup:



View-dependent versions of a part can be derived from parent views. As shown in the diagram above, the Engineering view is the first view in the structure and is the parent of the Manufacturing view. Also, the Manufacturing view is the parent of the Facility 1, Facility 2, and Facility 3 views. Therefore, a Manufacturing version of the part can be built from the Engineering version of the part. Facility 1, Facility 2, and Facility 3 views can be derived from Manufacturing.

When a user creates a view-dependent version of a part, the new version is assigned an initial revision letter, which is prefixed with the revision letter of the part version from the parent view. For example, if the Engineering version of a part is Revision B, the Manufacturing version of the part is labeled Revision B.A. Each view-dependent version of a part goes through its own life cycle process.

As described in the next section, Windchill provides a command-line utility to assist you in defining and managing views and their associations.

## Managing Views and View Associations

You can use the `wt.vc.views.LoadViews` command-line utility to create and manage views and view associations. To invoke this utility, enter the following command from within a windchill shell:

```
windchill wt.vc.views.LoadViews
```

When launched, this utility displays the following selections:

```
[V]iew structure
[C}reate view
[R]ename view
[D]elete view
[A]ssociate views
[I]nsert view
[Q]uit
Your choice:
```

Type one of the bracketed characters at the **Your choice** prompt to select a command. For example, to create a view, enter **C**.

**Note:** Enter the uppercase character representing your selection. Lowercase values are not recognized.

### Examining a View Structure

To examine the existing view structure, enter **V**. The structure is indented to illustrate parent/child relationships. For example, the view structure for the preceding diagram is as follows:

```
Engineering
  Manufacturing
    Facility 1
    Facility 2
    Facility 3
```

### Creating a View

To create a view, enter **C**. You are prompted to enter a name for the new view. The name you specify must be a unique view name.

### Renaming a View

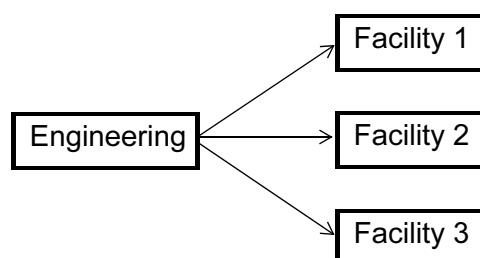
To rename an existing view, enter **R**. You are prompted for the current view name, which you intend to change. After you have specified the existing name, you are prompted to enter a new name for the view. Again, the name you specify must be unique.

## Deleting a View

**Note:** After parts have been created in a view, you cannot delete the view. Attempting to delete a view that has associated parts, returns an error message.

To delete an existing view, enter **D**. You are prompted for the name of the view to delete.

When you delete a view, all child views of the deleted view become child views of the parent of the deleted view. For example, if you delete the Manufacturing view from the previous example shown in the [Views and View Associations](#) section, the Facility 1, Facility 2, and Facility 3 views become child views of Engineering as shown in the following diagram:



## Creating a View Association

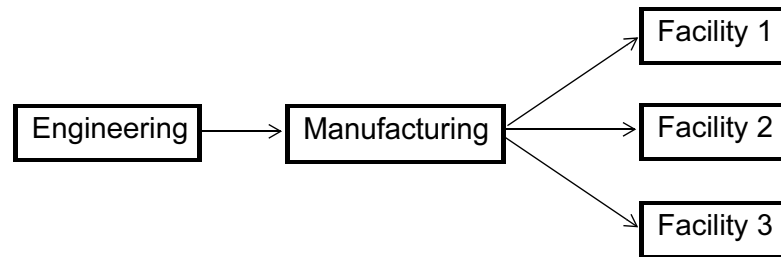
To associate two views, enter **A**. You are prompted for the name of the view that will assume the parent role in the new association. After you have specified a parent view, you are prompted for the name of the view that will play the child role. For example, the [Views and View Associations](#) section shows an association between the Engineering view and the Manufacturing view, in which the Engineering view is parent to the Manufacturing view.

## Inserting a View

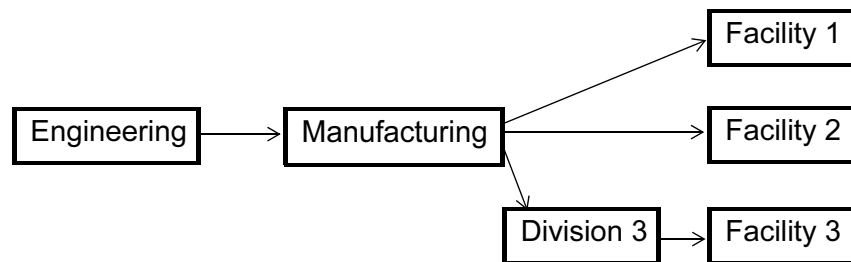
To insert a view into the view structure, enter **I**. You are prompted to enter the name of the parent view. Specify the name of a view below which the new view will be inserted. Next, you are prompted to enter the name of a view that is currently a child of this parent. You can either enter a view name, or click **Return**. Finally, you are prompted for the name of the view to be inserted.

If you provide values for both the parent name and the child name, the inserted view becomes a child of parent view and parent of the child view.

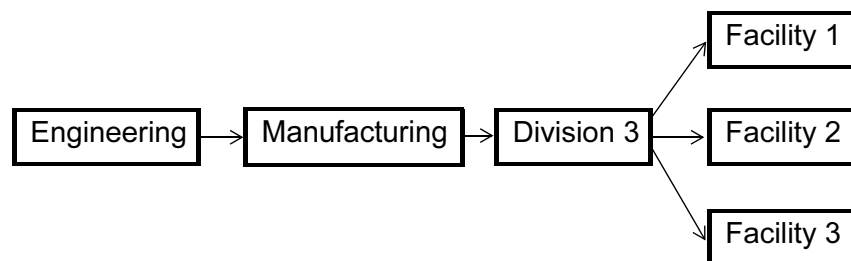
For example, assume the following views exist (as described earlier in this section):



If you specify **Manufacturing** as the parent view and **Facility 3** as its child, and then insert a **Division 3** view. The Division 3 view would be a child of Manufacturing and new parent of Facility 3, appearing in between the two as shown in the following diagram:



If you do not enter a child view when prompted, the inserted view will appear between the parent view you specified and all of its current children in the view structure. For example, assume you enter **Manufacturing** as the parent view as described in the previous example, but do not enter a view name when prompted for the name of a child view. Then, the **Division 3** view you insert is a child of the Manufacturing view and a parent of the Facility 1, Facility 2, and Facility 3 views:





# 18

## Administering Visualization Services

Topic	Page
Overview .....	18-2
Architecture .....	18-3
Troubleshooting.....	18-6
Visualization Collaboration.....	18-21
Exporting Watermarks for ProductView.....	18-22
Copy Forward Functionality .....	18-23
Windchill Visualization Service Properties .....	18-26

## Overview

To enable the Windchill Visualization Service (WVS), you must install Windchill and follow the configuration steps. This section is an overview of WVS functionality and architecture, providing a context for the troubleshooting guidelines in the remainder of this chapter.

## File Types

ProductView can display many file types. However, it is important that you are familiar with the following four file types:

- OL files

An OL file is a binary file, which is created by publishing a CAD part. It contains the 3D and 2D CAD information. A single CAD part may create many OL files.

- PLT files

The PLT file contains 2D-vector information, and is created when drawing output is requested during publication of a CAD part. A CAD part can produce many PLT files.

- ED files

An ED file is an ASCII file that contains product structure and file relationship information. Within the context of a single CAD part, an ED file is used to associate OL and PLT files. A part-level ED file can also contain attribute information from the CAD part. When an assembly is converted or a product structure is traversed, the resulting ED file contains the component nodes, as well as their relationship (to form the hierarchy) and attributes. Orientation and units are held as attributes of the components. The OL and PLT files are also associated with the components. Files may be associated as a URL or as a file system reference.

**Note:** The conversion of a CAD part can result in an ED file that contains substructure information, in addition to OL and PLT file references. Consequently, when a product structure is traversed during publishing, any component node that has an ED file associated must have the substructure merged into the resulting file to give the complete product structure definition, down to the subpart level.

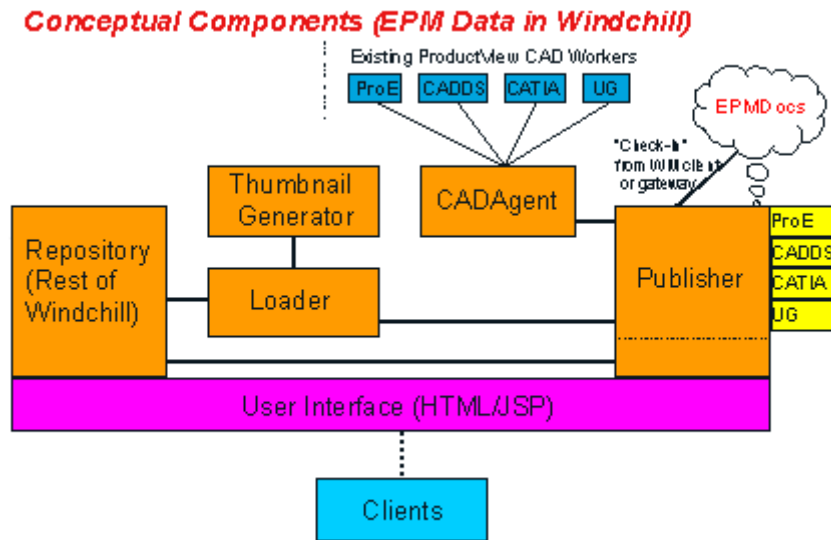
- EDZ files

The EDZ file is a ZIP file containing an ED file, and all associated OL and PLT files. The EDZ acts as a snapshot of a product structure. As it is a single file, it provides a faster way to access a large amount of information, as only one download allows you to view a complete product. The EDZ also provides a single file that can be easily exchanged (for example, through e-mail).



# Architecture

The following figure is a graphical representation of the Windchill Visualization Structure architecture.



WVS allows Windchill users to generate viewable files, store those files in the Windchill database, and view data in ProductView. ProductView displays many document formats directly from the file, requiring no preparation. However, CAD data must be published before it can be viewed in ProductView.

The loader is responsible for preparing data for storage in Windchill before it is converted. The loader can be used in two ways:

- As a Windchill service, which looks for ticket files in a directory.

A *ticket file* is a text file that defines the location of the preconverted data and specifies the way in which it will be catalogued when stored in Windchill. See the [CAD Agent](#) section in this chapter for guidelines on how to analyze and correct problems in this process.

- As an operation called directly by the publisher.

Calls from the publisher are performed programmatically. The data is handled in the same way as if it were loaded through a ticket.

The loader can optionally call the thumbnail generator to create a JPG image and 3D thumbnail file of the 3D geometry. If required, the thumbnail generator can be configured by its recipe file to only create a JPG image. The thumbnail generation must be installed from the Visualization - Windchill Support CD.

Data stored in Windchill by the loader is stored on a Representation, which is associated with a Representable (currently a WTPart, WTDocument, or EPMDocument).

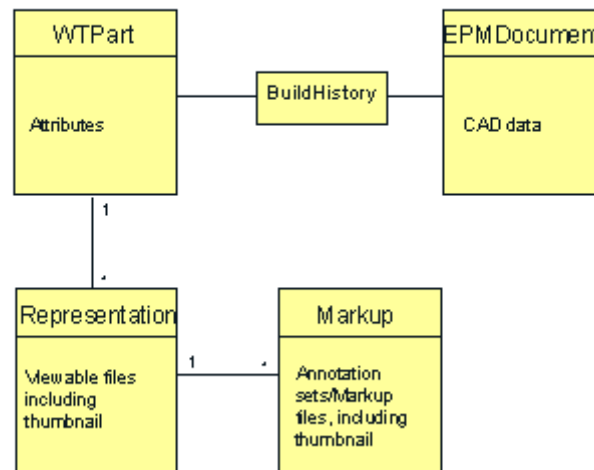
The loader can also create an EDZ file, which is stored as role PRODUCT\_VIEW\_EDZ on the Representation. All the data files are loaded as secondary content on the Representation. The ED (product structure) file is processed, so the references to the viewable files point to Windchill content through URLs. The ED file is then stored as role PRODUCT\_VIEW\_ED on the Representation.

If a thumbnail image or 3D thumbnail file has been created, it is stored on the Representation as content role THUMBNAIL/THUMBNAIL3D. If this is the default Representation, the Representable also receives a copy of the thumbnail content. The actual content in the database is not duplicated, if it is shared with the Representation.


From any visualization link, you can view the data stored on a Representation in Windchill within ProductView. ProductView can author Windchill information in the form of annotations and groups saved as a markup in the database. Markups are associated to a particular Representation.

The following figure shows the conceptual Windchill visualization data model:

### **Persistent Server Objects**



If no WTParts are created in Windchill through EPM operations, the Representation is associated to the EPMDocument. Where the CAD data has been stored in Windchill by an EPM client, (for example, the Workgroup Manager for Pro/ENGINEER) or is referenced by an EPM gateway (for example, the Pro/INTRALINK Gateway), WVS can publish and store the data. The publishing of data can be accomplished upon the following:

- Using the Create Representation wizard.
- User request by clicking the publish icon .
- Checkin of data from the EPM client or gateway.
- Change of life cycle, life cycle state, team, or folder within Windchill.
- Execution of a scheduled job.

A publish job is always created and passed to a Windchill processing queue. The publish job is *self-logging*; that is, all end-user messages are contained within the job itself, and they can be viewed through the WVS publish monitor.

When the publisher receives a publish job, it traverses structures within Windchill and extracts necessary CAD files. The traversal and file selection is based on the type of data being processed. The publisher uses the CAD agent to schedule the conversion of the CAD data.

The CAD agent has CAD workers configured to it. A *CAD worker* is a program written using the API of a particular CAD system, and it produces files that ProductView can read from native CAD files and assemblies. The CAD agent manages the CAD worker resource.

The following are characteristics of the CAD agent:

- Aware of which CAD workers are configured.
- Can start and stop CAD workers as necessary.
- Allow viewing of CAD worker log files.
- Allow CAD worker to be tested during configuration.
- Manages the passing of data to and from the CAD worker.

The CAD worker typically runs on a remote system. When the CAD agent receives the published data, it returns it to the publisher. The publisher then stores the resulting data in Windchill, by programmatically invoking the loader.

# Troubleshooting

This section provides information you can use to analyze and resolve issues that may arise with the WVS components.

## Windchill Visualization Service Loader

The method server starts the WVS loader service. The relevant entry in the wt.properties file is in the following form:

```
wt.services.service.nn=com.ptc.wvs.server.loader.GraphicsServer  
LoaderService/com.ptc.wvs.server.loader.StandardGraphicsServerL  
oaderService
```

The service first reads the wvs.properties file to ensure that the wvs.enabled property is set to true. If this property is not set to true, the service is not started.

The loader may output additional debugging information to the method server start window and log if the edrload.verbose property is set to true.

Every 5 seconds, the loader polls the directory defined by the following property:

```
edrload.directory=$(wt.temp)\\wcinput
```

If this directory does not already exist, the loader creates it. When polling the directory, the loader looks only for INI files. All other files are ignored. If the contents of an INI file (located in the directory) are terminated with <!>, it is renamed with a .txt extension. (For example, ticket.ini would be renamed ticket.txt.)

If content is not terminated with <!>, the loader waits 5 more seconds to make sure the file is not currently being written to. If, after 5 seconds, the file content still does not terminate with <!>, the file is deleted. The loader requires write access to the file so that it can rename or delete it.

In order to start processing them, the loader next parses the file and validates the contents.

The file should contain entries of the form Keyword=value (for example, Partnumber=123456). The following table lists valid keywords:

Keyword	Value or Description
Directory	Specifies the fully qualified directory location of the converted data.
Documentnumber	Specifies the number of an existing WTDocument to which you associate the representation.
Documentversion	Specifies the version of the WTDocument to which you associate the representation.

<b>Keyword</b>	<b>Value or Description</b>
Encoding	Specifies the character set encoding of the ED file (if the ED file has no J tag present). The default is to use the J tag-specified encoding or the encoding of the Windchill server.
Edzcreate	Specifies if an EDZ file is to be created. Can be set to true or false.  The default is false.
Ignoreonmerge	Adds a flag to children of the ED files root node to indicate that those children should be ignored when using this representation in a WTPart structure. For example, if the root WTPart includes a representation of the complete assembly, but you want to view the data from the individual WTParts when viewing a structure, use ignoreonmerge.
Includemarkups	Specifies if markups in the input data should be stored with the representation in Windchill. Can be set to true or false.  The default is true.
Iteratepart	Specifies if an existing part is to be iterated. Can be set to true or false.  The default is false.
Partfolder	Specifies the folder in which the part is created.
Partlifecycle	Specifies the life cycle associated with the new part.
Partcontainer	Specifies the context (for example, Project, Product, or Library) that in which a new part is created. The folder and life cycle values are determined by the context and need not be specified.  You can specify the context as a name or parentname/name. For example, if an organization PTC contains a project proj1, you can specify partcontainer as proj1 or PTC/proj1 (to distinguish it from other projects called proj1 in other organizations).
Partname	Specifies the part name.
Partnumber	Specifies the part number of an existing part (the part number is created if it does not exist).

Keyword	Value or Description
Partoid	Specifies the Windchill ID of an existing part.
Partteam	Specifies the team associated with the new part.
Partrevision	Specifies the part revision.
Repdefault	Specifies whether the representation is to be the default. Can be set to true or false.  The default is false.
Repdesc	Describes the representation to be created.
Repname	Specifies the name of the representation to be created.
Representableoid	Specifies the Windchill ID of an existing representable.
Thumbnailcreate	Specifies whether a thumbnail is to be created. Can be set to true or false.  The default is false.
Ticketencoding	Specifies the character set encoding of the ticket file. If specified, this must be the first line in the ticket file. If not specified, the encoding of the Windchill server is assumed.

**Note:** The file must end in <!.>. Keywords are case-insensitive, as are true and false values.

The initial checks of the file ensure that the directory specified by the Directory keyword exists and that the loader can write to it.

- Additional checks include the following actions, some of which depend on keyword values:
- If a Partoid is specified in the file, it is checked to ensure that it references a valid WTPart.
- If a Partoid is not specified, the Partfolder, Partlifecycle, and Partteam values are checked to ensure that they exist.
- If the Partnumber/Partname does not exist, a WTPart is created. If it does exist, and Iteratepart is set to true, the part is iterated. The result is a WTPart in the database to which a new representation will be added, with the specified Repname and Repdescription.
- The specified directory is scanned to locate the ED file. Only one ED file is allowed. All other files are uploaded to Windchill, associated as secondary content of the representation.

- If a Representableoid is specified in the file, it is checked to ensure that it references a valid representable.
- If the Thumbnailcreate keyword is set to true, a thumbnail image is created and uploaded as content of the representation, provided the thumbnail generator has been installed. If it is the default representation, the thumbnail is copied (shared) to the Representable.
- If the Edzcreate keyword is set to true, an EDZ file containing all the files in the directory is saved as content of role PRODUCT\_VIEW\_EDZ on the representation, provided edrload.edzenabled=true is also set in wvs.properties.
- The ED file is modified to reference the secondary content in Windchill, rather than the local files, before being stored as the role PRODUCT\_VIEW\_ED on the Representation.
- The loader removes the ticket.txt file from the directory it is polling. This occurs whether the loading task succeeds or fails. If an error occurs, it is reported only in the method server log. Data referenced by the ticket is not removed. The removal of the ticket.txt file signifies that the loader has completed its task.

**Note:** For a large assembly, the loader task can be time-consuming, especially if thumbnail generation is performed. For more information about generating thumbnails, see the *Windchill Installation and Configuration Guide - Visualization Services*.

## CAD Agent

In order to properly configure the CAD workers to the CAD agent, the files must be configured correctly, and when CAD workers are on a remote system, the network connectivity between machines must be correct.

The CAD Agent usually runs as a service, and is defined in the wt.properties file as follows:

```
wt.services.service.nn=com.ptc.wvs.server.cadagent.CadAgentService/com.ptc.wvs.server.cadagent.StandardCadAgentService
```

It can also run as a standalone executable.

The CAD agent reads configuration settings from a file. The name of the file is defined by the following entry in wvs.properties:

```
cadagent.inifile=$(wt.home)\\codebase\\agent.ini
```

The basic settings in this file can be altered using the CAD Agent Configuration Wizard. To access the wizard, click the **Configure** button on the CAD Agent Monitor. You must have administrative permissions in order to use the wizard.

The CAD agent listens on a port for requests. The port number is defined in the agent.ini file. By default, this value is set to 5600. If another process on the system uses this port, the CAD agent fails to initialize. If that happens, you must manually edit the agent.ini file to change the port. See [Manually Starting CAD Workers](#) for more information.

If a Windchill cluster environment is used, the CADAgent is executing in a single background method server only, and for the CAD Agent administration UI to operate, the host name where the background method server is running the CAD Agent must be manually added to the agent section of the agent.ini file by adding the following line:

```
host=<hostname>
```

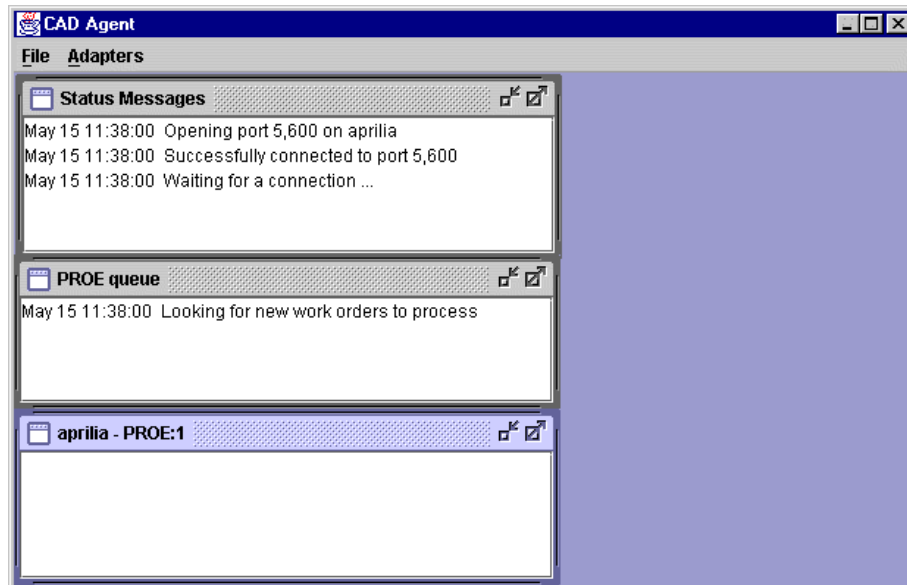
## Running the CAD Agent in Debugging Mode

If you have problems configuring the CAD agent and getting a CAD worker to connect, start the CAD agent in debugging mode. Use the following procedure:

1. Stop the method server if it is running.
2. To start the CAD agent, execute the following command from a window configured to run Windchill:

```
java com.ptc.wvs.server.cadagent.CadAgent -d
```

The **CAD Agent** window opens.



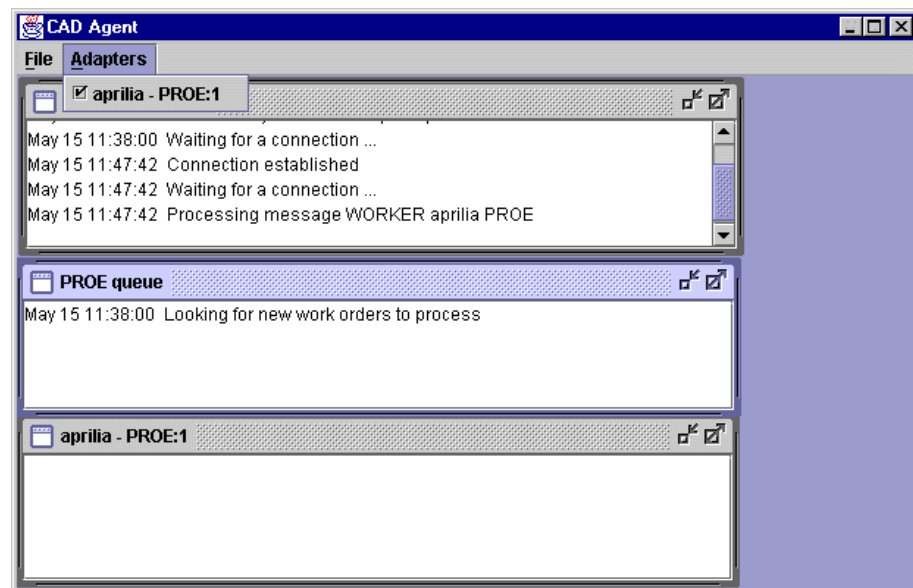


The number of panes displayed depends upon the numbers and types of configured workers. The Status Messages pane displays the requests being processed by the CAD agent on the listening port (5600). For each type of CAD worker (Pro/ENGINEER, CADDs, CATIA, Unigraphics, and so on), a pane displays the state of the queue. Each worker has its own pane that logs its transactions. In this case, there is a single Pro/ENGINEER worker configured, which is local to the server on which the CAD agent is executing.

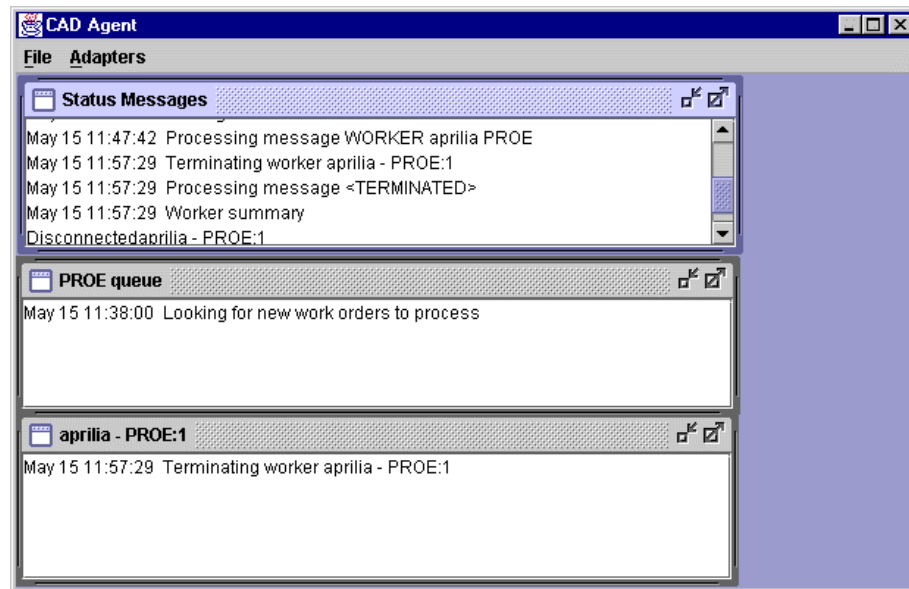
## Manually Starting CAD Workers

To manually start a CAD worker, you must be logged into the system on which the worker is configured. From there, execute the BAT file or shell script that has been configured as the command to start the worker.

If you are successful, messages are displayed in the Status Messages pane, and the checkbox for the worker, in the **Adapters** drop-down menu, is selected.



To stop the worker, select the appropriate **Adapters** entry by clicking its check box. This sends a message to stop a running worker.



If the CAD worker fails to connect to the CAD agent, no activity is displayed in the CAD Agent windows. The CAD worker may continue running or exit after a period of time. An incorrect setting in the CAD worker configuration file typically causes this.

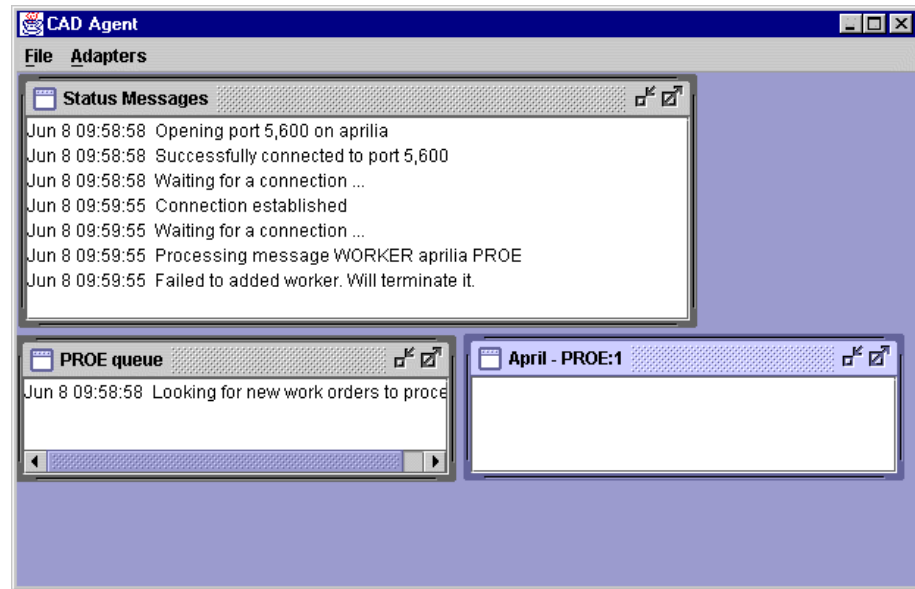
If the CAD worker and the CAD agent are not running on the same host, the CAD worker host must be able to communicate with the CAD agent host, as the name specified in the CAD worker configuration file.

For remote UNIX workers, the network routing, IP addresses, and DNS (name resolution) must be configured to resolve machine hostnames and permit traffic from the server to the remote machine via ftp or telnet. The remote machine should correctly service ping, ftp, and telnet requests to its hostname. (The telnet command is not required for remote Windows NT systems.)

If the following message appears in the Status Messages pane when you are attempting to manually start a worker, it is because a request is being made from a worker that is not recognized as being configured in the agent.ini file. The host name that is specified in the message must be the host name that is specified in the agent.ini file.

```
Processing message WORKER <hostname> <CADTYPE>  
Failed to add worker. Will terminate it.
```

In the example window that follows, the CAD agent is looking for a host name of aprilia, but the hostname is set to April in the agent.ini file. In this case, the CAD agent sends a message back, telling the worker to stop.



## Starting CAD Workers from the CAD Agent

After you have successfully started a CAD worker manually, stop it and try to start it again from the CAD agent. To start it, select the entry for the worker in the **Adapters** drop-down menu. The entry stays selected until the start-up timeout is reached, or a connection is successful.

If this procedure fails, check to determine whether the execute command is correct.

- A worker on a remote Windows NT client uses the worker daemon, because a telnet server is not available. Ensure that the worker daemon service is running and is configured to listen on the port specified for the worker entry in the agent.ini file. By default this is 601.
- For remote UNIX workers, execute `nohup` and put the task in the background. For example, you would use a command such as the following for a remote CADDSS5 worker:

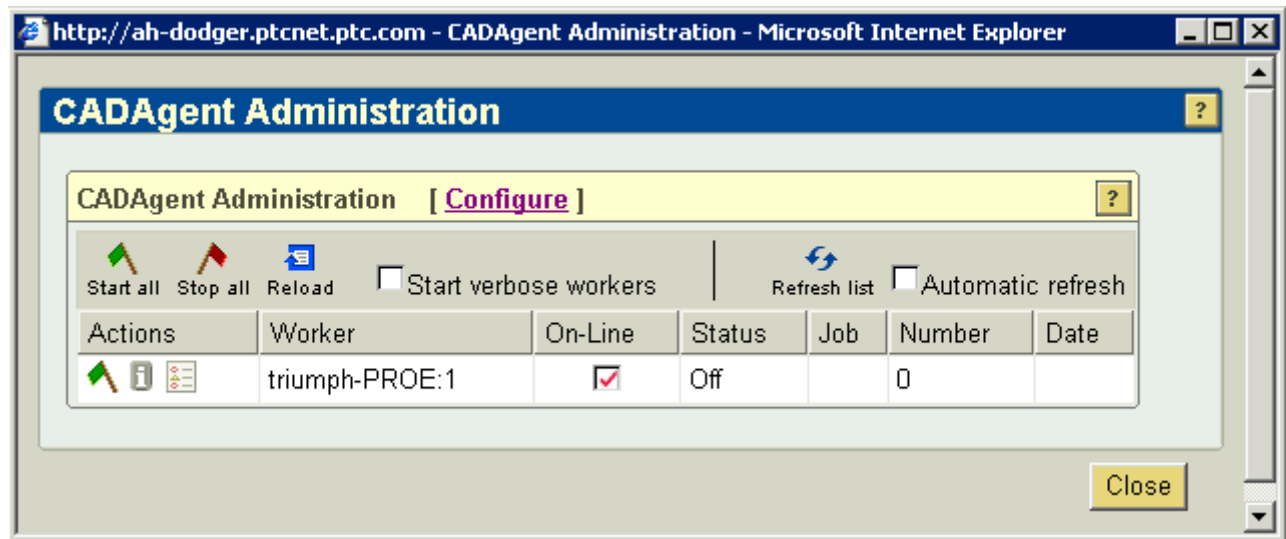
```
nohup cassaoa/cadds2pv &
```


Use the telnet command to connect to the remote worker.

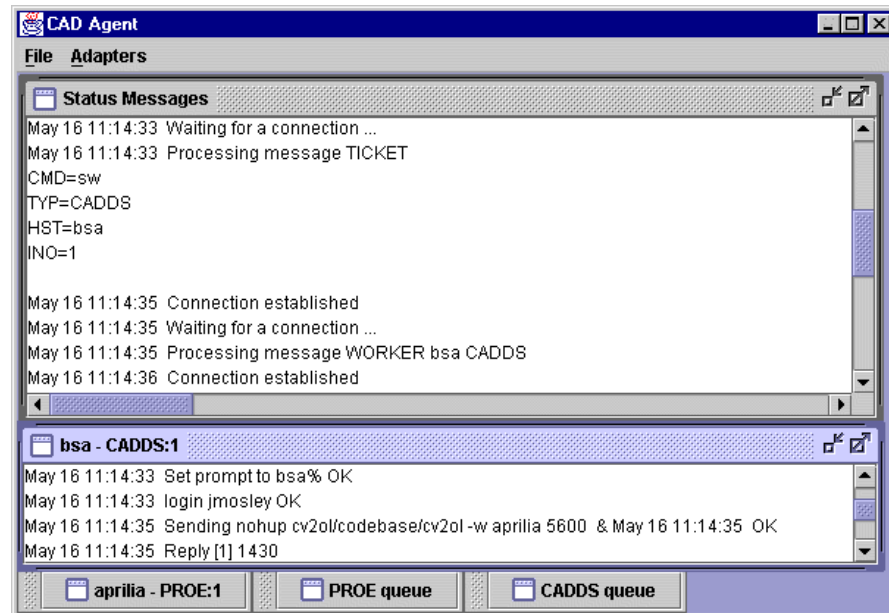
To test a connection, use the telnet command from the CAD agent host to connect to the worker host. Specify the host name, user name, and password defined in the agent.ini file when the worker was configured. If the connection is successful, ensure that the system prompt does not change. Manually executing the specified worker command from this environment should create a connection to the CAD agent. If not, there is probably a difference in the environment used by telnet and the default user login. Adjust the environment to ensure that the command causes a connection through telnet. Confirm that the environment settings are correct, particularly DISPLAY, path, and the shell. The CAD agent should then be able to start the worker.


### Starting CAD Workers from the CAD Agent Monitor

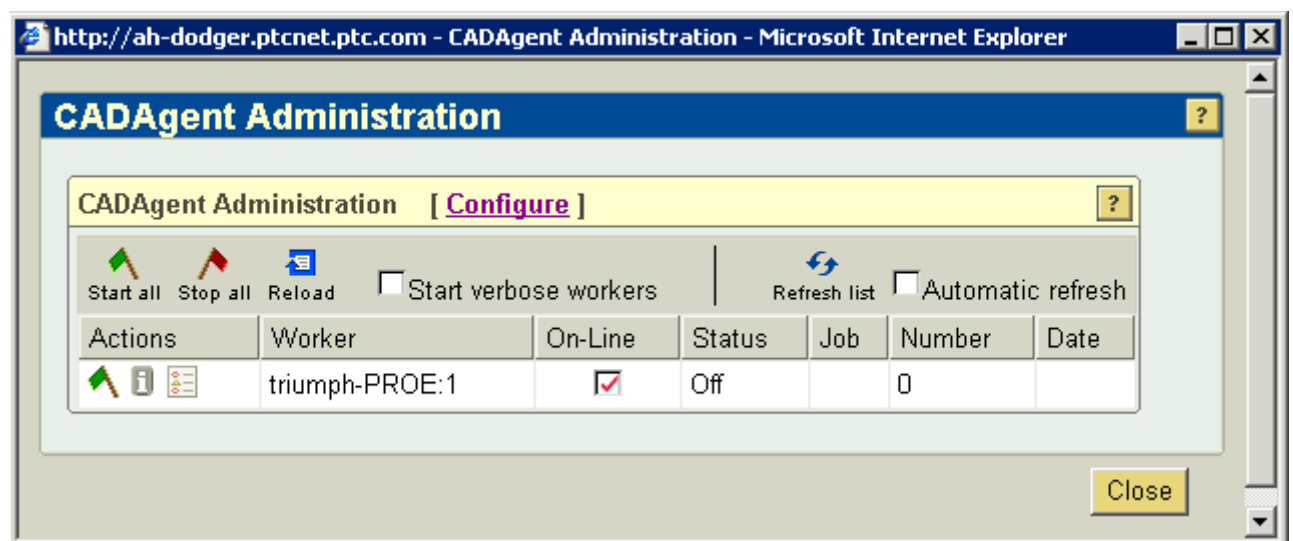
If the CAD agent is running in debugging mode, and the workers are disconnected, you can go to the CAD Agent Monitor to view the worker's status.



Click the "go" Action icon (  ) for a specific worker to send a message to the CAD agent to start that worker. Messages displayed in the **Status Messages** pane indicate that the agent has received the request and has attempted to start the worker. Additionally, the check box for the appropriate worker is selected in the **Adapters** menu list.



The status of the worker in the CAD Agent Monitor is then updated to show that the worker is available, and the Action icon changes to "stop" (  ), to indicate that clicking it stops the worker.



The worker will become available if the timeout period specified is too small, however, the UI stops waiting at the end of the time period, and won't be updated. If the CAD Agent Monitor indicates that the worker has connected, but its status is **Off**, select the **Automatic refresh** check box to update the UI. Click the **Configure** button to open the CAD Agent Configuration Wizard and change the setting. For detailed help, click **Show Help** on the **CAD Configuration Wizard**.

It is important that the worker starts within the specified timeout period. When the system is running, the worker should be started automatically. If the timeout value is too small, the CAD agent makes three attempts to start the worker before marking it as unable to start.

During setup, if there is not any CAD data stored in Windchill, the test icon in the CADAgent admin UI can be used to send a test file of the appropriate type to the worker. To use the test function, the worker should be started and off-line.


## Publishing CAD Documents






This section provides troubleshooting information for publishing operations.

### Manual Publishing

When the desired CAD worker is successfully connected to the CAD agent, the next step is to publish data. PTC recommends that you run the first test of this capability from the Visualization portal page.

To publish a CAD document, use the following procedure:

1. On a CAD document in the database.
2. From the Actions listing, click the publish icon ().

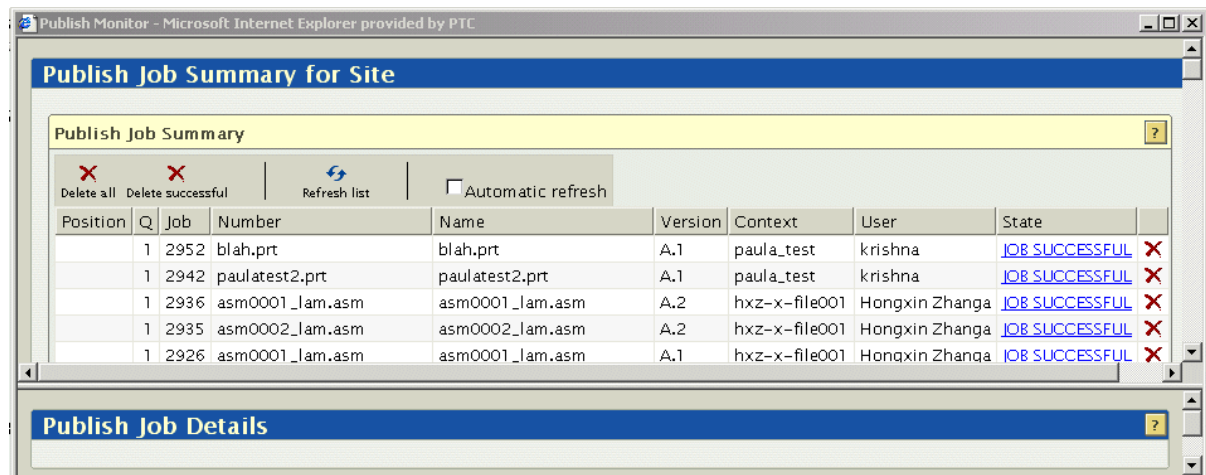
Results - CAD Documents : 1						
 Copy all		<input checked="" type="checkbox"/> Show all thumbnails				
Actions	Number	Name	Version	State	Type	Author App
   	<a href="#">ADJ.ARM</a>	ADJ.ARM	A	Default	CAD Component	CADDS5

The following window opens.



The first time the publisher process is used, four new processing queues, named PublisherQueueH, PublisherQueueM, PublisherQueueL, and PublisherQueueI, are created in Windchill. You can use the Windchill Queue Manager to confirm that these queues exist and are active.

3. To view the status of the job, click the **Publish Monitor** link.



Click the status link in the State column to display the job details. If no entries appear in this panel, or if they appear and then disappear when they are completed, check to confirm that the following properties are set in the wt.properties file:

```
#Visualization Publishing Queue

# Specify if the WVS PublisherQueue entries are to be deleted

# Delete the main queue entries once complete.

wt.queue.removeCompleted.PublisherQueue=true

# For each PublisherQueue keep the entries so that the log may
be seen in the publisher

wt.queue.removeCompleted.PublisherQueueI=false
```

```
# As extra queues get created, entries must be added here also  
wt.queue.removeCompleted.PublisherQueue2=false
```

By default, completed queue entries are removed. For the information to be available in the publish monitor, the completed jobs in the WVS "numbered" queues (eg. PublisherQueue1, Publisher Queue2, and so on) should not be removed on completion.

Initially all publish jobs are submitted to one of the processing queues PublisherQueueH, PublisherQueueM, or PublisherQueueL, which represent high, medium or low priority jobs. Only one job is executing in each queue. Other entries are set to Ready. The executing job in PublisherQueueH, PublisherQueueM, or PublisherQueueL looks for an available queue with a name of the form PublisherQueue<n>, where <n> is an integer. When a queue is available, the publish job is submitted to that queue, which immediately executes it.

Additional queues can be added to ensure scalability of the publishing capability. The processing queue should be created with the Queue Manager, and named with the next sequential integer, for example, PublisherQueue2, PublisherQueue3, and so on. For each additional queue, the appropriate entry must be added to the wt.properties file so that the completed entries from the queue are not removed:

```
wt.queue.removeCompleted.PublisherQueueN=false
```

**Note:** The publishing queue is displayed to all users, but job details are available only to the job owner. When the publish monitor is displayed in the context of a project or product, the publish jobs displayed are those associated to that context.

When the job completes, the details indicate success or failure. Errors that caused failures are identified in a message.

## Timeouts

When the CAD agent sends a request to the CAD worker, it has no way of determining the status of the job. Therefore, the CAD agent waits for a specified period of time. In the wvs.properties file, the following properties define timeout values for publishing:

```
publish.cadtimeout.component=600  
publish.cadtimeout.assembly=3600  
publish.cadtimeout.drawing=600
```

These properties specify the number of seconds that the CAD agent waits when the publisher is processing a single component, assembly, or drawing, respectively. These values should be adjusted to the needs of your site, so that they will process the largest data sets. If the values are too small, errors are displayed, and no viewable CAD data is created.



Alternatively, many of the CAD workers can be configured with long and short timeout values that are sent back to the CADAgent. If these have been configured, the last timeout value sent to the CADAgent is used. See the CAD worker documentation for details of setting CAD worker timeouts in its recipe file.

You should also tune the CAD agent settings for Auto Idle Stop and Auto Busy Stop to help control system resources. (These values are specified when you use the CAD Agent Wizard to configure a CAD worker.) For example, for CADD5, when processing of drawings is enabled, Auto Idle Stop should be set to about 900 seconds. For Pro/ENGINEER, setting Auto Busy Stop ensures that system memory is released on a regular basis.


When you set values that automate the stopping of CAD workers, you should enable Auto Start and correctly configure it so that the worker can be restarted.

## Automated Publishing

There are several ways in which the publishing of viewable CAD data can be automated. An event is emitted when a CAD document is checked in from a Workgroup Manager, the Pro/INTRALINK Gateway, or the Optegra Gateway. If the following property is set to true in the wvs.properties file, this event results in submission of a publish job:

```
publish.service.readytopublish.enabled=true
```

This job is processed exactly as if the user (who checked the file in) had made a

publish request using the Publish action icon  on the Property page. The config spec that is used for the publishing process, is provided by the application that emits the event, for example, a WGM. If that config spec is "null," WVS uses either the latest config spec or an as-stored config spec depending on the value of the property "publish.configspec.default.useasstoredifavailable."

## Publish Scheduler

With the Publish Scheduler, requests can be submitted to the Windchill Schedule queue for processing. Examples are provided below, but typically you will set automated publishing schedules specific to your site, to automate the publishing of certain types of data on a regular basis.

In Windchill PDM, you can access the Publish Scheduler from the **Administrator** page of the Visualization portal. In Windchill PDMLink or Windchill ProjectLink, you can access it from the **Utilities** page of the **Site/Project/Product/Organization** tab.

The first time that a scheduled job is submitted, a new Windchill schedule queue called WVSScheduleQueue is created. You can use the Windchill Queue Manager to confirm that this queue exists and is active. The schedule queue executes a publish request at the date and time specified, with the specified frequency.

Settings in the `wvs.properties` file define the jobs that are available for selection from the **Schedule Publish Job** page. There are a number of example jobs already configured. An appropriate schedule job can be created if you have specific requirements.

There are two parts to the process of creating a Schedule Publish Jobs:

- configuring the `wvs.properties` file
- writing the java code to select the objects to be published

The following property defines the list of available entries (where `<n>` is an increasing integer, starting with 1).

```
Schedulejobs<n>=<schedulename>
```

The *schedulename* is then used to find additional properties of the following forms:

```
<schedulename>.description=pull-down description
<schedulename>.class=<ClassContainingMethod>
<schedulename>.method=<nameOfMethod>
<schedulename>.enableOnContainers=<true/false>
```

The `<schedulename>.description` property defines the text that appears in the drop-down menu on the Schedule Publish Job page. The class and method are used to identify the specific method that will be invoked by the schedule job. The value of `enableOnContainers` determines if this schedule job is displayed in the list of jobs when the scheduler UI is invoked in a specific context. This indicates that the schedule job contains code to filter the objects to be published based on the context.

The following are the signatures of the publish job method:

```
public static QuerySpec <nameOfMethod>()
or
public static QueryResult <nameOfMethod>()
```

If a `QuerySpec` is returned, then `PersistenceHelper.manager.find()` is used to make the query and return a `QueryResult`. This `QueryResult` or the one returned directly from the `schedule jobs` method contains the EPMDocuments/WTParts/WTDocuments/Representations that are sent for publishing.

A default configuration spec is used. If the `QueryResult` contains a `Representation`, then that is sent for republishing. If the `QueryResult` contains a `WTDocument`, all publishable files (that is, those with `worker XXX=mapping` defined in `wvs.properties`) are sent for publishing.

To obtain the current container context for the schedule job, use the following method call in the jobs method:

```
WTContainerRef cr = com.ptc.wvs.server.schedule.ScheduleJobs.getCurrentContainer();
```

The following example schedule job method will publish all EPMDocuments in the current context:

```
public static QuerySpec allEPMDocuments()
{
    QuerySpec qs = null;
    try {
        qs = new QuerySpec(EPMDocument.class);
        WTContainerRef cr =
com.ptc.wvs.server.schedule.ScheduleJobs.getCurrentContainer();
        if( cr != null ) {
            ContainerSpec cs = new ContainerSpec();
            cs.addSearchContainer(cr);
            qs.setAdvancedQueryEnabled(true);
            qs.appendWhere(WTContainerHelper.getWhereContainerIn(cs,
EPMDocument.class),new int[]{0});
        }
    } catch (Exception e ) { e.printStackTrace(); }
    return qs;
}
```

To obtain debug information for a schedule job, set the following property to true in the wvs.properties file:

```
publish.publishqueuehelper.verbose=true
```

## Visualization Collaboration

Install the collaboration agent from the Visualization - Windchill Support CD, which adds the required entry into wvs.properties (via site.xconf) to specify the command to execute the collaboration agent. If required (for example, to change the assigned port numbers), you can change the value of the "collaboration.server" property.

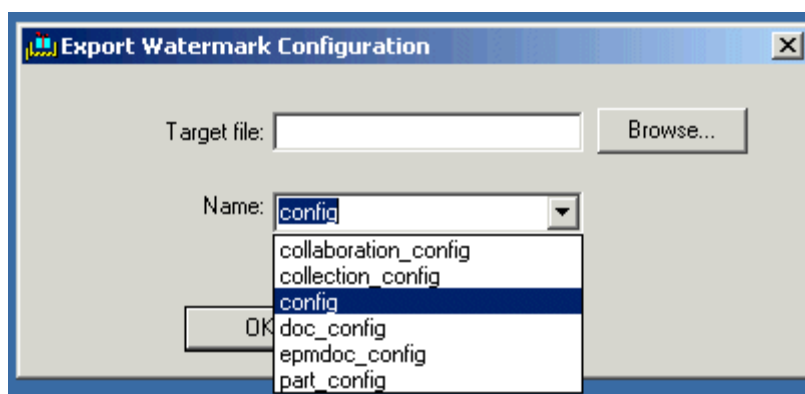
The options for the collaboration server define the minimum (-p) and maximum (-m) port numbers that are used by that server. Each collaboration session has its own running collaboration server, so this range should be large enough to accommodate the expected number of concurrent collaboration sessions. The default is 40. If the ProductView clients are operating through a firewall to the Windchill server, these ports must be open. The idle timeout can also be specified, in seconds, by the -i option. The default is to shut down a collaboration server if no one is connected for 10 minutes.

Only the user who starts a collaboration session can stop it or can host data into the session. All other users can only join the session. If the user who starts a session logs out and logs back in, he or she must go to the collaboration listing and make the session hosted into his or her current Windchill session before being able to load data into it.

## Exporting Watermarks for ProductView

ProductView supports watermarking of 3D, drawings, images, and documents. Watermarks are defined in INI files created and edited using the ProductView watermark editor. The administrator that manages watermarks manually transfers the INI files from the watermarks directory into the Windchill server.

To export the watermark configuration, go to the location that ProductView Standard Edition is installed. Run the wcexport executable file, found in the productview installation directory. The **Export Watermark Configuration** window appears:



Specify the file name you want to export and click **Browse** to locate the directory. The wcexport tool creates a ZIP file including the main config.ini file, any watermark INI files referenced by the registry definitions, and any images referenced by any watermark file.

To add the ZIP file to the ProductView configuration for Windchill Foundation & PDM, click **Administration** on the Visualization navigation bar; then click **Server Controlled Configuration of ProductView**. To add the ZIP file for Windchill PDMLink or Windchill ProjectLink, click **Visualization Configuration Administration** on the **Utilities** pages of the appropriate tabs.

**Note:** For Windchill PDMLink or Windchill ProjectLink, the context of the Utilities page defines where the watermark configuration is stored. For example, if you are accessing Visualization Configuration from the Organization tab, you are defining the watermark configuration for the organization.

For more information on ProductView/Visualization configuration, see the Windchill online help for the **ProductView/Visualization Configurations** table.

# Copy Forward Functionality

## Copying Representations Forward

This functionality controls the copying forward of Representations to new iterations of Representables (e.g., Windchill Parts, CAD Documents, and Windchill Document objects). The fundamental principal is that Copy Forward applies only to out-of-the-box, non-derived Representations. However, in the case of a Copy or Revise of a Representable, all Representations (derived and non-derived) will be copied forward out-of-the-box. This behavior also applies to Representables being checked out to a new container. (You can adjust this out-of-the-box behavior using the Copy Forward properties described later in this section.)

For example, where a Representation has been created from pre-converted viewables on the local file system and associated to a Windchill Part that has no describing CAD Document, when that Windchill Part iterates, all of the Representations will copy across to the new iteration, as they cannot be derived from anything.

For a Representation that is associated directly to a CAD Document (because it has no described Windchill Part) or to a Windchill Part with an active link to a describing CAD Document, where the Representation was derived from the CAD Document content: when the CAD Document iterates, the Representation should not be copied forward. This is because it should have a new Representation created from the updated content and metadata of the new CAD Document iteration. This behavior is referred to as "Restricted Mode" Copy Forward, and is enforced at the current releases, out-of-the-box.

## Copying Markups Forward

Markups (ProductView Annotations, Markups, and Groups) can optionally be configured to copy forward when their parent Representation is copied forward or a new Representation of the same name is created on a new iteration of the Representable. The decision to copy Markups forward is based on the allowCopyForward flag, which is set on each individual Markup object and manually changed using the Annotations and Markups list. By default, the value of this flag is false, but if it is set to true, Markups associated directly to a Windchill Document (no Representation for natively viewed content) will be copied forward to the new iteration.

## Copy Forward Properties

The Copy Forward properties in `wvs.properties` are described next, grouped into the following categories:

- Enabling the Copy Forward functionality
- Configuring the Representation Copy Forward functionality
- Configuring the Markup Copy Forward functionality

### Enabling the Copy Forward Functionality

<code>publish.copyrepresentationsforward=true</code> <code>publish.copymarkupsforward=true</code>	Allow Representations to be copied forward when parts iterate, as long as it does not compromise the validity of data published from EPM data. When Representations are copied forward, Markups and Annotations can be copied forward as well. If the copied forward Representation has Markups, it will not be replaced during publishing.
<code>publish.documents.copymarkupsforward=true</code> <code>publish.documents.copyrepresentationsforward=true</code>	Set copy forward of Representations and Markups specifically for WTDocuments. All Representations of WTDocuments are candidates for copy forward, even if they are published from the document content files, as publishing only occurs when document files are upload. If markups on the WTDocuments Representations are copied forward, the Representations will not be replaced by publishing.

## Configuring the Representation Copy Forward Functionality

<code>publish.copyrepresentationsforward.restrict=true</code>	Restricted Representation copy forward mode will not copy a published Representation forward when a WTPart iterates, and the old and new iterations both point to the same EPMDocument.
<code>publish.documents.copyrepresentationsforward.restrict=true</code>	Restricted Representation copy forward mode will not copy a published Representation forward when a WTDocument iterates, and the old and new iterations both point to the same WTDocument.
<code>publish.copyforwardallrepresentationsoncontainerchange=true</code>	When a new Representable iteration is created in a different container (such as during a sandbox checkout), this setting determines whether all Representations should be copied forward, or only those that would normally be copied forward.
<code>publish.copyforwardallrepresentationsoncopy=true</code> <code>publish.copyforwardallrepresentationsonrevise=true</code>	When a Representable is copied/revise, determines whether to copy forward all the Representations, or only those that would normally be copied forward.

## Configuring the Markup Copy Forward Functionality

<code>publish.copymarkupsrestricttosametype=false</code>	Determines whether the Markup copy should be restricted to copying to same types (from a Representation to a Representation, or a Viewable to a Viewable) only.
<code>edrload.copymarkupsfrompreviousiteration=</code> <code>edrload.copymarkupsfrompreviousversion=</code> <code>annotation,markup,group,pair_group,sequence</code>	<p>Determines whether to copy markups from a previous iteration when a new Representation is created. If the previous iteration is in the same version, the <code>copymarkupsfrompreviousiteration</code> list is used, to decide the list of markup types to copy, e.g. <code>annotation</code>, <code>Markup</code>, <code>group</code>, <code>pair_group</code>, or <code>sequence</code>.</p> <p>If the previous iteration is in a different version, the <code>edrload.copymarkupsfrompreviousversion</code> list is used, to determine the list of markup types to copy.</p>
<code>edrload.copymarkupsmatchdefaultfirst=false</code>	When copying Markups from a previous iteration, the Representation to copy from is first matched based on the name and then on the default status. Setting this property to true will match on default status in preference to name.
<code>markup.allowcopyforward.default=false</code>	When a new Markup is created, determines whether its <code>allowCopyForward</code> flag should be true or false. If true, the Markup can be copied forward, depending on other settings.
<code>publish.viewable.copymarkupsforward=true</code>	Allows Markups associated directly to a Viewable (e.g., <code>WTPart</code> or <code>WTDocument</code> ) to be copied forward when the Viewable iterates.

## Windchill Visualization Service Properties

Windchill uses standard Java property files to dynamically configure many optional or site-dependent settings. The primary property file, *wt.properties*, is located in the Windchill codebase directory, where it is available for downloading into clients. It contains properties that affect both client and server Java classes.

You can edit these files by using the System Configurator application, which allows you to add properties and values, delete properties, and save your changes to the properties files, for implementation when you restart the Windchill system.



Windchill Visualization Service (WVS) uses the properties described in the following table. They are set in the wvs.properties file.

Windchill Visualization Service property	Description
cadagent.inifile	<p><b>Default Value:</b> \$(wt.home)\\codebase\\agent.ini</p> <p><b>Synopsis:</b> Configuration file for CADAgent.</p> <p><b>Description:</b> Specifies the configuration file used by the CADAgent. This file configures the CAD Workers that are available for use by the WVS Publisher.</p>
cadagent.logs	<p><b>Default Value:</b> \$(wt.logs.dir)\\cadagent</p> <p><b>Synopsis:</b> Directory for CADAgent log files.</p> <p><b>Description:</b> Specifies the directory where CADAgent log files are written.</p>
cadagent.pvfiletypes	<p><b>Default Value:</b> OL ED PLT DXF HPGL PGL TXT AST CCZ CC GIF JPG PDF PVT GRP EMK ETB PVA CGM TGA</p> <p><b>Synopsis:</b> File types that the CadAgent will retrieve.</p> <p><b>Description:</b> Add to this space-delimited list, any file extensions that the CAD workers may create which need to be stored in Windchill. This list is not case sensitive.</p>
cadagent.filetypes.XXX	<p><b>Default Value:</b> NONE</p> <p><b>Synopsis:</b> File types that the CadAgent will retrieve for a specific worker type XXX.</p> <p><b>Description:</b> Add to this space-delimited list, any file extensions that the specific worker type XXX may create which need to be stored in Windchill. This list is not case sensitive. For example, the types for a THUMBNAIL worker are defined as JPG PVT EXTENTS GIF.</p>
cadagent.monitor.workerverboseargs	<p><b>Default Value:</b> @debug_options.txt</p> <p><b>Synopsis:</b> Arguments passed to via the worker monitor to the worker when started in debug mode.</p> <p><b>Description:</b> Arguments passed to via the worker monitor to the worker when started in debug mode from the CADAgent admin page.</p>

Windchill Visualization Service property	Description
cadagent.testfiledirectory	<p><b>Default Value:</b> \$(wt.home)\\loadFiles\\wvs\\testfiles.</p> <p><b>Synopsis:</b> The directory where the test files are stored.</p> <p><b>Description:</b> The directory where the test files are stored when loaded from the Visualization - Windchill Support CD. The test files are used by the CADAgent admin test worker function.</p>
collaboration.enabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Enables the Collaboration option from the WVS portal.</p> <p><b>Description:</b> Enables the Collaboration option from the WVS portal. This allows ProductView collaboration to be used in a Windchill context. The property is set when the collaboration agent is installed from the Visualization - Windchill Support CD.</p>
collaboration.server	<p><b>Default Value:</b> \$(wt.home)\\bin\\pview_collaboration.exe -p 5620 -m 5660 -i 600</p> <p><b>Synopsis:</b> Collaboration server start command.</p> <p><b>Description:</b> Specifies the command for starting the collaboration server. Typical options are; -p start_port_number, -m max_port_number, -i idle_timeout_in_seconds. The collaboration server executable referred to here should be placed in the correct location. The property is set when the collaboration agent is installed from the Visualization - Windchill Support CD. For more information on collaboration see ProductView documentation.</p>
collaboration.tempdir	<p><b>Default Value:</b> \$(wt.temp)\\collaboration</p> <p><b>Synopsis:</b> Directory used for Temporary files for collaboration.</p> <p><b>Description:</b> Specifies the directory used for Temporary files for collaboration.</p>
collaboration.verbose	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enable verbose output for services related to ProductView collaboration</p> <p><b>Description:</b> Enable verbose output for services related to ProductView collaboration.</p>

Windchill Visualization Service property	Description
distributedcadagent.enabled	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Set to true to enable the configuration of a Distributed CadAgent.</p> <p><b>Description:</b> Specifies if the CAD Agent Monitor wizard will present options and screens allowing a Distributed CadAgent to be configured. This only may be relevant when working with data populated via the Pro/INTRALINK gateway.</p>
edrload.copymarkupsfrompreviousiteration	<p><b>Default Value:</b> annotation,markup,group,pair_group,sequence</p> <p><b>Synopsis:</b> Copy markups from previous iteration when a new representation is created.</p> <p><b>Description:</b> If previous iteration is in the same version, specifies to copy markups from previous iteration when a new representation is created</p>
edrload.copymarkupsfrompreviousversion	<p><b>Default Value:</b> annotation,markup,group,pair_group,sequence</p> <p><b>Synopsis:</b> Copy markups from previous iteration when a new representation is created.</p> <p><b>Description:</b> If previous iteration is in a different version, specifies to copy markups from previous iteration when a new representation is created.</p>
edrload.copymarkupsmatchdefaultfirst	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Setting this property to true matches the default status in preference to name.</p> <p><b>Description:</b> When copying markups from previous iteration, the representation to copy from is first matched on the name and then on the default status. Setting this property to true matches the default status in preference to name.</p>
edrload.copyreferencedanddescribingtorep	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Copy content from referenced and describing documents to the representation.</p> <p><b>Description:</b> Specifies whether to copy content from referenced and describing documents to the representation or reference the content from the document.</p>

Windchill Visualization Service property	Description
edrload.copytransform	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Copy transform information from EPM./CAD System to Part structure.</p> <p><b>Description:</b> Copies transform information from EPM./CAD System to Part structure. For CAD systems whose structure is file driven, this will cause publish.matchcadnames to be treated as true.</p>
edrload.directory	<p><b>Default Value:</b> \$(wt.temp)\\wcinput</p> <p><b>Synopsis:</b> Directory for ticket files.</p> <p><b>Description:</b> Specifies the directory polled by loader for ticket files that are used to load preconverted visualization data into Windchill.</p>
edrload.docpropertygroup	<p><b>Default Value:</b> WindchillDocument</p> <p><b>Synopsis:</b> ProductView property group for WTDocument properties.</p> <p><b>Description:</b> Specifies the ProductView property group for WTDocument properties. A change to this value will require the corresponding change in the ProductView installation.</p>
edrload.edzenabled	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enables creation of an EDZ file when visualization data is stored.</p> <p><b>Description:</b> Enables the creation of an EDZ file when visualization data is stored. This includes data stored by the WVS publisher. Based on a user preference, a user will view an EDZ (if available) or an ED file. An EDZ file will download to the client all file data from a Representation initially, while with an ED file, individual files are downloaded on demand.</p>
edrload.epmpropertygroup	<p><b>Default Value:</b> WindchillEPM</p> <p><b>Synopsis:</b> ProductView property group for EPMDocument properties.</p> <p><b>Description:</b> Specifies the ProductView property group for EPMDocument properties. A change to this value will require the corresponding change in the ProductView installation.</p>

Windchill Visualization Service property	Description
edrload.includedescribing	<b>Default Value:</b> False <b>Synopsis:</b> Includes Describing WTDocuments in ed file. <b>Description:</b> Includes Describing WTDocuments in ed file for WTPart structure traversal. Significantly increases the structure traversal time.
edrload.includeepmpropertiesinpartstructure	<b>Default Value:</b> False <b>Synopsis:</b> Flag to specify if EPMDocument properties/property page link is to be included in part structure viewing/representations. <b>Description:</b> Flag to specify if EPMDocument properties/property page link is to be included in part structure viewing/representations.
edrload.includepartmastersdefault	<b>Default Value:</b> True <b>Synopsis:</b> Specifies whether to include part masters in the part structure. <b>Description:</b> Flag to specify if part masters are to be included in the part structure.
edrload.includeproperties	<b>Default Value:</b> True <b>Synopsis:</b> Includes Windchill properties in ed file. <b>Description:</b> Includes Windchill properties in ed file.
edrload.includepropertypagelink	<b>Default Value:</b> True <b>Synopsis:</b> Link to properties pages is to be added. <b>Description:</b> Flag specifies if link to properties pages is to be added.
edrload.includereferenced	<b>Default Value:</b> False <b>Synopsis:</b> Includes Referenced WTDocuments in ed file. <b>Description:</b> Includes Referenced WTDocuments in ed file for WTPart structure traversal. Will significantly increase the structure traversal time.
edrload.new.encoding	<b>Default Value:</b> <b>Synopsis:</b> Character set to use for new ED files. <b>Description:</b> Specifies the character set to use for new ED files (for example, a representation of a part structure). If not specified (default), the system default encoding is used.

Windchill Visualization Service property	Description
edrload.overwritetransform	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Overwrites transform information on Part structure.</p> <p><b>Description:</b> Overwrites transform information on Part structure, when copying data from EPM/CAD System.</p>
edrload.partpropertygroup	<p><b>Default Value:</b> WindchillPart</p> <p><b>Synopsis:</b> ProductView property group for WTPart properties.</p> <p><b>Description:</b> Specifies the ProductView property group for WTPart properties. A change to this value will require the corresponding change in the ProductView installation.</p>
edrload.propertyexplicitlist	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Windchill properties that should be retrieved explicitly and added to the ProductView representation.</p> <p><b>Description:</b> Windchill properties that should be retrieved explicitly and added to the ProductView representation.</p>
edrload.propertyskiplist	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Windchill properties that should not be included in ProductView representations.</p> <p><b>Description:</b> Windchill properties that should not be included in ProductView representations.</p>
edrload.read.encoding	<p><b>Default Value:</b> \$(wvs.edfileencoding)</p> <p><b>Synopsis:</b> Default character set to use for reading ED or ETB files.</p> <p><b>Description:</b> Specifies the default character set to use for reading ED or ETB files. The default is to use the default charset of the server. You can also specify the charset in ticket file to override this value. If the ED file specifies a value, that value is always used.</p>
edrload.verbose	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enables debug mode for loader.</p> <p><b>Description:</b> Enables debug mode for loader.</p>

Windchill Visualization Service property	Description
edrload.write.encoding	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Character set to use for storing ED files in database.</p> <p><b>Description:</b> Specifies the character set to use for storing ED files in database. If not specified (default), the existing encoding or the system default is used.</p>
markup.allowcopyforward.default	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> When a new markup is created, should it be set to copy forward or not.</p> <p><b>Description:</b> When a new markup is created, should it be set to copy forward or not. The copy forward setting for individual markups can be set from the Annotations and Groups listing.</p>
markup.tempdir	<p><b>Default Value:</b> \$(wt.temp)</p> <p><b>Synopsis:</b> Temporary directory used when creating markups.</p> <p><b>Description:</b> Specifies the temporary directory used when creating markups.</p>
markup.verbose	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enables debug information when creating markups.</p> <p><b>Description:</b> Enables debug information when creating markups.</p>
productview.batchprintoptions	<p><b>Default Value:</b> batchprint='true'</p> <p><b>Synopsis:</b> Options to send to ProductView at startup when batch printing.</p> <p><b>Description:</b> Options to send to ProductView at startup when batch printing.</p>
productview.collectionconfigbasedonfirstobject	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> When viewing objects from the visualization collection, the ProductView configuration selected is based on the container of the first object.</p> <p><b>Description:</b> When viewing objects from the visualization collection, the ProductView configuration selected is based on the container of the first object.</p>

Windchill Visualization Service property	Description
productview.collectionconfigbasedonorganization	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> When viewing objects from the visualization collection, the ProductView configuration selected is based on the organization of the user.</p> <p><b>Description:</b> When viewing objects from the visualization collection, the ProductView configuration selected is based on the organization of the user.</p>
productview.configfromserver	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Get ProductView configuration information from server.</p> <p><b>Description:</b> Get ProductView configuration information from server (for example, this retrieves watermark information from the WVSTemplate for the container of the object being viewed).</p>
productview.modifyconfigtype	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> The type for the WVSTemplate to be used when a user has modify access to the object being viewed.</p> <p><b>Description:</b> When ProductView receives configuration files from the server, users who have modify access to the representable can use a different WVSTemplate object. The type is specified when the WVSTemplate is created and that string should match the value of this property. If no value is specified, all users, irrespective of access rights, use the same configuration, based on the container.</p>
productview.options	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Options to send to ProductView at startup.</p> <p><b>Description:</b> Options to send to ProductView at startup.</p>
productview.redirectoptions	<p><b>Default Value:</b> redirecturl='unload.jsp'</p> <p><b>Synopsis:</b> Web page that ProductView will redirect to upon exit.</p> <p><b>Description:</b> Specifies the Web page that ProductView will redirect to upon exit. The default page simply closes the small web browser window that hosts the ProductView plugin.</p>



Windchill Visualization Service property	Description
publish.alwaysusespecifiedobject	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Publish specified object event if the specified config spec does not select it.</p> <p><b>Description:</b> When the object specified to be published would not be selected with the specified config spec, the specified object or the one selected by the config spec can be used for the publish job. For example, publishing an old iteration of an EPMDocument when using a Latest config spec.</p>
publish.cadconvert."xxx"	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Application Type to Java class lookup.</p> <p><b>Description:</b> Specifies a required entry that relates the Application Type to a Java class that will handle the CAD system specific publishing for any CAD system supported by the WVS Publisher.</p>
publish.cadtimeout.assembly	<p><b>Default Value:</b> 3600</p> <p><b>Synopsis:</b> Timeout for conversion of CAD assemblies.</p> <p><b>Description:</b> Specifies the timeout for CADAgent conversion of CAD assemblies in seconds.</p>
publish.cadtimeout.component	<p><b>Default Value:</b> 600</p> <p><b>Synopsis:</b> Timeout for conversion of CAD components.</p> <p><b>Description:</b> Specifies a timeout for CADAgent conversion of CAD components in seconds.</p>
publish.cadtimeout.drawings	<p><b>Default Value:</b> 600</p> <p><b>Synopsis:</b> Timeout for conversion of CAD drawings.</p> <p><b>Description:</b> Specifies the timeout for CADAgent conversion of CAD drawings in seconds.</p>

Windchill Visualization Service property	Description
publish.configspec.default.useasstoredifavailable	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Use as stored config spec when publishing EPMDocuments</p> <p><b>Description:</b> When publishing an EPMDocument structure and no config spec has been specified, or the Create Representation wizard specifies default (which is the default option), the config spec used depends on the EPMDocument. Setting this option to true causes the as stored config spec to be use when default is specified.</p>
publish.copyforwardallrepresentationsoncontainerchange	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> All representations can be copied forward when a new representable iteration is created in a different container (for example, in an integral checkout).</p> <p><b>Description:</b> Indicates if all representations should be copied forward when a new representable iteration is created in a different container (for example, in an integral checkout).</p>
publish.copyforwardallrepresentationsoncopy	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> When a representable is copied, copy forward all the representations or only those that would normally be copied forward.</p> <p><b>Description:</b> When a representable is copied, copy forward all the representations or only those that would normally be copied forward.</p>
publish.copyforwardallrepresentationsonrevise	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> When a representable is revised, copy forward all the representations or only those that would normally be copied forward.</p> <p><b>Description:</b> When a representable is revised, copy forward all the representations or only those that would normally be copied forward.</p>
publish.copymarkupsforward	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Allows the copy forward of Annotation and Groups when enabled.</p> <p><b>Description:</b> Applies in the case where a Representation is copied forward when a Part iterates. If this is enabled, the associated Annotation and Groups will also be copied.</p>

Windchill Visualization Service property	Description
publish.copymarkupsrestricttosametype	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Allow the copy markups to be restricted to copying from a representation to a representation and a viewable to a viewable only.</p> <p><b>Description:</b> When copying markups, allows the copy to be restricted to copying from a representation to a representation and a viewable to a viewable only.</p>
publish.copyrepresentationsforward	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> When enabled, allows Representations to be copied forward (to the next iteration) when Parts iterate.</p> <p><b>Description:</b> Specifies if the copy forward of Representations takes place. This will take place if set to true and the copy does not compromise the validity of data published from EPM.</p>
publish.copyrepresentationsforward.restrict	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Restrict representation copy forward mode.</p> <p><b>Description:</b> Restricted representation copy forward mode does not copy a published representation forward when a WTPart iterates and old and new iteration point to the same EPMDocument.</p>
publish.createpublishaudit	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Create publish audit information for each publish job.</p> <p><b>Description:</b> Create publish audit information for each publish job.</p>
publish.createrepresentationcadtypes	<p><b>Default Value:</b> PROE CADDSS5 CATIA PRODESKTOP UG SOLIDWORKS CATIAV5 OTHER</p> <p><b>Synopsis:</b> List of CAD types that can be loaded or converted from the <b>Create Representation</b> window.</p> <p><b>Description:</b> Lists CAD types that can be loaded or converted from the <b>Create Representation</b> window on a WTPart without an EPMDocument. This is a space-separated list of keys from EPMAuthoringAppTypeRB.rbinfo. If the type OTHER is included, the file to be processed is treated as if it is on a document: the document worker mapping determines the conversion that occurs.</p>

Windchill Visualization Service property	Description
publish.deletepreconvertededz	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Delete temporary EDZ file.</p> <p><b>Description:</b> When publishing an EPMDocument that has an EDZ file of preconverted data (client side-generated viewables), deletes the EDZ file when the EPMDocument is published.</p>
publish.documents.copymarkupsforward	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Set copy forward of markups for WTDocuments.</p> <p><b>Description:</b> Sets copy forward of markups for WTDocuments. Individual markups are copied forward only if that markup has its copy forward flag set.</p>
publish.documents.copyrepresentationsforward	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Set copy forward of representations for WTDocuments.</p> <p><b>Description:</b> Sets copy forward of representations for WTDocuments. All representations of WTDocuments are candidates for copy forward, even if they are published from the document content files. Publishing occurs only when document files are uploaded. If markups on the WTDocuments representations are copied forward, the representations are not replaced by publishing.</p>
publish.forcerepublish	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enables forcing of republishing existing Representations.</p> <p><b>Description:</b> Forces the republishing of existing Representations that are already valid, (for example, when CAD Worker settings have been changed and it is required to reconvert data.)</p>
publish.markonlydirectusesrepresentationsoutofdate	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Restrict marking out of date to only direct uses EPMDocuments.</p> <p><b>Description:</b> Restricts marking out of date to only direct uses EPMDocuments.</p>

Windchill Visualization Service property	Description
publish.markoutofdatefiltermethod	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Method to filter which representations are marked out of date.</p> <p><b>Description:</b> By default, representations that are derived with a Latest Config Spec are candidates for being marked out of date. You can use a custom method to provide different criteria, for example, life cycle state. The method is defined by the following property in the form classname/methodname. The method should have the following signature:  public static Boolean  methodname(EPMDocument epmdoc,  Representation rep). A return of Boolean TRUE indicates that the passed in representation of the passed in EPMDocument is a candidate for being marked out of date. A return of Boolean FALSE indicates that is should not be a candidate.</p>
publish.markoutofdateoncheckin	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Marks parent's representations out of date on checkin.</p> <p><b>Description:</b> When set to true, representations of parent objects created with a latest configuration spec will be marked out of date when the child is checked in.</p>
publish.markoutofdateonpdmcheckin	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Marks parent's representations out of date on PDM checkin.</p> <p><b>Description:</b> When set to true, representations of parent objects created with a latest configuration spec will be marked out of date when a PDM checkin is complete for a child.</p>
publish.markoutofdateonpublishcomplete	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Marks parent's representations out of date on publish.</p> <p><b>Description:</b> This is the default setting. To specify any other setting for marking out of date, such as on checkin or revise, you must set this value to False.</p>

Windchill Visualization Service property	Description
publish.markoutofdateonrevise	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Marks parent's representations out of date on revision.</p> <p><b>Description:</b> When set to true, representations of parent objects created with a latest configuration spec will be marked out of date when the child is revised.</p>
publish.markoutofdaterepublishmethod	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Method to filter which representations that are marked out of date should be sent for republishing.</p> <p><b>Description:</b> When a representation is marked out of date for the first time, it can be republished. This causes many extra publishing jobs. You can define a method to use to filter which EPMDocuments should have the representation automatically republished, for only released data could be selected. The method is defined by the following property in the form classname/methodname. The method should have the following signature:  public static Boolean  methodname(EPMDocument epmdoc,  Representation rep).</p> <p>A return of Boolean TRUE indicates that the passed in representation of the passed in EPMDocument is sent for republishing; a return of Boolean FALSE indicates that it should not.</p> <p>For example, the markRepublishAll method sends all representations for republishing.  publish.markoutofdaterepublishmethod=com.ptc.wvs.server.publish.PublishHelper/markRepublishAll</p> <p>When representation are not sent for republishing, the schedule job republishOutOfDate, or a customer schedule job, can be used to republish representations marked as out of data at a time best suited to system resource usage.</p>

Windchill Visualization Service property	Description
publish.markpositioningassemblyrepresentationsoutofdate	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Provide a warning to the user when viewing representations marked as out of date.</p> <p><b>Description:</b> Representations can be marked as out of date which provides a warning to the user when viewing that representation. Generally, positioning assemblies do not need to be marked out of date, as typically they are using a Latest config spec to select the latest children at the time of viewing.</p>
publish.markreferencesrepresentationsoutofdate	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Provide a warning to the user when viewing representations marked as out of date.</p> <p><b>Description:</b> Representations can be marked as out of date which provides a warning to the user when viewing that representation. When an EPMDocument is published, it can find referencing EPMDocuments and mark their representations that are older than the EPMDocument being published and use a Latest Config Spec as being potentially out of date.</p>
publish.markusesrepresentationsoutofdate	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Provide a warning to the user when viewing representations marked as out of date.</p> <p><b>Description:</b> Provides a warning to the user when viewing representations marked as out of date. When an EPMDocument is published, it can find using EPMDocuments and mark their representations (that are older than the EPMDocument being published and use a Latest Config Spec) as being potentially out of date. If required, the marking of uses of representations be limited to representation that directly use the object being published.</p>
publish.matchcadnames	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Allows matching of data from CAD system with Windchill when enabled.</p> <p><b>Description:</b> Enables matching of data from CAD system with Windchill for "file based" systems. Allows Windchill property pages to be referenced from ProductView and population of Windchill properties in ProductView. Also implied when transform information is being populated from EPM/CAD System to Part structure.</p>

Windchill Visualization Service property	Description
publish.monitor.displaylimit	<p><b>Default Value:</b> 250</p> <p><b>Synopsis:</b> Display limit for number of jobs in the publish monitor.</p> <p><b>Description:</b> Specifies the display limit for jobs in the publish monitor. The value of the limit is the number of ready and the number of executing/completed jobs (for example, a limit of 250 displays up to 250 ready jobs and 250 executing/completed jobs).</p>
publish.positioningassembly.filename publish.positioningassembly.filtermethod publish.positioningassembly.ibaname	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Used to identify assemblies to be published as "positioning assemblies."</p> <p><b>Description:</b> Used to identify assemblies to be published as "positioning assemblies." See wvs.properties.xconf for details.</p>
publish.positioningassembly.publishchildren	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> When publishing a positioning assembly, publish children.</p> <p><b>Description:</b> When publishing a positioning assembly, publish children if the child does not have a representation for the positioning assembly to use.</p>
publish.publishqueue.priorities."type"."source"	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Defines the priority of queue that a particular type of publish job, from a particular source, will use.</p> <p><b>Description:</b> Defines the priority of queue that a particular type of publish job, from a particular source, will use. See wvs.properties.xconf for details.</p>
publish.publishqueuehelper.verbose	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enables debug information for publisher queues.</p> <p><b>Description:</b> Enables debug information for publisher queues and schedule queue jobs.</p>
publish.publishqueuepollinterval	<p><b>Default Value:</b> 5</p> <p><b>Synopsis:</b> Polling interval for PublisherQueue to look for free queues.</p> <p><b>Description:</b> Specifies the polling interval (in seconds) for PublisherQueue to look for free queues.</p>



Windchill Visualization Service property	Description
publish.republishondocumentchange	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Republish or update representation when metadata on a WTDocument is changed and the WTDocument does not iterate.</p> <p><b>Description:</b> When the metadata on a WTDocument is changed and the WTDocument does not iterate (for example, a life cycle state is changed) the associated representations are updated. In certain cases (for example, when an OFFICE file on the WTDocument has an association to that metadata), it may be desirable to republish the representation rather than just updating it.</p>
publish.republishonepmdocumentchange	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Republish or update representation when metadata on an EPMDocument is changed and the EPMDocument does not iterate.</p> <p><b>Description:</b> When the metadata on an EPMDocument is changed and the EPMDocument does not iterate (for example, a life cycle state is changed) the associated representations are updated. In certain cases (for example, when the CAD file on the EPMDocument has an association to that metadata), it may be desirable to republish the representation rather than just updating it.</p>
publish.retrieveallfiles	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Retrieve all dependent files.</p> <p><b>Description:</b> If true, retrieves all files; if false, retrieves only those files whose links are marked as required.</p>
publish.service.addprohibitiononpublishfailure."type". "source"	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Automatically add a publishing prohibition when a type of publish job from a specific source fails.</p> <p><b>Description:</b> Automatically add a publishing prohibition when a type of publish job from a specific source fails. See wvs.properties.xconf for details.</p>

Windchill Visualization Service property	Description
publish.service.documents.checkin.enabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Send WTDdocuments for publishing on checkin.</p> <p><b>Description:</b> Specifies if WTDdocuments are to be sent for publishing on checkin. Only publishable files (those with an associated worker) are considered. All publishable files on the checked in document are sent for publishing. In addition, any files uploaded to a non-checkout document are published (for example, a new document or one in the user's personal cabinet). This option is mutually exclusive with publish.service.document.upload.enabled.</p>
publish.service.documents.options	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Pass options to document publishing.</p> <p><b>Description:</b> Pass options for document publishing. Comma-separated list of name=value pairs. For example:  encodefilename={true, false} - encode non-ascii filename before sending to worker  encoding= - specify the character encoding for the ed file, for example, SJIS or UTF-8</p>
publish.service.documents.upload.enabled	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Send WTDdocuments for publishing on each file upload.</p> <p><b>Description:</b> Specifies if WTDdocuments are to be sent for publishing on each file upload. Only publishable files (those with an associated worker) are considered. This option is mutually exclusive with publish.service.document.checkin.enabled.</p>
publish.service.documents.upload.markoutofdate	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Mark document representation, published from a file on the document, as out of date when a new version of the file is uploaded.</p> <p><b>Description:</b> Marks document representation, published from a file on the document, as out of date when a new version of the file is uploaded. The file is not sent for publishing (for example, for an update when publishing on checkin or if publish on checkin and upload are disabled).</p>

Windchill Visualization Service property	Description
publish.service.enabled	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Provides listener for publishing CAD data on checkin when enabled.</p> <p><b>Description:</b> Enables WVS listener that listens for EPM CheckIn complete events. Checked in objects will be sent for publishing.</p> <p><b>Note:</b> This property is not used in Windchill 7.0 and should be left with the default setting of false. See <code>publish.service.readytopublish.enabled</code>.</p>
publish.service.filterdocumentpublishmethod	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Supply method to provide custom filtering of the WTDocument content to be published.</p> <p><b>Description:</b> Specifies a method to provide custom filtering of the WTDocument content to be published as a result of file upload or checkin (provided the content has a worker associated with it). If not method is supplied, all WTDocument content that meets the above criteria is sent for publishing. The property value is specified in the form <code>classname/methodname</code>, with the following signature:</p> <pre>publish static Boolean methodname(WTDocument doc, ContentItem ci)</pre> <p>A return of Boolean TRUE indicates the WTDocument should be published, Boolean FALSE indicates that it should not be published.</p>

Windchill Visualization Service property	Description
publish.service.filterepmdocumentpublishmethod	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Custom filtering of EPMDocuments published as a result of CheckIn Complete or Ready To Publish events.</p> <p><b>Description:</b> You can supply a method to provide custom filtering of the EPMDocuments that are published as a result of a CheckIn Complete or Ready To Publish event. If no method is supplied, all EPMDocuments in the event are sent for publishing. The property value is specified in the form classname/methodname, with the following signature:  public static Boolean  methodname(EPMDocument epmdoc)  a return of Boolean TRUE indicates the EPMDocument is published, Boolean FALSE, the EPMDocument is not published.</p>
publish.service.filterpublishmethod	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Supply a method to provide custom filtering of objects that can be published.</p> <p><b>Description:</b> Allows you to supply a method to provide custom filtering of objects that can be published. This method checks all publishing, even from preconverted. If the publish is to convert data stored in Windchill, the flag publishFromDB is true; if the publish is for local data or data from the clipboard, publishFromDB is false. The property value is specified in the form classname/methodname, with the following signature:  public static Boolean methodname(Persistable p, Boolean publishFromDB). A return of Boolean true indicates the object can be published; Boolean false indicates it cannot.</p>
publish.service.ignoredefaultepmevents	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Restrict EPMDocument publishing from CheckIn Complete or Ready To Publish events.</p> <p><b>Description:</b> Restricts EPMDocument publishing from CheckIn Complete or Ready To Publish events to cases where the client initiating the event, for example the work group manager, has specified options for the creation of the representation. If this option is set, default CheckIn Complete or Ready To Publish events are ignored.</p>

Windchill Visualization Service property	Description
publish.service.ignoreeventconfigspec	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Ignore the config spec specified in Ready To Publish events.</p> <p><b>Description:</b> Ignore the config spec specified in Ready to Publish events. WVS will then use either a latest or as stored config spec for the publishing.</p>
publish.service.onlypublishpreconvertededz	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Restrict EPMDocument publishing to EDZ files.</p> <p><b>Description:</b> When using EDZ client side viewables, restricts publishing of EPMDocuments to those that have EDZ files. This limits all publishing to be only from a temporary EDZ file (for example, client side viewables).</p>
publish.service.onlypublishpreconvertededzfromevents	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Restrict EPMDocument publishing to EDZ files, for the check in listener only.</p> <p><b>Description:</b> When using EDZ client side viewables, restricts publishing of EPMDocuments to those that have EDZ files. This limits publishing initiated by an event only).</p>
publish.service.readytopublish.enabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Listen for Ready To Publish event to initiate publishing of EPMDocuments.</p> <p><b>Description:</b> Listens for Ready To Publish event to initiate publishing of EPMDocuments. Work group managers should emit this event for EPMDocuments that require visualization.</p>
publish.service.verbose	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Enables debug mode for publish listener.</p> <p><b>Description:</b> Enables debug mode for publish listener.</p>
publish.tempdir	<p><b>Default Value:</b> \$(wt.temp)\pubtemp</p> <p><b>Synopsis:</b> Temporary directory for WVS publisher.</p> <p><b>Description:</b> Stores WVS publisher's temporary files in a directory.</p>

Windchill Visualization Service property	Description
publish.tempuploadir	<p><b>Default Value:</b> \$(publish.tempdir)</p> <p><b>Synopsis:</b> Temporary directory for uploaded files.</p> <p><b>Description:</b> Creates a temporary directory for uploading files when creating representations from local files. For a cluster environment, this directory must be a shared, common directory, accessible to all involved method servers.</p>
publish.viewable.copymarkupsforward	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Allow markups associated directly to a viewable (for example, WTPart or WTDocument) to be copied forward when the viewable iterates.</p> <p><b>Description:</b> Allows markups associated directly to a viewable (for example, WTPart or WTDocument) to be copied forward when the viewable iterates. An individual markup is copied only if its copy forward flag is set.</p>
schedulejobsN	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Internal name for a schedule job definition.</p> <p><b>Description:</b> Specifies the internal name for a schedule job definition. All other entries for this job definition will use this name as the prefix on the property name.</p>
"schedulejobname".description	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Display name for the schedule job definition.</p> <p><b>Description:</b> Specifies the display name for the schedule job definition.</p>
"schedulejobname".enableOnContainers	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Should this schedule job be available in the context of a specific container (for example, a Product)</p> <p><b>Description:</b> The queries in the schedule job should have been written appropriately in this option is set to be set to true.</p>
"schedulejobname".class	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Java class for the schedule job definition.</p> <p><b>Description:</b> Specifies the Java class for the schedule job definition.</p>

Windchill Visualization Service property	Description
"schedulejobname".method	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Java method for the schedule job definition.</p> <p><b>Description:</b> Specifies the Java method for the schedule job definition.</p>
scheduler.user	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> User name used by scheduler.</p> <p><b>Description:</b> Specifies the user name used by scheduler. If no user name is specified, schedule queue jobs will be executed as the user who submitted them.</p>
thumbnail.byworker.enabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Flag to enable thumbnail generation by a CAD worker.</p> <p><b>Description:</b> Flag to enable thumbnail generation by a CAD worker. The worker itself must be configured to generate thumbnails as well, or else this setting will have no effect.</p>
thumbnail.cadagenttimeout	<p><b>Default Value:</b> 3600</p> <p><b>Synopsis:</b> Timeout in seconds.</p> <p><b>Description:</b> Sets the timeout in seconds for thumbnail generation when using the cadagent and a remote thumbnail worker.</p>
thumbnail.enabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Flags to enable thumbnail image generation.</p> <p><b>Description:</b> Enables the generation of thumbnail images that are displayed on listings and property pages. The default thumbnail generation method uses Java3D, hence this must be installed on the Windchill Server machine.</p>
thumbnail.filelimit	<p><b>Default Value:</b> 800</p> <p><b>Synopsis:</b> Maximum number of files for thumbnail generation.</p> <p><b>Description:</b> Sets a limit on the size of an assembly (by the number of referenced files) that will have a thumbnail image generated. This can be used to control server resources.</p>

Windchill Visualization Service property	Description
thumbnail.generator.backcolor	<p><b>Default Value:</b> 183,183,183</p> <p><b>Synopsis:</b> Sets the image background color for thumbnail generator.</p> <p><b>Description:</b> Specifies the image background color to be created by the thumbnail generator. The three numbers represent red, green and blue in the range 0 to 255.</p>
thumbnail.generator.extents	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Enables extents generation for thumbnail generator.</p> <p><b>Description:</b> Specifies that the thumbnail generator should compute the bounding box for each of file that is processed. The bounding box is stored in the ed file, and is used by ProductView for proximity search and file loading optimization.</p>
thumbnail.generator.height	<p><b>Default Value:</b> 128</p> <p><b>Synopsis:</b> Sets the image height for thumbnail generator.</p> <p><b>Description:</b> Specifies the image height to be created by the thumbnail generator.</p>
thumbnail.generator.nativecmd	<p><b>Default Value:</b> set by Visualization - Windchill Support CD installer</p> <p><b>Synopsis:</b> Executes the native thumbnail generator.</p> <p><b>Description:</b> Specifies the command to execute the native thumbnail generator, which creates 2D thumbnail images and 3D thumbnail files. The value of this property is set by the Visualization - Windchill Support CD installer.</p>
thumbnail.generator.rx	<p><b>Default Value:</b> -60</p> <p><b>Synopsis:</b> Sets image X axis rotation for thumbnail generator.</p> <p><b>Description:</b> Specifies the image X axis rotation in degrees.</p>
thumbnail.generator.ry	<p><b>Default Value:</b> -25</p> <p><b>Synopsis:</b> Sets image Y axis rotation for thumbnail generator.</p> <p><b>Description:</b> Specifies the image Y axis rotation in degrees.</p>
thumbnail.generator.rz	<p><b>Default Value:</b> -10</p> <p><b>Synopsis:</b> Sets image Z axis rotation for thumbnail generator.</p> <p><b>Description:</b> Specifies the image Z axis rotation in degrees.</p>



Windchill Visualization Service property	Description
thumbnail.generator.width	<p><b>Default Value:</b> 192</p> <p><b>Synopsis:</b> Sets the image width for thumbnail generator.</p> <p><b>Description:</b> Specifies the image width to be created by the thumbnail generator.</p>
thumbnail.generator.zoomfactor	<p><b>Default Value:</b> 1.2</p> <p><b>Synopsis:</b> Sets image zoom factor for Java thumbnail generator. This is not used if the native thumbnail generator is installed.</p> <p><b>Description:</b> Specifies the image zoom factor, allowing the object to be made larger or smaller in the thumbnail image.</p>
thumbnail.usecadagent	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Set to true if thumbnail generation is performed on a remote system.</p> <p><b>Description:</b> Executes thumbnail generation via the CadAgent by configuring a worker type THUMBNAIL and using the GenericWorker. In most cases this is not required, but if the Windchill server is not capable of running the thumbnail generator, this does allow a remote machine to be used to execute the actual thumbnail generation process.</p> <p>This will be less efficient than executing directly.</p>
viewer.<DataFormatName>	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Maps Windchill file type to ProductView viewer type.</p> <p><b>Description:</b> Maps Windchill file type to ProductView viewer type. Windchill file type (the DataFormat name) specified here should have any spaces removed. Additional entries of this form can be added for any file types (for example, viewer.GIFImage=image).</p>

Windchill Visualization Service property	Description
viewer.<FileExtension>	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Maps file extension to ProductView viewer type.</p> <p><b>Description:</b> Maps file extension to ProductView viewer type. Note that the file extension includes the dot, and should be specified in upper case. There is an alternative technique for associating a file with a specific file viewer, using the DataFormat name. Additional entries of this form can be added for other file extensions (for example, viewer.HPGL=drawing).</p>
webpage.allowdeleterepresentation	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Allows representation deletion.</p> <p><b>Description:</b> Provides the ability for a user to delete a representation from the user interface. If true, all users can delete representations from the user interface; if false, no users can delete representations from the user interface. If admin, only administrators (system, project, or product administrators) can delete representations from the user interface.</p>
webpage.allowmakedefaultrepresentation	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Allows default representation selection.</p> <p><b>Description:</b> Provides the ability for a user to change the default representation from the user interface. If true, all users can change the default representation from the user interface; if false, no users can change the default representation from the user interface. If admin, only administrators (system, project, or product administrators) can change the default representation from the user interface.</p>
webpage.allowpublish.epmdocument webpage.allowpublish.wtpart webpage.allowpublish.wtdocument	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Allows publishing from the user interface.</p> <p><b>Description:</b> Provides the ability for a user to publish from the user interface, on a type basis. If true, all users can publish the specified type from the user interface; if false, no users can publish the type from the user interface. If admin, only administrators (system, project, or product administrators) can publish from the user interface.</p>

Windchill Visualization Service property	Description
webpage.autoload	<p><b>Default Value:</b> single</p> <p><b>Synopsis:</b> Enables autoloading of files in ProductView.</p> <p><b>Description:</b> Allows ProductView to autoload files. If value is "true" all files will be autoloading. If value is "single", ed files with a single associated file will be autoloading. If value is "false" no files will be autoloading.</p>
webpage.defaultquerytype	<p><b>Default Value:</b> 21</p> <p><b>Synopsis:</b> Sets the default query type for WVS portal, used only in Windchill Foundation &amp; PDM.</p> <p><b>Description:</b> Specifies default query type for the WVS portal page. The types supported are; 21 = WTParts, 22 = WTDocuments, 49 = EPMDocuments, 55=ProductInstances, 111=ProductConfigurations.</p>
webpage.displayproductthumbnail	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Displays thumbnail for ProductInstance and ProductConfiguration.</p> <p><b>Description:</b> For the view link for ProductInstance and ProductConfiguration, displays the thumbnail image. This gives only an approximate rendition.</p>
webpage.displayviewthumbnailfor2d	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Displays thumbnail viewer for 2D thumbnails.</p> <p><b>Description:</b> On the Representations listing page, displays the thumbnail viewer popup icon when there is only a 2D thumbnail (that is, no 3D thumbnail); otherwise, there must be a 3D thumbnail and the thumbnail view must be installed on the server.</p>
webpage.flagoutofdate	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Displays warning when viewing an out of date representation.</p> <p><b>Description:</b> Displays warning when viewing a representation that has be marked as out of date.</p>
webpage.longlistingdefault	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Set the long listing default for WVS portal.</p> <p><b>Description:</b> Specifies default for the Long List setting on the WVS portal.</p>

Windchill Visualization Service property	Description
webpage.markupenabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Enables the creation and storing of markups from ProductView.</p> <p><b>Description:</b> Enables the creation and storing of markups from ProductView.</p>
webpage.partstructurereview	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Enables display of the part structure view option.</p> <p><b>Description:</b> Allows part structures to be viewed from the WVS portal and the Product Structure web page, without the publishing of CAD data and the creation of a representation.</p>
webpage.pastemarkupswithrepresentation	<p>Default Value: True</p> <p>Synopsis: Adds (with add from collection) Annotations and Groups with a Representation.</p> <p>Description: Allows any associated Annotation and Groups to also be added, when a Representation is added from the collection.</p>
webpage.showedrdelete	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Sets the delete display for WVS portal.</p> <p><b>Description:</b> Specifies the display of a delete option on the WVS portal.</p>
webpage.showextendedclipboard	<p>Default Value: True</p> <p>Synopsis: Enables the add to collection and add from collection facilities to allow Representations and/or Markups to be copied.</p> <p>Description: Enables additional add to collection and add from collection facilities on the Representations, Annotation and Groups, and Visualization Collection pages.</p>
webpage.showpublishfordoc	<p>Default Value: True</p> <p>Synopsis: Display publish link for documents with publishable files.</p> <p>Description: For a document with publishable files (that is, worker mapping defined with a "worker.xxx=yyy" property) displays the publish link when there is no default representation. The default publish link publishes the first publishable file on the document.</p>

Windchill Visualization Service property	Description
webpage.showrepfordoc	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Display link to representation or markup page for documents.</p> <p><b>Description:</b> For a document with no representations and no markups and no displayed publish link, displays the link to either a markups page or a representations page.</p>
webpage.showrepforpart	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Displays representation link for a part.</p> <p><b>Description:</b> When a part displays the part structure link and has no markups, displays the link to the representation listing. If false, displays the link to the markups listing.</p>
webpage.showsavezip	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Displays the save representation action.</p> <p><b>Description:</b> Displays the save representation action in the representation listing. This allows you to save the representation as a local ZIP or JAR file or a link to view the representation to be saved (to include in an e-mail, for example).</p>
webpage.showstructureforpart	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Displays view default structure link for a part.</p> <p><b>Description:</b> When a part has no default representation and no associated EPMDocument, displays the view default structure link.</p>
webpage.showthumbnail	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Sets the thumbnail display default for WVS portal.</p> <p><b>Description:</b> Specifies default for the display of thumbnails on the WVS portal. Individual users can set their preference for thumbnail display from preference settings.</p>
webpage.verbose	<p><b>Default Value:</b> False</p> <p><b>Synopsis:</b> Provides user interface debug information.</p> <p><b>Description:</b> Provides user interface debug information.</p>

Windchill Visualization Service property	Description
worker.<DataFormatName>	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Maps Windchill file type to a worker type.</p> <p><b>Description:</b> Maps Windchill file type to a worker type. A worker of this type should have been configured in the agent.ini file (for example, worker.VRML=VRML).</p>
worker.<FileExtension>	<p><b>Default Value:</b> N/A</p> <p><b>Synopsis:</b> Maps file extension to a worker type.</p> <p><b>Description:</b> Maps file extension to a worker type. Note that the file extension includes the dot, and should be specified in upper case. There is an alternative technique for associating a file with a specific file viewer, using the DataFormat name. A worker of this type should have been configured in the agent.ini file (for example, worker..IGES=IGES).</p>
wvs.demo.data	<p><b>Default Value:</b> \$(wt.home)\\loadFiles\\wvs\\</p> <p><b>Synopsis:</b> Directory where WVS demo data is located.</p> <p><b>Description:</b> Specifies the location of the WVS demo data. This is referenced only in the WVS demo data ini files. A file delimiter is required at the end of the entry.</p>
wvs.edencoding	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Character encoding for communication to server.</p> <p><b>Description:</b> Specifies the character encoding used by the servlet engine.</p>
wvs.edfileencoding	<p><b>Default Value:</b></p> <p><b>Synopsis:</b> Character encoding for reading ed files.</p> <p><b>Description:</b> Specifies the default encoding to use for reading ED files. If none specified, then the system uses the default character encoding of the server. See also <b>edrload.encoding</b>.</p>

Windchill Visualization Service property	Description
wvs.enabled	<p><b>Default Value:</b> True</p> <p><b>Synopsis:</b> Enables Windchill Visualization Service.</p> <p><b>Description:</b> Enables Windchill Visualization Service. When set to false, users will not see thumbnails or be able to launch ProductView. Before visualization can be fully used, other parts of the system, such as CAD Workers, need to be installed and configured.</p>
wvs.etbencoding	<p><b>Default Value:</b> UTF-8</p> <p><b>Synopsis:</b> The character encoding used for the ProductView ETB file.</p> <p><b>Description:</b> The character encoding used for the ProductView ETB file.</p>





# 19

## Administering Audit Reports

Topic	Page
Overview .....	19-2
Enabling Auditing .....	19-2
Accessing Audit Reports .....	19-3
Example Audit Report.....	19-4

## Overview

By default, the auditing features provided by Windchill are disabled. This documentation provides the instructions to enable the auditing features and use audit reporting.

## Enabling Auditing

To enable the auditing features, use the following procedure:

1. Shut down servers/servlet engines.
2. Edit the wt.properties file to add the wt.audit.eventConfigFile property using the xconfmanager. From a windchill shell, execute the following command:

```
xconfmanager -s wt.audit.eventConfigFile=<Windchill>/codebase/registry/auditing/  
configAudit.xml -t <Windchill>/codebase/wt.properties -p
```

where <Windchill> is the file path to the installed Windchill location.  
Including the -p option propagates the change to wt.properties.

**Note:** As a best practice, use forward slashes ( / ) as the directory separator in file paths, as this is recognized on all operating systems. For further information on formatting property values in the xconfmanager, see [About the xconfmanager Utility](#) in the Administration Overview chapter.

3. Set the EventConfiguration enabled property to true on the <Windchill>/codebase/registry/auditing/configAudit.xml file:

```
EventConfiguration enabled="true"
```

4. Analyze your auditing rules.

The characteristics of what gets audited is described in the <Windchill>/codebase/registry/auditing/configAudit.xml file. This file is delivered with some predefined auditing rules. By default, auditing at the highest-level is turned off and everything else turned on. Therefore, when you set the property to true in step 3, everything gets audited. You can further edit this file to disable objects and events that you are not interested in

If you choose to add or change auditing rules, PTC recommends that you follow the maintenance best practices for modifying PTC files as described in the *Windchill Customizer's Guide*. Using the best practices, you create a backup of the original and apply your changes to a new file. If you decide to select a new name for this file, then you must edit the wt.audit.eventConfigFile property to reference the new name.

5. Restart servers and servlet engines.

# Accessing Audit Reports

**Note:** The Audit Reports link appears only if the feature has been enabled. See [Enabling Auditing](#) for information about enabling Audit Reports.

How you access Audit Reports is determined by your Windchill application:

- From Windchill Foundation & PDM, you can access Audit Reports by clicking the **Audit Reports** link that is on the **Business Administration** home page.
- From Windchill PDMLink and Windchill ProjectLink, you can access Audit Reports from the **Audit Reports** link directly below the **Site** and **Organization** tabs. The link from the **Site** tab provides the site administrator with unrestricted access to all reports. The link from the **Organization** tab provides access to only those reports that are in the organization context that is active.

For Windchill PDMLink and Windchill ProjectLink, there are 10 reports available. They include:

- Context Access
- Context Change
- Context-Access Change
- Object Access
- Object Change
- Object-Access Change
- Organization Access
- Team Change
- User Access
- User-Context Access

In Windchill Foundation & PDM, there are seven reports available. They include:

- Context Access
- Context Change
- Object Access
- Object Change
- Object-Access Change
- Team Change
- User Access

For information on how to use the Audit Report utility, see the online help.

## Example Audit Report

Assume you want to find out the accesses a user has made in the last month.

1. Click **User Access** on the **Audit Reports** page.
2. On the **User Access** page, click **Find User** to select a user.
3. Select a time period from the **Time Period** drop-down list. You can also select a custom time period by using the start and end calendar pickers.
4. Click **Generate Report**.

For Windchill PDMLink or Windchill ProjectLink, your browser should look something like this:

Windchill

Learn | Help | Publications | E-mail Page | Hot Links | Copy Page

Home | Product | Project | Change | Library | Organization | Site

Search:  Go

Details | Folders | Members | Creators | Administrators | Groups | Roles | Types | Reports | Templates | Audit Reports | Utilities

QA Recent Organizations:  Subscriber: Yes

Windchill QA Search within this organization:  Go

### User Access

\* User Name:  [Find User](#)

Time Period:  | Start Date:  MM/DD/YYYY | End Date:  MM/DD/YYYY

#### Report Results Table

User	Context	User Organization	Object	Object Type	Event Name	Event Date
wcadmin	ESI PRODUCT		Team -- ESI PRODUCT	Team	Create	Oct 21, 2003, 2:02 PM
wcadmin	ESI PRODUCT		Team -- ESI PRODUCT	Team	Modify User = ESICA3 Role = SUBMITTER User = ESICA3 Role = REVIEWER User = ESICA3 Role = ASSIGNEE Role = SUBMITTER Role = REVIEWER	Oct 21, 2003, 2:00 PM

For Windchill Foundation & PDM, your browser would look something like this:

The screenshot shows the 'User Access' report in a web browser. On the left is a navigation menu with links: \*Home, \*Personal Cabinet, \*Search, \*Worklist, \*Create, and Document. The main content area is titled 'User Access'. It contains a search section with a 'User Name' field set to 'wadmin' and a 'Find User' button. Below this is a 'Time Period' dropdown set to 'Last 7 Days', and 'Start Date' and 'End Date' fields both set to '10/15/2003' and '10/21/2003' respectively, with a date picker icon. At the bottom of this section are three buttons: 'Back to Reports', 'Generate Report', and 'Save to File'. Below the buttons is a 'Report Results Table' with a yellow header bar and a pagination control showing 16 items. The table has seven columns: User, Context, User Organization, Object, Object Type, Event Name, and Event Date. It lists ten audit events for the user 'wadmin'.

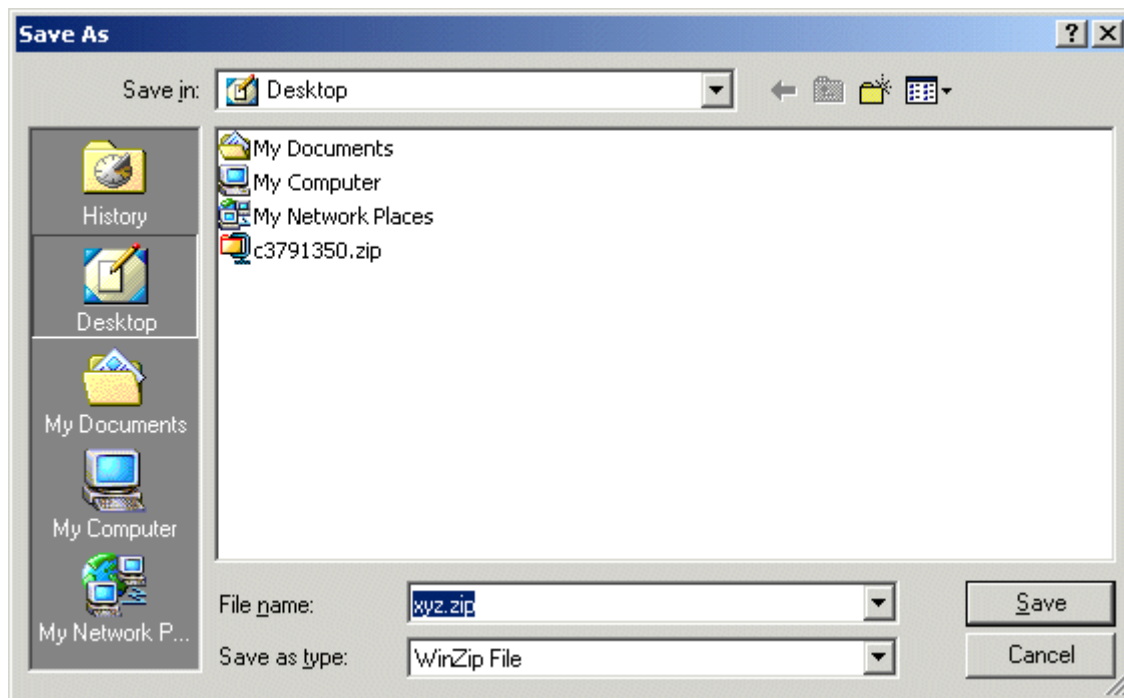
User	Context	User Organization	Object	Object Type	Event Name	Event Date
wadmin	Site		Team -- Site	Team	Create	Oct 20, 2003, 3:32 PM
wadmin	Windchill PDM		ba_wf	Wf Process Template Master	Create	Oct 20, 2003, 3:31 PM
wadmin	Windchill PDM		ba_lc	Life Cycle Master	Create	Oct 20, 2003, 3:29 PM
wadmin	Site		ba_group	Group	Create	Oct 20, 2003, 3:28 PM
wadmin	Site		Checked Out	Folder	Create	Oct 20, 2003, 3:28 PM
wadmin	Windchill PDM		ba_doc1	Document	Revise	Oct 20, 2003, 3:25 PM
wadmin	Site		Team -- Site	Team	Create	Oct 20, 2003, 3:25 PM
wadmin	Windchill PDM		Master -- ba_doc1	Document	Create	Oct 20, 2003, 3:25 PM
wadmin	Windchill		Team -- Windchill PDM	Team	Create	Oct 20, 2003, 3:25 PM

If you want to save this audit report to a file, click **Save to File**.

The Save to File window appears, allowing you to specify the name of the file:

The 'Save to File' dialog box has a blue title bar. Inside, it says '\* Please enter a name for the file' above a text input field. At the bottom, it says '\* Indicates required fields.' and has 'OK' and 'Cancel' buttons.

A **Save As** window appears in which you can choose a location in your file system or network to store the ZIP file.



# A

## Creating Business XML Files for Templates

This appendix provides information to assist in creating business XML files that can be used when creating an organization template, or creating or importing a product, library, or project template. The appendix provides an explanation of the XML elements that are required and optional in the business XML files.

<b>Topic</b>	<b>Page</b>
Overview .....	A-2
Creating Business XML Files for Organization Templates .....	A-2
Creating Business XML Files for Product Templates.....	A-6
Creating Business XML Files for Library Templates.....	A-8
Creating Business XML Files for Project Templates.....	A-10
Common XML Elements Used in Multiple Business XML Files .....	A-45

## Overview

Business XML files are used as input for the **Import Template** and **Create Template** actions provided in both Windchill PDMLink and Windchill ProjectLink. For details on the use of these actions, see [Creating Context Templates](#).

## Creating Business XML Files for Organization Templates

The content of the business XML file used as input when creating an organization template consists of a single, complex XML element named OrganizationConfig. This element is defined by the following DTD elements:

```
<!ELEMENT OrganizationConfig (OrgStructure?,
                               ContainerDefaults*,
                               ProductTemplate*,
                               LibraryTemplate*,
                               ProjectTemplate*,
                               ContainerTemplate*,
                               loadXMLFile*,
                               NmLoader*,
                               WTypeDefinitions*,
                               TypeBasedRule*,
                               SeedObjectRuleSet*,
                               createFromRule*,
                               ProductStructure?,
                               DocumentTemplate*,
                               ProjectEnumType*,
                               DiscussionForumTemplate?,
                               NotebookTemplate?)>

<!--ATTLIST OrganizationConfig
      autoPopulateCreators (true | false) #IMPLIED
-->
```

The following subelements contained in the OrganizationConfig element are not supported:

- ContainerDefaults
- ProductTemplate -- superseded by the ContainerTemplate element
- LibraryTemplate -- superseded by the ContainerTemplate element
- ProjectTemplate -- superseded by the ContainerTemplate element
- NmLoader
- WTypeDefinitions -- for information about defining data types, see the [Using Types and the Type Manager](#) chapter
- SeedObjectRuleSet -- superseded by the TypeBasedRule element
- createFromRule -- superseded by the TypeBasedRule element



- ProductStructure
- DocumentTemplate

To create the business XML file for an organization template, PTC recommends that you make a copy of an out-of-the-box organization template file (located in the <Windchill>/loadXMLFiles directory) and modify that template to produce a business XML file that you can use as input. For details on creating an organization template, see [Organization Templates](#) in the [Administering Containers](#) chapter

**Tip:** When modifying the XML file, include comments to document your changes and use indentation to indicate the hierarchical nature of the XML structure. Also use an XML editor that can validate your XML document against the DTD.

The supported subelements of OrganizationConfig are described in the following sections.

## OrgStructure Element

OrgStructure is an optional element of OrganizationConfig that is also used within elements other than OrganizationConfig.

Typically, the OrgStructure element is not used within OrganizationConfig and can be omitted.

For the details on this element, see [OrgStructure Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## ContainerTemplate Element

ContainerTemplate is an optional element of OrganizationConfig that is used to define the product, project and library template files that are available in an organization container.

The ContainerTemplate element can occur any number of times and is defined by the following DTD element.

```
<!ELEMENT ContainerTemplate (name,
                             description?,
                             containerClassName,
                             defaultValuesRule?,
                             locale?,
                             enabled?,
                             (xmlText | loadPath | zipEntry)?)>
```

If product, library, or project template files are referenced in an organization template file, then the templates are made available for use when the organization container is created. This means that the organization administrator does not need to create the templates. However, the organization administrator does need to

assign Product or Library Creators if the templates are product or library templates.

The following example illustrates how to use the ContainerTemplate element to add a product template, a project template and a library template to an organization template.

**Note:** The content of the containerClassName element must match these examples. Although, the DTD indicates that the business XML file can be specified in terms of XML text (xmlText element), a file path (loadPath element) or as a ZIP file (zipEntry element), only the loadPath element should be used in an organization template XML file. The value of the loadPath element is relative to the wt property wt.ixb.defaultLoadPath. This property defaults to *<Windchill>/loadXMLFiles*. For example, to specify a site-specific ProjectConfig and put it into file *<Windchill>/loadXMLFiles/siteSpecificProjectTemplate.xml*, the entry is *<loadPath>siteSpecificProjectTemplate.xml</loadPath>*. (The zipEntry element is used when the ContainerTemplate element is used to define a template being imported; see [Contents of Top-level XML File for Imported Templates](#) in the [Administering Containers](#) chapter for details.)

```
<ContainerTemplate>
  <name>Product Template Example</name>
  <description>Description goes here</description>
  <containerClassName>wt.pdmlink.PDMLinkProduct</containerClassName>
  <enabled>true</enabled>
  <loadPath>sampleProductTemplate</loadPath>
</ContainerTemplate>

<ContainerTemplate>
  <name>Project Template Example</name>
  <description>Description goes here</description>
  <containerClassName>wt.projmgmt.admin.Project2</containerClassName>
  <enabled>true</enabled>
  <loadPath>sampleProjectTemplate</loadPath>
</ContainerTemplate>

<ContainerTemplate>
  <name>Library Template Example</name>
  <description>Description goes here</description>
  <containerClassName>wt.inf.library.WTLibrary</containerClassName>
  <enabled>true</enabled>
  <loadPath>sampleLibraryTemplate</loadPath>
</ContainerTemplate>
```

## loadXMLFile Element

loadXMLFile is an optional element of OrganizationConfig that is used to include inline reading of any number of other XML files. It is also used within elements other than OrganizationConfig.

For the details on this element, see [loadXMLFile Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## TypeBasedRule Element

TypeBasedRule is an optional element of OrganizationConfig that defines default values for objects. The element is also used within elements other than OrganizationConfig.

For the details on this element, see [TypeBasedRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## ProjectEnumType Element

**Note:** ProjectEnumType is only applicable when Windchill ProjectLink is installed.

ProjectEnumType is an optional element of OrganizationConfig and is used to define the project roles. It is also used within elements other than OrganizationConfig.

For the details on this element, see [ProjectEnumType Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## DiscussionForumTemplate Element

DiscussionForumTemplate is an optional element of OrganizationConfig and is ignored for organization templates.

**Note:** This elements is not ignored when used within ProjectConfig elements.

## NotebookTemplate Element

NotebookTemplate is an optional element of OrganizationConfig and is ignored for organization templates.

**Note:** This element is not ignored when used within ProjectConfig elements.

## Creating Business XML Files for Product Templates

The content of the business XML file used as input when creating a product template consists of a single, complex XML element named ProductConfig. This element is defined by the following DTD element:

```
<!ELEMENT ProductConfig (projectMember*,
                          OrgStructure?,
                          DiscussionForumTemplate?,
                          ExportedRoleMemberMap?,
                          TeamTemplate*,
                          NmLoader*,
                          ACLRule*,
                          IndexPolicyRule*,
                          TypeBasedRule*,
                          TemplateDocument*,
                          DocumentTemplate*,
                          NotebookTemplate?,
                          ProductContainer*)>
```

The following subelements contained in the ProductConfig element are not supported:

- projectMember -- preferred method of defining roles is using the ExportedRoleMemberMap element (described in [ExportedRoleMemberMap Element](#))
- TeamTemplate
- NmLoader
- IndexPolicyRule
- TemplateDocument
- DocumentTemplate
- ProductContainer

PTC recommends that you export an existing product as a template, modify the business XML file (and associated files), and then import the template. For import details, see [Required Contents of ZIP File Used for Importing a Context Template](#) in the [Administering Containers](#) chapter.

**Tip:** When modifying the XML file, include comments to document your changes and use indentation to indicate the hierarchical nature of the XML structure. Also use an XML editor that can validate your XML document against the DTD.

The supported subelements of ProductConfig are described in the following sections.

## OrgStructure Element

OrgStructure is an optional element of ProductConfig that is also used within elements other than ProductConfig.

For the details on this element, see [OrgStructure Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## DiscussionForumTemplate Element

DiscussionForumTemplate is an optional element of ProductConfig and is ignored for product templates.

**Note:** This element is not ignored when used within ProjectConfig elements.

## NotebookTemplate Element

NotebookTemplate is an optional element of ProductConfig and is ignored for product templates.

**Note:** This element is not ignored when used within ProjectConfig elements.

## ExportedRoleMemberMap Element

ExportedRoleMemberMap is an optional element of ProductConfig that is also used within elements other than ProductConfig.

For the details on this element, see [ExportedRoleMemberMap Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## ACLRule Element

ACLRule is an optional element of ProductConfig that defines access control policy rules and is used within elements other than ProductConfig.

For the details on this element, see [ACLRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## TypeBasedRule Element

TypeBasedRule is an optional element of ProductConfig that defines default values for objects. The element is also used within elements other than ProductConfig.

For the details on this element, see [TypeBasedRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## Creating Business XML Files for Library Templates

The content of the business XML file used as input when creating a library template consists of a single, complex XML element named LibraryConfig. This element is defined by the following DTD element:

```
<!ELEMENT LibraryConfig (projectMember*,
                          OrgStructure?,
                          DiscussionForumTemplate?,
                          ExportedRoleMemberMap?,
                          TeamTemplate*,
                          NmLoader*,
                          ACLRule*,
                          IndexPolicyRule*,
                          TypeBasedRule*,
                          TemplateDocument*,
                          DocumentTemplate*,
                          NotebookTemplate?)>
```

The following subelements contained in the LibraryConfig element are not supported:

- projectMember -- preferred method of defining roles is using the ExportedRoleMemberMap element (described in [ExportedRoleMemberMap Element](#))
- TeamTemplate
- NmLoader
- IndexPolicyRule
- TemplateDocument
- DocumentTemplate

PTC recommends that you export an existing library as a template, modify the business XML file (and associated files), and then import the template. For import details, see [Required Contents of ZIP File Used for Importing a Context Template](#) in the [Administering Containers](#) chapter.

**Tip:** When modifying the XML file, include comments to document your changes and use indentation to indicate the hierarchical nature of the XML structure. Also use an XML editor that can validate your XML document against the DTD.

The supported subelements of LibraryConfig are described in the following sections.

## OrgStructure Element

OrgStructure is an optional element of LibraryConfig that is also used within elements other than LibraryConfig.

For the details on this element, see [OrgStructure Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## DiscussionForumTemplate Element

DiscussionForumTemplate is an optional element of LibraryConfig and is ignored for library templates.

**Note:** This element is not ignored when used within ProjectConfig elements.

## NotebookTemplate Element

NotebookTemplate is an optional element of LibraryConfig and is ignored for library templates.

**Note:** This element is not ignored when used within ProjectConfig elements.

## ExportedRoleMemberMap Element

ExportedRoleMemberMap is an optional element of LibraryConfig that is also used within elements other than LibraryConfig.

For the details on this element, see [ExportedRoleMemberMap Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## ACLRule Element

ACLRule is an optional element of LibraryConfig that defines access control policy rules and is used within elements other than LibraryConfig.

For the details on this element, see [ACLRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## TypeBasedRule Element

TypeBasedRule is an optional element of LibraryConfig that defines default values for objects. The element is also used within elements other than LibraryConfig.

For the details on this element, see [TypeBasedRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## Creating Business XML Files for Project Templates

The content of the business XML file used as input when creating a project template consists of a single, complex XML element named ProjectConfig. This element is defined by the following DTD element:

```
<!ELEMENT ProjectConfig (BaseProjectData?,  
                          (ExportedRoleMemberMap | projectMember*)?,  
                          OrgStructure?,  
                          NmLoader*,  
                          (TypeBasedRule | SeedObjectRuleSet)*,  
                          createFromRule*,  
                          ProductStructure?,  
                          DocumentTemplate*,  
                          TemplateDocument*,  
                          shareMapSet?,  
                          projectPlan*,  
                          ProjectEnumType*,  
                          DiscussionForumTemplate*,  
                          ForumSpec*,  
                          NotebookTemplate? ,  
                          NotebookSpec?)>
```

The following subelements contained in the ProjectConfig element are not supported:

- BaseProjectData
- projectMember -- preferred method of defining roles is using the ExportedRoleMemberMap element (described in [ExportedRoleMemberMap Element](#))
- NmLoader
- SeedObjectRuleSet-- superseded by the TypeBasedRule element
- createFromRule -- superseded by the TypeBasedRule element
- DocumentTemplate
- TemplateDocument
- ForumSpec
- NotebookSpec

PTC recommends that you export an existing project as a template, modify the business XML file (and associated files), and then import the template. For import details, see [Required Contents of ZIP File Used for Importing a Context Template](#) in the [Administering Containers](#) chapter.

**Tip:** When modifying the XML file, include comments to document your changes and use indentation to indicate the hierarchical nature of the XML structure. Also use an XML editor that can validate your XML document against the DTD.



The supported subelements of ProjectConfig are described in the following sections.

## ExportedRoleMemberMap Element

ExportedRoleMemberMap is an optional element of ProjectConfig that is also used within elements other than ProjectConfig.

For the details on this element, see [ExportedRoleMemberMap Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## OrgStructure Element

OrgStructure is an optional element of ProjectConfig that is used for defining domains, folders, groups and access control policy rules. The element also used within elements other than ProjectConfig.

For the details on this element, see [OrgStructure Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## TypeBasedRule Element

TypeBasedRule is an optional element of ProjectConfig that defines default values for objects. The element is also used within elements other than ProjectConfig.

For the details on this element, see [TypeBasedRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## ProductStructure Element

ProductStructure element is an optional element of ProjectConfig that contains all of the information required to define parts and documents within a product.

The ProductStructure element is defined by the following DTD element:

```
<!ELEMENT ProductStructure ((WTypeDefinitions | ibaDefinitions
| EPMDocument | EPMBuildLinksRule | EPMContainedIn
| EPMDescribeLink | EPMMemberLink | EPMReferenceLink
| EPMVariantLink | EPMBuildHistory
| WTPart | WTPartReferenceLink | WTPartUsageLink
| WTPartDescribeLink)*)>
```

The following subelements contained in the ProductStructure element can be present in an exported XML file for a project, but should not be changed when the exported project is imported:

- WTypeDefinitions -- for information about defining data types, see the [Using Types and the Type Manager](#) chapter
- ibaDefinitions

Many of the supported subelements referenced in the ProductStructure element are complex and are described in the following sections.

## EPMDocument Element

The EPMDocument element contains information that defines a CAD document and each CAD document represents a CAD model. The CAD model can be a component or an assembly. CAD document can contain a number of data files, URLs, or both data files and URLs. Therefore, a CAD document can store alternate representations of the CAD model. The primary content of the CAD document is usually the model file.

The EPMDocument element is defined by the following DTD elements:

```
<!ELEMENT EPMDocument (dtd?,
                        ObjectID,
                        objectContainerPath?,
                        objectContainerOid?,
                        ownerApplication?,
                        authoringApplication,
                        organizationId?,
                        number,
                        masterUfid?,
                        masterExternalTypeId?,
                        masterTypeOid?,
                        masterIba*,
                        name,
                        CADName?,
                        epmDocType,
                        epmDocSubType?,
                        defaultUnit?,
                        description?,
                        authoringApplicationVersion?,
                        dbKeySize?,
                        isVerified?,
                        revisionNumber?,
                        familyTableStatus?,
                        epmCADReferenceControl?,
                        extentsValid?,
                        epmBoxExtents?,
                        domainName?,
                        folderPath,
                        folderOid?,
                        versionInfo?,
                        lifecycleInfo?,
                        teamIdentity?,
                        contentItem*,
                        externalTypeId?,
                        typeOid?,
                        iba*,
                        excludedIba*,
                        EPMPParameterMap*,
                        isMissingDependents?,
                        EPMFeatureValue*,
                        EPMPParameterValue*,
                        archiveInfo?)>
```

The following subelements contained in the EPMDocument element are not supported:

- objectContainerPath
- objectContainerOid
- organizationId
- domainName
- folderPath
- folderOid
- versionInfo
- archiveInfo

The following subelements contained in the EMPDocument element can be present in an exported XML file for a project, but should not be changed when the exported project is imported:

- ObjectID
- externalTypeId
- typeId
- masterExternalTypeId
- masterTypeId
- masterUfid
- masterIba
- iba
- excludedIba
- ownerApplication
- authoringApplication
- authoringApplicationVersion
- epmDocType
- epmDocSubType
- defaultUnit
- familyTableStatus
- isMissingDependents
- isVerified

- extentsValid
- epmBoxExtents
- EPMPParameterMap
- EPMFeatureValue
- EPMPParameterValue
- dbKeySize
- revisionNumber
- epmCADReferenceControl
- LifecycleInfo
- teamIdentity
- contentItem
- TypeBasedRule Element

**Note:** The many of the values of these subelements are derived from the model file. As a result, changing them leads to data corruption and causes discrepancy between the model file and the information in Windchill.

The following table provides additional information about the customizable subelements in the DTD elements for EMPDocument:

Subelement	Description
number	Number of the EPMDocument.
name	Name of the EPMDocument.

Subelement	Description
CADName	<p>This element is the name of the CAD file. Even though this element is optional in the DTD, CAD name is required.</p> <p>This element can be changed if the document is authored by Pro/ENGINEER. That is, the authoringApplication on this document should be PROE.</p> <p><b>Note:</b> If this document is a top generic, you must also change the name on the EPMSepFamilyTable. CAD name of the top generic and the name of the internal family table object must be the same; otherwise, the family table will get corrupted and many operations on the family table will not work properly. A document is top generic if the familyStatus on this document is 2.</p>
description	Description of the EPMDocument.

## EPMBuildLinksRule

EPMBuildLinksRule represents the relationship between the build source and the build target. This relationship is required by the build process to determine the source for a target or the target for a source. The build source of the EPMBuildLinksRule is a CAD document. The target is a WTPart object.

**Note:** You cannot associate an EPMBuildRule to an EPMDocument that has an effectivity assigned to it.

The EPMBuildLinksRule element is defined by the following DTD element:

```
<!ELEMENT EPMBuildLinksRule (dtd?,
                             ObjectID,
                             (buildSource, buildTarget))>
```

The following table provides additional information about the customizable subelements in the DTD elements for EMPDBuildLinksRule:

Subelement	Description
ObjectID	Object identifier of the object. The identifier is set by the export. For details on ObjectID, see <a href="#">ObjectID Element</a> .
buildSource	This element is the source CAD document.

Subelement	Description
buildTarget	This element is the target WTPart. The information on the source CAD document is propagated to the target

**Note:** Adding a new build rule to the template is allowed if the following conditions are satisfied:

- The CAD document and WTPart object are in the same template file.
- The CAD document or WTPart object is not associated to another WTPart object or CAD document using a build rule.

## EPMDescribeLink

EPMDescribeLink is a relationship specifying that a CAD document provides descriptive information for a WTPart object.

The EPMDescribeLink element is defined by the following DTD elements:

```
<!ELEMENT EPMDescribeLink (dtd?,
                           ObjectID,
                           describedBy,
                           describes,
                           isBuiltFlag?)>
```

**Note:** Modifying the information on the link is not supported if isBuiltFlag is true. You can however add a new association between a CAD document and a WTPart to the template. The CAD document and the WTPart must exist in the same template file.

The following table provides additional information about the customizable subelements in the DTD elements for EMPDescribeLink:

Subelement	Description
ObjectID	Object identifier of the object. The identifier is set by the export. For details on ObjectID, see <a href="#">ObjectID Element</a>
describes	This element is the WTPart described by the CAD document.
describedBy	This element is the CAD document describing the WTPart.
isBuiltFlag	This element should be set to false.

## EPMMemberLink

EPMMemberLink describes a uses relationship between a CAD document and a CAD document master. This relationship is derived from the CAD model file.

The EPMMemberLink element is defined by the following DTD element:

```
<!ELEMENT EPMMemberLink (dtd?,
                           ObjectID,
                           usedBy,
                           uses,
                           externalTypeId?,
                           typeOid?,
                           depType?,
                           asStoredChildName?,
                           isRequired?,
                           uniqueLinkId?,
                           uniqueNDId?,
                           iba*,
                           excludedIba*,
                           EPMPParameterMap*,
                           isSuppressed?,
                           name?,
                           identifier?,
                           quantityAmount?,
                           quantityUnit?,
                           isPlaced?,
                           (hasTransform, transform)?,
                           compNumber?,
                           compRevNumber?,
                           compLayerIdx?,
                           epmCADReferenceControl?,
                           occurrences*)>
```

**Note:** The subelements contained in the EPMMemberLink element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.

The values of many of these subelements are derived from the model file. As a result, changing them leads to data corruption and causes discrepancy between the model file and the information in Windchill.

## EPMReferenceLink

EPMReferenceLink describes a reference relationship between a CAD document and a CAD document master. This relationship is derived from the CAD model file.

The EPMReferenceLink element is defined by the following DTD elements:

```
<!ELEMENT EPMReferenceLink (dtd?,
                             ObjectID,
                             referencedBy,
                             references,
                             externalTypeId?,
                             typeOid?,
```

```

depType?,
asStoredChildName?,
isRequired?,
uniqueLinkId?,
uniqueNDId?,
iba*,
excludedIba*,
EPMPParameterMap*)>

```

**Note:** The subelements contained in the EPMReferenceLink element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.

The values of many of these subelements are derived from the model file. As a result, changing them leads to data corruption and causes discrepancy between the model file and the information in Windchill.

## WTPart

A part is an item that can be produced or consumed. Parts could be assembled to create other parts. Parts used in Windchill repository have wt.part.WTPart type.

The WTPart element is defined by the following DTD element:

```

<!ELEMENT WTPart (dtd?,
  ObjectID,
  objectContainerPath?,
  objectContainerOid?,
  organizationId?,
  ((number, masterUfid?), name, defaultUnit?),
  partType,
  partSource,
  domainName?,
  folderPath,
  folderOid?,
  view?,
  versionInfo?,
  lifecycleInfo?,
  teamIdentity?,
  externalTypeId?,
  typeOid?,
  contentItem*,
  iba*,
  excludedIba*,
  archiveInfo?,
  markUp?)>

```

**Note:** The subelements contained in the WTPart element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.



## WTPartReferenceLink

A WTPart could reference a document. A part should reference a document if the document is not owned by the part or the document could be revised or iterated independently.

The WTPartReferenceLink element is defined by the following DTD element:

```
<!ELEMENT WTPartReferenceLink (dtd?,  
                                ObjectID,  
                                (referencedBy, references))>
```

**Note:** The subelements contained in the WTPartReferenceLink element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.

## WTPartUsageLink

The WTPartUsageLink is an association defined to be used to build structures. The WTPartUsageLink aggregated quantity could be used to indicate the amount of the component that is being consumed.

The WTPartUsageLink element is defined by the following DTD element:

```
<!ELEMENT WTPartUsageLink (dtd?,  
                             ObjectID,  
                             (usedBy, uses),  
                             externalTypeId?,  
                             typeOid?,  
                             iba*,  
                             excludedIba*,  
                             quantityAmount,  
                             quantityUnit,  
                             lineNumber?,  
                             occurrences?,  
                             occurrence*)>
```

**Note:** The subelements contained in the WTPartUsageLink element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.

## WTPartDescribeLink

A part could be described by WTPartDescribeLink. If a particular version of a part is described by a particular version of a document, you should use this link.

The WTPartDescribeLink element is defined by the following DTD element:

```
<!ELEMENT WTPartDescribeLink (dtd?,  
                                ObjectID,  
                                (describedBy, describes))>
```

**Note:** The subelements contained in the WTPartDescribeLink element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.

## shareMapSet Element

shareMapSet is an optional element of ProjectConfig that contains all of the share maps to be imported in a project. Typically, this element is included when a project that has been exported has shared objects. There is a single shareMapSet element in the import file that contains one shareMap subelement for each shared object.

**Note:** To obtain the same set of shared objects in an imported project template, use the shareMapSet element that was exported without making any manual changes.

The shareMapSet element can occur one time and is defined by the following DTD elements:

```
<!ELEMENT shareMapSet (shareMap*)>

<!ELEMENT shareMap (ObjectID,
                    shareSpec?,
                    shareTarget,
                    folderPath?)>

<!ELEMENT shareSpec (shareType, shareKey)>

<!ELEMENT shareType (#PCDATA)>

<!ELEMENT shareKey (#PCDATA)>

<!ELEMENT shareTarget (ObjectReference, iterationId?)>

<!ATTLIST shareTarget isVersion (true | false) "true">
```

The following table provides additional information about some of the subelements in the DTD elements for shareMapSet:

Subelement	Description
shareMap	Individual specification of a shared object.
ObjectID	Object identifier of the object (folder or single object) to share. The identifier is set by the export. For details on ObjectID, see <a href="#">ObjectID Element</a> .
folderPath	The folder specification of the target folder in which the imported objects are shared.
shareType	Valid values are: FOLDER and SINGLE
shareKey	String representation of long number (like an ID). Share key is 0 for SINGLE shares.
ObjectReference	The object reference to the object being shared

Subelement	Description
iterationId	The version information when the object is a version share (isVersion is true); omitted when isVersion is false.
isVersion	Determines whether the share is a version or object share. true indicates version share; false indicates object share.

## projectPlan Element

projectPlan is an optional element of ProjectConfig that contains all of the information required to create a project plan in a specific project template.

```
<!ELEMENT projectPlan ((manager | managerInfo)*,
                        subscriptionPolicy?,
                        planMode?,
                        creator?,
                        projectId?,
                        projectPlanId,
                        executionInfo,
                        (Activity | Milestone | Summary |
                         SubProject)*,
                        dependencyLink*,
                        containmentLink*,
                        Deliverable*,
                        Resource*,
                        resourceLink*,
                        duration?,
                        projectVariable*)>
```

The following subelements contained in the projectPlan element are not supported:

- manager -- not used when template is imported.
- managerInfo -- not used when template is imported.
- creator -- not used from within a project template; the user selecting the template is the creator of the project.
- projectId -- not used from within a project template; the container that the plan is being imported into is used as the plan's project container.
- SubProject
- projectVariable

Many of the supported subelements referenced in the projectPlan element are complex and are described in the following sections.

## ObjectID Element

The ObjectID element is used as a subelement to identify an object.

The ObjectID element is defined by the following DTD elements:

```
<!ELEMENT ObjectID (localId?, objectId?, ufid?)>
<!ELEMENT objectId (#PCDATA)>
<!ELEMENT localId (#PCDATA)>
<!ELEMENT ufid (#PCDATA)>
```

The subelements are described in the following sections.

### localId Element

The localId element refers to an object's local identifier. Local identifiers are only valid during the import process.

The content for the localId element can be any string. The only requirement is that the localId for each object defined in the XML file is unique.

### objectId Element

The objectId element refers to an object's Windchill object ID. The Windchill object ID is valid for the entire life of the object but is only valid on one Windchill system. The object does not need to be defined in the same XML file. Notice that this element is different from parent ObjectID element in that the "o" and the "d" are lower case characters).

**Note:** Using objectId for plan items is not currently supported.

The content of the objectId element must correspond to a Windchill object identifier. The format is as follows:

```
<RT>:<classname>:<OID>
```

Where <RT> is the reference type. The reference type is one of the following:

OR - object reference

VR - version reference

<classname> is the fully-qualified Java class name for the object. For example, wt.projmgmt.execution.SummaryActivity.

<OID> is the identifier for the object. This is the value found in the ida2a2 column in the object's database table.

The following is an example of a Windchill object identifier for a summary activity:

```
OR:wt.projmgmt.execution.SummaryActivity:12612
```

An object can be identified by both a localId and an objectId. But, if localId is specified then the objectId element is ignored.

### ufid Element

The ufid element refers to an object's Unique Federation Identifier (UFID). The UFID contains the distinguished name of the object. This ID is valid for the entire life of the object and is valid across a set of federated Windchill systems. Using this ID allows an object to reference an object in another federated Windchill system.

**Note:** Using ufid for plan items is not currently supported.

### subscriptionPolicy Element

Owner and manager subscriptions can be created for project plans, activities, milestones, summary activities, sub-projects and deliverables. Use the subscriptionPolicy element to specify the events for which project managers and object owners should receive notifications.

The subscriptionPolicy element is defined by the following DTD element:

```
<!ELEMENT subscriptionPolicy (ManagerSubscriptionPolicy?,
                               OwnerSubscriptionPolicy?)>
```

The following sections describe the subelements of subscriptionPolicy.

### ManagerSubscriptionPolicy Element

Use the ManagerSubscriptionPolicy element to create subscriptions for the project manager role.

The ManagerSubscriptionPolicy element is defined by the following DTD elements:

```
<!ELEMENT ManagerSubscriptionPolicy (EventSubscription*)>

<!ATTLIST ManagerSubscriptionPolicy
  statusRed (true | false) "false"
  onCompletion (true | false) "false"
>

<!ELEMENT EventSubscription EMPTY>

<!ATTLIST EventSubscription
  event (DEADLINE | DEADLINE_CHANGE | FINISH_CHANGE |
         OBJECT_DELETION | OWNER_CHANGE | PERCENT_CHANGE |
         STATE_CHANGE | STATUS_CHANGE | TIME_PAST_DEADLINE |
         TIME_TO_DEADLINE | RISK_CHANGE | NODE_CREATION)
  #REQUIRED
  subscribe (true | false) #REQUIRED
>
```

You subscribe to most events using an EventSubscription element for each event. Additionally, you subscribe to two special project manager events (Turns Red and Completion) by setting the values of attributes:

- Set statusRed attribute equal to true to subscribe to the Status Turns Red event.
- Set onCompletion attribute equal to true to subscribe to the On Completion event.

### OwnerSubscriptionPolicy Element

Use the OwnerSubscriptionPolicy element to create subscriptions for the object owner role. These events are subscribed to using an EventSubscription element.

The OwnerSubscriptionPolicy element is defined by the following DTD elements:

```
<!ELEMENT OwnerSubscriptionPolicy (EventSubscription*)>

<!ELEMENT EventSubscription EMPTY>

<!ATTLIST EventSubscription
    event (DEADLINE | DEADLINE_CHANGE | FINISH_CHANGE |
        OBJECT_DELETION | OWNER_CHANGE | PERCENT_CHANGE |
        STATE_CHANGE | STATUS_CHANGE | TIME_PAST_DEADLINE |
        TIME_TO_DEADLINE | RISK_CHANGE | NODE_CREATION)
    #REQUIRED
    subscribe (true | false) #REQUIRED
>
```

Use the EventSubscription element to indicate the events to which the owner role should be subscribed. Use one EventSubscription element for each event.

## subscriptionPolicy Element Example

The following example shows a set of events for manager and owner subscription using the subscriptionPolicy element:

```
<subscriptionPolicy>
  <ManagerSubscriptionPolicy onCompletion="true" statusRed="true">
    <EventSubscription event="DEADLINE" subscribe="true"></EventSubscription>
    <EventSubscription event="DEADLINE_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="FINISH_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="NODE_CREATION" subscribe="false"></EventSubscription>
    <EventSubscription event="OBJECT_DELETION" subscribe="true"></EventSubscription>
    <EventSubscription event="OWNER_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="PERCENT_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="RISK_CHANGE" subscribe="false"></EventSubscription>
    <EventSubscription event="STATE_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="STATUS_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="TIME_PAST_DEADLINE"
subscribe="true"></EventSubscription>
    <EventSubscription event="TIME_TO_DEADLINE"
subscribe="true"></EventSubscription>
  </ManagerSubscriptionPolicy>
  <OwnerSubscriptionPolicy>
    <EventSubscription event="DEADLINE" subscribe="true"></EventSubscription>
    <EventSubscription event="DEADLINE_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="FINISH_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="NODE_CREATION" subscribe="false"></EventSubscription>
    <EventSubscription event="OBJECT_DELETION" subscribe="true"></EventSubscription>
    <EventSubscription event="OWNER_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="PERCENT_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="RISK_CHANGE" subscribe="false"></EventSubscription>
    <EventSubscription event="STATE_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="STATUS_CHANGE" subscribe="true"></EventSubscription>
    <EventSubscription event="TIME_PAST_DEADLINE"
subscribe="true"></EventSubscription>
    <EventSubscription event="TIME_TO_DEADLINE"
subscribe="true"></EventSubscription>
  </OwnerSubscriptionPolicy>
</subscriptionPolicy>
```

## planMode Element

The planMode element is used to specify the plan execution control.

The planMode element is defined by the following DTD elements:

```
<!ELEMENT planMode EMPTY>

<!ATTLIST planMode
  mode (staticPlan | activityDriven | taskDriven) "staticPlan"
>
```

The following table provides additional information about the modes:

Mode	Description
staticPlan	Allows manual control project execution
activityDriven	Automatically execute activities (without task delivery)
taskDriven	Automatically execute activities and deliver tasks

## projectPlanId Element

The projectPlanId element provides an identifier for a project plan and indicates the plan to which a plan node belongs. This element is required for all plan nodes.

```
<!ELEMENT projectPlanId (ObjectID)>
```

During an import, the value of this element is only used if the container that the data is being imported into is not a project container. Otherwise, the plan for the project container being imported into is used.

For details on ObjectID, see [ObjectID Element](#).



## executionInfo Element

The executionInfo element is used to specify many of the attributes that are shared by project plan objects. This element is used when defining plans, activities, milestones, summary activities, sub-projects, and deliverables.

The executionInfo element is defined by the following DTD elements:

```
<!ELEMENT executionInfo (name,  
                           healthStatus?,  
                           statusDescription?,  
                           riskValue?,  
                           projectCategory?,  
                           projectState?,  
                           projectPhase?,  
                           executionDescription?,  
                           percentComplete?,  
                           riskDescription?,  
                           startDate?,  
                           finishDate?,  
                           timeToStart?,  
                           estimatedFinish?,  
                           deadline?,  
                           inheritedDeadline?,  
                           projectConfiguration?,  
                           workflowConfiguration?,  
                           timeToDeadline?,  
                           timePastDeadline?,  
                           totalWork?,  
                           performedWork?,  
                           totalWorkCost?,  
                           currentWorkCost?,  
                           priority?,  
                           responsibleRole?)>  
  
<!ATTLIST executionInfo  
  enabled (true | false) "true"  
>
```

In some cases, a project plan object is disabled rather than being deleted. The enabled attribute on the executionInfo element is used to specify whether the object is enabled or disabled. This attribute should always be set to true when working with a template and the default is true. Therefore, you can omit this element for project templates.

The following subelements contained in the executionInfo element are not supported for project templates:

- projectCategory
- projectState
- projectPhase
- percentComplete
- startDate

- finishDate
- timeToStart
- estimatedFinish
- inheritedDeadline
- projectConfiguration
- workflowConfiguration
- totalWork
- performedWork
- totalWorkCost
- currentWorkCost
- priority
- responsibleRole

The following sections describe the supported subelements of executionInfo.

### name Element

The name element is used to name activities, summary activities, milestones, sub-projects, and deliverables.

**Note:** This element is not supported for project plans; the name of the project plan is entered when the plan is created.

The name element is defined by the following DTD element:

```
<!ELEMENT name (#PCDATA)>
```

### healthStatus Element

The healthStatus element is used to state the health status for activities and deliverables. It is also used to state the health status for milestones when no deliverables are included.

**Note:** This element is not supported for project plans or summary activities; the health status is rolled up from child nodes for these objects. The element is also not supported for milestones when there is an associated delivery; the health status is rolled up from its deliverables.

The healthStatus element is defined by the following DTD elements:

```
<!ELEMENT healthStatus (#PCDATA)>

<!ATTLIST healthStatus
    status (GREEN | YELLOW | RED | UNAVAILABLE | CUSTOM)
    #REQUIRED
>
```

The CUSTOM health status value is not supported.

### **statusDescription Element**

The statusDescription element describes the status of the project plan.

The statusDescription element is defined by the following DTD element:

```
<!ELEMENT statusDescription (#PCDATA)>
```

### **riskValue Element**

The riskValue element specifies the risk value of the project plan object.

The riskValue element is defined by the following DTD elements:

```
<!ELEMENT riskValue (#PCDATA)>

<!ATTLIST riskValue
    value (VERYLOW | LOW | MEDIUM | HIGH | VERYHIGH | UNAVAILABLE
    | CUSTOM) #REQUIRED
>
```

The content of the element is normally empty. The risk value is specified using the riskValue attribute. The CUSTOM risk value is not supported.

This element is supported when importing milestones, summaries, activities and deliverables; it is not supported when importing project plans.

### **riskDescription Element**

The riskDescription element specifies the value of the object's risk description attribute.

The riskDescription element is defined by the following DTD elements:

```
<!ELEMENT riskDescription (#PCDATA)>
```

This element is supported when importing milestones, summaries, activities and deliverables; it is not supported when importing project plans.

### **executionDescription Element**

The executionDescription element is used to describe activities, summary activities, milestones, sub-projects, and deliverables.

**Note:** This element is not supported for project plans; the description of the project plan is displayed in the user interface.

The executionDescription element is defined by the following DTD element:

```
<!ELEMENT executionDescription (#PCDATA)>
```

## Elements for the Project Plan Schedule

The following group of elements provide schedule information about a project plan object:

The scheduling elements contained in the executionInfo element are defined by the following DTD elements:

```
<!ELEMENT deadline (#PCDATA)>

<!ELEMENT timeToDeadline (#PCDATA)>
<!ATTLIST timeToDeadline
    unit (day | hour | minute | second | millisecond) #IMPLIED
>
<!ELEMENT timePastDeadline (#PCDATA)>
<!ATTLIST timePastDeadline
    unit (day | hour | second | millisecond) #IMPLIED
>
```

The following table provides additional information about the supported subelements used in providing schedule information about a project plan:

Subelement	Description
deadline	Specifies the deadline for a project plan or plan node. The format required in this element is described after this table.
timeToDeadline	<p>Specifies an integer representing the amount of time before the deadline that the Days Before Deadline notifications should be sent. The integer represents either days, hours, seconds, or milliseconds (depending on the value of the unit attribute).</p> <p>This element is only applicable if project manager or object owner subscriptions have been created for the Days Before Deadline event.</p>
timePastDeadline	<p>Specifies an integer representing the amount of time after the deadline that the Days After Deadline notifications should be sent. The integer represents either days, hours, seconds, or milliseconds (depending on the value of the unit attribute).</p> <p>This element is only applicable if project manager or object owner subscriptions have been created for the Days After Deadline event.</p>

The format for the deadline element is as follows:

YYYY-MM-DD HH:MM:SS.NNN

Where:

- YYYY is the four-digit year
- MM is a two-digit number between 01 and 12 representing the month
- DD is a two-digit number between 01 and 31 representing the day of the month
- HH is a two-digit number between 00 and 23 representing the hour of the day
- MM is a two-digit number between 00 and 59 representing the minutes within the hour
- SS is a two-digit number between 00 and 59 representing the seconds within the minute
- NNN is a one-, two- or three-digit number representing the fraction of a second

The time zone is assumed to be GMT.

For example, use:

2005-04-18 04:00:00.0

For April 18, 2005 05:00 AM GMT

The convention used by the project management module is to align dates with day boundaries. A date representing the beginning of a time period (estimated start time and actual start time) is aligned with midnight on that day in the user's time zone. For example, if the user who has a preferred time zone is Central (United States) set the start date to April 18, 2005, the date would be exported as follows:

2005-04-18 05:00:00.0

This date represents April 18, 2005 12:00 AM CDT.

Dates that represent the end of a time period (estimated finish date, actual finish date and deadline) are aligned with midnight on the day following that day in the user's time zone. For example, if the user referred to above set the deadline to April 25, 2005, the date would be exported as follows:

2005-04-26 05:00:00.0

This date represents April 26, 2005 12:00 AM CDT.

## Activity, Milestone, and Summary Elements

As part of a project plan, you can specify one of the following subelements:

- Activity - defines activities in a project plan.
- Milestone - defines milestones in a project plan.
- Summary - defines summary activities in a project plan.

The Activity, Milestone, and Summary elements are defined by the following DTD elements:

```
<!ELEMENT Activity (ObjectID,
                    projectPlanId,
                    (owner | ownerInfo)?,
                    subscriptionPolicy?,
                    activityInfo,
                    resourceAssignee*,
                    duration?,
                    projectVariable*)>

<!ELEMENT Milestone (ObjectID,
                    subscriptionPolicy?,
                    projectPlanId,
                    (owner | ownerInfo)?,
                    activityInfo,
                    duration?,
                    projectVariable*)>

<!ELEMENT Summary (ObjectID,
                  projectPlanId,
                  (owner | ownerInfo)?,
                  subscriptionPolicy?,
                  activityInfo,
                  duration?,
                  projectVariable*)>

<!ELEMENT activityInfo (executionInfo, nodeNumber?)>

<!ELEMENT resourceAssignee (ObjectID, maxPercentAvailable?)>

<!ELEMENT ownerInfo (name, ObjectReference?)>

<!ELEMENT duration (#PCDATA)>
<!ATTLIST duration
    unit (day | hour | second | millisecond) #IMPLIED
>
```

The following subelements are not supported:

- owner
- maxPercentAvailable
- projectVariable

The following table provides additional information about the subelements that are used in one or more of the Activity, Milestone, Summary, and SubProject elements:

Subelement	Description
ObjectID	ID of the object being defined. The ID can be used to reference this object in other places in the XML file. For example, this ID can be used when defining a dependencyLink between two activities. For details on ObjectID, see <a href="#">ObjectID Element</a> .
projectPlanId	Specifies the identifier for a project plan. See <a href="#">projectPlanId Element</a> .
ownerInfo	<p>Specifies the owner of the deliverable. Set the content of the name element (a child element of the ownerInfo element) to the user name or e-mail address of the user.</p> <p>The user that is specified as an owner must be a member of the project. This can be accomplished by using the ExportedRoleMemberMap element to add the user to the project team as part of the XML import or by adding the user to the team prior to importing the XML file.</p>
subscriptionPolicy	Specifies owner and manager subscriptions created for the activity, milestone, summary, or subproject. See <a href="#">subscriptionPolicy Element</a> .
activityInfo	Specifies information for activities, summary activities, milestones and sub-projects. It holds the executionInfo and nodeNumber elements
duration	Specifies the duration of the activity, milestone, summary, or subproject. See <a href="#">duration Element</a> .
executionInfo	Attributes of the activity, milestone, summary, or subproject. See <a href="#">executionInfo Element</a> .
nodeNumber	<p>Allows the ID of the plan node to be specified. The ID is the number that is displayed in the left-most column of on the plan page. The ID is used to determine the order in which plan nodes are listed in the hierarchical plan view.</p> <p>The IDs must be consistent with the hierarchical structure of the plan. For example, the IDs for the children of a summary activity must immediately follow the ID for the summary activity itself.</p>

Subelement	Description
resourceAssignee	Allows resources to be assigned to an activity.

## dependencyLink Element

The dependencyLink element is used to define a predecessor constraint.

The dependencyLink element is defined by the following DTD elements:

```

<!ELEMENT dependencyLink (predecessorID, successorID,
                           nodeLinkType?)>
<!ELEMENT predecessorID (ObjectID)>
<!ELEMENT successorID (ObjectID)>
<!ELEMENT nodeLinkType (customType?)>
<!ATTLIST nodeLinkType
    linkType (FINISH_START | FINISH_FINISH | START_FINISH |
             START_START | CUSTOM) #REQUIRED
>

```

The following table provides additional information about the dependencyLink subelements:

Subelement	Description
predecessorID	ID of the predecessor node. For details on ObjectID, see <a href="#">ObjectID Element</a> .
successorID	ID of the successor node. For details on ObjectID, see <a href="#">ObjectID Element</a> .
nodeLinkType	Specifies the constraint type. Currently only FINISH_START is supported. This constraint type specifies that the successor can start as soon as the predecessor has finished.

**Note:** Cyclical dependency links are not allowed. This means that the successorID and the predecessorID must refer to distinct objects and the it must not be possible to traverse a series of dependency links and encounter the same node twice.



The following example shows the use of dependencyLink element:

```
<dependencyLink>
  <predecessorID><ObjectID>
    <localId>wt.projmgmt.execution.Milestone:105878</localId></ObjectID>
  </predecessorID>
  <successorID><ObjectID>
    <localId>wt.projmgmt.execution.SummaryActivity:106075</localId></ObjectID>
  </successorID>
  <nodeLinkType linkType="FINISH_START"></nodeLinkType>
</dependencyLink>
```

## containmentLink Element

The containmentLink element is used to define parent child relationships between certain plan objects and summary activities. Only milestones, activities, summary activities and sub-projects can be contained in a summary activity. Deliverables cannot be contained in a summary activity.

The containmentLink element is defined by the following DTD elements:

```
<!ELEMENT containmentLink (summaryID, nodeID*)>
<!ELEMENT summaryID (ObjectID)>
<!ELEMENT nodeID (ObjectID)>
```

The following table provides additional information about the containmentLink subelements:

Subelement	Description
summaryID	ID of the summary activity. For details on ObjectID, see <a href="#">ObjectID Element</a> .
nodeID	ID of the child node. The child node must be a milestone, activity, summary activity, or subproject. For details on ObjectID, see <a href="#">ObjectID Element</a> .

**Note:** Cyclical containment links are not allowed. This means that the summaryID and the nodeID must refer to distinct objects and the it must not be possible to traverse a series of containment links and encounter the same node twice.

The following example shows the use of containmentLink element:

```
<containmentLink>
  <summaryID>
    <ObjectID>
      <localId>wt.projmgmt.execution.SummaryActivity:106075</localId>
    </ObjectID>
  </summaryID>
  <nodeID>
    <ObjectID>
      <localId>wt.projmgmt.execution.ProjectActivity:106103</localId>
    </ObjectID>
  </nodeID>
</containmentLink>
```

## Deliverable Element

The Deliverable element defines a deliverable to include in a project plan.

The Deliverable element is defined by the following DTD elements:

```
<!ELEMENT Deliverable (ObjectID?,
                        subscriptionPolicy?,
                        projectPlanId?,
                        activityID?,
                        mileStoneId*,
                        (owner | ownerInfo)?,
                        executionInfo,
                        targetConfig?,
                        targetReference?,
                        duration?)>

<!ATTLIST Deliverable
  orphan (true | false) #IMPLIED
>

<!ELEMENT activityID (ObjectID)>

<!ELEMENT mileStoneId (ObjectID)>

<!ELEMENT owner (#PCDATA)>

<!ELEMENT ownerInfo (name, ObjectReference?)>

<!ELEMENT targetReference (ObjectID | createFromRule)>

<!ELEMENT duration (#PCDATA)>
<!ATTLIST duration
  unit (day | hour | second | millisecond) #IMPLIED
>
```

**Note:** If a user selects Deliverables when exporting a project template, all deliverables are exported as orphans.

If a deliverable is an orphan deliverable, then the following attributes are ignored:

projectPlanId  
activityID  
mileStoneId

The following subelements are not supported:

- owner
- orphan
- ObjectReference
- targetConfig
- createFromRule

The following table provides additional information about the supported Deliverable subelements:

Subelement	Description
ObjectID	ID of the deliverable. For details on ObjectID, see <a href="#">ObjectID Element</a> .
subscriptionPolicy	Specifies owner and manager subscriptions created for the deliverable. See <a href="#">subscriptionPolicy Element</a> .
projectPlanId	Specifies the identifier for a project plan. See <a href="#">projectPlanId Element</a> .
activityID	<p>Specifies the ID of the activity. Allows the deliverable to be associated with an activity.</p> <p>If you have specified link deliverables in your project configuration, then you ensure only one deliverable is associated with an activity.</p> <p>For details on ObjectID, see <a href="#">ObjectID Element</a>.</p>
mileStoneId	Specifies the ID of the milestone. This allows the deliverable to be associated with one or more milestones. One element is created per milestone reference. For details on ObjectID, see <a href="#">ObjectID Element</a> .

Subelement	Description
ownerInfo	Specifies the owner of the deliverable. Set the content of the name element (a child element of the ownerInfo element) to the user name or e-mail address of the user.  The user that is specified as an owner must be a member of the project. This can be accomplished by using the ExportedRoleMemberMap element to add the user to the project team as part of the XML import or by adding the user to the team prior to importing the XML file.
executionInfo	Attributes of the deliverable. See <a href="#">executionInfo Element</a> .
targetReference	Specifies the ID of the reference. This allows the deliverable to be associated with a target document or part. The ObjectID element is used to provide a reference to the target document or part. For details on ObjectID, see <a href="#">ObjectID Element</a> .
duration	Specifies the duration of the deliverable. See <a href="#">duration Element</a> .

## Resource Element

The Resource element defines the resources of the project plan.

The Resource element is defined by the following DTD elements:

```
<!ELEMENT Resource (ObjectID?, projectPlanId, resourceInfo)>
<!--
  resourceType (PERSON | ROLE | EQUIPMENT | MATERIAL |
  INFORMATION | FACILITY) #REQUIRED
-->
```

**Note:** Only PERSON, EQUIPMENT, MATERIAL and FACILITY are supported resource types.

The following table provides additional information about the supported Resource subelements:

Subelement	Description
ObjectID	ID of the resource. For details on ObjectID, see <a href="#">ObjectID Element</a> .
projectPlanId	Specifies the identifier for a project plan. See <a href="#">projectPlanId Element</a> .
resourceInfo	Specifies the details about the resource. Additional information is in the next section.

### resourceInfo Element

The resourceInfo element defines the resource information for a resource.

The resourceInfo element is defined by the following DTD elements:

```

<!ELEMENT resourceInfo (name,
                        description?,
                        resourceCategory?,
                        label?,
                        maxUnits?,
                        resourceReference?,
                        standardRate?,
                        costPerUse?,
                        overtimeRate?,
                        maxPercentUtilization?)>

<!ELEMENT name (#PCDATA)>

<!ELEMENT description (#PCDATA)>

<!ELEMENT maxUnits (#PCDATA)>

<!ELEMENT resourceReference (userInfo | ObjectReference)>

<!ELEMENT userInfo (name | ObjectReference)>

<!ELEMENT standardRate (#PCDATA)>

<!ATTLIST standardRate
  currencyLocale CDATA #IMPLIED
  unit CDATA #IMPLIED
>

<!ELEMENT costPerUse (#PCDATA)>

<!ATTLIST costPerUse
  currencyLocale CDATA #IMPLIED
>

<!ELEMENT overtimeRate (#PCDATA)>

<!ATTLIST overtimeRate
  currencyLocale CDATA #IMPLIED

```

```

    unit CDATA #IMPLIED
  >

  <!--ELEMENT maxPercentUtilization (#PCDATA)-->

```

The following subelements are not supported:

- resourceCategory
- label
- ObjectReference
- currencyLocale
- unit

The following table provides additional information about some of the supported resourceInfo subelements:

Subelement	Description
maxUnits	Specifies the maximum units for material resources. This element is only valid if the resourceType is equal to MATERIAL and is an integer.
resourceReference	Name of the resource as defined in the userInfo element.  ID of the resource. For details on ObjectID, see <a href="#">ObjectID Element</a> .
standardRate	Specifies a value for the resource's standard rate. The content is a decimal number representing the standard rate. For time resources (person, equipment and facility), this is the standard cost per hour. For material resources, this is the cost per unit.
costPerUse	Specifies the cost that is accrued each time a resource is used. The content is a decimal representing the cost per use.
overtimeRate	Specifies a value for the resource's overtime rate. The content is a decimal number representing the standard rate. For time resources (person, equipment and facility), this is the overtime cost per hour.  This element is not supported for material resources.

Subelement	Description
maxPercentUtilization	<p>Specifies the maximum allocation percentage for time-based resources such as person, equipment, and facility resources. This element is only valid if the resourceType is PERSON, EQUIPMENT, or FACILITY.</p> <p>The content of this element is an integer representing the maximum percent allocation.</p>

## resourceLink Element

The resourceLink element assigns resources to activities.

The resourceLink element is defined by the following DTD elements:

```

<!ELEMENT resourceLink (resourceHolderID,
                        resourceID,
                        totalResources?)>

<!ELEMENT resourceHolderID (ObjectID)>

<!ELEMENT resourceID (ObjectID)>

<!ELEMENT totalResources (#PCDATA)>

```

The following table provides additional information about the resourceLink subelements:

Subelement	Description
resourceHolderID	ID of the activity. For details on ObjectID, see <a href="#">ObjectID Element</a> .
resourceID	ID of the resource. For details on ObjectID, see <a href="#">ObjectID Element</a> .
totalResources	Specifies how much of the resource is allocated to the activity. For Person, Equipment and Facility resources this is a percentage allocation. For Material resources this is the number of units allocated to the activity.

The following example shows the use of resourceLink element:

```
<resourceLink>
  <resourceHolderID>
    <ObjectID>
      <localId>wt.projmgmt.execution.ProjectActivity:106103</localId>
    </ObjectID>
  </resourceHolderID>
  <resourceID>
    <ObjectID>
      <localId>wt.projmgmt.resource.PersonResource:106029</localId>
    </ObjectID>
  </resourceID>
  <totalResources>50</totalResources>
</resourceLink>
```

## duration Element

The duration element specifies the duration of a plan object. The duration can be specified in one of four different units: days, hours, seconds or milliseconds using the unit attribute.

The duration element is defined by the following DTD elements:

```
<!ELEMENT duration (#PCDATA)>
<!ATTLIST duration
  unit (day | hour | second | millisecond) #IMPLIED
>
```

## ProjectEnumType Element

ProjectEnumType is an optional element of ProjectConfig and is used to define the project roles for a specific type of project. It is also used within elements other than ProjectConfig.

For the details on this element, see [ProjectEnumType Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## DiscussionForumTemplate Element

DiscussionForumTemplate is a element common to Windchill context templates but is only used only within the ProjectConfig element. It defines discussion forum templates for project containers.

The DiscussionForumTemplate element is defined by the following DTD elements:

```
<!ELEMENT DiscussionForumTemplate (name,
                                     description,
                                     xmlLobString)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT xmlLobString (#PCDATA)>
```



Forums have their own XML loader. On import, the handler extracts the `xmlLobString` element and uses the service to instantiate the forum. On export, the `xmlLobString` element is extracted.

The following example defines the My Forum Template:

```
<DiscussionForumTemplate>
  <name>My Forum Template</name>
  <description>Defines my forum topics</description>
  <xmlLobString>
    :
    :
  </xmlLobString>
</DiscussionForumTemplate>
```

When used within the `DiscussionForumTemplate` element, the `xmlLobString` element can be either a CDATA section or fully-escaped XML that names topics for the discussion forum template being defined. The following `xmlLobString` example uses fully-escaped XML to name the General, Documents, and Parts topics:

```
<xmlLobString>
  &lt;Structure&gt;
    &lt;Topic&gt;
      &lt;Name&gt;General&lt;/Name&gt;
    &lt;/Topic&gt;
    &lt;Topic&gt;
      &lt;Name&gt;Documents&lt;/Name&gt;
    &lt;/Topic&gt;
    &lt;Topic&gt;
      &lt;Name&gt;Parts&lt;/Name&gt;
    &lt;/Topic&gt;
  &lt;/Structure&gt;
</xmlLobString>
```

**Note:** Instead of fully-escaped XML, you can use a CDATA section as is shown in the `NotebookTemplate` element example.

## NotebookTemplate Element

`NotebookTemplate` is a element common to Windchill context templates but is only used only within the `ProjectConfig` element. It defines notebook templates for project containers.

The `NotebookTemplate` element is defined by the following DTD elements:

```
<!ELEMENT NotebookTemplate (name, description, xmlLobString)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT xmlLobString (#PCDATA)>
```

Notebooks have their own XML loader. On import, the handler extracts the `xmlLobString` element and uses the service to instantiate the notebook. On export, the `xmlLobString` element is extracted.

The following example defines the My Notebook Template:

```
<NotebookTemplate>
  <name>My Notebook Template</name>
  <description>Defines my notebook folders</description>
  <xmlLobString>
    :
    :
  </xmlLobString>
</NotebookTemplate>
```

When used within the NotebookTemplate element, the xmlLobString element can be either a CDATA section or fully-escaped XML that defines notebook folders and optional bookmarks under each folder. The following xmlLobString example uses the CDATA section to define the General, Parts, Documents, and Links folders. Under the Links folder, the PTC bookmark is defined.

```
<xmlLobString><![CDATA[
  <Notebook>
    <folder>
      <name>General</name>
    </folder>
    <folder>
      <name>Parts</name>
    </folder>
    <folder>
      <name>Documents</name>
    </folder>
    <folder>
      <name>Links</name>
      <bookmark>
        <name>PTC</name>
        <url>http://www.ptc.com</url>
      </bookmark>
    </folder>
  </Notebook>]]>
</xmlLobString>
```

**Note:** Instead of the CDATA section, you can use fully-escaped XML as is shown in the DiscussionForumTemplate element example.

# Common XML Elements Used in Multiple Business XML Files

The following sections describe the XML elements that are used in more than one type of business XML file.

## OrgStructure Element

OrgStructure is an element common to Windchill context templates that has subelements for defining domains, folders, groups and access control policy rules. OrgStructure is defined by the following DTD element:

```
<!ELEMENT OrgStructure (DomainStructure?,  
                        FolderStructure?,  
                        (OrgGroup?, ACLRule*)*)>
```

The following sections describe the subelements in OrgStructure.

## DomainStructure Element

DomainStructure is an optional element of OrgStructure that defines a domain hierarchy for a container. This section provides the syntax of the DomainStructure element. For a detailed explanation of domains and the out-of-the box Site domains that are created, see [Administering Domains and Policies](#) in the [Administering Containers](#) chapter.

The DomainStructure element is defined by the following DTD elements:

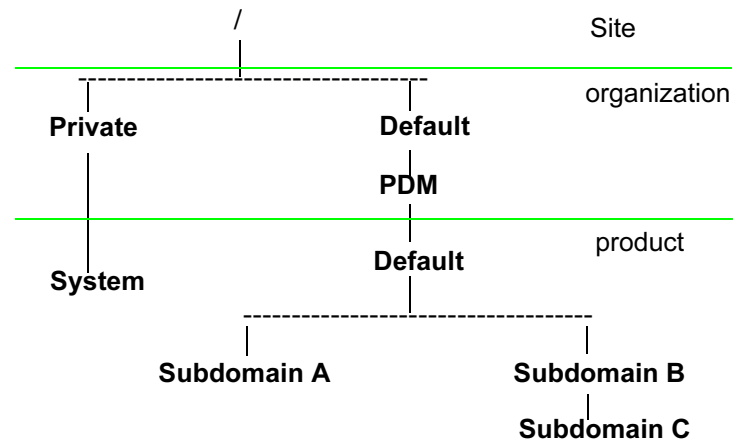
```
<!ELEMENT DomainStructure (domainLevel*)>  
<!ELEMENT domainLevel (parentDomain?,  
                        domainName,  
                        description?,  
                        domainLevel*)>
```

The following table provides additional information about the subelements in the DTD elements for domainLevel:

Subelement	Description
parentDomain	<p>Specifies the domain to use as the parent domain for the domain being defined in the domainLevel element. The value is a domain path that is relative to the container in which the parent domain resides. When defining a domain that is a top-level domain in a container, the parent domain must be in the parent container of the container being defined in the template. For example, if the container being defined through the template is a product container and a top-level domain is being defined, then the parent domain must be a domain in an organization container.</p> <p>Always start the path with the forward slash (/) and use the forward slash to separate the names of the domains in the hierarchy relative to the container.</p> <p>When defining the top-level domains in a container, always specify the parentDomain element.</p> <p><b>Note:</b> For backward compatibility, if parentDomain is not specified in the outermost domainLevel subelement of the DomainStructure element, then the parent domain is assumed to be the /Default domain in the container being defined.</p> <p>In nested domainLevel elements, omit the parentDomain element. When the parentDomain element is not specified in a nested domainLevel element, then the parent domain is derived using the domainName and parentDomain elements specified in the enclosing domainLevel element.</p> <p>If parentDomain is specified in a nested domainLevel subelement, then the domainLevel element is treated as if it was not in a nested structure, meaning that the included parentDomain element determines the parent domain.</p> <p>See the examples provided later in this section for suggested uses of this element.</p>
domainName	Specifies the name of the domain.
description	Specifies the description of the domain. You can enter a maximum of 200 characters in the description.

Subelement	Description
domainLevel	Allows you to nest domains.

The following example illustrates the domain structure of an exported product template where three subdomains have been included in addition to the /Default and /System domains normally defined in a product container. Specifically, it creates subdomains under the /Default domain in a product container as follows:



The following example DomainStructure element shows the domain structure of an exported product template with the new subdomains in the container as illustrated previously:

```

<DomainStructure>
  <domainLevel>
    <parentDomain>/Private</parentDomain>
    <domainName>System</domainName>
  </domainLevel>

  <domainLevel>
    <parentDomain>/Default/PDM</parentDomain>
    <domainName>Default</domainName>

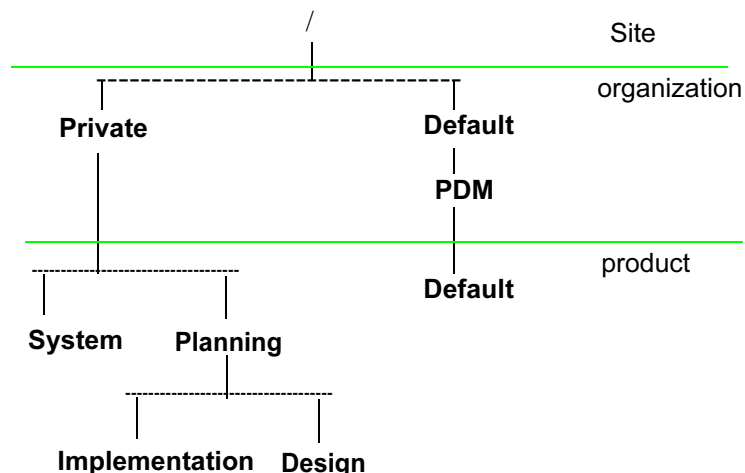
    <domainLevel>
      <domainName>Subdomain A</domainName>
    </domainLevel>

    <domainLevel>
      <domainName>Subdomain B</domainName>
      <domainLevel>
        <domainName>Subdomain C</domainName>
      </domainLevel>
    </domainLevel>
  </DomainStructure>

```

In this example, two top-level domains are defined for the product container: /System and /Default. Then under the /Default domain are subdomains named /Subdomain A and /Subdomain B. Under /Subdomain B is /Subdomain C.

The following example illustrates the addition of a top-level domain named /Planning in a product container and the addition of two domains named /Implementation and /Design that are nested under the /Planning domain. Specifically, it creates the top-level /Planning domain whose parent domain is the /Private domain that is in the organization container. Under the /Planning domain, there are two subdomains named /Implementation and /Design as follows:



The following example DomainStructure element shows the domain structure of an exported product template with the new /Planning top-level domain in the container as illustrated previously:

```

<DomainStructure>
  <domainLevel>
    <parentDomain>/Private</parentDomain>
    <domainName>System</domainName>
  </domainLevel>
  <domainLevel>
    <parentDomain>/Private</parentDomain>
    <domainName>Planning</domainName>
    <domainLevel>
      <domainName>Implementation</domainName>
    </domainLevel>
    <domainLevel>
      <domainName>Design</domainName>
    </domainLevel>
  </domainLevel>
  <domainLevel>
    <parentDomain>/Default/PDM</parentDomain>
    <domainName>Default</domainName>
  </domainLevel>
</DomainStructure>

```

In this example, three top-level domains are defined for the product container: /System, /Planning, and /Default. Then under the /Planning domain are subdomains named /Implementation and /Design.

## FolderStructure Element

FolderStructure is an optional element of OrgStructure that defines the initial folder structure defined in a container. Folders are available to organize information.

Additionally, you can specify ad hoc access control rules for folders using this element. For Windchill PDMLink, PTC recommends that you use a domain and access control policy rules for that domain instead of ad hoc rules. For details, see [Assigning Domains to Folders in Windchill PDMLink or Windchill Foundation & PDM](#) in the Administering Containers chapter.

FolderStructure is defined by the following DTD elements:

```
<!ELEMENT FolderStructure (cabinet,
                           domainName?,
                           rootPath,
                           (subFolder | nestedFolder | folderLink |
                            folderContentObjectReference)*,
                           folderAcl*)>
<!ELEMENT nestedFolder (name, domainName?, nestedFolder*)>
<!ELEMENT folderLink (domainName?,
                      folderPath,
                      name,
                      urlString,
                      description?)>
<!ELEMENT folderAcl (subFolder, Role, applicationKey? , aclPermissionList)>
<!ELEMENT folderContentObjectReference (folderPath,
                                         ((referenceName,referenceNumber) | ObjectID))>
<!ELEMENT ObjectID (localId?, objectId?, ufid?)>

<!ATTLIST aclPermissionList

        delete (true | false) #IMPLIED
        create (true | false) #IMPLIED
        modify (true | false) #IMPLIED
        read (true | false) #IMPLIED
        attach (true | false) #IMPLIED
        administrative (true | false) #IMPLIED
        use (true | false) #IMPLIED
        ALL (true | false) #IMPLIED>
<!ELEMENT revisePerm EMPTY>
<!ATTLIST revisePerm
        enabled (true | false) #REQUIRED>
<!ELEMENT newViewVersionPerm EMPTY>
<!ATTLIST newViewVersionPerm
        enabled (true | false) #REQUIRED>
<!ELEMENT customPerm EMPTY>
<!ATTLIST customPerm
        customMask CDATA #REQUIRED
        enabled (true | false) #REQUIRED>
<!-- customMask must be an integer less than 63 -->
```

```

<!--
    significant customMask values for 7.0 are
    REVISE=7
    NEW_VIEW_VERSION=8
    CHANGE_PERMISSIONS=9
-->

```

The following table provides additional information about some of the subelements in the DTD elements for FolderStructure:

Subelement	Description
applicationKey	Not supported.
folderContentObjectReference	Not supported.
domainName	<p>Relative domain path, starting at a top-level domain in the container. Include the forward slash (/) at the beginning of the path.</p> <p><b>Note:</b> The format of this element is different from the format of the domainName subelement contained in the DomainStructure element.</p>
aclPermissionList	<p>List of permissions to grant.</p> <p><b>Note:</b> The following permissions are listed in the DTD, but are not supported:</p> <p>attach use</p>



The following example illustrates a folder structure that parallels the additional domains illustrated in the previous section:

**Note:** A unique domain is assigned to two of the folders. Using a unique domain allows special access control policy rules to be set for those folders. Also note that the nestedFolder element is used to define each folder. A subFolder element can not be used because that element does not include a subelement for associating a domain.

```
<FolderStructure>
  <cabinet>Default</cabinet>
  <rootPath>/</rootPath>

  <nestedFolder>
    <name>Folder A</name>
  </nestedFolder>

  <nestedFolder>
    <name>Folder B</name>
    <domainName>/Default/Subdomain B</domainName>
    <nestedFolder>
      <name>Folder C</name>
      <domainName>/Default/Subdomain B/Subdomain C</domainName>
    </nestedFolder>
  </nestedFolder>
</FolderStructure>
```

The following example illustrates a folder structure that defines the Specifications subfolder and ad hoc access control rules that apply to the subfolder:

```
<FolderStructure>
  <cabinet>/Default</cabinet>
  <domainName>/Default</domainName>
  <rootPath>/</rootPath>
  <nestedFolder>
    <name>Specifications</name>
    <domainName>/Default</domainName>
  </nestedFolder>
  <folderAcl>
    <subFolder>/Specifications</subFolder>
    <Role roleType="PROJECT MANAGER"></Role>
    <aclPermissionList ALL="true"></aclPermissionList>
  </folderAcl>
  <folderAcl>
    <subFolder>/Specifications</subFolder>
    <Role roleType="CONFIRMED"></Role>
    <aclPermissionList modify="true"
                        read="true"></aclPermissionList>
  </folderAcl>
  <folderAcl>
    <subFolder>/Specifications</subFolder>
    <Role roleType="GUEST"></Role>
    <aclPermissionList read="true"></aclPermissionList>
  </folderAcl>
</FolderStructure>
```

## OrgGroup Element

OrgGroup is an optional element of OrgStructure that defines the names of the groups defined in a container. The OrgGroup element can occur any number of times.

The OrgGroup element is defined by the following DTD element:

```
<!ELEMENT OrgGroup (groupName, description?)>
```

The following example illustrates the creation of some groups in a container.

```
<OrgGroup>
  <groupName>Group_1</groupName>
  <description>Description of Group_1</description>
</OrgGroup>

<OrgGroup>
  <groupName>Group_2</groupName>
  <description>Description of Group_2</description>
</OrgGroup>
```

## ACLRule Element

The ACLRule element is a common element that can be used as a subelement within multiple elements.

For ACLRule element details, see [ACLRule Element](#) under [Common XML Elements Used in Multiple Business XML Files](#).

## loadXMLFile Element

loadXMLFile is an element common to Windchill context templates that is used to include inline reading of other XML files.

Additional files that can be loaded are life cycle templates, workflow templates, and object initialization rules. The base location for these additional files is `<Windchill>/loadXMLFiles`. For example, to load the `part_lifecycle.xml` file that is in the `<Windchill>/loadXMLFiles/lifecycles` directory, include the following element:

```
<loadXMLFile>lifecycles/part_lifecycle.xml</loadXMLFile>
```

For assistance in using this element, contact PTC technical support.

## TypeBasedRule Element

TypeBasedRule is an element common to Windchill context templates that is used to define default values for objects when those objects are initialized or copied (object Initialization rules).

The TypeBasedRule element can occur any number of times and is defined by the following DTD elements:

```
<!ELEMENT TypeBasedRule (EngineRule, className)>

<!ELEMENT EngineRule (ruleName, ruleSpecification?, ruleType)>
<!ATTLIST EngineRule
    isDefault (true | false) "false"
    enabled (true | false) "true"
>

<!ELEMENT ruleType EMPTY>
<!ATTLIST ruleType
    type (INIT | COPY | TYPE) #REQUIRED
```

To specify an object initialization rule in the TypeBasedRule element, include the following elements:

```
<TypeBasedRule>
  <EngineRule isDefault="false" enabled="true">
    <ruleName>name_of_rule</ruleName>
    <ruleSpecification><![CDATA[
      :
      :
    ]]>
  </ruleSpecification>
  <ruleType type="INIT"></ruleType>
</EngineRule>
<className>object_type</className>
</TypeBasedRule>
```

In this set of elements, specify your own rule name in place of *name\_of\_rule* and specify the object type for the object in which the rule is being set in place of *object\_type*.

Also, place the actual rule after the following line:

```
<ruleSpecification><![CDATA[
```

For additional information on the rules that you can specify in the ruleSpecification element, see [Administering Object Initialization Rules](#) in the [Administering Containers](#) chapter.

The following example shows all XML elements that are required in an XML file used for importing a context template that sets the default document folder path for a document:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE TypeBasedRule SYSTEM "standardX05.dtd">
<!-- WTDocument -->
<TypeBasedRule>
  <EngineRule isDefault="false" enabled="true">
    <ruleName>Document Folder Configuration</ruleName>
    <ruleSpecification><![CDATA[
      <AttributeValues objType="wt.doc.WTDocument">
        <AttrValue id="folder.id" algorithm=
"com.ptc.core.foundation.folder.server.impl.FolderPathAttributeAlgorithm">
          <Arg>/Default</Arg>
        </AttrValue>
      </AttributeValues>
    ]]>
    </ruleSpecification>
    <ruleType type="INIT"></ruleType>
  </EngineRule>
<className>wt.doc.WTDocument</className>
</TypeBasedRule>
```

The following example illustrates setting initialization values for life cycles associated with Design and BOM documents:

```
<!-- Design -->
<TypeBasedRule>
  <EngineRule isDefault="false" enabled="true">
    <ruleName>DESIGN_RULE_NAME</ruleName>
    <ruleSpecification><![CDATA[
      <AttributeValues objType="wt.doc.WTDocument">
        <!-- set the life cycle -->
        <AttrValue id="lifeCycle.id"
          algorithm="com.ptc.core.foundation.lifecycle.server.impl.
            LifeCycleTemplateAttributeAlgorithm">
          <Arg>BASIC_LIFECYCLE_NAME</Arg>
        </AttrValue>
      </AttributeValues>]]>
    </ruleSpecification>
  </EngineRule>

  <!-- Value of DESIGN_TYPE_NAME cannot contain spaces -->
  <className>wt.doc.WTDocument|DESIGN_TYPE_NAME</className>
</TypeBasedRule>

<!-- Bill of Material (BOM) -->
<TypeBasedRule>
  <EngineRule isDefault="false" enabled="true">
    <ruleName>BOM_RULE_NAME</ruleName>
    <ruleSpecification><![CDATA[
      <AttributeValues objType="wt.doc.WTDocument">
        <!-- set the lifecycle -->
        <AttrValue id="lifeCycle.id"
          algorithm="com.ptc.core.foundation.lifecycle.server.impl.
            LifeCycleTemplateAttributeAlgorithm">
          <Arg>BASIC_LIFECYCLE_NAME</Arg>
        </AttrValue>
      </AttributeValues>]]>
    </ruleSpecification>
  </EngineRule>

  <!-- Value of BOM_TYPE_NAME cannot contain spaces -->
  <className>wt.doc.WTDocument|BOM_TYPE_NAME</className>
</TypeBasedRule>
```

## ExportedRoleMemberMap Element

ExportedRoleMemberMap is an element common to Windchill context templates that is used to define the roles (and optionally, their members) associated with the container.

ExportedRoleMemberMap is defined by the following DTD elements:

```
<!ELEMENT ExportedRoleMemberMap (roleAccess?, projectMember*)>

<!ELEMENT roleAccess (Role*)>

<!ELEMENT projectMember (Role?,
                        (memberUserId*, memberGroupId*)*)>

<!ELEMENT Role (customType?, uiComponent*)>

<!ATTLIST Role roleType (APPROVER | ASSIGNEE | AUDITOR |
                        AUTHOR | CHANGE_MANAGER | CHANGE_OWNER |
                        CHEMICAL_ENGINEER | CONSULTANT |
                        CONTRACTOR | CONTROLLER |
                        CUSTOMER | DESIGN_ENGINEER | DESIGNER |
                        DETAILER | ENGINEER | FIELD_ENGINEER |
                        GROUP_LEADER | HYDRAULIC_ENGINEER |
                        IMPLEMENTOR | INDUSTRIAL_ENGINEER |
                        LOGIC_ENGINEER | MANUFACTURER |
                        MANUFACTURING_ENGINEER |
                        MANUFACTURING_MANAGER |
                        MATERIAL_ENGINEER | MEMBERS |
                        NC_PROGRAMMER | OBSERVER | OEM |
                        ORGANIZATION_LEADER | OWNER | PLANNER |
                        PRODUCT_MANAGER | PRODUCTION_PLANNER |
                        PROGRAM_DIRECTOR | PROGRAM_MANAGER |
                        PROJECT_ACCOUNTANT |
                        PROJECT_ADMINISTRATOR |
                        PROJECT_ENGINEER | PROJECT_MANAGER |
                        PROJECT_SPONSOR | PROMOTER |
                        PURCHASING_AGENT | QUALITY_ENGINEER |
                        QUALITY_MANAGER | RESOURCE_MANAGER |
                        REVIEWER | SOFTWARE_ENGINEER |
                        SUBMITTER | SUGGESTOR | SUPPLIER |
                        SUPPORT_ENGINEER | TEAM_LEADER |
                        TESTER | TOOL_DESIGNER | VENDOR |
                        EXPORTER | IMPORTER | PUBLISHER |
                        SUBSCRIBER | CUSTOM) #REQUIRED>

<!ELEMENT uiComponent (#PCDATA)>
<!ATTLIST uiComponent
    name CDATA #REQUIRED
    value (true | false) #REQUIRED
>
```

The following table provides additional information about some of the subelements in the DTD elements for ExportedRoleMemberMap:

Subelement	Description
roleAccess	<p>Identifies roles for which access to specific actions has been configured. Configuring roles for Windchill ProjectLink is described in the <a href="#">Controlling the Visibility of Actions By Role</a> section of the Administering Projects chapter.</p> <p><b>Note:</b> This subelement is only supported in project templates. Additionally, the subelements contained in the roleAccess element can be present in an exported XML file for a project, but should not be changed when the exported project is imported.</p>
Role	<p>Associates a role with the parent container. A role is a function that can be performed by a user or group. A role is defined by the roleType attribute.</p>
memberUserId	<p>Associates a user with a role. Defines the user name to add as a member.</p>
memberGroupId	<p>Associates a group with a role. Defines the group name to add as a member.</p>
customType	<p>When roleType ="CUSTOM", the customType element specifies a custom role name.</p>

Subelement	Description
uiComponent	<p>When roles have been configured for access in a project that has been exported, this element contains the name of an action and a value. The value is true if the action is allowed and false if it is not allowed. Valid action names are maintained in an XML file and correspond to action that can be selected from the <b>Configure Roles</b> window.</p> <p><b>Note:</b> Adding or modifying existing uiComponent elements that are in an exported project template is not supported.</p>

The following example illustrates an ExportedRoleMemberMap element with one roleAccess subelement and several projectMember subelements. The roleAccess subelement configures the **Members** role so that users in this role do not have access to the **View Project Details** action. The first projectMember subelement has no associated members. The second projectMember subelement has one associated member group. The last projectMember subelement specifies a custom role.

```

<ExportedRoleMemberMap>
  <roleAccess>
    <Role roleType="MEMBERS">
      <uiComponent name="PROJECT_DETAIL" value="false">
      </uiComponent>
    </Role>
  </roleAccess>
  <projectMember>
    <Role roleType="ASSIGNEE"/>
  </projectMember>

  <projectMember>
    <Role roleType="AUDITOR"/>
    <memberGroupId>My Org Auditors</memberGroupId>
  </projectMember>

  <projectMember>
    <Role roleType="CUSTOM">
      <customType>MY_CUSTOM_PRODUCT_ROLE</customType>
    </Role>
  </projectMember>
</ExportedRoleMemberMap>

```



## ACLRule Element

ACLRule is an element common to Windchill context templates that is used to define access control policy rules for the objects within a container.

Access control policy rules are associated with domains. For a detailed explanation of access control policy rules and how they are associated with domains, see the [Administering Access Control](#) chapter.

ACLRule is defined by the following DTD elements:

```
<!ELEMENT ACLRule ((domainName, ownerDomain?, description?),
                    aclClassName,
                    aclOn,
                    aclPrincipalName,
                    aclPermissionList?,
                    lifecycleState?)>
<!ELEMENT aclPermissionList (revisePerm?,
                              newViewVersionPerm?,
                              customPerm*)>
<!ATTLIST aclPermissionList
delete (true | false) #IMPLIED
create (true | false) #IMPLIED
modify (true | false) #IMPLIED
read (true | false) #IMPLIED
attach (true | false) #IMPLIED
administrative (true | false) #IMPLIED
use (true | false) #IMPLIED
ALL (true | false) #IMPLIED>
<!ELEMENT revisePerm EMPTY>
<!ATTLIST revisePerm
enabled (true | false) #REQUIRED>
<!ELEMENT newViewVersionPerm EMPTY>
<!ATTLIST newViewVersionPerm
enabled (true | false) #REQUIRED>
<!ELEMENT customPerm EMPTY>
<!ATTLIST customPerm
customMask CDATA #REQUIRED
enabled (true | false) #REQUIRED>
<!-- customMask must be an integer less than 63 -->
<!--
significant customMask values for 7.0 are
REVISE=7
NEW_VIEW_VERSION=8
CHANGE_PERMISSIONS=9
-->
```

The following table provides additional information about some of the subelements in the DTD elements for ACLRule:

Subelement	Description
domainName	Domain path relative to the container.
ownerDomain	Not supported.

Subelement	Description
description	Not supported.
aclClassName	A persisted type, logical type identifier, or external type identifier
aclOn	Specifying true grants the permission; specifying false denies the permission
aclPrincipalName	Name of a group or organization principal (cannot be a user). The name must be unique within the container hierarchy.
lifecycleState	State name; either "ALL" or a key defined in StateRB.rbInfo (wt.lifecycle package). If omitted, defaults to "ALL".
aclPermissionList	<p>List of permissions to grant or deny. The permissions correspond to keys defined in AccessPermissionRB.rbInfo (wt.access package). At least one permission must be specified.</p> <p><b>Note:</b> The following permissions are listed in the DTD, but are not supported:</p> <p>attach use</p>

**Note:** The domain, type, and principal specified by domainName, aclClassName, and aclPrincipalName respectively must exist.

The following illustrates the out-of-box access control policy rules for a part:

```
<ACLRule>
  <domainName>/Default</domainName>
  <aclClassName>wt.part.WTPart</aclClassName>
  <aclOn>true</aclOn>
  <aclPrincipalName>CONFIRMED</aclPrincipalName>
  <aclPermissionList read="true"/>
</ACLRule>

<ACLRule>
  <domainName>/Default</domainName>
  <aclClassName>wt.part.WTPart</aclClassName>
  <aclOn>true</aclOn>
  <aclPrincipalName>CONFIRMED</aclPrincipalName>
  <aclPermissionList modify="true"
    create="true"
    delete="true"/>
  <lifecycleState>INWORK</lifecycleState>
</ACLRule>

<ACLRule>
  <domainName>/Default</domainName>
  <aclClassName>wt.part.WTPart</aclClassName>
  <aclOn>true</aclOn>
  <aclPrincipalName>CONFIRMED</aclPrincipalName>
  <aclPermissionList>
    <revisePerm enabled="true"/>
    <newViewVersionPerm enabled="true"/>
  </aclPermissionList>
  <lifecycleState>RELEASED</lifecycleState>
</ACLRule>
```

## ProjectEnumType Element

**Note:** ProjectEnumType is only applicable when Windchill ProjectLink is installed.

ProjectEnumType is an element common to Windchill context templates that is used to define the project roles. If this element is set in both OrganizationConfig and ProjectConfig, then the roles are combined.

Any number of ProjectEnumType elements can be specified.

ProjectEnumType is defined by the following DTD elements:

```
<!ELEMENT ProjectEnumType (typeName, projectId?, typeValues)>

<!ATTLIST ProjectEnumType
  excludeByDefault (true | false) "true"
>
<!ELEMENT projectId (ObjectID)>
<!ELEMENT typeValues (typeEntry*)>
<!ELEMENT typeEntry (#PCDATA)>
```

The following table provides additional information about some of the subelements in the DTD elements for ProjectEnumType:

Subelement	Description
projectId	Object identifier of the project. For details on ObjectID, see <a href="#">ObjectID Element</a> .
excludeByDefault	<p>Defines what to do if the current typeName has no excluded values specified yet in the target container. If there are excluded values already present for the type in the target container, then this value is ignored; otherwise, a value of true tells the system to exclude any static types not specified in the list.</p> <p>A value of false tells the system that no static values are to be included. The net effect is that if any static values, (a static value is one defined in the rbInfo file for the enumerated type), are added to this tag they will be ignored.</p>

The following example illustrates ProjectEnumType:

```
<ProjectEnumType>
<typeName>wt.project.Role</typeName>
<typeValues>
  <typeEntry>APPROVER</typeEntry>
  <typeEntry>CONSULTANT</typeEntry>
  <typeEntry>CONTRACTOR</typeEntry>
  <typeEntry>DESIGNER</typeEntry>
  <typeEntry>ENGINEER</typeEntry>
  <typeEntry>GROUP LEADER</typeEntry>
  <typeEntry>MANUFACTURER</typeEntry>
  <typeEntry>MEMBERS</typeEntry>
  <typeEntry>OBSERVER</typeEntry>
  <typeEntry>OEM</typeEntry>
  <typeEntry>ORGANIZATION LEADER</typeEntry>
  <typeEntry>PRODUCT MANAGER</typeEntry>
  <typeEntry>PROGRAM DIRECTOR</typeEntry>
  <typeEntry>PROGRAM MANAGER</typeEntry>
  <typeEntry>PROJECT ACCOUNTANT</typeEntry>
  <typeEntry>PROJECT ADMINISTRATOR</typeEntry>
  <typeEntry>PROJECT MANAGER</typeEntry>
  <typeEntry>PROJECT SPONSOR</typeEntry>
  <typeEntry>RESOURCE MANAGER</typeEntry>
  <typeEntry>REVIEWER</typeEntry>
  <typeEntry>SUPPLIER</typeEntry>
  <typeEntry>TEAM LEADER</typeEntry>
</typeValues>
</ProjectEnumType>
```



# Index

## A

- Access control
  - cabinets, 10-18
  - content holder objects, 10-15
  - deleted groups, 9-16
  - deleted organizations, 9-19
  - deleted users, 9-12
  - domains, 10-2
  - life cycles and, 14-36, 14-45
  - overview, 10-2
  - Process Manager, 16-32
  - Product Design template, 14-17
- Access control lists, 10-2, 10-6–10-11
  - ad hoc, 10-2, 10-11, 10-19, 14-36, 14-45
  - defining, 10-2
  - derived from access control policies, 10-7–10-8
  - object types, 10-2
  - permissions in, 10-8–10-9
  - policy, 10-2, 10-20, 14-36, 14-45
  - types of, 10-2
- Access control policies
  - defined, 10-7
  - defining, 4-8
  - deriving access control lists from, 10-7–10-8
  - organizations, 5-5
- Access control rules
  - about, 10-6
  - creating, 10-4
  - Default domain, 3-13
  - default, for cabinets, 10-19
  - defined, 10-3
  - domains, 10-10
  - examples, 10-2, 10-10–10-11
  - introduction, 2-8
  - permissions in, 10-8–10-9
  - predefined, 10-12
  - root domain, 3-11
  - System domain, 3-14
  - User domain, 3-13
  - User/Unaffiliated domain, 3-13
  - Windchill PDMLink, 6-14, 6-32
  - window, 10-5
- Access Control Rules window, 10-4
- ACL. See Access control list
- Action items, 7-5
- Activities, 7-5
  - creating, 7-5
  - managing, 7-5
  - updating, 7-5
- Actors, 15-2
- Ad hoc ACLs. See Access control lists, ad hoc
- Administrative Domain window, 10-4, 12-3
- Administrative notifications, 9-20
- Administrative permission, 10-5
- Administrator user, 1-15, 2-6
- Administrators
  - access to objects, 2-6
  - group, 2-6
  - selecting users, 1-15
  - Windchill PDMLink, 1-13
  - Windchill ProjectLink, 1-14
- All permission, 10-5
- ALL pseudo-group, 10-7
- Application containers, 3-2, 3-5
- Arbortext
  - dynamic documents, 11-31
  - folders, 6-45
  - introduction, 2-19
  - products, 6-2
  - publishing, 2-18
  - soft attributes, 11-33
  - soft types, 11-32
- archiving, 4-11, 5-7
- ASME Y 14.35M Standard, 3-62
- Assigned activities. See Workflow activities
- Attribute Definition Manager, 11-37
- Attributes
  - Arbortext, 11-33
  - constraint types, 11-21
  - creating, 11-12
  - defining, 11-7
  - dynamic documents, 11-33
  - updating, 11-19
- Audience, xlv
- Audit reports, 19-2
- Auditing
  - custom reports, 6-6

- organization, 5-5
- system information, 4-8

## B

- Background queues, 2-3
- baseline.ManagedBaseline, 2-17
- Bootstrap client, 2-3
- Burst configuration files, 2-16
- Bursting configuration files, 2-17
- Business XML files, 3-28, A-1

## C

- Cabinets
  - access control, 10-15
  - default access control rules, 10-19
  - owners, 10-16
  - personal, 10-19
    - defined, 10-16
    - deleted users, 9-13
  - shared, 10-23
    - defined, 10-16
- Caching
  - of access control lists, 10-7, 10-10
  - of notification lists, 13-4
- CAD agent, 18-5
  - configuring, 18-9
  - debugging mode, 18-10
  - timeouts, 18-18
- CAD document templates, 3-16
- CAD documents
  - object type, 2-16, 11-29
  - publishing
    - automating, 18-19
    - manual, 18-16
  - templates, 2-18
  - type, 2-18
  - visualization, 2-18
- CAD workers, 18-5, 18-9
  - configuring, 4-10
  - starting
    - from the CAD agent, 18-13
    - from the CAD agent monitor, 18-14
  - starting, manually, 18-11
  - stopping, 18-12
- Change Activity
  - team roles, 15-4
  - workflow, 16-43
- Change Admin roles, 6-13, 6-28, 6-31, 6-39
- Change Management objects, 15-4
- Change Monitor reports, 6-6, 6-44

- Change Notice workflow, 16-45
- Change objects, 10-15, 11-26, 11-35
- Change Permissions permission, 10-5
- Change Request workflow, 16-44
- Change transition rule, 14-6, 14-11
- Code examples, object subscription, 16-38
- Collapse tables, 4-18, 5-20
- Collections
  - defined, 12-2
  - deleted users, 9-13
- Composite object initialization rules, 3-42
- Configuring
  - options, 4-5
  - worklist fields, 16-38
- CONFIRMED group, 6-8, 6-9, 6-14, 6-16, 6-32, 6-33
- Container configuration
  - description, 3-4
  - updating, 3-5
- Container data
  - description, 3-14
  - installation, 3-15
  - organizations, 5-7
  - site, 4-15
  - updating, 3-15
- Container environment
  - projects, 7-8
- Container object initialization rules
  - accessing, 3-41
  - description, 3-19
  - installation, 3-19
  - loading, 3-40
  - site, 4-15
  - specifying, 3-44
  - updating, 3-40
  - XML, 3-45
- Container participation
  - description, 3-8
  - installation, 3-8
  - organization, 5-7
  - projects, 7-8
  - site, 4-15
  - updating, 3-9
- Container policies
  - description, 3-10
  - installation, 3-10
  - organization, 5-7
  - site, 4-15
  - updating, 3-14
- Container structure
  - description, 3-6
  - installation, 3-6
  - organizations, 5-7



- projects, 7-8
  - site, 4-15
  - updating, 3-7
- Container templates
  - description, 3-16
  - installation, 3-17
  - site, 4-15
  - updating, 3-18
- Containers
  - creating, 3-21
  - hierarchy, 3-2
    - Windchill PDMLink, 2-7
    - Windchill ProjectLink, 2-7
  - libraries, 6-2, 6-41
  - organization, 3-4, 4-3
  - overview, 2-3
  - products, 6-2, 6-40
  - Site, 3-2, 3-4
  - Windchill PDM, 2-5
- Content holder objects, 10-15
  - access control, 10-15
- Content replication
  - projects, 7-7
  - system, 2-3
  - Windchill PDM, 8-5
- Context
  - definition, 2-5
  - domains, 3-36
  - teams, 2-14
  - templates, 2-6, 3-16
  - use, 3-3
- Create permission, 10-4
- Create Project window, 7-10
- Create Team window
  - Windchill PDMLink, 15-18
- Creating
  - access control rules, 10-4
  - action items, 7-5
  - activities, 7-5
  - attributes, 11-12
  - containers, 3-21
  - context templates, 3-23
  - deliverables, 7-5
  - documents and folders, 7-4
  - groups, 9-14
  - indexing rules, 12-3–12-4
  - life cycle templates, 14-19
  - notification rules, 13-2
  - organization folders and documents, 5-4
  - organizations, 1-9, 4-3, 5-13, 9-18
  - project space, 7-2
  - projects, 7-10

- resources, 7-5
  - roles, 15-17
  - site folders and documents, 4-5
  - types, 11-12
  - users, 9-10
  - views and view associations, 17-2
  - workflow
    - activities, 16-14
    - process links, 16-28
    - subprocesses, 16-26
- Creator
  - actor, 15-2
  - product and library, 1-13, 5-3, 5-8
  - project, 1-14, 5-8
- Custom reports, 6-6, 6-44

## D

- Data types, 2-17
- Default domain, 3-13, 3-32
- Default type, 11-25
- DefaultEPMDocument, 2-18, 11-29
- Defining, 15-17
- Delete permission, 10-5
- Deleting
  - groups, 9-14, 9-16
  - organizations, 9-18, 9-19
  - users, 9-10, 9-12
- Deliverable icon, 4-6
- Deliverables, 7-5
- Disconnected principals, 9-22
- Discuss icon, 4-5
- Document Library template, 6-9
- Document templates, 3-16, 3-18
- Documents
  - content holder, 10-15
  - organization, 5-4
- Domains
  - access control rules, 10-10, 10-14, 10-17
  - administering, 3-28
  - assign to folders, 3-38
  - creating, 3-33
  - definition, 3-28
  - display, 10-16
  - indexing rules and, 12-3, 13-3
  - installed, 3-29
  - introduction, 1-8
  - naming, 3-33
  - overview, 2-3, 3-34
  - Root, 3-32, 3-33, 3-52
  - Site container, 3-32
  - subfolders, 10-17

- System, 3-32, 3-52
- DTD
  - importing, 3-27
  - library elements, A-8
  - object initialization rules, 3-44
  - organization elements, A-2
  - product elements, A-6
  - project elements, A-10
- Dynamic documents
  - attributes, 11-33
  - folders, 6-45
  - introduction, 2-19
  - managing, 6-45
  - numbering scheme, 6-46
  - soft type, 2-16, 2-17, 2-19, 11-31

## E

- ED files, 18-2
- EDZ files, 18-2, 18-4
- Electronic signatures, 9-10, 16-58
- E-mail notifications. See Notifications
- Enabling audit reports, 19-2
- Enterprise directory, 1-2
- Enumerated types, 15-19
- EPMDocument, 2-16, 2-18, 18-4
- ERP system, working with, 4-11
- Example audit report, 19-4
- execution flags, 16-48
- Execution Options, 16-23
- Exporting
  - information, 4-10
  - life cycles, 14-42
  - templates, 3-25
  - watermarks, 18-22
  - workflow process templates, 16-30
  - ZIP file contents, 3-26
- External file vaults, 2-3

## F

- File vaults, 2-3
- File-based versioning, 3-55
- Folder browser, 4-5
- Folder members, 10-16, 10-24
- Folder paths, 3-19
- Foldered objects. See Folder members
- Folders, 7-4, 10-16
  - assigning domains, 3-38
  - contents view, 4-7
  - dynamic documents, 6-45
  - organization, 5-4

- subscription, 13-5
- table, 4-6
- Full name, 15-20

## G

- Gates. See Life cycle gates
- General Library template, 6-9, 6-31
- General Product template, 6-9, 6-31
- Generating thumbnails, 18-9
- Groups
  - deleting, 9-16
  - managing, 9-14
  - organization, 5-3
  - setting permissions, 10-4
- Guests role, 6-8, 6-9, 6-13, 6-18, 6-31, 6-36

## H

- Harvard series, 3-52
- Help
  - customizing, 2-3
- Home page
  - Windchill Foundation & PDM, 1-4
  - Windchill PDMLink, 1-5
  - Windchill ProjectLink, 1-7
- Home tab, 1-5, 1-7

## I

- Import and export, 4-10, 5-6
  - between projects, 7-5
- Importing
  - information, 4-10
  - life cycles, 14-42
  - Microsoft Project plan, 7-6
  - templates, 3-25
  - workflow process templates, 16-30
  - ZIP file contents, 3-27
- importList.txt file, 3-26
- Indexing rules
  - combinations, 12-5
  - creating, 12-3–12-4
  - object types, 12-4
  - root domain, 3-14
  - See also Collections
- Integral Windchill solutions, 2-8
- INVITED group, 6-14, 6-32
- Iterated objects, 14-18

## J

- JAR files, 14-43

java.security.acl package, 10-6, 10-8

## L

### Libraries

- creating in Windchill PDMLink, 6-41
- templates, 6-8, A-8
- Windchill PDM, 8-1
- Windchill PDMLink, 3-2, 6-1

Library creators, 1-13, 5-3, 5-8

### Library domains

- access control, 6-14, 6-32
- teams, 15-9

Library manager, 1-13, 6-31, 6-38

Library tab, 1-5

Library templates, 3-16, 6-9, A-8

LibraryConfig element, 3-28, A-8

### Life cycle access control rules

- defining, 14-36
- overview, 10-19

Life Cycle Administrator page, 14-18, 14-19, 14-20, 14-43

### Life cycle gates

- associating workflow processes, 14-32, 14-38
- overview, 14-3

### Life cycle phases, 14-22

- access control, 14-29
- associating workflow processes, 14-32, 14-38
- defining, 14-24
- properties, 14-28

### Life cycle promotion

- defining criteria, 14-32, 14-40

### Life cycle roles, 10-20, 14-29, 15-12, 16-17

- assigning permission, 14-38
- mappings, 14-34
- predefined, 14-33
- selecting, when defining phases, 14-32

### Life cycle states

- indexing, 12-2
- predefined, 14-42

### Life cycle templates

- advanced, 14-18
- basic, 14-18
- checking out, 14-18
- container, 3-16
- creating, 14-19
- installation, 3-17

Life cycle-managed information, 10-19, 14-19

### Life cycles, 14-1, 14-2

- access control, 10-19, 14-45
- default, 14-4, 14-23
- defined, 14-2

importing and exporting, 14-42

iterations, 14-18

properties, 14-22, 14-25

team roles, 15-10

team templates, 15-3

updating, 14-18

Lock transition rule, 14-11

Logging On, 1-3

### Logical identifier

- Arbortext, 11-33
- Windchill, 11-24

## M

Maintenance only release, 14-42, 16-30

Manage Layouts page, 16-38

### Managing

- action items, 7-5
- activities, 7-5
- deliverables, 7-5
- documents and folders, 7-4
- libraries, 6-1
- organization folders and documents, 5-4
- organizations, 4-3
- products, 6-1
- project roles, 7-3
- project team members, 7-3
- project templates, 7-5
- resources, 7-5
- site folders and documents, 4-5

### Members

- adding to a site, 4-4
- organization, 2-14

Members role, 6-13, 6-31

MilSpec, 3-62

MILSTD, 3-58, 3-62

Modify permission, 10-4

## N

New View Version permission, 10-5

Nodes. See Workflow processes, node types

Notification lists, 13-2, 13-4–13-5

Notification Policy Manager, 13-6

### Notification rules

- combinations of, 13-4
- creating, 13-2
- defined, 13-2
- object types, 13-4–13-5

Notifications, 13-1–13-6

- administrative, 9-20
- defined, 13-2

- deleted groups, 9-17
- deleted organizations, 9-19
- policies, defined, 13-2
- process, 13-6
- queue, 13-6
- See also Workflow notifications
- notify.properties file, 13-6
- NotifyPrincipalDisabled.xml, 9-20
- NotifyPrincipalRepair.xml, 9-20
- Null permission, 10-8
- Numbering schemes
  - configuring, 5-6
  - dynamic documents, 6-46
  - loading, 3-40
  - object initialization rules, 3-19
  - out of the box, 3-48
  - overview, 3-48
  - projects, 7-6
  - updating for projects, 7-6

## O

- o attribute, 1-13, 1-15, 2-6, 4-3
- Object initialization rules, 14-3
  - accessing, 3-41
  - description, 3-19
  - installation, 3-19
  - loading, 3-40
  - site installed, 3-19
  - specifying, 3-44
  - updating, 3-40
  - XML, 3-45, A-53
- Object subscription
  - code examples, 16-38
- Object types
  - access control, 10-7, 10-11, 10-24
  - business data, 2-16
  - comparison of Windchill Foundation & PDM and Windchill PDMLink, 11-8, 15-8
  - defined, 10-3, 12-3, 13-2
  - indexing rules, 12-4
  - life-cycle managed, 10-19
  - team templates, 15-7
- OL files, 18-2
- One Phase Development template, 6-11, 14-16
- Online
  - help, 2-3
  - tutorials, 2-3
- Optegra Gateway, 18-19
- Oracle database, 1-2
- Organization
  - administrators, 1-13, 1-14, 3-3, 5-2, 5-8
  - affiliation, 2-14
  - attribute, 1-13, 1-14, 2-6, 9-13
  - attributes, 4-3
  - containers, 1-8, 2-3, 3-4, 4-3
  - folders and documents, 5-4
  - groups, 5-3
  - members, 5-3
  - principals, 5-2
  - roles, 5-3
  - tab, 1-6, 1-8
  - templates, 1-11, 3-16, 3-17, 3-23, 5-7, A-2
- Organization ID, 4-4
- Organization Utilities page, 5-15
- OrganizationConfig element, 3-28, A-2
- Organizations
  - deleting, 9-19
  - managing, 9-18
  - viewing, 9-18
- Out-of-the-box
  - business object types, 2-16
  - container participation, 6-13, 6-31
  - numbering schemes, 3-48
  - translated context templates, 3-22
  - Windchill PDMLink library templates, 6-8
  - Windchill PDMLink product templates, 6-8
- Out-of-the-box templates
  - organization, 5-7
  - product and library, 6-8
  - project, 7-7
  - using, 3-22
- OWNER pseudo-user, 10-7

## P

- Part Library template, 6-9
- Participants
  - assigning
    - to life cycle roles, 14-34
    - to team roles, 15-20
    - to workflow activities, 16-17
  - defined, 15-2
  - searching, 3-63
  - window, 14-34
- PDM domain, 2-10
- PDMLink. See Windchill PDMLink
- PDMLinkOrgContainer.xml file, 1-6
- Permissions
  - access control lists, 10-8–10-9
  - ad hoc access control lists, 14-36
  - calculating, 10-11
  - defined, 10-3
  - examples, 10-10–10-11

- setting, 10-4–10-5
  - types of, 10-4
- See also Access control, Access control rules
- Personal cabinet names, 9-10
- PLT files, 18-2
- Policies
  - definition, 3-28
  - overview, 3-34
- Policy ACLs. See Access control lists, policy
- Policy Administrator, 3-28, 3-34
- Policy rules, 3-34, 3-37
- Preference Administrator, 14-8
- Principal Administrator, 4-4, 9-4
- Principal cache, 9-10, 9-15, 9-18, 9-20
- Principals
  - administering, 9-1
  - defined, 10-3, 13-3, 15-2
  - net permissions, 10-8–10-9
  - repair, 9-22
  - searching, 3-63
- Pro/ENGINEER Wildfire, 2-3
- Pro/INTRALINK Gateway, 18-19
- Problem Report workflow, 16-46
- Process Manager toolbar, 16-32
- Processes
  - notification, 13-6
- Product creators, 1-13, 5-3, 5-8
- Product Design template, 6-8, 6-10, 14-16
  - access control, 14-17
- Product domains
  - access control, 6-14, 6-32
  - teams and, 15-9
- Product manager, 1-13, 6-13, 6-27, 6-31, 6-38
- Product tab, 1-5
- ProductConfig element, 3-28, A-6
- Products
  - container, 3-2
  - templates, 3-16, 3-24, 6-8, A-6
  - Windchill Foundation & PDM, 8-1
  - Windchill PDMLink, 6-1, 6-40
- ProductView, 2-18, 18-2, 18-3
- ProductView Configuration Administrator
  - for projects, 7-7
- Project containers, domains, 2-12
- Project creators, 1-14, 5-3, 5-8
- Project managers, 1-14, 7-2
- Project roles
  - resolution, 15-10
- Project space
  - creating and updating, 7-2
- Project tab, 1-7
- Project team members, 7-3

- Project templates, 3-16, 3-24, 7-5, A-10
  - out-of-the-box, 7-7
- Project Utilities page, 7-6
- ProjectConfig element, 3-28, A-10
- ProjectLink. See Windchill ProjectLink
- Projects, 3-2, 7-2
- Promote transition rule, 14-6, 14-11
- Promotion Request
  - Approval Process, 14-10
  - Approval Process workflow, 16-47
  - Review Process, 14-10
  - Review Process workflow, 16-48
  - subfolder, 6-13
- Promotion. See Life cycle promotion
- Proxy processes. See Workflow subprocesses
- PTC documentation, xlviii
- Publish Monitor
  - for projects, 7-7
- Publish Scheduler, 18-19
- Publish Scheduler Administrator
  - for projects, 7-7
- Purge Manager, 4-11, 5-7
- Purging
  - groups, 9-15
  - jobs, 4-11
  - organizations, 9-18
  - users, 9-10

## Q

- Queues
  - notification, 13-6
  - visualization publish jobs, 18-5, 18-18
  - Windchill Schedule, 18-19
  - See also Windchill Queue Manager

## R

- Read permission, 10-4
- Related documentation, xlv
- Repair principals, 9-22
- Replication Administrator
  - for projects, 7-7
- Replication sites
  - configuring, 4-8
- Report Manager, 6-6
- Report templates, 3-16
- Resources, 7-5
- Restoring jobs, 4-11, 5-7
- Revise permission, 10-5
- Revise transition rule, 14-6, 14-11
- Robots. See Workflow processes, robot types

- RoleRB.rbinf resource file, 14-33, 14-42, 15-19
- Roles, 15-1
  - project, 7-3
- Root domain, 3-11, 3-32, 3-33, 3-52, 10-19
- Root view, 17-2
- Routing events, 16-21
- Routing icon, 4-6
- Rule antecedents
  - for access control lists, 10-6
  - for indexing, 12-2
  - for notifications, 13-2
- Rule consequents
  - for access control lists, 10-6
  - for indexing, 12-2
  - for notifications, 13-2
- Rules. See Access control rules, Notification rules
- Runtime Services, 2-3

## S

- Schedule Publish Job, 18-20
- Searching
  - deleted organizations, 9-19
  - deleted users, 9-13, 9-17
  - groups, 9-14
  - indexes, 12-2
  - organizations, 9-18
  - users, 9-10
- SessionIterationDomain domain, 3-32
- Set State transition rule, 14-6, 14-11
- Setting
  - default preferences, 4-18, 5-20
  - permissions, 10-4–10-5
- Site
  - administrators, 1-12, 1-13, 1-14, 3-3, 4-2
  - container, 2-3, 3-4, 3-21
  - container domains, 3-32
  - document templates, 3-18
  - tab, 1-6, 1-8
  - team templates, 3-18
  - workflow templates, 3-17
- Soft attributes. See Attributes
- Soft types. See Types
- Starting
  - CAD agent, 18-10
  - CAD workers, 18-11
  - Type Manager, 11-10
- State property, 14-28
- State-based versioning, 3-57
- StateRB.rbinf resource file, 14-42
- Stopping
  - CAD workers, 18-12
- Subfolders
  - defined, 10-17
  - domains, 10-17
  - moving, 10-17
- Subprocesses. See Workflow subprocesses
- System, 14-4
- System administrators, 1-12
- System domain, 3-14, 3-32, 3-52
- System information, auditing, 4-8

## T

- Tables
  - setting preferences to collapse, 4-18
- TAG-0.xml file, 3-26
- TAG-ContainerTemplate XML file, 3-26
- Team association, 15-8
  - rules, 15-9
- Team roles
  - creating, 15-17
  - defined, 15-2
  - products and libraries, 6-3
  - See also Life cycle roles, Participants
- Team templates, 15-2
  - Change Activity Team, 15-4
  - Change Control Board, 15-5
  - Change Request Team, 15-4
  - Change Team, 15-5
  - container, 3-16
  - Default, 15-5, 15-8
  - installation, 3-18
  - object initialization rules, 3-19
  - out-of-the-box, 15-3
    - Windchill PDMLink, 15-3
  - Problem Report Team, 15-4
- Teams, 15-1, 15-2
  - context, 2-14
  - creating, 15-17
  - roles resolution, 15-10
  - Windchill PDMLink, 6-42
  - See also Team templates
- Technical support, xlviii, 1-xlviii
- Templates
  - creating, 3-23
  - exporting, 3-25
  - importing, 3-25
  - library, 6-8, 6-31, A-8
  - object initialization rules, 3-47
  - organization, 1-11, 3-23, 5-7, A-2
  - product, 6-8, 6-31, A-6
  - product and library, 3-24
  - Product Design, 6-8

- project, 3-24, A-10
- removing, 3-21
- site-level, 4-7
- translated, 3-22
- Thumbnail generator, 18-3
- Thumbnail images, 18-4
- Transitions property, 14-28
- Transitions, for workflow activities, 16-22
- Translated context templates, 3-22
- Troubleshooting
  - Visualization services, 18-9
- Two Phase Development
  - state-based versioning, 3-59
  - template, 6-11, 14-16
- Type Manager
  - examples, 11-11
  - initial window, 11-11
  - using, 11-11
- Types
  - CAD documents, 11-29
  - change objects, 11-25
  - creating, 11-12
  - display lists, 11-25, 11-27
  - documents, 11-25
  - dynamic documents, 11-31
  - enumerated, 15-19
  - localizing soft types, 11-6
  - logical identifier, 11-24
  - organization, 5-4
  - parts, 11-25
  - searching, 11-39
  - site, 4-7
  - soft, 2-17, 11-2
  - specifying, 11-24
  - Type Manager, 11-7
  - updating, 11-19
  - use case, 11-11

## U

- UFID, 9-2, 9-7
- Unaffiliated domain, 2-15, 3-13, 3-32
- Unique Federation Identifier, 9-2
- Update Life Cycle window, 14-23, 14-30
- Updating
  - action items, 7-5
  - activities, 7-5
  - deliverables, 7-5
  - documents and folders, 7-4
  - groups, 9-14
  - life cycles, 14-18
  - members, 4-4

- organization folders and documents, 5-4
- organizations, 9-18
- project space, 7-2
- resources, 7-5
- users, 9-10
- workflow process templates, 16-3
- User domain, 2-14, 3-13, 3-32, 10-19
- User ID, 9-9
- User name, 9-9, 15-20
- Users
  - creating, 10-19
  - deleting, 9-12
  - managing, 2-6, 2-14, 9-9
  - notifying. See Notifications
  - objects, 9-3
  - organization affiliation, 2-14, 9-13
  - setting permissions, 10-4–10-5

## V

- Version Series property, 14-28
- Versioning schemes
  - administering, 3-50
  - configuring, 5-6
  - file-based, 3-55
  - Harvard series, 3-52
  - loading, 3-40
  - object initialization rules, 3-19
  - setup, 3-59
  - state-based, 3-57
- Versioning XML file, 3-60
- Viewable publishing, 5-6
- View-dependent objects
  - defined, 17-2
- Viewing
  - groups, 9-14
  - organizations, 9-18
  - users, 9-10
  - workflow history, 16-32
- Views and view associations, 17-1
  - command-line utility, 17-3
  - creating, 17-2
- Visualization collaboration interface, 18-21
- Visualization data model, 18-4
- Visualization service
  - See also Thumbnail generator, Thumbnail images
- Visualization services, 18-1
  - architecture, 18-3
  - file types, 18-2
  - loader, 18-3, 18-6
  - properties, 18-26
  - Publish Scheduler, 18-19

troubleshooting, 18-6

## W

Watermarks, 18-22

Windchill environment, 2-3

Windchill Foundation & PDM

getting started, 1-2

home page, 1-4

organization, 2-6

Windchill PDM

containers, 2-5

library, 3-2

library container, 2-9, 8-2

Windchill PDMLink

application containers, 2-10

container hierarchy, 2-7

demo data, 2-4

getting started, 1-2

home page, 1-5

templates, 3-24

Windchill ProjectLink

administrators, 1-14

container hierarchy, 2-7

getting started, 1-2

home page, 1-7

project containers, 2-12

templates, 3-24

Windchill Queue Manager, 18-19

Windchill Runtime Architecture, 2-2

Windchill URL, 1-3

Windchill Visualization Service, 2-18

Windows

Administrative Domain, 10-4

Create Project, 7-10

Create Team, 15-18

Participants, 14-34

Update Life Cycle, 14-23, 14-30

Work items

deleted groups, 9-17

Workflow activities

access control, 10-20

assigning participants, 16-17

creating, 16-14

deadlines, 16-18

declaring variables, 16-20

defining, 16-14

defining transitions, 16-22

routing, 16-20

Workflow Administrator page, 16-4, 16-31

Workflow history, 16-32

events, 16-33

Workflow History Viewer, 16-2, 16-34

Workflow notifications, 13-2

Workflow Process Editor, 16-2, 16-3

connector types, 16-27

navigating diagrams, 16-8

Workflow Process Manager, 14-40, 16-2

Workflow process templates

checking out, 16-3

deleted groups, 9-17

deleted users, 9-12

importing and exporting, 16-30

iterations, 16-3

updating, 16-3, 16-7

Workflow processes, 16-2

associating, with life cycle phases and gates, 14-32, 14-38

declaring variables, 16-12

defining connectors, 16-27

defining links, 16-28

deleted groups, 9-17

life cycles, 14-3

node types, 16-8

robot types, 16-9

Workflow routing. See Workflow activities, Routing events

Workflow runtime system, 16-2

Workflow states, 16-40

Workflow subprocesses

creating, 16-26

Workflow system

components of, 16-2

defined, 16-2

Workflow templates

container, 3-16

execution flags, 16-48

installation, 3-17

Workflows

Change Activity, 16-43

Change Notice, 16-45

Change Request, 16-44

Problem Report, 16-46

Promotion Request Approval Process, 16-47

Promotion Request Review Process, 16-48

Workgroup Manager CAD Document soft type, 2-18

Worklist fields

configuring, 16-38

wt.epm.EPMDocument, 2-18

wt.principal.cache.timeToLive property, 9-21

wt.principal.cache.timeToLiveRandomizer property, 9-21

wt.properties file, 12-5, 16-39

domain names, 3-32



- wt.typepickerexclude, 11-27
- wt.vc.views.LoadViews command-line utility, 17-3
- WTAnalysisActivity, 2-16
- WTChangeActivity2, 2-16
- WTChangeInvestigation, 2-16
- WTChangeIssue, 2-16
- WTChangeOrder2, 2-16
- WTChangeProposal, 2-16
- WTChangeRequest2, 2-16
- WTDocument, 2-16, 10-2, 10-7
- WTObject, 10-11, 10-23, 13-4
  - access control rules, 10-19
- WTPart, 2-16, 18-4, 18-5
- WTProductConfiguration, 2-17
- WTProductInstance2, 2-17
- wvs.properties file, 18-6, 18-21

## **X**

- xconfmanager, 2-26
- XML element (Arbortext), 11-32
- XML template files, A-1
- XML versioning file, 3-60

## **Z**

- ZIP files, 14-43
  - exporting, 3-26
  - life cycle, 14-43