



Deployment Guide Series: Tivoli IT Asset Management Portfolio



Prepare to install asset management products

Plan for a deployment

Bart Jacob Rajat Khungar Carlos J. Otálora Y. James Pittard TP Raghunathan David Stephenson

Redbooks

ibm.com/redbooks



International Technical Support Organization

Deployment Guide Series: Tivoli IT Asset Management Portfolio

July 2008

Note: Before using this information and the product it supports, read the information in "Notices" on page xiii.

First Edition (July 2008)

This edition applies to Tivoli Asset Management for IT Version 7.1, Tivoli License Compliance Manager Version 2.3, and Tivoli License Manager for z/OS Version 4.2.

© Copyright International Business Machines Corporation 2008. All rights reserved. Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Part

| | Figures | i |
|-----------|---|--------------|
| | Tablesx | i |
| | Notices | i v |
| | Preface xx The team that wrote this book xx Become a published author xvii Comments welcome xvii | / / ii |
| 1. Servic | e Management and Asset Management Overview | 1 |
| | Chapter 1. IBM Service Management platform 2 1.1 IBM Service Management 2 1.1.1 Why Do Businesses Need IT Service Management 2 1.1.2 What is IBM Service Management 2 1.2 Information Technology Infrastructure Library (ITIL) 2 1.2.1 ITIL Version 3 2 1.2.2 Critical Success Factors to Implement ITIL 1 1.3 IBM Tivoli Unified Process 12 1.3.1 ITUP Composer 14 | 311339121 |
| | Chapter 2. IBM Tivoli IT Asset Management portfolio 15 2.1 IT asset management 17 2.1.1 Challenges associated with IT asset management 17 2.1.2 Software asset management 27 2.2 Solution components 26 2.3 Implementing the solutions with Tivoli products 37 2.3.1 Tivoli Asset Management for IT product 37 2.3.2 Tivoli License Compliance Manager (TLCM) 34 2.3.4 Tivoli Asset Compliance Center (TACC) 35 | 5791311445 |

| | 2.4 Summary | . 35 |
|----------------|--|---|
| Part 2. Produc | ct Deployment | . 37 |
| | Chapter 3. Tivoli Asset Management for IT. 3.1 Management disciplines 3.2 Data loading and the Integration Composer 3.3 Installation overview 3.3.1 Installation flowchart 3.4 Middleware installation 3.4.1 Installing middleware. 3.4.2 Installing middleware. 3.5 Installing the Tivoli's process automation engine for Asset Management | . 39 . 42 . 44 . 45 . 45 . 48 . 48 . 49 for IT |
| | 3.5.1 Installation steps . 3.6 Tivoli Asset Management for IT solution package installation. 3.6.1 Process solution package overview 3.6.2 Pre-process solution package installation checklist 3.6.3 Enable Tivoli Asset Management for IT License. 3.7 Post Installation Tasks 3.7.1 Ensure Tivoli Asset Management for IT services are running. 3.7.2 Ensure Tivoli Asset Management for IT application connectivity . 3.7.3 Verify the language pack installation. 3.7.4 Report administration | . 78 104 105 106 113 114 115 115 115 |
| | Chapter 4. Tivoli License Compliance Manager.4.1 IBM Tivoli License Compliance Manager4.2 Planning4.3 Three-tier architecture.4.3.1 Administration server4.3.2 Runtime server4.3.3 Agent.4.3.4 Mail server.4.4 Installation4.4.1 Administration server4.4.2 Deploy agents4.4.3 Deploy Catalog Manager4.4.4 Installing Fix Pack 24.4.5 Installing Fix Pack 3 | 121 122 123 125 128 129 131 131 131 137 142 144 149 |
| | Chapter 5. Tivoli License Compliance Manager for z/OS 5.1 Installation 5.1.1 Introduction to distributed components 5.1.2 Plan for your installation of the distributed components | 155 157 157 159 |

| 5.1.3 Installation process |
|--|
| 5.1.4 Gather installation information for distributed components 168 |
| 5.1.5 Installing distributed server components |
| 5.1.6 Installing distributed client components |
| 5.1.7 Creating a response file or a template response file |
| 5.2 Collecting raw data |
| 5.2.1 Taking an inventory of software data |
| 5.3 Creating a software product inventory |
| 5.3.1 Identifying products |
| 5.4 Capturing software use data |
| 5.5 Generating reports |
| 5.5.1 Generating data for the distributed component of the License |
| Compliance for z/OS 211 |
| 5.5.2 Mainframe server-based reports |
| 5.5.3 Exporting data to other tools and programs |
| 5.5.4 Managing software use data from the Monitor |
| 5.6 Sub-Capacity Reporting Tool |
| 5.7 Tivoli Asset Compliance Center |
| 5.7.1 Administering and using the Tivoli Asset Compliance Center 217 |
| 5.7.2 Managing software assets with Tivoli Asset Compliance Center 218 |
| 5.8 Summary |
| |
| Chapter 6. Deployment considerations |
| 6.1 Planning for a client engagement 222 |
| 6.1.1 Service engagement preparation |
| 6.1.2 Implementation skills |
| 6.1.3 Available resources |
| 6.1.4 Solution scope and components 223 |
| 6.1.5 Services engagement overview |
| 6.1.6 Creating a contract (statement of work) 224 |

| | 6.2 Required skills2256.3 Solution description and assumptions2276.4 Task breakdown229 |
|---------------|--|
| Part 3. Appen | dixes |
| | Appendix A. IBM Tivoli License Manager Administration server database schema |
| | Appendix B. IBM Tivoli License Compliance Manager agents 275 |
| | Related publications295IBM Redbooks publications295Other publications295Online resources296How to get IBM Redbooks publications296Help from IBM296 |
| | Index |

Figures

| 1-1 Infrastructure complexity | . 5 |
|---|-----|
| 1-2 Optimal IBM IT Service Management approach | . 7 |
| 1-3 IBM IT Service Management overview | . 8 |
| 1-4 ITUP Overview | 13 |
| 2-1 IT and Software Asset Management as part of IT Service Management . | 17 |
| 2-2 IT asset management categories | 19 |
| 2-3 Software asset management in IBM IT Service Management | 22 |
| 2-4 Annualized cost avoidance with a six year forecast | 24 |
| 2-5 Required detail for software cost analysis | 25 |
| 2-6 Extra funds available in fourth quarter | 25 |
| 2-7 Software asset management maturity level model | 27 |
| 2-8 IT asset management summary | 29 |
| 2-9 Product positioning within the solution portfolio | 30 |
| 3-1 Menu structure of Tivoli Asset Management for IT | 41 |
| 3-2 Modules associated with Tivoli's process automation engine | 42 |
| 3-3 Installation flowchart | 47 |
| 3-4 Initial launchpad window | 49 |
| 3-5 Installation menu | 50 |
| 3-6 Language selection | 50 |
| 3-7 Welcome page | 51 |
| 3-8 License agreement | 52 |
| 3-9 Middleware selection | 53 |
| 3-10 Deployment plan | 54 |
| 3-11 Middleware password selection | 55 |
| 3-12 DB2 Enterprise Edition configuration parameters | 56 |
| 3-13 Database instance configuration | 57 |
| 3-14 DB2 user groups configuration | 58 |
| 3-15 IBM Tivoli Directory Server configuration (panel 1 of 2) | 59 |
| 3-16 Parameters for IBM Tivoli Directory Server configuration (panel 2 of 2). | 60 |
| 3-17 IBM Tivoli Directory database instance configuration | 61 |
| 3-18 WebSphere Application Server security configuration (panel 1 of 2) | 62 |
| 3-19 WebSphere Application Server security configuration (panel 2 of 2) | 63 |
| 3-20 WebSphere Application Server ND version configuration | 64 |
| 3-21 WebSphere Application Server configuration parameters | 65 |
| 3-22 WebSphere Application Server ND version | 66 |
| 3-23 IBM HTTP Server configuration | 67 |
| 3-24 IBM HTTP Server plug-in configuration | 68 |
| 3-25 IBM Rational Agent Controller configuration | 69 |

| 3 | 3-26 | Middleware product images location | . 70 |
|---|---------------|--|------|
| 3 | 8-27 | Middleware product image directory | . 71 |
| 3 | 8-28 | Middleware temporary files directory | . 72 |
| 3 | 3-29 | Manage middleware deployment plan | . 73 |
| 3 | 3-30 | Middleware deployment plan | . 74 |
| 3 | 3-31 | Middleware installation completed | . 75 |
| 3 | 3-32 | Launchpad | . 78 |
| 3 | 3-33 | InstallAnywhere extraction | . 79 |
| 3 | 3-34 | Language selection | . 79 |
| 3 | 3-35 | Introduction panel | . 80 |
| 3 | 3-36 | Software license agreement | . 81 |
| 3 | 3-37 | Import middleware configuration information | . 82 |
| 3 | 8-38 | Choosing simple or custom deployment | . 83 |
| 3 | 3-39 | IBM Tivoli base services configuration message | . 84 |
| 3 | 8-40 | Choose installation folder | . 85 |
| 3 | 8-41 | Maximo database type selection | . 86 |
| 3 | 3-42 | DB2 database configuration | . 87 |
| 3 | 3-43 | Automate database configuration prompt | . 88 |
| 3 | 8-44 | Remote access authorization | . 89 |
| 3 | 3-45 | DB2 administration configuration | . 90 |
| 3 | 8-46 | DB2 Tablespace configuration | . 91 |
| 3 | 8-47 | Maximo Application Server Type | . 92 |
| 3 | 8-48 | WebSphere Connectivity configuration | . 93 |
| 3 | 8-49 | WebSphere Remote Access Authorization | . 94 |
| 3 | 8-50 | Automate WebSphere Configuration prompt | . 95 |
| 3 | 8-51 | WebSphere Deployment Manager Configuration | . 96 |
| 3 | 3-52 | WebSphere Application Server Configuration. | . 97 |
| 3 | 8-53 | Application Server Security configuration | . 98 |
| 3 | 8-54 | Integration Adapter JMS Configuration | . 99 |
| 3 | 8-55 | Run Configuration Step. | 100 |
| 3 | 8-56 | Choose a Shortcut Folder option | 101 |
| 3 | 8-57 | Installation configuration Input Summary | 102 |
| 3 | 8-58 | Pre-Installation Summary | 103 |
| 3 | 8-59 | Language Support option | 104 |
| 3 | 8-60 | Enable Tivoli Asset Management for IT License for usage | 106 |
| 3 | 8-61 | | 107 |
| 3 | 8-62 | Package Validation Results | 108 |
| 3 | 8-63 | Middleware Login information: WebSphere administrator and remote | 400 |
| ~ | | | 109 |
| 3 | 5-64 | | 110 |
| 3 | 5-65 | | 111 |
| 3 | 8-66 0 - 7 | Deployment Progress | 112 |
| 3 | 5-67 | Package Successfully Deployed | 113 |

| 3-68 Report administration | 116 |
|--|---------------|
| 3-69 Generate Request Pages success | 116 |
| 3-70 Labor panel | 117 |
| 3-71 Run Reports | 118 |
| 3-72 Labor List | 118 |
| 3-73 Request Page | 119 |
| 3-74 Report output | 119 |
| 4-1 TLCM three-tier architecture | 124 |
| 4-2 Component relationships | 125 |
| 4-3 IBM Tivoli License Compliance Manager Catalog Manager too | Ι127 |
| 4-4 RSH and SSH installations | 131 |
| 4-5 Check for fully qualified host name | 132 |
| 4-6 TLCM Administration Server Services | 134 |
| 4-7 Organization warning | 135 |
| 4-8 Create an Organization | 136 |
| 4-9 Deploy the IBM License Compliance Manager Agent. | 137 |
| 4-10 Active X security settings | 139 |
| 4-11 Enable signed ActiveX controls. | 140 |
| 4-12 Verify agent status | 142 |
| 4-13 Catalog Manager installation parameters | 143 |
| 4-14 Catalog load message | 143 |
| 4-15 Server upgrade results for Fix Pack 3 | 149 |
| 4-16 Updated catalog after Fix Pack 3 installed | 150 |
| 4-17 Catalog Manager: Catalog version | 151 |
| 5-1 Distributed server components: IBM Tivoli License Compliance | Manager for |
| z/OS | 158 |
| 5-2 IBM Tivoli License Compliance Manager for z/OS structure and | d flow 160 |
| 5-3 Minimal configuration for IBM Tivoli License Compliance Mana | ger for z/OS: |
| Single computer with WebSphere Application Server | 161 |
| 5-4 Alternate configuration with four servers | 162 |
| 5-5 Surveyor flow | 204 |
| 5-6 Extractor flow | 205 |
| 5-7 Identifier flow | 206 |
| 5-8 Monitor flow | 210 |
| 5-9 Distiller flow | 212 |
| 5-10 Reporter flow | 213 |
| 5-11 Exporter flow | 214 |
| 5-12 Selector flow | 215 |
| 5-13 Tivoli Asset Compliance Center flow | 216 |
| 5-14 Administration perspective | 218 |
| 6-1 Simplified Tivoli Asset Management for IT solution layout | 228 |



Tables

| 4-1 Agent platforms and JRE download locations. | 138 |
|--|-----|
| 4-2 Installed files and directories. | 140 |
| 5-1 Operating systems | 163 |
| 5-2 Directories for the Windows operating system | 167 |
| 5-3 Directories for the Linux operating system | 167 |
| 5-4 Installation setup file names by operating system | 170 |
| 5-5 IBM Tivoli License Compliance Manager for z/OS parameters | 176 |
| 5-6 Parameters for any distributed server component | 179 |
| 5-7 Parameter for installations with any distributed database component | 180 |
| 5-8 Parameters for installations of Asset Management server component | 181 |
| 5-9 Parameter for installations | 182 |
| 5-10 Parameter for Asset Management database component installations . | 182 |
| 5-11 Parameter for installations of Data Aggregation server component | 183 |
| 5-12 Parameter for installation | 183 |
| 5-13 Installation parameters | 184 |
| 5-14 Installation parameter | 186 |
| 5-15 Installation setup file name by operating system | 189 |
| 5-16 Distributed client components | 190 |
| 5-17 Installation parameters | 191 |
| 5-18 Parameter descriptions | 193 |
| 5-19 Directories for the Windows operating system | 201 |
| 5-20 Directories for the Linux operating system | 201 |
| 5-21 Installation setup file names | 203 |
| 6-1 Client involvement examples during implementation and deployment | 233 |
| A-1 IBM Tivoli License Compliance Manager Administration server database | Э |
| 238 | |



Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

| AIX® | IBM® |
|------------------------------|---------------------|
| AlphaBlox® | Maximo® |
| DB2 Universal Database™ | OS/400® |
| DB2® | Passport Advantage® |
| Enterprise Asset Management® | Rational® |
| i5/OS® | Redbooks® |

Redbooks (logo) System i™ Tivoli® WebSphere® z/OS®

The following terms are trademarks of other companies:

Distiller, and Portable Document Format (PDF) are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library, IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

J2EE, Java, JavaServer, JDBC, JRE, JSP, Solaris, Streamline, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Active Directory, ActiveX, Internet Explorer, Microsoft, SQL Server, Windows Server, Windows Vista, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This deployment guide provides information related to the deployment of the products that make up the IBM® Tivoli® IT Asset Management portfolio of products. These products include IBM Tivoli Asset Management for IT, IBM Tivoli License Compliance Manager, and IBM Tivoli License Compliance Manager for z/OS®.

The Asset Management for IT V7.1 product is based on the Maximo® Asset Management product for managing enterprise assets. Specific functions are provided that are targeted to the needs of IT asset managers. The Tivoli License Compliance Manager and Tivoli License Compliance Manager for z/OS are complementary products that specifically address the discovery and management of the software being used in an enterprise and the management of the associated licenses.

In the first part of this book, we introduce IBM Service Management and describe how IT asset management is a critical component of service management. We then provide more detailed information about each of the three products that make up the IT Asset Management portfolio and provide guidance on their installation and deployment. Finally, we provide a high level overview of how to plan for an engagement for the deployment of an IT asset management solution.

The team that wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.

Bart Jacob is a Senior Consulting IT Specialist at IBM Corporation -International Technical Support Organization, Austin Center. He has over 25 years of experience providing technical support across a variety of IBM products and technologies, including communications, object-oriented software development, and systems management. He joined the ITSO in 1989, where he has been writing IBM Redbooks® publications and creating and teaching workshops around the world on a variety of topics. He holds a Masters degree in Numerical Analysis from Syracuse University.

Rajat Khungar is a Senior Specialist - Asset and Service Management with Birlasoft in India. He is an IT Infrastructure Library® (ITIL®) Service Manager Certified (Red Badge holder) and an IBM Certified Maximo ITSM Deployment Professional with experience in implementing and managing solutions based on IBM Tivoli products. He has been a keynote speaker at various international symposiums and conferences, such as MaximoWorld, TTUC, and IBM Pulse and focuses on advising and envisaging Asset and Service Management solutions. He holds an engineering degree in Electronics and Communications and an MBA from NMIMS, one of the top Business Schools in India, and is working with Birlasoft in a Leadership Track Program, "Alchemy".

Carlos J. Otálora Y. is currently working as a Tivoli Technical Specialist in the IBM Business Partner Innovation Center (BPIC) in Colombia with NEXSYS DE COLOMBIA S.A. (The IBM Software Value Added Dealer (VAD) in Colombia). He holds a Bachelor of System Administration degree in Computer Science and a degree in Security Management. He has 22 years of experience in the IT field. He has technical certifications in IBM Certified Deployment Professional - Tivoli Productivity Center For Data, IBM Certified Deployment Professional - Tivoli Storage Manager. He has IBM Tivoli Sales Mastery Certifications in Storage, Automation, and Security Solutions, He has Microsoft® Certified Professional - MCP and ITIL Foundation Certification in IT Service Management.

James Pittard has extensive experience with Maximo and Tivoli's process automation engine implementations at client sites, especially implementations for IT asset and service management. He is currently working for IBM as a developer for and focusing on managing license procurement, distributions, and compliance.

TP Raghunathan is an IT Specialist working for Bharti Airtel Project in India. He is responsible for infrastructure in the Central Production Data Center (CPDC), which includes Inventory and Asset Management, Physical Access Control and Security, and Media Management, and is a single point of contact for client interactions with regard to CPDC. He has over 20 years of experience in the IT field and has been with IBM for 3 1/2 years. With IBM, he has been an Asset Manager for the Bharti Airtel Project and involved in B2O (Build to Operate) applications in the Bharti Airtel Project. His areas of expertise include Asset Management, Infrastructure Management, and Process and Tool implementations.

David Stephenson is a native of Sydney, Australia, and for the last 9 1/2 years has worked for IBM Global Technology Services in distributed Systems Management roles. David specializes in Event Management and his depth of recent experiences includes diverse roles encompassing test management, database administration and design, network management, software and license management, financial modelling, and leverage of IBM Tivoli Monitoring Version 6. David has co-authored many IBM Redbooks publications about Systems Management and holds a Masters of Commerce with an Advance Specialization





Figure 1 I to r - David Stephenson, TP Raghunathan, James Pittard, Bart Jacob, Rajat Khungar, and Carlos Otalora

Thanks to the following people for their contributions to this project:

Melody Bienfang David Hirshfield Tim McCrimmon Margaret Radford Ed Rossi Lisa Wood

Thanks to the authors of a predecessor book: *IBM Tivoli Asset Management for IT Portfolio Overview*, published in August, 2007:

Nicholas Ara Michael Beres David Boggs Carmen Ces Julio Zaima

Become a published author

Join us for a two- to six-week residency program. Help write a book dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You will have the opportunity to team with IBM technical professionals, Business Partners, and Clients.

Your efforts will help increase product acceptance and client satisfaction. As a bonus, you will develop a network of contacts in IBM development labs and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

Use the online Contact us review IBM Redbooks publications form found at:

ibm.com/redbooks

Send your comments in an e-mail to:

redbooks@us.ibm.com

Mail your comments to:

IBM Corporation, International Technical Support Organization Dept. HYTD Mail Station P099 2455 South Road Poughkeepsie, NY 12601-5400

Part 1

Service Management and Asset Management Overview



1

IBM Service Management platform

Before discussing the specifics of IT asset management, we start this book with an overview of service management from an IBM perspective. IT asset management is a discipline of overall service management, and to understand how the IBM Tivoli IT Asset Management portfolio of products is structured and positioned, it is important to first have a basic understanding of the IBM Service Management strategy.

1.1 IBM Service Management

IBM has developed thought leadership to improve the "state of the art" in service management for the last 25 years and has supported other organizations in their efforts as well. In addition to the advancement of management disciplines and technologies, IBM recognized early on that acceptance of common practices and standards is vital to achieving improved value from information technology (IT).

Advances in technologies and management disciplines provide the greatest value once they become part of, and extend, the body of generally accepted practices and open standards. IBM supports the advancement of practices and open standards, such as the IT Infrastructure Library (ITIL), Control Objectives for Information Technology (COBIT), International Organization for Standardization (ISO) IEC 20000, and Carnegie Mellon University's e-Sourcing Capability Model (e-SCM). The fundamental characteristics of service management require integration and agreement on standards, not only between tools and roles within IT, but also among organizations and even industries.

IT Service Management is the integrated set of activities required to ensure the cost and quality of IT services valued by the client. It is the management of client-valued IT capabilities through effective processes, organization, information, and technology, including:

- Aligning IT with business objectives
- Managing IT services and solutions throughout their life cycles
- Service management processes, such as those processes described in ISO IEC 20000, ITIL, and the Process Reference Model for IT

1.1.1 Why Do Businesses Need IT Service Management

Today's enterprises face an ever-increasing problem of managing their IT processes to deliver IT services in a manner that is:

- Efficient
- Reliable
- Secure
- Consistent

At the same time, the complexity of the infrastructure needed to deliver these IT-enabled business services has been increasing rapidly. A simple example that shows the complexity of IT environments is shown in Figure 1-1 on page 5.



Figure 1-1 Infrastructure complexity

Several of the key challenges that businesses face are:

- Complexity: The root cause of the problems IT organizations face lies in the dramatic increase of business complexity due to heterogeneity of environments and the interconnection of applications (composite applications). Architectural and organizational issues, accelerating the proliferation of composite applications and hardware entities, and worldwide operations spanning multiple time zones all contribute to reducing the efficiency and effectiveness of the IT organization.
- Change: Complexity makes for very brittle, hard-to-manage infrastructures that often break under change and whose management requires a discipline that few companies achieve without flaws. Increasing workloads, more stringent service-level assurance requirements, staff turnover, and new market opportunities all lead to pressure for change in the IT organization.

Change is the leading cause of service or application disruption today, and it often results in visible business impact. In fact, experience suggests that nearly 80 percent of all critical outages can be traced to faulty change management.

Cost: Currently, operational IT labor cost constitutes almost 70 percent of the total IT budget of businesses. In the late 1990s, half of the IT labor budget was devoted to new application development and half was devoted to operations. Because IT budgets have been held flat, the chief information officers of IT organizations have faced two unappealing choices: shift resources from new application development or reduce the level of support for current applications. Both options serve to reduce the efficiency and effectiveness of IT.

Governance and compliance: The introduction of government regulations, such as the Sarbanes-Oxley Act (SOX) and the Health Insurance Portability and Accountability Act (HIPAA), has put an additional burden on the IT organization to support the needs of the business to audit for compliance through the institution of better process controls and the maintenance of audit trails for IT infrastructure changes. This audit requires careful consideration because of the penalties of noncompliance, including criminal and civil liabilities and adverse public opinion.

1.1.2 What is IBM Service Management

For many businesses, service excellence is increasingly a competitive differentiator, because organizations need to rapidly adapt to changing conditions in the marketplace and create and deploy new services quickly and efficiently. However, service excellence can only be achieved through effective and efficient service management.

A fundamental goal of IT service management is the management of IT services and infrastructure with the same kinds of quality control that enterprises strive to use for all business processes. When this quality control is achieved, businesses have the confidence to deploy new and updated services that are critical to their missions.

An effective IT service management capability reduces the time needed to deliver a company's IT services according to business policies and reduces the labor cost of the people involved in executing the processes by replacing manual IT process management with autonomic management.

Figure 1-2 on page 7 depicts IBM IT Service Management and its various components.



Figure 1-2 Optimal IBM IT Service Management approach

IBM has taken a comprehensive approach to IT service management that spans from business models to technologies and products.

The IBM IT Service Management approach in terms of processes is based on the Information Technology Infrastructure Library (ITIL), which is described in 1.2, "Information Technology Infrastructure Library (ITIL)" on page 8.

IBM Service Management (ISM) is an approach designed to automate and simplify the management of business services. It concentrates on four areas of study:

- Technology integration and standards
- Improved collaboration among IT people spread across organizational silos
- Best practices-based process modules to enable automated process execution
- Sharing of business-critical IT information to improve decision making

In finding workable solutions to these areas, IBM solutions cover four key areas:

- Process Managers that provide automated ITIL-aligned workflows for key IT processes
- An open, standards-based IBM IT Service Management platform

- Integration between process tasks and operational management products (OMPs) to automate the running of those tasks from the process flow
- Best practices to help pull it all together

Those four areas are pictured in Figure 1-3.



Figure 1-3 IBM IT Service Management overview

Refer to the following Web site for details about IBM IT Service Management:

http://www.ibm.com/software/tivoli/features/it-serv-mgmt/index.html

Aside from traditional IT management disciplines that have focused heavily on availability management, IT asset management is an important piece of IT service management.

1.2 Information Technology Infrastructure Library (ITIL)

Information Technology Infrastructure Library (ITIL) is an internationally recognized framework that provides comprehensive *best practice* guidelines on all aspects of end-to-end service management. It covers people, processes, products, and the use of partners. It began in the 1980s when the UK government initiated an exercise to standardize its diverse IT processes.

The best practices contained in ITIL are independent of tool, vendor, or industry and can be applied to an organization of any size. ITIL encourages organizations

to adapt and adopt its suggestions to meet business needs and improve processes. Though there is a significant amount of detail in the books that make up the library, the books are not themselves the solution to all IT management issues. The processes require significant work to deploy at a level of detail enabling day-to-day use, with dependencies on the three key components (process, people, and tools) of a management system.

It must be noted that although many people make reference to ITIL as a standard, it is not one. Organizations cannot comply with ITIL. It is a set of guidelines that an organization can adopt and adapt to their needs.

1.2.1 ITIL Version 3

In 2007, ITIL Version 3 was launched that includes a life cycle management approach in five core volumes:

- Service Strategy
- Service Design
- Service Transition
- Service Operation
- Continual Service Improvement

Service Strategy

This core volume provides a view to align business and IT so that each area brings out the best in the other. It ensures that every element of the service life cycle is focused on client outcomes and relates to all the companion process elements that follow. The four main activities in the Service Strategy are to define the market, develop the offerings, develop the strategic assets, and prepare for execution. Service Strategy encompasses the following processes:

- Strategy Generation
- Market Intelligence
- IT Financial Management
- Service Portfolio Management
- Demand Management
- Risk Management

Service Design

This core volume provides guidance for the design of a new or changed service for introduction into the live environment, ensures there is a holistic approach to all aspects of design, and considers all aspects when changing or amending any of the individual elements of a design. Service Design encompasses the following processes:

Service Portfolio Management

- Service Catalog Management
- Service Level Management
- Capacity Management
- Availability Management
- Service Continuity Management
- Information Security Management
- Supplier and Contract Management

Service Transition

This core volume provides guidance for the development and improvement of capabilities for transitioning new and changed services into the production environment. It focuses on the broader, long-term change management role and release practices. Service Transition encompasses the following processes:

- ► Change Management
- Service Asset and Configuration Management
- Knowledge Management and Service Knowledge System
- Service Release and Deployment Planning
- Performance and Risk Evaluation
- ► Testing
- Acquire, Build, and Test Release
- Service Release, Acceptance, Test, and Pilot
- Deployment, Decommission, and Transfer

Service Operation

Service Operation introduces, explains, and details delivery and control activities to achieve operational excellence on a day-to-day basis. Many of the familiar processes from the former service support and service delivery books of ITIL Version 2 are in this book. Service Operation encompasses the following processes:

- Monitoring and Event Management
- Incident Management
- Request Fulfillment
- Problem Management
- Access Management

Continual Service Improvement

This volume provides guidance for continual alignment of the portfolio of IT Services with the current and future business needs, growth, and maturity of the enabling IT processes for each service in a continual service life cycle model, activities to support a continual process improvement plan, and how to measure, interpret, and take action. Continual Service Improvement encompasses the following processes:

- Measurement and Control
- Service Measurement
- Service Assessment and Analysis
- Process Assessment and Analysis
- Service Level Management
- Improvement Planning

1.2.2 Critical Success Factors to Implement ITIL

Because ITIL is a framework of best practices and not a methodology, it only describes what needs to be done. ITIL does not provide guidance on how to implement the processes, so each organization must decide how to implement these processes to fit their own requirements.

A key mind-set when implementing ITIL is "adopt and adapt": "Adopt" ITIL as a common language and reference point for IT Service Management, and "adapt" ITIL best practices to achieve business objectives.

Generally, IT organizations do not implement all ITIL processes, because they do not have the budget, or they judge that they do not need all of the processes. Initially, not implementing all processes can be seen as a way to avoid extra costs. However, depending on the processes chosen to be implemented, choosing not to implement other processes can result in fewer benefits from the implemented processes. For example, choosing to implement Change and Release processes without implementing Configuration can result in an inaccurate impact assessment when approving changes.

Carefully select the service management processes and take into consideration the relationship among all processes and not only the cost perspective and the implementation complexity of individual processes.

A successful implementation of IT Service Management must:

- Be aligned with business needs; business-driven not technology-driven
- Improve staff awareness about business goals
- Be adapted to the culture of the organization. This adaptation must be done when defining the roles, responsibilities, tools, processes, procedures, tasks, and so on. After IT Service Management is implemented, it must be rigorously followed.
- Have its processes clearly defined, documented and available
- Have its main processes integrated with each other
- ► Have its inputs measurable and repeatable
- ► Have IT process tools supported and customized to fit the processes defined

- Have processes easily changed as necessary
- Be integrated with external suppliers
- Include proper training and communication to all people who will use or provide IT services
- Have clearly measurable and repeatable key performance indicators

A successful ITSM implementation needs to result in improved IT client satisfaction, better resource utilization, and improved client perception of IT service quality.

1.3 IBM Tivoli Unified Process

This section describes the IBM Tivoli Unified Process (ITUP) and its relationship with IT industry best practice models.

As described in 1.2, "Information Technology Infrastructure Library (ITIL)" on page 8, ITIL was developed as a set of Information Technology best practices, and its primary goal is to define and organize IT processes. ITIL documents only what must be done; it does not show how to implement processes.

ITIL is aimed at identifying best practices. ITIL describes a systematic approach to creating a service-oriented culture and practice for IT service management. The library emphasizes the central importance of meeting business requirements economically. However, IT organizations will need to look beyond ITIL to understand the IT management process disciplines that are central to delivering on the growth agenda. Leaders in IT management must handle the competing strategic priorities that force trade-offs between cost-efficiency, flexibility, and service availability.

To assist IT organizations in this critical challenge, IBM developed the Process Reference Model for IT. The IBM model supplements the content of ITIL based on the IBM extensive IT management experience, which was gained from managing thousands of IT environments, both large and small. The Process Reference Model for IT identifies the set of IT management processes required to move beyond a singular cost focus to principled decision making that accounts for changing business and technology conditions while managing the complexity of existing systems.

You cannot directly implement ITIL and the Process Reference Model for IT, and to address the gaps between them, IBM developed the IBM Tivoli Unified Process (ITUP).

ITUP describes a comprehensive set of IT processes within an IT organization. It is aligned not only to ITIL Version 3 and the Process Reference Model for IT, but also with best practices from industry-wide specifications of IT best practices, such as ISO 20000, Enhanced Telecommunications Operations Map (eTOM), Six Sigma, and COBIT (Control Objectives for Information and Related Technology), and proposes a process framework that incorporates the best from each practice. Figure 1-4 provides an overview of ITUP.



Figure 1-4 ITUP Overview

Each ITUP process is defined by:

- Its goals, mission, scope, and key performance indicators (KPIs)
- A workflow
- People (Roles)
- Information (Work Products by Name)
- Products (Tools) that help implement aspects of the process

In addition, problem scenarios describe how processes work together to solve common IT issues.

1.3.1 ITUP Composer

ITUP Composer is a tool that allows for an implementation of the ITUP framework by defining and creating IT service management processes that will fit the business needs of an organization. ITUP Composer is a library made up of the IBM Rational® Method Composer (RMC), which is a tool platform that enables the development, customization, and publication of methods and processes. ITUP Composer's key elements and concepts are described in the following sections.

In summary, ITUP composer provides a tool set to enable you to customize and enhance ITUP content to fit the needs of a particular organization and its unique processes.

2

IBM Tivoli IT Asset Management portfolio

IT asset management addresses issues related to two important initiatives that many businesses are addressing: Increasing the efficiency of the IT environment and enterprise asset management.

Enterprises are constantly looking for ways to operate more efficiently and cost-effectively. The costs associated with information technology are under continuing scrutiny, and to stay competitive, enterprises must discover ways to reduce costs and increase the value of the IT environment to their core business.

Enterprises are also looking to better control all of their corporate assets and ensure that their corporate assets are managed in a fiscally responsible way. This approach not only makes good business sense, but because of government regulations, such as Sarbanes-Oxley, enterprises are having to be more aggressive in implementing asset management processes.

The assets of an IT organization need to be managed along with other assets of the enterprise. Applying the disciplines of enterprise asset management to the IT environment helps meet the dual objectives of managing all corporate assets and making the IT environment more cost-effective.

The IBM Tivoli IT Asset Management portfolio helps organizations manage their IT assets efficiently and effectively, lowering cost, mitigating license and regulatory compliance risks and better aligning IT with business goals.

IT asset management is part of the larger discipline of IT service management as described in Chapter 1, "IBM Service Management platform" on page 3.
2.1 IT asset management

IT asset management is a subset of the larger discipline of IT Service Management as depicted in Figure 2-1.



Figure 2-1 IT and Software Asset Management as part of IT Service Management

IT asset management covers hardware and software, as well as related contracts. From the hardware perspective, enterprises might be somewhat further along in adopting an adequate set of policies and processes to manage the various IT equipment. The capital expenses involved and associated depreciation costs must be managed by any fiscally responsible organization.

Of course, this is not to say that IT hardware assets are fully under control. With the number of personal computers and various related accessories, such as memory, hard disks, and other options, organizations easily lose track of these IT resources. Costs for managing this environment are on the rise, and the sheer number of systems involved make managing this environment a daunting task.

Software asset management is a discipline that has historically had much less focus. However, software asset management is becoming a critical discipline that organizations must implement to stay competitive. Several of the factors driving the need for more discipline in the area of managing software assets include:

Training and support costs

- Risks associated with license compliance issues
- ► Government regulations
- Growing costs of software

Several of the benefits of IT asset management are:

- Reliable asset inventory information facilitates decision making around IT spend and asset allocation.
- It helps to optimize efficient IT asset utilization.
- Effective IT asset management provides the cost and driver information to help better align IT with business requirements.
- Understanding your IT assets helps you meet your regulatory compliance initiatives and manage your IT risk (Sarbanes-Oxley, Basel II).
- Gaining an accurate view of the IT infrastructure, contracts, and financial data enables rapid recovery in the event of a disaster.
- IT asset management simplifies the growing complexity of license compliance while mitigating risks and avoiding fines due to the under-licensing of software.
- Effective IT asset management helps reduce IT costs by avoiding over-licensing and redeploying under-utilized assets.
- Good IT asset management improves your portfolio management.
- IT asset management creates standardization of IT technology in all asset classes.
- IT asset management helps you avoid manual inventory audits.
- IT asset management includes data gathering to assist in early virus detection and prevention.
- ► IT asset management provides better server management.
- IT asset management can help you reduce property taxes.
- Effective IT asset management can help reduce unplanned downtime and help decrease the number of non-productive IT assets.

Figure 2-2 on page 19 illustrates the "touch points" of IT asset management.



Figure 2-2 IT asset management categories

The key to effective IT asset management seems quite simple: "Know what you have." But there is a catch; to know what you have depends on the processes and tools that are used to manage IT assets. Software license compliance is among the most evasive challenges facing organizations today. CxOs and IT managers are turning their attention to best practice frameworks to migrate ad hoc practices to comprehensive, process-driven IT asset management programs. An improved IT asset management approach can significantly reduce IT costs as well as mitigate the financial risks of improper software license management.

It is no secret that over the past decade the cost trends for hardware and software are clearly progressing in inverse proportion to each other. Hardware costs are becoming a small percentage of the overall IT budget, where software and personnel costs are increasing significantly.

2.1.1 Challenges associated with IT asset management

Unfortunately, IT asset management is not without its challenges, which can include:

Maverick purchasing

Often, corporate IT policies and procedures are complicated and time-consuming. In an effort to reduce red tape and improve agility, divisions or departments are often allowed to conduct their daily operations with little

oversight from corporate headquarters. As a result, these groups acquire IT assets, including software, as they see fit.

For large organizations, even a 5 to 15 percent reduction in maverick purchasing can lead to millions of dollars in savings, because every dollar saved has a direct impact on the organization's bottom line. Recognizing that maverick purchasing is occurring and implementing policies and procedures to control the spending is not a simple process. Senior level executives are asking why this is happening and the answers are not surprising:

- Poor initial analysis
- Inadequate control
- Inability to enforce a process

With that being said, one must ask the following question, "How does an organization go about eliminating maverick spending on IT products and what steps need to be taken in order to realize the goal?"

An enterprise can significantly reduce IT costs if it has a centralized process, where possible, for asset acquisition. When there is centralized management of the purchasing process, companies can use scale to negotiate more favorable software agreements, redeploy and reuse licenses and reduce costs associated with over-licensing, and reduce the risk of fines and penalties due to under-licensing.

Another source of purchasing control problems is the inability of an enterprise to impose software standards. For example, policies, such as the cross-organizational expensing of software, make the enforcement of an approved standard virtually impossible. Not only is unauthorized software unaccounted for, an increase in the unbudgeted costs of license management and infrastructure support occurs. Unauthorized software is frequently incompatible with the "approved" environment, causing conflicts, crashes, and system failures, reducing productivity due to employee downtime, and increasing IT support costs.

It is important to put in place IT asset management processes that control, identify, prohibit, and remove unauthorized software from the enterprise. These processes ensure employees are purchasing only approved software, and these processes provide an equitable policy for software across all employees based on role or job function.

Disconnected processes

Financial, procurement, contract management, and IT groups are all involved in the IT asset life cycle process. Often, these groups do not communicate or do not leverage the central asset repositories. Disconnected processes are especially common with software, where there can be pools of software licenses that might not be taken advantage of, resulting in poor control of software assets. For example, installed software is often not reconciled with procured software, resulting in a misrepresentation of deployed software. Also, software licenses are typically not reused when systems are retired or upgraded, resulting in excessive software license purchases. Finally, it is common to find enterprises that are not capturing application usage details, thus preventing the ability to renegotiate more advantageous contract terms.

Technology limitations and evolution

The basis for any good IT asset management system is accurate and up-to-date information, along with the processes and technology to maintain accuracy. Many organizations, particularly in sensitive fields, such as government and financial services, maintain strict controls over the IT assets for security purposes. These tight controls also help to keep the software repository or hardware warehouse more relevant, because unapproved software and hardware are much more difficult to bring into the environment. However, IT asset management is still a challenge for these large organizations. Furthermore, in most industries it is less practical to "lock down" user environments to prevent unapproved software and hardware installation.

2.1.2 Software asset management

As a subset of IT asset management, software asset management's goal to analyze actual software usage and manage related contracts makes it a fundamental part of the IBM IT Service Management strategy. As part of IBM IT Service Management, the IT asset management discipline allows a business to transform from a siloed IT asset management approach to one aligned with overall business operations.

Figure 2-3 on page 22 represents key components of software asset management within the context of IBM IT Service Management.



Figure 2-3 Software asset management in IBM IT Service Management

As depicted in Figure 2-3, software asset management touches on several aspects of IBM IT Service Management, including:

IT Asset Management

As already mentioned, software asset management is a component of IT asset management. The ITIL definition of IT Asset Management is Software Asset Management and Hardware Asset Management together. IBM includes contract management as well as software and hardware management as the key components of IT asset management. Software asset management is concerned not only with the software itself, but also with its related contracts associated with software licensing.

In general, hardware costs tend to remain fairly stable and have been declining over the years, as opposed to software and contracts, whose costs are on the rise and remain unstable.

► IT Financial Management

IT Financial Management in IT Service Management deals with understanding the costs of IT assets including software. Software asset management therefore is a key component of the IT Financial Management process, because it allows clients to fully account for all IT software expenditures across the enterprise. With software asset management in the IT Finance Management process, clients can control and manage their IT software budget while at the same time assisting them with software decisions. A secondary benefit is that a software asset management solution enables the allocation of software costs to the users receiving them. It provides the framework for recording the costs of software, so that licenses can be allocated and costs recovered for the IT service to the user.

Release Management

Release Management in IBM IT Service Management is the process of performing a successful software or hardware change in an IT environment. Software asset management's involvement in this process is primarily in the front end, confirming that the software release is within the contract policy and does not have any interdependencies with other IT assets. The software asset management piece of release management is critical for the planning and assessment portion of the software release cycle.

Capacity Management

Capacity Management is an area concerned with ensuring that IT processing meets the evolving needs of the business. A software asset management solution allows you to understand where there is over-spending and under-spending on software licenses so that you can shift spending to align with the business needs. An IT asset management solution provides a current analysis of resource usage so that decisions can be made on actual usage rather than static contracts. In this way, an IT asset management solution ensures that the hardware and software license capacity of the company meets the current and future needs of the business.

Compliance Management

It is important that a software asset management solution provides the ability to audit and account for IT assets, software, and expenditures, which forms the foundation to meeting the mandates of numerous regulations. It is the intention of IBM to fully integrate its software asset management solution with other IBM compliance solutions to provide a common infrastructure to help companies with their compliance activities.

The IBM Tivoli IT Asset Management portfolio provides a process to automate, measure, and audit software assets, helping you to manage your regulatory compliance requirements.

Benefits of software asset management

As a subset of IT asset management, here is a simple example of the benefits related to software asset management.

Savings and cost avoidance

Let us look at a real-world example of the potential savings achievable through the implementation of minimal software asset management practices. In Figure 2-4, all upgrade and maintenance costs are shown in blue, while software savings are shown in yellow. Notice that the initial savings will begin immediately after action is taken to eliminate unused (or under-used) software, typically in the first year. Estimates put these savings at 30% or more. The savings will continue annually at a smaller but still significant percentage. However, as software costs increase, that percentage will result in a higher dollar amount, thereby increasing the return on investment (ROI).



Figure 2-4 Annualized cost avoidance with a six year forecast

Continuing to carry under-utilized software year after year can have an accumulative effect. It becomes more difficult to stay on budget as IT funding comes under greater scrutiny over time. The graph in Figure 2-5 on page 25 presents a different perspective of software costs on a macro level. Without numeric values, you can see how easily a department's annual software budget can be exceeded.



Figure 2-5 Required detail for software cost analysis

With effective license and software asset management, a far different outcome can be expected. With little budget increase and careful monitoring of software assets, additional funds are available for new projects (Figure 2-6) with an IBM Tivoli end-to-end software asset management solution.



Figure 2-6 Extra funds available in fourth quarter

Risk mitigation

Risk mitigation is another important consideration, because software makers have increased their focus on license compliance, and the ability to audit is more common in software contracts. Software investments are a major component of any company's financial profile, and corporate reputations can be severely compromised in the event of a noncompliance scandal. For a company to be in full compliance with Sarbanes-Oxley, it must report all assets accurately. A solid software asset management strategy helps firms maintain up-to-date inventories of their software, ensuring accurate financial and tax statements. Poor documentation and weak license control are at the root of software license compliance problems.

Often, there is no central location for enterprise agreements; instead, licenses are scattered across business units. In addition, generally little or no accountability exists to deter employees from installing unauthorized software on their personal computers. It is important to manage software across the entire enterprise, connecting key departments that interact with software assets, while ensuring license compliance. Software asset management enforces entitlement, assures software redeployment, and reduces the overall expense of software. With these benefits, many organizations are starting to implement asset management programs.

To ensure that an organization is in complete control of the purchase, usage, distribution, and disposal of its valuable software assets, both business and technical process disciplines are required to coexist and complement each other. Software asset management is not solely a technical issue; it requires investment in new business processes that underpin a corporate wide software usage policy in order to be successful. In other words, software asset management is best achieved through an implementation that combines ITIL disciplines and best practice asset life cycle management.

Software asset management maturity model

In the final analysis, software management is a multi-disciplinary process. Static product functionality is not sufficient to ensure that software management policies are enforced. Processes need to be defined and documented to ensure that software compliance policies and procedures are followed.

A strong workflow mechanism is required to "string" together all required activities to ensure that procedures are followed. A five-layer model, based on a capability maturity model, has been built to help measure and track the activities that an enterprise will need to undertake to move forward.

The software asset management (SAM) maturity model serves this purpose, with the layers or levels defined as shown in Figure 2-7 on page 27.



Figure 2-7 Software asset management maturity level model

Level 1: Chaos - Legacy Renovation

The initial environment has ill-defined procedures, controls, and a lack of adequate tools to track or manage software assets, which results in a chaotic environment. The organization generally does not know what it owns, where it is located, and who is using it. Unused software assets are often kept with no control mechanisms or accountability by designated employees. These enterprises often do not have a centralized purchasing or negotiation team. IT contracts are not tracked and are typically stored in folders in filing cabinets. There are no systems in place to track the intellectual capital and negotiation strategies associated with acquisition. Until the process is under statistical control, no orderly progress in process improvement is possible.

Level 2: Reactive - Process Efficiency

At Level 2, IT organizations create spreadsheets or a database to help track software assets. A software asset management program is focused on counting the software assets and involves annual physical inventories to identify and catalog all software, and the process is now repeatable. The organization might use an auto-discovery tool to supplement this data. However, installation, move, add, and change processes are not consistently followed, which reduces the accuracy of the data. Reports are basic and lack sufficient detail to identify and resolve problems. Linking and sharing data with purchasing are sporadic at best. Often, multiple tools overlap without centralized oversight or governance.

Level 3: Proactive - Business Integration

In Level 3, the process is well characterized and reasonably well understood. The organization moves into proactive management by defining its process in terms of software engineering standards and methods, and by making a series of organizational, methodological, and SAM software tool improvements. Inventory data can now be linked with financial and contractual data to create a centralized view of how assets perform. The repository and auto-discovery tools are integrated to provide real-time inventory and faster problem resolution.

Level 4: Service - Extending and Innovating

In Level 4, the SAM process is not only understood, but it is quantified, measured, and reasonably well controlled. The organization has metrics in place to measure the program's value, and service levels can be created to meet broader business goals. Opportunities for savings are identified and communicated to the business units on a regular basis. Process and tools are used increasingly to control and enable faster procurement time frames, more efficient order tracking and improved invoice reconciliation. Inventory levels are managed to prevent overbuying while maintaining low-cost inventory stock levels and retiring or disposal of under performing assets. The organization is learning to project expected costs and problems with reasonable accuracy. This level is when the most significant quality improvements begin.

Level 5: Value Creation - Transformation

At Level 5, organizations have not only achieved a high degree of control over their process, they have implemented and integrated all three tools of a SAM program (repository, auto-discovery, and software-usage) and have a major focus on improving and optimizing its operation. At this level, business units are charged back for computing services, total cost of ownership (TCO) metrics are linked with SAM metrics, and data from management and data from business applications are used to audit the efficiency and effectiveness of established business practices across all software assets. IT cost recovery encompasses usage-based pricing models. The data on the process is used interactively to improve the process and achieve optimum performance.

2.2 Solution components

The components of an IT asset management solution must cover a range of functions and capabilities. The basic questions that must be answered by an asset management solution include:

- What assets do I have?
- What assets are being used?

- In the case of assets requiring licenses (such as software), licenses can also be seen as assets, and therefore, what licenses are being used?
- Where are we over-licensed (potential for savings, by eliminating unused or unneeded licenses)?
- ► Where are we under-licensed (exposed to compliance risk)?
- What are the associated costs for our assets?

Figure 2-8 summarizes the preceding questions and associates them with classes of applications.



Figure 2-8 IT asset management summary

In practice, answering these questions can be quite difficult. Several of the reasons for this difficulty include:

- Lack of centralized cross-platform data repository for:
 - Software inventory
 - Product use activity
 - Contractual and financial information
- Rapid business change causes unexpected software spending and inefficient software deployment.

- The complexities of license metrics, terms and conditions, and software deployment can be challenging.
- Asset life cycle management within functional areas and departmental silos can be complex.
- There is no clear knowledge of what software is installed, how much it is being used, and who is using it.
- Most software asset management practices lack the maturity to drive maximum business value.

The Tivoli IT Asset Management portfolio of products provides the end-to-end capabilities required for implementing an IT asset management solution that meets business goals. The various components of the portfolio and how they fit to meet the overall solution are shown in Figure 2-9.



Figure 2-9 Product positioning within the solution portfolio

Note: The IBM Tivoli License Compliance Manager products not only provide software inventory, but they also provide hardware inventory information.

As IBM enhances and evolves its IT Asset Management solution, the product names and functions might shift over time. However, the basic structure provides the solution architecture, aligned with the four basic questions (stated in Figure 2-8 on page 29).

Working from left to right, it is important to first get an inventory of the assets within the enterprise. The IBM Tivoli License Compliance Manager for z/OS and IBM Tivoli License Compliance Manager products provide the ability to identify the software that is currently installed in the z/OS and distributed environments.

The next phase is to look at usage information to see what software is actually being used and to map that software usage to license information. License compliance can then be determined as well as identifying areas where you might be over-licensed. Finally, all of this information must be reconciled with the contracts and other financial aspects of the business. The IBM Tivoli Asset Management for IT product provides a rich set of capabilities, including:

- Procurement
- Financial Management
- Contract Management

More information about the capabilities of the Tivoli License Compliance Manager products and IBM Tivoli Asset Management for IT product will be provided in the next part of this book.

2.3 Implementing the solutions with Tivoli products

Today, more than ever before, an enterprise must have an integrated IT asset management solution focused on the total asset management practice. Several products exist to serve the IT asset management professional. They are described briefly in the following sections.

2.3.1 Tivoli Asset Management for IT product

In order to implement an IT asset management policy, the supporting tools must be flexible and comprehensive enough to handle various asset types, including a wide variety of license contracts. This flexibility is critical to ensure compliance and provide the capabilities necessary to proactively manage the deployed hardware and software portfolio. The Tivoli Asset Management for IT product is based on the popular Maximo Enterprise Asset Management® offering from what was formerly MRO, until it was acquired by IBM. The Enterprise Asset Management offering has been adapted and enhanced to include functions and capabilities specific to IT asset management. The Enterprise Asset Management product for non-IT assets is still available from IBM and is continuing to be enhanced. That product is now made available as IBM Maximo Asset Management. Based on the Maximo Asset Management foundation, Asset Management for IT is one of the most configurable, process-driven solutions on the market, with functionality including:

- Asset Management:
 - Including hardware and software contracts, terms and conditions, and financials
 - Manage software licenses and associated financial information
 - Identification of installed software and usage
 - Identification of installed hardware
- License Optimization:
 - Identification of opportunities to consolidate or renegotiate licenses
 - Cost reduction through the elimination of unneeded software
- Compliance Management:
 - Identification and elimination of compliance problems prior to a vendor audit
 - Identification of installed software that is unlicensed or unauthorized
- Planning for upgrades:
 - Analysis of software for a hardware upgrade
 - Determining additional license cost for increased capacity

Contract management

Contract management includes tracking and managing contract-associated costs (including cost allocations to financial centers and so on) and the terms and conditions of usage associated with the contract. Tivoli Asset Management for IT acts as the fundamental point of reference for all users of the system and defines the duration of the contract, the parties involved, and the ability to proactively manage the re-negotiation process. This capability gives the purchasing function the information that it needs to be able to negotiate the best possible commercial terms.

The contract management module underpins the entire system and is used to track and manage overriding contracts. The contracts module will proactively monitor contract usage and will automatically initiate workflow when contract thresholds are approached. The solution will also monitor contract expiration dates and ensure that the relevant purchasing personnel are notified with enough lead time to be able to renegotiate terms when the contract is due for renewal.

Inventory management

A virtual storeroom for unused software licenses and automatic reorder levels enables the Tivoli Asset Management for IT user to handle complex software license transactions, including upgrades, software bundles (productivity suites and such), exchanges, and reissues. Inventory management allows users to maintain a virtual stockroom of uninstalled software licenses. This approach creates what are called *software pools* and allows for the management of all deployed (installed) and non-deployed software licenses. Procedures for booking items in and out of the store are included within the solution, and these procedures can be integrated with software distribution tools if required. Furthermore, the inventory management module allows for financial management of software licenses as well. When a software license is "issued", a general ledger transaction record (credit/debit) is created. In this fashion, charge backs and internal billing can be managed to ensure proper cost allocations are assigned to business units, regional offices, departments, or users.

Cost tracking, cross-charges, and charge backs

The primary driver for the majority of software asset management initiatives is financial. For this reason, Tivoli Asset Management for IT is designed to track and allocate costs accurately in accordance with the organization's financial system and policies.

Request management and self-service

An investment in a central software asset repository will not deliver the level of returns expected unless it is being continuously used and updated. Tivoli Asset Management for IT's request management functionality enables employees to request software and provides the ability to review, approve, and manage requests in a consistent and efficient manner.

Procurement and purchasing

The Procurement capabilities of Tivoli Asset Management for IT enable the full procurement process, purchase requisition, purchase order, receipt inspection, and final receipt and asset registration, to be managed and controlled. As a result, you can also monitor and report on all of the purchasing steps that are required to fulfill a change management event (purchase of a new software license). User-definable purchasing permissions can be configured to ensure that corporate policies regarding approval are enforced automatically, minimizing the amount of manual intervention necessary to move requests through the system. Furthermore, Tivoli Asset Management for IT can be integrated with Enterprise Resource Planning (ERP) purchasing modules and external e-commerce marketplaces for automated ordering.

2.3.2 Tivoli License Compliance Manager (TLCM)

Often, software products do not provide technical support for License Management activities, leaving product License Compliance Management completely up to the organizations buying those products.

The Tivoli License Compliance Manager provides software inventory, use metering, and license allocation services on Windows®, UNIX®, and OS/400® platforms. As an IBM solution for collecting data on distributed platforms, it can be scaled to meet the needs of large and small organizations and support the management of multiple organizations. Information about installed software and software use is collected from monitored computers by an agent that can be deployed on a range of Windows and UNIX platforms and is reconciled with license and contract information that you define. Information is stored in a central IBM DB2® database and can be accessed using a Web browser-based user interface.

2.3.3 Tivoli License Compliance Manager for z/OS

In similar fashion to TLCM for the distributed platform, Tivoli License Compliance Manager for z/OS (TLCM z/OS) identifies software inventory, measures use activity, and automatically links license entitlements to help manage software costs and license compliance in the mainframe environment. This Software Asset Management solution enables IT to align software spending with business priorities. With the information provided, organizations can reduce unnecessary software costs and compliance risk to allocate additional resources to priority projects.

The linked software inventory, use activity, and license entitlement information provided by Tivoli License Compliance Manager for z/OS enables IT to:

- Proactively manage and verify license compliance
- Identify software inventory with no or low use activity to reduce unnecessary license fees
- Monitor software use trends to plan financially optimal capacity upgrades and server consolidations
- Report on the millions of instructions per second (MIPS) capacity of each logical partition (LPAR) under which software is running
- Effectively prepare for contract negations and audits with key information
- > Plan the migration off of the old version and duplicate function software
- Enhance the charge-back process by allocating the expensive low use software to the appropriate department

Improve Disaster Recovery planning with software use activity information

Tivoli License Compliance Manager for z/OS solves three major information challenges that you must resolve for effective mainframe software asset management:

- Identifies software inventory
- Measures product use activity by LPAR
- Automatically links license entitlements to installed inventory and use activity

2.3.4 Tivoli Asset Compliance Center (TACC)

The Tivoli Asset Compliance Center component of IBM Tivoli License Compliance Manager for z/OS provides a comprehensive user interface to work with the output of TLCM z/OS. This interface features a convenient, easily customizable layout of panes that displays information for you to use the product effectively. It provides fields in which you can enter information, search for data, and perform other tasks. The GUI offers two views:

- Asset Compliance perspective, which helps you efficiently view and manage licenses, discovered hardware, and discovered software. The Asset Compliance perspective gives you one place to go to manage all of your IT assets. You can track hardware, software, and licenses across your environment. You can also find opportunities for underused software, ensuring that your organization is obtaining the maximum benefit from existing software licenses.
- Administration perspective, which enables an administrator to establish and control user access to licenses, software inventories, and other business objects. With the Administration perspective, you can easily define domains, roles, groups, and users, efficiently ensuring that the right people have access to the data that they need.

2.4 Summary

The potential benefits of implementing an effective IT asset management policy and approach can be significant. These benefits are not only realized financially but also through improvements to the robustness and resilience of the entire IT infrastructure.

This non-exhaustive list highlights many of the key areas where benefits can be achieved:

- Maintain license compliance:
 - Avoid audits, non-compliance fines, and associated negative publicity

- Identify the level of corporate exposure due to pre-existing unapproved software usage
- ► Reduce software purchasing costs:
 - Negotiate improved terms and conditions with software and hardware vendors at initial purchase and subsequent contract renewal
 - Deploy only the assets that are actually required
 - Downgrade casual software users to standard versions, freeware viewers, and open source applications
 - Redeploy installed, but unused software and hardware to users that need it
- Better purchasing decisions:
 - Make informed IT purchases
 - Recycle previously purchased software and hardware internally
 - Streamline[™] software license provisioning practices
 - Eliminate maverick buying
- Reduce software maintenance and support costs:
 - Only pay maintenance on software that is actually in use
 - Fewer incident support calls caused by unapproved software or hardware
 - Fewer software packages or hardware platforms mean technicians need less training and can resolve problems quicker
 - Reduce and mitigate the risks associated with implementing planned changes

Part 2

Product Deployment



3

Tivoli Asset Management for IT

Tivoli Asset Management for IT V 7.1 is one of several products integrated on Tivoli's process automation engine. Tivoli's process automation engine is what used to be known as the base services capabilities of the Maximo Asset Management product. Since acquiring the Maximo product set, IBM has adapted and adopted the core functions to provide a consistent platform on which to implement critical service management applications, such as Service Request Manager, Change and Configuration Management Database, Maximo Asset Management, and the Tivoli Asset Management for IT products.

Tivoli's process automation engine is based on a Java[™] 2 Platform Enterprise Edition (J2EE[™]) architecture. Its primary logic executes in an application server (either IBM WebSphere® or BEA WebLogic). It also utilizes a relational database (such as DB2, Oracle®, or SQLServer).

Each product installed on top of Tivoli's process automation engine is enabled through its own license. Therefore, the single platform can provide support for various applications within the service management discipline and provide integration through the user interface and by also sharing data as appropriate and meaningful.

The user interface for Tivoli's process automation engine-based applications is provided through a Web browser. The standard interface looks similar for all

applications, with a standard set of menus. However, each application provides its own functions and adds menus and dialogs specific to the application. The user interface can also be customized by individual clients to provide their own look and feel as desired.

You can consider Tivoli's process automation engine as an application in itself, but we often talk about groups of functionality within Tivoli's process automation engine as "applications."

Within Tivoli's process automation engine, there is a "Go To" menu (Figure 3-1) through which a user accesses the various applications. Different users will see different subsets of this menu depending on their authorizations in Tivoli's process automation engine security and the licenses applied to the server.

When we talk about modules, we are often referring to the way that applications are grouped on the Go To menu. For instance, there is an Assets application, a Locations application, and several other applications within the Asset Management for IT module of Tivoli's process automation engine.



Figure 3-1 Menu structure of Tivoli Asset Management for IT

The Tivoli Asset Management for IT license key enables access to a specific set of applications, but it also includes many applications from Tivoli's process automation engine's "Base Services". Tivoli's process automation engine is the set of applications that comes with all licensing packages and allows the integration of the packages with each other and with external systems.

3.1 Management disciplines

One way to organize the sets of applications and modules that make up Tivoli Asset Management for IT and Tivoli's process automation engine is in terms of several traditional management disciplines, as shown in Figure 3-2.



Figure 3-2 Modules associated with Tivoli's process automation engine

Each of these disciplines interacts with the other disciplines through the exchange of information and the partial overlap of processes, but for the sake of discussion, we discuss each discipline separately. We will define each of these disciplines briefly here.

Procurement management looks at how to improve efficiencies in procurement processes and leverage vendor relationships. Tivoli's process automation engine allows for visibility into the processes by providing an integrated framework for procurement information. Control is usually achieved through integration with an external procurement system or ledger and through the configuration of Tivoli's process automation engine's workflow features to automate and track repetitive tasks involving approvals. Procurement management depends on good contracts management, the process through which vendors are approved and the prices are negotiated. Contracts management can also involve decisions about whether to lease or purchase assets, labor negotiations, and volume software agreements.

When items are procured, they can be managed through the disciplines of asset management or materials management. Asset management involves accounting for fixed assets, maintaining assets over their life cycles, and tracking asset costs in order to spot opportunities for increased efficiencies or savings. Materials management involves managing inventory and finding the best inventory balances to fulfill the organization's needs.

Work management is similar to project management in that it estimates and tracks the time and costs that are required to complete tasks. In fact, there are several integration packages available to move information between Tivoli's process automation engine's work management applications and enterprise project management tools. In addition to project management, work management includes the preventive maintenance and the recurring tasks that are required for regulatory compliance or to extend the life of assets, as well as work requests.

Work management is integrated with asset management in that the cost of the work performed can be rolled up into the total life cycle cost of the asset being maintained. It is integrated with materials management in that reservations for the materials needed for work can be automatically placed into the inventory applications, so that reordering can occur with sufficient lead time if needed. It is integrated into contracts management in that labor contracts can be referenced, automatically referencing appropriate labor rates and incrementing balances. The workflow engine from Tivoli's process automation engine helps to automate repetitive tasks and approvals.

Service management augments work management by providing a central clearinghouse for requests from throughout an organization. A *service catalog* is a way of looking at the services that support organizations provide for clients in terms that both the support organization and the clients understand. Service Catalogs can be integrated with operational level agreements or service level agreements to specify the particular levels of service to be expected and the costs for providing those levels. Throughout service management, there is an emphasis on continually updating and improving services by looking at the process and the root causes of problems.

Configuration management overlaps with asset management in that it requires a list of assets being tracked, but it focuses more on the relationships and dependencies between the configuration items that make up the assets, where asset management focuses on the financial details of the assets.

There are certain management disciplines that are not covered in Tivoli's process automation engine. For instance, while Tivoli's process automation engine records double-entry transaction records related to each of the management disciplines mentioned, the platform cannot be substituted for a general ledger system. Instead, the platform provides tools for integration with ledger systems. Tivoli's process automation engine also provides only the most basic functionality for budget management, client management, and application and other product development management, instead relying on the ability to flexibly integrate with external solutions.

Another management discipline that is not covered in Tivoli's process automation engine, but is of particular note for IT departments, is network management. Network devices can be discovered and fruitfully tied to authorized asset information within Tivoli's process automation engine. Also, events from network monitoring tools can be brought into Tivoli's process automation engine framework for uniform resolution from a work and service management. Visualization of dependencies for network equipment can be aided through Tivoli CCMDB. However, at this point, visualization of network topology and performance and configuration of infrastructure are best performed through specialized network management tools outside of Tivoli's process automation engine.

3.2 Data loading and the Integration Composer

The previous section provided a high-level functional overview of the Tivoli Asset Management for IT product. One of the key questions raised by clients wanting to implement an IT asset management solution is how to initially load data about existing assets into Tivoli Asset Management for IT.

One of the key tools to load this data is the IBM Tivoli Integration Composer. This software transfers software license, hardware, and software data to Tivoli Asset Management for IT from existing tools and repositories. It is a likely choice for organizations requiring data integration between Tivoli Asset Management for IT and Tivoli License Compliance Manager, but it also enables integration from virtually any other data store containing IT asset information.

The IBM Tivoli Integration Composer was formerly known as Fusion and is a stand-alone tool that provides the capability to map data from one data store to another data store. In our case, we create a map of the data in another data store, such as the data store used by Tivoli License Compliance Manager, and import the appropriate fields into the Tivoli Asset Management for IT database.

There are a variety of pre-build adapters available for common discovery and asset repository tools, including:

- Altiris Inventory Solution
- Centennial Discovery
- Maximo Discovery
- Microsoft Systems Management Server (SMS)
- Tivoli Application Dependency Discovery Manager
- Tivoli Configuration Manager
- ► Tivoli License Compliance Manager
- Tivoli License Compliance Manager for z/OS
- Tivoli Provisioning Manager
- IBM Tivoli Network Manager IP Edition

In addition, you can define your own mapping and adapter to load data into Tivoli Asset Management for IT from other tools or repositories that you might use.

For more information related to the installation and use of the IBM Tivoli Integration Composer (ITIC), refer to *IT Asset Management Processes for Tivoli Asset Management for IT*, SG24-7601.

3.3 Installation overview

In this section, we provide an overview of the installation of IBM Tivoli Asset Management for IT. We illustrate the steps that will enable you to configure and install the product in a single server stand-alone environment. We perform this installation with the following software components:

- Windows 2003 Server
- WebSphere 6.1 Application Server
- DB2 Database Platform
- IBM Tivioli Asset Manager for IT

Note that in a production environment, you might choose to utilize multiple servers. You also have the choice of using various middleware components, including the BEA Weblogic application server and Oracle or Microsoft SQLServer database products.

3.3.1 Installation flowchart

The flowchart shown in Figure 3-3 on page 47 defines the installation flow required to successfully install IBM Tivoli Asset Management for IT. There is a launchpad utility that will lead you through the installation of various components of the product. In general, there are multiple phases of the installation:

1. Installing the required middleware

Through the launchpad, you can install an appropriate configuration of WebSphere, DB2, and Tivoli Directory Server. Although you can change default parameters, if you take the default options, the installation of these middleware components through the launchpad is simple and straightforward.

2. Installing Tivoli's process automation engine (also referred to as *base services*). As mentioned earlier, Tivoli Asset Management for IT is based on the Maximo Asset Management product. There are a set of common services that provides the general functions for all Maximo-based applications. These base services are called Tivoli's process automation engine. These services comprise a set of modules that sit on an application server (such as WebSphere) and a database (often referred to as the Maximo database).

If you used the launchpad to install the middleware, the launchpad will "remember" the configuration that was installed and utilize that configuration to simplify the installation of Tivoli's process automation engine.

If you installed the middleware separately and used other products, such as Oracle or BEA Weblogic, you can still use the launchpad, but will have to specify the appropriate parameters for the middleware environment that you have installed.

- 3. The IBM Tivoli Integration Composer is a utility that enables the importing and exporting of data to and from the Maximo database. This utility is key to being able to bring asset-related data from other products, such as the Tivoli License Compliance Manager or other data stores currently used to manage IT assets, into the database used by Tivoli Asset Management for IT. The launchpad provides the capability to install this utility as part of the overall solution installation.
- 4. The next phase of the installation is to add the Asset Management for IT specific capabilities to Tivoli's process automation engine installed in step 2. Again, the launchpad will guide you through this process, helping to ensure that all appropriate parameters are specified for your target environment.
- 5. The last option available through the launchpad is to install any language packs that are required based on the geography and the preferences of the users.



Figure 3-3 Installation flowchart

3.4 Middleware installation

The Tivoli Asset Manager for IT utilizes and sits on top of a set of middleware. There is flexibility in the brand and in the versions of the middleware. However, IBM provides a middleware installer that installs and configures IBM DB2, IBM WebSphere, and IBM Tivoli Directory Server. For new installations, the middleware installer can save a lot of time and effort in ensuring that the prerequisite environment is prepared for the actual installation of Tivoli Asset Management for IT. We describe the use of the middleware installer next. If you choose to use an existing environment or install your own middleware, you can skip this next section and refer to your middleware product documentation for its installation process.

3.4.1 Installing middleware

Before you can install IBM Tivoli Asset Management for IT, you must deploy several middleware products.

Specifically, you must install and deploy the following software:

Database server

Tivoli Asset Management for IT uses a database (we will refer to this database as the Maximo database) to store details about the attributes and history of each configuration item and the details about the relationships between configuration items.

If you do not utilize the middleware installer, you can manually install a new instance of DB2 UDB 9.1, use a preexisting instance of DB2 UDB 8.2 or DB2 UDB 9.1, or install and configure Oracle 10 or MS SQL Server® 2005 for your Tivoli Asset Management for IT deployment.

Directory server

The directory server is used to secure the Java 2 Platform Enterprise Edition (J2EE) application. You can choose to configure a preexisting Microsoft Active Directory® server.

J2EE server

The J2EE server is the application server that is used to serve and manage the Tivoli Asset Management for IT application.

3.4.2 Installing middleware

To install the prerequisite middleware products using the provided middleware installer, follow these steps:

- 1. Log in as a user with administrative authority.
- 2. Launch the Tivoli middleware installer from the launchpad from the appropriate CD image.

Figure 3-4 will be displayed.

| 🍟 IBM Tivoli Asset Management f | or IT 7.1 |
|--|---|
| IBM software | |
| | Language selection: English |
| Welcome | Welcome to the IBM Tivoli Asset Management for IT 7.1 Installation |
| Installation Planning Install the Product Information Center Exit | Velcome to the IBM Tivoli Asset Management for IT 7.1 Installation Use this launch pad to install IBM Tivoli Asset Management for IT 7.1. When installing IBM Tivoli Asset Management for IT 7.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional installs, including the IBM Tivoli Integration Composer and the Tivoli Asset Management for IT 2.1, you will need to launch additional language pack if you are installing additional languages besides English. To get started, select a language from the dropdown list. Then use the links at left to review the product prerequisites and install the product. For detailed product information, link to the information center. |
| | |

Figure 3-4 Initial launchpad window

3. From the leftmost menu, select **Install the Product**. A list of options will be presented as shown in Figure 3-5. Select **Middleware** to install the required middleware for Tivoli Asset Management for IT.



Figure 3-5 Installation menu

4. Select a language for the installation and click **OK** (Figure 3-6).



Figure 3-6 Language selection

5. From the Welcome panel, click **Next**.

| IBM. | Welcome to the Tivoli Middleware Insta | ıller |
|---------------|--|--|
| | The Tivoli Middleware Installer can install or uninstall the midd base services. | leware used by Tivoli |
| | To install middleware, you must create a deployment plan . A d consists of deployment choices and configuration parameters. Middleware Installer will assist you in providing. After you have configured the plan, the middleware associated with your depl installed. To continue, choose Next. | eployment plan , which the Tivoli successfully oyment choices will be |
| | | |
| InstallShield | < Back Next > | Cancel |
| | | |

Figure 3-7 Welcome page

 The Tivoli middleware installer license agreement window is displayed. Read the license information and select I accept both the IBM and the non-IBM terms if you agree with the terms. Click Next.

| International Program License Agreement Part 1 - General Terms BY DOWNLOADING, INSTALLING, COPYING, ACCES PROGRAM YOU AGREE TO THE TERMS OF THIS AG ACCEPTING THESE TERMS ON BEHALF OF ANOTH COMPANY OR OTHER LEGAL ENTITY, YOU REPRES THAT YOU HAVE FULL AUTHORITY TO BIND THAT P LEGAL ENTITY TO THESE TERMS. IF YOU DO NOT A - DO NOT DOWNLOAD, INSTALL, COPY, ACCESS, O AND Read non-IBM terms | SSING, OR USING THE REEMENT. IF YOU ARE ER PERSON OR A SENT AND WARRANT ERSON, COMPANY, OR GREE TO THESE TERMS R USE THE PROGRAM; |
|--|---|
| Part 1 - General Terms BY DOWNLOADING, INSTALLING, COPYING, ACCES PROGRAM YOU AGREE TO THE TERMS OF THIS AG ACCEPTING THESE TERMS ON BEHALF OF ANOTH COMPANY OR OTHER LEGAL ENTITY, YOU REPRES THAT YOU HAVE FULL AUTHORITY TO BIND THAT P LEGAL ENTITY TO THESE TERMS. IF YOU DO NOT A - DO NOT DOWNLOAD, INSTALL, COPY, ACCESS, O AND Read non-IBM terms | SSING, OR USING THE REEMENT. IF YOU ARE ER PERSON OR A SENT AND WARRANT ERSON, COMPANY, OR GREE TO THESE TERMS R USE THE PROGRAM; |
| BY DOWNLOADING, INSTALLING, COPYING, ACCES PROGRAM YOU AGREE TO THE TERMS OF THIS AG ACCEPTING THESE TERMS ON BEHALF OF ANOTH COMPANY OR OTHER LEGAL ENTITY, YOU REPRES THAT YOU HAVE FULL AUTHORITY TO BIND THAT P LEGAL ENTITY TO THESE TERMS. IF YOU DO NOT A - DO NOT DOWNLOAD, INSTALL, COPY, ACCESS, O AND Read non-IBM terms | SSING, OR USING THE REEMENT. IF YOU ARE ER PERSON OR A SENT AND WARRANT 'ERSON, COMPANY, OR GREE TO THESE TERMS |
| Read non-IBM terms | |
| | |
| I accept both the IBM and the non-IBM terms | |
| C I do not accept the terms in the license agreement | t |
| Print | |
| | |
| Shield | |

Figure 3-8 License agreement

- 7. If the Tivoli middleware installer cannot locate the directory containing the middleware installation program files, you will be prompted to supply the directory that contains them. The IBM middleware product files are available on another DVD included in the package. Supply the proper directory, and then click **Next**. This panel will only appear if the Tivoli middleware installer cannot locate the directory containing the middleware installation program files.
- 8. From the Choose Workspace panel, specify the directory that you will use as the Tivoli middleware installer workspace, and then click **Next**.

The default location for the workspace will be the last workspace location used by this user, as specified in the middleware user preferences node. If no previous workspace location exists in the middleware user preferences node, the default location for the workspace will be C:\ibm\tivoli\mwi\workspace for Windows. If the selected directory does not exist, it will be created. After deployment, the Tivoli middleware installer also generates a topology file in this directory. This topology file can be manually copied by the user to the workspace of the next machine in the topology, so that information about the
deployment of the middleware will be available to the Tivoli middleware installer when it is executed on the next machine.

- 9. From the Deployment Choices panel (Figure 3-9), select the features to deploy on this machine, and then click **Next**. Choices include:
 - Database Server: This server will manage the Maximo database that is used to store information about assets.
 - Directory Server: Data maintained by the directory server is used to secure Tivoli Asset Management for IT.
 - J2EE Server: The J2EE server is used to host and manage the Tivoli Asset Management for IT application. If you choose to only install the J2EE server portion of the middleware, you will be prompted to supply the directory server that you will use to secure it. Your choices will be to secure it with an existing instance of IBM Tivoli Directory Server or an existing instance of Microsoft Active Directory.

| IBM. | Select the features to deploy on the local machine. |
|------------|---|
| | ✓ <u>Database Server</u> The Database Server is used to store details about the attributes and history of each configuration item and the details about the relationships among configuration items. |
| | Directory Server The Directory Server is used to secure the J2EE Server. This feature should be selected to either install a new directory server locally or reuse a local directory server. |
| | ✓ J2EE Server The J2EE Server is the application server used to serve and manage the application. ✓ Secure the J2EE Server using the Directory Server |
| | |
| tallShield | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-9 Middleware selection

10. From the Deployment Plan Summary window, click **Next** to configure the parameters for this plan. The deployment plan is generated, and you will be provided details about the plan (Figure 3-10).

| IBM. | The following plan has been generated. To enter configuration parameters for this plan, choose Next. | |
|------------|---|--|
| | IBM Rational Agent Controller Version 7.0.3 DB2 Enterprise Server Edition Version 9.1.2 Configuration for DB2 Enterprise Server Edition Configuration of DB2 Enterprise Server Edition for reuse IBM Tivoli Directory Server Version 6.1 Configuration for IBM Tivoli Directory Server WebSphere Application Server ND Version 6.1.0.9 Configuration for WAS ND IBM HTTP Server Version 6.1 Embedded Security Services version 6.1 | |
| tallShield | | |
| | < Back Next > Cancel | |
| | | |

Figure 3-10 Deployment plan

11. From the Credentials panel (Figure 3-11), enter the Username and Password with which you will deploy the plan, and then click **Next**.

You can choose to enable the option of using the same password as the default user password value in all panels of the Tivoli middleware installer.

| IBM. | Use this passw This will overrid fields during th the passwords | word as the value t de any previously (is installation as a ; later. | for all subsequent passw configured values and dis a convenience to get starte | ords. able all password ed. You may change |
|---------------|--|--|--|--|
| | Password: | : | | |
| | | | | |
| InstallShield | | < <u>B</u> ack | <u>N</u> ext ≻ | <u>C</u> ancel |

Figure 3-11 Middleware password selection

- 12.Enter the following configuration parameters (Figure 3-12) for DB2 Enterprise Edition Server, and then click **Next**:
 - Install location: Enter the location to install DB2.

Windows: The default is C:\Program Files\IBM\SQLLIB

DB2 Administration Server username: Enter the DB2 administrative account name.

Windows: The default is db2admin

 DB2 Administration Server password: Enter the password for the DB2 administrative account.

| IBM. | Enter the configuration parameters for DB2 Enterprise Server Edition Version 9.1.2. |
|------------|---|
| | Install location: |
| | Browse |
| | db2admin |
| | DB2 Administration Server password: |
| | Confirm password: |
| | |
| tallShield | - Deale Nexts Concel |
| | < Back <u>Next > C</u> ancel |

Figure 3-12 DB2 Enterprise Edition configuration parameters

- 13.Enter the following configuration parameters (Figure 3-13) for the Maximo database instance, and then click **Next**:
 - Instance name: Enter the name of the Maximo database instance.

The default for all platforms is ctginst1.

- Port: Enter the port that the Maximo database instance will use.

The default for all platforms is 50005.

Instance username: Enter the user name for the Maximo database instance.

Windows: The default is db2admin.

Instance username password: Enter the password for the Maximo database instance user name.

| IBM. | Enter the configuration parameters for Database Instance. |
|------------|---|
| | Instance name: |
| | ctginst1 |
| | Port |
| | 50005 |
| | Instance username: |
| | db2admin |
| | Instance user password: |
| | ****** |
| | Confirm password: |
| | ****** |
| | |
| | |
| tallShield | |
| | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-13 Database instance configuration

14. Enter information about the DB2 user groups (Figure 3-14):

DB2 administrators group: Enter the name of the DB2 administrators group.

Windows: The default is DB2ADMNS.

DB2 users group (Windows only): Enter the name of the DB2 users group.

The default is DB2USERS.

| IBM. | Enter the configuration parameters for DB2 Enterprise Server Edition. |
|-----------|---|
| | DB2 administrators group: DB2ADMNS DB2 users group: DB2USERS |
| allShield | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-14 DB2 user groups configuration

- 15. Enter the following configuration parameters (Figure 3-15) for IBM Tivoli Directory Server, and then click **Next**:
 - Install location: Enter the location to install IBM Tivoli Directory Server.

Windows: The default is C:\Program Files\IBM\LDAP\V6.1

- Administrator distinguished name: Enter the distinguished name of the IBM Tivoli Directory Server administrator. The default for all platforms is cn=root.
- Administrator password: Enter the password for the IBM Tivoli Directory Server administrator.

| | Install location: | |
|-----------|-----------------------------------|-----------------|
| | C:\Program Files\IBM\LDAP\V6.1 | |
| | | B <u>r</u> owse |
| | Administrator distinguished name: | |
| | cn=root | |
| | Administrator password: | |
| | ****** | |
| | Confirm password: | |
| | ***** | |
| | | |
| | | |
| | | |
| Shield —— | | |

Figure 3-15 IBM Tivoli Directory Server configuration (panel 1 of 2)

- 16. Enter the following configuration parameters (Figure 3-16 on page 60) for IBM Tivoli Directory Server, and then click **Next**:
 - Organizational unit: Enter the name of the IBM Tivoli Directory Server organizational unit to use with Tivoli Asset Management for IT. The default for all platforms is ou=SWG.
 - Organization and country suffix: Enter the name of the IBM Tivoli Directory Server organization and country suffix to use with Tivoli Asset Management for IT. The default for all platforms is o=IBM,c=US.

- Directory server port: Enter the port number of the IBM Tivoli Directory Server. The default for all platforms is 389.
- Directory server secure port: Enter the secure port number of the IBM Tivoli Directory Server. The default for all platforms is 636.
- Administration port: Enter the administration port number of the IBM Tivoli Directory Server. The default for all platforms is 3538.
- Administration secure port: Enter the secure administration port number of the IBM Tivoli Directory Server. The default for all platforms is 3539.

| Organizational unit: |
|----------------------------------|
| ou=SWG |
| Organization and country suffix: |
| o=IBM,c=US |
| Directory server port: |
| 389 |
| Directory server secure port: |
| 636 |
| Administration port: |
| 3538 |
| Administration secure port: |
| 3539 |
| |

Figure 3-16 Parameters for IBM Tivoli Directory Server configuration (panel 2 of 2)

- 17.Enter the following configuration parameters (Figure 3-17) for IBM Tivoli Directory Server database instance, and then click **Next**:
 - Database name: Enter the name of the DB2 database that you are using to hold IBM Tivoli Directory Server data. The default for all platforms is security.
 - **Instance name**: Enter the name of the IBM Tivoli Directory Server database instance. The default for all platforms is idsccmdb.
 - Port: Enter the port number used by the IBM Tivoli Directory Server database instance. The default for all platforms is 50006.
 - Instance user password: Enter the password for the instance user ID.

| IBM. | Enter the configuration parameters for IBM Tivoli Directory Server Database Instance. |
|-------------|---|
| | Database name: |
| | security |
| | Instance name: |
| | idsccmdb |
| | Port: |
| | 50006 |
| | Instance user password: |
| | ***** |
| | Confirm password: |
| | ***** |
| | |
| | |
| allShield — | |
| | < <u>Back</u> <u>N</u> ext> <u>C</u> ancel |

Figure 3-17 IBM Tivoli Directory database instance configuration

- 18. Enter the following configuration parameters (Figure 3-18) for WebSphere Application Server security, and then click **Next**:
 - LDAP Host Name: Enter the host name of the system hosting the Lightweight Directory Access Protocol (LDAP) instance to use for WebSphere security.
 - Directory server port: Enter the port number used by the LDAP server to use for WebSphere security. The default is 389.
 - LDAP base entity: Enter the LDAP base entity of the LDAP instance to use for WebSphere security. The default is ou=SWG,o=IBM,c=US.
 - User suffix: Enter the user suffix of the LDAP instance to use for WebSphere security. The default is ou=users,ou=SWG,o=IBM,c=US.
 - Group suffix: Enter the group suffix of the LDAP instance to use for WebSphere security. The default is ou=groups,ou=SWG,o=IBM,c=US.
 - Organization container suffix: Enter the organizational container suffix of the LDAP instance to use for WebSphere security. The default is ou=SWG,o=IBM,c=US.

| | Enter the configuration parameters for websphere Application Server Security. |
|-----------|---|
| | LDAP Hostname: |
| | fenway.itsc.austin.ibm.com |
| | Directory server port: |
| | 389 |
| | LDAP base entry: |
| | ou=SWG,o=IBM,c=US |
| | User suffix: |
| | ou=users,ou=SWG,o=IBM,c=US |
| | Group suffix: |
| | ou=groups,ou=SWG,o=IBM,c=US |
| | Organization container suffix: |
| | ou=SWG,o=IBM,c=US |
| | |
| Shield —— | |
| | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-18 WebSphere Application Server security configuration (panel 1 of 2)

- 19.Enter the following configuration parameters (Figure 3-19) for WebSphere Application Server security, and then click **Next**:
 - Bind distinguished name: Enter the bind distinguished name for binding to the LDAP instance. The default is cn=root.
 - Bind password: Enter the password for the bind distinguished name.

| IBM. | Enter the configuration parameters for WebSphere Application Server security. |
|----------|---|
| | Bind distinguished name: |
| | cn=root |
| | Bind password: |
| | ****** |
| | Confirm password: |
| | ****** |
| | |
| | |
| | |
| | |
| | |
| IIShiold | |
| | r Book Novt > Consol |
| | |

Figure 3-19 WebSphere Application Server security configuration (panel 2 of 2)

- 20.Enter the following configuration parameters (Figure 3-20) for WebSphere Application Server Version, and then click **Next**:
 - **Install location**: Enter the location to install WebSphere:

Windows default is C:\Program Files\IBM\WebSphere\AppServer.

Linux® default is /opt/IBM/WebSphere/AppServer.

- WebSphere Administration username: Enter the WebSphere administrative account name. The default for all platforms is wasadmin.
- WebSphere Administration password: Enter the password for the WebSphere administrative account.

| IBM. | Enter the configuration parameters for WebSphere Application Server ND Version 6.1.0.9. |
|---------------|---|
| | Install location: |
| | C:\Program Files\IBM\WebSphere\AppServer |
| | Browse |
| | Administrator username: |
| | wasadmin |
| | Administrator password: |
| | ***** |
| | Confirm password: |
| | **** |
| | |
| InstallShield | · · |
| | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-20 WebSphere Application Server ND version configuration

- 21.Enter the following configuration parameters (Figure 3-21) for WebSphere Application Server, and then click **Next**:
 - Deployment Manager profile name: Enter the WebSphere profile name of the deployment manager server. The default for all platforms is ctgDmgr01.
 - Application server profile name: Enter the WebSphere profile name of the application server. The default for all platforms is ctgAppSrv01.

| IBM. | Enter the configuration parameters for WebSphere Application Server ND Version 6.1.0.9. |
|------------|---|
| | Deployment Manager profile name: |
| | ctgDmgr01 |
| | Application server profile name: |
| | ctgAppSrv01 |
| tallShield | |
| | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-21 WebSphere Application Server configuration parameters

- 22. Enter the following configuration parameters (Figure 3-22) for WebSphere Application Server, and then click **Next**:
 - Cell name: Enter the WebSphere Cell name. The default for all platforms is ctgCell01.
 - Deployment Manager node name: Enter the name of the WebSphere deployment manager node. The default for all platforms is ctgCellManager01.
 - Application server node name: Enter the name of the WebSphere application server node. The default for all platforms is ctgNode01.
 - Update Installer install location: Enter the location where the WebSphere update installer will be installed.

Windows: The default is C:\Program Files\IBM\WebSphere\UpdateInstaller.

| | IBM. | Enter the configuration parameters for WebSphere Application Server ND Version 6.1.0.9. |
|----------|------|---|
| | | Cell name: |
| | | ctgCell01 |
| | | Deployment Manager node name: |
| | | ctgCellManager01 |
| | | Application server node name: |
| | | ctgNode01 |
| | | Update Installer install location: |
| | | C:\Program Files\IBM\WebSphere\UpdateInstaller |
| | | Browse |
| tallShie | bl | |
| anome | iu - | « Back Nexts Cancel |
| | | |

Figure 3-22 WebSphere Application Server ND version

- 23.Enter the following configuration parameters (Figure 3-23) for IBM HTTP Server, and then click **Next**:
 - Install location: Enter the location to install IBM HTTP Server.

Windows: The default is C:\Program Files\IBM\HTTPServer.

- HTTP port: Enter the port used by the IBM HTTP Server. The default for all platforms is 80.
- Admin Server port: Enter the port to use to administer IBM HTTP Server. The default for all platforms is 8008.

| IBM. | Enter the configuration parameters for IBM HTTP Server Version 6.1. | |
|-----------|---|-----------------|
| | | |
| | | |
| | Install location: | |
| | C:\Program Files\IBM\HTTPServer | |
| | | B <u>r</u> owse |
| | HTTP port: | |
| | 80 | |
| | Admin Server port: | |
| | 8008 | |
| | | |
| | | |
| | | |
| allShield | | |
| | < Back Next > | Cancel |
| | | <u>_</u> |

Figure 3-23 IBM HTTP Server configuration

24.Enter the following configuration parameters (Figure 3-24) for WebSphere Application Server plug-in for the IBM HTTP Server, and then click **Next**:

Profile name: Enter the profile name. The default for all platforms is ctgAppSvr01. This value cannot be changed.

| IBM. | Enter the configuration parameters for IBM WebSphere Application Server plugin for the IBM HTTP Server. |
|------------|---|
| | Profile name: |
| | ctgAppSiv01 |
| | |
| tallShield | |
| tallShield | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

Figure 3-24 IBM HTTP Server plug-in configuration

25. Enter the following configuration parameters (Figure 3-25) for IBM Agent Controller, and then click **Next**:

Install location: Enter the location to install IBM Agent Controller.

Windows: The default is C:\Program Files\IBM\AgentController.

| IBM. | Enter the configuration parameters for IBM Rational Agent Controlle | er Version 7.0.3. |
|------|---|-------------------|
| | Install location: | |
| | | Browse |
| | | |
| | | |

Figure 3-25 IBM Rational Agent Controller configuration

- 26.Specify the location of the middleware images (Figure 3-26), and then click **Next:**
 - Copy the middleware install images from the source media to a specified directory. Select this option to copy the middleware images from the product media to a directory that you will specify.
 - Specify a directory containing all the required middleware install images. Select this option if you intend to specify a file system directory that already contains all of the middleware installation images.

| WS-WAS_UpdateInstaller_6.1.0_FP13/download.updii.61013. windows.ia32.zip DB2-ESE_9.1.0_FP4/v9fp4_win_ese.exe WS-ESS_6.1_GA/IBMESSAuthnSvc.ear WS-ESS_6.1_GA/com.ibm.security.ess.server_config.6.1.0.jar WS-WAS_IHS_6.1.0_FP13/6.1.0-WS-IHS-WinX32-FP0000013.pak TIV-DirectoryServer_6_1_0/tds61_win_is32-CD1_w_entitlem |
|--|
| Select an option for the required middleware install images. |
| C Copy the images. |
| Copy the middleware install images from the source media to a specified directory. |
| Specify a directory Specify a directory containing all the required middleware install images. |
| |
| < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |
| |

Figure 3-26 Middleware product images location

27. If you selected the option to copy install images from the source media, specify the source and destination directories, and then click **Next**. If you selected the option to specify a directory that already contained the middleware images, specify that directory (Figure 3-27), and then click **Next**.

| C:\mwIMAGES\windows | | | | | _ 🗆 🗡 |
|--|---------|-------------|-------------------|------------|----------|
| Eile Edit View Favorites Iools Help | | | | | 1 |
| 😋 Back 🝷 🕥 👻 🏂 🔎 Search 🌔 Folders 🛛 🕼 | 🐵 🗙 🍤 🛛 | | | | |
| Address C:\mwIMAGES\windows | | | | | 💌 🄁 Go |
| Name 🔺 | Size | Туре | Date Modified | Attributes | |
| DB2-ESE_9.1.0 | | File Folder | 4/12/2008 1:28 PM | | |
| DB2-ESE_9.1.0_FP4 | | File Folder | 4/12/2008 1:27 PM | | |
| Calibrian Rational-AgentController_7.0.3.1 | | File Folder | 4/12/2008 1:27 PM | | |
| TIV-DirectoryServer_6.1.0 | | File Folder | 4/12/2008 1:26 PM | | |
| TIV-DirectoryServer_6.1.0_FP0001 | | File Folder | 4/12/2008 1:25 PM | | |
| WS-ESS_6.1_GA | | File Folder | 4/12/2008 1:25 PM | | |
| 🚞 WS-WAS_IHS_6.1.0_FP13 | | File Folder | 4/12/2008 1:25 PM | | |
| BWS-WAS_ND_6.1.0.13_Custom_ISCAE71 | | File Folder | 4/12/2008 1:24 PM | | |
| WS-WAS_ND_6.1.0_Supplemental | | File Folder | 4/12/2008 1:22 PM | | |
| WS-WAS_Plugins_6.1.0_FP13 | | File Folder | 4/12/2008 1:22 PM | | |
| S-WAS_UpdateInstaller_6.1.0_FP13 | | File Folder | 4/12/2008 1:22 PM | | |

Figure 3-27 Middleware product image directory

28. Specify a directory to use for Tivoli middleware installer temporary files and extracted middleware installation images (Figure 3-28), and then click **Next**.

| mporary files directory: | |
|---------------------------------|---|
| DOCUME~1\ADMINI~1\LOCALS~1\Temp | |
| | Brows |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| er >:1 | emporary files directory: ::\DOCUME~1\ADMINI~1\LOCALS~1\Temp |

Figure 3-28 Middleware temporary files directory

29. From the Deployment Plan Operation panel (Figure 3-29), select **Deploy the plan**, and then click **Next**. You can also elect to make changes to the deployment plan or parameters that you have previously configured from this panel.

| IBM. | The deployment plan located at C:\ibm\tivoli\mwi\workspace\fenway.itsc.austin.ibm.com\deploymentPlan is ready to be deployed. Select an operation to perform upon this deployment plan. |
|-------------|---|
| | © Deploy the plan |
| | C Edit the configuration parameters |
| | C Delete the deployment plan and create a new one |
| | C Restart the plan |
| | C Undeploy the plan |
| | |
| | |
| stallShield | |
| | < <u>Back</u> |

Figure 3-29 Manage middleware deployment plan

30. From the Deployment Plan and Parameter Configuration summary panel (Figure 3-30), review the contents of the summary, and then click **Deploy** to initiate the installation and configuration of the middleware you selected.

| IBM. | The following plan will be deployed, and the html data will be saved to: C: \ibm\tivoli\mwi\workspace\parameterSummary.html | |
|----------------|--|----------|
| | Middleware Dependency Checks | 1 |
| | root.native.zipped.location root.native.deflated.location C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp localhost.allowed false | |
| | Middleware Disk Space Check | |
| | diskSpaceCheck.in_02.productID WAS_ND_6.1.0.5 diskSpaceCheck.in_05.productID ITDS_6.1 diskSpaceCheck.in_05.productID | <u>_</u> |
| stallShield | | |
| <u>F</u> inish | < <u>B</u> ack Deploy <u>C</u> ancel | |

Figure 3-30 Middleware deployment plan

31. The deployment will now run uninterrupted, which takes a while. In our environment, the deployment took about two hours to complete. After the deployment completes successfully, click **Finish** to exit (Figure 3-31).



Figure 3-31 Middleware installation completed

Tivoli middleware installer logs

Tivoli middleware installer log files are located in the workspace directory that was defined in the Tivoli middleware installer. Next, we describe the various types of log files.

User interface logs

The logs generated by the Tivoli middleware installer user interface are located in the workspace directory. The mwi.log file is the high-level log file that was generated by the most recent invocation of the Tivoli middleware installer. If an error occurs, examine this log file first. An entry in this log file can direct you to a lower level log file. Log files named mwi.log*X*, where *X* is a number, are copies of the mwi.log file from earlier invocations of the Tivoli middleware installer So, for example, mwi.log0 is produced after the first invocation of Tivoli middleware installer, mwi.log1 is produced after the second invocation of Tivoli middleware installer, and so on.

Logs for steps run by the user interface

In addition to collecting input from the user, the user interface of the Tivoli middleware installer also performs several system checks. Examples of system checks run by the user interface runs include:

- Dependency checking to ensure that the operating system meets the deployment requirements
- Inventorying the software on the system to locate existing instances of middleware products that are deployed by the Tivoli middleware installer
- Checking the available disk space to ensure that there is enough disk space for the deployment

Each of these system checks is produced in the form of a step so that it can also be run as part of the deployment plan. When the user interface runs a step, it copies the step into a subdirectory of the workspace directory. The log files generated by a step are located in the same subdirectory and follow the same pattern as a step that is run as part of the deployment plan.

Logs for the deployment plan

The deployment plan is located in the directory *<Workspace Directory>/hostname*/deploymentPlan, where *hostname* is the host name of the current system. Each time that the deployment plan is used to install or uninstall middleware products, a process ID is assigned and log files are generated.

The log files for the deployment plan are located in the subdirectory logs/processID. The primary log file for the deployment plan is DeploymentPlan.log, a high-level log file that lists the steps invoked as part of the deployment plan.

Logs for the machine plan

The machine plan is located in the directory *<Workspace Directory>*/hostname/deploymentPlan/MachinePlan_hostname. The log files for the machine plan are located in the logs subdirectory. The primary log files for the machine plan are named MachinePlan_hostname_processID. These log files contain the output generated by ANT when running the machine plan ANT script.

Logs for steps in the deployment plan

Each step in the deployment plan is located in a directory named <*Workspace Directory*>/hostname/deploymentPlan/MachinePlan_hostname/stepNum_stepID where stepNum is the sequence number of this step in the install processing order of the deployment plan, and stepID identifies the step. The log files for the step are located in the logs subdirectory.

Certain steps can provide a message log file named stepID_processID.message, which contains a few entries that summarize the result of invoking the step. All steps will provide a trace log file named stepID_processID.log, which contains many entries, usually including information about the input parameters and the sub-steps that are invoked.

Logs for sub-steps

Each step contains one or more sub-steps. The sub-steps perform the actual install and uninstall, and the sub-steps also check the work for the Tivoli middleware installer.

Each sub-step is located in the directory <*Workspace Directory*>/hostname/deploymentPlan/MachinePlan_hostname/stepNum_stepID /operation/substepNum_substepID, where operation is the ANT target in the step ANT script that invokes this sub-step. The substepNum is the sequence number of this sub-step in the processing order of the step, and substepID identifies the sub-step. Typical values for operation are install, uninstall, and check.

The log files for the sub-step are usually located in a subdirectory named processID/logs. Log files generated by the native middleware installation programs will also be kept here.

3.5 Installing the Tivoli's process automation engine for Asset Management for IT

This procedure explains how to use the launchpad installation program to install the base services (or Tivoli's process automation engine) required for Tivoli Asset Management for IT. In addition to configuring new instances of middleware products installed by the Tivoli middleware installer, the Tivoli Asset Management for IT installation program can configure existing instances of prerequisite products, including those products from other vendors, that you want to use with Tivoli Asset Management for IT. The instructions provided are for a single machine installation using default values and assume that you chose to have the Tivoli Asset Management for IT installation program automatically configure middleware across multiple machines to work with Tivoli Asset Management for IT. If you do not allow the Tivoli Asset Management for IT installation program to automatically configure middleware, it will still perform programmatic checks to verify that the documented manual steps were performed properly. If any errors are encountered, a dialog box detailing the error will appear. You will not be permitted to continue in the Tivoli Asset Management for IT installation task until the errors are resolved. The Tivoli Asset Management for IT installation program can only be run from a Windows-based system.

3.5.1 Installation steps

Avoid using localhost for host name values in the install program. Specify the actual fully qualified host name of the system for all host name values. To install Tivoli's process automation engine, follow these steps:

 Log in as a user with administrative authority. Launch the installer from the launchpad (Figure 3-32). In the launchpad navigation pane, click Install the Product → Tivoli Asset Management for IT 7.1. (Yes, we are installing Tivoli's process automation engine , even though the prompt is for Tivoli Asset Management for IT. Tivoli's process automation engine makes up the majority of Tivoli Asset Management for IT.)

| IBM Tivoli Asset Management for IT 7.1 |
|---|
| Install the middleware Select Middleware if you are using a DB2 Database and/or IBM WebSphere Application Server and have n them as a pre-requisite for Maximo. All other databases and Web Applications servers (e.g. BEA Weblogic, must be installed manually prior to the installation of Base Services. |
| Middleware Install IBM Tivoli Asset Management for IT 7.1 Select the Tivoli Asset Management for IT 7.1 link below launch the installer. |
| Tivoli Asset Management for IT 7.1 |
| Select the IBM Tivoli Asset Management for IT Language Pack Installer to add additional language suppo |
| Tivoli Asset Management for IT Language Pack installer |
| Install IBM Tivoli Integration Composer Select the Tivoli Integration Composer link below launch the installer. |
| <u>Tivoli Integration Composer</u> |
| 4. Enable IBM Tivoli Asset Management for IT License for usage Select Enable IBM Tivoli Asset Management for IT License for usage. The PSI License installation is required Tivoli Asset Management for IT functionality. |
| Enable IBM Tivoli Asset Management for IT License for usage |
| |
| |

Figure 3-32 Launchpad

2. At this point, the Tivoli Asset Management for IT installer is extracting (Figure 3-33 on page 79), wait until the task has completed.

| InstallAnyw | here | |
|---------------|--|--------|
| 1 | InstallAnywhere is preparing to install | |
| | | |
| | 3% | |
| | | Cancel |
| (C) 1997-2008 | Macrovision Europe Ltd. and/or Macrovision Corporation | |

Figure 3-33 InstallAnywhere extraction

3. Select a language for the installation (Figure 3-34) and click OK.



Figure 3-34 Language selection

4. From the Introduction panel (Figure 3-35), click Next.



Figure 3-35 Introduction panel

5. From the Software License Agreement panel (Figure 3-36), accept the General Terms and Conditions if you are in accordance with the terms provided.

| ease read the accompanying license agreement carefully copt the terms of this agreement. If you select "Decline", i | before using the Program. By selecting "Accept" below or using the Program, you agree nstallation will not be completed and you will not be able to use the Program. |
|---|--|
| International Program License Agreement | |
| Part 1 - General Terms | |
| BY DOWNLOADING, INSTALLING, COPYING, ACCESSIN AGREEMENT, IF YOU ARE ACCEPTING THESE TERMS ENTITY, YOU REPRESENT AND WARRANT THAT YOU ENTITY TO THESE TERMS. IF YOU DO NOT AGREE TO - DO NOT DOWNLOAD, INSTALL, COPY, ACCESS, OR U - PROMPTLY RETURN THE PROGRAM AND PROOF OF REFUND OF THE AMOUNT YOU PAID. IF YOU DOWNLO | IG, OR USING THE PROGRAM YOU AGREE TO THE TERMS OF THIS ON BEHALF OF ANOTHER PERSON OR A COMPANY OR OTHER LEGAL HAVE FULL AUTHORITY TO BIND THAT PERSON, COMPANY, OR LEGAL D THESE TERMS, ISE THE PROGRAM; AND ENTITLEMENT TO THE PARTY FROM WHOM YOU ACQUIRED IT TO OBTAIN A ADED THE PROGRAM, CONTACT THE PARTY FROM WHOM YOU ACQUIRED IT. |
| nom is international business machines corporation of o | |
| | Read non-IBM terms |
| | |
| | |
| | |

Figure 3-36 Software license agreement

- 6. From the Import Middleware Configuration Information panel (Figure 3-37 on page 82), specify that you want to use field values that you input into the Tivoli middleware installer as default values for those same fields in the Tivoli Asset Management for IT installation program:
 - Import Middleware Configuration Information: Select this check box if you want to allow the Tivoli Asset Management for IT installation program to reuse values entered in the Tivoli middleware installer. The middleware default information will not be used if you select the Simple deployment path.
 - Host name: Enter the host name of the system where the Tivoli middleware installer was run.
 - **User ID**: Enter the User ID that was used to run the Tivoli middleware installer.
 - Password: Enter the password of the User ID that was used to run the Tivoli middleware installer.

 Workspace Location: Enter the location of the topology file that contains the values entered for the Tivoli middleware installer. This file is found in the workspace that was defined during the middleware installation task, for example, C:\ibm\tivoli\mwi\workspace.

Click Next.

| 堰 IBM Tivoli base services | |
|--|---|
| | Import Middleware Configuration Information |
| | Check this box to import the configuration information recorded by the Tivoli middleware installation program. Provide the information for the machine where the Tivoli middleware installation workspace is located. Enter a fully qualified host name or IP address. The user ID must have remote access privileges to the machine. Import middleware configuration information Host name fenway.itsc.austin.ibm.com User ID Administrator Password ******* Workspace location C:\ibm\tivoli\mwi\workspace |
| InstallAnywhere by Macrovision – <u>C</u> ancel | Previous Next |

Figure 3-37 Import middleware configuration information

- 7. From the Choose Deployment panel (Figure 3-38), select the **Custom** deployment topology, and then click **Next**:
 - Select Simple if you want to deploy all Tivoli Asset Management for IT components on a single system. This deployment option is typically only used for demonstration, proof-of-concept, or training purposes.
 - Select Custom if you want to deploy Tivoli Asset Management for IT components across several systems. This deployment option is typically used in a production environment.

In our lab environment, we chose Custom to be able to explore and show the panels, even though we had a simple environment.

| 📲 IBM Tivoli base services | |
|----------------------------------|--|
| | Choose Deployment |
| | There are 2 deployment types, simple and custom. Simple deployment means the most common application features will be installed on a single machine and configured with default values. This option is recommended only for users who want to use defaults for all components, generally for a demonstration system. Custom deployment means the most common application features can be installed across multiple machines and are configurable. This option is recommended for most users. Simple Custom |
| InstallAnywhere by Macrovision – | Previous Next |

Figure 3-38 Choosing simple or custom deployment

- IBM Tivoli base services
 Please Wait

 Please Wait
 Please Wait

 Please wait, IBM Tivoli base services is being configured for your system. This may take a moment...
 InstallAnywhere by Macrovision

 Please Wait
 Please Wait
 InstallAnywhere by Macrovision
- 8. Wait until the base services have finished configuring (Figure 3-39).

Figure 3-39 IBM Tivoli base services configuration message

- 9. From the Choose Install Folder panel (Figure 3-40), specify the directory that you will use to install Tivoli Asset Management for IT, and then click **Next**:
 - Where Would You Like to Install? Enter the path to install Tivoli Asset Management for IT. By default, this value is C:\IBM\SMP. The path that you specify must not contain spaces.

| 🐙 IBM Tivoli base services | | | |
|----------------------------------|----------------------------------|--------------------------------|------------------|
| | | Choose | e Install Folder |
| | Where Would You Like to Install? | | |
| | C:\IBM\SMP | | |
| | | <u>R</u> estore Default Folder | Ch <u>o</u> ose |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| InstallAnywhere by Macrovision - | | 100 | |
| <u>C</u> ancel | | <u>P</u> revious | Next |

Figure 3-40 Choose installation folder

- 10.From the Maximo Database Type panel (Figure 3-41), select the platform that you will be using for the Maximo database, and then click **Next**:
 - DB2: Select this choice to use IBM DB2 UDB as the Maximo database.
 - Oracle: Select this choice to use Oracle as the Maximo database.
 - SQL Server: Select this choice to use Microsoft SQL Server 2005 as the Maximo database.

Each database will have its own unique set of configurable parameters and values. The panels in the next few figures are based on our selection of **DB2**.

| 堰 Database Configuration | | |
|----------------------------------|--|----------------------|
| | | Maximo Database Type |
| | Select the Maximo database type © DB2 © Oracle © SQL Server | Maximo Database Type |
| InstallAnywhere by Macrovision - | | Previous Next |

Figure 3-41 Maximo database type selection

- 11. From the Maximo Database panel (Figure 3-42), enter the configuration information about the database, and then click **Next**:
 - **DB2 Host name**: Enter the host name of the machine hosting DB2. The host name must be fully qualified.
 - Port: Enter the port that is being used by the DB2 instance. The default is 50005.
 - Database Name: Enter the name of the database to use with Maximo. The default database name is maxdb71. The database will be created if it does not already exist.
 - Instance: Enter the name of the database instance to be used with Maximo.

After you have entered the configuration information for the database that was selected, the Tivoli Asset Management for IT installation program will connect to the database server to validate the information that you have entered.

| 🐙 Database Configuration | | × |
|--------------------------------|---|---|
| | Maximo Databas | е |
| InstallAnywhere by Macrovision | Enter the Maximo database information. Enter a fully qualified host name or IP address. The database user ID will be created if it does not exist. Host name fenway.itsc.austin.ibm.com Port 50005 Database name maxdb71 Instance ctginst1 Database user ID maximo Database password ****** Confirm password ***** | |
| <u>C</u> ancel | Previous <u>N</u> ext | |

Figure 3-42 DB2 database configuration

12. From the Automate Database Configuration panel (Figure 3-43), select automate database configuration if you want to have the database automatically created, and then click **Next**.

If you choose not to automate the database creation, this step assumes you have already created a database instance, a database, table spaces, a user, and a schema for use with Tivoli Asset Management for IT.

Refer to install/c_ccmdb_manconfigdb.dita#c_ccmdb_manconfigdb. If you have not manually configured the database prior to selecting **Do not automate database configuration**, the installation will verify that you have not completed these pre-install tasks, and you will be reminded to complete them prior to restarting the installation program.

| 🛃 Database Configuration | | |
|--------------------------|--|---|
| | | Automate Database Configuration |
| | Select whether you would like to automate would like to automate the configuration st protocol enabled. For UNIX machines, ssh machines, the ssh or SMB protocol is requ Automate database configuration Do not automate database configuration | the configuration of the Maximo database. If you teps, the machine must have a remote access or rsh rexec protocol is required. For Windows lired. |
| <u>C</u> ancel | | Previous Next |

Figure 3-43 Automate database configuration prompt
- 13. From the Remote Access Authorization panel (Figure 3-44), enter the authorization information for the WebSphere configuration, and then click **Next**:
 - Operating system user ID: Enter a valid user ID that will allow the Tivoli Asset Management for IT installation program to access the system that is hosting WebSphere. This user ID must have administrative rights on the machine that you are accessing.
 - Operating system password: Enter the password for the system user ID.

| 📲 Database Configuration | |
|--|--|
| | Remote Access Authorization |
| | You have selected Maximo database automatic configuration. Enter a user ID that can access the Maximo database server. This user ID must be able to access the server using the remote access protocol enabled on that machine. User ID Administrator Password ******* Confirm password ******* |
| InstallAnywhere by Macrovision – Cancel | Previous Next |

Figure 3-44 Remote access authorization

14. In the DB2 Administration panel (Figure 3-45), enter the DB2 database information. Enter the Windows service user ID and password only if the DB2 server is installed on a Windows machine, click **Next**.

| 📲 Database Configuration | |
|--------------------------|---|
| | DB2 Administration |
| | Enter the DB2 database information. Enter the Windows service user ID and password only if the DB2 Server is installed on a Windows machine, otherwise leave those fields blank. Installation directory C:\Program Files\IBM\SQLLIB Instance administrator user ID db2admin Instance administrator password ***** Confirm password ***** Windows service user ID db2admin Windows service password ***** Confirm password ***** |
| <u>Cancel</u> | Previous Next |

Figure 3-45 DB2 administration configuration

15. From the Database Configuration DB2 Tablespace panel (Figure 3-46), specify the tablespace configuration properties if the DB2 database will be automatically created. Click **Next**.

| 🖳 Database Configuration | |
|--------------------------|---|
| | DB2 Tablespace |
| | Enter the DB2 tablespace information. Tablespace sizes are measured in megabytes. Data tablespace name MAXDATA Data tablespace size Medium (5000 Mb) Temporary tablespace name MAXTEMP Temporary tablespace size (Mb) 1000 Index tablespace name MAXDATA Index tablespace size Small (3000 Mb) |
| <u>Cancel</u> | Previous Next |

Figure 3-46 DB2 Tablespace configuration

16. From the Maximo Application Server Type panel (Figure 3-47), select the relevant application server on which you want to deploy your Maximo application. In this case, choose **IBM WebSphere Application Server**, and click **Next**.



Figure 3-47 Maximo Application Server Type

- 17.From the WebSphere Connectivity panel (Figure 3-48), enter the host information about the WebSphere server, and then click **Next**:
 - **Host name**: Enter the fully qualified host name of the system hosting WebSphere. Alternatively, you can provide the IP address for the system.
 - SOAP port: Enter the SOAP port of the WebSphere system. The default value for this field is 8879.

| 📲 apps v r | |
|--|--|
| | WebSphere Connectivity |
| | Enter the information that is used to connect to the WebSphere Application Server Deployment Manager. Enter a fully qualified host name or IP address Host name fenway.itsc.austin.ibm.com SOAP port 8879 |
| InstallAnywhere by Macrovision — <u>C</u> ancel | Previous Next |

Figure 3-48 WebSphere Connectivity configuration

- 18. From the Remote Access Authorization panel (Figure 3-49), enter the authorization information for the WebSphere configuration, and then click **Next**:
 - Operating system user ID: Enter a valid user ID that will allow the Tivoli Asset Management for IT installation program to access the system that is hosting WebSphere. This user ID must have administrative rights on the machine that you are accessing.
 - Operating system password: Enter the password for the system user ID.

| 🖳 appsvr | |
|--|--|
| | WebSphere Remote Access Authorization |
| | You have selected to automate the WebSphere configuration. Enter a user ID that can access the WebSphere Deployment Manager. This user ID must be able to access the server using the remote access protocol enabled on that machine. The machine must have a remote access protocol enabled. For UNIX machines, the ssh or rsh rexec protocol is required. For Windows machines, the ssh or SMB protocol is required. Operating system user ID Administrator Operating system password ******* Confirm password ******* |
| InstallAnywhere by Macrovision <u>C</u> ancel | Previous <u>N</u> ext |

Figure 3-49 WebSphere Remote Access Authorization

19. From the Automate WebSphere Configuration panel (Figure 3-50), select **Automate WebSphere configuration**, and then click **Next**.

If you choose not to have the Tivoli Asset Management for IT installation program automatically configure the middleware, you will have had to configure WebSphere manually prior to the installation of Tivoli Asset Management for IT. Configuration tasks include creating a profile, running WebSphere as a Windows service, copying the WebSphere keystore file from the machine where WebSphere is installed to the administrative workstation, setting up Java Message Service (JMS) queues, and so on.

| 🖳 appsvr | |
|--------------------------------|---|
| | Automate WebSphere Configuration |
| | Select whether you would like to automate the WebSphere configuration. The machine must have a remote access protocol enabled. For UNIX machines, the ssh or rsh rexec protocol is required. For Windows machines, the ssh or SMB protocol is required. |
| InstallAnywhere by Macrovision | Previous Next |

Figure 3-50 Automate WebSphere Configuration prompt

- 20.From the WebSphere Deployment Manager Configuration panel (Figure 3-51), enter values for the following fields, and then click **Next:**
 - WebSphere installation directory: Enter the directory where WebSphere is installed on the host system.

On Windows, this value might be C:\Program Files\IBM\WebSphere\AppServer Linux.

- User ID: Enter the administrative user ID used. The default user ID for all platforms is wasadmin.
- Password: Enter and confirm the password for the administrative WebSphere server.
- Profile name: Enter the name of the WebSphere profile. The default for all platforms is ctgDmgr01.

| 🐙 appsvr | |
|----------------------------------|--|
| | WebSphere Deployment Manager Configuration |
| | Enter the information for the WebSphere Application Server. WebSphere installation directory C:\Program Files\IBM\WebSphere\AppServer User ID wasadmin Password ****** Confirm password ****** Profile name ctgDmgr01 |
| InstallAnywhere by Macrovision – | Previous Next |

Figure 3-51 WebSphere Deployment Manager Configuration

- 21. From the WebSphere Application Server Configuration panel (Figure 3-52), enter the following information, and then click **Next**:
 - Web server port: Enter the Web server port used by WebSphere. The default for all platforms is 80.
 - Web server name: Enter the name of the Web server. The default for all platforms is webserver1.
 - Node name: Enter the name of the WebSphere node containing the application server. The default for all platforms is ctgNode01.
 - Cluster name: Enter the name of the WebSphere cluster containing the application server. The default for all platforms is MAXIMOCLUSTER. The cluster name is optional. The cluster and application server will be created if they do not exist.

| 2 appsvr | |
|---|--|
| | WebSphere Application Server Configuratio |
| | Enter the information for the WebSphere Application Server. Web server port 80 Web server name webserver1 Node name ctgNode01 Cluster name MAXIMOCLUSTER |
| nstallApywhere.by Macrovision Cancel | Previous Next |

Figure 3-52 WebSphere Application Server Configuration

22.From the Security panel (Figure 3-53), indicate whether application server security needs to be enabled automatically, and then click **Next**.

| 🐙 appsvr | | - 🗆 🗵 |
|--|---|---------|
| | S | ecurity |
| | Select whether J2EE application security should be configured automatically. This requires J2EE security to already be enabled in your application server. Configure J2EE application security | 5 |
| | | |
| InstallAnywhere by Macrovision — Cancel | Previous | Next |

Figure 3-53 Application Server Security configuration

- 23. From the Integration Adapter JMS Configuration panel (Figure 3-54), enter the following information, and then click **Next**:
 - JMS DataSource name: A JMS server requires a DB2 data repository to be configured to maintain messages. Enter the name of the database to be used by JMS. The default is intjmsds.
 - Select whether the JMS datastore needs to be persisted. Select the appropriate option.

A database will not be used to persist messages. If you later decide that you want to persist JMS messages, you will have to configure the JMS implementation manually. The next several steps of this Tivoli Asset Management for IT installation procedure assume that you are allowing the Tivoli Asset Management for IT installation program to configure the JMS implementation to persist messages.

| 🐙 appsvr | | | |
|--------------------------------|---|-----------------------------------|---------|
| | | Integration Adapter JMS Configu | uration |
| | Enter the information for the Java Message JMS DataSource name intimsds Select whether the JMS datastore should to C Persist JMS messages C Do not persist JMS messages | e Service (JMS). ne persisted. | |
| InstallAnywhere by Macrovision | | Previous | Vext |

Figure 3-54 Integration Adapter JMS Configuration

24. In the Run Configuration Step panel (Figure 3-55), you can select to run the configuration step now or not. If you select **Run the configuration step now**, Tivoli Asset Management for IT will be configured when you select this option and press **Next**.



Figure 3-55 Run Configuration Step

25. From the Choose Shortcut Folder panel (Figure 3-56), select the type of shortcut that you want to arrange for Tivoli Asset Management for IT, and then click **Next**.

| 📲 IBM Tivoli base services | | | |
|----------------------------------|---------------------------------|--------------------------|-------------------------------|
| | | | Choose Shortcut Folder |
| | Where would you like to creat | e product icons? | |
| | In a new Program Group: | IBM Tivoli base services | |
| | C In an existing Program Group: | Accessories | * |
| | O In the <u>S</u> tart Menu | | |
| | C On the Desktop | | |
| | C In the Quick Launch Bar | | |
| | C Other | | Choose |
| | C Don't create icons | | |
| | C Dong cleatercons | | |
| | | | |
| | | | |
| | Create Joons for All Users | | |
| | | | |
| InstallAnywhere by Macrovision - | | | |
| <u>Cancel</u> | | | <u>P</u> revious <u>N</u> ext |

Figure 3-56 Choose a Shortcut Folder option

26. From the Input Summary panel (Figure 3-57), review the information that you have provided to the Tivoli Asset Management for IT installation program, and then click **Next**. Click **Previous** to return to previous panels to make any changes.



Figure 3-57 Installation configuration Input Summary

27. From the Pre-Installation Summary panel (Figure 3-58), review the installation information presented, and then click **Install**. The installation task will begin. You can monitor progress by viewing messages displayed above the progress bar. Click **Next**. The installation will take place. Again, in our environment, this installation took between one and two hours to complete.



Figure 3-58 Pre-Installation Summary

28. You will be prompted to set a new base language or add additional language packs (Figure 3-59). Click **No.**

| 堰 IBM Tivoli base services | | |
|----------------------------------|--|---------------------|
| | | Language Support |
| | Do you want to set a new base language or add additional languages? ○ Yes ⓒ No | |
| InstallAnywhere by Macrovision – | | revious <u>Next</u> |

Figure 3-59 Language Support option

29. From the Install Complete panel, click **Done**. After the Tivoli Asset Management for IT installation program has completed the installation and configuration tasks, it will exit. The logs are at *<TAM4IT_Home*/logs.

3.6 **Tivoli Asset Management for IT solution package** installation

After the middleware and Tivoli's process automation engine (base services) are installed, you must now add and enable the modules that are specific for Tivoli Asset Management for IT.

3.6.1 Process solution package overview

In Maximo terminology, a *process solution package* is a self-contained ZIP file of installation artifacts and deployment logic that can be deployed using the process solution installation program. *Installation artifacts* are the files and

content that are installed into your Maximo environment to enable the services management functionality of a Process Manager Product or Integration Module. The Tivoli Asset Management for IT specific functions are added to our environment through this process. The deployment logic is the actions that are carried out in order to deploy the process solution into the Maximo environment. Typically, these actions include building and deploying J2EE applications, running database scripts that load the process solution content into the Maximo database, and adding users and groups for security.

3.6.2 Pre-process solution package installation checklist

When you perform a deployment operation using the process solution installation program, you are running actions that modify the configuration and content of your J2EE, database, and directory middleware servers. Review the following steps before invoking the process solution installation program:

Have middleware login information available.

The process solution installation program requires access to middleware servers in order to automate the deployment of the process solution package. You will need to know the administrative user IDs and passwords for the impacted middleware servers. The actual middleware servers whose login information is required will depend on the process solution package being installed. The process solution installation program ensures that any required login information is specified before continuing with the deployment operation.

Back up middleware servers and administrative workstation.

You must create backups for impacted J2EE, database, and directory servers before you deploy a process solution package using the process solution installation program.

Ensure middleware servers are started.

Start any impacted middleware servers before running the process solution. Starting these servers saves a lot of time if, for whatever reason, you want to revert to the environment where you have both the middleware and Tivoli's process automation engine installed, without having to start from the beginning.

3.6.3 Enable Tivoli Asset Management for IT License

You will be entitled to use Maximo according to the license that you have purchased only after you install the process solution package. Complete the following steps to install the Tivoli Asset Management for IT process solution package into the Maximo environment using the process solution installation wizard:

 From the launchpad (Figure 3-60), launch the process solution installation program by clicking the link under option 4, Enable IBM Tivoli Asset Management for IT License for usage. The Install Anywhere Installer executes on the Maximo administrative workstation. The launch script is deployed and configured by the installation program. No post-installation configuration is required.



Figure 3-60 Enable Tivoli Asset Management for IT License for usage

The process solution installation program performs a series of validation checks to verify that the package that you selected is valid. The system is checked to insure that the package has not already been deployed.

2. Select a language for your installation (Figure 3-61), and then click **OK**.



Figure 3-61 Language selection

3. The process solution installation program license agreement window (Figure 3-62 on page 108) is displayed. Read the license information and select I accept the terms in the license agreement if you agree with the terms. Click Next.



4. Allow the package to be validated (Figure 3-62), and then click Next.

Figure 3-62 Package Validation Results

5. From the Middleware Login Information panel (Figure 3-63), enter the credentials for which you are being prompted, and then click **Next**. After you have entered the requested user IDs and passwords, the process solution installation wizard will validate the credentials by connecting to the middleware servers using the supplied credentials. Enter information for both the WebSphere Admin and WebSphere Remote access as shown in Figure 3-63.

| 📲 IBM Tivoli Asset Management f | or IT 7.1 Package | | |
|---|---|--|---|
| | Mic | Idleware Login Information | |
| ::::::::::::::::::::::::::::::::::::::: | The package you selected deploy middleware. Specify the required press "Next" when you have com | /s artifacts to the following user IDs and passwords and pleted. | |
| | WebSphere Admin WebSphere R WebSphere Application | emote) Server - Administration | |
| | Userid: wasadmin Password: ****** | | |
| InstallAnywhere by Macrovision | IBM Tivoli Asset Management fo | r IT 7.1 Package Middle | ware Login Information |
| | | The package you selected deploys ar middleware. Specify the required use press "Next" when you have complete | lifacts to the following r IDs and passwords and d. |
| | | WebSphere Admin WebSphere Remot | e ver - Remote Access |
| | | Userid: Administrator Password: ****** | |
| | InstallAnywhere by Macrovision — <u>C</u> ancel | | Previous <u>N</u> ext |

Figure 3-63 Middleware Login information: WebSphere administrator and remote access

6. After the credentials have been verified, a Package Options panel (Figure 3-64) is displayed that details the deployment options that the package supports. Leave the Package Options box unchecked to have the enterprise archive (EAR) files deployed and Maximo database updated by the process solution installation program, and click **Next**.



Figure 3-64 Package Options

7. From the Pre-Install Summary panel (Figure 3-65), review and verify the information that is displayed, and then click **Next**.



Figure 3-65 Pre-Install Summary

8. At this point, the process solution installation program begins the package installation process. The Deployment Progress panel (Figure 3-66) will inform you of the deployment progress of the installation.



Figure 3-66 Deployment Progress

9. When the installation has completed successfully, from the Package Successfully Deployed panel (Figure 3-67), click **Done** to exit the process solution installation wizard. If there is a package failure, a message will be displayed for the step that failed. You might see an installation progress bar displayed briefly after you click **Done**. The Process Solution Installation Wizard is actually terminating, and no installation activities are being performed. The deployment of the Process Solution Package that you were installing has already completed, and the progress bar can be safely ignored.

| 🐙 IBM Tivoli Asset Management fo | or IT 7.1 Package |
|----------------------------------|---|
| | Package Successfully Deployed |
| | The package was successfully deployed. All deployment actions associated with the package completed successfully. Note that the package might have additional post-install configuration steps that need to be completed before the package can be used. Consult the documentation for the package that you just installed for more information. |
| | Deployment Messages |
| | CTGIN0148I: Deployment Engine is performing a system check to ensure all requirements are satisfied. |
| | CTGIN0147I: Deployment Engine is resolving the topology. |
| | CTGIN0146I: Completed system check for check "1" of "1". Check display name: "Check Common PMP Installed". |
| | CTGIN0147I: Deployment Engine is resolving the topology. |
| | CTGIN01461: Completed system check for check "1" of "1" Check |
| InstallAnywhere by Macrovision - | |
| <u>C</u> ancel | Previous Done |

Figure 3-67 Package Successfully Deployed

3.7 Post Installation Tasks

This section describes some of the key tasks that must be performed after the installation process to ensure that Tivoli Asset Management for IT is correctly installed and ready to use.

Prior to assuming the configuration stage of your Tivoli Asset Management for IT deployment, take the time to ensure that the basic installation is functional. To confirm that the basic installation is functional, read through the following section, verify, and adjust where necessary.

Tip: We recommend that you document all of your findings carefully and meticulously. After all your tasks have been completed, store all of your installation instructions and properties for future reference.

3.7.1 Ensure Tivoli Asset Management for IT services are running

Before you can attempt to access Tivoli Asset Management for IT, ensure that all of the *relevant* services are started.

Tip: From Windows Services, set the following services to automatic startup:

- DB2 COPY Services
- ► DB2 Governor
- DB2 License Server
- DB2 Management Service
- DB2 Remote Command Server
- ► DB2 Security Server
- ► DB2DAS
- ► IBM HTTP Server 6.1
- IBM HTTP Administration 6.1
- IBM WebSphere Application Server V6.1
- IBM WebSphere Application Server V6.1 Node Agent
- IBM Rational Agent Controller
- IBM Tivoli Directory Admin Daemon V6.1
- IBM Tivoli Directory Server Instance V6.1

If you are running a demonstration Maximo Asset Manager installation on limited resources, for example, a personal computer or a virtual machine, we recommend that you do not run these specified services as automatic but rather keep these services configured to start manually.

3.7.2 Ensure Tivoli Asset Management for IT application connectivity

After you have successfully started the requested services, ensure connectivity by logging into Tivoli Asset Management for IT. Use your Web browser to access the application. By default, the URL is:

http://<your full qualified host name>:9080/maximo

After you have accessed the Tivoli Asset Management for IT through your browser, attempt to log in to the system administrator user ID and password. (By default, the system administrator user ID and password is maxadmin and maxadmin.)

3.7.3 Verify the language pack installation

To determine if the language pack installation is successful, ensure that you can log in with the various languages listed on your Maximo login page.

3.7.4 Report administration

Prior to the configuration phase of the deployment, perform the following report administration checks.

Generate request pages

Check to make sure that you can generate request pages:

- 1. Log in to Tivoli Asset Management for IT.
- From the Start Center (Figure 3-68 on page 116), navigate to the Go To function and select Administration → Report Administration.

| 🚰 Start Center - Microsoft Internet Explorer | | | X |
|--|--|--|--|
| Eile Edit View Favorites Iools Help | | | AT |
| 🕒 Back 🔹 🕥 🖌 📓 🐔 🔎 Search 🤺 Favorites | 😌 🍰 🦉 - 📙 🎇 | | |
| Address 🕘 http://boston:9080/maximo/ui/?event=loadapp&value=startcr | itr8uisessionid=1208799656859 | | 💌 🔁 Go 🛛 Links や 📆 🕶 |
| Welcome, MAXADMIN | P <u>B</u> uletins: (0) | na 🖉 🖉 🖉 🖉 🖉 🖉 🖉 🖉 🖉 | Center 🔺 Profile 🗶 Sign Out ? Help 🏾 📺 🛌 🔺 |
| | | 🔄 Administration 🕴 | Sets |
| | 🕼 Change Content/Layout 🕮 Display Settings 🤻 | Assets | Organizations te Start Center |
| Quick Insert 🥒 🗇 🗊 | Inbox / Assignments | Change | Calendars |
| You do not have access to the selected actions. | Description APP DESCRIPTION | Contracts Financial | Resources Refresh Bulletin Board ite |
| Favorite Applications 🥒 🗅 🕯 | No Assignments found for MA | KPI Manager | Communication Templates Reporting |
| You do not have access to the selected actions. | 0 Bulletin Board 🔍 Filter 👍 🗯 | Report Administration | Conditional Expression Manager |
| ITUP Documents 🔍 Filter > 🚜 🖾 🖗 🥒 🖉 🗊 | Subject Message | Preventive Maintenance | Classifications Viewed? |
| This portlet has not been set up. To set up, select the edit icon $^{\!$ | There are currently no bulleting | Purchasing Release Security | Work View |
| | My Active Process Requests 🔻 Filter > 🏟 😂 🏟 | 🛒 Self Service | 200 |
| | This portlet has not been set up. To set up, select the edit icon ${}^{\mathbb{Z}}$ in the portlet header. | Service Desk System Configuration Task Measurement | |
| | My Process Requests Rejected in the past 7 days 🛛 🔻 Filter > 🖓 📄 🧀 | VVork Orders | 288 |
| | This portlet has not been set up. To set up, select the edit icon ${}^{\ell}$ in the portlet header. | | |
| | My Draft Process Requests 🛛 🐨 Filter > 🖓 🗍 😂 🖗 | | / 8 0 |
| | This portiet has not been set up. To set up, select the edit icon ${}^{\prime}$ in the portiet header. | | |

Figure 3-68 Report administration

3. In the center and to the right of the page (Figure 3-69), click **Generate Request Pages.**

| 🗿 Report Administration - Microsoft Inter | net Explorer | | | | | | | _8> |
|---|---------------------------|--|--------------------|----------------|-------------------------------|------------------|--------------|--------------------------------------|
| <u>File Edit View Favorites Tools H</u> elp | | | | | | | | |
| 🕒 Back + 🕥 - 🖹 💈 🚮 🔎 | Search 🐈 Favorites | 🔗 😒 🗟 · 😓 🛍 | 1 | | | | | |
| Address 💩 http://boston:9080/maximo/ui/?eve | ent=loadapp&value=report& | uisessionid=1208799656859 | | | | | 💌 🔁 Go |)Links » 📆 🔹 |
| Report Administration | | Allen Anna | | Belletins: (0) | na 🖉 🥵 To 🛛 🗠 Reports 👘 Start | Center 🌲 Profile | × Sign Out ? | ' ^{Helo} IBM. ¹⁴ |
| Find: | Belect Action | 💌 🎦 😡 🖉 🄶 🔅 | | | | | | |
| List Report Security | | | | | | | | |
| 🕅 Advanced Search 🔻 📓 Save Query 👻 | Pookmarks | | | | | | | |
| Reports 🛛 🛩 Filter > 🖓 🗐 😭 🛊 🐳 🛶 | | | | | | | G# () | tervnicadi ? 🗔 |
| Report File Name | Description | | Application | Report Folder | B | eport Type | Created By | Priority |
| | | | | | | | | |
| Select Records | | System Message BMXAA5308: The reque | it pages have beet | n generated. | | 1 | Oenerate Req | quest Pages |
| | | | | Close | | | | |

Figure 3-69 Generate Request Pages success

At this stage, the request pages have been successfully generated.

Generating a report output

As a best practice, assess to see whether the reports generate successfully. During the Tivoli Asset Management for IT installation process, the default report tool, BIRT, is automatically configured to run reports from within Tivoli Asset Management for IT. In the next few steps, we test and run a standard report:

- 1. Log in to Maximo.
- 2. From the Start Center (Figure 3-70), navigate to the **Go To** function and select **Administration** \rightarrow **Resources** \rightarrow **Labor**.

| Start Center - Microsoft Internet Explorer | | | |
|--|--|--|-----------------------------|
| Eile Edit View Favorites Tools Help | | | |
| 🕞 Back 🔹 🕥 🖌 🖹 🛃 🏠 🔎 Search 🤺 Favorites | 🔗 🔗 - 😓 🔟 - 🛄 🎎 | | |
| Address 🕘 http://boston:9080/maximo/ui/?event=loadapp&value=startcr | tr&uisessionid=1208799656859 | | |
| Welcome, MAXADMIN | P <u>B</u> ulletins: (0) | 🔗 💁 To 🛛 🛄 Reports 👘 Start | Center & Profile X Sign Out |
| | | a Administration | Sets |
| | 🛱 Change Content/Layout 📲 Display Settings | 🗧 🐸 Assets 🕴 | Organizations |
| Quick Insert 🥒 🗇 🗊 | Inbox / Assignments | Change | Calendars |
| You do not have access to the selected actions. | Description APP DESCRIPTION | Fini Qualifications | Bulletin Board |
| Favorite Applications 🥒 🗋 🛍 | No Assignments found for M | A S Intel Person Groups | Reporting |
| You do not have access to the selected actions. | 9 Bulletin Board , ⊽Filter . ∰, | inv Crafts ≅ Planninα | Conditional Expression Mana |
| ITUP Documents 🔻 Filter > 🚓 🕽 🏟 🥒 🥒 🖉 🗇 | Subject Message | Preventive Maintenance | CI Types |
| This portlet has not been set up. To set up, select the edit icon 🖌 in the portlet header. | There are currently no bullet | in Purchasing Purchasi | Work View |
| | My Active Process Requests 🛛 🔻 Filter > 🖓 🗄 🗊 🖗 | 🛒 Self Service 🔹 🕨 | |
| | This portlet has not been set up. To set up, select the edit icon $^{\ell}$ in the portlet header. | Service Desk System Configuration | |
| | My Process Requests Rejected in the past 7 days 🛛 🔻 Filter > 🏟 🖾 🖗 | Ask Management | |
| | This portlet has not been set up. To set up, select the edit icon \mathscr{I} in the portlet header. | - YHIN OFMOID | |
| | My Draft Process Requests 🛛 🔻 Filter > 🏟 📋 🖗 | | |

Figure 3-70 Labor panel

3. From the Labor application (Figure 3-71), click Select Actions \rightarrow Run Reports.

| Labor - M | licrosoft Internet Explorer | | | | | | | | - 18 × |
|------------------|-------------------------------|--|---|---|--|---|--------------------------------|---------------------------------|--------|
| <u>File Edit</u> | View Favorites Tools H | jelp | | | | | | | 1 |
| G Back 🔸 | 🕤 🖻 본 - 🕑 | 🔎 Search | rites 🧭 🔗 🎍 | · 🐨 - 🔜 🎎 | | | | | |
| Address | http://boston:9080/maximo/ui/ | ?event=loadapp&value=la | bor&uisessionid=12087996 | 56859 | | | | 💌 🔁 Go 🛛 Links 🏾 🕯 | 🔁 🔹 |
| Jaha Labo |)r | | | | 9 Bull | etins: (0) 🎤 <u>G</u> o To 🛛 💷 <u>R</u> e | oorts 👘 Start <u>C</u> enter 🔒 | A Profile × Sign Out ? Help 플로클 | M. 🖂 |
| List M Advanc | Find: | Setect Action Setect Action Chang Chang Attack Attack Pook Run R | e Status nment Library/Folders I eports | <u>ା ଥା ବ କାଷ୍ଟାର</u> ୍ଭ | | | | | |
| Labor 💘 | Filter > 🖍 🔅 🔅 🔶 🔶 | | | | | | | Cir Download 2 | 8 |
| Labor | First Name | Last Name | Default Craft | Default Skill Level | Calendar | Work Location | Work Site | Organization | |
| | | | F | To find records, use the filter file for more search options, use the To enter a new record, se | elds above and then press Ei Advanced Search button ab lect the Insert icon in the too | nter. ove. Ibar. | | | |
| Select R | lecords | | | | | | | | |

Figure 3-71 Run Reports

4. From the Reports dialog (Figure 3-72), select the Labor List report.

| | | | | | | _ 8 × |
|--|--|---|---|---|---|---|
| | | | | | | R |
| Favorites 🧭 🔗 💺 | 🔟 - 🛄 🎎 | | | | | |
| alue=labor&uisessionid=12088065 | 45812 | | | | 💌 🔁 G | o Links » 📆 🔹 |
| | | U Bulk | etins: (0) 🎓 <u>G</u> o To 🛛 Lat <u>R</u> ep | oorts 🕈 Start <u>C</u> enter 🌲 | Profile × Sign Out 1 | Help IEM. A |
| t Action 🛛 🖌 🏹 📄 | 21001010 | , | | | | |
| | | | | | | |
| 5 | | | | | | |
| | | | | | G#) | Lowenicand ? 🗇 |
| Default Craft | Default Skill Level | Calendar | Work Location | Work Site | Organization | |
| 1 | 9 | P | 1 | P | =ENGLENA | 1 |
| Con Deen Reports 1 Description Labor List Labor List | Ind Reports Scheduling | Status | ntead 7 | | | |
| | Favorites Favorites | Favorites Favorites Favorites Favorites Favorites Favorite Favori | Pavorites Pavorites Pavorites Pavorites Pavorites Pavorites Pavorite Pavor | Favorites Favorites Favorites Favorites Favorites Favorites Favorites Favorites Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Favorite Fa | Pavorites Pavorites Pavorites Pavorites Pavorites Pavorites Pavorite Pavor | Proventes Prove |

Figure 3-72 Labor List

5. From the Request Page (Figure 3-73), specify your parameter properties. In our example, we specify the value ACTIVE as the status parameter value.

| Labor - Microsoft Internet Explorer | | | | | | | | _ 8 × |
|--|--|--|--|--|----------------------|---------------------|--------------------------|----------------|
| <u>File Edit View Favorites Tools H</u> elp | | | | | | | | |
| 🕒 Back 🔹 💮 🖌 🗾 😰 🐔 🔎 Search 🤸 Fav | vorites 🔗 😒 🌭 | 🐨 - 🔜 🎎 | | | | | | |
| Address 🐻 http://boston:9080/maximo/ui/?event=loadapp&value= | abor&uisessionid=1208799656 | 6859 | | | | | | 50 Links » 🐔 🔹 |
| Labor | | | 9 | Bulletins: (0) 🍖 Go To 💷 Repo | rts 📌 Start <u>C</u> | enter 🌲 <u>P</u> ro | ofile 🔰 <u>S</u> ign Out | ? Help IRM, 🖂 |
| (a | | | entre antes antes antes also | | | | | |
| Find: A Select Act | tion 📉 🎦 🗔 | 2 🔶 🔶 🛟 K | 0 | | | | | |
| List Labor Crafts Qualifications | | | | | | | | |
| 🕅 Advanced Search 🔻 🎴 Save Query 👻 🔗 Bookmarks | | | | | | | | |
| Labor 🛛 😾 Filter > 🚜 👔 🖙 🔶 🗇 🔶 | | | | | | | G. | Download 1?10 |
| Labor First Name Last Name | Default Craft | Default Skill Level | Calendar | Work Location | Work Site | | Organization | |
| | Request Page | | | □ ? ⊠ | | P | =ENGLENA | 1 |
| | Parameters a cheduced rep scheduced rep Ste Cheduced C Immediate C Atthis Time C Recurring Email To Subject Comments | te displayed, the report will a dimensional and the schedule and Email Score and Score | execute against the current ex | Jelečet record set. run times and receive the | | | | |

Figure 3-73 Request Page

6. Click Submit. Figure 3-74 is displayed.

| and Report Hewel - Flich | source and composite | | | | |
|---|----------------------------|----------------------|---------------------|-----------------|------------------|
| e <u>E</u> dit <u>V</u> iew F <u>a</u> vorites | <u>T</u> ools <u>H</u> elp | | | | 4 |
|) Back 🔹 🕥 👻 🚺 | 💈 🏠 🔎 Search | Savorites 🎸 | 3) 🔗 - 😓 🖂 | 🗔 🛍 | |
| ress 🙋 http://boston:9080, | /maximo/report?report | =labor.rptdesign&app | name=LABOR&requesti | d=1208801947515 | 🔁 Go 🛛 Links 🌺 🖣 |
| Reporting | | | | | IBM. |
| | | | 1 an 17 | | |
| Page 1 of 1 | | | - 🗪 | | |
| Page 1 of 1 | | | <u></u> | | |
| Page 1 of 1 | | | <u></u> | | |
| Page 1 of 1 Tivoli _abor List | | | | | |
| Page <u>1</u> of 1 Tivoli _abor List _status: ACTIVE | | | | | |

Figure 3-74 Report output

7. Upon receiving a report output as specified, you can determine that the report has generated successfully.

4

Tivoli License Compliance Manager

Organizations use IBM Tivoli License Compliance Manager (TLCM) V2.3 to manage their software assets.

TLCM is a distributed software license management system that performs:

- Software inventory use and analysis
- License entitlement management
- Web-based administration
- XML import and export
- Reporting

This chapter provides an overview of planning, installing, and configuring Tivoli License Compliance Manager. Specific emphasis is given to tasks that are required to integrate into IBM Tivoli Asset Management for IT.

4.1 IBM Tivoli License Compliance Manager

IBM Tivoli License Manager is a Web-based solution that meets the challenge to support the complex software asset management process. It provides software usage metering, procurement management, and license allocation and compliance support services on different host platforms.

IBM Tivoli License Manager can help enterprises meet their software asset management objectives by accomplishing, often silently, a certain number of tasks, described as follows:

- Collecting information about installed products using the Tivoli Common Inventory Technology for scans
- Identifying the start and the stop of software products on any machines on which the IBM Tivoli License Compliance Manager Agent is running
- Providing reports that allow the comparison of installed, used, and procured licenses in support of procurement management activities
- Metering software usage for products that have no license definitions or requirements
- Generating alerts that automatically inform administrators when license usage reaches a predefined level
- Assigning a pool of licenses to users and machines with a product-specific, context-driven quantity that defines the number of software instances, either installed or in execution, for a specific product for compliance checks
- Associating the information contained in contracts to licenses' definitions
- Providing the security of confidential information that is ensured by the maintenance of security profiles for access to the Administrative user interface, and by the encryption of data during transmission between the various components of the IBM Tivoli License Compliance Manager solution, as well as when accessing the Administrative user interface
- Establishing electronic entitlement for IBM Software using prefilled, error-free IBM license terms and definitions
- Providing the capability to manage a complex product made up of several components or bundles, possibly installed on different systems
- Offering logical partitioning and virtualization support using layer abstraction based on partitioning and virtualization elements in the virtualization stack

4.2 Planning

The planning phase of any project is arguably the most critical for a successful implementation, because proper planning identifies the scope, risk, and costs for your project. As part of your planning estimates, understanding the major application component purposes and interactions will improve the accuracy of your planning and contribute to the success of your project.

This section describe the major components and design influences that need to be part of your IBM Tivoli License Compliance Manager implementation planning.

Give consideration to:

- Physical placement of the Administration server
- Physical placement of the Runtime server
- Coverage of supported platforms
- ► Firewalls and network
- Sizing of components for disk space, memory, and CPU

4.3 Three-tier architecture

IBM Tivoli License Manager V2.3 leverages a three-tiered architecture:

Tier 1

A client named IBM Tivoli License Compliance Manager Agent that runs on the user server or desktop

Tier 2

A resource manager providing real-time services for the client named ITLM Runtime server

Tier 3

An application server that contains the business logic and stores the core data for the Administration Server

Figure 4-1 on page 124 depicts the IBM Tivoli License Manager three-tiered architecture.



Figure 4-1 TLCM three-tier architecture

All communications between IBM Tivoli License Compliance Manager components use the HTTP protocol. HTTP requests made by an TLCM Agent to the ITLCM Runtime server or by the TLCM Runtime server to the ITLCM Administration server are first received by an HTTP server that must be installed on each ITLM Administration server and TLCM Runtime server. The HTTP server must forward the HTTP request to the corresponding IBM Tivoli License Compliance Manager application running on IBM WebSphere Application Server. To allow this type of transfer, the IBM WebSphere Application Server application installs a plug-in for WebSphere on the HTTP server. This plug-in is able to transfer the HTTP request to the TLCM application running on the IBM WebSphere Application Server.

Both the ITLCM Administration server and the ITLCM Runtime server need to store and access data in a relationship database management system (RDBMS). To access this RDBMS, IBM WebSphere Application Server uses the Java DataBase Connectivity (JDBC[™]) technology. In Version 2.3, a type 4 (Pure Java) driver is used.

IBM Tivoli License Manager three-tiered architecture is comprised of physical and logical components. The following sections go into detail about these components and how they interact with each other.


Figure 4-2 shows the relationships and communication between the elements that will be detailed in the following sections.

Figure 4-2 Component relationships

4.3.1 Administration server

The Administration server is the central point of control for the TLCM environment. From the Administration server, you can:

- Produce reports
- Manage and assign licenses
- Define product properties
- Schedule software scans
- Manage the Administration server, Runtime server, and agents
- Control logical access
- Define custom fields

The ITLM Administration server is the core of the solution and works as the central arbiter within the License Management strategy by providing the following services:

- Stores and maintains the information about products and licenses in a central database, which is arranged by the organization and the divisions
- Gathers the software usage and installation data collected by the IBM Tivoli License Compliance Manager Agents and processed by the IBM Tivoli License Compliance Manager Runtime servers
- Provides an administrative user interface that can be used to perform all of the administrative tasks and to create historical reports of license usage and product installation over time
- Forwards e-mail notifications to the license administrator upon detection of a possible violation of software product usage

There are two interfaces to the Administration server: One is the Catalog Manager and the other is the Web interface.

Catalog Manager

The Catalog Manager holds definitions of software. In Version 2.3, the Catalog Manager contains two types of entries: One type of entry is a private catalog of IBM branded software, and the other type of entry is a public catalog.

Note: The public catalog can be modified by the Catalog Manager. The private catalog is limited to extending the product hierarchy and disabling IBM defined signatures.

The IBM Tivoli License Compliance Manager Master Catalog is a central repository of product information about all software components and related files for products that can be monitored by IBM Tivoli License Manager.

The IBM Tivoli License Compliance Manager Master Catalog resides in the ITLM Administration server database, and a subset of it is periodically downloaded to each IBM Tivoli License Compliance Manager Runtime server. This subset of the IBM Tivoli License Compliance Manager Master Catalog, called the *Runtime Catalog*, only includes those entries from the IBM Tivoli License Compliance Manager Master Catalog that relate to products that have been discovered running on nodes by IBM Tivoli License Compliance Manager Runtime server. A subset of the Runtime Catalog is also downloaded to each registered IBM Tivoli License Compliance Manager Runtime server.

IBM Tivoli License Manager includes an IBM Tivoli License Compliance Manager Catalog Manager tool that enables you to maintain the IBM Tivoli License Compliance Manager Master Catalog of products that need to be monitored. The IBM Tivoli License Compliance Manager Catalog Manager tool has a graphical interface that enables you to perform the following tasks:

- Update the IBM Tivoli License Compliance Manager Master Catalog with updates provided by IBM on a regular basis.
- Extend the IBM Tivoli License Compliance Manager Master Catalog with custom defined products using a hierarchical structure: product, version, release, component, and signature.
- Create additional entries to the IBM Tivoli License Compliance Manager Master Catalog from unknown signatures that have been detected by IBM Tivoli License Compliance Manager Agents for which no corresponding entry exists in the IBM Tivoli License Compliance Manager Master Catalog.
- Manage signatures by defining custom signatures and assigning or removing them to components.

Figure 4-3 provides an overview of the IBM Tivoli License Compliance Manager Catalog Manager tool in relation to other IBM Tivoli License Manager components.



Figure 4-3 IBM Tivoli License Compliance Manager Catalog Manager tool

Note: The IBM Tivoli License Compliance Manager Catalog Manager is not available in the *IBM Tivoli License Manager for IBM Software* package.

Catalog management tasks are of vital importance to the overall IBM Tivoli License Manager solution, because procurement management and software usage monitoring activities are dependent on software signatures defined and maintained as entries in the IBM Tivoli License Compliance Manager Master Catalog. In procurement management terms, licenses must be assigned to a software product signature in order to be valid and usable upon distribution. From a usage monitoring perspective, software usage monitoring is automatically enabled for software products with a valid license assigned to them, and manually defined by the IBM Tivoli License Manager administrator for software products that might not have a license defined to them. The IBM Tivoli License Manager administrator first selects a valid software signature for the product and then enables software usage monitoring based on the software signature of the software product.

Web interface

The Web interface is the primary point of user interaction with the TLCM application. The interface can be accessed by using the URL http://<admin server>/slmadmin/login where <admin server> is your license manager Administration server.

4.3.2 Runtime server

An IBM Tivoli License Manager environment must have at least one IBM Tivoli License Compliance Manager Runtime server, but more servers can be set up to scale the solution and cover large sites. They are managed directly by the IBM Tivoli License Compliance Manager Administration server and are mainly in charge of the IBM Tivoli License Compliance Manager Agents' registration process and sending IBM Tivoli License Compliance Manager Agents' software inventory information to the ITLM Administration server. In addition to that, each IBM Tivoli License Compliance Manager Runtime server provides the following functions:

- Receives the results of software scans from the IBM Tivoli License Compliance Manager Agents and processes them to build a view of the software that is installed on each machine
- Provides a Web interface that is used to deploy IBM Tivoli License Compliance Manager Agents to nodes
- Forwards software catalog information containing known software signatures to IBM Tivoli License Compliance Manager Agents
- Monitors the activity of the IBM Tivoli License Compliance Manager Agents, notifying the system administrator when an IBM Tivoli License Compliance Manager Agent has been stopped or removed from the system

Forwards e-mail notifications to the license administrator upon detection of events that have occurred on the IBM Tivoli License Compliance Manager Runtime server and its IBM Tivoli License Compliance Manager Agents

4.3.3 Agent

A small agent footprint must be deployed on all of the client machines that are to be monitored by IBM Tivoli License Manager. IBM Tivoli License Compliance Manager Agents provide the primary interface of License Management. IBM Tivoli License Compliance Manager Agents run a very small amount of software, less than 500 KB of executable software on all platforms.

Each IBM Tivoli License Compliance Manager Agent performs the following functions silently and without any user intervention:

- Executes a complete scan of the target machine, providing the IBM Tivoli License Compliance Manager Runtime server with the software and certain hardware information that is collected. Hardware inventory information is limited to those aspects of the hardware inventory that are necessary for effective software asset management.
- IBM Tivoli License Compliance Manager Agent communicates with only one IBM Tivoli License Compliance Manager Runtime server. If the communication is not possible, the IBM Tivoli License Compliance Manager Agent stores data and sends it as soon as possible. Agent storage limits are configurable in Version 2.3, with the default storing 15 MB of data on the agent.
- Identifies the starting or stopping of software products and sends software usage information to the IBM Tivoli License Compliance Manager Runtime server. In case the IBM Tivoli License Compliance Manager Runtime server is not available at that time or the machine on which the IBM Tivoli License Compliance Manager Agent is running is not connected to the network, IBM Tivoli License Compliance Compliance Manager Agents can cache software usage information and send it at a time when its ITLM Runtime server is available.
- Periodically checks for upgrades of the IBM Tivoli License Compliance Manager Agent code, so that user intervention on the target machine is not required.

Note: As of Version 2.3, the release notes list Windows 2000 Server as no longer supported as an agent platform. Support for agents running Windows 2000 Server was restored in Version 2.3 Fix Pack 2.

Installation Methods

You can install agents through multiple methods:

- ► Pull the deployment from a Web page
- Windows login scripts
- ► Automated remote bulk installation by IBM Tivoli Configuration Manager
- Manual install
- Bulk remote install via Remote Execution and Access (RXA)-based ssh/rexec or Windows protocols

Pull the deployment from a Web page

Each IBM Tivoli License Compliance Manager Runtime server is set up to deploy agents via a Web page running on the Runtime server. This Web page requires a Java applet to be executed to complete the installation. The name of the Runtime server and the name of the division are also required at installation time.

Windows login scripts

Windows servers and desktops can have their agents deployed via the Windows login process. This method requires the deployment of code to the Primary Domain Controller, as well as the user logging in to have Administrative rights. The TLCM installation media contains a sub directory webdoc\agent\pdc_deploy that contains the necessary scripts and installation code for the agent.

If you have multiple divisions, for example, one division for mobile computers and desktops and one for servers, we recommend one of two methods for allocating the division:

- Setting up an exclude list for servers on the personal digital cellular (PDC) and installing the server agents with a different method
- Using a server list on the PDC to determine the correct division for the installation

Automated remote bulk installation

IBM Tivoli Configuration Manager can take the agent software package blocks that ship with TLCM and automate the distribution and installation of the agent software.

Manual installations

TLCM has a wizard that deploys the agent software by executing the installation on the operating system where the agent will be run. This method might be suitable for environments where network connectivity does not permit network installations or where the implementation is on a small scale, such as a pilot.

Bulk remote installations

Bulk remote installations in Version 2.3, as shown in Figure 4-4 on page 131, are now handled with RXA tooling, which drives native remote access methods for installation.

| ompliance Manager, Version 2.3, us | ing RSH/SSH | | | | | | |
|------------------------------------|-------------|------|------------------|-------------|---|-------|----------------|
| Organization IBMITSO | | птас | Division _WIN | | Runtime address mulitsclaustin.ibm.com | | |
| | | | | | | | |
| | | | . 1 | | | | |
| Add Remove Add Remove Add Remove | | | | | | | |
| get com Organization Division | Runtime a | Port | SSL port | Agent insta | Security level | OS | userName |
| a.itsc.au IBMITSO ITSO_WIN | mu.itsc.au | 80 | 443 | c:/windows | MIN | win32 | administrat ** |

Figure 4-4 RSH and SSH installations

The installer, which is shown in Figure 4-4, collects the required parameters for the installation for each target requiring an installation, such as the user name, password, organization, Runtime server name, Runtime server address, ports, and installation locations. There is also an import function where the parameters for each server can be loaded from an XML file.

4.3.4 Mail server

Access to mail servers from both your Runtime servers and Administration servers will improve the robustness of your license management solution. TLCM has the capability to mail out various internal events to interested parties.

This capability includes e-mailing administrators from the administrator server if the procured license counts are exceeded

4.4 Installation

The following sections cover the installation of the various components of TLCM.

4.4.1 Administration server

The server must have a fully qualified host name; otherwise, the Web server will not start. To verify the fully qualified host name, click Start \rightarrow Control Panel \rightarrow System.

Select the **Computer Name** tab and check that the host name is fully qualified (Figure 4-5 on page 132).

| Advanced General | Automatic Updates Computer Name | Remote Hardware | | |
|---|--|---------------------|---|--|
| Windows use on the networ Computer <u>d</u> escription: | s the following information to iden k. TLCM 2.3 Admin Server For example: "IIS Production S | itify your computer |] | |
| Full computer name: Workgroup: | "Accounting Server". mu.itsc.austin.ibm.com WORKGROUP | | | Fully qualified with .itsc.austin.ibm.com |
| To rename this compute | r or join a domain, click Change. | <u>C</u> hange | | |
| | | | | |
| | | ncel Apr | | |

Figure 4-5 Check for fully qualified host name

If the host name is not fully qualified, click **Change**, then click **More**, and add the Primary Domain Name System (DNS) suffix.

Windows Server 2003 Service Pack 2 (SP2) issues

If you are running Windows Server® 2003 Service Pack 2, the middleware installers for WebSphere, IBM HTTP Server, and the IBM HTTP Server plug-ins need to have the prerequisites updated for Windows Service Pack 2. For further information about updating prerequisites, refer to the related IBM Technote 1259542, *IBM WebSphere Application Server might not recognize Microsoft Windows Vista as a supported operating system* at:

http://www.ibm.com/support/docview.wss?rs=180&uid=swg21259542

To update the prerequisites, either follow the technote instructions or perform the following steps:

1. For each of the maintenance.xml files in your application installation directories, find and replace the Windows 2003 Service Pack 1 dependency entry with the Service pack 2 entry, as shown in Example 4-1 on page 133.

2. Example 4-2 is the entry for Service Pack 2. Note the Key CSDVersion is now at 2 instead of 1.

Example 4-2 Windows 2003 SP2 dependency

- 3. You can locate the maintenance.xml files requiring the Service Pack 2 modification for the HTTP Server bundle (e-Assembly C87PNML) in these subdirectories:
 - AppClient\client.primary.pak
 - \IHS\ihs.primary.pak
 - UpdateInstaller\updi.primary.pak
 - \InstallationFactory\eclipse\plugins\com.ibm.ws.install.factory.was.cip.v61_ 6.1.0\mergeengine\data\OpenInstallImage\was.primary.pak
 - /plugin/plg.primary.pak
- You can locate the maintenance.xml files requiring modification for the WebSphere bundle (e-Assembly C87QTML) in the subdirectory named \WAS\was.primary.pak.

Errors

If you receive an error stating that the URL is not available for the Admin server:

1. Check that the server is not running on an alternate port:

```
http://servername:9081/slmadmin/login
http://servername:9081/slmadmin/service
```

- 2. If the server is still not available, check the WebSphere Application Server administration console to see if the application SLM_Runtime_Application has started.
- 3. Check whether the following services have started:
 - HTTP SSL
 - IBM HTTP Server 6.1
 - IBM HTTP Administration 6.1
 - The two WebSphere services as listed in Figure 4-6

| HTTP SSL Muman Interface Device Access |
|--|
| 🐝 IBM HTTP Administration 6.1 |
| BIM HTTP Server 6.1 |
| 🐝 IBM WebSphere Application Server V6.1 - IBM_TLCM_Administration_Server |
| BIBM WebSphere Application Server V6.1 - muNode01 |

Figure 4-6 TLCM Administration Server Services

- 4. Enter http://servername:9081/slmadmin/login where servername is the name of your Administration server. Note that the port number is 9081. This port number indicates that you are accessing the WebSphere TLCM servlet directly, and using this port number is a useful debugging technique. To access the servlet via the http server, use the URL to perform the redirection automatically.
- 5. Log in as the tlmroot user.

Note: As of TLCM V2.3, the timroot user default password changed from service to service01.

6. TLCM will display a warning that the organization needs to be set up (Figure 4-7 on page 135).

| | i |
|---|--------|
| Welcome to IBM Tivoli License Compliance Manager | |
| Click create to create an organization. | |
| | |
| Tivoli. License Compliance Manager Version 2.3 Image: Compliance Manager | |
| | |
| | |
| No organizations are currently registered. | |
| Explanation: | |
| The administration server database must include at least one organization. You must create at least one organization before | |
| you can create any infrastructure elements, resources, or licenses, | |
| or produce any reports. | |
| Action: | |
| Create an organization. | |
| Close message | |
| | |
| Create | |
| I | |
| 🙆 Done 🖉 🖉 Trusted site | :s //. |

Figure 4-7 Organization warning

7. Click Create.

| 🌁 http://mu:9081 - IBM Tivoli License Compliance Manager - Microsoft Interne 💶 🗵 🗶 | | | | | |
|--|---------------------------|--|--|--|--|
| Tivoli, License Compliance Manager | | | | | |
| 🖀 😂 🕹 🗎 | | | | | |
| | Manage Organizations | | | | |
| Create an Organization | , I | | | | |
| Enter the organization details. | | | | | |
| *Name IBM ITSO | Organization code IBMITSO | | | | |
| *Choose an action | * Country | | | | |
| O Do not select country | United States | | | | |
| Select country | | | | | |
| < Back Next > Finish | Clear Close Cancel | | | | |
| Done | Trusted sites | | | | |

Figure 4-8 Create an Organization

8. Populate the Name of the organization, the Organization code, and select a country. Select **Finish** to save.

Create a division

Next, you need to create a division. From the ITLCM Administration server user interface, select **Manage Resources** and then select **Divisions**. Enter your division.

The division is associated with a client, and the Runtime server is associated to the client. To make the division visible on the Runtime server deploy Web page, you can wait for the adminDownloadPeriod specified in the Runtime server's configuration file. The default value for the download period is 360 minutes (6 hours). This might not be quick enough for your build cycle, so you can force the download by restarting the Runtime server.

Restarting the Runtime Server

The Runtime server can be restarted from the TLCM Command Line Interface (CLI). The CLI can be started from Start \rightarrow All Programs \rightarrow IBM Tivoli License Compliance Manager \rightarrow Runtime server command line. Use the srvstop and srvstart commands as in Example 4-3 on page 137.

Example 4-3 Stop and start Runtime server CLI

ITLCM Runtime Server-CLI>srvstop

CODCL0013I The Tivoli License Compliance Manager runtime server is stopped

ITLCM Runtime Server-CLI>srvstart

CODCL0012I The Tivoli License Compliance Manager runtime server is started

ITLCM Runtime Server-CLI>

4.4.2 Deploy agents

The most common method for small or incremental deployments is to use the Runtime server Web page. To start the Web page, navigate to http://slmruntime/deploy on your Runtime server as shown in Figure 4-9.



Figure 4-9 Deploy the IBM License Compliance Manager Agent

The agent deploy page requires that the division name and the Runtime server are specified.

Note: You must have at least a Java SE Runtime Environment (JRE[™]) 1.4.2 installed to utilize the Web download facility. (Linux PPC and System i[™] use JRE 1.3.1.)

You can specify the JRE prerequisite automatically at the Runtime server for the Internet Explorer® (IE) browser by editing the file:

<TLM_INSTALLDIR>\runtime\SLM_Runtime_Application.er\slm_runtime.war\WEB -INF\conf\agent_install.properties

The JRE IE plug-in is shown in Example 4-4.

Example 4-4 JRE IE plug-in

parm.ie_plugin_page=http://java.sun.com/products/plugin/autodl/jinstall -1_4_1_03-windows-i586.cab

For other browsers, such as Mozilla, the agent_install.properties file also has parameters for the supported platforms and JRE download locations as shown in Table 4-1.

| Agent platform | JRE download location |
|-------------------|---|
| win32 | http://java.sun.com/j2se/1.4.1/download.html |
| aix | <pre>http://www-106.ibm.com/developerworks/java/jdk/aix/service.h tml</pre> |
| hpux | <pre>http://www.hp.com/products1/unix/java/java2/jpi/downloads/li cense_jpi_1-4-2-00_pa-risc.html</pre> |
| linux | http://java.sun.com/j2se/1.4.1/download.html |
| sun | http://java.sun.com/j2se/1.4.1/download.html |
| linuxppc | <pre>https://www6.software.ibm.com/dl/lxdk/lxdk-p</pre> |
| linux390 | https://www6.software.ibm.com/dl/lxdk/lxdk-p |

Table 4-1 Agent platforms and JRE download locations

Agent installation issues

You might have browser security settings that cause the agent to not install, as shown in Figure 4-10.



Figure 4-10 Active X security settings

If Your security settings require modification, select **Tools** \rightarrow **Internet Options** \rightarrow **Security** \rightarrow **Local Intranet** \rightarrow **Sites**. Then, add the Runtime server to the trusted site list.

Next, ensure that ActiveX® controls are enabled for the local intranet zone by clicking **Custom Level**. Scroll down to find Download signed ActiveX controls, and then select **Prompt** as indicated in Figure 4-11 on page 140.



Figure 4-11 Enable signed ActiveX controls

The installer will then install the gskit (which is used for encryption), and, depending on the platform, the following files will be installed in the default locations.

Windows

The windows default location is %WINDIR%\itlm, which translates into either C:\Windows\itml or c:\WINNT\itlm on most installations. A list of the installed files and directories is described in Table 4-2.

| File and directory | Purpose |
|--------------------|--|
| agent | Directory containing logs. The GSKIT install log \agent\log\TIVOLI_ITLM_AGT_GSKIT_WIN32^2.3.log can be found here. |
| agt_uninstall.txt | This file contains a list of Software Package Block (SPB) files to uninstall. |
| cache | This is the agent work directory. The directory must be cleared if you are having problems and need to begin a new trace. The maximum size of this directory is limited to 15 MB. |
| codeset | This is the agent codeset files. |
| keydb | This is the agent keystore. |

Table 4-2 Installed files and directories

| File and directory | Purpose |
|--------------------|---|
| nls | This contains the agent messages. |
| reboot_needed.txt | This is a file that is used to indicate if a reboot is required after the GSKIT installation. |
| scanner | This is the Common Inventory Technology (CIT) scanner. |
| swdis_tmp | This is the agent deployment staging area. |
| tlmagent.exe | This is the binary file for the agent. |
| tlmagent.ini | This is the agent settings file. Refer to Appendix B, "IBM Tivoli License Compliance Manager agents" on page 275 for details. |
| TLMAGENT0203.sys2 | This is an agent version marker file. This file indicated Version 2.3 general availability (GA). |
| tlmlog.properties | This file contains logging properties. The most important line is the trace settings via agentLogger.agentTraceLogger.level, which can be set to DEBUG_MIN, DEBUG_MID, or DEBUG_MAX. |
| tlmunins.bat | This is the uninstaller batch file. Execute this file to uninstall the agent. |
| utilities | This is a set of agent utilities, including backup and CLI. |
| wastmp | This is a directory for the WebSphere plug-in. |

Agent installation verification

Verify that your agent has contacted the Runtime server and is visible in the Administration server. Use the Administration Web server interface to check that the agent status is "Active" as in Figure 4-12 on page 142.

| License Compliance Manager | | | | | | | |
|---------------------------------------|---|--------------------------------------|-------------------|---------------------|------------------------|----------------------------|--------|
| 8 & 🖻 | | | | | | | Lo |
| Portfolio 💽 | | Agents | | | | | |
| uce Reports age Batch Reports | Search for Agents | | | tails or to delet | | | |
| age Licenses | Couronn | or an agoin to | non agon doi | | | | |
| n Licenses | Division | | lost name Proces | | or Type Agent | Status | Se |
| ne Product Properties | All | | | All | All | • | |
| dule Software Scans | | | | | | | |
| age Resources age Complex Products | *Agents matching previous selections Rows Per F | | | | | | age [|
| age Infrastructure | Dele | ete Detalis. | | | | | |
| <u>vers</u> ints | Select | Host name | Processor Type | Operating system | Runtime server name | Agent security level | St |
| age Organizations age Access | ¢ | MU | Other Other | Windows 2003 5.2 | mu_runtime_server | Min | Ac |
| ne Custom Fields | 🚺 Pa | O Page: 1 of 1 1 60 Displayed 1 of 1 | | | | | |
| ort IBM Use | | | | | | | |
| | | | | | | 🛛 🔍 Local in | tranet |

Figure 4-12 Verify agent status

4.4.3 Deploy Catalog Manager

The Catalog Manager is used to import and export the software catalog from the Administration server database. The Catalog Manager installs its own command line utility for user interactions.

Also included with the Catalog Manager are a number of debug and trace facilities, such as the problem determination tool, statisinfo, and the log viewer.

Installing the Catalog Manager

The Catalog Manager is available via Passport Advantage®. The e-assembly number for the Windows platform is C97ZSML. Unpack the assembly, or use your CD media and execute the setup file:

ITLCM_untar\setup\catalog_manager\Win32\SetupCatMan.exe.

The installer, in addition to the license agreement confirmation, requires the installation location, the host name, and the database port for the Administration server as in Figure 4-13 on page 143.



Figure 4-13 Catalog Manager installation parameters

Starting the Catalog Manager

The Catalog Manager executable is found in the *<CATALOG_INST_DIR*>\bin directory where the *CATALOG_INST_DIR* is the same as specified in Figure 4-13.

For Windows systems, there is a shortcut available. Select Start \rightarrow All Programs \rightarrow IBM Tivoli License Compliance Manager \rightarrow Catalog Manager.

The Catalog Manager then loads the catalog that shipped in the install bundle (Figure 4-14).



Figure 4-14 Catalog load message

4.4.4 Installing Fix Pack 2

Fix Pack 2 for TLCM Version 2.3 is the minimum level required for IBM Tivoli Asset Management for IT.

The significant updates to Fix Pack 2 include:

- ► The getconf CLI command for retrieving agent configuration
- The unlicevent CLI command for controlling the unlicensed session tracking feature
- The mapcomps CLI command for controlling how mapping rules apply to agents
- The hwscanguidrule CLI command for controlling the generation of unique hardware identifiers

Note: You can download the TLCM Fix Pack 2 from:

ftp://ftp.software.ibm.com/software/tivoli_support/patches/patches_2
.3.0

The fix pack directory contains files for:

- ► The agent gateway
- For each supported platform:
 - Agent deploy files and packages
 - RSH/SSH agent deploy enablers
 - Server components
 - Catalog Manager

Note: Back up your TLCM installation directory and database before beginning the upgrade.

The product.xml file

TLCM uses an XML file to record installed components. Example 4-5 on page 145 lists an example of a server with an Administration server, administration database, Runtime server, and runtime database installed at the generally available 2.3.0 release level.

Example 4-5 Installed TLCM components 2.3.0

```
<?xml version="1.0"?>
<TLM>
  <Product>
      <Name>IBM Tivoli License Compliance Manager</Name>
      <Version>2.3.0</Version>
      <InstallDir>C:\IBM\TLM</InstallDir>
     <BackupDir></BackupDir>
      <Integrated>false</Integrated>
   </Product>
   <Components>
      <Component>
         <Name>admCli</Name>
      </Component>
      <Component>
         <Name>admDb</Name>
      </Component>
      <Component>
         <Name>admSetup</Name>
      </Component>
      <Component>
         <Name>admEar</Name>
      </Component>
      <Component>
         <Name>admTcd</Name>
      </Component>
      <Component>
         <Name>rtmCli</Name>
      </Component>
      <Component>
         <Name>rtmDb</Name>
      </Component>
      <Component>
        <Name>rtmSetup</Name>
      </Component>
      <Component>
         <Name>rtmEar</Name>
     </Component>
      <Component>
         <Name>rtmEarAgent</Name>
      </Component>
      <Component>
        <Name>rtmEarWasAgent</Name>
      </Component>
```

```
<Component>
<Name>rtmTcd</Name>
</Component>
</Components>
</TLM>
```

Install

Unpack the files and install the server component by executing 2.3.0-TIV-TLCM-FP0002.exe.

The server installer will ask you for a backup location, and then, it proceeds to read the product.xml file to determine which components to upgrade. The installer then stops the administration and Runtime servers and backs up the servers before continuing with the installation.

Verifying the server installation

Check the Tivoli Common Logging directory for any installation errors in the trace files. The Tivoli Common Logging directory is often C:\Program Files\ibm\tivoli\common<*PRODUCT CODE*> where the product code for TLCM is COD.

Note: On most systems, you can find the install traces in C:\Program Files\ibm\tivoli\common\COD\logs\install\trace.

The two trace files that you need to examine are:

- trace_2.3.0-TIV-TLCM-FP0002_servers.log
- trace_2.3.0-TIV-TLCM-FP0002_db_servers.log

Update the Catalog Manager

Next, update the Catalog Manager. Unpack the installation files and execute the file 2.3.0-TIV-TLCM-FP0002-catman.exe. The installer has a single parameter for the directory to which you want the Catalog Manager backed up before the installer upgrades to Fix Pack 2.

The product.xml file after the Fix Pack 2 upgrade

The product.xml file after the upgrade has the contents shown in Example 4-6.

Example 4-6 product.xml after upgrade

```
<?xml version="1.0"?>
<TLM>
<Product>
<Name>IBM Tivoli License Compliance Manager</Name>
```

```
<Version>2.3.0.2</Version>
     <InstallDir>C:\IBM\TLM</InstallDir>
      <BackupDir>C:\IBM\TLM 23 backup\FP02</BackupDir>
      <Integrated>false</Integrated>
   </Product>
   <Components>
      <Component>
        <Name>admCli</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
     <Component>
        <Name>admDb</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
     <Component>
        <Name>admSetup</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
     <Component>
        <Name>admEar</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
      <Component>
        <Name>admTcd</Name></Component>
      <Component>
        <Name>rtmCli</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
      <Component>
        <Name>rtmDb</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
      <Component>
         <Name>rtmSetup</Name></Component>
```

```
<Component>
         <Name>rtmEar</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
         <FixPack timestamp="2008.01.31</pre>
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
     <Component>
         <Name>rtmEarAgent</Name>
         <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
         <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
     <Component>
         <Name>rtmEarWasAgent</Name>
        <FixPack timestamp="2008.01.31
18:58:51">2.3.0-TIV-TLCM-FP0001</FixPack>
         <FixPack timestamp="2008.01.31</pre>
18:58:51">2.3.0-TIV-TLCM-FP0002</FixPack></Component>
      <Component>
         <Name>rtmTcd</Name></Component>
  </Components>
</TLM>
```

Note that the installer has installed both the Fix Pack 1 and Fix Pack 2 components together as part of the Fix Pack 2 installation. The Fix Pack 1 components can be seen as 2.3.0-TIV-TLCM-FP0001, and the Fix Pack 2 components can be seen as 2.3.0-TIV-TLCM-FP0002.

The Catalog Manager has its own product.xml file, which is located in the root of the Catalog Manager installation path. Example 4-7 illustrates the product.xml file after the Fix Pack 2 upgrade.

Example 4-7 product.xml for the Catalog Manager after Fix Pack 2

```
<Component>
<Name>catman</Name>
```

```
<FixPack timestamp="2008.01.31
19:49:09">2.3.0-TIV-TLCM-F0002</FixPack></Component>
</Components>
</TLM>
```

4.4.5 Installing Fix Pack 3

After unpacking the server image, run the 2.3.0-TIV-TLCM-FP0003.exe to start the install wizard. The installer needs one parameter, which is an empty directory for the backup files.

At the end of the upgrade, the Catalog Manager indicates the installation results in the GUI. Check your results against Figure 4-15.



Figure 4-15 Server upgrade results for Fix Pack 3

Catalog Manager

Update the Catalog Manager with 2.3.0-TIV-TLCM-FP0003-catman.exe. The installer will again ask for a single parameter for an empty backup directory.

At the end of the upgrade, the Catalog Manager indicates the installation results in the GUI.

Start the Catalog Manager with Start \rightarrow All Programs \rightarrow IBm Tivoli License Compliance Manager \rightarrow Catalog Manager. On startup, the admin database login will be displayed. Enter the tlmsver user ID and password.

On successful login, the Catalog Manager will update the catalog, and you will see the message shown in Figure 4-16.



Figure 4-16 Updated catalog after Fix Pack 3 installed

Check your results against the totals in Figure 4-16.

You can also determine the catalog release that is being used by the Catalog Manager GUI. Figure 4-17 on page 151 shows the version release date on the status bar of the GUI.



Figure 4-17 Catalog Manager: Catalog version

Update catalog

Keeping your catalog current is important for accuracy.

Downloading and importing the catalog from IBM

The IBM catalog is updated periodically, normally, once a month. You can download new versions from:

http://www-306.ibm.com/software/sysmgt/products/support/IBMTivoliLicens
eManager.html

The catalog files are also available via direct ftp at:

ftp://ftp.software.ibm.com/software/tivoli_support/misc/CamdO/TivoliCat
alog

Update agents

You can use the automatic agent self-update facility to apply upgrades to the agents or to any of its corequisites, for example, the GSKit security software or the Common Inventory Technology (CIT). Following an update, the agent

normally restarts automatically. If the agent does not restart automatically, restart the computer.

The self-update facility is controlled by settings in the system.properties file of each Runtime server that allow you to enable and disable self-update and define the time interval for checks for a new version of the agent code. The settings defined for a Runtime server apply to all of the agents that are connected to it.

If a Runtime server has more than 2000 agents, performance problems and delays might occur during the period when agents are being updated. An additional, optional level of control at a division level allows you to stage the update, enabling and disabling the update of agents, served by the Runtime server, according to their division.

To enable agents for self-update, complete the following steps:

- 1. Optionally enable self-update at the division level:
 - a. Connect to the Runtime server database with a DB2 client.
 - b. Obtain the ID of a division for which you do not want to update agents. For example:

select id from rtm.division where name="division_one"

c. Change the value of the rtm.division.selfupdate column for the division in which agents are not to be updated. For example:

update rtm.division set selfupdate=0 where id=1

This statement updates the Runtime server division table, setting the value of selfupdate to 0 (disabled) for the division with ID = 1.

- d. Repeat these two steps for each division for which agents are not to be updated.
- e. Disconnect from the Runtime server database.
- 2. Enable self-update at the Runtime server level:
 - a. On the computer where the Runtime server is installed, open the system.properties file.

The system.properties file is stored in the following location:

```
<INSTALL_DIR>\runtime\SLM_Runtime_Application.ear
\slm_runtime.war\WEB-INF\conf
```

b. Ensure that the following parameters are set to the shown values:

Enables the agent self-update feature
Type: string, Range: [yes, no]
updateAgentEnabled=yes

Time interval between checks by the agent # for a new version of the agent code # at the runtime server (minutes) # Type: integer, Range: [10080-129600] updateAgentPeriod=10080

- c. Save and close the file.
- d. Stop and restart the Runtime server to apply the configuration changes.

Following the next download of agent parameters to agents, each agent will start to check the Runtime server for a changed version of the agent deployment package for the platform on which it is running. The maximum interval between checks is defined by the updateAgentPeriod. When a new version of an appropriate agent deployment package is found, the agent downloads it and applies the changes that it contains. These changes can be changes to the agent itself or changes to one of its corequisites. If the upgrade fails to apply a change, all changes made up to that point are rolled back to leave the agent in its original state.

When all agents have been upgraded, reset the updateAgentEnabled parameter to No and restart the Runtime server.



Tivoli License Compliance Manager for z/OS

IBM Tivoli License Compliance Manager for z/OS enables IT to align software spending with business priorities.

It identifies software inventory, measures use activity, and automatically links license entitlements to help manage costs and license compliance in mainframe environments. IBM Tivoli License Compliance Manager for z/OS:

- Monitors software use trends to help you plan financially optimal capacity upgrades and server consolidations, allocate additional resources to priority projects, and plan the migration of old-version and duplicate-function software
- Proactively manages and verifies license compliance
- Identifies software inventory with no or low-use activity to reduce unnecessary license fees and allocate expensive low-use software to the appropriate department, enhancing the charge-back process
- Enables you to effectively prepare for contract negotiations and audits with key information. Reports on the MIPS capacity of each LPAR under which software is running Improves disaster recovery planning with software-use activity information.
- Reports on the millions of instructions per second (MIPS) capacity of each logical partition (LPAR) under which software is running

- ► Improves disaster recovery planning with software use activity information
- Solves major information challenges so that you can have effective mainframe software asset management by identifying software inventory, measuring product use activity by LPAR, and automatically linking license entitlements to installed inventory and use activity

5.1 Installation

The chapter contains information about:

- Planning the installation of the distributed components to determine the installation directories
- Installation modes
- Required authorization
- Information that must be gathered before you start the installation of the distributed server or client components
- Information about the installation of the prerequisite products

5.1.1 Introduction to distributed components

IBM Tivoli License Compliance Manager for z/OS is a product that allows the storage, search, retrieval, and manipulation of software and license information, as well as basic hardware inventory, software inventory, product information, and license usage information. IBM Tivoli License Compliance Manager for z/OS also can import hardware, software, and license usage data from other sources. When license, inventory, and usage information is available, you can perform software compliance and cost optimization functions.

You can install the distributed components of IBM Tivoli License Compliance Manager for z/OS either interactively with an installation wizard (graphical user interface) or silently with installation parameters that are supplied from a response file. The installation wizard is intended to provide an easy-to-use installation that minimizes technical knowledge requirements to install the distributed client or server components for a proof of concept deployment scenario. The installation wizard is not intended to address all types of complex, multi-server installations.

You can perform a full, new installation of all distributed components on the same computer or a single component installation where each distributed component is installed on a different computer, or you can use a combination of these deployment scenarios. IBM Tivoli License Compliance Manager for z/OS supplies a separate installation wizard for the distributed client and server components:

- The distributed server components include servlets, business and data layers, import service, reporting, and databases. The installation for the distributed server offers the following components:
 - Asset Management server

- Asset Management database
- Data Aggregation server
- Data Aggregation database

The Data Aggregation commands (daCLI command shell) are installed when the Data Aggregation server component is installed. These commands import and export data from the asset management products and other sources (Figure 5-1).



Figure 5-1 Distributed server components: IBM Tivoli License Compliance Manager for z/OS

- ► The installation for the distributed client offers the following components:
 - Tivoli Asset Compliance Center (Asset Compliance perspective) is the primary user interface for the client.
 - The Import Command Line Interface (import) and the Data Aggregation commands (daCLI command shell) are installed when the Tivoli Asset Compliance Center component is installed. These commands import and export data from the asset management products and other sources.

- The Administration Console is the Administration perspective of the Tivoli Asset Compliance Center.
- The console manages security, administers users, and configures the application.

Typically, you use the installation wizard for all client installations.

The majority of server installations have more complex requirements, such as using multiple computers for the servers, finer control over database creation parameters, and so on. For these more complex installations, you typically create a response file and use a silent installation. As an option, the installation wizard can create a response file for silent installations that can be performed at a later time or on another computer.

5.1.2 Plan for your installation of the distributed components

You plan the installation of the distributed components of IBM Tivoli License Compliance Manager for z/OS (Figure 5-2 on page 160) to determine:

- The components to install
- The installation directories
- The installation modes to use
- The required authorization
- The information that must be gathered before you start the installation of the distributed server or client components
- Information about the installation of the prerequisite products



Figure 5-2 IBM Tivoli License Compliance Manager for z/OS structure and flow

Note: IBM Tivoli License Compliance Manager for z/OS, Version 4.2, supports only a new installation. This version does not support any upgrade scenarios.

Installation modes

Use either the graphical user interface, which prompts you for input, or the silent installation using a response file.

The installation wizard can only install the distributed client, server, and database components on the computer where the installation wizard is running.

Installation considerations

You must install distributed client and server components as a new installation.

You can perform a full installation of all distributed components on the same computer or a single component installation where each distributed component is installed on a different computer, or you can use a combination of these
deployment scenarios. Use the deployment scenarios in Figure 5-3 and Figure 5-4 on page 162, as a guide.



Figure 5-3 Minimal configuration for IBM Tivoli License Compliance Manager for z/OS: Single computer with WebSphere Application Server

The minimum configuration scenario is a server running both the Asset Management server and the Data Aggregation server, as well as DB2 on the same computer with two databases: one database for the Asset Management server and the other database for the Data Aggregation server.

For optimum performance and flexibility, use the following deployment scenario (Figure 5-4 on page 162):

- One computer for WebSphere Application Server and Asset Management server with AlphaBlox®
- One computer for WebSphere Application Server and Data Aggregation server
- One computer for DB2 and Asset Management database
- One computer for DB2 and Data Aggregation database



Figure 5-4 Alternate configuration with four servers

Use the DB2 features for backup and restore functions and disaster recovery.

Note: Refer to the publications for WebSphere Application Server and DB2 to plan for and size your DB2 database and WebSphere Application Server.

For distributed server component or client component installations, a check is performed to ensure that the user has administrative rights on Windows or has root privileges on Linux. If the user does not have the required authority, the installation process ends.

As an option, the installation wizard can create a response file to be used for silent installations at a later time or for use on another computer.

You can set up a distributed server for IBM Tivoli License Compliance Manager for z/OS as a test system in the same way that you install and configure a production system. After testing is complete, you can move the test system into production status by creating a copy of the test database.

The test database contains all of the required data, such as user definitions, user authorization data with links to external authentication data, and so on.

Operating systems

You can install the distributed server and client components on the following operating systems listed in Table 5-1.

Table 5-1 Operating systems

| Operating system | Distributed server and database components | Tivoli Asset Compliance Center (Asset Compliance perspective) | Administration Console (Administration perspective of the Tivoli Asset Compliance Center) |
|--|---|---|---|
| Red Hat Enterprise Linux AS+ 3.0 IA32 | Yes | Yes | Yes |
| Windows Server 2003 Standard | Yes | Yes | Yes |
| Windows Server 2003 Enterprise | Yes | Yes | Yes |

You can configure and deploy the distributed server components to any application server. However, running multiple applications in one WebSphere Application Server complicates the administration, because taking the WebSphere Application Server down will impact the availability of multiple applications.

The following items identify how the distributed server components address multiple applications running in the same application server:

- The distributed server components will not attempt to share resources, such as the Java Database Connectivity (JDBC) Provider or Secure Sockets Layer (SSL), with other applications. When resources are not shared, the distributed server components can be installed or removed without impacting other applications running in the same WebSphere Application Server.
- If any distributed server components are installed into a running application server, the WebSphere Application Server must be stopped and restarted in order to implement the configuration changes.
- After a distributed server component is installed, WebSphere Application Server can be automatically started. If the application server was already started, the application server must be restarted in order to implement changes for the distributed server component.

The removal of a distributed server component does not delete WebSphere Application Server.

5.1.3 Installation process

The installation process consists of the following phases:

- Initialization for locale and disk resources
- Requests for parameter values
- Installation and configuration of the distributed components

During the initialization phase, the language selection window is displayed. After the distributed components are selected for installation, the required disk space needed for each component and the available space are determined. If the disk space requirements are not satisfied, the installation wizard exits and a warning message is displayed. You must make more disk space available.

During the parameter request phase, the values for the parameters are collected and verified. For example, the value for a port number has to be numeric, and the port must not be in use.

After a successful installation of the distributed server components, the only required configuration is to set up the appropriate user definitions for authentication and authorization. By default, after installation, no users are authorized to connect to the IBM Tivoli License Compliance Manager for z/OS distributed components.

After a successful installation of the distributed client components, the only required configuration is pointing the client component to the appropriate IBM Tivoli License Compliance Manager for z/OS Asset Management server.

If the installation wizard determines that certain distributed components selected for installation are already installed and have not been modified by the user, such as changes to the configuration file, those components will not be installed again. The installation wizard shows a progress bar during long running installation steps.

Tip: We recommend that you first install the distributed server components before installing the distributed client components, although the installation process does not require this sequence.

Prerequisite products

The following prerequisite products are required and must be installed and configured before the distributed server and database components are installed.

These prerequisite products are not automatically installed, but they are supplied with IBM Tivoli License Compliance Manager for z/OS. Refer to the corresponding product documentation for installation directions. Typically, you can install the prerequisite products using their default values:

 WebSphere Application Server, Version 6.0.2.x (where x is equal to or greater than 5, such as 6.0.2.5)

Required for the Asset Management server and the Data Aggregation server

DB2 Universal Database[™], Version 8.2.x (where x is equal to or greater than 0, such as 8.2.0)

Required for the Asset Management database and the Data Aggregation database

DB2 Alphablox, Version 8.4 (where x is equal to or greater than 0, such as 8.4.0) is required for the Asset Management server.

DB2 Alphablox must be installed in the same WebSphere Application Server instance as the WebSphere Application Server instance that is used for the server component. Therefore, DB2 Alphablox and WebSphere Application Server must be on the same computer.

DB2 Alphablox installation

To access DB2 Alphablox, the user ID and password have to be configured with the Java Authentication and Authorization Service (JAAS) component of WebSphere Application Server. In the case of local location selection, the Alphablox-related JavaServer[™] Pages (JSPs) are installed under the context provided in the base URL in the Alphablox Information Panel.

For the base Alphablox platform, the Alphablox enterprise archive (EAR) files must be installed in a secure WebSphere Application Server environment, and access to the platform is controlled by users supplying credentials that are mapped to the applicable roles in the application. In turn, the user IDs that are mapped to these roles are JAAS entries.

Similarly, access to the AssetManagementReport Enterprise Application is controlled by a JAAS ID that is mapped to the correct role defined within the application. HTTP requests sent to the AssetManagementReport Enterprise Application include credentials embedded in the request itself, and WebSphere Application Server handles the authentication operation. AssetManagementReport Enterprise Application deserializes the request object, sends it over to Alphablox as a ResultSet, and produces a report.

However, AssetManagementReport Enterprise Application is not installed in Alphablox itself, and instead, it is installed as a stand-alone EAR file listening on a unique context root (/ITAMReports). Also, from the IBM Tivoli License

Compliance Manager for z/OS server perspective, no JAAS entries are used to connect to the application. Instead, entries are read from the itam.properties file for the base URL, context root, JavaServer Pages (JSP[™]) path, user ID, and password. A URL connection is opened, the request is written, and then the response from AssetManagementReport Enterprise Application is read and sent back to the user interface.

Refer to the DB2 Alphablox documentation for details.

Required authorization for distributed server installations

For server or client installations, a check is performed to ensure that the user has administrative rights on Windows or has root privileges on Linux. If the user does not have the required authority, the installation process ends.

Error handling during the installation

Files for both tracing and logging errors are created for the interactive and silent installations. Exceptions, errors, and warnings are logged for all installations, but they are only displayed during interactive installations.

Installation directories

The following directories are the default directories for the servers, clients, and logs:

- ► For Windows: Tivoli_common_dir is C:\Program Files\IBM\tivoli\common.
- ► For Linux: Tivoli_common_dir is opt/IBM/tivoli/common.

Table 5-2 on page 167 shows the directories for the Windows operating system, and Table 5-3 on page 167 shows the directories for the Linux operating system.

| Distributed component or log | Directories for the Windows operating system |
|--|---|
| Server components | C:\Program Files\IBM\ITACC |
| Client components for an administrative user | C:\Program Files\IBM\ITACC |
| Client components for a non-administrative user | %USERPROFILE%\IBM\ITACC |
| Installation and removal message log | Tivoli_common_dir\IXU\logs\client\install\ message |
| Tivoli_common_dir\IXU\log\server\install\ message | N/A |
| Installation and removal trace log | Tivoli_common_dir\IXU\logs\client\install\t race |
| Tivoli_common_dir\IXU\logs\server\install \trace | N/A |

Table 5-2 Directories for the Windows operating system

Table 5-3 Directories for the Linux operating system

| Distributed component or log | Directories for the Linux operating system |
|---|---|
| Server components | /opt/IBM/ITACC |
| Client components for an administrative user | /opt/IBM/ITACC |
| Client components for a non-administrative user | \$HOME/IBM/ITACC |
| Installation and removal message log | Tivoli_common_dir/IXU/logs/client/install/ message |
| Tivoli_common_dir/IXU/logs/server/install /message | N/A |
| Installation and removal trace log | Tivoli_common_dir/IXU/logs/client/install/t race |
| Tivoli_common_dir/IXU/logs/server/install /trace | N/A |

5.1.4 Gather installation information for distributed components

This topic lists the installation parameters that are used when installing the distributed server or client components.

Gather the following installation information before you begin a distributed server installation:

- Location of the DB2 database, the database administrator ID, and administrator password
- Location of the WebSphere Application Server, administrator ID, administrator password, profile name, and server name
- Location of the DB2 Alphablox
- Port number for a remote Asset Management database (if on a computer other than the Asset Management server)
- Port number for a remote Data Aggregation database (if on a computer other than the Data Aggregation server)

Gather the following installation information before you begin a client installation:

- Collect the WebSphere Application Server host name, port number, and context name so that a connectivity test can be performed between the client computer and the WebSphere Application Server computer.
- If the user performing the installation has administrative (or root) privileges, determine the user access permission (all users or only the current user).

5.1.5 Installing distributed server components

You can use the installation wizard to interactively provide values for the installation parameters or create a response file to provide the installation parameters and then perform the installation silently.

Use one of the following installation modes:

- Installing the same component on more than one computer and using the same parameter values for each installation
- Running the installation in less time than using the interactive installation wizard
- ► Using the interactive installation wizard with preloaded parameter values

Typically, you install the distributed server components before the distributed client components.

Note: Before you begin an installation, you must install the prerequisite products.

Specifying host name values

The preferred value for a host name is a host name alias, which allows the application to use a host name that is defined on the Domain Name System (DNS) server, and the host name will not change, even if the Tivoli Asset Compliance Center server is moved to a different IP address or host.

If you specify an IP address, and the IP address changes, the computer will not be able to contact the Tivoli Asset Compliance Center server. For example, the IP address can change in a Dynamic Host Configuration Protocol (DHCP) environment where dynamic IP addresses are assigned, and a computer might acquire a new address periodically or after restarting the computer.

If you specify a short host name, computer systems outside of the Tivoli Asset Compliance Center server's subnet must be configured to resolve addresses in the Tivoli Asset Compliance Center server subnet. If the computers are configured incorrectly, the client application on those computers cannot contact the Tivoli Asset Compliance Center server. Therefore, we recommend that you use a short host name only if your environment contains a single IP domain or if all of the computer systems are configured with the correct domain search information.

Interactively installing the distributed server components

This topic describes how to install the IBM Tivoli License Compliance Manager for z/OS distributed server components using the graphical user interface. You can also silently install the distributed server components using a response file to supply the installation parameters.

These instructions assume that you have WebSphere Application Server, DB2, and DB2 Alphablox installed.

During the installation, you select the distributed server components that you want to install at this time. At a later time, you can run the installation wizard again to install additional distributed server or client components on this computer.

As an option, you can prepare a response file and then start the interactive installation with the values supplied from the response file. For details, refer to "Silently installing the distributed server components" on page 174.

During a distributed server component installation, the installation wizard completes the following tasks:

- The location of the database, the database administrator ID, and the administrator password are supplied. With these values, the installation wizard silently creates the database alias and schema. Then, the installation wizard loads the metadata and creates the table indexes.
- The WebSphere Application Server location, administrator ID, and administrator password are supplied. With these values, the installation wizard configures the business layer and data layer with WebSphere Application Server.

To install the distributed server components, locate and start the installation setup file (the file names are shown in Table 5-4) on the product distribution media or other source for the distributed server installation image.

| Operating system | Installation file name |
|------------------|------------------------|
| Windows | setupwin32.exe |
| Linux | setupl inux bin |

Table 5-4 Installation setup file names by operating system

The steps to install the distributed server components interactively are:

- 1. Select a language from the drop-down list. The language choice determines the locale in which the installation wizard is run. If the selected language locale is not present in the computer, the language of the computer is used.
- 2. Read the Welcome window.
- 3. Read the Software License Agreement. You must accept the terms in the license agreement by clicking the corresponding radio button.

Specify the installation directory. Use one of the following techniques:

- Use the default directory.
- Type a directory name in the entry field.
- Click **Browse** to select a directory.

Specify a valid directory name. If the directory does not exist, the directory will be created. If the directory path contains spaces, enclose the path in double quotation marks.

If several of the distributed server components are already installed on this computer, the directory selection window is not displayed.

- 4. Select the type of installation, Custom or Full:
 - Custom: You select the distributed server and database components that you want to install at this time. At a later time, you can run the installation wizard again to install additional distributed server and database components.

You can install one or more of the following distributed components:

- Asset Management server
- Asset Management database
- Data Aggregation server

Note: If you are installing the servers and databases on separate computers or sequentially on the same computer, the Asset Management server, Asset Management database, and Data Aggregation database must be installed and running before the Data Aggregation server is installed.

The Data Aggregation commands (daCLI command shell) are installed when the Data Aggregation server component is installed. These commands import and export data from the asset management products and other sources.

 Full: All distributed server and database components that are not already installed on this computer will be installed.

The distributed client components have a separate installation wizard.

The installation wizard determines if any distributed server and database components are already installed. Any previously installed server and database components are displayed with a check mark and are not available for selection.

By default, a Custom installation is selected.

- 5. Specify the location of a supported version of DB2. Use one of the following techniques:
 - Use the default directory:
 - For Windows: C:\Program Files\IBM\SQLLIB
 - For Linux, identify the home directory of the DB2 instance owner: /opt/IBM/SQLLIB
 - Type a directory name in the entry field.
 - Click Browse to select a directory.

If a supported version of DB2 is not installed on this computer and you are installing a distributed database component, you must stop this installation by clicking **Cancel**, install DB2, and then restart this installation.

- 6. Specify the location of a supported version of WebSphere Application Server. Use one of the following techniques:
 - Use the default directory:
 - For Windows: C:\Program Files\IBM\WebSphere\AppServer
 - For Linux: /opt/IBM/WebSphere/AppServer
 - Type a directory name in the entry field.
 - Click Browse to select a directory.

If a supported version of WebSphere Application Server is not installed on this computer and you are installing a distributed server component, you must stop this installation by clicking **Cancel**, install WebSphere Application Server, and then restart this installation.

- 7. After you click **Next**, a progress bar appears while WebSphere Application Server information is retrieved.
- 8. Specify the location of a supported version of DB2 Alphablox, and then select the WebSphere Application Server profile name and server name.

Use one of the following techniques to locate a supported version of DB2 Alphablox:

- Use the default directory:
 - For Windows: C:\Program Files\IBM\DB2 Alphablox
 - For Linux: /opt/IBM/DB2 Alphablox
- Type a directory name in the entry field.
- Click Browse to select a directory.

If a supported version of DB2 Alphablox is not installed on this computer, you must stop this installation by clicking **Cancel**, install DB2 Alphablox, and then restart this installation.

9. Select the WebSphere Application Server profile name and server name.

Select the WebSphere Application Server profile name for the Alphablox Platform application from the drop-down list. The default profile name is default.

- 10. Select the WebSphere Application Server name where the Alphablox Platform application is deployed from the drop-down list. The default server name is named server1.
- 11. After you click **Next**, a progress bar appears while the deployed location of the Alphablox Platform application is verified.

If the Alphablox Platform application is not deployed in the specified WebSphere Application Server profile name and server name, a message appears, and you must click **Back** to specify the correct deployment location.

12. Specify and confirm the password for the user named "itamusr".

The installation wizard creates a user named "itamusr" on all computers where a distributed server or database component is installed.

If you are installing one or more distributed components on the target computer for the first time, supply the password for the "itamusr" user. If you have already installed one or more components on the target computer, the user already exists and you need to supply the current password for the "itamusr" user.

The password rules include:

- Maximum length of 20 characters
- Allowed characters are: A-Z, a-z, 0-9, +, and -

The password rules of the target computer must be satisfied.

13. Specify the WebSphere Application Server profile name and the WebSphere Application Server name for the Tivoli Asset Compliance Center.

Select the WebSphere Application Server profile name for the Tivoli Asset Compliance Center from the drop-down list. The default profile name is named default.

Select the WebSphere Application Server name where the Tivoli Asset Compliance Center is deployed from the drop-down list. The default server name is ITACC_Application_Server.

- 14. After clicking **Next**, if the application server specified (such as ITACC_Application_Server) is not yet created, the installation wizard provides a choice to create it.
- 15. Specify the port number for the remote Asset Management database with which the Asset Management server communicates. The default value is 50000.

Note: DB2 uses 50000 as the default port number for communication with the application. The default value is selected when DB2 is initially installed. If you specified a different port number during the DB2 installation, you must enter that port number. The port number must be static (not dynamic).

Specify the port number for the remote Data Aggregation database with which the Data Aggregation server communicates.

The default value is 50000.

Use a valid port number. The value for the port number has to be numeric, and the port must not be in use.

- 16. Review the installation specifications window and click **Next** to begin the server installation. An installation progress window is displayed.
- 17.Read the installation summary.
- 18. The installation is complete. Click **Finish**.

If the installation is successful, the port number for the Asset Management server is displayed, such as 9081. Record the port number, because that port number is used during the client installation.

If the installation fails, an error window is displayed. Use the message log file (msg_server.log) and trace file (trace_server.log), as needed, for installation details and error handling.

After a successful installation of the distributed server component, metadata is loaded in the database; the server components are configured with WebSphere Application Server; and other configurations and deployments are completed.

After a successful installation of the distributed server component, the only required configuration is to set up the appropriate user definitions for authentication and authorization. By default, after installation, no users are authorized to connect to the IBM Tivoli License Compliance Manager for z/OS servers.

Silently installing the distributed server components

This topic describes how to silently install the IBM Tivoli License Compliance Manager for z/OS distributed server components using a response file to supply the installation parameters. You can also use a graphical user interface to install the distributed server components.

These instructions assume that you have WebSphere Application Server, DB2, and DB2 Alphablox installed.

When preparing the response file, you supply the following information:

 Identify the distributed server components that you want to install on this computer.

You can install one or more of the following components:

- Asset Management server
- Asset Management database
- Data Aggregation server

Note: If you are installing the servers and databases on separate computers or sequentially on the same computer, the Asset Management server, Asset Management database, and Data Aggregation database must be installed and running before the Data Aggregation server is installed.

Depending on the distributed components being installed, values for certain parameters are not needed:

- The installation directory for the distributed server and database components
- The location of WebSphere Application Server, DB2, and DB2 Alphablox
- Setup information, such as port numbers, the instance owner, and passwords, for WebSphere Application Server, DB2, and DB2 Alphablox

At a later time, you can run the installation again to install the distributed client components or additional server components on this computer or another computer.

When preparing the server response file, use the following parameter descriptions, the sample response files provided next, and the server response file that is shipped with the product as a guide. Use a plain text editor when creating and editing your response file, such as Notepad on Windows.

Parameters for your response file

The parameters for all IBM Tivoli License Compliance Manager for z/OS component installations are grouped into several tables (Table 5-5 on page 176 through Table 5-14 on page 186) based on the distributed components that are being installed. If you are not installing all distributed server and database components, you can omit those corresponding parameters from your response file.

Note: If a parameter is not included in the response file, the default value, if any, is used. If the response file contains a parameter that does not apply, a warning message about that parameter is written to the message log file.

| Parameter | Description | Samples and tips |
|-----------------------------|--|---|
| -P installLocation | Identifies the installation directory. | C:\Program Files\IBM\TACC is the Windows directory |
| | Specifies a valid directory name. | /opt/IBM/TACC is the directory on Linux |
| | If no value is specified, the installation location defaults to the directory from where the installation wizard was started. | If the directory path contains spaces, enclose the path in double quotation marks. |
| | | If the directory does not exist, it will be created. |
| -W baseConfig.itamusrPwd | Specifies the password for "itamusr" user. The installation wizard creates a user named "itamusr" on all computers where a distributed server or database component is to be installed. If you are installing one or more distributed components on the target computer for the first time, supply the password for the "itamusr" user. If you have already installed one or more distributed components on the target computer, the user already exists and you must supply the current password for the "itamusr" user. | myPassword The password rules include: Maximum length of 20 characters Allowed characters are: A-Z, a-z, 0-9, +, and Password rules of the target computer must be satisfied. Never store, e-mail, or otherwise make a response file available that has a password saved in it. The password must be added to the response file just before the installation is run, and then it must be removed after the installation has completed successfully. Make sure that file and directory permissions prevent unauthorized persons from accessing the response file. |

 Table 5-5
 IBM Tivoli License Compliance Manager for z/OS parameters

| Parameter | Description | Samples and tips |
|---|--|--|
| -W setupType.selectedSetup TypeId | Determines the setup type used when installing the product. Valid values are Custom or Full. | Full If you want to install any distributed client components, you must use a separate installation. |
| | Use Custom to select any combination of distributed server and database components. | |
| | Use Full to install the Asset Management server and its database and the Data Aggregation server and its database. | |
| -P CAMTServer.active | Determines if the Asset Management server component is to be installed. Valid values are true or false. Use true to install this component. The default value is true. | true Any value other than true is considered to be a value of false. |
| -P CAMTDB.active | Determines if the Asset Management database component is to be installed. Valid values are true or false. Use true to install this component. The default value is false. | true Any value other than true is considered to be a value of false. |
| | | |

| Parameter | Description | Samples and tips |
|----------------------|---|---|
| -P DAGGServer.active | Determines if the Data Aggregation server component is to be installed. | true Any value other than true is considered to be a value of |
| | Valid values are true or false. Use true to install this component. The default value is true. | laise. |
| | The Data Aggregation commands (daCLI command shell) are installed when the Data Aggregation server component is installed. | |
| -P DAGGDB.active | Determines if the Data Aggregation database component is to be installed. | true Any value other than true is considered to be a value of false. |
| | Valid values are true or false. Use true to install this component. The default value is false. | |
| -G licenseAccepted | Determines if the product license has been accepted. The value must be true before a silent installation will run. | You must uncomment this parameter and change the value to "true" before a silent installation will be successful. |
| | The valid values are true or false. The default value is false. | -G licenseAccepted=true |
| | If the value is false, the silent installation will not run. | |

Example 5-1 on page 179 shows a partial sample response file.

Example 5-1 Partial sample response file

```
-P installLocation="C:\Program Files\IBM\ITACC"
-W baseConfig.itamusrPwd="myPassword"
-W setupType.selectedSetupTypeId="Custom"
-P CAMTServer.active="true"
-P CAMTDB.active="true"
-P DAGGServer.active="false"
-P DAGGDB.active="false"
-G licenseAccepted="true"
```

Table 5-6 provides the parameters for any distributed server component.

| Parameter | Description | Samples and tips |
|---|--|--|
| -W appServerPathPanel.locati onPath | Identifies the location of the supported WebSphere Application Server. | If the directory path contains spaces, enclose the path in double quotation marks. C:\Program Files\WebSphere\AppServ er |
| -W serverConfig.profileName | Identifies the WebSphere Application Server profile name where the application has to be deployed and configured. | default |
| -W serverConfig.appSrvName | Identifies the WebSphere Application Server name where the application has to be deployed and configured. If the server does not exist, the installation wizard creates it. | server1 |

Table 5-6 Parameters for any distributed server component

Example 5-2 shows a sample partial response file.

Example 5-2 Partial sample response file

```
-W appServerPathPanel.locationPath="C:\Program
```

```
Files\WebSphere\AppServer"
```

-W serverConfig.profileName="default"

Table 5-7 lists the parameter for installations that include any distributed database component.

| Parameter | Description | Samples and tips |
|--|---|---|
| -W databaseServerPathPanel .locationPath | Identifies the location of the supported DB2 database. | If the directory path contains spaces, enclose the path in double quotation marks. |
| | For Windows, use the directory having SQLLIB as the subdirectory: | |
| | C:\Program Files\IBM\DB2 | |
| | For Linux, use the directory named as the target DB2 instance: | |
| | /home/db2inst1 | |

Table 5-7 Parameter for installations with any distributed database component

An example of a partial sample response file is:

-W databaseServerPathPanel.locationPath="/home/db2inst1"

Table 5-8 on page 181 shows the parameters for installations of the Asset Management server component.

| Parameter | Description | Samples and tips |
|---------------------------------------|---|---|
| -W aBloxPathPanel.locationP ath | Identifies the location of the supported DB2 Alphablox. | If the directory path contains spaces, enclose the path in double quotation marks. |
| | | C:\Program Files\IBM\DB2 AlphaBlox |
| -W aBloxPathPanel.profileNa me | Identifies the WebSphere Application Server profile name where the Alphablox Platform application has been deployed and configured. | default |
| -W aBloxPathPanel.appSrvNa me | Identifies the WebSphere Application Server name where the Alphablox Platform application has been deployed and configured. | server1 |
| | If the server does not exist, the installation wizard creates it. | |
| -W dbInstallCAMT.portNumbe r | Specifies the port number for the remote Asset Management database with which the Asset Management server communicates. The default value is 50000. | 50000 Use a valid port number. The value for the port number has to be numeric, and the port must not be in use. |

Table 5-8 Parameters for installations of Asset Management server component

Example 5-3 shows a partial sample response file.

Example 5-3 Partial sample response file

```
-W aBloxPathPanel.locationPath="C:\Program Files\IBM\DB2 AlphaBlox"
-W aBloxPathPanel.profileName="default"
```

```
-W aBloxPathPanel.appSrvName="server1"
```

```
-W dbInstallCAMT.portNumber="50000"
```

Table 5-9 shows the parameter for installations of the Asset Management server component including the remote Asset Management database component.

| Parameter | Description | Samples and tips |
|------------------------------|---|--|
| -W dbInstallCAMT.hostName | Specifies the host name, host name alias, short host name, or the IP address where the Asset Management database is installed (or will be installed). | localhost AM_Host.myOrganization. com 9.26.42.111 |
| | If the Asset Management database is already installed (or will be installed) on this computer, either during this installation or later, set this parameter to "localhost". | |
| | The default value is "localhost". | |

Table 5-9 Parameter for installations

An example of a partial sample response file is:

```
-W dbInstallCAMT.hostName="localhost"
```

Table 5-10 shows the parameter for installations of the Asset Management database component.

| Table 5-10 Parameter for Asset Management database component install | ations |
|--|--------|
|--|--------|

| Parameter | Description | Samples and tips |
|-------------------------------------|---|--|
| -W dbInstallCAMT.dbAdminU ser | Specifies the user ID of the owner of the database instance on the computer. Linux only. | db2inst1 Use this parameter only if -P CAMTDB.active=true. |

An example of a partial sample response file is:

-W dbInstallCAMT.dbAdminUser="db2inst1"

Table 5-11 on page 183 shows the parameter for installations of the Data Aggregation server component.

| Parameter | Description | Samples and tips |
|------------------------------|---|--|
| -W dbInstallDAGG.portNumb | Specifies the port number for the remote Data | 50000 |
| er | Aggregation database with which the Data | Use a valid port number. |
| | Aggregation server communicates. | The value for the port number has to be numeric, and the port must not be in |
| | The default value is 50000. | use. |

 Table 5-11
 Parameter for installations of Data Aggregation server component

An example of a partial sample response file is:

-W dbInstallDAGG.portNumber="50000"

Table 5-12 lists the parameter for the installation of the Data Aggregation server component, including the remote Data Aggregation database component.

Table 5-12 Parameter for installation

| Parameter | Parameter Description Samples and t | |
|------------------------------|---|--------------------------------|
| -W dbInstallDAGG.hostName | Specifies the host name, host name alias, short host | localhost |
| | name, or the IP address where the Data Aggregation database is | DA_Host.myOrganization. com |
| | installed (or will be installed). | 9.26.43.124 |
| | If the Data Aggregation database is already installed (or will be installed) on this computer, either during this installation or later, set this parameter to "localhost". | |
| | The default value is "localhost". | |

An example of a partial sample response file is:

-W dbInstallDAGG.hostName="localhost"

Table 5-13 on page 184 shows the parameters for the installation of the Data Aggregation server component, but not the Asset Management server component.

| Parameter | Description | Samples and tips | |
|---------------------------|---|---|--|
| -W DAConfig.hostnameAM | Specifies the host name, host name alias, short host name, or the IP address of the Asset Management server with which the Data Aggregation server will communicate. | AM_Host.myOrganization. com 9.26.42.132 | |
| -W DAConfig.portAM | Specifies the port number for the remote Asset Management server with which the Data Aggregation server communicates. The default value is 50000. | 50000 Use a valid port number. The value for the port number has to be numeric, and the port must not be in use. | |
| -W DAConfig.userName | Specifies the user name to connect to the Asset Management server. If the user name and password have not been changed since the Asset Management server was installed, you can use the default value. The default value is admin. | admin | |

Table 5-13 Installation parameters

| Parameter | Description | Samples and tips |
|---------------------|--|--|
| -W DAConfig.userPwd | Specifies the user password to connect to the Asset Management server. | myPassword The password rules include: Maximum length of 20 characters Allowed characters are: A-Z, a-z, 0-9, +, and Password rules of the target computer must be satisfied. Never store, e-mail, or otherwise make a response file available that has a password saved in it. The password must be added to the response file just before the installation is run and then removed after the installation has completed successfully. |
| | | Make sure that file and directory permissions prevent unauthorized persons from accessing the response file. |

Example 5-4 is an example of a partial sample response file.

Example 5-4 Partial sample response file

- -W DAConfig.hostnameAM="your_Asset_Management_host_name"
- -W DAConfig.portAM="9443"
- -W DAConfig.userName="admin"
- -W DAConfig.userPwd="myPassword"

Table 5-14 on page 186 shows the parameter for the installation of the Data Aggregation database component.

Table 5-14 Installation parameter

| Parameter | Description | Samples and tips |
|-------------------------------------|---|--|
| -W dbInstalIDAGG.dbAdminU ser | Specifies the user ID of the owner of the database instance on the computer. Linux only. | db2inst1 Use this parameter only if -P DAGGDB.active=true. |

An example of a partial sample response file is:

-W dbInstallDAGG.dbAdminUser="db2inst1"

Response file considerations

When creating and using your response file, consider the following items:

- Parameter values are case sensitive. When a set of allowable values is specified, the values must use only the case that is indicated.
- If the installation is started with a response file, the current locale remains in effect if the locale is present on the computer. Otherwise, the locale defaults to the language of the computer.

Steps to silently install the distributed server components

To perform a silent installation of the distributed server components:

Prepare your response file.

Use the parameter descriptions, sample values, and sample response file as a guide.

If the value contains a space character, enclose the value in double quotation marks ("my value"). Double quotation marks are allowed for all values.

If you include an installation parameter in the response file, you must provide a value for that parameter.

- ► To run the installation wizard with the parameters from your response file, issue one of the following commands. The response file location is relative to the current working directory. Locate the installation setup file on the product distribution media or other source for the server installation image:
 - Silently running the installation wizard with a response file:
 - **Windows**: To silently run the installation wizard with the response file, use the following command:

setupwin32.exe -options response_file_name -silent

• Linux: To silently run the installation wizard with the response file, use the following command:

```
setupLinux.bin -options response_file_name -silent
```

- Interactively running the installation wizard with a response file:
 - **Windows**: To interactively run the installation wizard using values from the response file, use the following command:

setupwin32.exe -options response_file_name

• **Linux**: To interactively run the installation wizard using values from the response file, use the following command:

setupLinux.bin -options response_file_name

Silently running the installation wizard without a response file

The default values are used for all of the installation parameters. The value for licenseAccepted must be set to "true" before a silent installation will be successful, so include that parameter on the command line:

• **Windows**: To silently run the installation wizard *without* the response file, use the following command:

```
setupwin32.exe -silent -G licenseAccepted="true"
```

• Linux: To silently run the installation wizard *without* the response file, use the following command:

```
setupLinux.bin -silent -G licenseAccepted="true"
```

 Silently running the installation wizard and using the command line to specify an installation parameter

The default values are used for all of the installation parameters that are not listed on the command line:

• **Windows**: To silently run the installation wizard and use the command line to specify an installation parameter, use the following command as an example (for readability, the command is displayed on two lines; always use a single line):

```
setupwin32.exe -silent -W serverConfig.profileName="default"
-G licenseAccepted="true"
```

• Linux: To silently run the installation wizard and use the command line to specify an installation parameter, use the following command as an example (for readability, the command is displayed on two lines; always use a single line):

setupLinux.bin -silent -W serverConfig.profileName="default"
-G licenseAccepted="true"

Use the message log file (msg_server.log) and trace file (trace_server.log), as needed, for installation details and error handling.

After a successful installation of a distributed server component, the only required configuration is to set up the appropriate user definitions for authentication and authorization. By default, after installation, no users are authorized to connect to IBM Tivoli License Compliance Manager for z/OS distributed server components.

5.1.6 Installing distributed client components

You can use the installation wizard to interactively provide values for the installation parameters or to create a response file to provide the installation parameters and then perform the installation silently.

Use one of the following installation modes:

- Interactively installing the distributed client components
- Silently installing the distributed client components

There are several advantages in creating a response file for your installation:

- Installing the same components on more than one computer and using the same parameter values for each installation
- Running the installation in less time than it takes to use the interactive installation wizard
- Using the interactive installation wizard with preloaded parameter values
- Removing the product from a remote computer over the network

Typically, you install the distributed server components before the distributed client components.

Note: There are no prerequisite products required before you begin the installation of the distributed client components.

Specifying host name values during either an interactive or a silent installation

The preferred value for a host name is a host name alias, which allows the application to use a host name that is defined on the DNS server. The host name will not change even if the Tivoli Asset Compliance Center server is moved to a different IP address or host.

If you specify an IP address, and the IP address changes, the computer will not be able to contact to the Tivoli Asset Compliance Center server. For example, the IP address can change in a Dynamic Host Configuration Protocol (DHCP) environment where dynamic IP addresses are assigned and a computer can acquire a new address periodically or after restarting the computer.

If you specify a short host name, computer systems outside of the Tivoli Asset Compliance Center server's subnet must be configured to resolve addresses in the Tivoli Asset Compliance Center server subnet. If the computers are configured incorrectly, the client application on those computers cannot contact the Tivoli Asset Compliance Center server. Therefore, we recommend that you use a short host name only if your environment contains a single IP domain, or if all of the computer systems are configured with the correct domain search information.

Interactively installing the distributed client components

This topic describes how to install the IBM Tivoli License Compliance Manager for z/OS distributed client components using the graphical user interface. You can also silently install the distributed client components using a response file to supply the installation parameters.

During the installation, you supply the host name, port number, and context name for WebSphere Application Server to perform the connectivity test between this computer and WebSphere Application Server. You have the option of omitting the host name, and then the connectivity test is not performed.

During the installation, you select the distributed client components that you want to install at this time. At a later time, you can run the installation wizard again to install additional distributed client or server components on this computer.

As an option, you can prepare a response file and then start the interactive installation with the values supplied from the response file.

To install the distributed client components, follow these steps:

1. Locate and start the installation setup file on the product distribution media or other source for the distributed client installation image. Table 5-15 shows the installation setup file names by operating system.

| Operating system | Installation file name |
|------------------|------------------------|
| Windows | setupwin32.exe |
| Linux | setupLinux.bin |

Table 5-15 Installation setup file name by operating system

- 2. Select a language from the drop-down list. The language choice determines the locale in which the installation wizard is run. If the selected language locale is not present in the computer, the language of the computer is used.
- 3. Read the Welcome window.
- 4. Read the Software License Agreement. You must accept the terms in the license agreement by clicking the corresponding radio button.
- 5. Specify the installation directory. Use one of the following techniques:
 - Use the default directory.
 - Type a directory name in the entry field.
 - Click **Browse** to select a directory.

Specify a valid directory name. If the directory does not exist, the directory will be created. If the directory path contains spaces, enclose the path in double quotation marks.

If any of the distributed client components are already installed on this computer, the directory selection window is not displayed.

6. Select the distributed client components that you want to install now. The components listed in Table 5-16 are available.

Table 5-16 Distributed client components

| Distributed component | Description |
|---|---|
| Tivoli Asset Compliance Center (Asset Compliance perspective) | Main user interface for the client. |
| Administration Console (Administration perspective of the Tivoli Asset Compliance Center) | Console to implement security, administer users, and configure the application. |

The installation wizard determines if any distributed client components are already installed. Any previously installed distributed client components are displayed with a check mark and are not available for selection.

The Import Command Line Interface (import) and the Data Aggregation commands (daCLI command shell) are installed when the Tivoli Asset Compliance Center component is installed. These commands import and export data from the asset management products and other sources.

By default, only the Tivoli Asset Compliance Center (Asset Compliance perspective) is selected.

7. Specify the WebSphere Application Server host name, port number, and context name. Table 5-17 on page 191 describes these parameters.

Table 5-17 Installation parameters

| Installation parameter | Description |
|--|---|
| WebSphere Application Server host name | Use a fully qualified host name, host name alias, short host name, or the IP address to identify the host name. Although not recommended, you can use a network alias to identify the host name. |
| Port number | Use a valid port number for communication with WebSphere Application Server. The value for the port number has to be numeric and the port must not be in use. The default value is 9080. |
| Context name | Use the root context name for WebSphere Application Server. The default context name is CCMServlets. There is no validation for the context name. |

A connectivity test is performed between the client computer and the WebSphere Application Server computer using the specified host name, port number, and context name. The result of the connectivity test is displayed in a message. If the connectivity test failed, clicking **OK** returns you to the previous window. Therefore, you can correct the connectivity problem or delete the host name.

If a host name is not specified (null), no connectivity test is performed and a warning message is displayed. After reading the "no host name" message, click **Return** to specify a host name or **Continue** for the next step of the installation.

- 8. Select the user access permission and the action, if any, after the installation is successful. Depending on the user access rights, the installation can be for all users or only the current user:
 - Current user

Select **Current user** to allow access to the distributed client components, which are being installed, to only the person logged on to this computer during this installation.

Assume that the following values were specified with this installation:

- WebSphere Application Server host name: was6025win.mycompany.com
- Port: 9080
- Context name: CCMServlets

For this information, the following Application Server URL string is built and displayed on the Tivoli Asset Compliance Center client logon:

http://was6025win.mycompany.com:9080/CCMServlets

All users

Select **All users** to allow access to the distributed client components, which are being installed, to all users of this computer.

The WebSphere Application Server host name, port, and context name are collected, but the Application Server URL is not built from these values, so the Application Server URL field is blank for the Tivoli Asset Compliance Center client logon.

- 9. Select the check box to view the readme file after a successful installation.
- 10. Select the check box to start the Tivoli Asset Compliance Center (Asset Compliance perspective) after a successful installation.
- 11. Review the installation specifications window, and click **Next** to begin the client installation. An installation progress window is displayed.
- 12.Read the installation summary.
- 13. The installation is complete. Click **Finish**.

If the installation fails, an error window is displayed. Use the message log file (msg_client.log) and the trace file (trace_client.log), as needed, for installation details and error handling.

After the installation of the client, the connection information is written to a client properties file; therefore, you do not have to enter that information again.

After a successful installation of the distributed client component, the only required configuration is pointing the client component to the appropriate IBM Tivoli License Compliance Manager for z/OS distributed server.

Silently installing the distributed client components

This topic describes how to silently install the IBM Tivoli License Compliance Manager for z/OS distributed client components using a response file to supply the installation parameters. You can also use a graphical user interface to install the distributed client components. When preparing the response file, you supply the following information:

 Identify the distributed client components that you want to install on this computer.

You can install one or more of the following components:

- Tivoli Asset Compliance Center (Asset Compliance perspective)
- Administration Console (Administration perspective of the Tivoli Asset Compliance Center)
- Installation directory for the distributed client components
- Host name, port number, and context name for WebSphere Application Server
- Determine the user access options
- Determine the startup options

At a later time, you can run the installation again to install additional distributed client or server components on this computer.

When preparing the client response file, use the following parameter descriptions that are shown in Table 5-18, the sample response file, and the client response file that is shipped with the product as a guide. Use a plain text editor, such as Notepad on Windows, when creating and editing your response file.

| Parameters | Description | Samples and tips |
|--------------------|--|---|
| -P installLocation | Identifies the installation directory. | C:\Program File\IBM\ITACC %USERPROFILE%\IBM\I TACC /opt/IBM/ITACC |
| | | If the directory path contains spaces, enclose the path in double quotation marks. |

Table 5-18 Parameter descriptions

| Parameters | Description | Samples and tips |
|---------------------------------------|--|---|
| -W ITAMinstallClient.mainui | Determines if the Tivoli Asset Compliance Center (Asset Compliance perspective) is to be installed. | true Any value other than true is considered to be a value of false. |
| | Valid values are true or false. Use true to install this component. The default value is true. When the Tivoli Asset Compliance Center (Asset Compliance perspective) is installed, the Import Command Line Interface (import) and the Data Aggregation commands (daCLI command shell) are also installed. These commands import and export data from the asset management products and other sources. | |
| -W ITAMinstallClient.adminto ol | Determines if the Administration Console (Administration perspective of the Tivoli Asset Compliance Center) is to be installed. Valid values are true or | true Any value other than true is considered to be a value of false. |
| | talse. Use true to install this component. The default value is false. | |
| -W ITAMinstallClient.wasHost | Identifies the WebSphere Application Server host name. There is no default value. | Use a fully qualified host name, host name alias, short host name, or the IP address to identify the host name. |
| | | Although not recommended, you can use a network alias to identify the host name. |

| Parameters | Description | Samples and tips |
|---|--|--|
| -W ITAMinstallClient.port | Identifies the port number. The default value is 9080. | Use a valid port number for communication with WebSphere Application Server. The value for the port number has to be numeric. |
| -W ITAMinstallClient.contextN ame | Identifies the context name. The default value for the context name is CCMServlets. | Use the root context name for WebSphere Application Server. There is no validation for the context name. |
| -W ITAMinstallClient.currentU ser | Determines the user access for the current user. When set to true, allow access to the components being installed to only the person logged on to this computer during this installation. The valid values are true or false. The default value is true. | false Any value other than true is considered to be a value of false. Assume the following was specified with this installation: WebSphere Application Server host name: was.company.com Port: 9080 Context name: CCMServlets When set to true, the following Application Server URL string is built and displayed on the Tivoli Asset Compliance Center client logon (for readability, the URL is split on three lines): http://was.company.com :9080 /CCMServlets |

| | Parameters | Description | Samples and tips |
|--|--|---|---|
| | -W ITAMinstallClient.allUser | Determines the user access for all users. When set to true, allow access to the components being installed to all users of this computer. The valid values are true or false. The default value is false. | true Any value other than true is considered to be a value of false. When set to true, the WebSphere Application Server host name, port, and context name are collected, but the Application Server URL is not built from these values, so Application Server URL field is blank for the Tivoli Asset Compliance Center client logon. |
| | -W ITAMinstallClient.showRe adme | Determines if the readme is displayed. When set to true, the readme file is displayed after a successful installation. The valid values are true or false. The default value is false. | false Any value other than true is considered to be a value of false. |
| | -W ITAMinstallClient.spawnUI | Determines if the main user interface is opened. When set to true, the Tivoli Asset Compliance Center (Asset Compliance perspective) is opened after a successful installation. The valid values are true or false. The default value is false. | false Any value other than true is considered to be a value of false. |
| Parameters | Description | Samples and tips |
|--------------------|---|--|
| -G licenseAccepted | Determines if the product license has been accepted. The value must be true before a silent installation will run. The valid values are true or false. The default value is false. | You must uncomment this parameter and change the value to "true" before a silent installation will be successful. -G licenseAccepted=true |
| | If the value is false, the silent installation will not run. | |

Note: If a parameter is not included in the response file, the default value, if any, is used. If the response file contains a parameter that does not apply, a warning message about that parameter is written to the message log file.

Response file considerations

When creating and using your response file, consider the following items:

- Parameter values are case sensitive. When a set of allowable values is specified, the values must use the indicated case.
- If the installation is started with a response file, the current locale remains in effect if the locale is present on the computer. Otherwise, the locale defaults to the language of the computer.
- The host name, port number, and context name for WebSphere Application Server are used to perform the connectivity test between this computer and WebSphere Application Server. The result of the connectivity test is written to the message file.

If you omit the host name from the response file, the connectivity test is not performed. If you omit the port number, the root context name, or both the port number and the root context name from the response file, the default values are used for these parameters.

Steps to silently install the distributed client components

Use the parameter descriptions, sample values, and sample response file as a guide.

Example 5-5 on page 198 shows a sample response file for a client installation.

Example 5-5 Sample response file for a client installation

| -P | <pre>installLocation="C:\Program Files\IBM\ITACC"</pre> |
|----|---|
| -W | ITAMinstallClient.mainui="true" |
| -W | ITAMinstallClient.admintool="false" |
| -W | ITAMinstallClient.wasHost="localhost" |
| -W | ITAMinstallClient.port="9080" |
| -W | ITAMinstallClient.contextName="CCMServlets" |
| -W | ITAMinstallClient.currentUser="false" |
| -W | ITAMinstallClient.allUser="true" |
| -W | ITAMinstallClient.showReadme="false" |
| -W | ITAMinstallClient.spawnUI="false" |
| -G | licenseAccepted="true" |

If the value contains a space character, enclose the value in double quotation marks ("my value"). Double quotation marks are allowed for all values.

If you include an installation parameter in the response file, you must provide a value for that parameter.

Select either a silent installation or an interactive installation

To run the installation wizard with the parameters from your response file, issue one of the following commands. The response file location is relative to the current working directory. Locate the installation setup file on the product distribution media or other source for the client installation image. The options are:

- Silently running the installation wizard with a response file:
 - Windows: To silently run the installation wizard with the response file, use the following command:

```
setupwin32.exe -options response_file_name -silent
```

 Linux: To silently run the installation wizard with the response file, use the following command:

```
setupLinux.bin -options response file name -silent
```

- Interactively running the installation wizard with a response file:
 - Windows: To interactively run the installation wizard using values from the response file, use the following command:

setupwin32.exe -options response_file_name

 Linux: To interactively run the installation wizard using values from the response file, use the following command:

setupLinux.bin -options response_file_name

• Silently running the installation wizard without a response file

The default values are used for all the installation parameters. No connectivity test is performed. The value for licenseAccepted must be set to "true" before a silent installation will be successful, so include that parameter on the command line. The commands by operating system are:

Windows: To silently run the installation wizard *without* the response file, use the following command:

setupwin32.exe -silent -G licenseAccepted="true"

 Linux: To silently run the installation wizard *without* the response file, use the following command:

setupLinux.bin -silent -G licenseAccepted="true"

 Silently running the installation wizard and using the command line to specify an installation parameter

The default values are used for all of the installation parameters that are not listed on the command line. If the WebSphere Application Server host name is not valid, no connectivity test is performed. The commands by operating system are:

 Windows: To silently run the installation wizard and use the command line to specify an installation parameter, use the following command as an example (for readability, the command is displayed on two lines; always use a single line):

setupwin32.exe -silent -W ITAMinstallClient.wasHost="localhost"
-G licenseAccepted="true"

 Linux: To silently run the installation wizard and use the command line to specify an installation parameter, use the following command as an example (for readability, the command is displayed on two lines; always use a single line):

setupLinux.bin -silent -W ITAMinstallClient.wasHost="localhost"
-G licenseAccepted="true"

Use the message log file (msg_client.log) and trace file (trace_client.log), as needed, for installation details and error handling.

After the successful installation of the distributed client component, the connection information is written to a client properties file; therefore, the user does not have to enter that information again.

After a successful installation of the distributed client components, the only required configuration activity is to point the client component to the appropriate IBM Tivoli License Compliance Manager for z/OS distributed server.

Error handling

This topic describes how to handle errors when installing the distributed server and client components of IBM Tivoli License Compliance Manager for z/OS.

Error messages

The first five characters of the error identifier represent the product code. For IBM Tivoli License Compliance Manager for z/OS, the product code is IXUIN. The next four characters of the error identifier are numbers in the range from 3250 to 3500 for distributed client components and from 4000 to 4250 for distributed server components. The final character of the error identifier is either an "E" for error, "W" for warning, or "I" for information.

Exceptions, errors, and warnings are logged for all installations, but only displayed during interactive installations.

Failure to connect to the database and WebSphere Application Server messages are shown as informational messages.

Error flow during an installation

Errors, such as loss of power, disk I/O errors, loss of network connection, and lack of disk space, can cause failure during the installation phases. For example:

- If an error occurs when values for the parameters are being gathered, no recovery occurs because no distributed components were installed. You need to correct the error and start the installation again.
- If an error occurs during the installation of the files phase, no recovery occurs because the installation program only commits all of the changes at the end of a successful installation. You need to correct the error and start the installation again.
- If an error occurs during the post-installation phase, the distributed component must be removed. You need to correct the error and start the installation again.

However, a failure in the connectivity test for the distributed client is not considered a failed installation and does not require the removal of the distributed component and the installation of the component again.

Installation log file directories

All log files that are specific to the distributed server and client components of IBM Tivoli License Compliance Manager for z/OS are located in the following directories:

- ► Windows: *Tivoli_common_dir* is C:\Program Files\IBM\tivoli\common
- ► Linux: *Tivoli_common_dir* is opt/IBM/tivoli/common

Table 5-19 shows the directories for the Windows operating system.

| Log | Directories for Windows operating system |
|--------------------------|--|
| Installation message log | <i>Tivoli_common_dir</i> \IXU\logs\client\install\messa ge\msg_client.log <i>Tivoli_common_dir</i> \IXU\logs\server\install\mess age\msg_server.log |
| Removal message log | <i>Tivoli_common_dir</i> \IXU\logs\client\install\messa ge\msg_client_uninstall.log <i>Tivoli_common_dir</i> \IXU\logs\server\install\mess age\msg_server_uninstall.log |
| Installation trace log | <i>Tivoli_common_dir</i> \IXU\logs\client\install\trace\t race_client.log <i>Tivoli_common_dir</i> \IXU\logs\server\install\trace\ trace_server.log |
| Removal trace log | <i>Tivoli_common_dir</i> \IXU\logs\client\install\trace\t race_client_uninstall.log <i>Tivoli_common_dir</i> \IXU\logs\server\install\trace\ trace_server_uninstall.log |

Table 5-19Directories for the Windows operating system

Table 5-20 shows the directories for the Linux operating system.

Table 5-20 Directories for the Linux operating system

| | Log | Directories for Linux operating system |
|--|--------------------------|--|
| | Installation message log | <i>Tivoli_common_dir/</i> IXU/logs/client/install/ message/msg_client.log <i>Tivoli_common_dir/</i> IXU/logs/server/instal l/message/msg_server.log |
| | Removal message log | <i>Tivoli_common_dir</i> /IXU/logs/client/install/ message/msg_client_uninstall.log <i>Tivoli_common_dir</i> /IXU/logs/server/instal l/message/msg_server_uninstall.log |
| | Installation trace log | <i>Tivoli_common_dir/</i> IXU/logs/client/install/ trace/trace_client.log <i>Tivoli_common_dir/</i> IXU/logs/server/instal l/trace/trace_server.log |
| | Removal trace log | <i>Tivoli_common_dir/</i> IXU/logs/client/install/ trace/trace_client_uninstall.log <i>Tivoli_common_dir/</i> IXU/logs/server/instal l/trace/trace_server_uninstall.log |

To help recover from installation problems, the installation wizard uses various utilities to build a post-installation flow. During an installation, you can monitor the previous installation steps, pause the installation, and resume the installation later. When an installation problem occurs, a message will help you understand the failure, repair the failure, and recover the installation from the failure point.

When repairing an existing installation, the files will be installed again with the original version, and values in the configuration files will be changed to the original values. The corrupted or modified files will be backed up at the beginning of the repair installation.

5.1.7 Creating a response file or a template response file

You can interactively create a response file or a template response file to provide values for the installation wizard or to perform the installation silently. A *template response file* is similar to a response file with comments that have been added for each parameter.

Advantages for creating a response file or template response file include:

- Installing the same component on more than one computer and using the same parameter values for each installation
- Running the installation in less time than it takes to use the interactive installation wizard
- Using the interactive installation wizard with preloaded parameter values
- Removing the product from a remote computer over the network

Interactively creating a response file for distributed client or server components

This topic describes how to create a response file for distributed client or server components of IBM Tivoli License Compliance Manager for z/OS using the installation wizard.

As an alternative, you can create a template response file. It is similar to a response file with comments added for each parameter in the file.

The following steps assume that you have WebSphere Application Server, DB2, and DB2 Alphablox installed.

To interactively create a response file for distributed client or server components:

 Locate the installation setup file for the distributed client or server component on the product distribution media. Table 5-21 on page 203 shows the installation setup file names by operating system.

Table 5-21 Installation setup file names

| Operating system | Setup file name |
|------------------|-----------------|
| Windows | setupwin32.exe |
| Linux | setupLinux.bin |

- To create a response file for the distributed client or server component by using the installation wizard, run the following command:
 - Windows:

setupwin32.exe -options-record response file name

– Linux:

setupLinux.bin -options-record response_file_name

After the response file is created, you can customize the response file before starting the silent or interactive installation. Edit the response file with a flat text editor.

5.2 Collecting raw data

The two types of data that the product provides for Software Asset Management are software product data and software use data. The relationship between the software product data and the software use data is binary. That is, each type of data is of little or no value without the other type of data. Having an inventory of software products without the use data for those products is of limited value to a Software Asset Manager. It is illogical to have use data without matching it to a software product.

5.2.1 Taking an inventory of software data

First, we discuss collecting software inventory data. In the z/OS environment, the raw software inventory data is stored in load libraries. The z/OS UNIX data is stored in UNIX directories. Additional inventory data for z/OS and z/OS UNIX software is stored in SMP/E for z/OS (SMP/E) consolidated software inventory (CSI) data sets. Figure 5-5 on page 204 shows the data surveyor flow, which shows where data is stored.



Figure 5-5 Surveyor flow

z/OS load libraries

Generally speaking, each load library on a given z/OS logical partition (LPAR) is defined with a unique load library name. Each load library contains load modules that provide the functions of a given software product or products. A load library can contain the set of load modules for one software product or sets of load modules for many software products.

The *Surveyor* component (shown in Figure 5-5) is used to gather data by scanning and storing data from the z/OS load libraries. The Surveyor collects the information that is required to create a software product inventory. The data that is collected from the z/OS load libraries will be used in a downstream component of the product.

z/OS UNIX directories

The Surveyor is also used to collect data from the z/OS UNIX directories. In the z/OS UNIX environment, the raw software inventory data is stored in a directory structure. A discussion of the technical differences between z/OS load libraries and z/OS UNIX directories is beyond the scope of this book. The resultant data that the Surveyor collects from z/OS UNIX directories will be used in a downstream component of Tivoli License Compliance Manager for z/OS.

SMP/E

Another form of software inventory data for z/OS and z/OS UNIX products is stored in SMP/E data sets. The software inventory data that is stored in SMP/E data sets includes product names, product versions, and load module names that are attributable to a given product.

Figure illustrates the Extractor flow of data.



Figure 5-6 Extractor flow

The *Extractor* component is used to gather data from the SMP/E consolidated software inventory (CSI) data sets, by extracting and storing data from the z/OS load libraries. The Extractor collects the information that is required to create a software product inventory. The data that is collected from the SMP/E CSIs will be used in a downstream component of the product.

5.3 Creating a software product inventory

After you have collected raw software inventory data for z/OS and z/OS UNIX using the Surveyor and Extractor components, you are ready to generate an inventory of software products.

5.3.1 Identifying products

The *Identifier* component (which is shown in Figure 5-7) is used to generate a software product inventory. The Identifier uses three sets of data as input to generate a software product inventory:

- The raw software data collected by the Surveyor
- The CSI data collected by the Extractor
- The Software Catalog



Figure 5-7 Identifier flow

Software Catalog

The *Software Catalog* (called the Knowledge Base in the previous version of the IBM Tivoli License Compliance Manager for z/OS) is a proprietary database

containing information that is used to associate objects discovered by the Surveyor with software products. The contents of the Software Catalog include:

- Vendor information (Vendor Name, Vendor ID, and so on)
- Product information (Product Name, Product ID, and Vendor ID)
- FMID to Product ID mapping table (for use with data extracted from SMP/E)

The IBM Software Catalog is constantly updated with new and changed products and vendors and is released quarterly to clients.

The Identifier uses a set of algorithms to compare the raw software data to the contents of the Software Catalog to produce a software product inventory. This raw software data is used to generate a file of installed software products that includes the vendor name, the fully qualified product name, and the version of the product.

The Identifier produces a data set and a report, which contains all of the installed software products. The data set contains the names of the z/OS load libraries and z/OS UNIX directories where the product was discovered.

New editions of the Software Catalog are made available four times a year. The product definitions that are contained in the Software Catalog represent the profiles of software products at a specific point in time. That is, various elements contained in the Software Catalog will become superseded or replaced, because software products are not static. For example, product names change; new versions of products become available; vendors acquire and divest software products; and new load module names are attributed to a software product.

There are four optional data sets that you can include in an Identifier run to further adjust, enhance, and influence product identification to achieve your goals:

- User product data (USRPDATA)
- Software customization tool (SCCUT)
- Reference identification file (RIF)
- Signature Software Catalog

User product data (USRPDATA)

Because new editions of the Software Catalog are made available approximately every calendar quarter, there might be instances when you need to adjust various elements that define the individual software products that are not yet contained in your edition of Software Catalog. You might want to modify various elements of the software products contained in the Software Catalog to conform to your local tracking and licensing of software products. The Software Catalog contains definitions of commercially available software products. You might need to use the License Compliance Manager for z/OS to track internally developed applications and utilities.

The user product data (USRPDATA) statements provide you with the mechanism to make adjustments to product identification. The USRPDATA statements enable you to adjust and influence software product identification to fit your enterprise requirements. Several of the adjustments that you can make to product identification include:

- Define new products not contained in the Software Catalog
- Change the external name of software products
- Attribute a software product to a different vendor
- Define locally developed applications and utilities

To use the facility to influence product identification, you include the appropriate USRPDATA statements that you develop in the particular run of the Identifier. There are several considerations when using USRPDATA statements to influence product identification:

- The USRPDATA statements do not alter the contents of the Software Catalog.
- The changes that you make using USRPDATA statements are contained only in the output from the Identifier: the Identifier data set and the Identifier report.
- You run as many iterations of the Identifier using the USRPDATA statements as necessary to achieve your product identification goal.

Software Customization Tool (SCCUT)

Another option that is available to you to adjust and influence software product identification is the *Software Customization Tool* (SCCUT). SCCUT enables you to change product identification in a similar manner to USRPDATA with one major difference - SCCUT applies your changes to the Software Catalog. The input statements to SCCUT are similar in format and content to USRPDATA statements. The SCCUT component provides you with a safeguard by generating a copy of the Software Catalog. The end result of an SCCUT run is a Software Catalog that contains your changes defined by the SCCUT statements, a copy of the Software Catalog without your changes, and a report detailing your changes. The theory for and reasons why to use USRPDATA and SCCUT are the same; the difference is that SCCUT applies permanent changes to a copy of the Software Catalog.

Reference identification file (RIF)

Many enterprises choose to centrally install software on a base system and then roll out or clone the base system to deploy the software on subordinate systems. This process results in effectively replicating the same set of base system software on all of the subordinate systems. When the Identifier has identified a set of modules in one library as belonging to a particular product-version-release, that identification is available for reference in the identification of the same set of modules in other libraries. Use of the reference identification file (RIF) helps ensure consistent software product identification across the enterprise.

Signature Software Catalog

The Software Catalog contains several releases and versions of products with their modules and module signatures. The Software Catalog is used to perform version identification.

5.4 Capturing software use data

Capturing and reporting the use of software is critical to the discipline of Software Asset Management. The collection of software use data must be comprehensive. It must include the use of products in the batch environment, the interactive or Time Sharing Option (TSO) environment, and the many system tasks that are used to manage and control subsystems. The collection of software use data must also be vendor neutral. That is, software use data must be captured regardless of the vendor of the software.

Figure 5-8 on page 210 shows the Monitor flow.



Figure 5-8 Monitor flow

The *Monitor* collects software use activity data within the z/OS logical partition in which it is running. The Monitor must run as a started task in each LPAR where use activity must be captured. The Monitor captures information about each of the three types of z/OS work:

- Batch jobs (JOB)
- TSO sessions (TSO)
- Started tasks (STC)

Software use recording of z/OS work is activated when the following z/OS supervisor calls (SVCs) are detected within the LPAR where the Monitor is running:

- Load
- Link
- Attach
- Transfer control

The Monitor captures a comprehensive set of data points including the name of the load library that was used by the work unit - JOB, TSO, or STC.

The Monitor data is recorded in a log data set, and the management of the Monitor logs is similar to the techniques that are used to manage system management facility (SMF) data.

Periodically, the use data that is collected by the Monitor is moved into a Monitor log accumulation data set. This data set represents a continuous log of use data, and it can reside on a device type that is appropriate to your environment - DASD, tape, or virtual tape.

The Monitor simply records software use data as z/OS work is initiated and processed in the LPAR. The task of identifying software products adds significant overhead to the Monitor. Identifying software products and building the software product inventory is done by the Identifier component. The accumulated use data set does not contain product or vendor specific information. You will combine the accumulated use data with the software product inventory that you generated using the Identifier.

5.5 Generating reports

The following sections describe the components and the process for generating reports that are based on the use data and the software product inventory data.

5.5.1 Generating data for the distributed component of the License Compliance for z/OS

We have generated software product inventory, and we have collected software use data. We now need to combine the software product inventory data with the software use activity data. At this point, we have two files of unrelated data:

- The software product inventory that is produced by the Identifier
- The software use activity data that is collected by the Monitor

We use the *Distiller*® to combine these two files into a software use file. The Distiller can combine the two files using the z/OS load library name. The Distiller uses the z/OS load library name, which is contained in the software product inventory data and the use activity data, as the linkage. The output of the Distiller is a file in extended markup language (XML) format. We now have a file that contains the use information for specific products that are contained in the software product inventory file.

Figure 5-9 on page 212 shows the Distiller flow.



Figure 5-9 Distiller flow

We will then import the XML output of the Distiller into the Tivoli Asset Compliance Center.

5.5.2 Mainframe server-based reports

You use the *Reporter* component of the product to produce mainframe-based reports about software product inventory and software use.

Figure 5-10 on page 213 shows the Reporter flow.



Figure 5-10 Reporter flow

The Reporter uses the software product inventory data that is produced by the Identifier and the software use data that is collected by the Monitor to produce reports. The Reporter provides a number of reports about software product inventory only and software produce use. All of the available reports are parameter-driven.

5.5.3 Exporting data to other tools and programs

The *Exporter* is used to export the software product inventory and software use data.

Figure 5-11 on page 214 illustrates the Exporter flow.



Figure 5-11 Exporter flow

The Exporter uses the software product inventory data that is produced by the Identifier and the software use data that is collected by the Monitor to produce sequential data sets that you can use with analysis tools and reporting programs. You can use the Exporter to produce sequential data sets or data sets in system management facility (SMF) format. You can process the sequential data sets that are produced by the Exporter within the mainframe environment or convert them to files that are suitable for processing in your distributed environment.

5.5.4 Managing software use data from the Monitor

You use the *Selector* to manage use data after it is collected by the Monitor. The volume of the use data can quickly accumulate, because the Monitor sees every unit of work that is executed in each LPAR.



Figure 5-12 Selector flow

For example, an enterprise might have the Monitor running in 30 LPARs, for 24 hours, seven days a week. Many of those LPARs have significant batch loads, while other LPARs have a large number of TSO user sessions that start and stop during the Monitor collection period. The Selector is used in this example to reduce the volume of software use data by filtering out unneeded or unwanted software use records.

You also use the Selector to filter out unwanted software data from the Distiller process so that extraneous data does not appear in the distributed component of the IBM Tivoli License Compliance Manager for z/OS.

5.6 Sub-Capacity Reporting Tool

The *Sub-Capacity Reporting Tool* (SCRT) is a no-charge IBM tool that reports the required license capacity for sub-capacity eligible products that run on z/OS, z/OS.e, or z/TPF. The Sub-Capacity Reporting Tool is not required to use the License Compliance Manager for z/OS product. However, if you are using the

Sub-Capacity Reporting Tool, you can import the data that it generates into the distributed component of the License Compliance Manager for z/OS product.

5.7 Tivoli Asset Compliance Center

Figure 5-13 illustrates the Tivoli Asset Compliance Center flow.



Figure 5-13 Tivoli Asset Compliance Center flow

The Data Aggregation server accepts the Distiller files and the data generated by the Sub-Capacity Reporting Tool (SCRT) that contain software usage information for applications licensed under sub-capacity-based pricing terms. This data, too, is combined with license information and is made part of what is presented to the client.

After the data (whether Distiller XML data or SCRT data) is available on the Data Aggregation server machine, it must be imported into the Data Aggregation server itself.

The Data Aggregation server will process the imported data files on an ongoing basis. Periodically, processed data must be extracted from the Data Aggregation server and made available to the Asset Management server.

For most data, the flow is from the Distiller to the Data Aggregation server and from the Data Aggregation server to the Asset Management server. However, before the Data Aggregation server performs its periodic compliance processing, it first connects to the Asset Management server to obtain the current set of license information together with certain topology information (installed machines, for example). This data transfer is automatic after the communication parameters have been configured (which is usually done as part of the server installation processes).

5.7.1 Administering and using the Tivoli Asset Compliance Center

The Administration perspective provides the administrator with the functions to define and manage:

- Users
- ► Groups
- Roles
- Domains
- Locations

Figure 5-14 on page 218 shows the Administration perspective.



Figure 5-14 Administration perspective

The Administration perspective is used to import:

- Distiller files
- SCRT files

5.7.2 Managing software assets with Tivoli Asset Compliance Center

The Asset Compliance perspective provides the Software Asset Manager administrator with the functions to:

- Create and manage business objects
- Work with activities
- Work with software licenses
- Search for and filter data
- Run reports
- Export information

5.8 Summary

The Tivoli License Compliance Manager for z/OS product is composed of the mainframe components and the distributed components.

The mainframe components of the product provide:

- The Surveyor and Extractor programs to take the raw physical data that represents your installed software
- The Identifier to produce an inventory of installed software products using the raw physical data
- The Monitor to collect software use data in the LPARs in which you are interested
- The Distiller component that enables you to generate files that can be imported into the distributed component of the product
- The Reporter, Exporter, and the Selector components produce mainframe reports, export data, and manage software use data

The distributed components of the product provide the graphical user interface (GUI) view of your z/OS software product inventory, which allows you to:

- Import data generated by Distiller component.
- Optionally import data generated by SCRT.
- Manage license compliance for those products.
- Produce reports and export information.
- Use these activities to perform the other tasks of Software Asset Management.



6

Deployment considerations

This chapter describes various parameters that you need to consider in planning an implementation starting with your technical options in terms of database, application server, and operating system to required skills. After reading this chapter, the implementation team can create a high level project plan with effort and time estimates for a typical implementation. We have not created an implementation plan to fit every implementation, because every implementation differs. Each client has a unique set of requirements. Consider the the following variables when planning an implementation:

- Infrastructure selected:
 - Operating system: Windows, Linux, or AIX®
 - Database: Oracle, DB2, or SQL
 - Application Server: WebSphere or WebLogic
- Integration with other third-party systems
- Existing data sets for data migration
- Process maturity

The combinations that these variables generate are immense and can greatly impact the project schedule. Carefully examine these variables during the requirements gathering phase. Several of these variables are too subjective and can often create political challenges within the client organization and, therefore, must be handled with great care.

6.1 Planning for a client engagement

Next, we summarize key considerations for planning an engagement.

6.1.1 Service engagement preparation

Obviously, the first step in preparing for a potential service engagement is to fully understand the requirements of the client. Then, map these requirements to the capabilities across the Tivoli portfolio and whether Tivoli has all of the required capabilities to meet the client's expectations.

6.1.2 Implementation skills

Developing and deploying a Tivoli Asset Management for IT V7.1 solution successfully requires the following specialized skills:

- General skills: Operating system skills and an understanding of client/server communication concepts
- Skills in relevant middleware, such as DB2, WebSphere, Lightweight Directory Access Protocol (LDAP), and any other middleware that might be included
- An understanding of Tivoli Asset Management for IT, Tivoli License Compliance Manager, and Tivoli License Compliance Manager for z/OS

6.1.3 Available resources

The prerequisite skills that we listed are needed to customize or develop the solution. For each of these skills, there are variety of resources available to help acquire the necessary skill levels. The available resources are:

- Online help and product documentation are available.
- Class room training. The IBM Partnerworld Web site provides current information about available classes, their dates, locations, and registrations.
- IBM Technical Education Services offers a variety of classes at all knowledge levels to help you achieve any of the offering's prerequisite skills.
- IBM Redbooks publications provide access to various practical and architectural information regarding IBM hardware and software platforms.

6.1.4 Solution scope and components

The scope of the solution can be either:

- Basic offerings
- A combination of cross-sell and up-sell opportunities

The Tivoli Asset Management for IT portfolio products can be integrated with each another, other Tivoli products, and other enterprise applications to offer a complete management solution.

6.1.5 Services engagement overview

A services engagement typically consists of:

- Building an executive assessment
- Setting up a demonstration system or proof of technology
- Analyzing solution tasks
- Creating a contract commonly known as a *statement of work*:
 - This contract helps everyone to understand the implementation details, size the solution more accurately for the clients, and ensure a profitable engagement.
 - It is important to work with clients to understand their expectations. The statement of work acts as a contractual agreement with the client for the duration of the project.
 - A good overall understanding of the solution scope is a crucial prerequisite to successfully selling, developing, and implementing it.

The executive assessment is a billable service that you can offer to your prospective clients. The benefits of using the executive assessment in the sales process include:

- More effectively qualifying prospective clients
- Shortening the sales cycle
- Streamlining the development process

The executive assessment is a business case assessment, not a technical assessment. Therefore, your audience needs to be business owners, line-of-business executives, marketing and sales managers, and, finally, the IT manager.

Over the course of the executive assessment, you determine who will be involved in the project, what they want to accomplish, when they plan to deploy, who plays a mission critical role in their business, and how the project will be funded. A *Demonstration system* is typically set up in advance to show your clients the attributes of the solution. The demonstration system allows your clients to evaluate whether the solution suits their particular need.

The Analyze solution tasks require:

- Working knowledge of the operating system
- ► Good understanding of client/server communication concepts
- Understanding of security concepts, such as secure socket layer (SSL), LDAP authentication, password management, and certificate usage
- Working knowledge of all Tivoli modules
- Understanding of other Tivoli operational management products and other enterprise applications, which will require integration with Tivoli Asset Management for IT

6.1.6 Creating a contract (statement of work)

A statement of work typically includes:

- An executive summary of the solution, which is typically a short summary of the solution and its benefits. You must specify any major restrictions of the implementation.
- A solution description includes the major components and solution building blocks that will be implemented.
- All assumptions must be documented. Any deviation to the assumptions will change the scope and cost of the project.
- ► The client responsibilities must be clearly defined.
- Staffing estimates list the estimated personnel that must implement the solution.
- The project schedule and milestones show the major steps, schedule, and achievements calendar that can be used to check the project progress.
- A testing methodology lists the test cases to ensure that the project implementation is successful.
- A deliverable provides tangible items that the client will receive at the completion of the service, including:
 - Machine installations
 - Documentation
 - Training
- The completion criteria lists the items provided to the client that indicate that the engagement is successfully completed.

6.2 Required skills

Fundamental to the success of a project is the team that delivers it. The exact number of people required varies with the project, organization's experience, reusable components, and methodology. Nevertheless, the people comprising a Tivoli Asset Management for IT deployment team typically include the following roles.

Project manager

The project manager performs the following activities:

- Own the implementation end-to-end.
- Be the single point of contact for the client.
- Facilitate client interactions.
- Report project status.
- Monitor project progress on regular intervals.
- Escalate issues to the correct people.
- Change control agent for scope creep.
- Make available resources, staff and material, for the project.

Business analyst

The business analyst is a functional person who understands the product and also the client's domain. This person can speak with the client in their own vernacular and has the ability to translate the client's requirements to the technical team in implementing a Tivoli IT Asset Management solution. This role is an extremely key role and is vital to the success of the project. Attributes of this role include:

- Domain knowledge
- Functional knowledge of Tivoli Asset Management for IT, Tivoli License Compliance Manager, and Tivoli License Manager for z/OS
- Working knowledge of Uniform Modelling Language (UML)
- Knowledge of industry best practices

Solution architect

The solution architect is the person who designs the complete solution. This role includes the integration with various third-party systems. The various attributes and tasks of this role are:

- Strong technical skills on Maximo base services and the Tivoli Asset Management for IT portfolio
- The ability to map business processes to technology

- Understanding of functional integration points
- Understanding of technical integration points
- Builds the overall solution design
- Educates the team about how the various aspects of implementation will be achieved
- Resolves technical issues

Tivoli Asset Management for IT developer

This group of people actually install and configure the software to work in accordance with the client's needs. The major tasks that the developers perform are:

- Perform base services and install Tivoli Asset Management for IT portfolio
- Configure panels, workflows, and information flow
- Integrate the products
- Develop custom classes, if required
- Optimize the system

Testers

Testing is critical to every project: system integration testing, functional testing, and performance testing. Depending on the complexity of the project, the developer might perform this role, or a specialized group might perform the testing. The major tasks include:

- Functional testing: Testing the workflows, communication templates, data flow between panels, and functions for accuracy
- System integration: Testing the integration to ensure that the data between systems flows accurately
- Performance testing: Testing to ensure that response times are acceptable and that the system responds optimally

Database administrators (DBAs)

As with any enterprise system, the underlying database and its performance are critical. The DBA performs the following activities:

- Troubleshoots the database creation
- ► Installs the database
- Optimizes the database
- Facilitates database-level integration with enterprise systems

Application server specialist

Depending on the choice of application server, WebSphere or WebLogic, part-time involvement of an application server specialist or administrator might be required. This role becomes more important if the implementation is complex with many integration points and is in a clustered environment.

Integration specialist

This is an optional group of people, and the integration team can vary from client to client. Special systems, such as SAP®, Oracle, and other niche products might require technical experts. Depending on the project, the major task of this group is to facilitate integration with external systems and ensure the bidirectional movement of data.

These various roles are not definitive for every project. There might be more roles that are required and more than one person in a role. Each project differs, and it is hard to generalize. However, we have tried to describe most of the scenarios that we have encountered in our implementation experiences.

6.3 Solution description and assumptions

This section describes a high-level solution architecture and the standard assumptions that are made while planning a project.

Figure 6-1 on page 228 is a standard functional solution layout that specifically shows a Tivoli Asset Management for IT solution.



Figure 6-1 Simplified Tivoli Asset Management for IT solution layout

You can access Tivoli Asset Management for IT, which is an application server-based application with a Web browser-based user interface, remotely using an Internet connection if the application is hosted on a public Internet Protocol (IP).

The three layers can be complicated depending on a client's requirements and network setup. You might implement all or part of the modules that are provided by the base services. The integration layer separates the enterprise applications from the core Maximo engine.

Standard assumptions while planning a project are:

- The software is procured prior to project initiation.
- There are three environments: development, testing, and production.
- The application server, database server, and reporting engine are all kept in separate physical servers.
- The client is responsible for any configuration that is required in the enterprise systems to facilitate integration.

- The client's involvement during the initial project stages is high, because a significant amount of the functional and business process knowledge that is specific to the client needs to transferred to the implementation team.
- ► There is a management approval or buy-in for this project.
- A dedicated single point of contact and project manager is appointed from the client to manage interactions with the users.
- ► The implementation team can migrate the historical data, but the client must perform the data cleansing and data scrubbing.

6.4 Task breakdown

The standard phases of an Asset Management for IT implementation project are:

- Phase 1: Requirement gathering and analysis
- Phase 2: Solution design
- Phase 3: Deployment
- Phase 4: Testing and going live
- Phase 5: Support

We describe the key activities in each of the phases in the following sections.

Phase 1: Requirement gathering and analysis

This phase involves the project manager, business analyst, and solution architect.

The key activities are:

- Understand the current business processes for:
 - Maintenance management
 - Procurement
 - Inventory management
 - Asset management
- Document the painful areas with the existing business processes
- Demonstrate the base product
- Understand and document the end goal and the client's vision
- Document "To-Be" business processes for each of the functions mentioned
- Analyze the points of integration
- Analyze the current data sources
- ► Create a Software Requirements Specification (SRS) document

Phase 2: Solution design

This phase involves the project manager, business analyst, and solution architect.

The key activities are:

- Identify the module requirements
- Identify the configuration requirements in terms of:
 - Roles
 - Security groups
 - Escalations
 - Communication templates
 - Work order templates
 - Job plans
 - Service level agreements (SLAs)
 - Vendors and contracts
 - Panel configurations
 - General ledger codes
 - Workflow
- Create a detailed implementation plan
- Identify the methodology for integration
- Identify the scripting or manual data migration approach from identified data sources
- Create a solution architecture diagram

Phase 3: Deployment

This phase involves the project manager, solution architect, and Maximo developers. The optional roles are the DBA and application server specialist (depending on the environment).

The key activities are:

- Software installation:
 - Install the various components if you perform a manual install.
 - Run the launchpad for an automated install.
 - Test if the installation is successful.

The software installation differs for various deployment topologies: single server, multi-server, or clustered environment. It also depends on what infrastructure is selected.

- Software configuration:
 - Basic operation configuration, which includes, but is not limited to:
 - Creation of organization
 - Creation of locations
 - Creation of General Ledger (GL) codes
 - · Creation of users, roles, and security groups
 - Configure role-based start centers
 - Creation of job plans and work orders
 - Configuration of the tool to map to business processes, which include, but are not limited to:
 - Add or delete fields on panels and the database
 - · Configure workflows to map to business processes
 - Create communication templates
 - · Create escalation points and define SLAs in the system
 - · Create and test data migration scripts
 - · Migrate data from its existing source to Maximo
 - Build integration interfaces either using custom code or Maximo Enterprise Adapter (MEA)
 - · Create and distribute reports

Phase 4: Testing and going live

This phase involves the project manager, solution architect, testers, and Maximo developers. Optional roles are the DBA and the application server specialist.

Key activities are:

Testing

The first key task is the migration from the development environment to the test environment. Depending on the client's environment and the complexity of implementation, the client might decide to use automated tools for testing or a manual approach. Nevertheless, you need to perform all or part of the following types of testing:

- Functional testing:
 - Test validations in new fields
 - · Test e-mails being sent on escalations
 - Test workflows

- · Test the accuracy of data flowing between modules in the system
- Test the data accuracy in reports
- System integration testing:
 - · Test the accuracy of data coming from external systems
 - · Test the accuracy of data going into external systems
 - Use scenario testing to ensure that nothing affects the availability of any system
- Performance testing:
 - Page load time
 - Peak load
 - Concurrent users
 - Transaction processing time with database and external data sources
- Going live:
 - Plan the cutover strategy.
 - Migrate to the production environment.
 - Execute each of the previous test cases in the production environment to ensure zero performance errors in the live environment.
 - Provide training to all or several of the users. We recommended the "Train the Trainer" approach to three types of users:
 - Users: Functional training to the user community about how to use various functional aspects of the software. You can train them by either using focused groups or general groups depending on how the client organization is structured.
 - Role-based training: This training is specific to certain groups. Train the support purchase administrator, supervisor, and so on, about how they can approve or disapprove certain tasks, manage their queues, and so forth.
 - Administrative training: After the implementation team finishes, there
 has to be an internal designated person to manage the system. This
 person is responsible for system availability, the creation of new
 reports, enhancements, adding users, roles, and security groups, and
 so on.

Phase 5: Support

This phase involves the project manager and the Tivoli Asset Management for IT developers. Optional roles are the business analyst and the solution architect.
The key activities are:

Certain organizations do not have a strong IT presence, and they opt for the implementation team to provide continued system support. The basic tasks that the support team has to perform are, but are not limited to:

- Manage the system availability.
- Enhance the system to accommodate evolving business processes.
- Create and distribute new reports.
- Build new integrations.
- Plan capacity for additional users and organizations.

The five phase methodology is generic, but it includes most of the basic activities that you need to consider and plan for in an implementation. However, we have not yet discussed the client's involvement at each stage. For each of the mentioned phases, Table 6-1 provides the expectations from the client.

Table 6-1 Client involvement examples during implementation and deployment

| Phase | Expectations from the client |
|------------------------------------|--|
| Requirement gathering and analysis | Assign a single point of contact from within the company. Explain the project vision, short-term objective, and long-term objective. Facilitate meetings with individual process owners. Facilitate meetings with tool owners. Procure the hardware and software that are required for the project. Review documentation. |
| Solution design | Review the solution architecture. Review the documentation and project plan. |
| Deployment | Provide the hardware and software. Provide administrative access to systems wherever required. Monitor project progress. Participate in reviews. |
| Testing and going live | Provide test and production environment. Provide User Acceptance Testing (UAT) data. Facilitate in UAT. Attend training. Provide the staff and material resources that are required for training. |
| Support | Define the support scope of work boundaries. Review the support plan and SLAs. |



Part 3

Appendixes



Α

IBM Tivoli License Manager Administration server database schema

This appendix provides a list of database table descriptors and columns for the Tivoli License Compliance Manager Administration server database schema.

Table A-1 on page 238 shows the schema, which has been updated to Tivoli License Compliance Manager V2.3 Fix Pack 3 and is ordered by table name.

Major updates were added to Fix Pack 2 to cater for additional hardware information.

| Schema.Table name | Description | Columns |
|--------------------|--|------------------|
| ADM.ADMIN_CUST_REL | Maps the ADMINISTRATOR table to the CUSTOMER table (many-to-many). ADMINISTRATOR_ID is the external key to ADMINISTRATOR table. | ADMINISTRATOR_ID |
| | Unique identifier of the organization. External key to the CUSTOMER table. | CUSTOMER_ID |
| | Administrator profile. Foreign key to the PROFILE table. | PROFILE_ID |
| | Enables or disables the visibility of node information, for privacy reasons, for nodes to which the administrator has access, in inventory scan reports. Possible values: 0 Show 1 Hide (default) | HIDE_HOST_INV |
| | Enables or disables the visibility of node information, for privacy reasons, for nodes to which the administrator has access, in use reports. Possible values: 0 Show 1 Hide (default) | HIDE_HOST |
| | Enables or disables the visibility of user information, for privacy reasons, for users to which the administrator has access. Possible values: 0 Show 1 Hide (default) | HIDE_USER |
| | Enables or disables the visibility of user group information, for privacy reasons, for user groups to which the administrator has access. Possible values: 0 Show 1 Hide (default) | HIDE_GROUP |
| | Timestamp for when record was last modified. | LAST_MODIFIED |

Table A-1 IBM Tivoli License Compliance Manager Administration server database

| Schema.Table name | Description | Columns |
|-------------------|---|---------------------------|
| ADM.ADMINISTRATOR | ADMINISTRATOR stores details of Administration server administrators. ID is a unique identifier for the administrator. | ID |
| | User ID for logging on to Tivoli License Compliance Manager. | LOGON_NAME |
| | First name of the administrator. | FIRST_NAME |
| | Middle name or names of the administrator. Optional. | MIDDLE_NAME |
| | Last name of the administrator. | LAST_NAME |
| | Password (hashed) of the administrator. | PASSWORD |
| | The time when the password was defined or changed. | PASSWORD_CREATION_T |
| | The time when the password was locked. | LOCK_TIME |
| | The number of times that the user tried to log in using a wrong password. | FAILED_LOGIN_ATTEMPT S |
| | E-mail address of the administrator. | EMAIL_ADDRESS |
| | E-mail event notification flag. The default is 0. | EVENT_NOTIFICATION |
| | Telephone number of the administrator. | PHONE |
| | Fax number of the administrator. | FAX |
| | Unique identifier for importing (dataimp command) or exporting (dataexp command) action. | EXT_ID |
| | Timestamp for when record was last modified. | LAST_MODIFIED |
| ADM. AGENT | Stores details of installed Tivoli License Manager agents. ID is the unique identifier for the agent. | ID |
| | System host name (for example, server.ibm.com). | HOSTNAME |
| | Network IP address. Optional. | IP_ADDRESS |

| Schema.Table name | Description | Columns |
|-------------------|--|-------------|
| | Version of the agent that is installed, for example, 2.3. | VERSION |
| | The operating system name, such as: • Windows • Linux • AIX • Hewlett-Packard UNIX (HPUX) • Solaris™ • i5/OS® • UNIX | OS_NAME |
| | The operating system version, that is "Windows Server 2003 5.2". Optional. | OS_VERSION |
| | Date and time of the agent's last plug-in to the Runtime server. The agent sends the plug-in information when: The agent is initially deployed. The agent is upgraded. The agent has detected a change in the hardware of the node that it is monitoring. The Administration server database has been restored, and the Runtime server database has been dropped due to a recovery process. The tlmagent command is issued with the -p option. The default is the current timestamp. | PLUGIN_TIME |
| | Date and time of the last available software scan.Optional. | SCAN_TIME |

| Schema.Table name | Description | Columns |
|-------------------|---|---------------|
| | Indicates the agent status. Possible values: O false. The agent is not active. 1 true (default value). If true, the agent is active. An agent is considered inactive if it has not contacted the Runtime server for a length of time that can be configured by the user. 2 unknown. When the Runtime server has been recreated or is unavailable, the Administration server shows the entry related to the agent, but its status information is not available. | ACTIVE |
| | Unique identifier of the Runtime server. External key to the SERVER table. | SERVER_ID |
| | Unique identifier of the node. External key to the NODE table. | NODE_ID |
| | Unique identifier of the division. External key to the DIVISION table. | DIVISION_ID |
| | Unique identifier of the organization. External key to the CUSTOMER table. This is redundant (equates to ADM.DIVISION.CUSTOMER_ID). | CUSTOMER_ID |
| | Date and time that the agent was created. The default is the current timestamp. | CREATION_TIME |
| | | |

| Schema.Table name | Description | Columns |
|-------------------|--|----------------|
| | The agent security level. It is defined at the agent plug-in to the Runtime server and its value is dependent on the agent version. Possible values for Version 2.2 and 2.3 agents: 0 HTTP protocol 1 HTTPS protocol with server authentication 2 HTTPS protocol with server and client authentication Possible values for Version 2.1 and Version 1.1.1 agents: 1 HTTP protocol 4 HTTPS protocol with server authentication | SECURITY_LEVEL |
| | The inventory synchronization level. Possible values: 0 Inventory out of synchronization 1 Inventory synchronized Optional | INV_SYNCED |
| | Indicates whether the agent is plugged or the plug-in has to be forced. Possible values: 0 Agent is plugged (default value) 1 Force agent plug-in | FORCE_PLUG_IN |
| | New to 2.3. Custom agent data. Can be set at the agent command line or configuration file. Optional. | CUSTOM_DATA1 |
| | New to 2.3. Custom agent data. Can be set at the agent command line or configuration file. Optional. | CUSTOM_DATA2 |
| | New to 2.3. Custom agent data. Can be set at the agent command line or configuration file. Optional. | CUSTOM_DATA3 |
| | Date and time of the last available hardware scan. Introduced with 2.3 Fix Pack 1. Optional. | HWSCAN_TIME |
| | Timestamp for when record was last modified. | LAST_MODIFIED |

| Schema.Table name | Description | Columns |
|-----------------------|---|-------------|
| ADM. AGENT_DELETED | Deleted agent.ID is the identifier for deleted agents. | ID |
| | System host name (for example, server.ibm.com). | HOSTNAME |
| | Network IP address. Optional. | IP_ADDRESS |
| | Version of the agent that is installed, for example, 2.3. | VERSION |
| | The operating system name: • Windows • Linux • AIX • HPUX • Solaris • i5/OS • UNIX | OS_NAME |
| | The operating system version, that is "Windows Server 2003 5.2". Optional. | OS_VERSION |
| | Date and time of the agent last plug-in to the Runtime server. The agent sends the plug-in information when: The agent is initially deployed. The agent is upgraded. The agent has detected a change in the hardware of the node that it is monitoring. The Administration server database has been restored, and the Runtime server database has been dropped due to a recovery process. The tlmagent command is issued with the -p option. | PLUGIN_TIME |
| | Date and time of the last available software scan.Optional. | SCAN_TIME |
| | Unique identifier of the Runtime server. External key to the SERVER table. | SERVER_ID |
| | Unique identifier of the node. External key to the NODE table. | NODE_ID |

| Schema.Table name | Description | Columns |
|-------------------|---|----------------|
| | Unique identifier of the division. External key to the DIVISION table | DIVISION_ID |
| | Unique identifier of the organization. External key to the CUSTOMER table. This is redundant (equates to ADM.DIVISION.CUSTOMER_ID). | CUSTOMER_ID |
| | Date and time the agent was created. The default is current timestamp. | CREATION_TIME |
| | The agent security level. It is defined at the agent plug-in to the Runtime server, and its value is dependent on the agent version. Possible values for Version 2.2 and 2.3 agents: 0 HTTP protocol 1 HTTPS protocol with server authentication 2 HTTPS protocol with server and client authentication Possible values for Version 2.1 and Version 1.1.1 agents: 1 HTTP protocol 4 HTTPS protocol with server authentication | SECURITY_LEVEL |
| | New to 2.3. Custom agent data. Can be set at the agent command line or configuration file. Optional. | CUSTOM_DATA1 |
| | New to 2.3. Custom agent data. Can be set at the agent command line or configuration file. Optional. | CUSTOM_DATA2 |
| | New to 2.3. Custom agent data. Can be set at the agent command line or configuration file. Optional. | CUSTOM_DATA3 |
| | Date and time of the last available hardware scan. Introduced with 2.3 Fix Pack 1. Optional. | HWSCAN_TIME |
| | Date and time when the agent was deleted. | DELETED_TIME |

| Schema.Table name | Description | Columns |
|---------------------|--|--------------|
| ADM. AGENT_EVENT | Stores details of events at agents. AGENT_ID is Unique identifier of the agent. External key to the AGENT table. | AGENT_ID |
| | The ID of the event. Possible values: 1 Inventory scan | EVENT_ID |
| | Time when the event occurred. | TIME |
| ADM .AGENT_INV | Provides historical software inventory of each agent. ID is the unique identifier of the inventory record. | ID |
| | Unique identifier of the agent. External key to the AGENT table. | AGENT_ID |
| | Unique identifier of the organization. External key to CUSTOMER table. | CUSTOMER_ID |
| | Software product ID. External key to the COMPONENT table. | COMPONENT_ID |
| | Date and time of the scan that detected the software. | SCAN_TIME |
| | The installation path of the discovered component. Optional. | SCOPE |
| | The hash code of the installation path of the discovered component. Optional. | SCOPE_ID |
| | The number of product inventory records obtained from this component record. Default=0. | LINKS |
| | | |

| Schema.Table name | Description | Columns |
|----------------------|---|---------------|
| | The reason why the agent has disabled the multi-instance. Possible values: 1. Disabled because the component has registry keys as recognition signatures. 2. Disabled because the usage signatures of the component are instrumented applications or Java 2 Platform Enterprise Edition (J2EE) applications. 3. Disabled because there is no match between the installation path of the recognition signatures of the component and the installation path of the usage signatures of the component. | MI_REASON |
| | Timestamp for when record was last modified. | LAST_MODIFIED |
| ADM. AGENT_STATUS | AGENT_STATUS maps the Tivoli License Compliance Manager agent to the Tivoli Configuration Manager agent. SYS_ID is the unique identifier of the Tivoli endpoint where the Tivoli License Compliance Manager agent runs. This table is new to 2.3. | SYS_ID |
| | Unique identifier of the Tivoli License Compliance Manager agent. | AGENT_ID |
| | Unique identifier of the division where the agent runs. | DIVISION_ID |
| | The customer to which the agent belongs. | CUSTOMER_NAME |
| | The agent status. | STATUS |
| | The Tivoli endpoint (object identifier) to which the agent belongs. | ENDPOINT_OID |
| | Timestamp for when record was last modified. | LAST_MODIFIED |

| Schema.Table name | Description | Columns |
|---------------------------------------|--|------------|
| ADM. AGGREGATION_PROPERT Y | Stores the information generated by the aggregation process. REF_ID is the external key that identifies the subject of the property. New table in 2.3. | REF_ID |
| | The type of property. | ТҮРЕ |
| | The initial time that the property has been detected. | START_TIME |
| | The time when the property no longer qualifies the subject. | END_TIME |
| | The property value as a string. Optional. | VALUE_STR |
| | The property value as a number. Optional. | VALUE_NUM |
| ADM. AGGREGATION_SKIPPED_ AGENT | Includes the agents that were skipped during the aggregation process. AGENT_ID is the identifier of the agent. New table in 2.3. | AGENT_ID |
| | The identifier of the agent. | START_DATE |
| | The identifier of the agent. | END_DATE |
| ADM. AGT_PARTIAL_VM | Shows the partial relationship (session) between the agent and the virtual machine (VM) layer. AGENT_ID is the unique identifier of the agent. External key to the AGENT table. New table in 2.3. | AGENT_ID |
| | The hash of the VM layer. | LAYER_HASH |
| | The date and time when the session was started. | START_TIME |
| | The date and time when the session was stopped. | END_TIME |
| ADM. AGT_VM_REL | Stores the relationship between the VM layer and the agent. AGENT_ID is the unique identifier of the agent. External key to the AGENT table. | AGENT_ID |
| | Unique identifier of the VM layer. External key to the VM table. | VM_ID |

| Schema.Table name | Description | Columns |
|------------------------------------|---|---------------------|
| | Date and time the agent was started. | START_TIME |
| | Date and time the agent was stopped. | END_TIME |
| ADM. ALGORITHM_RELATIONSH IP | Stores the relationships between each mapping rule and each entity (organization) defined in the database. New table in 2.3 Fix Pack 2. | ID |
| | Unique identifier of a mapping rule defined in the SW_INV_ALGORITHM table. | SW_INV_ALGORITHM_ID |
| | Indicates the relationship between the entity (organization) and the mapping rule: ► 1 Applied ► 2 Not applied | TYPE |
| | Type of entity for which the mapping relationship is defined: ▶ 1 organization ▶ 2 division (not currently used) | TARGET_TYPE |
| | The unique ID of the organization. | TARGET_ID |
| ADM. BASE_PROPERTY | Stores information about the activation or deactivation of mapping rules and information about the definition of component to product links. New table in 2.3 Fix Pack 2. | ID |
| | Identifies the type of property: 1 Server (automatic)-defined link 2 User-defined link 100 Enablement/disablement of mapping rules | TYPE |

| Schema.Table name | Description | Columns |
|-------------------|--|---------------|
| | Identifies the table to which this property refers: 1 Link: The row is related to information in the LINK table about links between components and products. 2 Sw_Inv_Algorithm: Not currently used. 3 Algorithm_Relationship: The row relates to information in the ALGORITHM_RELATIONSHIP table about the relationship between organizations and mapping rules. | OWNER_TYPE |
| | Identifies the row in the LINK or ALGORITHM_RELATIONSHIP table to which this row relates. | OWNER_ID |
| | For properties of TYPE 100, this identifies the mapping enablement setting: ► 0 Off ► 1 On Optional. | VALUE_INT |
| | For server or user-defined links, identifies the user that defined the link. For server-defined links, the value is "Server", for user-defined links, it is a Web UI user ID. Optional. | VALUE_STR |
| | Timestamp for when record was last modified. | LAST_MODIFIED |
| BATCH_REPORT | Stores requests for batch reports. ID is the unique identifier of the batch report request. | ID |
| | Unique identifier of the organization that requested the batch report. External key to CUSTOMER table. | CUSTOMER_ID |

| Schema.Table name | Description | Columns |
|-------------------|---|-----------------|
| | Type of batch report. Possible types: installs_by_component installs_by_endpoint installs_by_component_full installs_by_endpoint_full license_compliance unlicensed_use use_level licensed_use_trend product_use_trend | TYPE |
| | Universal Time, Coordinated (UTC) time of request submission. | REQUEST_TIME |
| | UTC time of report generation by the Administration server. | GENERATION_TIME |
| | Logon name of the user who submitted the request for the batch report. | USER_ID |
| | Current status of the batch report request. Possible values: • queued • running • completed | STATUS |
| | Stores options that define the report scope. | PARAMETERS |
| | Data of the requested report in XML format. | REPORT_DATA |
| SWCAT. BRANCH | Contains the software catalog structure required by the Administration server. ID is the unique identifier of the branch. New table in 2.3. | ID |
| | Unique identifier of the software product. External key to SWPRODUCT table. | PRODUCT_ID |
| | Unique identifier of the component. External key to COMPONENT table. | COMPONENT_ID |

| Schema.Table name | Description | Columns |
|----------------------|--|---------------|
| | Relation type. Possible types: CHARGE=11 The component must be charged to the product. NOCHARGE=10 The component is not charged to the product. MANAGEDBY=20 The component is managed by the reference product. | TYPE |
| | Connection path of all catalog objects' IDs related to the top level component. Optional. | PATH |
| | Indicates whether the branch is valid or has been deleted. Possible values: O The branch is valid (default) The branch has been deleted | DELETED |
| | Indicates whether the branch is complex or simple. Possible values: 1 The branch is simple (default) 2 The branch is complex | IS_COMPLEX |
| | Timestamp for when record was last modified. | LAST_MODIFIED |
| ADM. CATALOG | Stores information about the catalog. New in 2.3. PLATFORM is in the form of: Windows i5/OS Linux AIX HPUX Solaris | PLATFORM |
| | Unique identifier of the catalog. | HASH |
| SWG. COMP_SIG_REL | Relationships between software components. COMPONENT_ID is the external key to the COMPONENT table. | COMPONENT_ID |
| | Unique identifier for the signature. External key to the SIGNATURE table. | SIGNATURE_ID |

| Schema.Table name | Description | Columns |
|---------------------|--|---------------|
| | Status of the link between components and signatures. Possible values: 0 enabled (default) 1 disabled (use) 2 disabled (install) 3 disabled (both) | STATUS |
| | Timestamp for when record was last modified. | LAST_MODIFIED |
| SWCAT. COMPONENT | Stores details about software components. ID is the unique identifier for the software component. | ID |
| | Software component name. | NAME |
| | Software component version. | VERSION |
| | Unique identifier of the operating system. External key to the PLATFORM table. | PLATFORM |
| | Free format text. Optional. | DESCRIPTION |
| | Unique identifier of the software vendor. External key to the VENDOR table. | VENDOR_ID |
| | Internal use. External ID used by the Catalog Manager. | EXT_ID |
| | Indicates if the component is valid or has been deleted. Possible values: 0 Valid component (default) 1 Component has been deleted | DELETED |
| | Timestamp for when record was last modified. | LAST_MODIFIED |
| TLMHW. COMPONENT | New in 2.3 Fix Pack 1. Stores information on hardware components. | ID |
| | The component model. Optional. | MODEL |
| | The component serial number. Note you must define this in the custom fields. Optional. | SERIAL_NUMBER |
| | The component manufacturer. Optional. | MANUFACTURER |
| | The component version. Optional. | VERSION |
| | The component type. Optional. | ТҮРЕ |

| Schema.Table name | Description | Columns |
|--------------------|---|--------------------|
| | The model class of the component. Optional. | MODEL_CLASS |
| | Date and time when the entry was added to the database. | CREATE_TIME |
| | Timestamp for when record was last modified. | UPDATE_TIME |
| TLMHW. COMPUTER | New in 2.3 Fix Pack 1. Stores common information about a computer system. One entry exists for each system scanned. COMPUTER_SYS_ID is the unique identifier for hardware-related data. | COMPUTER_SYS_ID |
| | The Globally Unique Identifier (GUID) of the system. Optional. | COMPUTER_GUID |
| | Indicates the rule for generation of the GUID that was in use when the current information was collected. Possible values are: 1 Concatenate the values for manufacturer, model, and serial number. 3 Use the Media Access Control (MAC) address of the first network card discovered on the system. GUID creation is controlled with the command line interface (CLI) hwscanguidrule command. | COMPUTER_GUID_TYPE |
| | The date and time of the most recent hardware scan in Greenwich Mean Time. Optional. | COMPUTER_SCANTIME |
| | The model of the system. Optional. | COMPUTER_MODEL |
| | The date and time that the system was last restarted. Optional. | COMPUTER_BOOT_TIME |
| | The host name of the system. Optional. | COMPUTER_ALIAS |
| | The serial number of the system. Optional. | SYS_SER_NUM |

| Schema.Table name | Description | Columns |
|-------------------|---|------------------|
| | The specific operating system that is installed, for example, Windows Server 2003 Standard Edition. Optional. | OS_NAME |
| | The type of operating system that is installed, for example, Windows Server 2003. Optional. | OS_TYPE |
| | The major version of the operating system. Optional. | OS_MAJOR_VERS |
| | The minor version of the operating system. Optional. | OS_MAJOR_VERS |
| | The sub-version of the operating system. Optional. | OS_SUB_VERS |
| | The date when the operating system was installed. Optional. | OS_INST_DATE |
| | The name of the owner of the system. Optional. | REGISTERED_OWNER |
| | The name of the organization to which the system belongs. Optional. | REGISTERED_ORG |
| | A description of the type of keyboard assigned to the system. Optional. | KEYBOARD_TYPE |
| | The number of function keys on the keyboard. Optional. | FUNCTION_KEYS |
| | The locale of the time zone in which the system is located. Optional. | TZ_LOCALE |
| | The name of the time zone in which the system is located. Optional. | TZ_NAME |
| | The name of the daylight saving time zone in which the system is located. Optional. | TZ_DAYLIGHT_NAME |
| | Indicates whether the system is on daylight saving time. Optional. | ON_SAVINGS_TIME |
| | The number of seconds difference between the system time zone and Greenwich Mean Time. Optional. | TZ_SECONDS |

| Schema.Table name | Description | Columns |
|----------------------------|--|-------------------|
| | Indicates whether the time zone is early or later than Greenwich Mean Time. Optional. | TIME_DIRECTION |
| | The operating system architecture for the system. Optional. | OS_ARCH |
| | The operating system kernel mode. Optional. | OS_KERNEL_MODE |
| | The operating system language version identifies the default language for the system. Optional. | OS_LANG_VERS |
| | The operating system locale identifier. Optional. | OS_LCID |
| | The current locale identifier. Optional. | CURRENT_LCID |
| | The hash of the virtual layer that hosts the agent. Optional. | VM_CHAIN_HASH |
| | Indicates whether the monitored system is virtualized. Possible values are: ► 0 (not virtualized) or ► 1 (virtualized) | VIRTUAL_COMP_TYPE |
| | Date and time when the entry was added to the database. | CREATE_TIME |
| | Timestamp for when record was last modified. | UPDATE_TIME |
| TLMHW. COMPUTER_SYS_MEM | Stores information about the physical and virtual memory installed on a system. One entry exists for each system scanned. New in 2.3 Fix Pack 1. COMPUTER_SYS_ID is the unique identifier for the system. | COMPUTER_SYS_ID |
| | The total physical memory in KB. Optional. | PHYSICAL_TOTAL_KB |
| | The amount of free physical memory in KB. Optional. | PHYSICAL_TOTAL_KB |
| | The total number of physical memory pages. Optional. | TOTAL_PAGES |

| Schema.Table name | Description | Columns |
|-------------------|---|-----------------------|
| | The number of free physical memory pages. Optional. | FREE_PAGES |
| | The size of a page. Optional. | PAGE_SIZE |
| | The total amount of virtual memory in KB. Optional. | VIRT_TOTAL_KB |
| | The amount of free virtual memory in KB. Optional. | VIRT_FREE_KB |
| | Date and time when the entry was added to the database. | CREATE_TIME |
| | Timestamp for when record was last modified. | UPDATE_TIME |
| LIC. CONF_KEY | Stores meta-information for each agent configuration parameter. When the fix pack is installed, this table is created and populated with an entry for each of the agent configuration parameters listed in Agent configuration parameters. New in 2.3 Fix Pack 2. | ID |
| | The name of the configuration parameter. | NAME |
| | Identifies the highest hierarchy node for this implementation. | HIER_ID |
| | Indicates whether the value can be inherited from higher topology nodes. Possible values are: ► 0 No ► 1 Yes | IS_INHERITABLE |
| | Indicates the maximum number of levels in the hierarchy by which a value for the configuration parameter can be inherited. | MAX_VISIBILITY_HEIGHT |
| | The value of this parameter, which is set at installation time. | PREDEFINED_VALUE |

| Schema.Table name | Description | Columns |
|------------------------|---|------------------|
| | Indicates whether the value for the configuration parameter must be set for the highest hierarchy node and inherited by all underlying nodes. 0 No 1 Yes | ONLY_INHERITABLE |
| | Indicates the unit of measurement for the configuration parameter. Possible values are: 0 integer 1 string 2 timestamp | VALUE_TYPE |
| | Text description of the units of measurement. Optional. | UNITS |
| LIC. CONF_KEY_VALUE | Stores all configuration parameter values that have been defined for topology nodes. New table 2.3 Fix Pack 2. ID is the unique identifier of the CONF_KEY_VALUE. | ID |
| | The date and time when the configuration was defined. | TIMESTAMP |
| | The value assigned to the configuration parameter. Optional. | VALUE |
| | Indicates whether the configuration value is in the active or hold state. Possible values are: 0 hold 1 active | STATE |
| | Uniquely identifies the configuration parameter. | CONF_KEY_ID |
| | Uniquely identifies the topology node for which the configuration parameter value is defined. Optional. | GROUP_ID |
| | Uniquely identifies the resource with which the configuration parameter value is associated. Optional. | RESOURCE_ID |
| CONTRACT | Stores details of contracts. | CONTRACT_NUM |
| | | TYPE |
| | | STATUS |
| | | SUPPLY_VENDOR |
| | | PURCHASE_VENDOR |

| Schema.Table name | Description | Columns |
|-------------------|--|---------------|
| | | OWNER |
| | | START_DATE |
| | | END_DATE |
| | | COST |
| | | COST_CURR |
| | | TERMS |
| | | NOTES |
| | | MODIFIED_BY |
| | | EXT_ID |
| | | CUSTOMER_ID |
| | | LAST_MODIFIED |
| CONTROL | Stores configuration information in key-value format. | NAME |
| | | VALUE |
| COUNTRY | Stores country information. | ISO_CODE3 |
| | | NAME |
| | | ISO_CODE2 |
| | | ISO_CODE4 |
| | | CURRENCY_NAME |
| | | CURRENCY_CODE |
| CUST_OPT_FIELD | Maps the CUSTOMER table to the OPT_FIELD_DEF table (many-to-many). It provides the list of available custom fields for each organization. | OPT_FIELD_ID |
| | | CUSTOMER_ID |
| CUSTOMER | Stores information about each organization. | ID |
| | | NAME |
| | | ACCOUNT_ID |

| Schema.Table name | Description | Columns |
|-------------------|---|----------------|
| | | COUNTRY_CODE |
| | | DESCRIPTION |
| | | CHECK_PERIOD |
| | | REQUEST_SCOPE |
| | | LAST_MODIFIED |
| DIVISION | Stores details of divisions that represent the organization primary structural level. | ID |
| | | NAME |
| | | INV_START_DATE |
| | | INV_RATE_TYPE |
| | | INV_RATE_VALUE |
| | | DESCRIPTION |
| | | SELFUPDATE |
| | | EXT_ID |
| | | CUSTOMER_ID |
| | | CREATION_TIME |
| | | LAST_MODIFIED |
| DIVISION_DELETED | Deleted division. | ID |
| | | NAME |
| | | INV_START_DATE |
| | | INV_RATE_TYPE |
| | | INV_RATE_VALUE |
| | | DESCRIPTION |
| | | SELFUPDATE |
| | | EXT_ID |
| | | CUSTOMER_ID |
| | | CREATION_TIME |

| Schema.Table name | Description | Columns |
|-------------------|---|-----------------|
| | | DELETED_TIME |
| ENDUSER | Stores details of users of the applications that are monitored by Tivoli License Manager. | ID |
| | | LOGON_NAME |
| | | FIRST_NAME |
| | | MIDDLE_NAME |
| | | LAST_NAME |
| | | EMPLOYEE_NO |
| | | EMAIL_ADDRESS |
| | | LOCATION |
| | | PHONE |
| | | FAX |
| | | EXT_ID |
| | | CUSTOMER_ID |
| | | LAST_MODIFIED |
| ENTITLEMENT | Details the usage rules for product licenses for which an organization has set up an entitlement. | ID |
| | | PRODUCT_ID |
| | | CUSTOMER_ID |
| | | ACTIVE |
| | | INV_STATUS |
| | | MI_STATUS |
| | | LAST_MODIFIED |
| IBM_REPORT | Contains the generated IBM reports and possibly the client comment associated to them. | ID |
| | | GENERATION_TIME |

| Schema.Table name | Description | Columns |
|-------------------|---|------------------|
| | | REPORT |
| | | START_DATE |
| | | END_DATE |
| | | CUSTOMER_ID |
| | | IS_EXPORTED |
| | | CUSTOMER_COMMENT |
| INV_H_DIV | Stores inventory scan data aggregated by division. | DIVISION_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | HWM |
| | | UNLICENSED_HWM |
| INV_H_LIC | Stores inventory scan data aggregated by license. | LICENSE_ID |
| | | DATE_USAGE |
| | | ID |
| | | нwм |
| INV_H_LIC_PROD | Stores inventory scan data aggregated by license and component. | LICENSE_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | НШМ |
| INV_H_PLIC | Stores inventory scan data aggregated by procured license. | PLICENSE_ID |
| | | DATE_USAGE |
| | | ID |
| | | Н₩М |

| Schema.Table name | Description | Columns |
|-------------------|--|----------------|
| INV_H_PLIC_PROD | Stores inventory scan data aggregated by procured license and component. | PLICENSE_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | HWM |
| INV_H_PROD | Stores inventory scan data aggregated by component. | CUSTOMER_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | нмм |
| | | UNLICENSED_HWM |
| LIC_TARGET_HREL | History of LIC_TARGET_REL table. | LICENSE_ID |
| | | TARGET_ID |
| | | START_TIME |
| | | END_TIME |
| LIC_TARGET_REL | Maps license distributions to an organization or to one or more targets (divisions, nodes, or agents). | LICENSE_ID |
| | | TARGET_ID |
| | | LAST_MODIFIED |
| LIC_USER_HREL | History of LIC_USER_REL table. | LICENSE_ID |
| | | ENDUSER_ID |
| | | START_TIME |
| | | END_TIME |
| LIC_USER_REL | Maps the relationships between licenses and users (many-to-many). | LICENSE_ID |
| | | ENDUSER_ID |

| Schema.Table name | Description | Columns |
|-------------------|---|---------------|
| | | LAST_MODIFIED |
| LICENSE | Stores information about license pools. | ID |
| | | PLICENSE_ID |
| | | SERIAL_ID |
| | | QUANTITY |
| | | TARGET_TYPE |
| | | ALL_USERS |
| | | LAST_MODIFIED |
| LICENSE_H | History of LICENSE table. | ID |
| | | PLICENSE_ID |
| | | SERIAL_ID |
| | | QUANTITY |
| | | TARGET_TYPE |
| | | ALL_USERS |
| | | START_TIME |
| | | START_ACTION |
| | | END_TIME |
| | | END_ACTION |
| LINK | Stores information about the user deployment of installed components. | ID |
| | | SCOPE |
| | | SCOPE_ID |
| | | COMPONENT_ID |
| | | AGENT_ID |
| | | ТҮРЕ |
| | | BRANCH_ID |
| | | CUSTOMER_ID |

| Schema.Table name | Description | Columns |
|-------------------|--|--------------------|
| | | PRODUCT_ID |
| | | BRANCH_DEL |
| | | LAST_MODIFIED |
| MEASURE | Stores aggregated capacity data. | ID |
| | | METRIC_ID |
| | | TARGET_ID |
| | | TARGET_TYPE |
| | | QUANTITY |
| | | START_TIME |
| | | END_TIME |
| MEASURE_RAW | Stores raw capacity data. | ID |
| | | METRIC_ID |
| | | LAYER_HASH |
| | | AGENT_ID |
| | | QUANTITY |
| | | DATE_SCAN |
| METRIC | Stores the available metrics. | ID |
| | | NAME |
| | | CAPACITY_TYPE |
| NODE | Stores details of systems on which agents are installed. | ID |
| | | TAG |
| | | PLATFORM |
| | | NAME |
| | | HASH |
| | | HARDWARE_MODEL |
| | | HARDWARE_MANUFACTU |

| Schema.Table name | Description | Columns |
|-------------------|--|--------------------|
| | | HARDWARE_TYPE |
| | | LOCATION |
| | | DESCRIPTION |
| | | PLUGIN_TIME |
| | | CUSTOMER_ID |
| | | CREATION_TIME |
| | | LAST_MODIFIED |
| NODE_DELETED | Table of deleted nodes. | ID |
| | | TAG |
| | | PLATFORM |
| | | NAME |
| | | HASH |
| | | HARDWARE_MODEL |
| | | HARDWARE_MANUFACTU |
| | | HARDWARE_TYPE |
| | | LOCATION |
| | | DESCRIPTION |
| | | PLUGIN_TIME |
| | | CUSTOMER_ID |
| | | CREATION_TIME |
| | | DELETED_TIME |
| OID | Stores the most recent (highest) system-generated number for various tables throughout the database. | TABLE_NAME |
| | | LAST_ID |
| | | BLOCK_SIZE |
| OPT_FIELD | Stores the values of optional fields. | OPT_FIELD_ID |
| | | TABLE_RECORD_ID |

| Schema.Table name | Description | Columns |
|-------------------|--|--------------------|
| | | VALUE |
| OPT_FIELD_DEF | Defines optional fields. Each line defines one optional field for one table. | ID |
| | | TABLE_NAME |
| | | NAME |
| | | TYPE |
| PACKET_CONTROL | This table is used to avoid processing a specific packet coming from a specific Runtime twice. | CLIENT_ID |
| | | SERVICE_ID |
| | | PACKET_ID |
| PLATFORM_MAPPING | | AGENT_PLATFORM |
| | | COMPONENT_PLATFORM |
| PLIC_PROD_HREL | History of PLIC_COMP_REL table. | PLICENSE_ID |
| | | PRODUCT_ID |
| | | START_TIME |
| | | END_TIME |
| PLIC_PROD_REL | Maps the relationships between procured licenses and components (many-to-many). | PLICENSE_ID |
| | | PRODUCT_ID |
| PLICENSE | Stores details of procured software licenses. | ID |
| | | EE_ID |
| | | REF_CODE |
| | | TREE_LEVEL |
| | | SOFTWARE_NAME |
| | | OWNER |
| | | LICENSE_TYPE |

| Schema.Table name | Description | Columns |
|-------------------|----------------------------|-----------------|
| | | QUANTITY |
| | | PURCHASE_TYPE |
| | | COST |
| | | COST_CURR |
| | | DELIVERY_DATE |
| | | START_DATE |
| | | EXPIRATION_DATE |
| | | CONTRACT_REF |
| | | ORDER_REF |
| | | LICENSE_KEY |
| | | тс |
| | | NOTES |
| | | THRESHOLD |
| | | VERIFIED |
| | | EXT_ID |
| | | CUSTOMER_ID |
| | | MODIFIED_BY |
| | | PID |
| | | CCID |
| | | PROTECTED_MASK |
| | | LAST_MODIFIED |
| PLICENSE_H | History of PLICENSE table. | ID |
| | * | REF_CODE |
| | | TREE_LEVEL |
| | | SOFTWARE_NAME |
| | | LICENSE_TYPE |
| | | EE_ID |

| Schema.Table name | Description | Columns |
|-------------------|---------------------------|-----------------|
| | | QUANTITY |
| | | DELIVERY_DATE |
| | | START_DATE |
| | | EXPIRATION_DATE |
| | | THRESHOLD |
| | | CUSTOMER_ID |
| | | PID |
| | | CCID |
| | | START_TIME |
| | | START_ACTION |
| | | END_TIME |
| | | END_ACTION |
| PROD_HINV | Inventory products table. | ID |
| | | SCOPE |
| | | SCOPE_ID |
| | | COMPONENT_ID |
| | | PRODUCT_ID |
| | | AGENT_ID |
| | | BRANCH_ID |
| | | START_TIME |
| | | END_TIME |
| PROD_INV | | ID |
| | | SCOPE |
| | | SCOPE_ID |
| | | COMPONENT_ID |
| | | AGENT_ID |
| | | BRANCH_ID |
| Schema.Table name | Description | Columns |
|-------------------|--|-----------------|
| | | PRODUCT_ID |
| | | AGENT_INV_ID |
| | | LINK_ID |
| | | ТҮРЕ |
| | | START_TIME |
| | | END_TIME |
| PROFILE | Describes administrator profiles. | ID |
| | | NAME |
| | | DESCRIPTION |
| PROFILE_ACTION | Actions not allowed for administrator profiles. | PROFILE_ID |
| | | ACTION_KEY |
| | | ACTION_VALUE |
| SERVER | Stores information about the Tivoli License Manager Runtime servers. | ID |
| | | NAME |
| | | ADDRESS |
| | | PASSWORD |
| | | PASSWORD_TYPE |
| | | VERSION |
| | | PORT_NUMBER |
| | | SSL_PORT_NUMBER |
| | | FORCE_SERVICE |
| | | DOWNLOAD_PERIOD |
| | | UPLOAD_PERIOD |
| | | UPDATE_PERIOD |
| | | WARNING_TIME |
| | | LOCATION |

| Schema.Table name | Description | Columns |
|--------------------|--|--------------------|
| | | CUSTOMER_ID |
| | | ADMIN_RECOVERY |
| | | CREATION_TIME |
| | | AGT_SECURITY_LEVEL |
| | | LAST_MODIFIED |
| SERVER_DELETED | | ID |
| | | NAME |
| | | ADDRESS |
| | | PASSWORD |
| | | PASSWORD_TYPE |
| | | VERSION |
| | | PORT_NUMBER |
| | | SSL_PORT_NUMBER |
| | | FORCE_SERVICE |
| | | DOWNLOAD_PERIOD |
| | | UPLOAD_PERIOD |
| | | UPDATE_PERIOD |
| | | WARNING_TIME |
| | | LOCATION |
| | | CUSTOMER_ID |
| | | ADMIN_RECOVERY |
| | | CREATION_TIME |
| | | DELETED_TIME |
| SERVER_SERVICE_TIM | | |
| SERVICE | Provides a history of the service requests made. | SERVICE_ID |
| | | SERVER_ID |

| Schema.Table name | Description | Columns |
|--------------------|-------------|------------------|
| | | INSERTED |
| | | FORCE_CHECK_TIME |
| | | DESCRIPTION |
| TEMP_AGENT_INV | | |
| | | AGENT_ID |
| | | CUSTOMER_ID |
| | | COMPONENT_ID |
| | | SCAN_TIME |
| | | SCOPE |
| | | SCOPE_ID |
| TEMP_BRANCH_REPLAC | | |
| TEMP_ENTITLEMENT_M | | |
| TEMP_INV_LICENSED_ | | |
| TEMP_INV_PRODUCT_A | | |
| TEMP_INV_UNLICENSE | | |
| TEMP_PROD_INV | | SCOPE_ID |
| | | COMPONENT_ID |
| | | AGENT_ID |
| | | PRODUCT_ID |
| | | TYPE |
| | | IS_I5OS |
| TEMP_USAGE_LICENSE | | |
| TEMP_USAGE_PRODUCT | | |
| TEMP_USAGE_SW_PROD | | COMPONENT_ID |
| | | PRODUCT_ID |
| | | AGENT_ID |
| | | START_TIME |

| Schema.Table name | Description | Columns |
|--------------------|---|--------------------|
| | | END_TIME |
| | | SCOPE_ID |
| | | USERNAME |
| | | ТҮРЕ |
| | | AGT_TRANSACTION_ID |
| TEMP_USAGE_UNLICEN | | |
| USAGE_COMP | Stores product usage information. | ID |
| | | COMPONENT_ID |
| | | SCOPE_ID |
| | | AGENT_ID |
| | | START_TIME |
| | | END_TIME |
| | | USERNAME |
| | | AGT_TRANSACTION_ID |
| USAGE_H_DIV | Stores usage data aggregated by component and division. | DIVISION_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | HWM |
| | | HWM_TIME |
| | | AWM |
| | | UNLICENSED_HWM |
| USAGE_H_LIC | Stores usage data aggregated by license. | LICENSE_ID |
| | | DATE_USAGE |
| | | ID |
| | | HWM |

| Schema.Table name | Description | Columns |
|-------------------|---|-------------|
| | | HWM_TIME |
| | | AWM |
| USAGE_H_LIC_PROD | Stores usage data aggregated by license and component. | LICENSE_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | HWM |
| | | HWM_TIME |
| USAGE_H_PLIC | Stores usage data aggregated by procured license. | PLICENSE_ID |
| | | DATE_USAGE |
| | | ID |
| | | НММ |
| | | HWM_TIME |
| | | AWM |
| USAGE_H_PLIC_PROD | Stores usage data aggregated by procured license and product. | PLICENSE_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | HWM |
| | | HWM_TIME |
| USAGE_H_PROD | Stores usage data aggregated by component. | CUSTOMER_ID |
| | | PRODUCT_ID |
| | | DATE_USAGE |
| | | ID |
| | | Н₩М |

| Schema.Table name | Description | Columns |
|-------------------|--|----------------|
| | | HWM_TIME |
| | | AWM |
| | | UNLICENSED_HWM |
| VM | Stores the VM layers where the agents run. | ID |
| | | LAYER_NAME |
| | | LAYER_HASH |
| | | NODE_ID |
| | | VM_PARENT_ID |
| | | ТҮРЕ |
| | | DEPTH |

Β

IBM Tivoli License Compliance Manager agents

This appendix provides examples of:

- ► The tImagent.ini itIm agent settings file (Example B-1)
- The tivhscan.mif hardware mif file created during a hardware scan (Example B-2 on page 279). The hardware scanner is new to 2.3 Fix Pack 1.

Example: B-1 tlmagent.ini file

```
#
            TLM Agent configuration file
                                                  #
#
                                                  #
******
#
#
            AGENT PARAMETERS
# Agent Identifier
# (Reloadable: No)
agentid = 5161005547943887625
# Division Name
# (Reloadable: No)
division = ITSO WIN
# Organization
# (Reloadable: No)
organization = IBMITSO
# Max agent cache size (in bytes, values between 50bytes and 15MB
allowed)
# (Reloadable: No)
max cache size = 2097152
# Unknown files processing enabled [y/n]
# (Reloadable: Yes)
unknown files enabled = y
# Users whose home directory has to be searched for WAS Registry files
(user1, user2, ...;)
# (Reloadable: Yes)
was registry users = Administrator
# Trace size (in bytes, values between 64KB and 16MB allowed)
# This parameter has no effect unless parameter trace files is greater
than one
# (Reloadable: Yes)
trace size = 1000000
# Trace level (MIN,MID,MAX)
# (Reloadable: Yes)
```

```
trace level = MIN
# Number of trace files (values between 1 and 10 allowed)
# (Reloadable: No)
trace files = 10
# Message log size (in bytes, values between 64KB and 16MB allowed)
# This parameter has no effect unless parameter msglog files is greater
than one
# (Reloadable: Yes)
msglog size = 1000000
# Number of message log files (values between 1 and 10 allowed)
# (Reloadable: No)
msglog files = 4
# Tivoli Common Directory
# (Reloadable: No)
tivoli common dir = C:\Program Files\ibm\tivoli\common
# Agent Base Directory
# (Reloadable: No)
agent base dir = C:\WINDOWS\itlm
# Agent Temporary Directory
# (Reloadable: No)
agent temp dir = C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\\itlm
# Communication type (1=HTTP, 2=FILE)
# (Reloadable: No)
communication type = 1
**************
############ Specific Section for HTTP communication type #########
*******
#
             SERVER PARAMETERS
#
# Preferred Server
# (Reloadable: No)
server = mu.itsc.austin.ibm.com
# Server Port
# (Reloadable: No)
```

```
port = 80
# Servlet Path
# (Reloadable: No)
path = /slmruntime/service
# Use Proxy [y/n]
# (Reloadable: No)
use proxy = n
# Proxy
# (Reloadable: No)
proxy = none
# Proxy Port
# (Reloadable: No)
proxy port = 3128
# Secure Port
# (Reloadable: No)
secure port = 443
# Security Level (0 none, 1 ssl, 2 ssl client)
# (Reloadable: No)
security level = 0
# Fips Enabled [y/n]
# (Reloadable: No)
fips_enabled = y
#
               AGENT PARAMETERS
#
# Local Tcp-ip port
# (Reloadable: No)
tcpip port = 1239
# WAS Agent configuration file
# (Reloadable: No)
was_agent_conf_file = /slmruntime/webdoc/agent/wasagent/win32/was.conf
```

Example B-2 on page 279 shows a sample tivhscan.mif file.

Example: B-2 tivhscan.mif file

```
Start Component
Name = "Tivoli Hardware Inventory PC MIF Generator"
Start Group
  Name = "ComponentID"
  Class = "TIVOLI | ComponentID | 1"
  ID = 1
  Start Attribute
      Name = "Manufacturer"
      ID = 1
      Type = String(64)
      Value = "IBM"
   End Attribute
  Start Attribute
      Name = "Product"
     ID = 2
      Type = String(128)
      Value = "-[622149U]-"
   End Attribute
   Start Attribute
      Name = "Version"
      ID = 3
      Type = String(64)
      Value = "IntelR - 42302e31"
   End Attribute
   Start Attribute
      Name = "Serial Number"
     ID = 4
     Type = String(64)
      Value = "KPWWN28"
   End Attribute
   Start Attribute
      Name = "Type"
      ID = 7
      Type = String(32)
     Value = "6221"
  End Attribute
End Group
Start Group
  Name = "Processor"
  Class = "TIVOLI|Processor|1"
  Start Attribute
      Name = "Index"
```

ID = 1Type = Integer Value = -1End Attribute Start Attribute Name = "ID" ID = 2Type = String(32)Value = "" End Attribute Start Attribute Name = "Family" ID = 3 Type = String(32)Value = "" End Attribute Start Attribute Name = "MaxClockSpeed" ID = 4Type = Integer Value = -1End Attribute Start Attribute Name = "CurrentClockSpeed" ID = 5Type = Integer Value = -1 End Attribute Start Attribute Name = "ExternalClockSpeed" ID = 6Type = Integer Value = -1End Attribute Start Attribute Name = "Manufacturer" ID = 7 Type = String(128)Value = "" End Attribute Start Attribute Name = "Serial Number" ID = 8Type = String(64) Value = ""

```
End Attribute
Start Attribute
  Name = "Interface"
  ID = 9
  Type = Start ENUM
  Name = "Processor-Interface"
  0 = ""
  1 = "Other"
  2 = "Unknown"
  3 = "Daughterboard"
  4 = "ZIF Socket"
  5 = "Replaceable Piggyback"
  6 = "None"
  7 = "LIF Socket"
  8 = "Slot 1"
  9 = "Slot 2"
  10 = "370-pin socket"
  11 = "Slot A"
  12 = "Slot M"
  13 = "Socket 423"
  14 = "Socket A (Socket 462)"
  15 = "Socket 478"
  16 = "Socket 754"
  17 = "Socket 940"
  End ENUM
  Value = -1
End Attribute
Start Attribute
  Name = "Chip Family"
  ID = 10
  Type = Integer
  Value = -1
End Attribute
Start Attribute
  Name = "Chip Model"
  ID = 11
  Type = Integer
  Value = -1
End Attribute
Start Attribute
  Name = "Chip Stepping"
  ID = 12
  Type = Integer
  Value = -1
End Attribute
```

```
Start Attribute
   Name = "Chip Features"
   ID = 13
  Type = Integer
   Value = -1
End Attribute
Start Attribute
   Name = "Floating Point Unit Present"
   ID = 14
  Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Virtual Mode Extensions"
  ID = 15
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
  Name = "Debug Extension Present"
   ID = 16
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Page Size Extensions"
   ID = 17
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Time Stamp Counter"
   ID = 18
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Model Specific Registers"
   ID = 19
   Type = String(1)
  Value = ""
End Attribute
Start Attribute
   Name = "Physical Address Extensions"
   ID = 20
```

```
Type = String(1)
  Value = ""
End Attribute
Start Attribute
   Name = "Machine Check Exceptions"
   ID = 21
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "CMPXCHG8B Instruction Support"
   ID = 22
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "On chip APIC"
   ID = 23
   Type = String(1)
  Value = ""
End Attribute
Start Attribute
  Name = "Fast System Call
   ID = 24
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Memory Type Range Registers"
   ID = 25
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Page Global Enable"
   ID = 26
  Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Machine Check Architecture"
   ID = 27
   Type = String(1)
   Value = ""
End Attribute
```

```
Start Attribute
   Name = "Conditional Move Instruction"
   ID = 28
  Type = String(1)
  Value = ""
End Attribute
Start Attribute
   Name = "Page Attribute Table"
   ID = 29
  Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "36-bit Page Size Extension"
   ID = 30
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Process Serial Number Enabled"
   ID = 31
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "MMX(tm) Techonlogy"
   ID = 32
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
   Name = "Fast Floating Point Save/Restore"
   ID = 33
   Type = String(1)
   Value = ""
End Attribute
Start Attribute
  Name = "Streaming SIMD Extensions"
   ID = 34
   Type = String(1)
  Value = ""
End Attribute
Start Attribute
   Name = "AMD 3DNow!(tm) Technology"
   ID = 35
```

```
Type = String(1)
     Value = ""
  End Attribute
Key = 1,2
End Group
Start Table
  Name = "Processor Table"
  ID = 2
  Class = "TIVOLI Processor 1"
{1,"67da57f3eb61c19e66a045d79ab3b50e","Intel
Xeon",3066,3066,133,"GenuineIntel","",4,15,2,9,1072430079,"Y","Y","Y","
"Y"}
{2,"67da57f3eb61c19e66a045d79ab3b50e","Intel
Xeon",3066,3066,133,"GenuineIntel","",4,15,2,9,1072430079,"Y","Y","Y","
"Y"}
End Table
Start Group
  Name = "PhysicalProcessor"
  Class = "TIVOLI PhysicalProcessor 1"
  Start Attribute
     Name = "Id"
     ID = 1
     Type = String(64)
     Value = ""
  End Attribute
  Start Attribute
    Name = "Core Per Package Count"
     ID = 2
     Type = Integer
     Value = -1
  End Attribute
  Start Attribute
     Name = "Logical Proc Per Core"
     ID = 3
    Type = Integer
     Value = -1
  End Attribute
  Start Attribute
     Name = "Manufacturer"
     ID = 4
     Type = String(64)
```

```
Value = ""
End Attribute
Start Attribute
   Name = "Family"
   ID = 5
   Type = String(64)
   Value = ""
End Attribute
Start Attribute
  Name = "Type"
   ID = 6
   Type = String(64)
   Value = ""
End Attribute
Start Attribute
   Name = "CpuFreq"
   ID = 7
   Type = Integer
   Value = -1
End Attribute
Start Attribute
  Name = "L2 Cache Size'
   ID = 8
   Type = Integer
   Value = -1
End Attribute
Start Attribute
   Name = "L3 Cache Size"
   ID = 9
   Type = Integer
   Value = -1
End Attribute
Start Attribute
   Name = "Brandname"
   ID = 10
   Type = String(128)
   Value = ""
End Attribute
Start Attribute
   Name = "Signature"
   ID = 11
   Type = String(128)
   Value = ""
End Attribute
Start Attribute
```

```
Name = "Is64BitSupported"
     ID = 12
     Type = String(1)
     Value = ""
  End Attribute
Kev = 1
End Group
Start Table
  Name = "PhysicalProcessor Table"
  ID = 44
  Class = "TIVOLI PhysicalProcessor 1"
{"1",1,1,"Intel","Xeon","3.06GHZ",3066,512,0,"Intel(R) Xeon(TM) CPU
3.06GHz","0x00000F29","N"}
{"2",1,1,"Intel","Xeon","3.06GHZ",3066,512,0,"Intel(R) Xeon(TM) CPU
3.06GHz","0x00000F29","N"}
End Table
Start Group
  Name = "UNIX System Params"
  Class = "TIVOLI UNIX System Params 1"
  ID = 19
   Start Attribute
     Name = "Computer Alias"
     ID = 4
     Type = String(32)
     Value = "MU"
   End Attribute
End Group
Start Group
   Name = "Operating System"
   Class = "TIVOLI Operating System 1"
  ID = 4
   Start Attribute
     Name = "Name"
     ID = 1
     Type = String(128)
     Value = "Microsoft Windows Server 2003 Enterprise Edition English
(United States) version"
   End Attribute
   Start Attribute
     Name = "Type"
     ID = 2
     Type = Start ENUM
```

Name = "Operating System-Type" 1 = "Unknown" 2 = "Other"3 = "MACOS"4 = "ATTUNIX"5 = "DGUX"6 = "DECNT"7 = "Digital Unix" 8 = "OpenVMS" 9 = "HPUX"10 = "AIX"11 = "MVS"12 = "0S400"13 = "0S/2"14 = "JavaVM"15 = "MSDOS"16 = "WIN3x"17 = "WIN95"18 = "WIN98"19 = "WINNT"20 = "WINCE"21 = "NCR3000" 22 = "NetWare" 23 = "OSF"24 = "DC/OS"25 = "Reliant UNIX" 26 = "SCO UnixWare" 27 = "SCO OpenServer" 28 = "Sequent" 29 = "IRIX"30 = "Solaris" 31 = "SunOS"32 = "U6000"33 = "ASERIES"34 = "TandemNSK" 35 = "TandemNT" 36 = "BS2000"37 = "LINUX"38 = "Lynx"39 = "XENIX"40 = "VM/ESA"41 = "Interactive UNIX" 42 = "BSDUNIX"43 = "FreeBSD"44 = "NetBSD"

45 = "GNU Hurd"46 = "0S9"47 = "MACH Kernel" 48 = "Inferno" 49 = "QNX"50 = "EPOC"51 = "IxWorks" 52 = "VxWorks"53 = "MiNT"54 = "BeOS" 55 = "HP MPE"56 = "NextStep" 57 = "PalmPilot" 58 = "Rhapsody"59 = "Windows 2000" 60 = "Dedicated" 61 = "0S/390"62 = "VSE"63 = "TPF"64 = "Windows Millenium" 65 = "Windows XP" 66 = "Windows 2003" 67 = "Linux PPC" 68 = "Windows Vista" 69 = "Windows Longhorn" End ENUM Value = 66End Attribute Start Attribute Name = "Major Version" ID = 3Type = Integer Value = 5End Attribute Start Attribute Name = "Minor Version" ID = 4Type = Integer Value = 2End Attribute Start Attribute Name = "Sub Version" ID = 5Type = String(32)Value = "Build 3790: Service Pack 2"

```
End Attribute
Start Attribute
  Name = "Install Date"
  ID = 6
  Type = String(64)
  Value = "2008-01-29-10.30.30.000000"
End Attribute
Start Attribute
  Name = "Registered Organization"
  ID = 7
  Type = String(255)
  Value = ""
End Attribute
Start Attribute
  Name = "Registered Owner"
  ID = 8
  Type = String(255)
  Value = "mu"
End Attribute
Start Attribute
  Name = "Product ID"
  ID = 9
  Type = String(64)
  Value = "69713-640-3875066-45651"
End Attribute
Start Attribute
  Name = "OS Arch"
  ID = 10
  Type = String(24)
  Value = "x86"
End Attribute
Start Attribute
  Name = "OS Kernel Mode"
  ID = 11
  Type = String(24)
  Value = "32"
End Attribute
Start Attribute
  Name = "OS Language Version"
  ID = 13
  Type = String(64)
  Value = "0409"
End Attribute
Start Attribute
  Name = "OS Locale ID"
```

```
ID = 14
      Type = String(64)
      Value = "0409"
   End Attribute
   Start Attribute
      Name = "Last Boot Up Time"
      ID = 16
      Type = String(64)
      Value = "2008-01-29-15.56.24.000000"
  End Attribute
   Start Attribute
      Name = "Current User Locale ID"
      ID = 17
     Type = String(64)
      Value = "0409"
  End Attribute
End Group
Start Group
  Name = "IP Address"
  Class = "TIVOLI IP Address 1"
   Start Attribute
      Name = "Index"
      ID = 1
      Type = Integer
      Value = -1
  End Attribute
   Start Attribute
      Name = "Address"
      ID = 2
      Type = String(40)
      Value = ""
   End Attribute
   Start Attribute
      Name = "Hostname"
     ID = 3
      Type = String(64)
     Value = ""
   End Attribute
   Start Attribute
      Name = "Domain"
      ID = 4
      Type = String(64)
      Value = ""
   End Attribute
```

```
Start Attribute
     Name = "Subnet"
     ID = 5
     Type = String(40)
     Value = ""
  End Attribute
  Start Attribute
     Name = "Gateway"
     ID = 6
     Type = String(40)
     Value = ""
  End Attribute
  Start Attribute
     Name = "Primary DNS"
     ID = 7
     Type = String(40)
     Value = ""
  End Attribute
  Start Attribute
     Name = "Secondary DNS"
     ID = 8
     Type = String(40)
     Value = ""
  End Attribute
  Start Attribute
     Name = "DHCP Enabled"
     ID = 9
     Type = String(1)
     Value = ""
  End Attribute
  Start Attribute
     Name = "Permanent Mac Address"
     ID = 10
     Type = String(64)
     Value = ""
   End Attribute
  Start Attribute
     Name = "IPV6Address"
     ID = 11
     Type = String(64)
     Value = ""
  End Attribute
Key = 2
End Group
```

```
Start Table
  Name = "IP Address Table"
  ID = 6
  Class = "TIVOLI | IP Address | 1"
{1,"9.3.4.161","mu","itsc.austin.ibm.com","255.255.254.0","9.3.4.1","9.
3.4.2", "9.12.6.7", "Y", "00:11:09:9B:87:C0", ""}
End Table
Start Group
  Name = "Lpar"
  Class = "TIVOLI|Lpar|1"
  ID = 34
   Start Attribute
      Name = "Lpar ID"
      ID = 2
      Type = String(32)
     Value = ""
   End Attribute
   Start Attribute
      Name = "SharedPool ID"
      ID = 3
      Type = String(16)
      Value = ""
   End Attribute
   Start Attribute
      Name = "Node Capacity"
      ID = 4
      Type = String(64)
     Value = "2.000000"
   End Attribute
   Start Attribute
      Name = "Node Capacity In Cores"
      ID = 8
      Type = String(64)
      Value = "2.000000"
   End Attribute
   Start Attribute
     Name = "Lpar Capacity"
      ID = 5
      Type = String(64)
     Value = "-1.000000"
   End Attribute
   Start Attribute
      Name = "Lpar Capacity In Cores"
      ID = 9
```

```
Type = String(64)
     Value = "-1.000000"
  End Attribute
  Start Attribute
     Name = "SharedPool Capacity"
     ID = 6
     Type = String(64)
     Value = "-1.000000"
  End Attribute
  Start Attribute
     Name = "SharedPool Capacity In Cores"
     ID = 10
     Type = String(64)
     Value = "-1.000000"
  End Attribute
  Start Attribute
     Name = "Serial Number"
     ID = 7
     Type = String(64)
     Value = "KPWWN28"
  End Attribute
End Group
```

End Component

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks publications

For information about ordering these publications, see "How to get IBM Redbooks publications" on page 296. Note that some of the documents referenced here might be available in softcopy only:

- Implementation Best Practices for IBM Tivoli License Manager, SG24-7222
- Problem Determination for WebSphere for z/OS, SG24-6880
- ► IBM Tivoli Asset Management for IT Portfolio Overview, SG24-7376

Other publications

These publications are also relevant as further information sources:

- ► IBM Tivoli License Compliance Manager: Quick Start Guide, CF0SBML
- ► IBM Tivoli License Compliance Manager: Overview, SC32-1503
- IBM Tivoli License Compliance Manager: Administration, SC32-1430
- IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration, SC32-1431
- IBM Tivoli License Compliance Manager: Commands, SC32-1501
- IBM Tivoli License Compliance Manager: Data Dictionary, SC32-1432
- IBM Tivoli License Compliance Manager: Problem Determination, SC32-9102
- IBM Tivoli License Compliance Manager: Catalog Management, SC32-1434
- IBM Tivoli License Compliance Manager: Security Management, SC32-1502
- ► IBM Tivoli License Compliance Manager: Release Notes, SC32-1429
- IBM Tivoli License Compliance Manager for z/OS OS Administration Guide for Mainframe Components, SC32-1575

Online resources

These Web sites are also relevant as further information sources:

- Tivoli software information center: Tivoli Asset Management for IT V7.1:
 - http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp
- Tivoli software information center: IBM Tivoli License Compliance Manager V2.3:

http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?top ic=/com.ibm.itlm.doc/welcome.htm

IBM Service Management Software:

http://www.ibm.com/software/tivoli/features/it-serv-mgmt/index.html

IBM IT Service Management documentation:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp

How to get IBM Redbooks publications

You can search for, view, or download IBM Redbooks publications, Redpapers, Technotes, draft publications and additional materials, as well as order hardcopy IBM Redbooks publications, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Α

Access Management 10 admin server 128, 134 administration perspective 35, 159, 163, 190 administration server 123, 125-126, 237, 239-240 core data 123 hostname and database port 142 administrative user ld 96 administrative workstation 95, 105 Agent Controller 69 Altiris 45 Application Server 45-46, 48, 123, 132, 161-163, 221, 227-228 WebSphere profile name 65 Asset Management Configuration management overlaps 43 standard phases 229 Tivoli's process automation engine 77 asset management 40, 43 Asset Management (AM) 3, 8, 15–17, 32, 39, 45-46, 48, 222-224 assets application 40 audit trail 6 authorized asset information 44 auto-discovery tool 27 Availability Management 9-10

В

backup file 149 empty directory 149 base service 46, 77, 84, 225, 228 base URL 165 Basel II 18 basic structure 30 batch jobs 210 BEA 39 benefits 18, 23 best practice 8 best practice 8 Bind distinguished name 63 BIRT 117 Build, Test Release 10 business integration 28 business model 7

С

Capacity Management 9-10 capacity management 23 Carnegie Mellon University 4 catalog manager 126-127, 142, 252 CCMDB 44 Cell name 66 Centennial 45 challenges 5, 19 Change Management 10 chaos 27 charge backs 33 charge-back process 34, 155 CLI command 144 client component 157, 159, 163 COBIT 13 command line 158, 187, 190 Common Inventory Technology (CIT) 122, 141, 151 compliance 6 compliance management 23, 32 configuration item 43 configuration management database 39 Configuration Manager 130 configuration parameter 256-257 configuraton file 242, 244 connectivity test 168, 189, 191 consolidated software inventory (CSI) 203, 205-206 context name 168, 189-190 default value 195 context-driven license assignment 122 Continual Service Improvement 10 contract management 31–32 Contracts 29 converged asset management 19 cost analysis 25 cost avoidance 23

cost tracking 33 Credentials 55

D

data aggregation 158, 161 data aggregation server 217 Database Server 53 Database server 48 DB2 39.45 DB2 Alphablox 165 documentation 166 supported version 172 Demand Management 9 Deployment Manager node name 66 deployment plan 54, 73, 76 log files 76 primary log file 76 processing order 76 Deployment, Decommission and Transfer 10 DHCP environment 169, 189 directory path 170, 176 directory permissions 176, 185 Directory Server 53 Directory server 48 disconnected processes 20 distiller flow 212 distributed client component distributed server components 168, 188 Installation directory 193 successful installation 164, 199 distributed server component successful installation 164

Ε

E-mail Address 239 enterprise application 165, 223–224 error handling 166, 174, 188 e-Sourcing Capability Model 4 evolution 21 exporter flow 214 exporting data 213 extending and innovating 28 external system 41 extractor 206 extractor flow 205

F

Financial Management 9 financial management 22, 31 financial risk 19 Financials 29 Firewalls 123 FixPack 1 148, 242, 244, 252 FixPack 2 129, 144, 146, 148, 237, 248, 256 catalog manager 148 product.xml file 148 FixPack timestamp 147, 149 FMID 207

G

generating report 211 generating reports 211 governance 6 graphical user interface (GUI) 157, 160, 169 gskit 140–141, 151

H,

Health Insurance Portability and Accountability Act 6 Health Insurance Portability and Accountability Act (HIPAA) 6 host name alias 169, 182–184, 191, 194 preferred value 169, 188 value 78, 188 host platform 122 compliance support services 122 HTTP port 67 HTTP request 124, 165

L

IBM Agent Controller 69
IBM HTTP Server 67–68, 132, 134 plug-ins 132
IBM Redbooks 222 software platforms 222
IBM Service Management (ISM) 3, 6–7
IBM Tivoli Asset Management 23, 31, 45, 48, 121, 144 base services configuration 48 Configuraiton Manager 130 Directory Server 59–60 Directory Server administrator 59

Directory Server data 61 Directory Server database instance 61 Directory Server organization 59 Directory Server organizational unit 59 IT Asset Management portfolio 3 License Compliance Manager 30, 35, 121, 123-124, 128, 136, 155, 157, 160, 175-176 License Manager 122–124 Unified Process 12 IBM Tivoli base services configuration 84 IBM Tivoli Directory Database Instance 61 IBM Tivoli Directory Server 48, 53, 59 IBM Tivoli License Compliance Manager product 31 IBM Tivoli License Manager administrator 128 environment 128 IBM Tivoli License Manager (ITLM) 275 IBM Tivoli License Manager for IBM Software Package 127 Identifier 207, 211 Improvement Planning 11 Incident Management 10 Information Security Management 9–10 Information Technology Infrastructure Library 12 Information Technology Infrastructure Library (see also ITIL) 7 infrastructure complexity 5 Install location 64 installation Wizard 157, 159-160 installLocation 176, 179, 193 Instance name 57 Instance username 57 Integration Composer 44, 46 Inventory 29 inventory application 43 Inventory management 33 Inventory of software data 203 ISO 20000 13 IT Asset Management defined 17 IT assets 23 IT Financial Management 9, 22 IT Service Management 16 why 4 IT service management doal 6 ITAM4IT database instance 57

instance user name 57 ITIL 4.7-8.22 ITLCM application 124 ITLM Administration server 126 ITLM Agent 122-123, 126 code 129 communicate 129 software scans 128 stores data 129 ITLM Agents 129 ITLM Catalog Manager 127 ITLM Master Catalog 126 ITLM Master Catalogue 126–128 ITLM Runtime server 126, 128–129 ITUP 12 **ITUP Composer** 14

J

J2EE 39 J2EE Server 53 J2EE server 48 JAAS entry 165 JMS implementation 99

K

knowledge base 206 Knowledge Management & Service Knowledge System 10

L

Launchpad 49 LDAP 62 LDAP instance 62-63 group suffix 62 LDAP base entity 62 organizational container suffix 62 user suffix 62 legacy renovation 27 License compliance 29 license compliance 18–19, 26, 30, 155, 157, 159, 211 growing complexity 18 license optimization 32 License use 29 licenseAccepted 178-179, 187 life cycle 43 Linux PPC 138, 289

log file 75–77, 200 different types 75 logical partitioning 122 login information 105, 109 LPAR capacity 293

Μ

management discipline 42, 44 Market Intelligence 9 maturity model 26 maverick purchasing 19 Maximo Application Server type panel 92 Maximo Application Server Type 92 Maximo Asset Management foundation 32 License 106 Maximo Database 48.86 panel 110 Maximo database 86-87 process solution content 105 Maximo Enterprise Asset Management 31 Measurement & Control 11 Media Access Control (MAC) 253 meta data 170. 174 Microsoft SMS 45 middleware 46, 48-49, 53, 95, 222 middleware image 70-71 Middleware installation 48 middleware installation 45 image 70, 72 program file 52 middleware installer 48-49, 52, 132 monitor 211 monitor flow 210 MRO 31

Ν

network device 44 non-administrative user 167 Client components 167

0

operating system 76, 89, 94, 130, 132, 221, 240, 243, 252 major version 254 minor version 254 operational management products 8 Oracle 39, 45 organization need 15 Organizational unit 59 OS Locale Id 290

Ρ

param name 133 Performance and Risk Evaluation 10 planning for upgrades 32 plug-in information 240, 243 Port 61 port number 60-62, 134, 164, 168, 173 portfolio 31 preface xv proactive 28 proactive management 28 Problem Management 10 process efficiency 27 Process Reference Model for IT 4, 12 process solution package overview 104 procurement 31, 33 Procurement management 42 procurement management 122 product installation 126 profile name 65, 68, 96, 168, 172–173 project manager 225, 229-230 purchasing 33

R

Rational Method Composer (RMC) 14 reactive 27 Redbooks Web site 296 Contact us xviii Reference identification file (RIF) 207–208 release management 23 report administration 115 reporter flow 213 Request Fulfillment 10 request management 33 response file 157, 159-160, 162, 168 host name 197 installation parameter 186, 198 installation wizard 186, 198 return on investment (roi) 24 RIF 207-208 Risk Evaluation 10

Risk Management 9 Risk mitigation 26 runtime server 123–124, 126, 240–241, 243 address 131 database 126, 152, 240, 243 division table 152 level 152 name 131 Unique identifier 243 web page runnng 130

S

Sarbanes-Oxley 15, 26 Sarbanes-Oxley Act 6 SCCUT 207-208 SCRT 215 secure socket layer (SSL) 224 selector flow 215 self-service 33 server based reports 212 server component 144, 157-158, 160, 162-163, 169 server image 149 serverConfig.prof ileName 179, 187 service 9,28 Service Assessment & Analysis 11 Service Asset & Configuration Management 10 Service Catalog Management 9-10 Service Continuity Management 9–10 Service Design 9 Service Level Management 9–11 Service management 43 service management 3-4, 6, 16, 21-22, 224 fundamental characteristics 4 Service Measurement 11 Service Operation 10 Service Portfolio Management 9 Service Release & Deployment Planning 10 Service Release, Acceptance, Test & Pilot 10 Service Strategy 9 Service Transition 10 Signature Software Catalog 207, 209 silent installation 159-160, 178, 187 response file 159 SMF 211.214 SMP/E 203 software asset 17, 20, 23, 121-122, 218 careful monitoring 25

established business practices 28 poor control 20 Software Asset Management (SAM) 17, 21 Software Asset Management maturity model 26 software catalog 206 software catalogue 128, 142, 206-208 new editions 207 software cost analysis 25 software data 203, 206–207 software inventory 29-30, 34, 155-156, 203 data 203-205 software license 20, 23, 32, 218 financial management 33 software product 34, 122, 128-129, 203-206, 245, 250 external name 208 load modules 204 software signature 128 various elements 207 Software Requirements Specification (SRS) 229 software usage metering 122 Software use 29 software use activity data 210-211 data 203, 209, 211 file 211 software use data 203, 209, 211 solution architecture 30 solution components 28 SQLServer 39, 45 started tasks 210 stores detail 239, 245, 252 stores information 248, 251-252 Strategy Generation 9 Sub-Capacity Reporting Tool (SCRT) 215-216 substep 77 success factors 11 Supplier & Contract Management 9–10 surveyor 206 surveyor flow 204 system management facility data sets 214

Т

TACC 35 TAM4IT Developer 226 TAM4IT installation program 77, 81, 87 TAM4IT solution package installation 104 technology limitations and evolution 21 Testing Acquire 10 text description 257 Tivoli Asset Compliance Center 35, 216 Administration perspective 193–194 client logon 192, 195–196 component 158, 190 flow 216 server 169, 188-189 server subnet 169, 189 WebSphere Application Server name 173 WebSphere Application Server profile name 173 Tivoli Asset Compliance Center (TACC) 35, 158, 163, 169, 217 Tivoli Asset Management 23, 30-31, 39, 46, 48, 222.225 Procurement capabilities 33 Tivoli Common Inventory Technology 122 Tivoli License Compliance Manager 34 agent 246 downstream component 204 product 31 runtime server 137 Tivoli License Compliance Manager (TLCM) 31, 34, 121, 123, 136, 155, 157–158, 160–161, 175-176, 239, 246 Tivoli License Compliance Manager for z/OS 34 Tivoli License Manager agents.ID 239 runtime server 269 Tivoli middleware installer 52, 55, 72, 75, 77 first invocation 75 log file 75 recent invocation 75 second invocation 75 user interface 75 v checking 76 workspace 52 Tivoli Provisioning Manager 45 Tivoli's process automation engine 39, 77 Tivoli's process automation engine security 40 TLCM 34 TLCMz 34 trace file 146, 174, 188, 192, 277 transformation 28 TSO sessions 210

U

unique identifier 238–239, 241 Universal Time, Coordinated (UTC) 250 User Acceptance Testing (UAT) 233 User interface 75–76, 157–158, 160, 228 user-defined link 249 USRPDATA 207 USRPDATA statement 208 USRPDTA 207

V

valid value 177–178, 194 value creation 28 virtualization support 122 VM layer 247 Unique identifier 247

W

WebLogic 39 Weblogic 45 WebSphere 39 WebSphere Application Server 62-63, 162-163, 165 computer 168 context name 191, 195, 197 data layer 170 following configuration parameters 62-64 host name 190-192 information 172 instance 165 location 170 name 172-173, 179 plugin 68 profile name 172-173, 179 security 63 supported version 172 Version 64 WebSphere Application Server Configuration 65, 97 work management 43 workflow engine 43 workflow feature 42 Workspace 52

Х

xml version 145-146, 148

Z z/OS load libraries 204 z/OS UNIX data 203 directory 204, 207 environment 204 product 205 software 203 z/OS UNIX directories 204 z/OS work 210 Software use recording 210




Deployment Guide Series: Tivoli IT Asset Management





Deployment Guide Series: Tivoli IT Asset Management Portfolio



Learn about the IBM Tivoli IT Asset Management portfolio of products

Prepare to install asset management products

Plan for a deployment

This deployment guide provides information related to the deployment of the products that make up the IBM Tivoli IT Asset Management portfolio of products. Thes products include IBM Tvioli Asset Management for IT, IBM Tivoli License Compliance Manager, and IBM Tivoli License Compliance Manager for z/OS.

The Asset Management for IT product is based on the Maximo Asset Management product for managing enterprise assets. Specific functions are provided that are targeted to the needs of IT asset managers. The Tivoli License Compliance Manager and Tivoli License Compliance Manager for z/OS are complementary products that specifically address the discovery and management of the software being used in an enterprise and the management of the associated licenses.

In the first part of this book, we introduce IBM Service Management and describe how IT asset management is a critical component of service management. We then provide more detailed information about each of the three products and guidance about their installation and deployment. Finally, we provide a high level overview of how to plan for an engagement for the deployment of an IT asset management solution.

This book is for people interested in IT asset management.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information: ibm.com/redbooks

SG24-7602-00

ISBN 0738431168