IBM

Deployment Guide Series: Maximo Asset Management V7.1

Learn about Maximo Asset Management V7.1

Plan for a deployment

Study sample usage scenarios
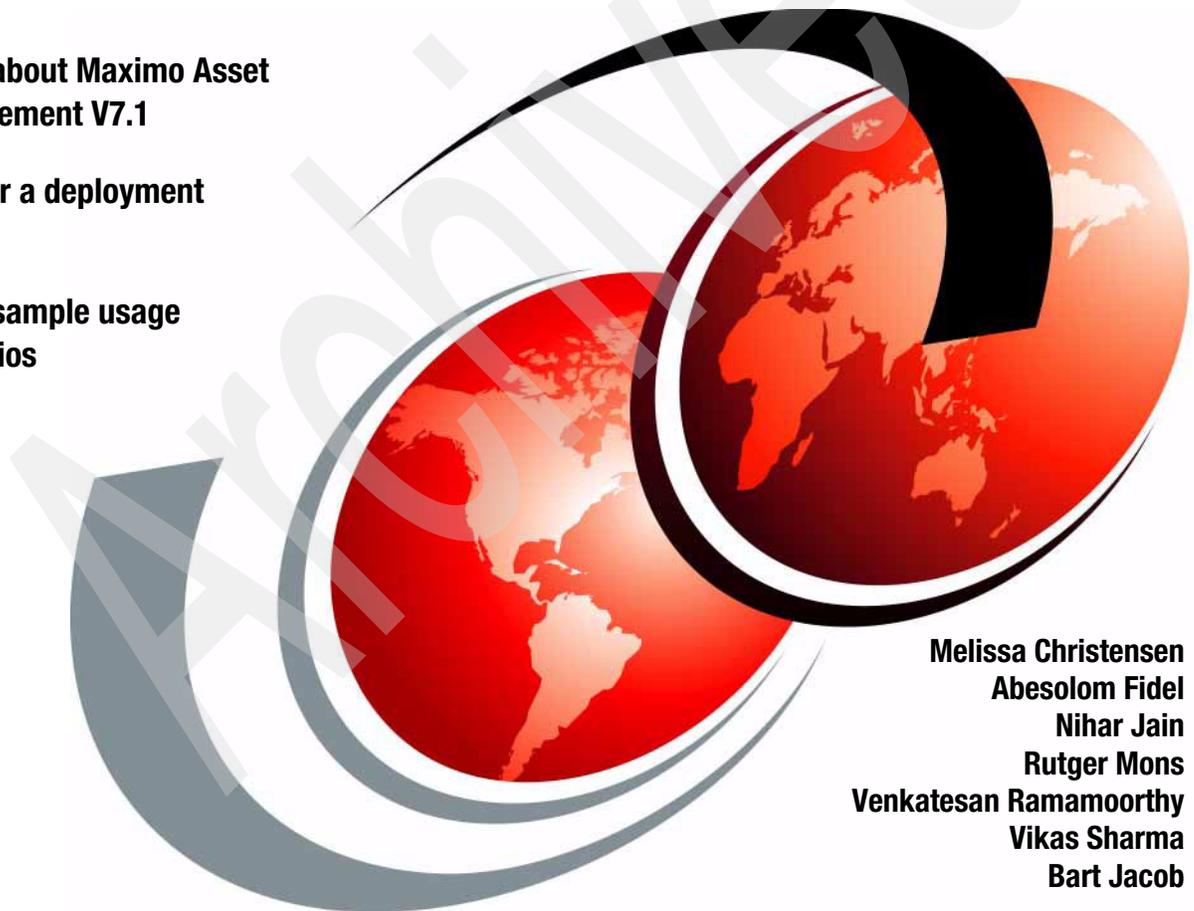
Melissa Christensen
Abesolom Fidel
Nihar Jain
Rutger Mons
Venkatesan Ramamoorthy
Vikas Sharma
Bart Jacob

Redbooks

IBM

International Technical Support Organization

Deployment Guide Series:
Maximo Asset Management V7.1

August 2008

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

First Edition (August 2008)

This edition applies to Version 7, Release 1 of IBM Maximo Asset Management.

Contents

Figures

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.

xiii

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 5L™	Maximo®	System z™
AIX®	OS/400®	Tivoli®
DB2®	PowerPC®	WebSphere®
Enterprise Asset Management®	Rational®	z/OS®
IBM®	Redbooks®	zSeries®
iSeries®	Redbooks (logo) ®	

The following terms are trademarks of other companies:

Acrobat, Adobe, 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.

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.

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

Active Directory, Expression, 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.

Intel, Pentium, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries 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 IBM® Redbooks® publication is for IBM Business Partners, field personnel, and clients who want to plan a deployment of IBM Maximo® Asset Management V7.1. We provide information to assist in the deployment and initial configuration of the environment, but we do not cover general product usage. This book assumes the reader has access to product documentation and some knowledge of previous versions of Maximo Asset Management or other Maximo products that utilize the same underlying framework.

Our intention is not to replace or reproduce the product documentation, but rather to provide useful, supplementary information to assist the reader in understanding deployment planning and processes for Maximo Asset Management. Some content in this book provides information helpful to those who are just becoming familiar with the Maximo product set. In other sections, we assume the reader has deeper technical knowledge and experience.

In most solution deployments, a team of individuals with different skill levels is required to successfully plan and deploy Maximo Asset Management. For various members of the deployment team, different sections of this book are valuable resources for performing their tasks.

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.

Figure 1   (From l to r) Abby Fidel, Venkat Ram, Rutger Mons, Melissa Christensen, Nihar Jain, Vikas Sharma, Bart Jacob

Melissa Christensen is a Senior Project Manager for EMA, based in California. Melissa has over 11 years of experience in project management, specifically implementing Tivoli® Maximo. Melissa holds degrees in anthropology, history, and liberal studies. Her areas of expertise include utilities and facilities maintenance management in both the private and public sectors.

Abesolom "Abby" Fidel is certified in SAP® Production Process with over 11 years experience in implementing enterprise solutions on UNIX® and Linux® platforms. Abby is currently employed as Enterprise Solution Consultant for Avenue Business IT Solutions based out of Ho Chi Minh City, Vietnam. He holds a degree in physics and mathematics.

Nihar Jain is a Techno Functional Consultant with Birlasoft Ltd. in India. He has over 3 years of experience with Maximo in asset and service management. He holds a degree in computers and certifications in ITIL® and IBM Maximo deployment.

Rutger Mons is a IBM-certified Tivoli Deployment Consultant based at Stork Asset Management Solutions in the Benelux region.He has close to 9 years of functional and technical experience in the Maximo field. He holds a diploma in financial management and has various certifications in project management and programming. He is currently completing a degree in informatics.

Venkatesan Ramamoorthy is an IBM-certified solutions expert with years of client implementation experience inside and outside of IBM. He is currently a next-generation IT Service Management Solution Leader in IBM Tivoli. He is ITIL certified and works closely with clients and globally distributed Maximo teams in architecting and developing service management solutions. He holds several patents in his area of expertise.

Vikas Sharma is a Solutions Consultant for Birlasoft Inc., in the U.S. He has approximately 5 years of experience primarily in designing complex and innovative IT solutions mostly in the fields of Enterprise Asset Management® and IT asset and service management. He holds a master's degree in computer applications from IP University, India. His areas of expertise include solution designing, project management, and client relationship management.

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 master's degree in numerical analysis from Syracuse University.

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

Don Busiek, IBM
Dave Calvert, IBM
Tom Davis, IBM
Anthony Honaker, IBM
Sharad Joshi, Birlasoft
Alfons Kemper, Stork - AMS
Michael Kuijl, Stork - AMS
Dean La Porte, IBM
Carlos Marin, IBM
Dan Matlis, Axendia Inc.
Colleen McCretton, IBM
John Pasini, EMA Inc.
Kumar Rampura, Birlasoft
John Reeve, TAIC
Eyco Rogge, Avenue Business IT Solutions
Tom Sarasin, IBM
Mohamed Skandaji, Groupe Createch
Sampath Sriramadhesikan, IBM
Vikas Srivastava, Birlasoft

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 customer 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 in one of the following ways:

► Use the online **Contact us** review Redbooks 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

Introduction

1

1
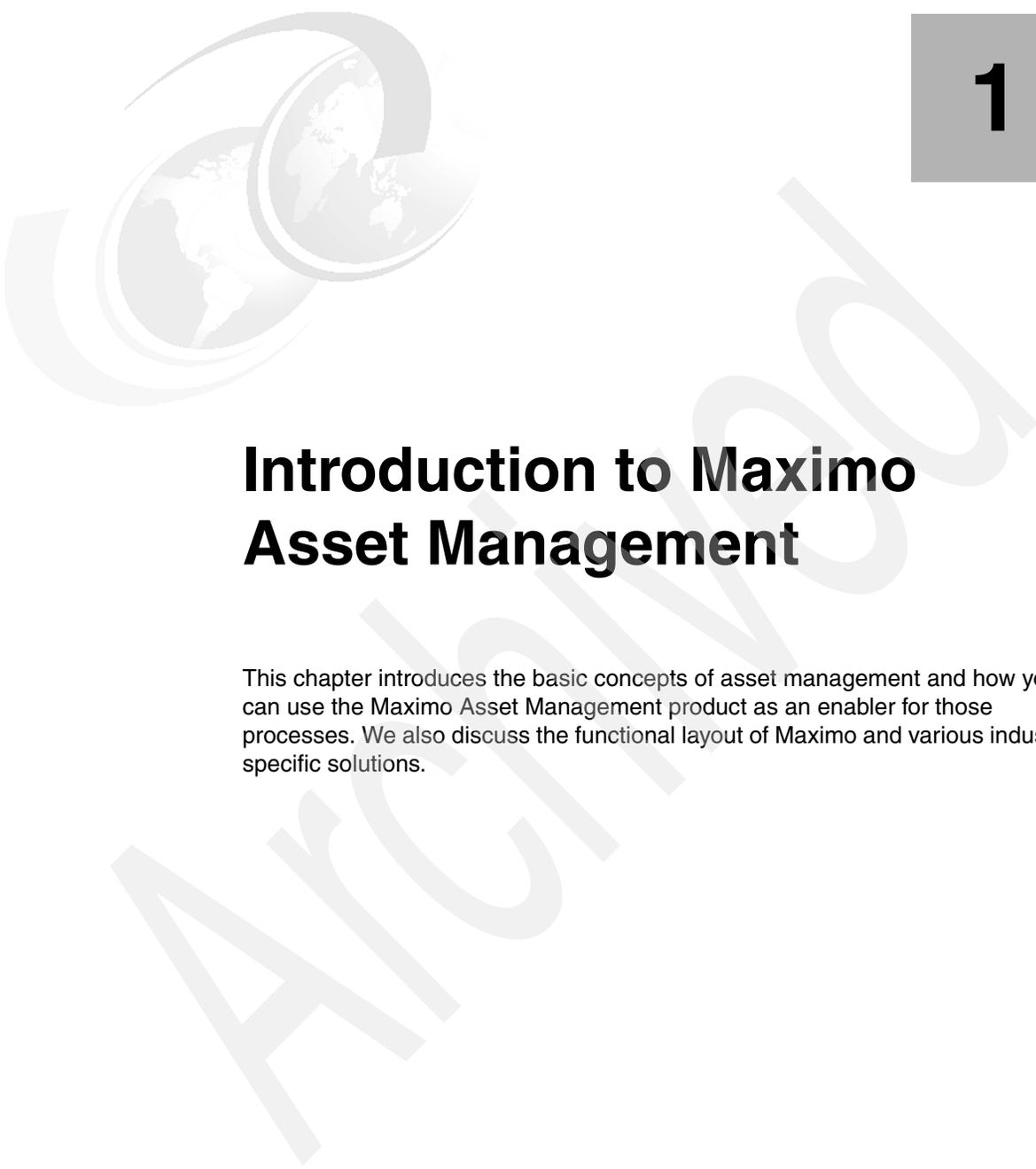
Introduction to Maximo Asset Management

This chapter introduces the basic concepts of asset management and how you can use the Maximo Asset Management product as an enabler for those processes. We also discuss the functional layout of Maximo and various industry specific solutions.

3

1.1 Enterprise asset management overview

Enterprise asset management is the process of managing an organization's assets throughout their life cycle - from requisition to retirement. The word *enterprise* refers to multiple organizations, departments, and functions, and in some cases business units. *Assets* in this context is defined as any equipment or facility that plays a key role in the core business of the enterprise.*Management* refers to improving the utilization and performance of an asset and thereby improving return on investment (ROI).

The asset life cycle typically contains the following phases:

► **Requisition**

At this stage someone or some function in an organization requests an asset. The asset can be procured from a vendor, or it can be an inter-departmental transfer.

► **Procurement**

The complete purchasing process is the second stage in the asset life cycle. This process includes creating purchase requisitions, requests for quotation, and invoices; generating purchase orders, receiving assets at locations, and creating an invoice.

► **Commissioning**

At this stage of the asset life cycle, the asset is assigned to a location, user or group of users. The cost of operating the asset is now attributed to the owner of the asset. Also, asset availability is now the responsibility of the owner. This includes maintaining the asset regularly to ensure the asset is up and running whenever the business needs it. The various activities that are typically undertaken at this point are work orders, job plans, safety planning, preventive maintenance, emergency maintenance, and seasonal maintenance.

► **Retirement**

Some enterprises call this phase *decommissioning*; this is the stage where an enterprise decides to replace or remove the asset. However, the importance of this stage from an enterprise asset management perspective is to be able to calculate the residual cost of the asset and, in certain cases, being able to transfer parts of the asset to another asset. For instance, say few months ago someone replaced a critical and expensive part of an asset because of a fault. The expensive part can be re-used because it is not old.

The concept of Enterprise Asset Management evolved from Computerized Maintenance Management System (CMMS). An enterprise's assets are critical, and with global competition, it is imperative for companies to strategically plan

and maintain various stages of the asset life cycle. Enterprise asset management should be a dedicated function within an organization that encompasses purchasing, engineering, accounting, and maintenance management.

Broadly, the benefits of enterprise asset management can be summarized as:

► Reducing costs
► Meeting legal requirements
► Enabling better planning
► Providing competitive advantage
► Enhancing information availability

1.2 Maximo Asset Management

IBM Maximo Asset Management is an enterprise asset management system that provides asset management, maintenance management, inventory management, and purchasing capabilities that enable corporations to maximize productivity and increase the life of revenue-generating assets.

Using Maximo, companies can manage assets by providing information and real-time data, thereby enabling the creation of a strategy for maintenance management through information-based decision-making capabilities and predicting the impact on productivity of asset downtime for all categories of assets - production equipment, facilities, and transportation assets.

1.2.1 Functional overview

IBM Maximo Asset Management is an integrated productivity tool and database that helps you achieve maximum efficiency in asset management by managing all of your asset types on a single software platform. Built on a service-oriented architecture (SOA), Maximo Asset Management delivers a comprehensive view of all asset types, their conditions, locations, and the work processes that support them to provide optimal planning, control, audit, and compliance capability.

The Maximo database provides critical information about asset resources, including key attributes, their configuration, and their physical and logical relationships to other resources. Through the Maximo Asset Management user interface, you can establish key performance indicators (KPIs) to continuously monitor asset conditions and trigger automated action based on changes. You can create, assign, monitor, notify, and report on key process components such as work orders, service desk tickets, and purchase orders, including status. You

can also include attachments, such as maps, pictures, and URLs to each record or task to further enhance communication and productivity.

IBM Maximo Asset Management enables you to effectively manage end-to-end asset operations and business processes to deliver efficient and effective services aligned with your business goals. It provides a comprehensive and modular approach to integrated asset control and visibility by providing an enterprise platform for storing standardized data on asset histories to help integrate people, processes, information, and technology. Maximo Asset Management offerings include tools, best practices, and services for incremental value.

Managing assets through Maximo Asset Management gives your business a competitive advantage.

1.2.2 Modules

Maximo Asset Management is composed of modules that in turn contain applications or subapplications. Six of these key management modules are:

- ► Assets
- ► Work
- ► Service
- ► Contracts
- ► Materials
- ► Procurement Management

Together, these modules help you capture and analyze your asset and work data and help you optimize maintenance and service initiatives throughout your enterprise. These six management modules are packaged in an enhanced service oriented architecture that helps simplify the creation of Web services and supports additional Web service standards such as WS-Security.

Built on the Java™ platform, Enterprise Edition (J2EE™) component-based Internet architecture, Maximo Asset Management fits any modern enterprise technology infrastructure and integrates easily with your business systems. With a significant focus on configuration tooling - such as Application Designer for modifying the user interface and Database Configuration for adding new tables and columns - Maximo Asset Management makes it easy to tailor the software to your needs without programming. Furthermore, you can use the Upgrade Utility to upgrade your configurations, and thus avoid being locked in from one application release to another.

The key modules and the business goals for which you can use them are described in the sections that follow.

Asset Management

The Assets module enables you to achieve the control you need to more efficiently track and manage asset and location data throughout the asset life cycle.

► Track asset detail - including location, work, cost and other attributes and their histories - over time to help maximize productivity and extend asset life.

► Establish location and asset hierarchies to roll up costs across systems, subsystems, departments, and locations, enabling a better understanding of the true cost of assets (initial cost, financial value, cost to maintain, and so on).

► Monitor asset and location conditions to enable proactive - rather than reactive - maintenance that helps reduce unplanned downtime.

► Support both conventional and linear assets.

Work Management

Manage both planned and unplanned maintenance activities, from initial work request and work order generation through completion and recording of actuals. Work planners can match job tasks to available resources, estimate and obtain approval of costs, establish priorities, and initiate maintenance activities across the enterprise. Work management supports your business goals by providing the following benefits:

► Tracking tools enable more detailed analysis of resources, inventory, and equipment use and costs, helping decrease labor and materials costs.

► Multiple assets, locations, and configuration items are allowed per work order or ticket. Work management also supports work order tracking, task sequencing, attaching documents at the task level, and generating work orders from asset information.

► A graphical assignment manager helps optimize maintenance schedules and labor use by assigning the right person with the right skills to the right job.

► Preventive maintenance (PM) functionality enables you to put PM schedules in place with the right job steps and resource requirements, facilitating planning as well as work, to help reduce unplanned downtime and reactive maintenance.

► Newly developed work management tools enable job plan hierarchies, automated workflow processes, enhanced status control, and support for linear assets.

► Contract correlation links SLAs to vendor contracts, helping you identify unreliable vendors as well as low-quality products. It also enables you to reference service-level agreement (SLA) performance metrics when renegotiating vendor terms.

- ► A terms and conditions library enables you to more consistently apply standardized policies across the organization.
- ► Automatic notifications and alerts help you meet vendor terms, avoid penalties, and get the most value out of every contract.

Service Management

Service request support enables users to submit new service requests, as well as to track and update open service requests. Apply additional service management best practices through IBM Tivoli Service Desk and further align asset management priorities with overall business objectives. By infusing comprehensive service-level management into your asset management practice, you can:

- ► Define service offerings to help improve organizational communication and verify that the services provided are those required to support the business.
- ► Establish service-level agreements (SLAs) to help increase communication between your organization and the business units, and help align service levels with business objectives.
- ► Proactively monitor service-level delivery against metrics to avoid missing service-level commitments.
- ► Implement escalation procedures to better manage the resources that support service-level commitments.

Contract Management

Enhanced control over vendor contracts is provided with this integrated contract management system. Provide comprehensive contract management support for purchase, lease, rental, warranty, labor rate, software, master, blanket, and user-defined contracts.

Materials Management

Know the details - what, when, where, how many, and how valuable - of asset-related inventory and its usage. Materials Management functionality records material movements and adjustments, enabling real-time inventory tracking, reporting, and auditing. In addition, you can display the embedded images of an asset in a catalog search. You can also use this module to:

- ► Track inventory transactions to help streamline parts and materials management.
- ► Help decrease costs by eliminating excess or obsolete inventory.
- ► Help optimize and plan inventory to more accurately meet maintenance demand, making the right parts available at the right location when needed.

As a result, you can reduce stock-outs, inventory shrinkage, and carrying costs, as well as help foster economies of scale through shared resources.

Procurement Management

The Procurement Management module supports the phases of enterprisewide procurement, including direct purchasing and inventory replenishment. You can provide buyers with more extensive requisition, quotation, vendor, purchase order, and contract capabilities, thereby enabling them to plan work more proactively. Maximo Asset Management easily integrates with enterprise business systems from vendors such as Oracle® and SAP, and connects to online marketplaces, supplier systems, and exchanges. This module provides:

► Vendor management and vendor performance analysis tools that can help reduce costly off-contract buying and help verify the reliability of vendors and the quality of inventory and services.

► Automated interval-based, meter-based, or event-driven purchasing capabilities to help you order the right parts and services at the right time, which improves purchasing efficiency.

► Global purchasing support that can enable group purchasing savings and efficiencies, and can help lower sourcing costs.

► Analysis tools and key performance indicators (KPIs) that measure procurement performance such as order processing times, invoice accuracy, and order delivery times.

1.2.3 Industry-specific solutions

The following are industry-specific solution options available with Maximo:

► IBM Maximo Asset Management
► IBM Tivoli Asset Management for IT
► IBM Tivoli Service Request Manager
► IBM Maximo for Government
► IBM Maximo for Service Providers
► IBM Maximo for Nuclear Power
► IBM Maximo for Life Sciences
► IBM Maximo for Oil and Gas
► IBM Maximo for Utilities
► IBM Maximo for Transportation

Each of these solution is configured for a specific industry and its type of assets. The windows, information capturing, and data flow between various modules is specific to that industry. Apart from industry-specific solutions, add-on applications or adapters are available that provide enhanced functionality and add value to the overall solution.

For example, the Maximo for Transportation solution has vehicle and fleet management functionality. This includes specific fields that capture vehicle reporting data such as miles, last maintenance, and oil change details, and fuel information. The integration of advanced features such as global positioning system (GPS) and radio frequency identification (RFID) further enhances vehicle traceability and prevents inventory losses. Some of the built-in vehicle diagnostics enable you to proactively schedule maintenance rather than reacting to breakdowns.

The other industry-specific solutions have built-in capabilities within their respective domains.

1.3 Integrating with existing enterprise applications

From a functional standpoint, most businesses already have huge IT investments. Installing Maximo does not mean these businesses have to get rid of existing applications. Maximo offers a robust integration platform to integrate with any existing applications and maintain a "single version of truth."

Some features that enable integration with leading enterprise systems are already available in the form of adapters, while some have to be custom built. Maximo is a Java-based solution with an open architecture, and therefore custom building integrations is not a daunting task.

In the sections that follow, we briefly discuss the various functional areas where asset management typically integrates with the enterprise system, and we list some of the tools we use for integration. More details about how integration can be accomplished is covered in Chapter 7, "Integration Framework" on page 157.

1.3.1 Functional integration points

It is important to note that generalization of integration scenarios cannot be done. Each organization is unique with its own IT standards, systems, and requirements. Based on our experience from previous implementations and from interacting with various organizations that plan to implement Maximo Asset Management, integration demands with the following functional systems are common:

► **Human Resource Management System**

This integration facilitates resource information mapping to Maximo Asset Management. An HRMS generally has employee information and contact details that may be required for work assignment in Maximo. In addition, HRMS enables you to access hourly rates and salary information to account

for labor costs in Maximo. Most companies have some sort of LDAP for single sign on, which also has employee demographic data. Maximo Asset Management V7.1 comes with prebuilt LDAP integration.

► **Financial System**

Accounting is a sensitive subject in most organizations. The CFO organization typically wants justification and information pertaining to investments and expenses being incurred by other departments. Because of the critical nature of accounting systems, CFOs are often reluctant to migrate to a new system unless something is terribly wrong with the existing finance management product. Typical integrations with Maximo Asset Management simply require the mapping of GL codes and cost centers to the existing financial system. Using Maximo, each asset, and thus its related expenses, are mapped to a cost center that can be mapped to the GL code in the financial application.

► **Procurement Management**

Some companies have mature vendor management processes and use specialist systems for vendor and contract management. The database fields in Maximo Asset Management, at a high level, require company information and company contact information in order to send requests for quotation (RFQs) or purchase orders. Typically the procurement and finance applications are tightly coupled to maintain invoice generation and PO creation in one place.

► **Supply Chain Management Software**

Spare parts management and inventory management are expensive activities. Not only does inventory require space, but if not used for a long time it may even expiration. Enterprises today, use mature supply chain management systems to order Just-in-Time (JIT) inventory management techniques. Maintenance is the biggest consumer of inventory, and thus its tight integration with SCM tools is imperative.

► **Project Management Software**

For resource planning and equipment availability prediction, project management tools such as Microsoft® Project are often integrated with Maximo Asset Management.

2

New features in Maximo Asset Management V7.1

This chapter provides an overview of the new features in Maximo Asset Management V7.1. As noted in this document, a tremendous amount of recently developed Version 7.1 functionality is specific to other Maximo products that run on the same framework as Maximo Asset Management - products such as Maximo Asset Management for IT, IBM Tivoli Change and Configuration Management Database, and IBM Tivoli Service Request Manager.

The following are the three main areas of focus for Maximo Asset Management Version 7.1:

► Continuing the strategy of one platform for key processes

► Lowering both the cost of ownership and the cost of environment migration

► Specific market-driven functionality such as linear assets, enhancements to work and process management, and enhancements to usability.

These enhancements are described in detail in subsequent sections.

2.1 Enhancements in Maximo Asset Management Version 7.1

The features described in this section are the functional enhancements in Maximo V7.1. They are pertinent from a usage perspective, and their addition to the Maximo product is based on client feedback. The aim is to enhance user experience with Maximo. Most of these features are now part of the standard out-of-the-box functionality.

2.1.1 New and enhanced capabilities

The sections that follow describe new or enhanced application functions that have been added to Maximo Asset Management V7.1.

Linear assets

Linear assets are assets that have linear properties and are often connected within a network or system. For example, linear assets may include:

► Roads
► Pipelines
► Railways

The management of linear assets differs from the management of non-linear assets in that non-linear assets occupy a defined space and can often be tracked by their location, or they can be expressed as a part of a parent-child hierarchy. Linear assets have linear properties (that is, pavement type, number of lanes, presence of guardrails) that often change over the length of the asset. So a single linear asset record, such as a highway that spans many miles, can possess characteristics that change over the span, allowing the linear asset to be virtually segmented without impacting the underlying geometry. In addition, relationships may be defined that articulate how linear assets join, cross, run in parallel, or pass over or under one another.

Work can now be carried out against single or multiple segments of a linear asset. For instance, grass cutting can be performed from mile post 10 to mile post 20 on Interstate 95. Clients can also query maintenance that has been performed against a specific segment (that is, a client can ask to see all work done from mile post 11 to mile post 18). This tool is valuable for any operation requiring linear asset functionality.

The linear assets functionality is available as an add-on and is not shipped as a standard feature.

Multiple assets and locations on work orders

Work orders can now include multiple assets and locations without the need for child work orders or tasks. However, Maximo Asset Management V7.1 does not support cost distribution at the asset or location level - the work order still has a single GL account.

Enhanced swap capabilities

With Maximo Asset Management V6, an action in Work Order tracking was introduced to Plan Asset Moves as part of the work process. Upon completion of the work order, the asset move transactions are recorded in Maximo. With Maximo Asset Management V7, this capability is extended to support Planned Asset Swaps. An asset swap is a single transaction that replaces one asset with another. In the past, multiple asset moves were required to perform this action. Once the work order is completed, the asset swap transactions are recorded in Maximo.

Optionally, this single-step swap action can also be performed from the Assets application and does not require a work order to accomplish this task. The swaps are captured within the Asset Move History just as multiple-step location changes have been in the past. The User and Custodian associated with Assets involved in the swap may also be modified at the time of the swap or move (planned or ad hoc).

Automated flow control - status management

Maximo Asset Management V7.1 provides the capability to define precedence connections between work orders and tasks. This then initiates the resulting network of records automatically applied with a finish-to-start action, thus automating the flow of status changes between them. For example, if flow control sequencing is enabled at the Job Plan level, each task generated when applying the Job Plan to a work order evaluates the one before it to determine if it has completed. If the predecessor has completed, its status is automatically changed to "In Progress". Any successive tasks may not have its status changed to "In Progress" until its immediate predecessor has completed. As such, the status on the work order cannot be changed to "Complete" until all tasks have been completed in the order in which they are sequenced. When all tasks are completed, the work order status is automatically changed to "Complete".

New Activities and Tasks application

A new Task application is available that is a subset of Work Order Tracking functionality to support single task-specific work orders.

For example, a user is assigned Task 20 on a work order. This task prompts a laborer to install a new pump on a boiler. When this assignment appears in the laborer's inbox, that individual can click that assignment and access the Task

application. At that point, the laborer may simply select the "start" timer to indicate that work on the task has commenced. Similarly, when the task is complete, the task may be selected in the same manner, and the laborer chooses the "stop" timer. Thus, the system automatically posts the elapsed time against the work order, including the extended cost of the labour. This entire booking is visible on the Resources tab. All windows in this new application have been simplified. Additional information available includes the Plans and Actuals, Related Records, and the Log tabs.

Attached documents at task level

The standard functional capability of attaching documents is applied to the Work Order Tasks and Job Plan Tasks. When displaying the work order, the user views the attachments for the whole work order including the tasks. When at the Task level (accessible from either Work Order or Job Plan), a user now views attachments applicable to the individual task.

Attaching a document to a task occurs in the Detail section of the Task Table window. An attachment icon indicated by a paper clip is visible. All task-attached document functionality remains the same as elsewhere in the Maximo product.

Nested job plans

This functionality allows job plans to be nested within a hierarchical relationship. Users can create a job plan that refers to other job plans. These plans are nested and generate a work order hierarchy when applied to a Work Order (not just to Tasks). Job plans may be nested in as many levels as needed.

For example, when a new pump is purchased and prepared for deployment within the plant, its preparation may include a number of unique steps. The remainder of the work may be exactly the same as the work specified in the 90-day PM. With nested job plans, the 90-day PM Job Plan may now be specified as part of the pre-deployment Job Plan without having to rewrite the entire 90-Day PM plan details. If your operation utilizes complex job plans with hundreds of line items, this feature enables significant time savings. The Apply Job Plan and Create Job Plan from Work Plan actions have also been enhanced to support the new functionality.

Classification-based attributes

With Maximo Asset Management Version 6, Work Orders were enhanced so they could be classified; however, they did not support Classification attributes. With Maximo V7.1, Work Order Classifications are enhanced to support Classification attributes and all the standard functionality that comes with this feature (which is the case with Assets, Locations, and Items). Furthermore, classification functionality was added to Job Plan Solutions and CIs features. Improved overall

classifications usability and a streamlined application are also available in Maximo Asset Management Version 7.

Work order detail

Maximo V7.1 provides a more comprehensive display of the View Work Orders and PM views than that were provided in prior versions of Maximo. View Work Orders is called View Work Details in Maximo V7.1 and is updated to include Tickets of all classes under the Work tab, Preventive Maintenance, Routes, and Collections tabs.

Classification enhancements

The concept of classification was introduced in Maximo Asset Management V6. Enabling Classifications for the objects supporting them involved a tedious check box-selection routine. In Maximo V7.1, the Classification functionality is both enhanced, and made available to more objects (Job Plans, CIs, Service Items, Purchasing Documents, and Items).

The Applies To check boxes have been replaced with a Table window where the objects that a particular Classification applies to may be selected. All objects that support Classifications also support Attributes. The Attributes can be defined for a Classification and are inherited down the Classification hierarchy. Duplicating a Classification includes the duplication of the associated Attributes.

Job Plan template

Another enhancement to the Job Plan application includes the addition of the Job Plan Template field. This field has a domain associated with it, pre-populated with three choices:

► Maintenance
► Activity
► Process

This field enables logical grouping of Job Plans, so that user interfaces may be conditionally altered.

Task and Work Order ownership

Job Plans and Work Orders (along with many other applications) currently support the concept of ownership. In Maximo Asset Management V7.1, Work Order tasks and Job Plan tasks support the Ownership concept and function consistently with other applications possessing the ownership entity.

Future date tolerance

Maximo Asset Management V6 enables a user to report labor to any point in the future. While this may be convenient, it causes problems when a user tries to change the status of the work order prior to the date of the future labor entry.

Maximo V7.1 includes a tolerance that can be set within the Administration Organization settings. Here the administrator can set the number of hours in the future when users may report their labor. Future tolerances ranging from 0-12 hours are accepted.

Route enhancements

Route stops have become child Work Orders in prior versions of Maximo Asset Management. In the Route application in Maximo V7.1, users have the option to specify Route Stops as the following:

► Child work orders
► Entries within the work orders multi-asset, location, or CI table
► Work order tasks

These selections are made at the header level of each route, not at the application level.

Assets

The following subcategories are included in the enhanced Assets application:

► Report downtime

  In earlier releases of Maximo Asset Management, Asset Downtime can be reported only with a work order reference. In Maximo V7.1, users can directly report downtime against an asset in the Assets application without a Work Order reference.

► Manage downtime

  Users can now edit asset downtime history from the Assets application. This enables users to correct any erroneously entered downtime.

► Asset collections

  This new capability enables the grouping of a number of assets into Collections. A Collection can contain Assets from multiple Sites. In the Security Group application the administrator can assign user groups to these collections so that users can have access to certain assets only.

► Issue items from inventory

  In Maximo Asset Management V7.1, a new capability in the Assets application enables users with access to storerooms to issue items directly to

an asset. If the issued item is a rotating asset, the user can optionally choose the rotating asset as a child of the current asset.

Item status

Maximo Asset Management V7.1 adds the ability to manage Items and Inventory based on a robust set of status rules and capabilities. Users can track and manage the life cycle of items from creation to obsolescence at each functional area (Item, Organization, and Inventory). With the introduction of Item Status, item records can be "workflowed" to enforce a custom business process or scenario. Full status functionality has been added for Tools and Service Items as well.

Item images

The Item Master, Tools, and Service Items applications all support the ability to associate images with their main records. This differs from an attachment in that the image is actually associated with the main definition in the database and is viewable within the application from the main tab. These images enable quick visual identification of items when users navigate through the records. In addition, the image can be called from the detail menu of any item field within Maximo.

Work management status flow control

In Maximo Asset Management, an embedded status change workflow has been added to the work order applications. Precedence relationships can be defined between work orders, activities, and tasks. Flow control can then be optionally set in either the Job Plan or Work Order applications to govern the status change rules in a hierarchy and its network of records. Records under flow control are set into progress when their predecessors are completed, improving the timeliness of the ownership of eligible tasks - for example, for notification purposes. Records in a branch of a hierarchy that are complete roll the completion up to the parent of that branch, moving progress to the next eligible phase of the project. Records with uncompleted predecessors may not be manually set into progress, preventing the user from inadvertently starting tasks that are not ready.

Extended attributes with Service Requests

The following list briefly describes new capabilities related to Service Requests in Maximo Asset Management V7.1:

► Self service

Use of extended attributes with the Self Service applications can greatly increase the quality of the information captured from the self-service user by presenting a set of questions related to the issue being reported.

► Service desk power applications

When using a Service Request Power application, you can drive the operator to consistently ask the appropriate questions and capture information in extended attributes that may help resolve the particular issue.

► Self-service users can create service requests, as shown in Figure 2-1.

Figure 2-1   Create Service Request window

Quick insert functionality

New quick insert functionality enables the rapid creation of service requests, incidents, and problems by referencing a Ticket template along with the quick insert action. In previous versions, the quick insert action typically opened an application and inserted a blank record. In Maximo Asset Management V7.1, we take the quick insert function one step further by applying a Ticket template that is referenced as part of the quick insert on the inserted record.

Clients use this new functionality to simplify the interaction a self-service user experiences by providing new quick inserts for the most commonly requested items and issues.

New Bulletin Board functions

The following are new features in the Bulletin Board application:

► Hide viewed messages

A new flag has been added to Bulletin Board messages that indicates whether a message has been viewed by a particular user. By default, the flag is set to N, and the flag is automatically set to Y after a user has opened the details for a particular message. The default value for the Bulletin Board

Portlet Viewed Filter is set to N. Only messages that have not been viewed are presented by default. The user can change the value of the filter to view previously viewed messages or to view all messages, both those viewed and those not viewed.

► Creating e-mail from a Bulletin Board Message

The Create Communication action has been added to the Bulletin Board application. When the action is called, the details from the message automatically populate the Create Communication dialog. The recipient list for the communication is derived from the targeted audience of the Bulletin Board message.

New create ticket and work order actions

New features have been added to the Assets, Locations, and Configuration Items applications. For example, dialog-based create actions are available in the Assets, Locations, and Configuration Items applications, enabling an operator to create a ticket or work order from a particular asset, location, or configuration item.

Communications log enhancement

To capture escalation and workflow-driven communications, a flag has been added to the Communication templates to drive the creation of communication log entries for communications generated as part of an escalation process or a workflow step. The communication log entry is displayed in the communication log of the targeted record.

Enhanced assign ownership actions

Maximo Asset Management V7.1 provides enhanced filtering capabilities to facilitate the selection of the appropriate owner for a ticket or work order:

► Filter by OwnerGroup

– Automatic crossover from Ownergroup on the ticket to the Person Group filter in the Select Owner dialog

– Ad hoc filtering - Clear or change the Person Group filter to refresh the result set

► Filter for people who are available to work according to their assigned calendar or shift

– Reported Date/Target Start Date/System Date from a ticket are used to filter person records.

– Configurable - Turn on or off the date field Crossover at the site level.

Enhanced related record functionality from multiple dialogs

The capability of creating related records from additional dialogs is available in Maximo Asset Management V7.1:

► View Tickets and Work Orders

– Detail menu on Reported by and Affected by fields

► View Work details

– Detail menu launched from Assets, Locations, CIs. In Maximo V6, this menu was launched from View Ticket, View Work Orders, and PMs

► View related records for Service Group and Service

– Tickets tab

– Work Order tab

Global ticket management enhancements

In Maximo Asset Management V7.1, global ticket functionality is enhanced to enable the global management of communications and solutions from the global ticket.

Classification description field lookup

The Classification Description field is a lookup that returns all classifications that contain the description entered in the field. The classification is then selected from the returned results.

List page functionality

The list page within all of the ticket applications includes the capability to change status or ownership for a group of selected records.

Solution exposure in the View Service Request application

The self-service user can view and print the details of the solution applied to a follow-up record in the View Service Request application.

2.1.2 Technical enhancements

The technical enhancements have been added to the basic services provided by Maximo Asset Management. These services are part of Tivoli's process automation engine.

Improved configuration applications

Figure 2-2 lists some enhancements and depicts the trend toward increased configuration tooling in Maximo V7.1.
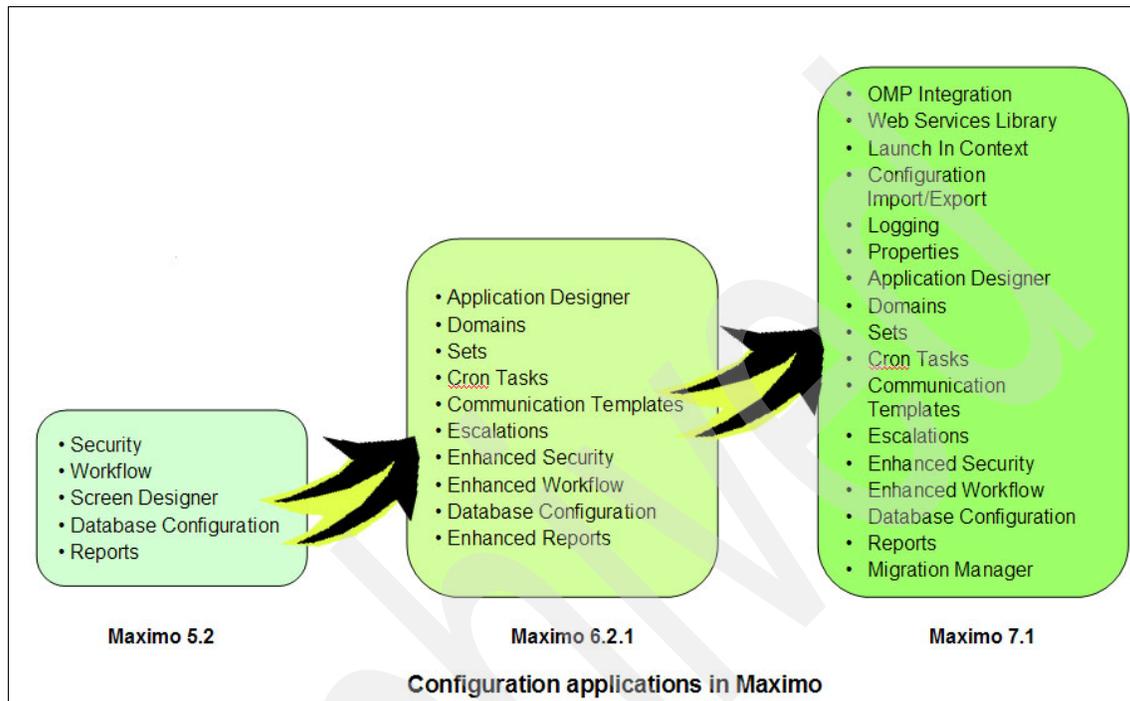
Figure 2-2   Technical evolution of Maximo Asset Management V7.1

System Properties

This application manages systemwide properties for a Maximo product installation. Properties consist of keys and values that determine the configuration of the product and the behavior of many of its components. For example, the mail.smtp.host property and its value represent an SMTP server to which Maximo-generated e-mail notifications are sent.

In previous releases of Maximo Asset Management, system properties were managed through files that were part of the Maximo Enterprise Archive (EAR). When a system property was changed, an administrator had to shut down the application server and rebuild and redeploy a new Maximo EAR file containing the appropriate updated properties file, and then restart the application server. This process is no longer required with the System Properties application. System administrators can change a property value using the application and have it take effect immediately. This saves the client considerable system administrator resources and time.

Logging

This dedicated application manages logging in Maximo. An administrator can configure various logging components, set log levels, associate logging with log files, and specify a folder where log files are to be written. For example, an administrator can enable the Maximo SQL logging component, set its log level to INFO, configure it to write log statements into the maximo.log file, and place the file in c:\mx\logs folder.

In previous releases of Maximo, logging was managed through files that were part of the Maximo EAR. When a logging parameter was configured or added, an administrator had to shut down the application server, rebuild and redeploy a new Maximo EAR file containing the appropriate updated files, and restart the application server. Using the Logging application, this process is no longer required; an administrator can enable a new logger, set its log level, and have it take effect immediately. This saves the client considerable system administrator resources and time.

Migration Manager

Migration Manager is a dedicated tool that promotes Maximo configurations from pre-production to production environments. Accordingly, it is an engine that transports configurations and customizations from one environment to another.

Migration Manager enables more effective deployment of Maximo configurations across development, test, and production environments, as shown in Figure 2-3.
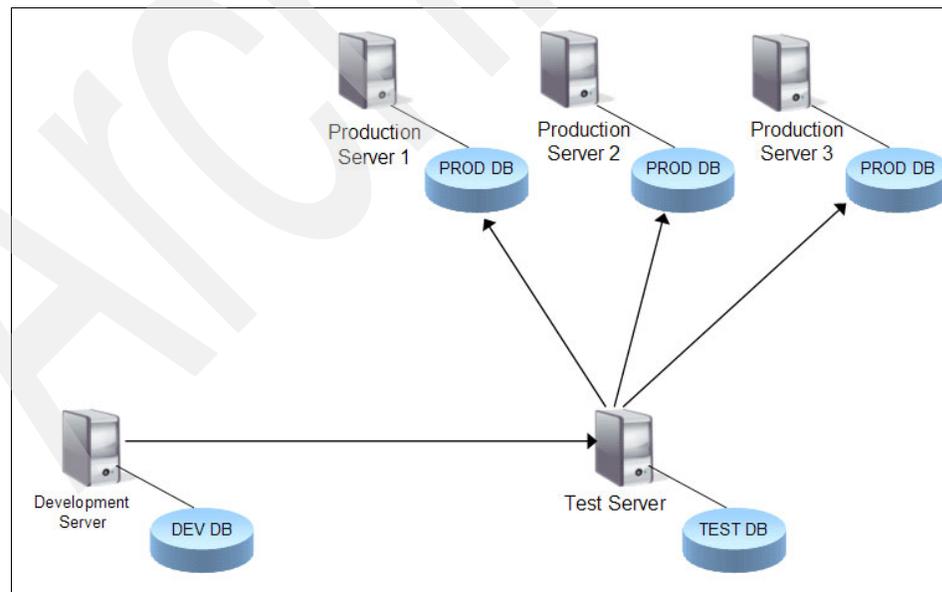
Figure 2-3 Sample deployment environment

Migration Manager enables you to perform the following two key functions for deploying configurations into production:

► Identify, control, create, and organize configuration content to be migrated to production

► Plan, prepare, create, distribute, and deploy migration packages that carry over configuration content from one environment to another

Migration Manager includes applications that enable you to assemble configuration content for migration and perform the steps involved in migrating that content from one product environment to another.

Enhanced user interface

Changes to the user interface in Maximo Asset Management V7.1 are accessibility and efficiency enhancements. For accessibility, users of the JAWS screen reader can interact with more aspects of the product once the screen-reader flag is set in the User record. To increase user efficiency, one-click printing is possible for BIRT report users. In addition, thumbnail images can be included in records, so that JPEG and GIF files can be viewed without having to open attachments.

Conditional Expression Manager

This release introduces the concept of a Maximo condition library. Within this library a user can define conditions, either as expressions or as custom class files that can drive application behavior. Conditions within the library can be used for conditional option access, data restrictions, and a conditional user interface. In the future, these conditions probably will be used in other areas as well. The syntax of expressions for these conditions is similar to SQL, but they leverage bind variables for reusability. A new application, Conditional Expression® Manager, is delivered with this release to facilitate the management of these conditions.

Conditional option access

Maximo Asset Management Version 7.1 makes it possible to tie a condition to a security group's access to an application option. For example, a security group can be given Read access to the Assets application but only under certain conditions - for example, when the status is operating. Conditional access is granted in the Security Groups application. A group can be granted access to an option unconditionally, as was the case in previous versions of Maximo, or a condition can be added. For a user is multiple groups, the highest level of access is granted when accesses are combined.

Data restrictions

In previous versions of Maximo, user access to data could be restricted with User Restrictions and Group Restrictions. These types of restrictions had limitations in that they applied only to viewing data, not to making modifications.

In Maximo Version 7.1, Group Restrictions are replaced with Data Restrictions that enable administrators to configure access to data for groups of users in many more ways. These restrictions are configured in the Security Groups application, as they were in the previous version. An entire object or an entire object within the context of an application can be designated hidden or read-only either conditionally or unconditionally for the entire system or for a security group. In addition, an object or application can be associated with a condition to filter the data to be returned. In this case, only data meeting the condition is fetched from the database, which differs slightly from data that is fetched from the database but hidden under a certain condition. Data restrictions can also be set for attributes within objects, either with or without specifying an application. In these restrictions the attribute can be hidden, required, or read-only conditionally or unconditionally for the entire system or for a security group.

Conditional user interface

In previous versions of Maximo, menu items and buttons were the only user interface elements that could be bound to Sig Option records and therefore controlled by security. In Maximo Asset Management V7.1, it is possible to bind any user interface control to a Sig Option and thereby grant or revoke it to groups of users, as described in "Conditional option access" on page 25. In addition to this granular granting of access, the latest Maximo version offers the ability to configure the properties of a control (label, color, value list, and so on) so that it differs from the default for specified groups of users unconditionally or when one or more conditions evaluates to true or false. The association of a control with a Sig Option and the configuration of conditional properties takes place in the Application Designer.

Integration Framework

The support of inbound interfaces (now called *enterprise services*) have been expanded to include object structure services and standard services. These services enable synchronous access to object structures (formerly called *integration objects*) using HTTP or SOAP. Using object structure services requires minimal configuration because queues and external system association are not required. Standard services are application service methods that are annotated and callable using HTTP or SOAP. Methods such as Change Status are annotated in applications services to provide the ability to invoke these fine-grained services from external applications.

The invocation channel has also been introduced in Maximo V7.1. The invocation channel is a synchronous outbound process that can invoke an external service and return the results to the invoker. Using Maximo Actions application, an invocation channel can invoke an external service from a Workflow process, an Escalation process, or an application using a user interface control (that is, a button or select action menu).

Launch in Context provides a means for launching an external Web-based application passing data (context) from the Maximo application to the external application. Launch URLs are created using a Maximo application and can be configured to reference a MBO attribute that is substituted in the URL at run time.

IBM Service Management enhancements provide framework support for Integration Modules, which are used to perform integration to Operational Management Products.

Out-of-the-box content has increased by providing more object structures (formerly integration objects) and more Enterprise Services and Publish Channels (inbound and outbound interfaces).

LDAP support

The V6.x framework functionality enables Maximo to authenticate a user with an external system. In Maximo Asset Management V7.1, additional LDAP integrations with directory structures have been added.There are now standard LDAP integrations with both Microsoft Active Directory® and IBM Directory Server. In addition, configurations enable all users that pass LDAP authentication to proceed into Maximo. By default, the creation of security groups can be performed in Maximo and user and group management can be performed in Maximo rather than in the directory.

Reporting

Maximo V7.1 introduces a new reporting tool option, BIRT. This tool complements existing reporting options, which are

► Business Objects/Crystal
► External Report Integration
► Open Database Integration

Enabling multiple reporting options gives clients the flexibility and configurability to use the report tool of their choice.

BIRT is automatically installed as the default reporting tool in Maximo V7.1. BIRT is being leveraged for several reasons, some of which are:

► Reduces complexity of installs
► Eliminates platform dependencies

- ► Backed by industry analysts
- ► Based on familiar Web architecture (Eclipse and Java)
- ► Increases growth and flexibility

Multiple report enhancements in Maximo V7.1 focus on the areas of security, performance, and configurability. Ad hoc reporting, the ability of users to create their own reports, is also available. Furthermore, Maximo V7.1 adds several new out-of-the-box reports to the standard offering. A variety of report types are delivered:

- ► Drill down
- ► Graphical
- ► Hierarchical
- ► Detail and analysis reports.

Oracle and SAP integration

Maximo Asset Management V7.1 provides enhancements targeting integration with both Oracle and SAP applications.

Upgrade utility

This utility is designed to assist with upgrading Maximo from version to version. The Maximo V7.1 Upgrade utility has streamlined upgrade workflow by integrating the database and interface upgrade functions into a single utility. Furthermore, consumability has also been improved by providing the Upgrade utility as part of the installation.

Application Designer

Application Designer was introduced in Maximo V6.x to assist with initiatives such as modifying the user interface. Application Designer in Maximo V7.1 continues to expand on this in two distinctive ways. First, new user interface framework controls are introduced, such as breadcrumb navigation and item images for enhancing the user experience of requesting services or searching for items in catalog-type applications. Second, new features are added that support the creation of a more dynamic, condition-based user interface, thus improving the user experience.

E-mail Listener

E-mail Listener functionality is significantly enhanced. The E-mail Listener processes incoming e-mail and generates or updates Maximo objects such as Service Requests (SRs). In Maximo V7.1, E-mail Listener can process both formatted and unformatted e-mails. A formatted e-mail message contains attributes and values, and specific commands that the E-mail Listener performs on behalf of the sender. Supported commands include create, update, query, and change status.

For example, the sender may want to query the status of three high-priority SRs. The user prepares a formatted e-mail in which the user specifies the Maximo object as SR and the command as QUERY, and includes a comma-separated list of the SR numbers to query. E-mail Listener processes the e-mail, queries the specified SRs, and returns the results of the query to the sender. E-mail message formatting may be based on attribute-value pairs or XML.

E-mail Listener implements a security model that is used to determine the sender's authorizations before performing an operation on the sender's behalf. The out-of-the-box workflow process that is part of E-mail Listener has been enhanced to generate automatic confirmation notifications when requested operations are successfully completed. Finally, the E-mail Listener application now provides a detailed view of incoming e-mail messages. Using this view, administrators can monitor and review e-mail messages being received and, if necessary, delete or resubmit specific messages.

Manage sessions

Maximo Asset Management V7.1 has enhanced tracking user session information and logon history information. A new action, Manage Sessions, has been added to the Users application that enables a user to view and download the information for current user sessions and logon history. In addition, more detailed information related to user connections is captured in the MAXSESSION table and persisted to the LOGINTRACKING table. New reports and KPIs for user types, logon history, and session information are also being delivered. Accordingly, this functionality enables companies to accurately track their license counts.

Adapter for Microsoft Project

Support for the scheduling of Changes, Releases, and Activities has been added to Maximo V7.1. Furthermore, support has been added for inserting new Microsoft Project task rows as either Maximo work orders or tasks.

Part 2

Pre-Installation

31

3

Project planning

This chapter describes various parameters that you must consider when planning an implementation. These parameters range from 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 for a typical implementation with estimations of effort and time.

Our implementation plan does not fit every implementation because each is different. Each client is unique, and clients' requirement sets differ. In addition to considering requirements, clients must consider the following parameters while planning an implementation:

► Infrastructure selection

  – Operating system - Microsoft Windows®, Linux, AIX®

  – Database - Oracle, DB2®, SQL

  – Application server - WebSphere® Application Server, WebLogic Server

► Integration with other third-party systems

► Existing data sets for data import

► Business process maturity

The number of combinations generated by these parameters are innumerable and can greatly impact the project schedule. You must carefully examine these parameters during requirements gathering. Some parameters are subjective and

can often create political challenges within the client organization. Thus, you have to be careful in dealing with them.

The following phases comprise a simple implementation:

► Phase 1: Requirements gathering and analysis
► Phase 2: Solution design
► Phase 3: Deployment
► Phase 4: Testing and go-live
► Phase 5: Support

The subsequent sections discuss the people, process, and technology required in each phase.

3.1 Required skills

The project delivery team is fundamental to the success of a project. The exact number of people required varies with project, organization experience, reusable components, and methodology. The people who make up the team perform the roles described in the sections that follow.

Project manager

The project manager performs the following activities:

► Owns the implementation end-to-end
► Is the single point of contact for the client
► Facilitates client interactions
► Reports project status
► Monitors project progress at regular intervals
► Escalates issues
► Changes control agent for scope creep
► Makes resources, individuals, and materials available

Business analyst

The business analyst understands the product and also the client's domain. This person can speak with clients in their language and can translate client requirements to the technical team. This role is vital to success of the project. The following are the attributes of this role:

► Domain knowledge
► Functional knowledge of Maximo Asset Management
► Working knowledge of Unified Modeling Language (UML)
► Knowledge of industry best practices to suggest to client

Solution architect

The solution architect designs the complete solution, including integration with various third-party systems. The attributes and tasks this role undertakes are as follows:

► Possess strong technical skills in Maximo
► Maps business processes to technology
► Understands functional integration points
► Understands technical integration points
► Builds the overall solution design
► Walks the team through various aspects of the implementation
► Resolves technical issues

Maximo developer

This group of people actually installs and configures the software to work in accord with client needs. The major tasks to be performed by the developers are:

► Installs Maximo
► Configures screens, workfows, information flow
► Works on integration
► Develops custom classes, if required
► Performs system optimization

Testers

Testing is critical to every project. This role is involved in testing system integration, functionality, and performance. Depending on the complexity of the project, this role can be performed by the developer or a specialized group. The major tasks include the following:

► System integration: Testing the integrations to ensure the data between systems is flowing accurately.
► Functional testing: Testing the workflows, communication templates, data flow between dialogs, and functions for accuracy.
► Performance testing: This test ensures response times are accurate and the system is responding optimally.

Database administrators

Like any enterprise system, the Maximo database can be accessed multiple times by multiple people. The uptime of the database is critical, and the transaction-processing capabilities have to be at their best at all times. The DBA performs the following activities:

► Troubleshoots database creation
► Installs database if other than DB2
► Performs database optimization
► Facilitates database-level integrations with enterprise systems

Application server specialist

Depending on the choice of application server, WebSphere Application Server or WebLogic Server, the part-time involvement of an application server specialist or administrator may be required. This becomes more important with complex implementations in a clustered environment with numerous integrations.

Integration specialist

The size and skills of the members of this optional group may vary from client to client. Some special systems such as SAP, Oracle, and other niche products may require technical experts. Depending on the project, the main task is to facilitate integration with external systems and ensure the bidirectional movement of data.

The various roles of individuals in the project delivery team are not a permanent fixture for each project. More roles may be required, and more than one person may perform in a role. As mentioned previously, each project is different, and it is difficult to generalize; however, we have attempted to cover scenarios that we have encountered in our implementation experience.

3.2 Solution description and assumptions

This section discusses a high-level solution architecture and the standard assumptions made when planning a project.

Figure 3-1 depicts a standard functional solution layout.
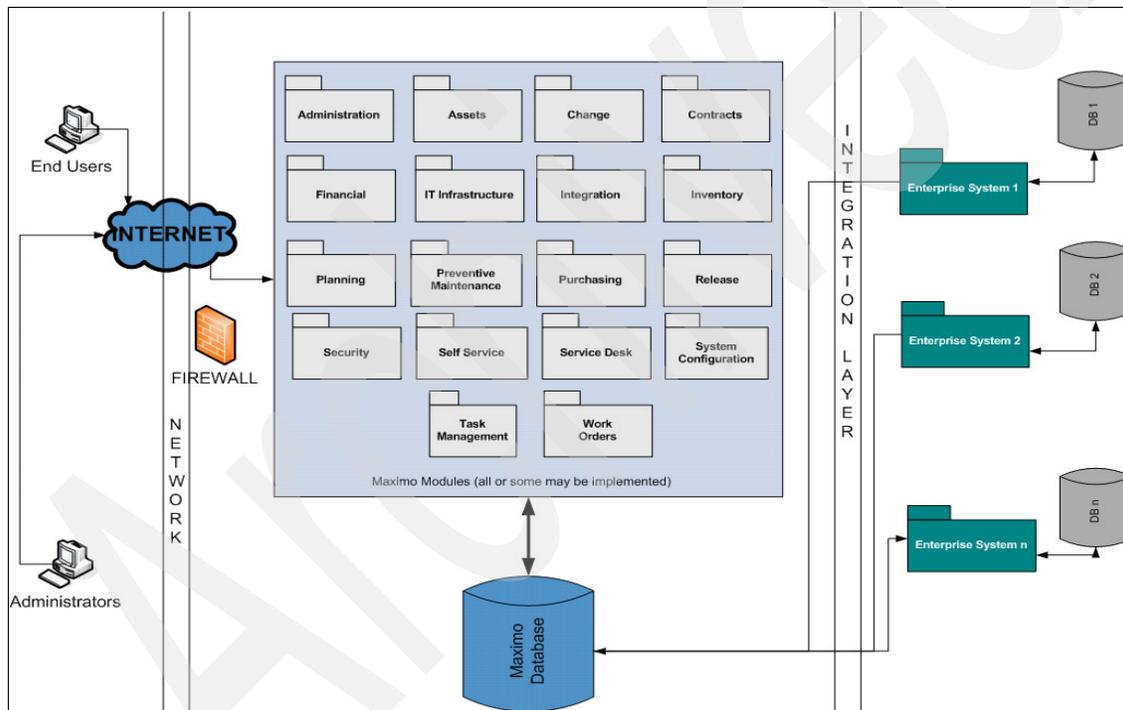
Figure 3-1   Functional architectural structure of a Maximo-based solution

The Maximo user interface is based on a Web browser so the application can be accessed remotely, depending on network connectivity and firewall configurations.

The three layers shown in Figure 3-1 can be complicated, depending on client requirements and network setup. All or some of the Maximo Asset Management

V7.1 modules may be implemented. The integration layer separates the enterprise applications from core Maximo base services.

> **Attention:** With the number of applications taking advantage of the underlying Maximo base services, these base services have no been renamed to Tivoli's process automation engine. Until all product documentation is revised, you may encounter inconsistencies in the terminology.

Standard assumptions made when planning a project are the following:

► The software is procured prior to project initiation.

► The three environments are development, testing, and production.

► The application server, database server, and reporting engine are kept in separate physical servers.

► The client owns any configurations required in the enterprise systems to facilitate integrations.

► Client involvement during the initial project stages is fairly high because much of the functional and business process knowledge specific to the client has to be transferred to the implementation team.

► Management has bought into this project.

► A dedicated single point of contact and a project manager are appointed from the client team to manage interactions with users.

► Historic data migration can be performed by the implementation team, but data cleansing and scrubbing must be owned by the client.

3.3 Task breakdown

As previously mentioned, the standard phases of a Maximo implementation project are:

► Phase 1: Requirement gathering and analysis
► Phase 2: Solution design
► Phase 3: Deployment
► Phase 4: Testing and go-live
► Phase 5: Support

The key activities in each of the phases are described in the following sections.

Phase 1: Requirements gathering and analysis

This section describes the roles involved in and key activities that occur during the requirement gathering and analysis phase.

Roles:

► Project manager
► Business analyst
► Solution architect

Key activities:

► Understand current business processes for managing:

– Maintenance
– Procurement
– Inventory
– Assets

► Document plan areas with existing business processes.

► Demonstrate the basic product.

► Understand and document the end goal and vision of the client.

► Document "to-be" business processes for each of the functions previously mentioned.

► Analyze the points of integration.

► Analyze the current data sources.

► Create a SRS (software requirements specification) document.

Phase 2: Solution design

This section describes the roles involved in and key activities that occur during the solution design phase of the implementation project.

Roles:

► Project manager
► Business analyst
► Solution architect

Key activities:

► Identify module requirements.

► Identify configuration requirements in terms of:

– Roles
– Security groups
– Escalations

- Communication templates
- Work Order templates
- Job plans
- SLAs
- Vendors and contracts
- Interface configurations
- GL codes
- Workflow

► Create a detailed implementation plan.

► Identify methodology for integration.

► Identify scripting or manual data import approach from identified data sources.

► Create a solution architecture diagram.

Phase 3: Deployment

This phase consists of the following roles and activities:

Roles:

► Project manager
► Solution architect
► Maximo developers

Optional roles:

► DBA
► Application server specialist (depending on environment)

Key activities:

► Software installation

- Install the various components if doing a manual installation.
- Run the launchpad for an automated installation.
- Test to verify the installation is successful.

Software installation is different for various deployment topologies - single server, multiserver, clustered environment. It also depends on the infrastructure selected.

► Software configuration

- Basic operation configuration, which includes but is not limited to the creation of:

  • Organization
  • Locations
  • GL codes

- Users, roles, and security groups
- Job plans and work orders

– Software configuration also includes the configuration of role-based Start Centers.

– Configuration of the tool to map to business processes includes but is not limited to the following:

- Add or delete (as the case may be) fields on the interface and in 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 import scripts
- Import data from its existing source to Maximo Asset Management
- Build integration interfaces - either using custom code or the Integration Framework
- Create and deploy reports

Phase 4: Testing and go-live

The roles taking part in and the key activities making up the test and go-live phase are described in this section.

Roles:

- ► Project manager
- ► Solution architect
- ► Testers
- ► Maximo developers

Optional roles:

- ► DBA
- ► Application server specialist

The key activity is testing.

The first key task is migration from the development environment to the test environment. Depending on the client's environment and the complexity of the implementation, a client may decide to use a manual approach or to use automated tools for testing. All or some of the following types of tests must be carried out:

- ► Functional testing

  – Validations in new fields

  – E-mails being sent on escalations

- Workflows
- Accuracy of data flowing between modules in the system
- Data accuracy in reports

► System integration testing

- Accuracy of data coming from external systems
- Accuracy of data going into external systems
- Scenario testing to ensure system availability

► Performance testing

- Page load time
- Peak load
- Concurrent users
- Transaction processing time with database and external data sources

► Go-live

- Plan cut-over strategy.
- Migrate to production environment.
- Execute each of the tests listed in this section in the production environment to ensure no performance glitch exists in the live environment.
- Training must be provided to all or some of the users. The recommended approach is "train-the-trainer." Provide training to three types of users:

  • Users: Provide functional training to the user community on how to use various functional aspects of the software. This training can be done with focus groups or generic groups of users, depending on how the organization is structured.

  • Groups of users: Role-based training is specific to certain groups. For example, the training of the purchase administrator, supervisors, and others must result in their ability to approve or disapprove certain tasks, manage their queues, and so on.

  • Administrators: After the implementation team is finished, an internal person must be designated to manage the system. This individual is responsible for system availability, the creation of new reports, enhancements, adding users, assigning roles to security groups, and so on.

Phase 5: Support

This section describes the roles involved in and the activities that take place during the support phase of the implementation.

Roles

- ► Project manager
- ► Maximo developers

Optional roles:

- ► Business analyst
- ► Solution architect

Key activities: Some organizations do not have a strong IT presence and opt for the implementation team to provide continued system support. The basic tasks the support team must perform are, but are not limited to, the following:

- ► Manage system availability
- ► Make enhancements to accommodate evolving business processes
- ► Create and deploy new reports
- ► Build new integrations
- ► Plan capacity for additional users and or organizations

This five-phase methodology is fairly generic but includes most of the basic activities that must be considered and planned for in an implementation.

At this point, we address the client's involvement in each stage. For each of the previously mentioned phases, expectations of client involvement are listed in Table 3-1.

Table 3-1   Expectations of client involvement in each implementation phase

Phase	Expectations of client involvement
Requirements gathering and analysis	► Assign a single point of contact from within ► Explain the project vision and the short-term and long-term objectives ► Facilitate meetings with individual process owners ► Facilitate meetings with tool owners ► Procure required hardware and software ► Review documentation
Solution design	► Review solution architecture ► Review documentation and project plan

Phase	Expectations of client involvement
Deployment	► Provide hardware and software ► Provide administrative access to systems wherever required ► Monitor project progress ► Participate in reviews
Testing and go-live	► Provide test and production environment ► Provide data for user acceptance testing (UAT) ► Facilitate UAT ► Attend training ► Provide human and material resources required for training
Support	► Define support scope of work boundaries ► Review support plan and service-level agreements (SLAs)

Solution environment

Maximo Asset Management requires multiple software servers that can be installed on separate, dedicated server machines (for best performance) or the same physical server (not recommended for production environments). The different components of the Maximo solution environment are as follows:

► **Database**: The Maximo database serves as the repository for all CI information.

► **Application server**: Maximo Asset Management is built using Java 2 Enterprise Edition (J2EE) technology, which requires a commercial application server, such as IBM WebSphere Application Server. The application server consists of Maximo Asset Management applications using JavaServer™ Pages (JSP™), XML, and Maximo Asset Management application-specific business components.

► **HTTP server**: A separate, dedicated HTTP server can be configured to work with the J2EE application server.

► **Directory server**: A directory server can be configured to work with Maximo Asset Management to maintain lists of users and groups for security purposes.

► **Administrative system**: The administrative system is used to deploy Maximo Asset Management. After the initial deployment, the administrative system is used to make updates or changes to the deployment. Changes to the Maximo Asset Management deployment typically require that Maximo Asset Management Enterprise Archive (EAR) files be rebuilt. See "Rebuilding and redeploying enterprise archive files" on page 294.

4.1 Hardware and software requirements

Maximo Asset Management hardware and software requirements are listed in the sections that follow. Each product version listed reflects the minimum requirement for use with Maximo Asset Management. Maximo components are designed to run within 32-bit environments only, with the exception of AIX V5.3, which is a 64-bit environment.

Software supported by Maximo Asset Management can be run on any hardware platform supported by the database software listed in the "Software" column (in Table 4-1), provided the system is hosting one of the supported operating systems listed in the "Operating system" column. If available, the Maximo Asset Management administrative workstation and systems hosting Maximo middleware can support an IPv6 network configuration.

Browser

Maximo Asset Management supports Microsoft Internet Explorer® Version 6 and later.

Database

Required database software is listed in Table 4-1.

Table 4-1   Required database software

Software	Operating system
► DB2 UDB V9.1.3 or V8.2.8 ► Oracle V9.2.0.8, Oracle V10.2.0.3, or Oracle V10.1.0.5, Standard or Enterprise version ► Microsoft SQL Server® 2005 SP2, Standard or Enterprise version	Refer to vendor specifications

Directory server

The products listed in Table 4-2 can serve as the directory server component of a Maximo Asset Management V7.1 deployment.

Table 4-2   Directory server requirements

Software	Operating system
► IBM Tivoli Directory Server V6.1 FP1 ► Microsoft Windows Server® 2003 SP2 Active Directory Microsoft Active Directory Application Mode (ADAM) not supported	► Windows Server 2003 SP2 (Standard, Enterprise, or DataCenter - 32-bit, 64-bit) ► Windows Vista® (Business, Enterprise, Ultimate - 32-bit, 64-bit) ► Windows XP Professional SP2 (32-bit, 64-bit) ► Red Hat Enterprise Linux V4 (Enterprise or Advanced) (update 4+5 or later) (Intel® 32-bit) ► IBM AIX 5L™ V5.3 ML level 5300-06 ► SuSE Linux (SLES) V9.0 Enterprise Server System z™ SP4 or later (manual installation only)

J2EE application server

You install WebSphere Application Server on the J2EE application server. In addition, Maximo Asset Management runs on the J2EE application server (see Table 4-3).

Table 4-3   J2EE application server requirements

Hardware	Software
▶ 2-4 dedicated Intel-based Pentium® processors ▶ 2 GB RAM per processor ▶ 1.5 GB or greater of disk space for Maximo and Java and Web server components	▶ Windows Server 2003 (Standard, Enterprise, or DataCenter - 32-bit, 64-bit) ▶ IBM AIX V5.3 or V6.1 ▶ Red Hat Enterprise Linux V4 and V5 (x86 and x64 processor-based systems) ▶ SuSE Linux (SLES) V9.0 Enterprise Server System z SP4 or later (manual installation only) **Note:** Although Maximo supports the following operating systems, if you are using IBM DB2 Universal database with Maximo Asset Management V7.1, you cannot use any of the following operating systems on your application server: ▶ HP-UX 11i v2 (PA-RISC processor-based systems) ▶ Sun™ Solaris™ Version 9 or 10 (SPARC processor-based systems) ▶ IBM WebSphere Application Server Network Deployment V6.1.0.11 (provided by IBM Corporation) ▶ BEA Weblogic V9.2.2 (provided by client)

Administrative system

Administrative system requirements are provided in Table 4-4.

Table 4-4   Administrative system requirements

Hardware	Software
► Intel-based Pentium processor ► 1 GB RAM ► SVGA 1024 x 768 resolution; if used for Application Designer, 1280 x 1024 resolution	► Windows Vista (Business, Enterprise, Ultimate - 32-bit, 64-bit) ► Windows XP Professional SP2 (32-bit, 64-bit) ► Adobe® Acrobat® Reader V6.0 **Note:** The Maximo Workflow Designer requires a Java Runtime Environment. JRE™ V1.4.2 – JRE V1.5 are supported.

Client system

Software and hardware requirements for the client system are listed in Table 4-5.

Table 4-5   Client system requirements

Hardware	Software
► Intel-based Pentium processor ► 1 GB RAM ► SVGA 1024 x 768 resolution	► Windows Vista (Business, Enterprise, Ultimate - 32-bit, 64-bit) ► Windows XP Professional SP2 (32-bit, 64-bit) ► Adobe Acrobat Reader V6.0 and later

4.2 Typical deployment environments

The Maximo Asset Management solution can be deployed primarily in two different topologies within an enterprise.

4.2.1 Single-server

The single-server topology consists of loading all Maximo Asset Management components onto one machine. This is typically done for proof-of-concept purposes, as a demonstration, or in a learning environment. For managing enterprise assets and processes, you typically implement a multiserver topology.

Figure 4-1 depicts the application server MXServer running Maximo Asset Management on a single physical machine.
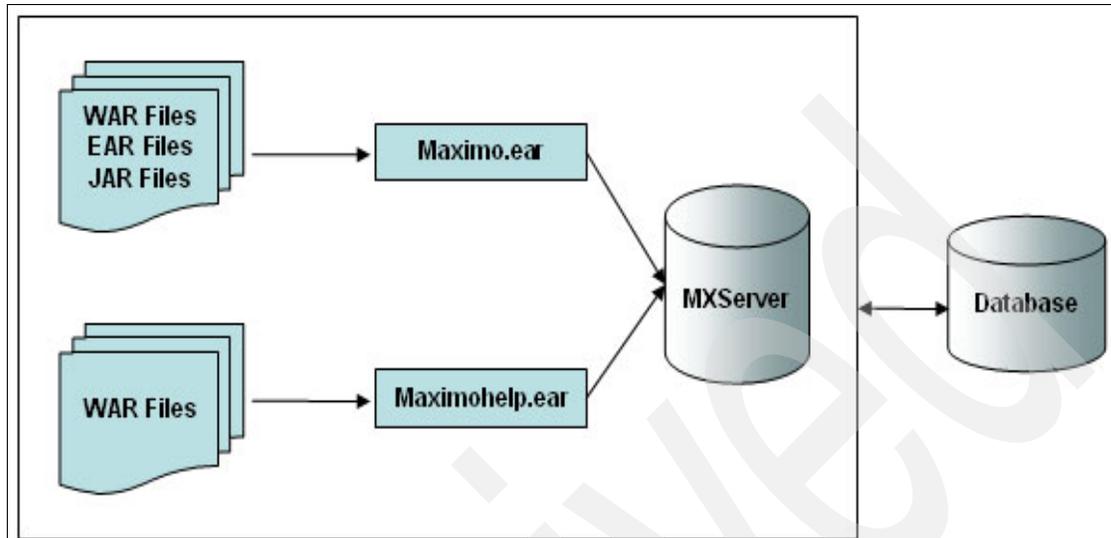
Figure 4-1   Single-server deployment

4.2.2 Multiserver

The multiserver topology consists of splitting Maximo Asset Management components across several different machines. This is beneficial because it optimizes resource use and decreases each system's workload. This type of deployment is typical for production use within an enterprise.

Maximo Asset Management should be deployed on multiple machines to provide load balancing, availability, reuse, and redundancy. The multiserver topology is the recommended deployment topology for a production environment.

When contemplating your deployment strategy, you must determine whether it will include systems already established in your network. Implementing Maximo Asset Management by installing all new components using the Maximo middleware and Maximo Asset Management installation programs simplifies the deployment. If you plan to reuse or migrate resources that already exist in your network, make adjustments to your rollout plan to allow time for steps such as bringing the existing resources to version levels that are compatible with Maximo Asset Management.

In a disparate environment, the collection of machines in this deployment can be a mixture of Microsoft Windows and UNIX machines. Only the administrative system must be hosted on a Microsoft Windows system.

Figure 4-2 displays a multiserver deployment environment.
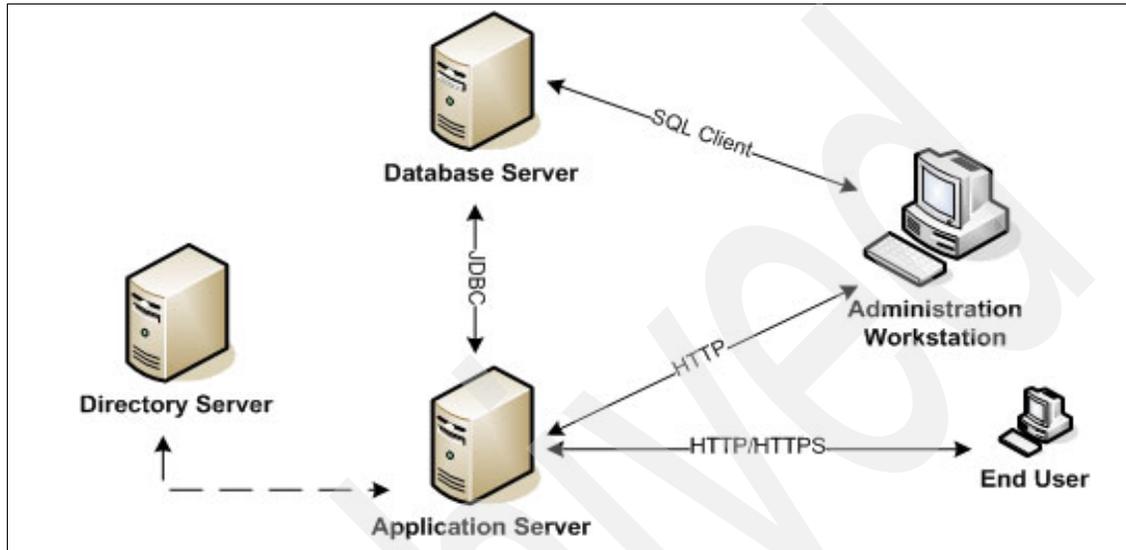
Figure 4-2   Multiserver deployment

A typical deployment life cycle might begin with a single-server topology that moves through the phases of demonstration, functional proof-of-concept, and testing integration within the existing environment. Then the life cycle gradually moves toward a pilot multiserver environment before finally implementing a production deployment within the enterprise.

4.3 Clustered environment

To achieve availability, load balancing, scalability, and reliability, Maximo Asset Management can be configured on multiple application servers. Maximo instances are run on different application servers either on the same physical server or on different physical servers to form a cluster of applications. Each Maximo instance can be configured identically, or all can be configured differently by deploying different EAR files for every Maximo instance configured differently.

With IBM WebSphere Application Server Network Deployment, you can build a distributed server configuration, which enables central administration, workload management, and failover. In this environment, you integrate one or more

application servers into a cell managed by a deployment manager. The application servers can reside on the same machine as the deployment manager or on multiple separate machines.

Administration and management are handled centrally from the administration interfaces through the deployment manager. With this configuration, you can create multiple application servers to run requests of applications and then manage those applications from a central location. More importantly, you can cluster application servers to enable workload management and failover capabilities.

Applications you install in the cluster are replicated across the application servers. When one server fails, another server in the cluster continues processing. Workload is distributed among Web containers and EJB™ containers in a cluster using a weighted round-robin scheme.

Figure 4-3 depicts Maximo Asset Management deployed in a clustered environment.
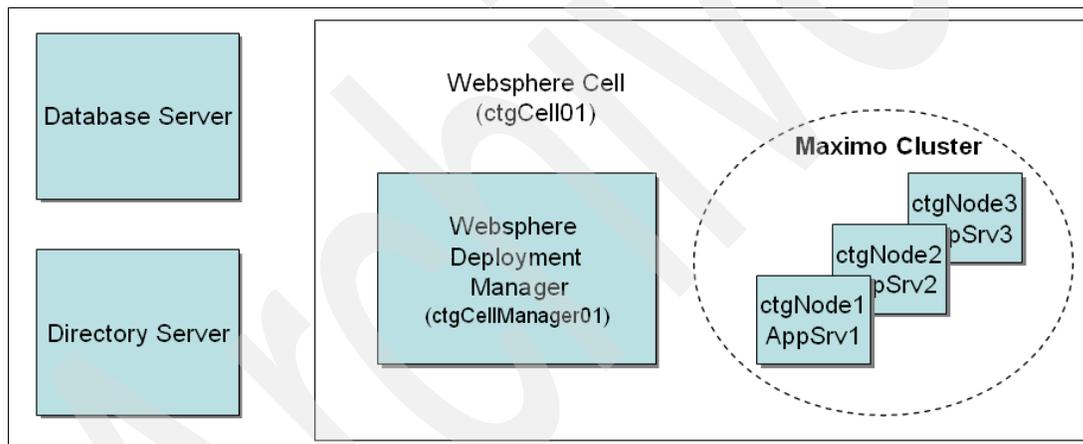
Figure 4-3   Maximo Asset Management deployed in a cluster environment

4.4 Sizing considerations

Sizing hardware for an application is a critical task. One objective is to achieve optimal performance; another is to support the growth of the enterprise in terms of number of users. Capacity planning and availability management are two important functions that must be considered while sizing hardware for an

application. You must consider the following factors to achieve proper hardware sizing.

► **Cron task**

Cron tasks are behind-the-scene jobs set to run automatically and on a fixed schedule. These tasks might process a number of records depending on the nature of the job. As the complexity and number of these jobs increases, they consume a lot of resources and therefore become an important point for sizing considerations.

► **Workflows**

Workflow defines the different paths that a record can take as it moves through the business process and the different actions and notifications that should take place at different points in the process. The complex workflows are composed of SLAs, escalations, notifications, and so on.

► **Users**

The number of maximum concurrent users is another important sizing criterion. Available resources must be shared to fulfill requests submitted by different users without compromising performance.

► **Sites**

By configuring multiple sites in the system, an organization can run the system in multiple facilities - all from the same database with site-specific data and processes. This way, rather than installing multiple instances of the system, one installation caters to all the sites in the organization. With multiple sites, you can have multiple GL accounts, workflows, and job plans, which again adds to the complexity of the deployment.

► **Integrations**

Multiple integration points are possible in an implementation such as PeopleSoft® Human Resources Management Solution (HRMS), invoicing, and finance, and these points can be both inbound as well as outbound. These integrations result in heavy transaction flow. The complexity of data filtering and data formatting and the frequency of the triggering events determine the resources required by these integrations.

► **Modules**

Not all implementations include each and every module in Maximo Asset Management. Module usage depends on the business process of the organization and whether any existing maintenance management solutions are in place. The different modules in an implementation involve the associated users, workflows, and transaction data, such as PM-implemented cron tasks, workflows, and inventory.

► **Level of customization**

The customization of different applications may vary from minor field re-labeling to cloning an entire application. Before the project begins, the implementation team should calculate the deviation level of the standard Maximo application from the requirements in place to decide on the necessary hardware to support the level of customization.

► **Reports**

Reporting is an important component of any Maximo Asset Management implementation. Management takes all the business decisions based on the data that is produced in the form of reports, which are extracted using highly complex methods and scripts from the database. The number of such reports makes a huge impact on the performance of the system.

► **Growth forecast**

It should be ensured that hardware sizing aligns with the growth of the organization. New sites coming online in the future, the replacement of an existing system, the expected increase in usage that occurs with the growth of the organization are few of the many factors that must be studied and analyzed before finalizing the hardware components of the implementation.

► **Language**

An increase in complexity depends on the different languages that must be supported by the system. System administrators must ensure that data is stored in supported languages, and that when text is displayed on the user interface, it is in the language of the logged on user.

► **Add-ons**

All the add-ons that the implementation might require must be identified and listed to ensure that hardware sizing takes add-ons into account as well.

► **Linked documents**

If the organization decides to configure linked documents and use this feature as a practice, the servers must have necessary storage space. In addition, the system administrator must ensure that attaching documents does not create storage and retrieval performance issues.

4.5 Maximo Asset Management supported platforms

Table 4-6 lists the platforms supported by IBM Maximo Asset Management.

Table 4-6   Supported platforms

	Maximo V6.1	Maximo V6.2.1	Maximo V7.1
Operating system			
AIX V5.2 (32 bit)	x	x	
AIX V5.3 (32 bit)	x	x	x
AIX V5.3 (64 bit)	x	x	x
Solaris 9 (Sparc)	x	x	x
Solaris 10 (Sparc)	x	x	x
HP-UX 11i V1 (PA-RISC)		x	
HP-UX 11i V2 (PA-RISC)		x	
Microsoft Windows Server 2000	x	x	
Windows Server 2003 Enterprise Edition		x	x
Windows Server 2003 Enterprise x64 Edition		x	x
Red Hat Enterprise Linux 3 (x86, 32 bit)	x	x	
Red Hat Enterprise Linux 4 (x86, 32 bit)			x
Red Hat Enterprise Linux 4 (zLinux)			x
Red Hat Enterprise Linux 5 (x86, 32 bit)			x
Application server			
IBM WebSphere Application Server V6.0	x	x	
IBM WebSphere Application Server V6.1			x
BEA WebLogic V8.1	x	x	
BEA WebLogic V9.2			x
Database			
IBM DB2 UDB V8.2 for Linux, UNIX, and Windows		x	x
IBM DB2 UDB V9.2 for Linux, UNIX, and Windows			x
Oracle 9i V2	x	x	x

	Maximo V6.1	Maximo V6.2.1	Maximo V7.1
Oracle 10 Rel1	x	x	x
Oracle 10 Rel2	x	x	x
Microsoft SQL Server 2000	x	x	
Microsoft SQL Server 2005 (32 bit)	x	x	x
Microsoft SQL Server 2005 (64 bit)		x	x
Browser and client operating system			
Internet Explorer V6.0.x	x	x	
Internet Explorer V7.0.x		x	x
Windows Vista			x
Windows XP Pro (client)	x	x	x
Report writers			
BIRT V2.1.2			x

Part 3

Installation and configuration

57

5

Installation

In this chapter we discuss installation of the prerequisite middleware, Maximo base services (Tivoli's process automation engine), and the Maximo Asset Management Process Solution package.

Our examples are based on the following software components:

- ► Microsoft Windows 2003 Server
- ► WebSphere Application Server V6.1
- ► DB2 database platform
- ► Maximo Asset Management V7.1 application

Although our examples primarily reflect a single-server environment, we realize most production environments are comprised of multiple servers. The process is much the same for both environments, but we take note of those areas where a variation in the process might exist.

For more information, refer to the following installation guides:

- ► *Installation Guide: IBM WebSphere Application Server* (mam71_install_was.pdf)
- ► *Installation Guide: BEA WebLogic Server* (mam71_install_bea.pdf)

  You can access these guides at:

  http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.mam.doc_7.1/mam_welcome.htm

5.1 Pre-installation checklist

Before we start, we need to assess our pre-installation checklist. Carefully consider the checklist items in the following sections.

Hardware and software considerations

Refer to Chapter 4, "Solution environment" on page 45 to verify that your system configuration adheres to the required specifications.

Verify the DVD content

The DVDs listed in this section contain files for the Maximo Asset Management product.

► Maximo Asset Management V7.1 contains the following:

  – Launchpad
  – Maximo Asset Management middleware installer
  – Maximo Asset Management installer, IBM Agent Controller installer
  – Maximo Asset Management Language Pack installation program for Windows 2003

► Middleware for Windows 2003

  Contains prerequisite middleware software for Windows 2003.

► Middleware for Red Hat Enterprise Linux V4 and V5

  Contains prerequisite middleware software for Red Hat Enterprise Linux version 4.

► Middleware for AIX V5.3 and V6.1

  Contains prerequisite middleware software for AIX version 5.3.

► Maximo Asset Management V7.1 Quick Start DVD

  Contains copies of the *Quick Start Guide* in all languages and a copy of the information center. You can also access the *Quick Start Guide* at:

  http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?top
  ic=/com.ibm.mam.doc_7.1/mam_welcome.htm

Backup and snapshot management

You should back up your system before installing any Maximo Asset Management V7.1 component on your system. Notably no automated uninstall feature is supplied with Maximo Asset Management V7.1. If the installation fails at any point, you need to restore your system from the backup or reinstall the respective OS on your machine.

JDK and JRE

We recommended you install IBM Java V5.1 on your system. When implementing BEA WebLogic, you are required to have the update SUN JDK™ V1.5.0 installed.

Internet browser

Maximo Asset Management V7.1 currently supports Mozilla Firefox Internet browser on Linux and Internet Explorer on Microsoft Windows.

Disable the firewall

This section describes how to disable the firewall if one is present on the system. You must disable the firewall for the system on which you are installing Maximo Asset Management middleware. You must disable the firewall prior to using the middleware installer.

The firewall must be disabled on the following platforms.

- – Microsoft Windows
- – Linux

Delete the TEMP and TMP user environment variables

The existence of the TEMP and TMP user variables can cause errors during the installation of DB2 on a Microsoft Windows system. Prior to using the middleware installer to install DB2, you must remove these variables for the user ID that performs the installation. It is important to note that TEMP and TMP are user environment variables, not system variables. To remove the TEMP and TMP user variables on a Windows system, complete the following steps:

1. Access the System Properties dialog by right-clicking the **My Computer** icon on your desktop and selecting **Properties**.

2. From the System Properties dialog, first select the **Advanced** tab, and then click **Environment Variables**.

3. In the User variables section, select **TEMP**, and then click **Delete**. Repeat the process for the TMP variable.

4. Click **OK**.

5. Exit the System Properties dialog by clicking **OK**.

Verify required installation of rpm-build package

This procedure describes how to verify that the rpm-build package is installed on Linux. This package must be installed before you run the middleware installer.

This applies only if you are installing on Linux. To verify that the rpm-build package is installed, perform the following steps:

1. Run the following:

```
rpm -qa | grep build
```

The rpm-build package is installed if the command returns a value such as the following:

```
rpm-build-4.3.3.-18_nonptl
```

2. If nothing is returned, you must install the rpm-build package, which is located on disk 3 (of 5) of the Red Hat Enterprise Advanced Server version 4 installation CDs, using the rpm tool with the -i option.

Set the ulimit

This section describes how to set the ulimit in Linux, which is used to define user system and process resource limits. For Linux systems, you must set the ulimit for the system prior to using the middleware installer. To set the ulimit, complete the following steps:

1. From the command line, type the following:

```
ulimit -f unlimited
```

2. Again from the command line, type the following:

```
ulimit -n 8192
```

If you set the ulimit in the .profile for root, the ulimit setting applies to all processes.

Set the swap size

Maximo Asset Management can be a resource-intensive application. We recommend you configure and tune your system for maximum performance.

This section describes how to set the size of the swap space used in Linux systems. Typically, the swap size set for Linux systems should be equivalent to twice the amount of physical RAM in the machine. Additional swap space can be made available to the system by doing the following:

▶ Increasing the size of the existing swap partition
▶ Creating a new, additional swap partition
▶ Creating a swap file

Refer to the product documentation for your Linux distribution for more information.

Set shared memory

This section describes how you can set a minimum shared memory value in Linux. For Linux systems, you must set a minimum shared memory value for the system prior to using the middleware installer. To set the minimum shared memory value, complete the following steps:

1. If the memory value is less than 268435456 bytes (256 Mb), type the following from the command line:

   ```
   sysctl -w kernel.shmmax
   ```

2. If you need to increase the value, type the following from the command line:

   ```
   sysctl -w kernel.shmmax=268435456.
   ```

3. Update the value in /etc/sysctl.conf.

Enable remote configuration

If you plan to take advantage of the Maximo Asset Management installation program feature that automates the configuration of Maximo middleware, you must enable a Remote Execution and Access (RXA) service for each system on which you intend to install Maximo middleware. RXA requires that the target system enable at least one of the protocols supported by RXA, which include rsh, rexec, SSH, and Microsoft Windows Server Message Block (SMB). Before you start the Maximo installation program, ensure that one of these protocols is running and accepts remote logons using a user name and password configured on the target machine.

If the remote system is a Microsoft Windows machine, you must configure RXA to work over SMB. For Microsoft Windows machines, you cannot use Cygwin ssh. If Cygwin is present on the Windows machine, the installation fails. Default installations of AIX systems might not include a suitable protocol and must have RXA-compatible protocols enabled. RXA does not support accessing network drives on the local or remote system.

5.1.1 Installation flowchart

The flowchart shown in Figure 5-1 on page 65 defines the installation flow required to successfully install Maximo Asset Management. A launchpad utility leads you through the installation of various components of the product. In general, the installation has three phases:

1. Install the required middleware:

   Through the launchpad, you can install an appropriate configuration of WebSphere Application Server, DB2, and Tivoli Directory Server. You can change the default parameters, but if you accept the default options, the

installation of these middleware components through the launchpad is simple and straightforward.

2. Install the base services.

A set of common base services (Tivoli's process automation engine) provide the general functions for all Maximo-based applications. These base services comprise a set of modules that reside on an application server:

– BEA WebLogic Server
– IBM WebSphere Application Server

The base services utilize one of the following database platforms:

– DB2
– Oracle
– Microsoft SQL Server 2005

Note: If you used the launchpad to install the middleware, the launchpad "remembers" the configuration you installed. The launchpad utilizes the configuration to simplify the installation of the base services. If you installed the middleware separately or used other products (namely Oracle, SQL Server, or BEA WebLogic), you can still use the launchpad, but you have to specify the appropriate parameters for the middleware environment you have installed.

3. In the third phase of the installation, you enable or add the Maximo Asset Management-specific package solutions to the base services installed in step 2. Again, the launchpad guides you through this process, helping ensure that all appropriate parameters are specified for your target environment.

Figure 5-1 shows how the Maximo Asset Management installation relates to the installation of Maximo base services (Tivoli's process automation engine), which are the common facilities used by this and other Maximo applications. The Process Solution Package enables the application-specific (in this case, Maximo Asset Management) functions on top of the base services.
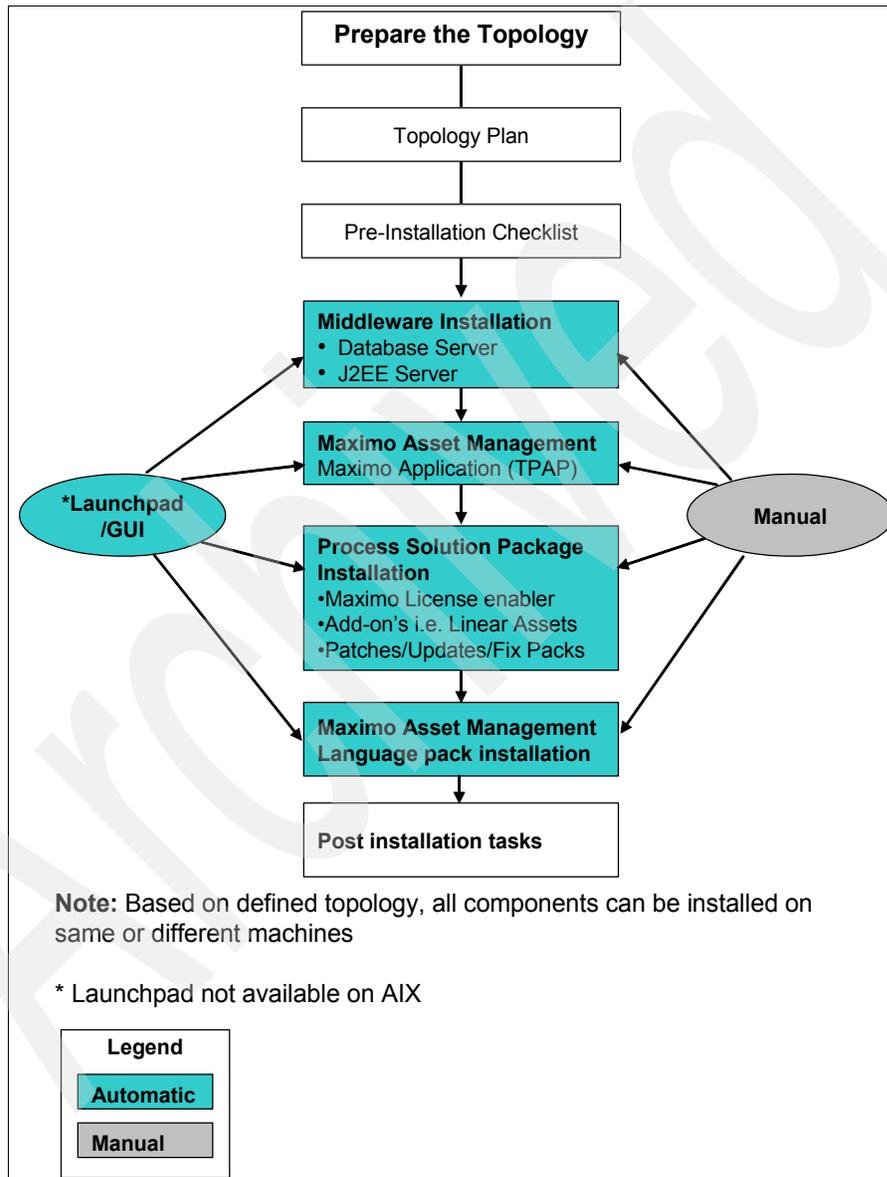
Figure 5-1   Installation flowchart

5.2 Middleware installation

The first step in the installation Maximo Asset Management is to ensure all the necessary middleware is installed and properly configured. The following sections describe how to do so.

5.2.1 Middleware installation overview

Before you can install IBM Maximo Asset Management, several middleware products must be deployed. Specifically, you must install and deploy the software described in the sections that follow.

Database server

Maximo Asset Management uses the Maximo database to store details about the attributes and history of each configuration item and the details about the relationships between configuration items.

You can manually install a new instance of DB2 UDB V9.1, or you can use a preexisting instance of DB2 UDB V8.2 or DB2 UDB V9.1. Another alternative is to install and configure Oracle 10 or Microsoft SQL Server 2005 for your Maximo Asset Management deployment.

Directory server

The directory server is used to secure the Maximo Asset Management J2EE application. You can choose between the IBM Tivoli Directory Server or Microsoft Active Directory server.

> **Note:** This installation step is applicable only if you are enabling J2EE security.

J2EE server

The J2EE server is the application server used to serve and manage the Maximo Asset Management application.You must have an instance of BEA WebLogic Server or IBM WebSphere Application Server installed in your environment.

The Maximo Asset Management product includes a middleware installer that provides a standard installation of the middleware previously listed. With minimum effort, it installs the following versions of middleware:

► DB2

   – DB2-ESE_9.1.0
   – DB2-ESE_9.1.0_FP4

- ► IBM Agent Controller
  - – Rational®-AgentController_7.0.3.1
- ► Tivoli Directory Server
  - – TIV-DirectoryServer_6.1.0
  - – TIV-DirectoryServer_6.1.0_FP0001
- ► WebSphere Application Server V6.1
  - – WS-ESS_6.1_GA
  - – WS-WAS_IHS_6.1.0_FP13
  - – WS-WAS_ND_6.1.0.13_Custom_ISCAE71
  - – WS-WAS_ND_6.1.0_Supplemental
  - – WS-WAS_Plugins_6.1.0_FP13WS-WAS_UpdateInstaller_6.1.0_FP13

This middleware installer can be used on different systems to install individual components, thus simplifying the installation in a multiserver environment. The next section "Middleware installer" describes the use of the middleware installer on a single system.

You do not have to use the middleware installer. You can obtain and install the middleware through the product's normal installation processes. You have to do so if you plan on using other vendors' products or different versions than those in the preceding list. Considerations for manual installation of the middleware are provided in Appendix A, "Manual configuring middleware" on page 249. Refer to this appendix for information about the manual configuration of Maximo Asset Management on the IBM WebSphere Application Server platform.

For more details on other platforms refer to the following installation guides:

- ► *Installation Guide: IBM WebSphere Application Server* (mam71_install_was.pdf)

- ► *Installation Guide: BEA WebLogic Server* (mam71_install_bea.pdf)

- ► Refer to the *IBM Maximo Asset Management Installation Guide* for details about using the middleware installer.

  You can access these guides at:

  http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?top
  ic=/com.ibm.mam.doc_7.1/mam_welcome.htm

The following section summarizes the steps you perform to install the middleware.

Middleware installer

To install the prerequisite middleware products for Maximo Asset Management, follow these steps:

1. Log on as a user with administrative authority.

2. Launch the middleware installer from the launchpad (included on the distribution media).

   On Microsoft Windows, navigate to the root directory of the product disc or the downloaded installation image, and run the following command:

   ```
   launchpad.exe
   ```

   On Linux and other platforms, the program is named similarly.

   On Linux navigate to the root directory of the downloaded installation image and run following commands:

   ```
   chmod -R +755 *
   ./launchpad.sh
   ```

   Important: The launchpad can be run on either Microsoft Windows or Linux. However, only the options to install the middleware function on both platforms. The options for installing the Maximo components work only from the launchpad when running on Windows.

   Basically, the deployment of the Maximo components requires the dynamic building of EAR and other support files. These processes currently work only on a Microsoft Windows platform. If they are to be installed on a Linux target, the launchpad (running on Windows) communicates to the target system and directs the installation of the EAR and other files through a remote connection.

   After completing the previously listed steps, a series of windows is displayed - for example, a language-selection window, welcome window, and license agreement window.

   Be aware that the installation images for the middleware must be available and may be located on another DVD within your product media. During this installation process, you are prompted for the location of the middleware images.You also are prompted for information such as the workspace to be used by the middleware installer.

3. From the Deployment Choices panel, as shown in Figure 5-2 on page 69, select the features to deploy on this machine and then click **Next**. Your choices of features include the following:

   – **Database server:** The Maximo Asset Management database is used to store information about assets.

- **J2EE server:** The J2EE server is used to host and manage the Maximo Asset Management application. If you choose to install only the J2EE server portion of the middleware, you are prompted to supply the directory server you plan to use to secure it. Your choices are to secure with an existing instance of IBM Tivoli Directory Server or an existing instance of Microsoft Active Directory.

- **Directory server:** Data maintained by the directory server is used to secure Maximo Asset Management.
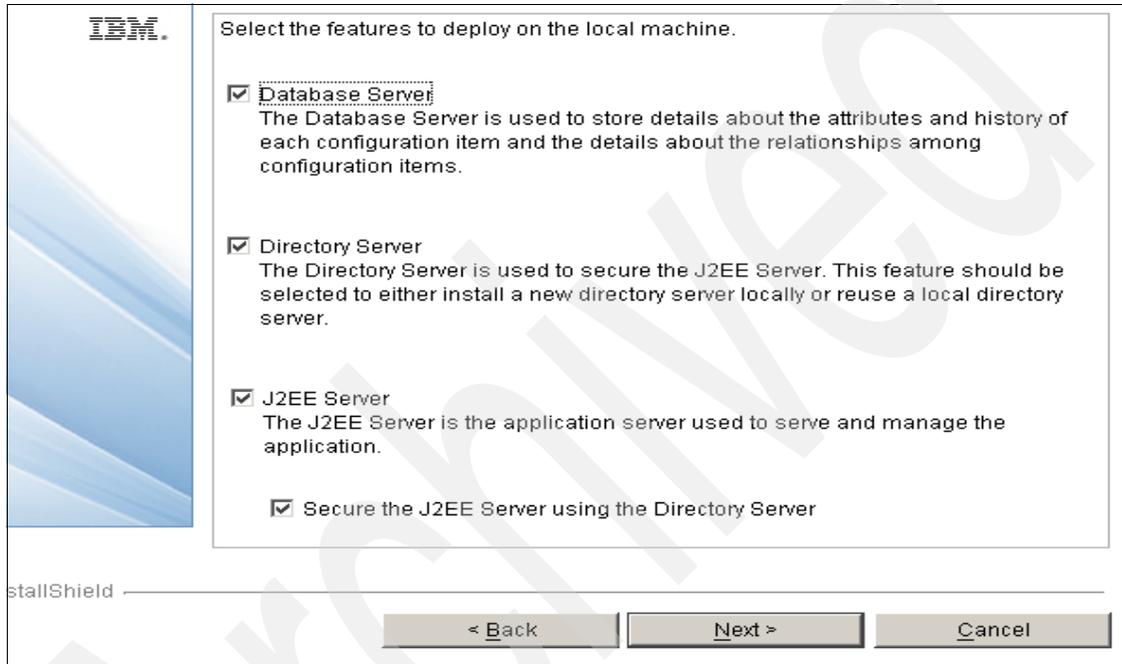
Figure 5-2   Middleware features selection

> **Note:** For a multiple server environment, you can launch this installer on separate systems and choose to install individual middleware components on the individual systems.

4. If you are not using IBM Tivoli Directory Server or not implementing directory services, deselect the **Directory Server** check box.

5. From the deployment plan summary window (see Figure 5-3), click **Next** to configure the parameters displayed. The deployment plan is generated, and you are provided details about the plan.
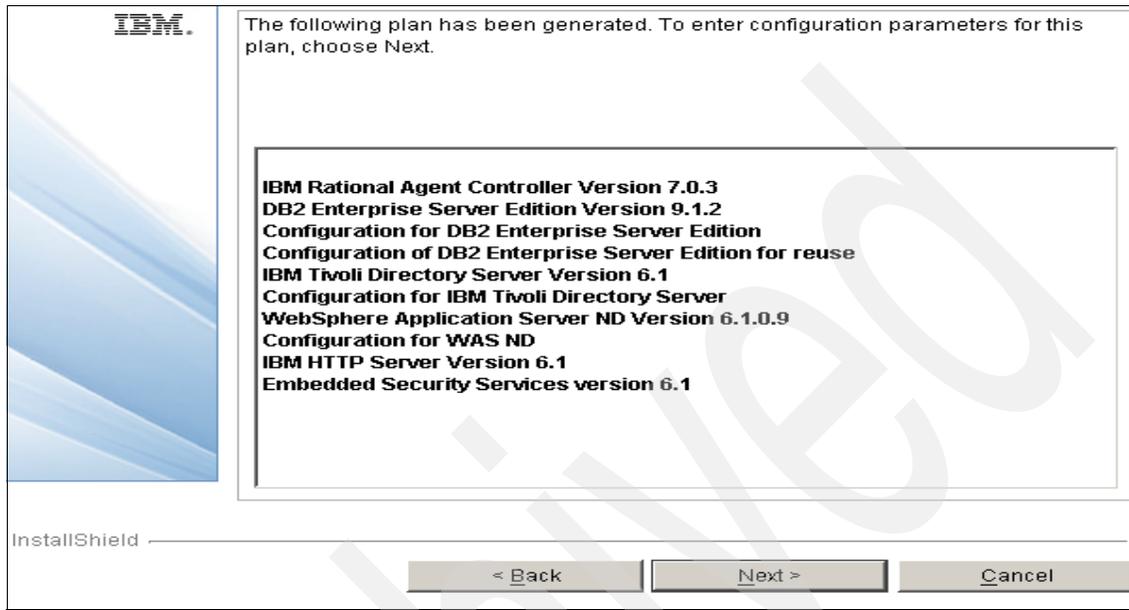
Figure 5-3   Deployment plan with directory services listed

6. From the Credentials panel, enter the user name and password you intend to use to 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 middleware installer. This provides a common password for all middleware components installed through this process.

Figure 5-4 shows the password selection process.

Figure 5-4   Middleware password selection

At this point, you are presented with a series of filled-in panels with default values for each of the middleware components you are installing. For simple environments, you can accept the defaults and move through these panels quickly. If you have special requirements, such as nonstandard port numbers or user IDs, you can make the appropriate changes. Table 5-1 includes a summary of the parameters.

Table 5-1   Summary of middleware component parameters

Parameter	Default value (if any)
Install location	C:\Program Files\IBM\SQQLIB
DB2 Administration Server username	db2admin
DB2 Administration Server password	
Fenced user (Linux only)	
Maximo Asset Management database instance name	ctginst1
Database port	50005
Database instance username	

Parameter	Default value (if any)
Instance user name password	
DB2 administrators group	DB2ADMNS
DB2 users group (Windows only)	DB2USERS
Location to install IBM Tivoli Directory Server	C:\Program Files\IBM\LDAP\V6.1 Default is /opt/IBM/ldap/V6.1
Administrator distinguished name	Default for all platforms is cn=root
Administrator password	
Organizational unit	Default for all platforms is ou=SWG
Organization and country suffix	Default for all platforms is o=IBM,c=US
Directory server port	Default for all platforms is 389
Directory server secure port	Default for all platforms is 636
Administration port	Default for all platforms is 3538
Administration secure port	Default for all platforms is 3539
TDS Database name	Default for all platforms is security
Instance name	Default for all platforms is idsccmdb
Port	Default for all platforms is 50006
Instance user password	
LDAP Host Name	
Directory server port	Default is 389.
LDAP base entity	Default is ou=SWG,o=IBM,c=US
User suffix	Default is ou=users,ou=SWG,o=IBM,c=US
Group suffix	Default is ou=groups,ou=SWG,o=IBM,c=US
Organization	Default is ou=SWG,o=IBM,c=US
WebSphere Application Server Bind distinguished name	Default is cn=root
Bind password	

Parameter	Default value (if any)
Install location	Windows: Default is C:\Program Files\IBM\WebSphere\AppServer Linux: Default is /opt/IBM/WebSphere/AppServer
WebSphere Administration username	Default for all platforms is wasadmin
WebSphere Administration password	
Deployment Manager profile name	Default for all platforms is ctgDmgr01
Application server profile name	Default for all platforms is ctgAppSrv01
Cell name	Default for all platforms is ctgCell01
Deployment Manager node name	Default for all platforms is ctgCellManager01
Application server node name	Default for all platforms is ctgNode01.
Update Installer install location	Windows: Default is C:\Program Files\IBM\WebSphere\UpdateInstaller Linux: Default is /opt/IBM/WebSphere/UpdateInstaller
HTTP Server	Windows: Default is C:\Program Files\IBM\HTTPServer Linux: Default is /opt/IBM/HTTPServer
HTTP port	Default for all platforms is 80
Admin Server port	Default for all platforms is 8008
HTP server Profile name	Default for all platforms is ctgAppSvr01 - this value cannot be changed.
IBM Agent Controller Install location	Windows: Default is C:\Program Files\IBM\AgentController Linux: Default is /opt/IBM/AgentController

7. Specify the location of the Maximo Asset Management middleware images, and then click **Next**.

   – Copy the middleware installation images from the source media to a specified directory. Select this option to copy the Maximo Asset Management middleware images from the product media to a directory that you specify.

   – Specify a directory containing all the required middleware install images (see Figure 5-5 on page 74). Select this option if you intend to specify a

file system directory that already contains all of the Maximo Asset Management middleware installation images.
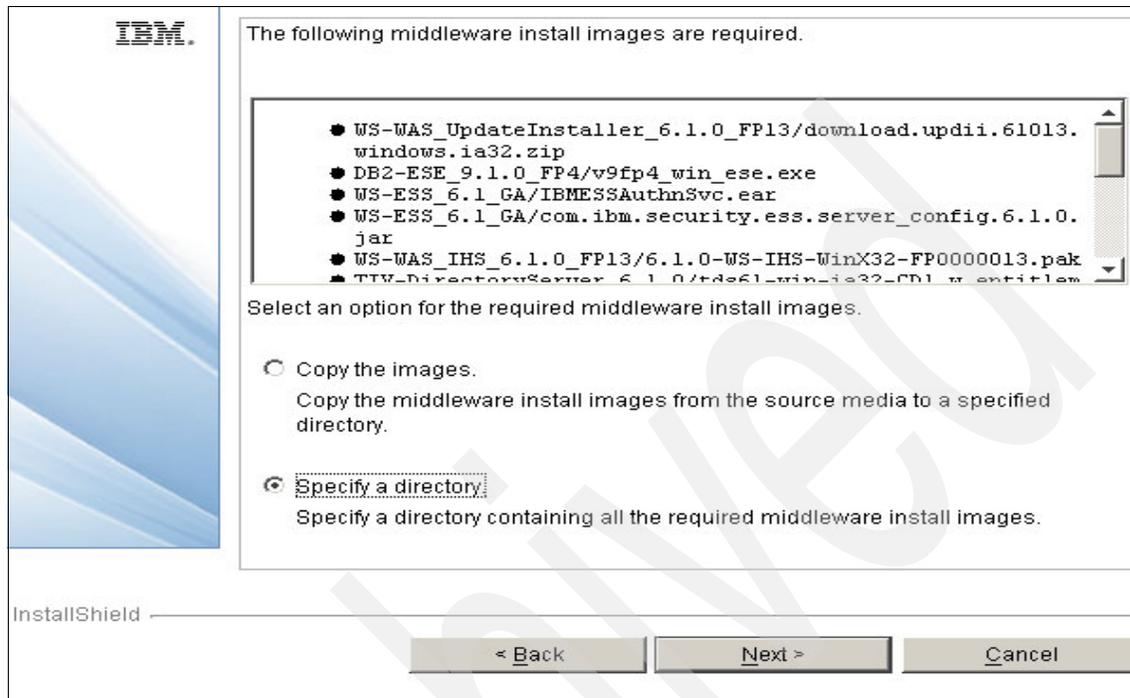
Figure 5-5   Middleware install image

8. For Linux only, the middleware installer creates a number of temporary files and extracts the middleware images to a temporary directory. Specify a directory or accept the default /tmp.

Tip: Make sure /tmp or another specified location has sufficient disk space.

9. From the Deployment Plan Operation panel, select **Deploy the plan**, and then click **Next**. You can also elect to make changes to the deployment plan

or parameters you have previously configured from this panel (see Figure 5-6).
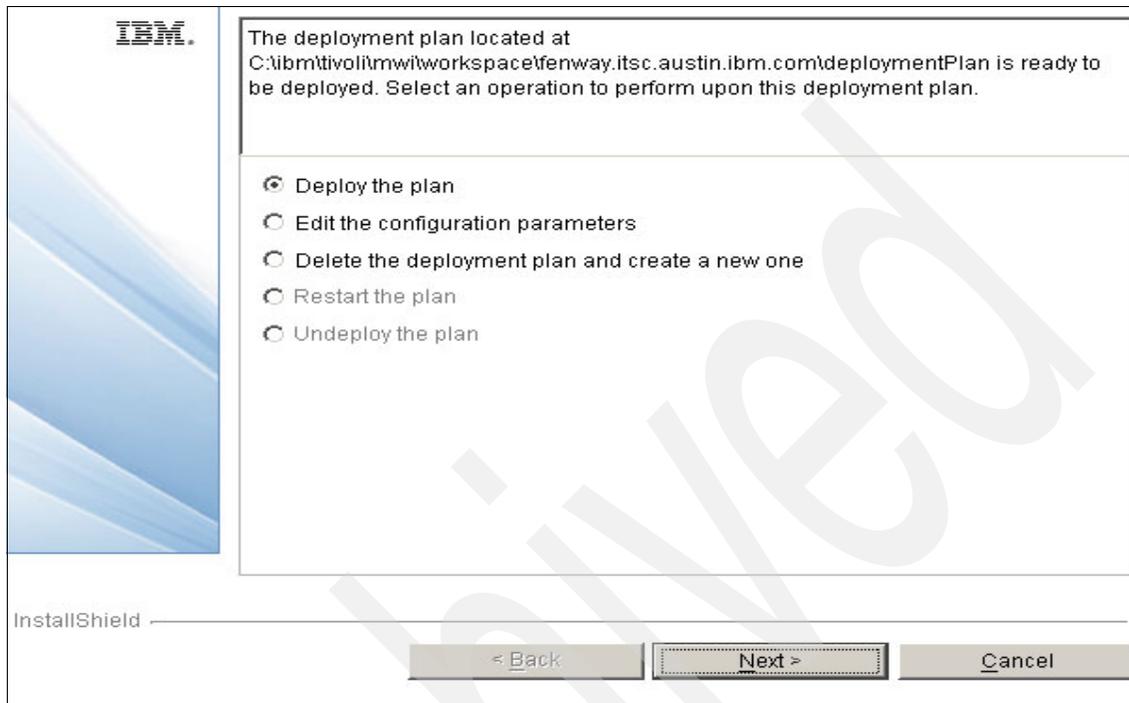
Figure 5-6   Manage middleware deployment plan

10.From the Deployment Plan and Parameter Configuration summary panel (see Figure 5-7), review the contents of the summary, and then click **Deploy** to initiate the installation and configuration of the middleware you selected.
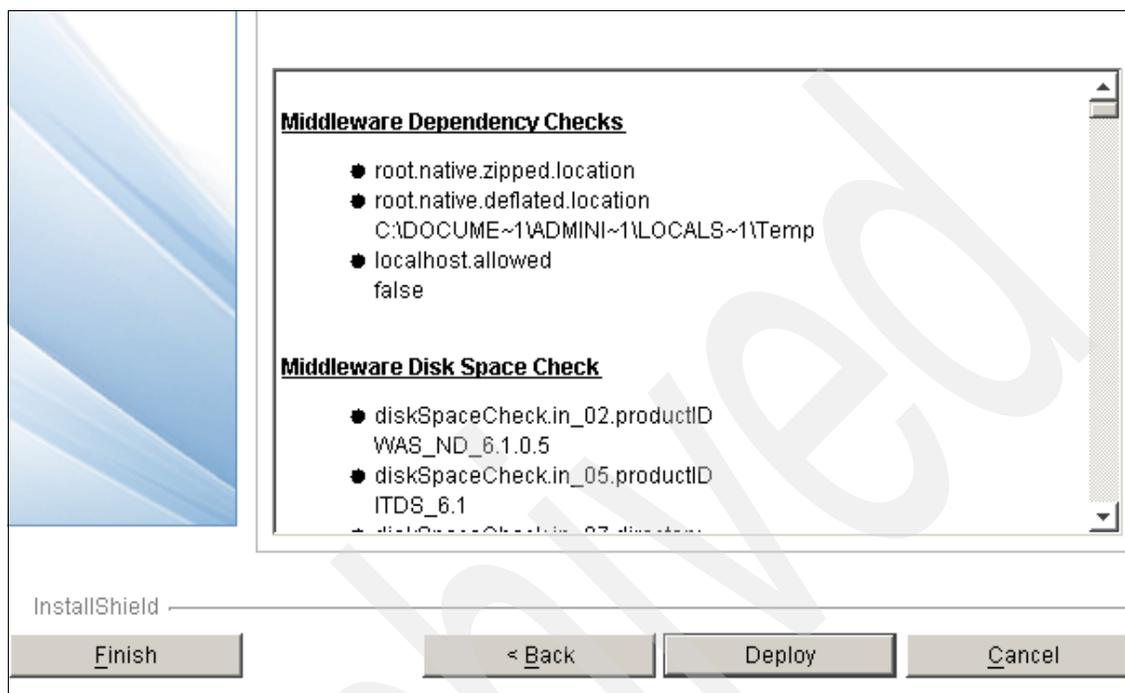
Figure 5-7   Middleware deployment plan

11.After the deployment completes successfully, click **Finish** to exit (see Figure 5-8).
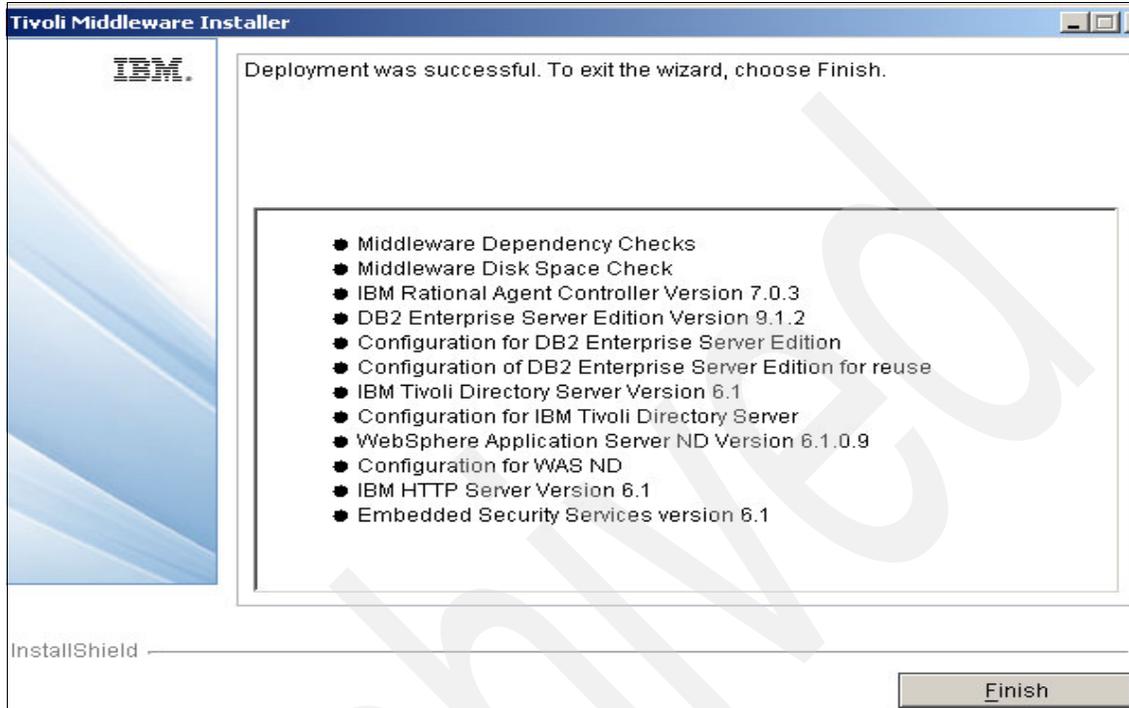
Figure 5-8   Middleware installation completed

5.2.2 Middleware installer logs

Middleware installer log files are located in the workspace directory that was defined in the middleware installer. The different types of log files are described in the sections that follow.

User interface logs

The logs generated by the middleware installer user interface are located in the workspace directory. The mwi.log file is a high-level log file that was generated by the most recent invocation of the middleware installer. If an error occurs, examine this log file first. An entry in this log file may direct you to a lower-level log file. Log

files named mwi.logX, where X is a number, are copies of the mwi.log file from earlier invocations of the middleware installer So, for example, mwi.log0 is produced after the first invocation of middleware installer, mwi.log1 is produced after the second invocation of 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 middleware installer also performs several system checks. Examples of system checks run by the user interface include the following:

► Dependency checking to ensure the operating system meets deployment requirements

► Inventorying the software on the system to locate existing instances of middleware products deployed by the middleware installer

► Checking the available disk space to ensure it is sufficient for the deployment

Each of these 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 run as part of the deployment plan.

Deployment plan logs

The deployment plan is located in the directory <workspace directory>/<host name>/deploymentPlan, where host name is the host name of the current system. Each time 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.

Machine plan logs

The machine plan is located in the directory <workspace directory>/<host name>/deploymentPlan/MachinePlan_<host name>. The log files for the machine plan are located in the logs subdirectory. The primary log files for the machine plan are named MachinePlan_<host name>_processID. These log files contain the output generated by ANT when running the machine plan ANT script.

Deployment plan logs

Each step in the deployment plan is located in a directory named <workspace directory>/<host name>/deploymentPlan/MachinePlan_host name/ stepNum_stepID where stepNum is the sequence number of this step in the installation processing order of the deployment plan and stepID identifies the step. The log files for the step are located in the logs subdirectory.

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

Logs for sub-steps

Each step contains one or more sub-steps. The sub-steps perform the actual installation and uninstallation, and then the sub-steps check work for the middleware installer.

Each sub-step is located in the directory <workspace directory>/<host name>/deploymentPlan/MachinePlan_hostname/stepNum_stepID/operation/sub stepNum_substepID, where operation is the ANT target in the step ANT script that invokes this sub-step. 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 the processID/logs subdirectory. Log files generated by the native middleware installation programs are also kept here.

5.3 Installing Maximo base services

This section describes the process of installing Tivoli's process automation engine through the launchpad.

5.3.1 Maximo Asset Management installation overview

This procedure explains how to use the Maximo Asset Management installation program to install Maximo. In addition to configuring new instances of Maximo middleware products installed by the middleware installer, the Maximo Asset Management installation program can configure existing instances of prerequisite products, including those from other vendors, that you wish to use with Maximo.

The instructions provided here are for either a single or multiple machine installation using default values. The instructions assume that you choose to have the Maximo Asset Management installation program automatically configure middleware across multiple machines to work with Maximo.

If you do not allow the Maximo Asset Management installation program to automatically configure middleware, it still performs programmatic checks to verify that the documented manual steps were performed properly.

If any errors are encountered, a dialog box detailing the error is displayed. You are not permitted to continue in the Maximo Asset Management installation task until you resolve the errors. The Maximo Asset Management installation program can be run only from a Microsoft Windows-based system.

5.3.2 Maximo Asset Management installation

Avoid using localhost for host name values in the installation program. Specify the actual fully qualified host name of the system for all host name values. To install Maximo Asset Management base services, follow these steps:

1. Log on as a user with administrative authority. Launch the middleware installer from the launchpad (see Figure 5-9 on page 81). In the launchpad navigation pane, click **Install the Product** → **Maximo Asset Management.**
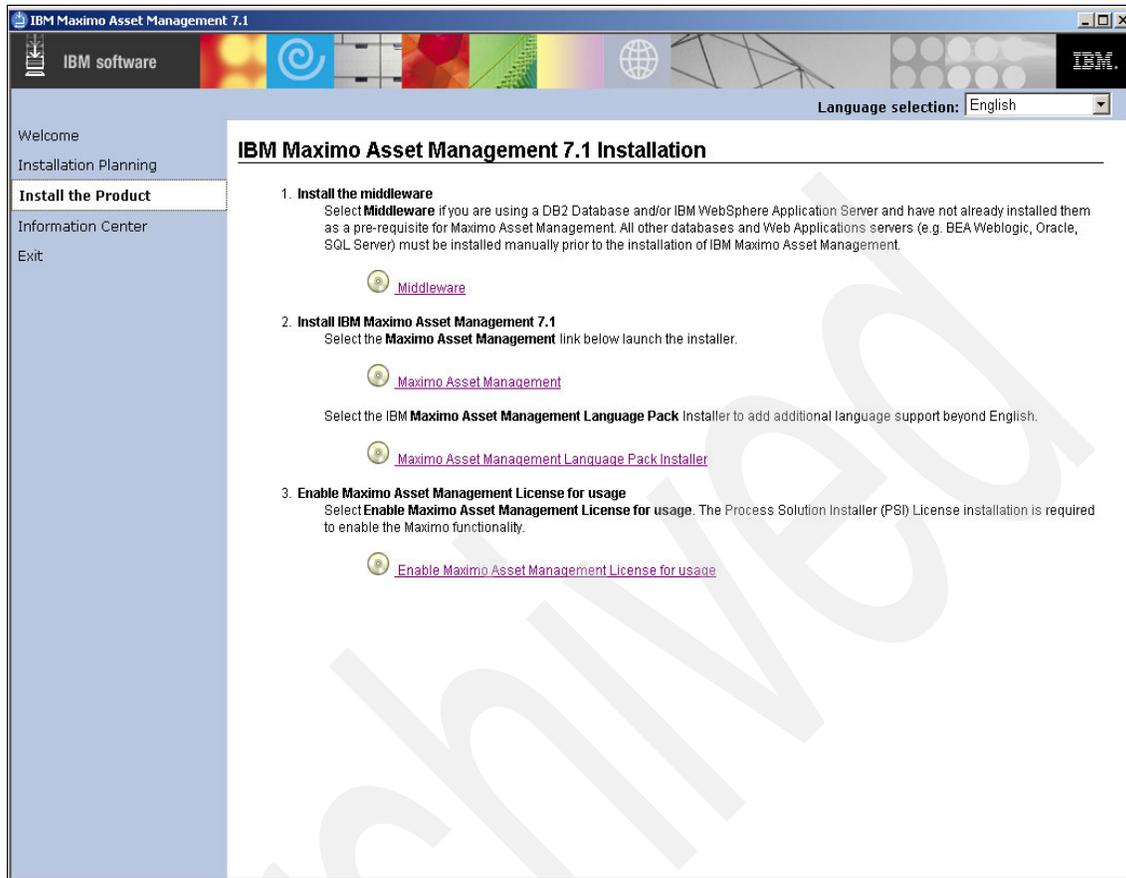
Figure 5-9   Launchpad

2. You are presented with a series of dialogs prompting you for parameters related to your middleware environment. If you used the middleware installer, most of these parameters are pre-filled. If you manually installed the middleware, you must provide the appropriate values for each of the middleware components.

   A special installation application (Install Anywhere) is installed and initiated for installing the rest of the components. Install Anywhere is based on IBM Autonomic installation technology, which helps drive complex, multistep installations.

3. You are prompted for information related to your middleware environment. If you used the middleware installer, you are prompted to use the configuration

information that was saved by the middleware installer to automatically pre-fill many of these parameters.

From the Import Middleware Configuration Information panel, specify that you want to use the field values you input into the middleware installer as default values for those same fields in the Maximo Asset Management installation program. The middleware default information is not used if you select the Simple deployment path. The field values are as follows:

- **Host name:** Enter the host name of the system where the middleware installer was run.

- **User ID**: Enter the user ID that was used to run the middleware installer.

- **Password:** Enter the password of the user ID that was used to run the middleware installer.

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

4. From the Choose Deployment panel, select the **Custom deployment** topology, and then click **Next**.

Select **Simple** if you want to deploy all Maximo Asset Management components on a single system. This deployment option is typically used only for demonstration, proof-of-concept, or training purposes.

Select **Custom** if you wish to deploy Maximo Asset Management components across several systems. This deployment option is typically used in a production environment. In our example, we chose Custom to enable us to specify a different host name for the database server, as though we were installing in a multiserver environment (see Table 5-2).

Table 5-2   Maximo installation components

Parameter	By default, this value is . . .	Description
Choose installation folder	C:\IBM\SMP	Installation location.
Maximo database type	DB2	Select the database vendor or product. Each database has its own unique set of configurable parameters and values.
DB2 host name		Enter the host name of the machine hosting DB2.The host name must be fully qualified.
Port	The default is 50005	

Parameter	By default, this value is . . .	Description
Database name	Default database name is maxdb71	The database is created if it does not already exist.
Instance		Enter the name of the database instance to be used with Maximo. After you have entered configuration information for the database that was selected, the Maximo Asset Management installation program connects to the database server to validate the information you entered.
Database user ID		
Database password		

5. From the Automate Database Configuration panel, select **Automate database configuration** if you want the database automatically created, and then click **Next** (see Figure 5-10).
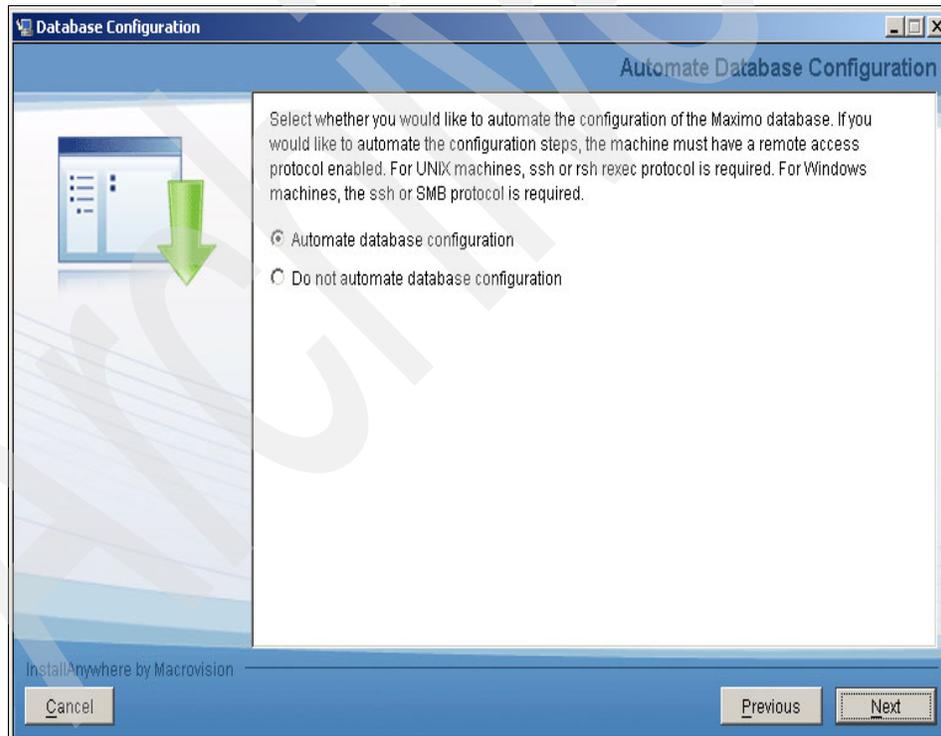
Figure 5-10   Automate Database Configuration

If you choose not to automate database creation, this step assumes you have already created a database instance, a database, tablespaces, a user, and schema for use with Maximo Asset Management. If you have not manually configured the database prior to selecting **Do not automate database configuration** from within the Maximo Asset Management installation program, the installation checks to determine whether you have completed these pre-install tasks, and you are reminded to complete them prior to restarting the Maximo Asset Management installation program.

To manually configure the database, refer to Appendix A, "Manual configuring middleware" on page 249.

6. Enter the DB2 database information (see Figure 5-11). Enter the Windows services user ID and passwords only if the DB2 server is installed on a Windows machine, and click **Next**.

Figure 5-11   DB2 Administration

7. From the DB2 Tablespace panel, specify the tablespace configuration properties if you want the DB2 database automatically created (see Figure 5-12).
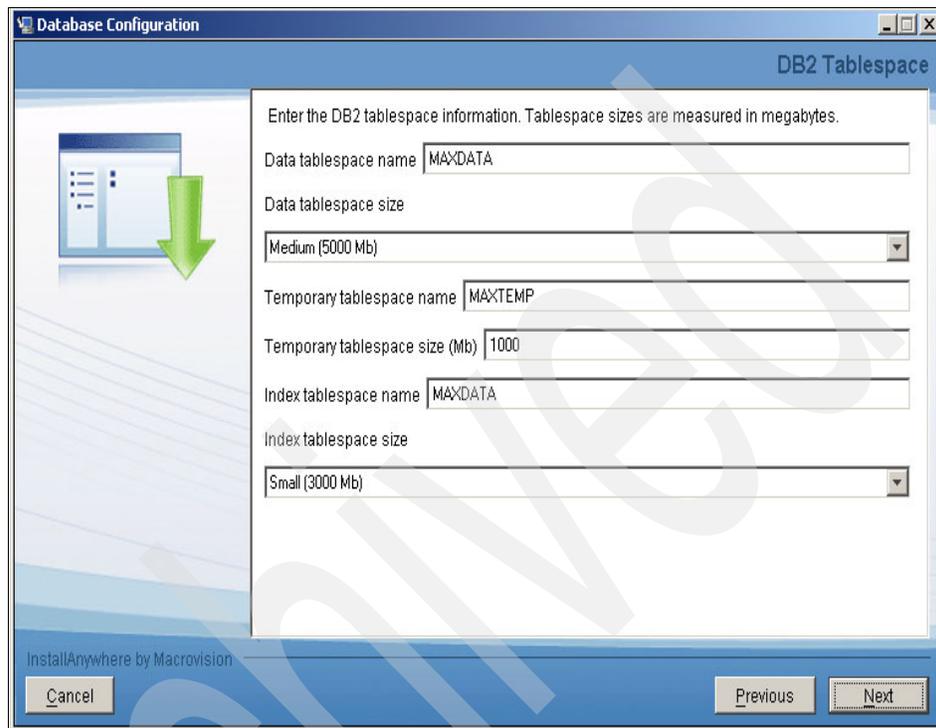
Figure 5-12   DB2 Tablespace

8. From the Maximo Application Server Type panel, select the relevant application server you wish to deploy your Maximo application on, choose **IBM WebSphere Application Server**, and click **Next** (see Figure 5-13).
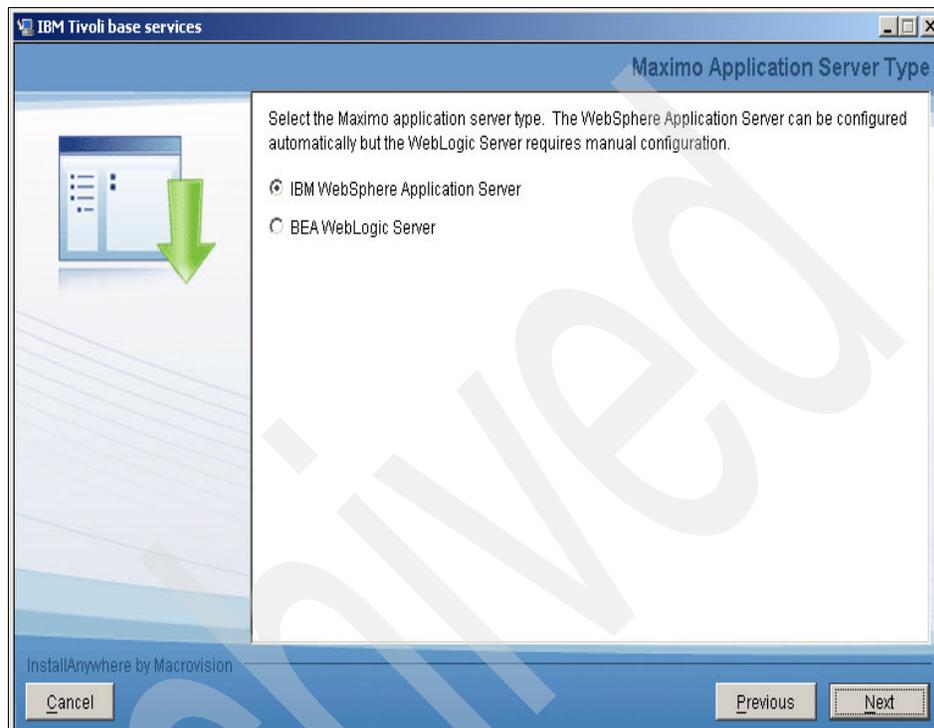
Figure 5-13   Maximo Application Server Type

9. From the **WebSphere Connectivity** panel, enter host information about the WebSphere Application Server, and then click **Next** (Figure 5-14 on page 87).

   – **Host name:** Enter the fully qualified host name of the system hosting WebSphere Application Server. Alternatively, you can provide the IP address for the system.

   – **SOAP port:** Enter the SOAP port of the WebSphere Application Server system. The default value for this field is 8879.
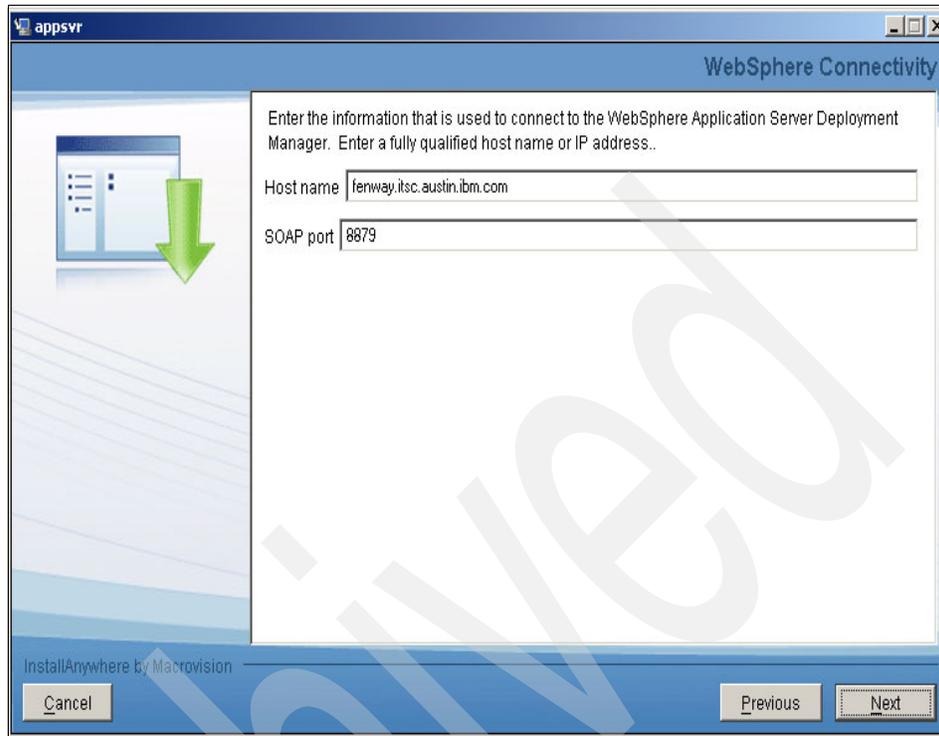
Figure 5-14   WebSphere Connectivity

10. From the Remote Access Authorization panel (see Figure 5-15 on page 88), enter authorization information for the WebSphere Application Server configuration, and then click **Next**.

   Operating system user ID: Enter a valid user ID that enables the Maximo Asset Management installation program to access the system that is hosting WebSphere Application Server. This user ID should have administrative rights on the machine you are accessing.

   Operating system password: Enter the password for the system user ID.
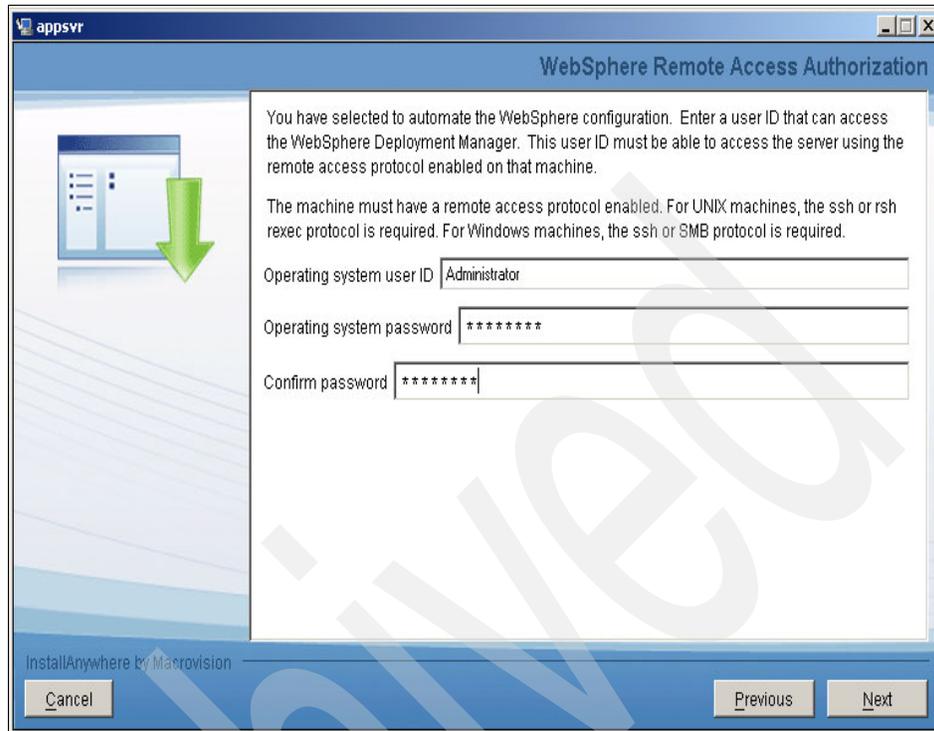
Figure 5-15   WebSphere Remote Access Authorization

11. From the Automate WebSphere Configuration panel, select **Automate WebSphere configuration**, and then click **Next** (see Figure 5-16 on page 89). If you choose not to have the Maximo Asset Management installation program automatically configure the middleware, you must have configured WebSphere manually prior to the installation of Maximo. Configuration tasks include creating a profile, running WebSphere Application Server as a Windows service, copying the WebSphere keystore file from the machine where WebSphere Application Server is installed to the administrative workstation, setting up JMS queues, and so on.

Refer to "Manually configuring the J2EE server" on page 256 for information about manually configuring the J2EE server.
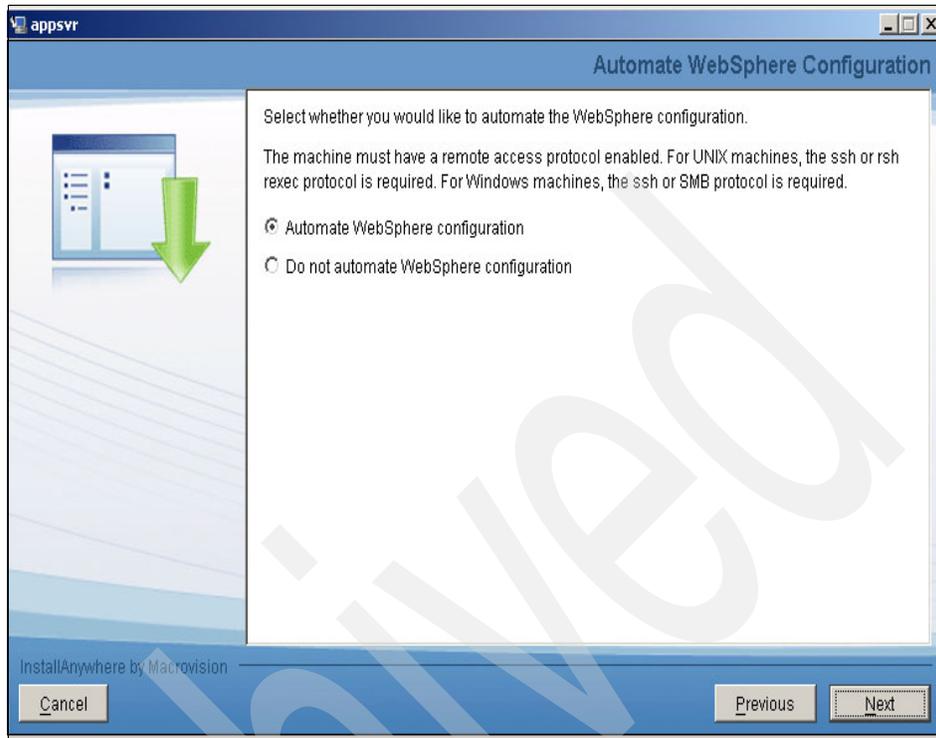
Figure 5-16   Automate WebSphere Configuration

12. From the WebSphere Deployment Manager Configuration panel (see
Figure 5-17 on page 90), enter values for the following fields, and then click
Next.

– **WebSphere installation directory**: Enter the directory where WebSphere
Application Server is installed on the host system.

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

- On Linux, this value might be /opt/IBM/WebSphere/AppServer.

- On AIX, this value might be /usr/AppServer.

- On HP-UX, this value might be /AppServer.

- On Sun Solaris, this value mightAppServer.

– **User:** Enter the administrative user ID used to install WebSphere
Application Server The default for all platforms is wasadmin.

    – **Password**: Enter the password for the administrative WebSphere server.

    – **Profile name**: Enter the name of the WebSphere Application Server profile. The default for all platforms is ctgDmgr01.

Figure 5-17   WebSphere Deployment Manager Configuration

13.From the WebSphere Application Server Configuration panel (see Figure 5-18 on page 91), enter the following information, and then click **Next.**

    – **Web server port**: Enter the Web server port used by WebSphere Application Server. 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 are created if they do not exist.
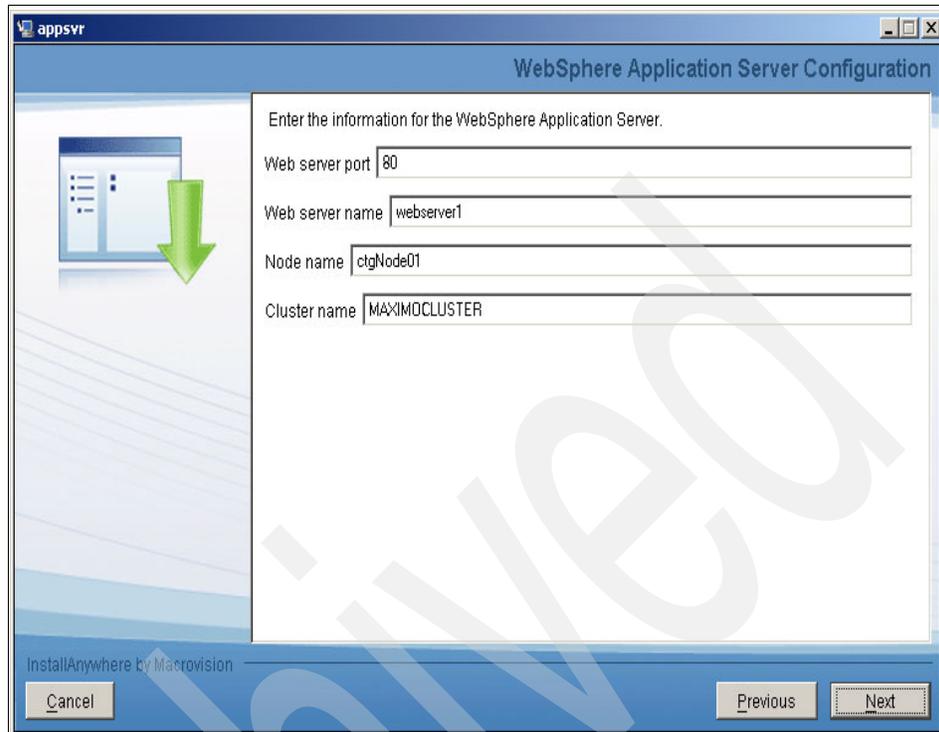
Figure 5-18 WebSphere Application Server Configuration

14.From the Security panel (Figure 5-19 on page 92), indicate whether application server security should be enabled automatically, and then click **Next.**

Note: This step is applicable only if you are implementing the Tivoli Directory Server or Microsoft Active Directory Server for security.
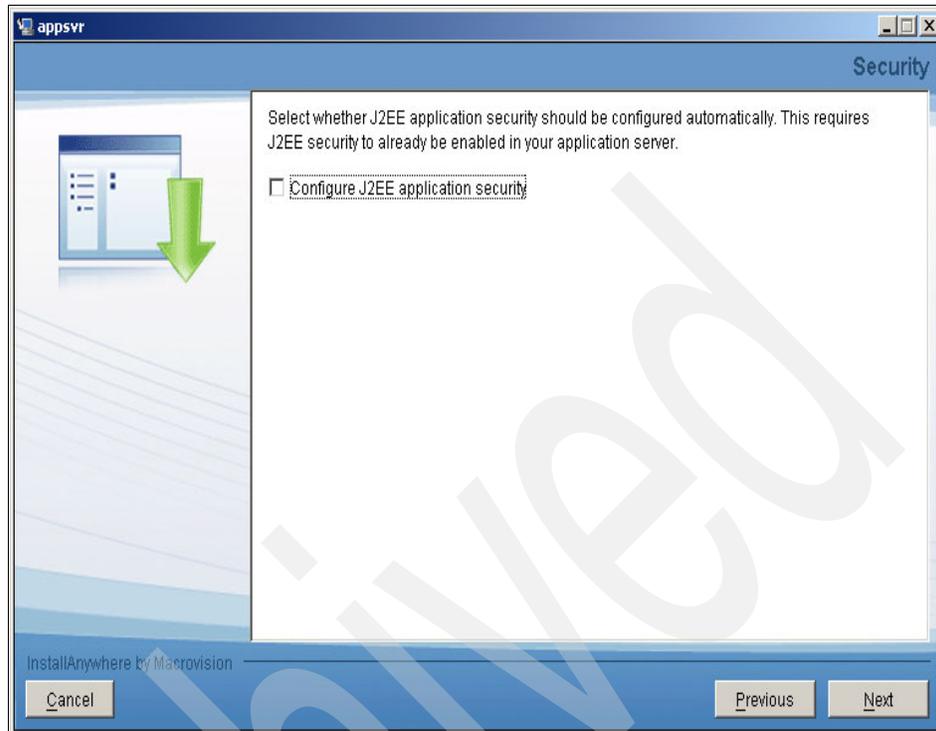
Figure 5-19   Security

15. From the Integration Adapter JMS Configuration panel (see Figure 5-20), 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.

– **Persist JMS messages**: Select this option if you want the Maximo Asset Management installation program to set the JMS implementation to persist messages.

– **Do not persist JMS messages:** Select this option if you do not want the Maximo Asset Management installation program to set the JMS implementation to persist messages automatically. When you select this option, a database is not used to persist messages. If you later decide that you want to persist JMS messages, you must configure the JMS implementation manually.
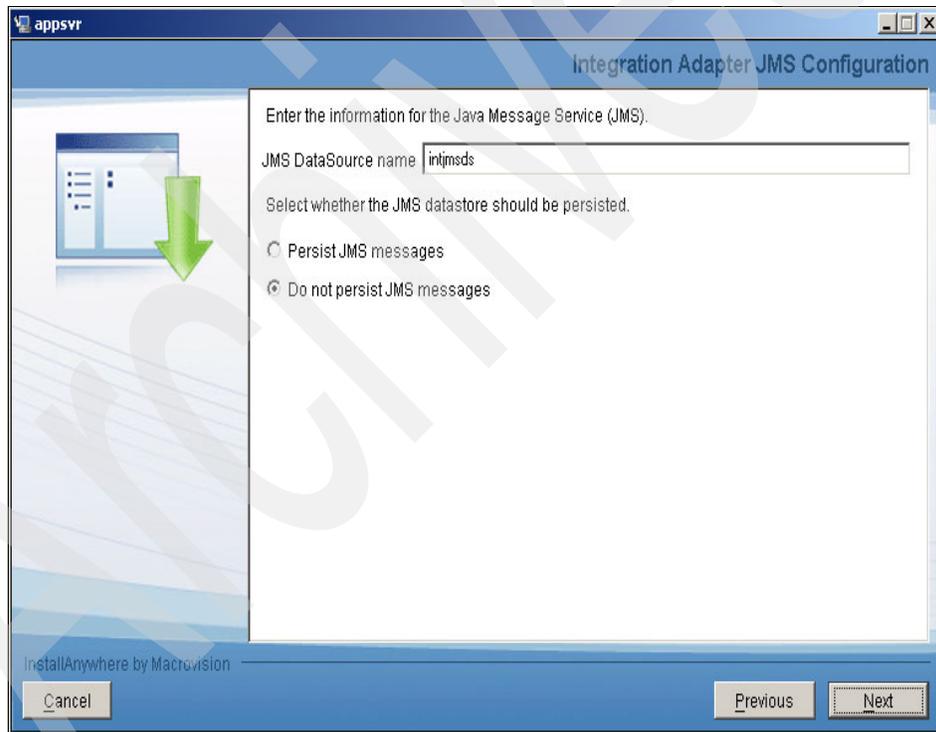
Figure 5-20   Integration Adapter JMS Configuration

To manually configure the database, refer to "Manually creating a datasource for the persistence store" on page 291.

16.From the Input Summary panel (see Figure 5-21), review the information you have provided to the Maximo Asset Management installation program, and then click **Next**.
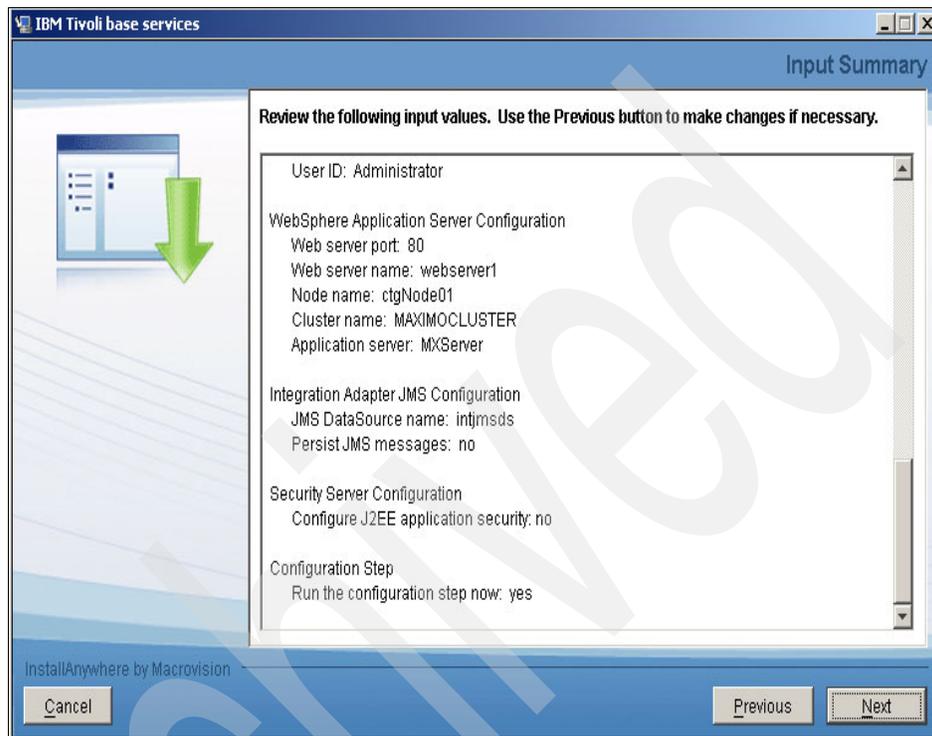
Figure 5-21   Input Summary

17.From the Pre-Installation Summary panel (see Figure 5-22 on page 95), review the installation information, and then click **Install**. The installation task begins. Progress can be monitored by viewing messages displayed above the progress bar. Click **Next**.

At this point, the installation takes place and may take quite some time. (The duration of the installation is mostly dependent on the physical capacity of the server you are installing the base services on.) In our test environment, installation took one to two hours.
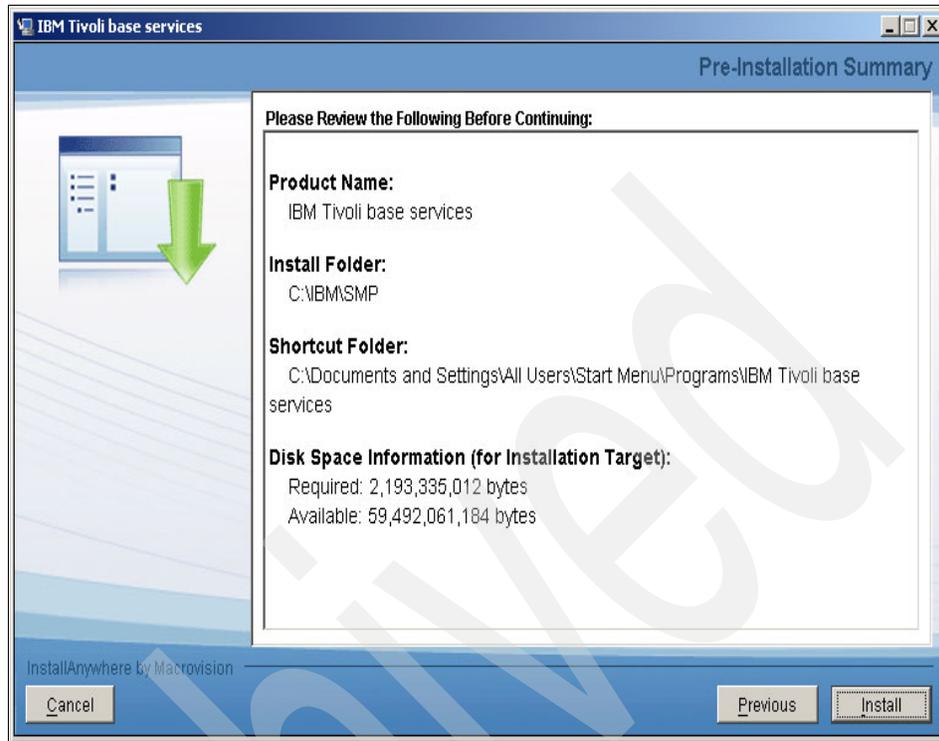
Figure 5-22   Pre-Installation Summary

18. You are prompted to install additional Language Packs over and above the base language (see Figure 5-23). (Configuring Language Packs is discussed in "Language pack installation" on page 113.) Click **No.**
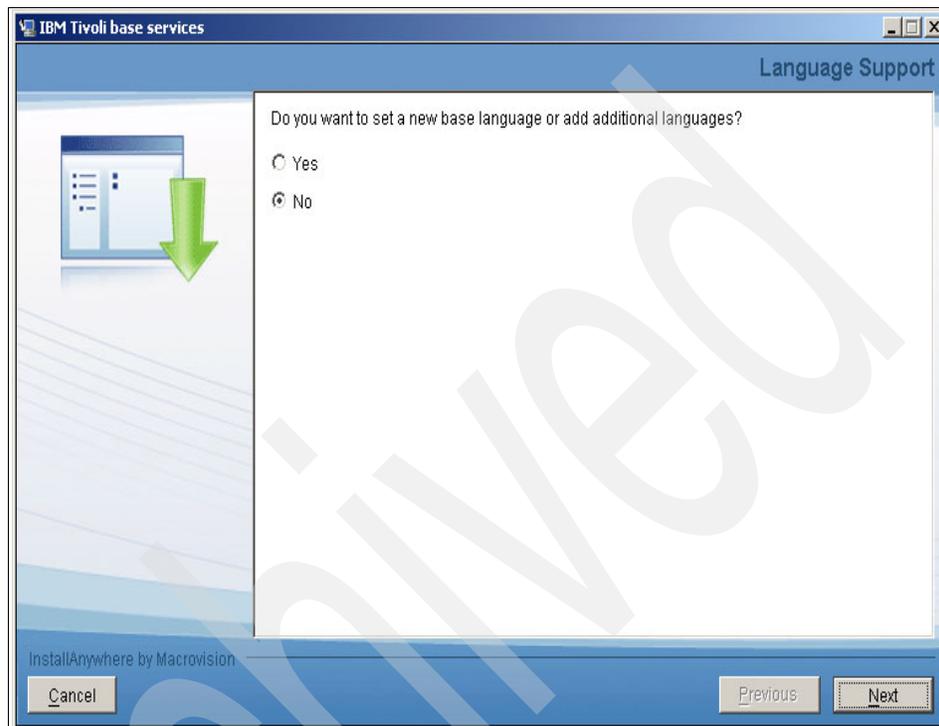
Figure 5-23   Language Support

19.From the Install Complete panel (see Figure 5-24), click **Done**. When the Maximo Asset Management installation program has completed installation and configuration tasks, it exits. Logs can be found at <MAM_Home>/logs.
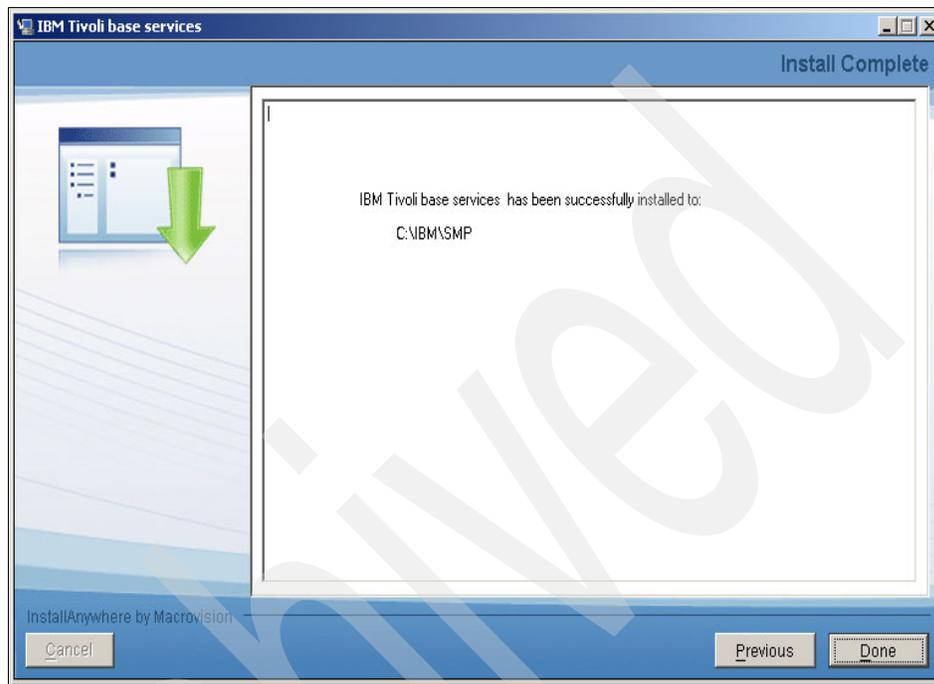
Figure 5-24   Install Complete

5.3.3 After Maximo Asset Management base services installation

After you have installed Maximo Asset Management base services, you must complete the following steps:

1. Before logging on to the newly installed Maximo Asset Management application, access the Microsoft Windows Services and ensure that the relevant services are started. These services are the following:

    – DB2 COPY Services
    – DB2 Governor
    – DB2 License Server
    – DB2 Management Service
    – DB2 Remote Command Server
    – DB2 Security Server
    – DB2DAS
    – IBM HTTP Server V6.1
    – IBM HTTP Administration V6.1

- 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

2. Although not required, you can optionally start the node agent as a Windows service.

   To create a node agent as a Windows service, perform the following steps on the WebSphere Application Server:

   a. Open a command prompt.

   b. Ensure the following directory exists or is created:

      ```
      C:\IBM\WebSphere\AppServer\logs\nodeagent
      ```

   c. Change the directory to <WAS_HOME>\bin.

   d. Run the following command (as a single line):

      ```
      WASService -add NodeAgent -serverName nodeagent -profilePath
      C:\IBM\WebSphere\AppServer\profiles\ctgAppSrv01 -wasHome
      C:\IBM\WebSphere\AppServer -logRoot
      C:\IBM\WebSphere\AppServer\logs\nodeagent -logFile
      C:\IBM\WebSphere\AppServer\logs\nodeagent\startServer.log
      -restart true
      ```

   e. Close the command prompt.

   Tip: When implementing the Tivoli Directory Server, if you have not resumed your Tivoli services, you might not be able to access the WebSphere Application Server console because the administrator user name and password have not been validated. Make sure your service is running.

3. When all the services are started, log on to the following URL:

   ```
   http://servername:9080/maximo
   ```

Check whether you are connected to Maximo Asset Management (see Figure 5-25).
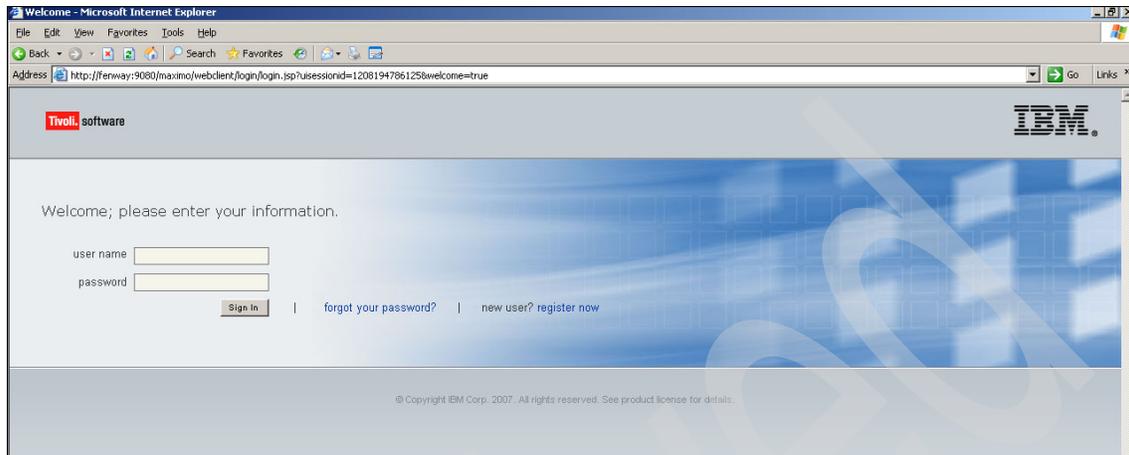
Figure 5-25   Maximo Asset Management logon page

5.4 IBM Agent Controller overview

The IBM Agent Controller imports log and trace information from remote systems to the IBM Log and Trace Analyzer application. To import log and trace data, the IBM Agent Controller must be installed and running on every server from which you wish to retrieve log file information using the Log and Trace Analyzer.

The middleware installer installs the Agent Controller during the installation of the following:

► IBM Tivoli Directory Server
► WebSphere Application Server
► DB2

In some circumstances you must manually invoke the IBM Agent Controller installation program using the launchpad application. For instance, if you are using other supported software such as the following:

► Oracle
► Microsoft Active Directory
► Microsoft SQL

Or if your product has its own log parser, IBM Agent Controller has to be manually installed on that remote location in order to import log and trace data. This is also the case if you have installed a previous version of Agent Controller. If you have a previous version, the installation program prompts you to uninstall that version and invoke a manual installation of the latest IBM Agent Controller.

These situations require that you manually install the IBM Agent Controller:

► You use other supported software.
► The other supported software you are using has its own log parser.
► You have a previous version of IBM Agent Controller installed.

5.4.1 IBM Agent Controller supported platforms

This section provides a list of the supported platforms for installing IBM Agent Controller. You can install IBM Agent Controller on the following supported platforms:

– AIX V5.2, V5.3, and V5L on PowerPC® (32-bit)
– z/OS® V1R4, V1R5, V1R6, and V1R7 on zSeries® (32-bit)
– OS/400® V5R2, V5R3, and V5R4 on iSeries® Linux V2.4 and V2.6 kernel.

Examples of supported distributions include the following:

– Red Hat Linux Advanced Server V2.1 on Intel IA32
– Red Hat Enterprise Linux AS release 3 on PowerPC (64-bit)
– Red Hat Enterprise Linux (RHEL) V3.0 and V4.0
– SuSE Linux Enterprise Server (SLES) V9 on Intel IA32
– SuSE Linux Enterprise Server (SLES) V8 on zSeries (32-bit)
– Windows 2000 Server or Advanced Server (SP4) on Intel IA32

The IBM Agent Controller helps to track and log the installation of various components across servers in your environment as shown in Figure 5-26.
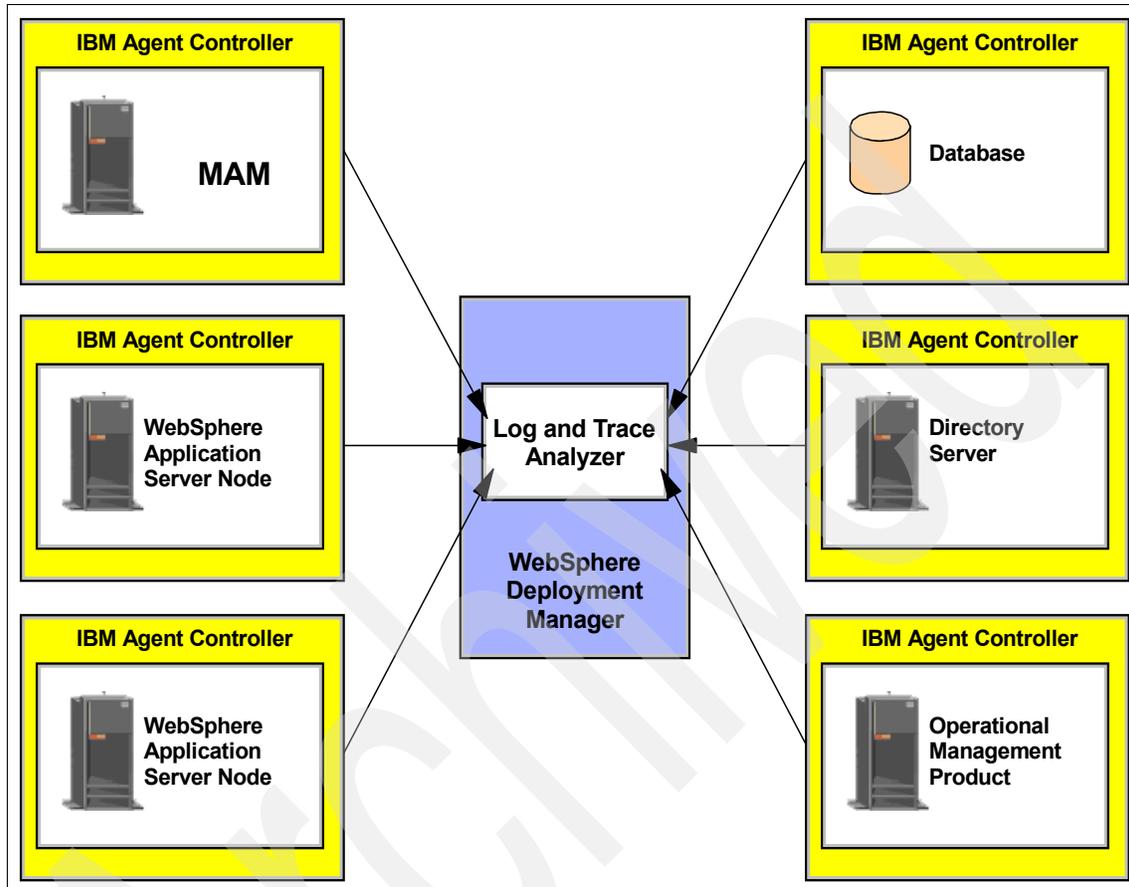
Figure 5-26   IBM Agent Controller

5.5 Process solution package installation

Process solutions are versioned software components. The Process Solution installation programs support a variety of software life cycle operations that may be applied against process solutions.

A Process Solution package is a self-contained ZIP file of installation artifacts and deployment logic. You can deploy it using the Process Solution installation program. Installation artifacts are the files and content installed on your Maximo Asset Management environment to enable the services management

functionality of the Process Manager product or Integration module. The deployment logic consists of actions carried out 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. Additionally, optional sample data can be installed.

The base Install operation installs and deploys a new process solution on your Maximo Asset Management environment. After installation, a process solution may be updated in several ways.

The Process Solution installation programs are able to process the following package types:

► Base install package - Required to install a new process solution using the base install operation.

► Incremental Update package - Required when performing an upgrade operation.

► Fix package - Required when applying an interim fix to a process solution.

► Full update package - Used to perform a base install operation if no instance of the Process Solution is currently installed. In addition, it may be used to perform an upgrade operation on a currently installed Process Solution. The Process Solution installation programs ensure that the appropriate package type is processed for any given operation.

5.5.1 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. You should review the following steps before invoking the Process Solution installation program:

1. Have middleware logon information available.

   The Process Solution installation program requires access to middleware servers to automate the deployment of the Process Solution package. You must know the administrative user IDs and passwords for the impacted middleware servers. The actual middleware servers whose logon information is required depends on the Process Solution package being installed. The Process Solution installation program ensures that required logon information is specified before continuing with the deployment operation.

2. Back up middleware servers and administrative workstation.

   You should create backups for impacted J2EE, database, and directory servers before you deploy a Process Solution package using the Process Solution installation program.

3. Ensure middleware servers are started.

   Start the impacted middleware servers before running the Process Solution installer.

5.5.2 Enabling Maximo Asset Management license

Only after you install the Process Solution package are you entitled to use Maximo Asset Management according to the license you have purchased. Complete the following steps to install a Process Solution package in Maximo using the Process Solution installation wizard:

1. From the launchpad, launch the Process Solution installation program by clicking the **3. Enable Maximo Asset Management License for usage** link (see Figure 5-27). The Process Solution Installation Install Anywhere Installer executes on the Maximo administrative workstation. The launch script is deployed and configured by the Maximo Asset Management installation program.
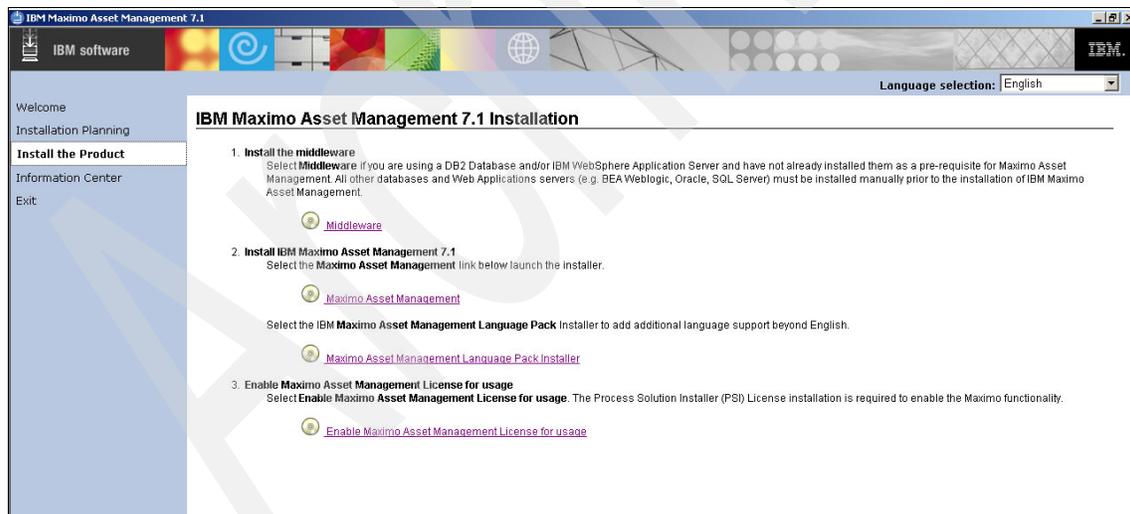
Figure 5-27   Enable Maximo Asset Management license

In our case, the Process Solution Installation package selected is a base install of the Maximo Asset Management V7.1 package. The Process Solution installation program performs a series of validation checks to verify that the

package you selected is valid. The system is checked to ensure that the package has not already been deployed.

2. Enable the package to be validated, click **Next** (see Figure 5-28).
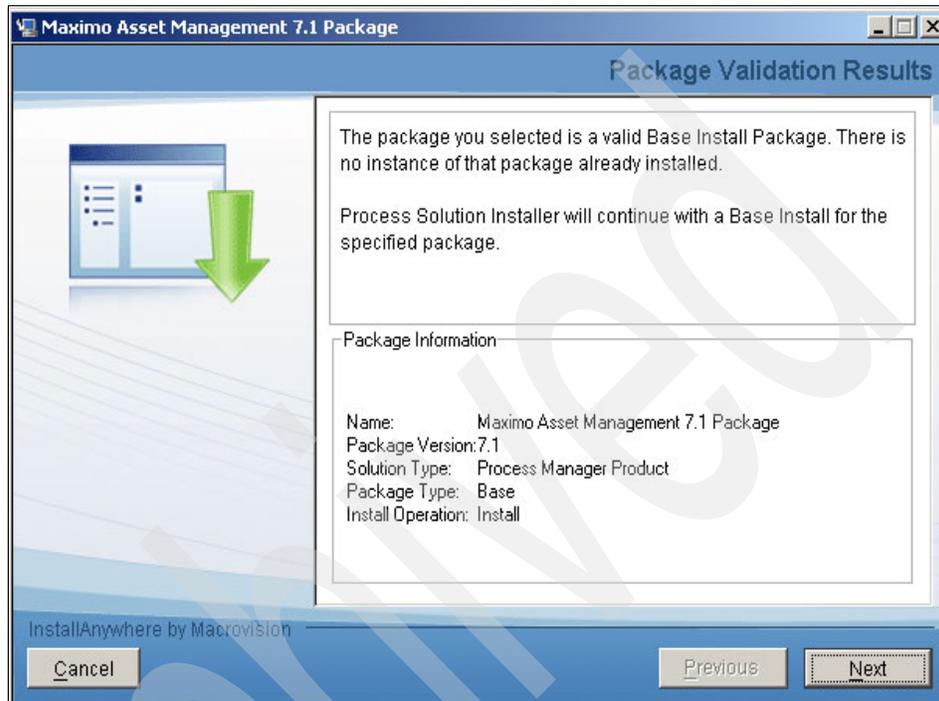
Figure 5-28   Package Validation Results

3. From the Middleware Login Information panel (see Figure 5-29), enter the credentials for which you are being prompted. Then click the **WebSphere Remote** tab and enter a user ID and password (see Figure 5-30 on page 106). The Process Solution installation wizard validates the credentials by connecting to the middleware servers using the supplied credentials. Click **Next**.
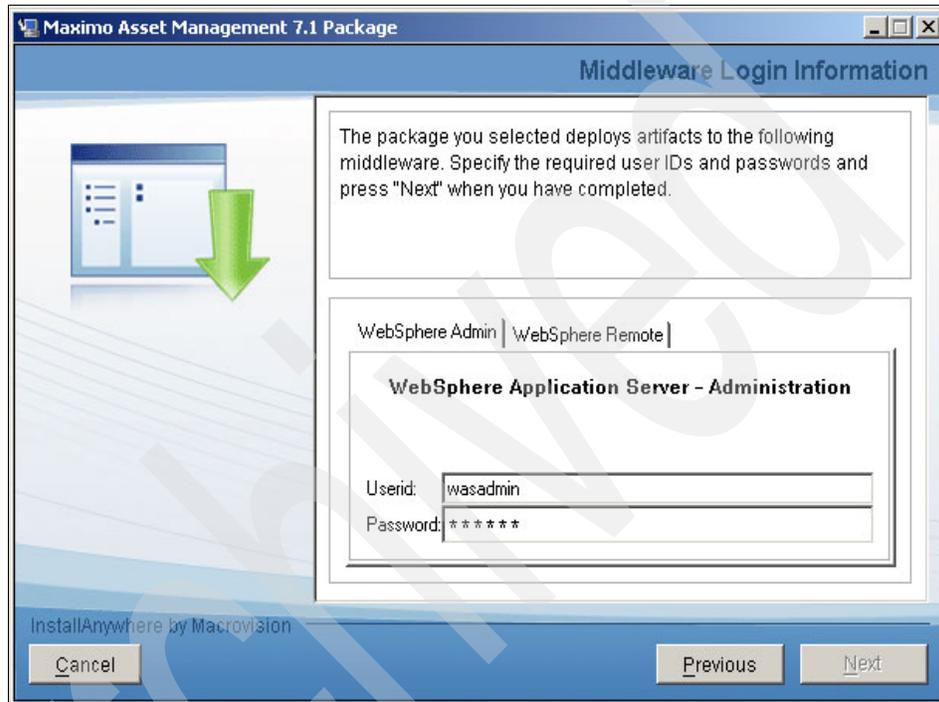
Figure 5-29   Middleware Login information
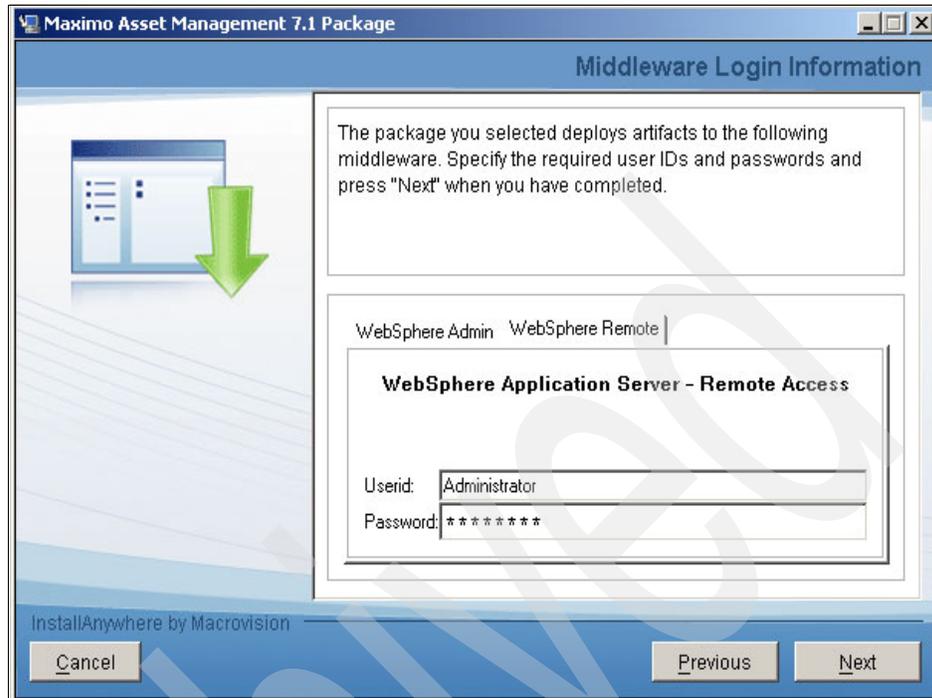
Figure 5-30   Middleware Login information-WebSphere Remote

4. After the credentials have been verified, a Package Options panel (see Figure 5-31 on page 107) is displayed that details the deployment options that the package supports. Leave both the **Supported Package Options** boxes unchecked to deploy the EAR files and update the Maximo database with the Process Solution installation program, and click **Next.**

► Supported package options definitions:

– **Defer Maximo Application Redeployment**

If you intend to apply more than one package solution, select this check box. By doing so you have to redeploy the EAR files only once after all the package solutions have been installed.

– **Defer the update of the Maximo Database**

If you intend to apply more than one package solution, select this check box. By doing so you have to update the database only once after all the package solutions have been installed.
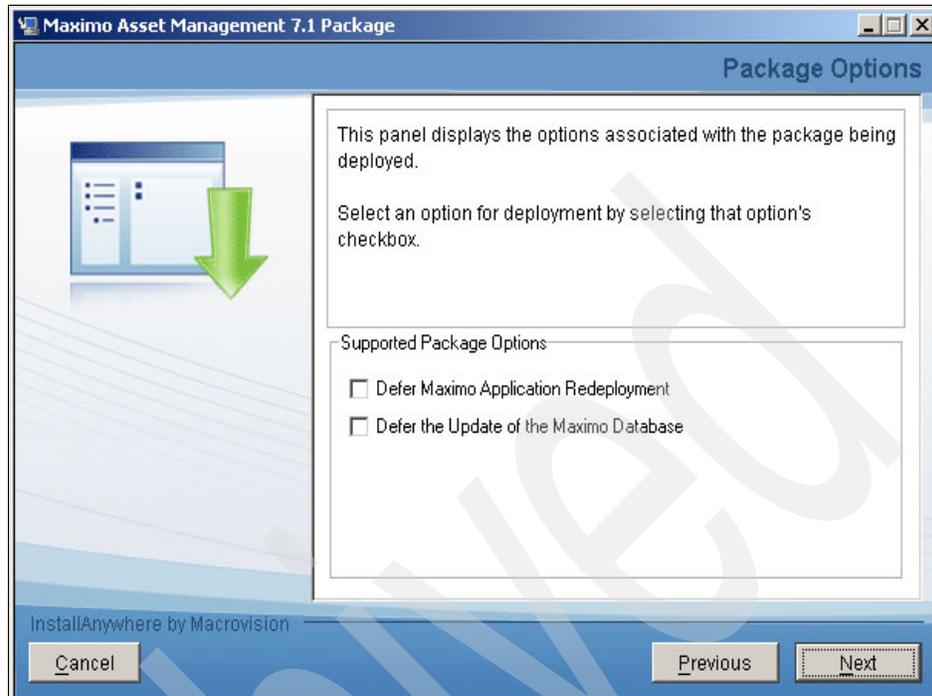
Figure 5-31   Package Options

5. From the Pre-Install Summary panel, review and verify the information displayed, and then click **Next**.

6. At this point, the Process Solution installation program begins the package installation process. The Deployment Progress panel (see Figure 5-32 on page 108) displays the deployment progress of the installation.
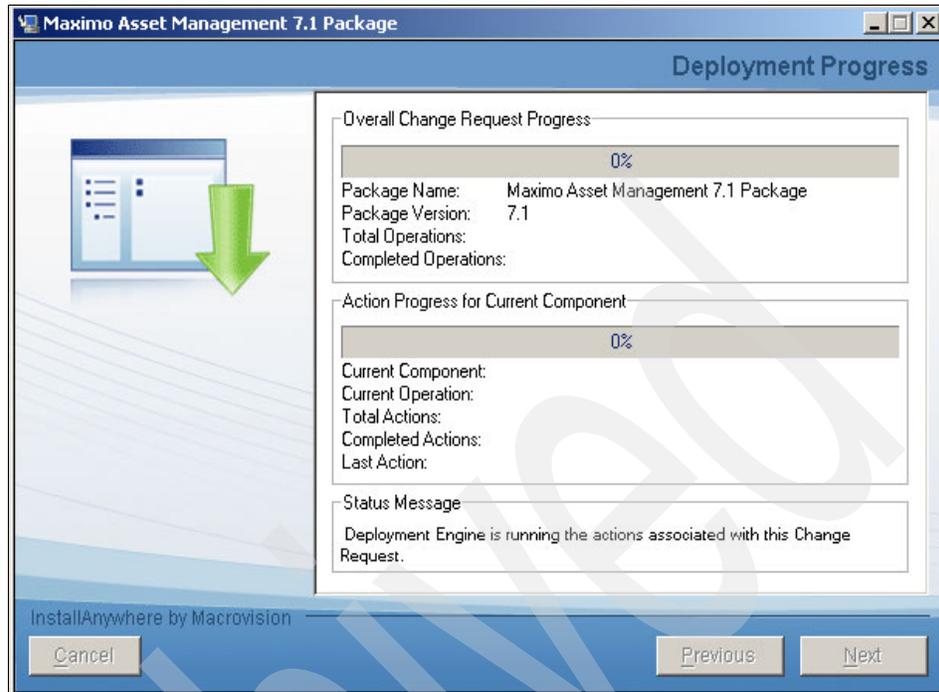
Figure 5-32   Deployment Progress

7. When the installation has completed successfully, from the Package Successfully Deployed panel (see Figure 5-33), click **Done** to exit the Process Solution installation wizard.

If a package failure occurs, a message is 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 terminating, and no installation activities are being performed. The deployment of the Process Solution package you were installing has already completed, and you can safely ignore the progress bar.
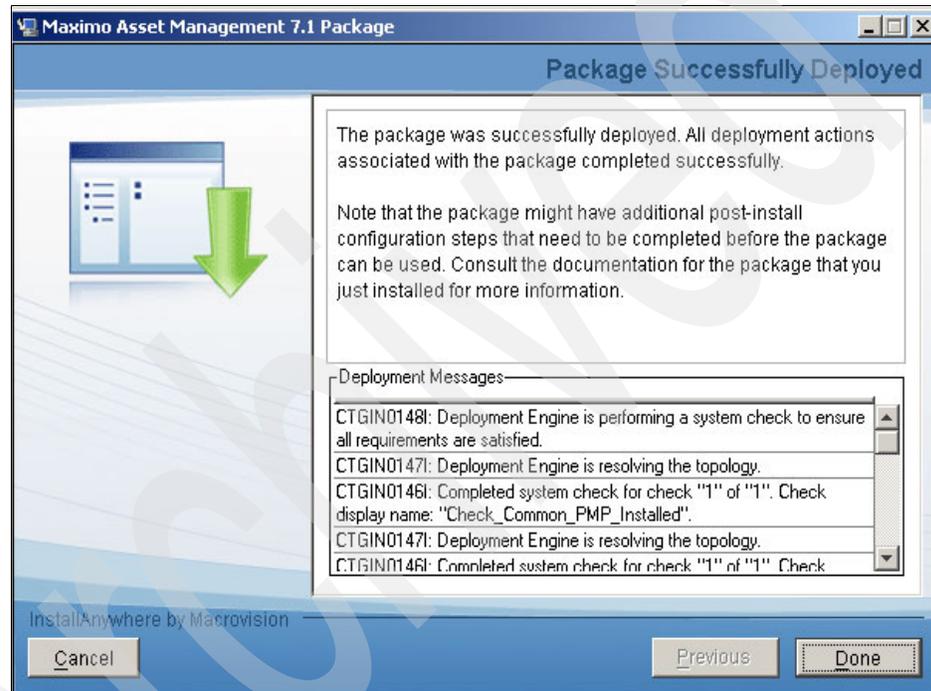
Figure 5-33   Package Successfully Deployed

5.5.3 Installing an additional base package solution

In the previous section we briefly explained how to enable the Maximo Asset Management license.To implement another base package solution, you run the solutionInstallerGUI.bat program on the following path:

```
C:\ibm\SMP\bin
```

Complete the following steps to install a Process Solution package using the Process Solution installation wizard.

1. In our example we install the Linear Assets Process Solution. Ensure you copy the Linear Assets Process Solution ZIP file to the following path before continuing the next step (see Figure 5-34):

   `C:\ibm\SMP\pmp\`
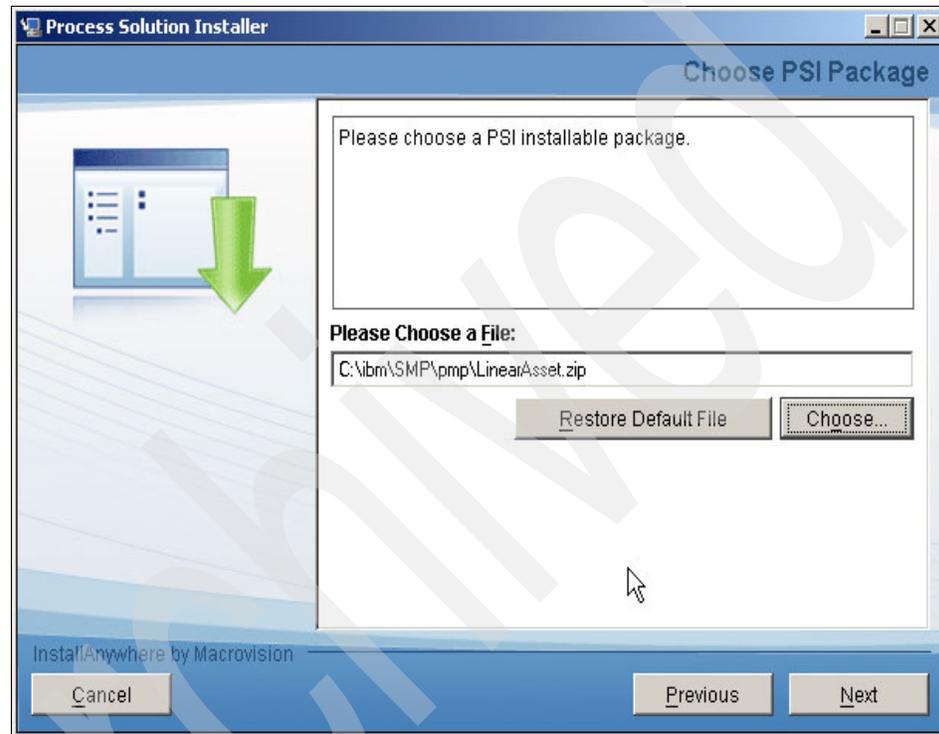
Figure 5-34   Process solution installation package deployment

2. Launch the Process Solution installation program by navigating to the <MAM_HOME>\bin directory of Maximo Asset Management and invoking the solutionInstallerGUI.bat program.

3. Select a language for your installation, then navigate to the package file you wish to deploy, and select it.

   After the package has been selected, a validation is performed to verify whether the package solution has already been implemented or has the correct middleware or version requirements in the case of an update.

4. Allow the installation to continue. When the installation is complete and successfully deployed, you are prompted to add another package should you

wish to do so (see Figure 5-35). If not, click **Done** to complete the Process Solution process.
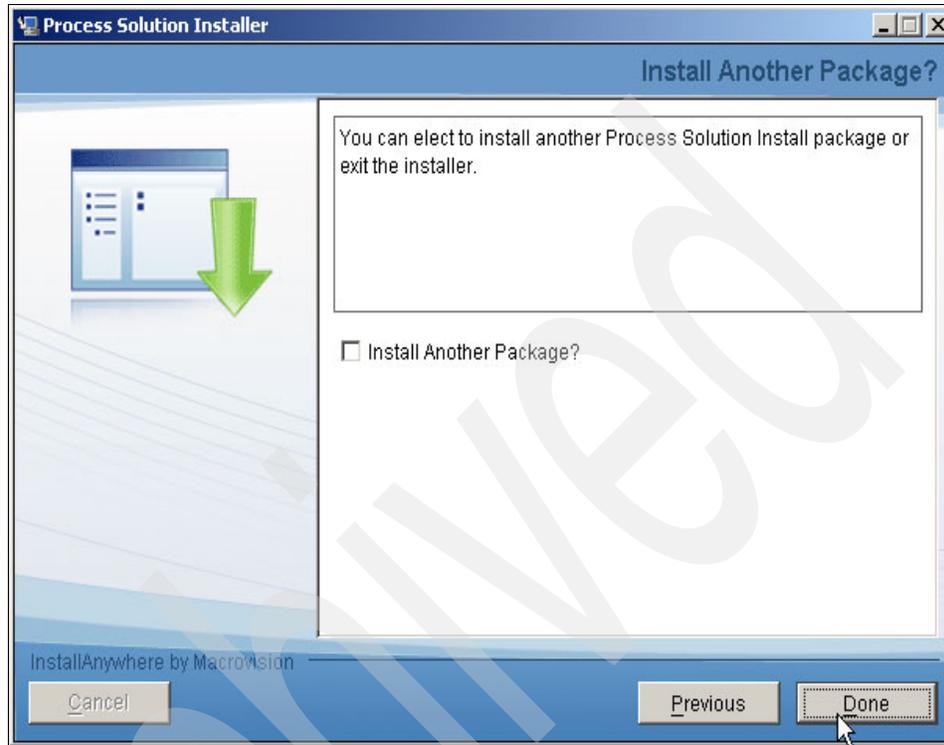
Figure 5-35   Install Another Package?

5.5.4 Process solution installation logs

If any problems or error messages occur during the Process Solution installation program, refer to the log files stored in the locations specified in Table 5-3 on page 112).

Table 5-3   Log file locations

Log type	Description	Location/Path
Package log	These are log files containing the StdOut/StdErr output of external commands launched by the package as it is processed by the Deployment Engine. These log files are typically vital to the proper debugging of package issues. In general, logs have two parts: an .out. and an .err. file, both with the same pre-extension file name. The .out files contain the contents of the Standard Output stream as output by the external command. The .err files contain the contents of the Standard Error stream. It is normal for one to be blank, provided there was no error output (or there was ONLY error output). Note that you might discover numerous (10-20) package log files generated for any particular package installed.	<MAM_HOME>\solutions\logs\<PACKAGE_NAME>\
Solution Installer/Deployment engine logs	These logs are kept by the IBM Solution Installer/Deployment engine runtime. Process solution installation utilizes the IBM technology as the means to install and keep track of installed packages. This runtime has its own logging system.	C:\Program Files\IBM\Common\acsi\logs\Administrator\
WebSphere Application Server logs	These logs include connections, exceptions, and other failures experienced by the WebSphere Application Server in its day-to-day running. These logs are often helpful in the diagnosis of errors - in particular, EAR files or other back-end operations, such as database connections.	C:\IBM\WebSphere\AppServer\profiles\AppSrv01\logs\
Maximo Asset Management logs	A few logs kept by Maximo Asset Management are useful in tracking the progress, success, or failure of a few Maximo back-end commands.	C:\IBM\Maximo\tools\maximo\log\updatedb20070419170607.log
WAS thin client logs	The WAS thin client is the mechanism by which the process manager packages communicate with the WebSphere Application Server. If this automated deployment should happen to fail, the exact actions taken by the thin client and the associated responses from the WebSphere Application Server are stored in logs.	C:\IBM\SMP\wasclient\logs\

5.6 Language pack installation

A single database can contain data in multiple languages, which enables diverse users to run Maximo Asset Management in their native languages. By default, multiple languages is enabled for:

► Data dictionary tables
► Company and item objects
► Maximo messages

After you have successfully deployed Maximo Asset Management, you can add language support to the Maximo Asset Management user interface using the Maximo Asset Management Language Pack installation program. Note that MXServer must be started prior to running the Maximo Asset Management Language Pack installation program.

> **Important:** If you plan to add language support to Maximo, you must use the Maximo Language Pack installation program before you perform post-installation steps.

Take note that the installed Language Packs are applicable and effective only on the Maximo Asset Management installation. All add-ons should be provided with their own separate Language Pack installation packs.

5.6.1 Language Pack installation considerations

Because Language Pack installations can take several hours to complete, we recommend you make a clear definition as to which languages are to be installed in addition to your base installation.

> **Note:** Your base language is installed directly when you install Maximo Asset Management. Your Domain values (valuelists) internal values always are displayed in the base language regardless of the additional Language Packs you install.

Be careful when choosing Language Packs you wish to implement. Language packs are considered add-ons and should be implemented only if the diversity of your users require it. Adding a language to Maximo impacts the sizing considerations of the implementation itself.

Language Pack definition

When you have decided on the languages you wish to install, list all the languages you wish to implement with the Maximo Asset Management Installation. For instance, you can use Table 5-4 as a checklist for the languages you want to install in your environment. This is not a complete list of languages available for Maximo Asset Management.

Table 5-4   Checklist for installing languages

Language	Abbreviation	Base	Already implemented?
English	EN	Yes	Yes
German	DE	No	No
Spanish	ES	No	No
French	FR	No	No
Japanese	JA	No	No

Ensuring all middleware servers and services are running

Before you begin your Language Pack installation, ensure all your relevant middleware services are running.

5.6.2 Installing the Language Pack

Log on as the administrator and run the launchpad. Select **Install the Product** → **IBM Maximo Asset Management Language Pack Installer**. You are prompted to specify the language you wish to use to install the Language pack. The Language Pack is not the language you wish to install but the language of the instructions and prompts you receive during the actual installation process.

Next, select a base language to be used with Maximo Asset Management. This is the only opportunity you have to select a base language. You cannot change the base language at a later time.

You are asked to select zero or more additional languages to be supported. A summary panel is displayed for you to review your inputs. After you have confirmed the inputs, wait for the Language Pack installation process to complete.

> **Note:** Language pack installations are have a fairly lengthy duration of one to two hours.

► For more details about the installation process, refer to either the *Installation Guide: IBM WebSphere Application Server* (mam71_install_was.pdf) or *Installation Guide: BEA WebLogic Server* (mam71_install_bea.pdf) available at:

`http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.mam.doc_7.1/mam_welcome.htm`

5.6.3 Tips regarding multi-language deployments

The following are a few tips related to multi-language installations

► Locale settings

Even if you added additional languages through the Language Pack installation program, and you set the locale or your machine to a language that was installed as an additional language, the Maximo user interface may still display items in the language you identified as the base language of the machine. This is a known limitation and does not indicate that the Maximo Language Pack installation failed.

► Shortcut elements

In some cases, the Maximo user interface displays shortcut elements - for example, menu choices - in the designated base language, or these elements are displayed in English only.

► Migrating interface representations (XML)

When using the application designer to migrate interface representations between environments, ensure you are using the same default language on both systems (the export and import). If you fail to do so, you might encounter language errors on certain tabs and panels.

► User-defined fields

As a best practice when using multi-language platforms and when configuring user-defined fields in Maximo Asset Management, we strongly recommend you assign the title manually for each language that has been implemented. If you fail to do so, you encounter interface representations that do not have visible titles.

► Reports

Choosing your Reporting tool and designing reports requires careful consideration as you define your report labels for each different language

being implemented. This adds a time constraint to your implementation and design of the reports. Although Maximo-supported tools enable you to create multi-language reports, when selecting external reporting applications, ensure that these external tools facilitate multi-language reporting features.

5.7 Post-installation tasks

Before configuring your Maximo Asset Management deployment, take the time to ensure that the basic installation is functional. To confirm this, read the following sections, and verify and adjust where necessary.

> **Tip:** We recommend you document all your findings meticulously. When you have completed all your tasks, store your installation instructions and properties for future reference.

5.7.1 Ensuring Maximo Asset Management services are running

Before you can attempt to access Maximo Asset Management, ensure that all the relevant services are running. From Windows Services, set the following services to automatically start up:

- ► 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

> **Tip:** If you are running a demo Maximo Asset Management installation on limited resources - for example, a mobile computer or a virtual machine - we recommend you do not run the specified services as automatic, but rather keep these services configured to start manually.

5.7.2 Ensuring connectivity to Maximo Asset Management application

When you have successfully started the requested services, ensure connectivity by accessing the logon panel and logging on to Maximo Asset Management.

The default Maximo system administrator user ID and passwords are as follows

► User ID: maxadmin

► Password**:** maxadmin

> **Tip:** We strongly recommend that you change your password at regular intervals. Also, if the maxadmin user is administering on a multilanguage implementation, take care to note in which locals this user is applying the changes. Aim wherever possible to remain and administer in the base language.

5.7.3 Ensuring the Language Pack installation is functional

To determine whether the Language Pack installation is successful, ensure you can log on into the various languages listed on your Maximo Asset Management logon page. Carefully assess whether all the Language Packs are installed as planned. If some of your languages are not installed carefully, re-assess and implement those languages.

5.7.4 Ensuring client connectivity

Because Maximo Asset Management has a Web-based infrastructure, it is imperative to verify accessibility from the clients who will be accessing Maximo. Ensure the users can access the logon page. A number of client hardware and software requirements (refer to 4.1, "Hardware and software requirements" on page 46) must be met to confirm this task.

5.7.5 Checking report administration

Before configuring your deployment, perform the report administration checks described in the following sections.

Generate request pages

Request pages are the parameter inputs required for each report. Before the reports can be accessed from with Maximo Asset Management, the request pages must be generated; follow these steps:

1. Log on to Maximo as the system administrator.

2. From the Start Center, navigate to the Go To function and select **Administration** → **Reporting Administration** (see Figure 5-36).
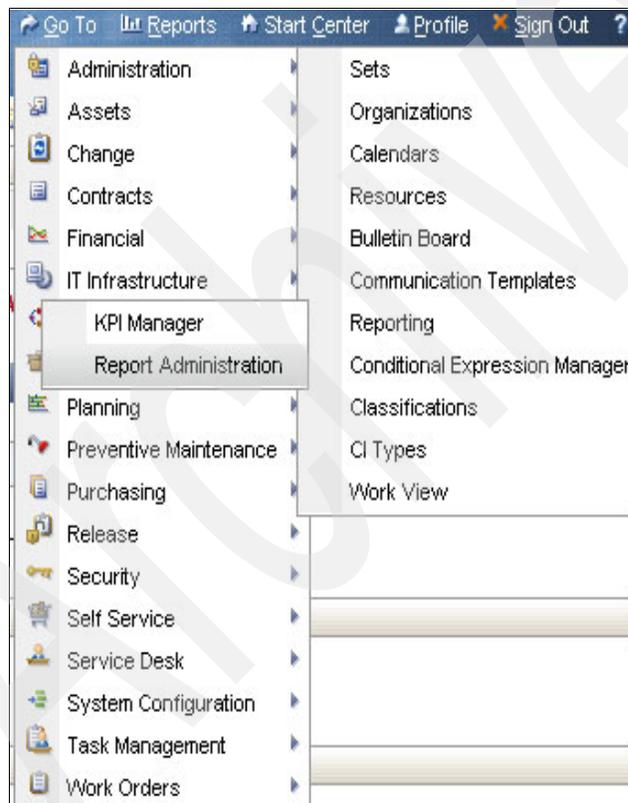
Figure 5-36   Report Administration

3. In the center of the page is the Generate Request Pages button, click **Generate Request Pages**.

At this stage, the request pages have been successfully generated, as shown in Figure 5-37.

Figure 5-37   Generate request pages

> **Note:** Generating the request pages is a step you execute only when you have created a new installation or uploaded a new report. We recommend that you complete this step when other users have logged out of the system.

Generate report output

As a best practice, we suggest you verify whether the reports generated successfully. During the Maximo Asset Management installation process, the default report tool BIRT is automatically configured to run reports from within Maximo Asset Management. In the next few steps we test and run a standard report.

1. Log on to Maximo Asset Management.

2. From the Start Center navigate to the Go To function and select **Administration** → **Resources** → **Labor** (see Figure 5-38).
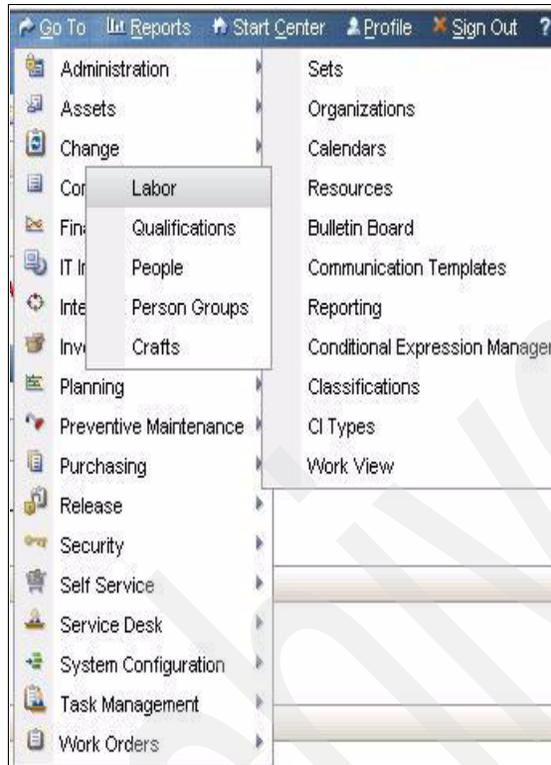
Figure 5-38   Labor

3. From the Labor application, click **Select Actions** → **Run Reports** (see Figure 5-39).
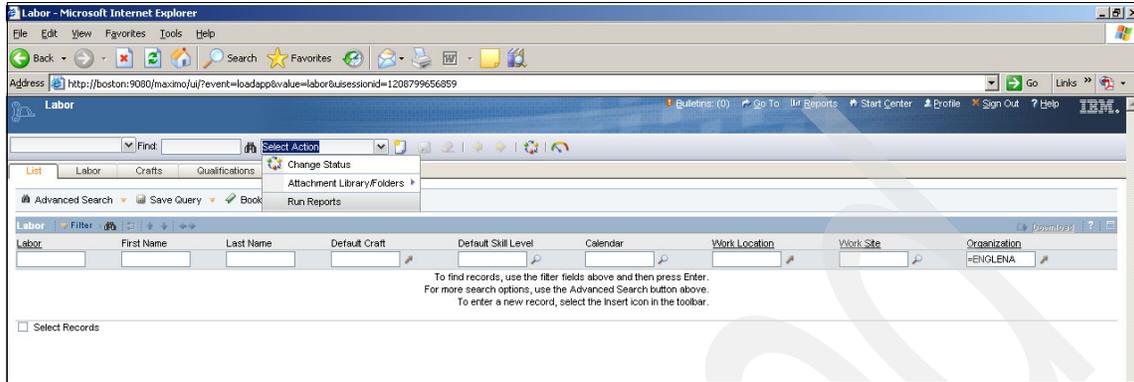
Figure 5-39   Run Reports

4. From the Reports dialog, select the **Labor List** report (see Figure 5-40).
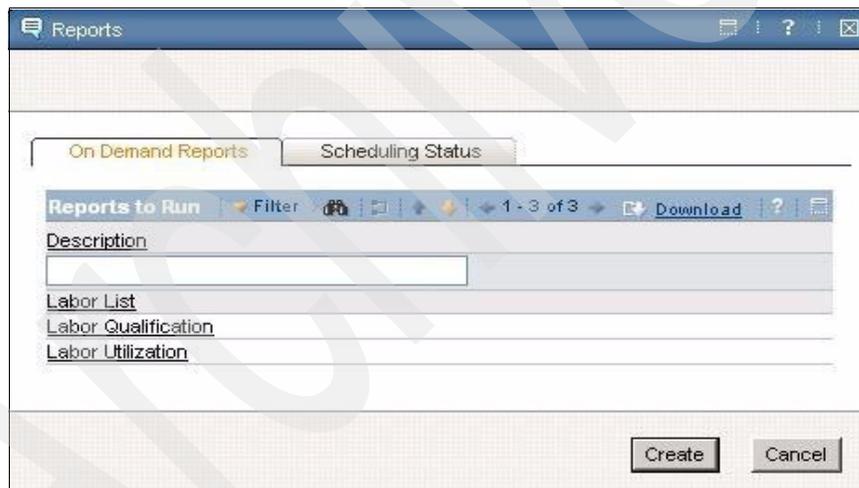
Figure 5-40   Labor List

5. From the Request Page, specify your parameter properties. In our example we specify ACTIVE as the Status parameter value and click **Submit** (see Figure 5-41).

Figure 5-41   Request page

When you receive the report output, as shown in Figure 5-42, you have
verified that the report was generated successfully.
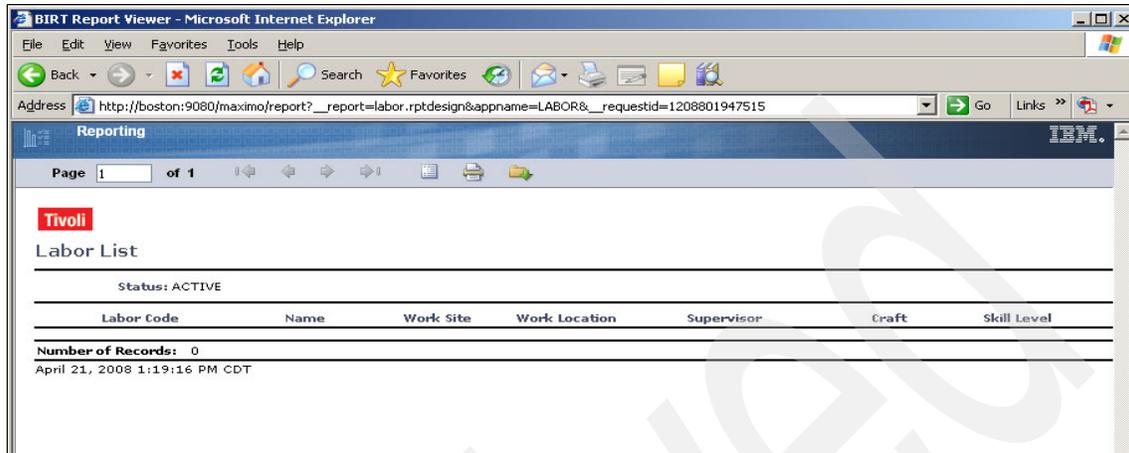
Figure 5-42   Report output

For more information regarding report administration, refer to the guide
Report Developer Guide, which you can find here:

```
http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?top
ic=/com.ibm.mam.doc_7.1/mam_welcome.htm
```

5.7.6 Configuring Attached Documents

You use the Attached Documents application in Maximo Asset Management to
attach various documents to individual Maximo records.The following formats are
supported:

► PDF
► XLS
► CSV
► TXT
► DOC
► GIF
► DOC
► JPG
► PPT

By default Attached Documents are not configured during the Maximo Asset
Management installation process. You have to manually configure properties
within Maximo System Properties application and specific HTTP server

properties to enable attachments to be uploaded and viewed from Maximo Asset Management.

Follow these steps to ensure Attached Documents are accessible in Maximo when using a Microsoft Windows WebSphere Application Server middleware installation.

1. Create a shared doclinks directory on the machine storing the document files with your predefined subdirectories if you wish to group your attachment types:

   ```
   C:\DOCLINKS
   ```

2. On your HTTP server, search for your httpd.conf file - for example, on Microsoft Windows, search the following path:

   ```
   C:\IBM HTTP Server\conf\httpd.conf
   ```

   Tip: Make a backup of this file before you begin.

3. Edit the directory line as follows:

   a. Keeping in mind that the values you enter are case sensitive, edit this directory line to specify the doclinks directory you created:

      ```
      <Directory "C:\DOCLINKS">
      ```

   Tip: Search on the words "this should".

   and the DocumentRoot line as follows:

   ```
   DocumentRoot "C:\DOCLINKS"
   ```

   Tip: Search on the word "DocumentRoot".

   After you have made these changes, save your file and restart the HTTP server.

   At this point, you log on to Maximo and configure two more DOCLINK properties to point to the folder you have created on your drive and to point to the path the HTTP server must use to access these files.

4. Select **Go To** → **System Configuration** → **Platform Configuration** → **System Properties** and specify the required properties as listed in Table 5-5.

Table 5-5   Required system properties

Property name	Description	Example value
mxe.doclink.doctypes.defpath	Default path for doclinks	C:\DOCLINKS
C<PATH>\DOCLINKS = http://yourservername	Path 01 for doclinks	C<PATH>\DOCLINKS = http://yourservernameC<PATH>\DOCLINKS = http://yourservername

A common problem with Attached Documents setup is that the folders for your attachments are not assigned with a drive letter in their default paths during Maximo Asset Management installation. To resolve this problem, open any of your Maximo applications and from Select Actions, choose **Attachment/Library Folders** → **Manage Folders** and amend the default path as shown in Table 5-6.

Table 5-6   Amend default path for Attached Documents

Old path	New path
\DOCLINKS\ATTACHMENTS	C:\DOCLINKS\ATTACHMENTS

Figure 5-43 and Figure 5-44 on page 127 illustrate single-server and multiserver Attached Documents configurations.
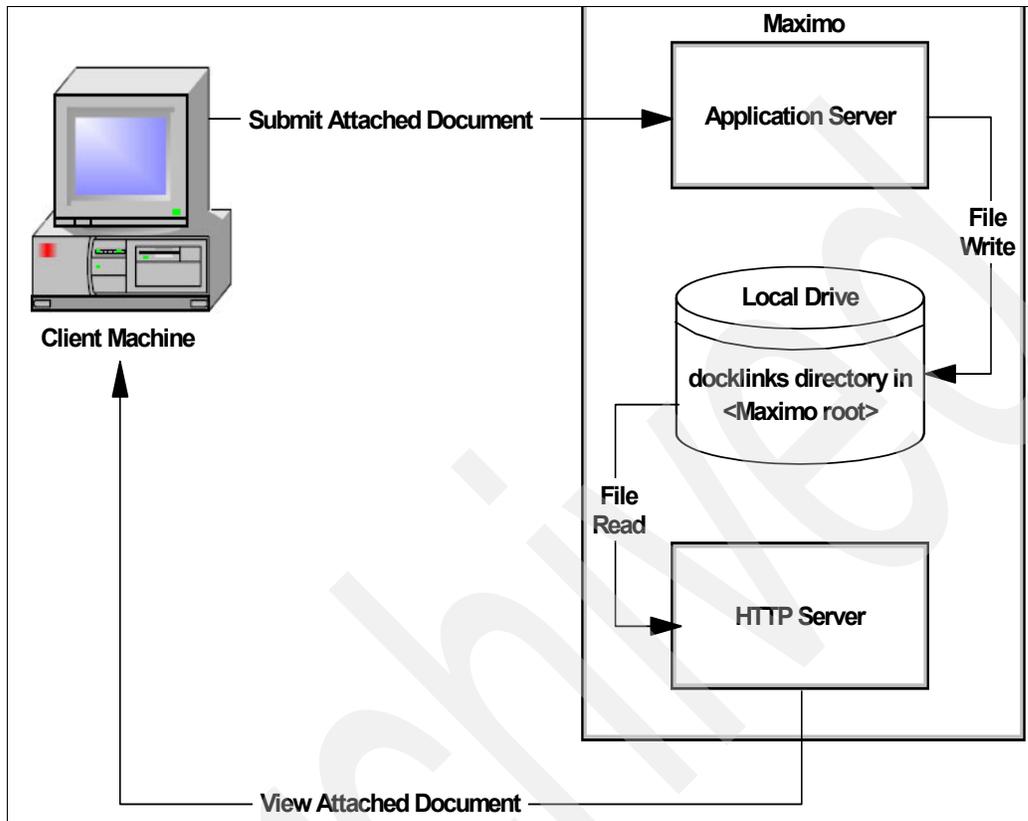
Figure 5-43   Single machine configuration
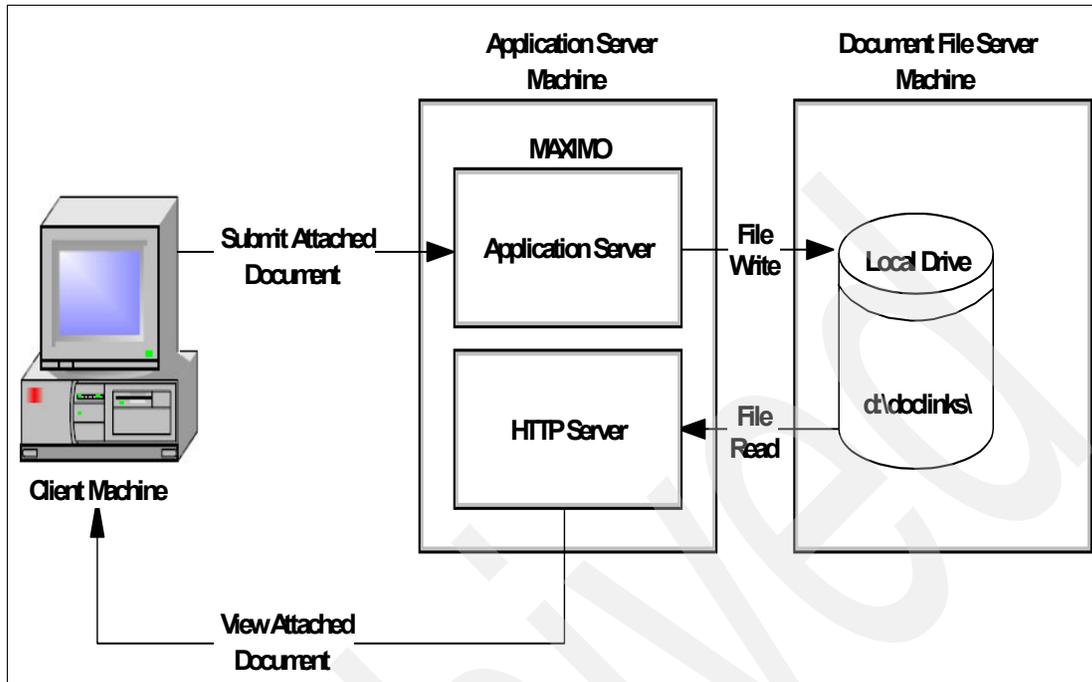
Figure 5-44   Multiple machine configuration

Note: The implementation of Attached Documents is briefly discussed in this chapter. For complete instructions on how to install and deploy Attached Documents, refer to Chapter 11, "Attached Document Administration and Configuration" in the *System Administrator Guide*, available at:

```
http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.
mam.doc_7.1/mam_welcome.htm
```

5.7.7 Ensuring relevant cron tasks are working and enabled

Depending on your business requirements, you may be required to implement cron tasks to perform certain actions or updates on your Maximo Asset Management application. Ensure system properties are defined and set up as enabled in your Cron Task Setup module, which can be found by selecting **System Configuration → Platform Configuration**.

If you are running JMS queue configurations for your Integration Framework, it is essential for the relevant cron task instances to be enabled; otherwise, the

queues cannot function. In the event that cron tasks do not run, we recommend you reload the cron tasks.

> **Tip:** The scheduling of cron tasks must be assessed carefully because cron tasks are "batch" tasks, and when a large task is executed frequently, it can have an adverse affect on system performance.

For more detail about implementing cron tasks, refer to the *System Administrator Guide* (mam71_sys_admin_guide.pdf), which you can access here:

```
http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=
/com.ibm.mam.doc_7.1/mam_welcome.htm
```

5.7.8 Choosing the backup environment

After the installation is complete, it is advisable to create an image of your environment. Backup procedures depend on the size of your database and the type of operation you are running. For instance, you can back up to:

► Hard disk drive: Restores your system quickly.

► Tape drive: Slower, but you can keep multiple tapes of backups.

► CDs, DVDs, diskettes: Limited capacity, but this alternative is useful for smaller databases, archive files, or specific executables.

5.7.9 Automating the backup procedure

Schedule and regularly perform system and database backups. We recommend you perform system restores from the production environment to your development or acceptance environment at regular intervals to test this procedure.

5.7.10 Assessing the server workload

Taking the sizing considerations described in 4.4, "Sizing considerations" on page 52 into account, we recommend you monitor the performance of Maximo Asset Management on a continuous basis. This step also forms part of user acceptance testing. However, you can implement many methods to improve and maintain application performance on a continuous basis.

If your implementation increases in size - that is, the workload increases or system itself grows over time - consider load-balancing the server environments. We recommend server clustering as a solution that enables you to distribute user

loads over various application servers to maintain consistent server performance.

For more information regarding the server workload, read the sizing considerations as discussed in Chapter 4, "Solution environment" on page 45.

5.7.11 Setting client local or regional properties

If your users are based in one location, ensure that the regional settings of their workstations correspond to their regions. A common issue concerns date and time format conflicts.

5.7.12 Enabling Internet and firewall security

For security purposes, Maximo Asset Management can also be configured to take advantage of the more secure protocol, Hypertext Transfer Protocol Secure (HTTPS). If Maximo clients exist outside the corporate network, you can add a firewall or other security measure.

Firewalls are configured to enable communication over HTTP (typically port 80) or HTTPS (typically port 443). The information in the following sections is generic and does not reflect any particular firewall brand.

Secure Sockets Layer (SSL) overview

Secure Sockets Layer provides secure connections over a network connection by doing the following:

– Enabling two applications to authenticate each other's identity
– Encrypting data exchanged between the two applications

Authentication enables a server and optionally a client to verify the identity of the application on the other end of a network connection. Encryption makes data transmitted over the network intelligible only to the intended recipient. IBM WebSphere Application Server and BEA WebLogic Server support SSL, and IBM Corporation has certified the SSL implementation with Maximo-WebSphere Application Server and Maximo-BEA WebLogic Server integrations.

Configure Secure Sockets Layer

A Web server must have an associated certificate for each external interface (IP address) that accepts secure connections. After you install the certificate on the Web server, replacing .http with .https encrypts a session between the browser and server.

For example, `http://MaximoAppServer` is entered as `https://MaximoApp Server`.

If a proxy server or firewall controls network traffic, this port and protocol must be opened. SSL carries additional overhead for encryption and decryption of data - that is, encryption and decryption can affect performance.

Set Internet Explorer stored pages

We recommend you enable your browser to automatically check for newer versions of stored pages when you log on to Maximo Asset Management. To do this, perform the following steps:

1. From your Internet browser, navigate to **Tools** → **Internet Options** → **General** → **Settings**.

2. Select the radio button **Automatically** to check for newer versions of stored pages.

> **Tip:** Ensure the automatic stored pages feature is applied to all clients. If you fail to do so, users view previous page representations, rather than the latest. As an alternative means of ensuring your users are viewing the latest pages, you can remove and destroy all cookies and temporary files in their Internet browsers.

5.7.13 Setting up SMTP mail exchange server

Some applications and cron tasks within Maximo Asset Management require you perform a mail exchange. To enable mail exchange, assign a valid SMTP server name to the global value of the following property in the System Properties application:

```
mail.smtp.host
```

5.7.14 Authenticating LDAP through Virtual Member Management

You can enable LDAP user authentication using Windows Server Active Directory. If your organization has Virtual Member Management (VMM) in place, consider using it to perform your authentication. When you configure the application server to enable LDAP user authentication using an Active Directory, you create and manage users in the LDAP directory server.

The VMM cron task updates the Maximo Asset Management database when users, groups, and group membership are changed in the directory server. When

users and groups are deleted from the Active Directory, they are not deleted from the Maximo database because these records might be needed for auditing purposes. You can also configure the system to populate person, user, and group information from the external directory. The system currently supports the synchronization of information from Microsoft Active Directory. Synchronization with other directories is possible but is not supported as a standard feature and might require programming to configure. Both BEA WebLogic Server and IBM WebSphere Application Server support authentication using Windows Server Active Directory.

During the installation if you have not chosen to automatically configure the middleware, refer to the following appendix sections:

► "Manually configuring VMM on WebSphere Application Server" on page 257
► "Manually configuring VMMSYNC cron task for Active Directory" on page 291

5.8 Setting system and logging properties

When the installation is complete, you can configure the different system properties and enable logging for the application. With Maximo Asset Management V7.1, these tasks have become far easier, because the product includes dedicated applications to perform these activities.

5.8.1 System Properties application

System Properties is a new application that is a part of Maximo Asset Management V7.1 and can be used to configure systemwide properties at a global level or instance level. These properties consist of keys and values that determine the configuration of the product and the behavior of many of its components.

Global and instance properties are defined as follows:

► **Global properties**

A global property exists only at a systemwide level. This means that the property is applicable to all the product server instances (for example, the application server), working with a common database. For example, the system property mxe.logging.rootfolder is a global property whose value is a directory on the hard disk of the server machine where the product's log files reside.

▶ **Instance properties**

An instance property is defined and associated with a specific product server instance. For example, you can configure the system property mxe.crontask.donotrun to be an instance-specific property. You do this by associating the mxe.crontask.donotrun system property with a specific server (for example, MXServer1) and a value applicable only to that server (for example, a value of BBCron). As a result of this configuration, the Bulletin Board cron task (BBCron) is not executed on MXServer1; yet it can execute in another product instance, such as MXServer2.

Each property defined in the System Properties application has a number of characteristics that you can manage. Table 5-7 describes these characteristics.

Table 5-7   System Properties application characteristics

Property characteristic	Description
File Override	Specifies whether the property and its value are loaded from a file rather than from the database.
Global Only	Specifies whether this property must exist only at a systemwide level. It implies the property cannot be overridden at the instance level.
Instance Only	Specifies whether this property must be defined at the instance level. If so, you provide an instance-specific value (the property is not a global value).
Online Changes Allowed	Specifies whether the System Properties application is used to change the property's value. For example, the global property mxe.db.driver does not allow online changes.
Live Refresh	Specifies whether the property value can take effect immediately after saving the value.
Encrypted	Specifies whether the property is stored in an encrypted manner in the underlying product database. The value is encrypted using the product's standard encryption functionality. For example, the global property mxe.int.uddipassword is encrypted.
Security Level	Specifies the level of access to this property by various product components. Access level may be PUBLIC, SECURE, or PRIVATE.
User Defined	Specifies whether the property is created by a user or is provided with the product.
Nulls Allowed	Specifies whether the property can have null values. You can change this characteristic only for user-defined properties.

Property characteristic	Description
Data Type	Specifies the type of value that can be provided for the property. The value can be an integer, alphanumeric, or a Yes or No value (YORN). For example, the global property mxe.allowLocalObjects is associated with a data type of YORN; if you enter a value other than 1 or 0, an error message is displayed indicating invalid value.
Domain	Specifies a domain that provides a list of values that the property can be set to. For example, the global property mxe.db.transaction_isolation is associated with the TRANSISO domain. Thus the property's values must match a corresponding domain value.

5.8.2 Logging application

The Logging application is a part of the product's System Configuration module and is a dedicated application for managing log settings and configuring log files. You can configure various logging components, set log levels, associate logging with log files, and specify a folder where log files are to be written.

Logging has three main components:

▶ Loggers

Loggers are components that prepare log statements to be written to console or log file. Loggers are named entities or keys, for example: log4j.logger.maximo.sql.

Loggers form a hierarchy. A logger is defined as an ancestor of another logger if its name, followed by a dot, is a prefix of the descendant logger name. A logger becomes the parent of a child logger if no ancestors exist between itself and the descendant logger. For example, log4j.logger.maximo.sql is the parent of log4j.logger.maximo.sql.WORKORDER. You can assign the following levels to loggers:

– DEBUG
– INFO
– WARN
– ERROR
– FATAL

A level indicates a type of event that the system logs.

▶ Appenders

You can send logging requests to multiple destinations. An output destination is called an *appender*. Appenders can exist for consoles or files. You can associate one or more loggers with a given appender. Alternatively, you can associate a single logger with multiple appenders.

▶ Layouts

A layout determines the output format of a log statement. A layout is always associated with an appender. For example, a conversion pattern such as %d{dd MMM yyyy HH:mm:ss:SSS} [%-2p] %m%n results in the following log statement: 2007-05-07 14:07:41,508 [main] INFO MyApp - Entering application.

The Logging application supports two types of loggers: root and child. One or more child loggers always inherit from one root logger. In the Logging application, a root logger is termed *root logger*, while a child logger is termed *logger*.

The Logging application can be used to:

▶ Create loggers that are product components that prepare log statements to be written to the console or a log file.

▶ Associate the appropriate log level with each logger.

▶ De-activate loggers except root loggers.

▶ Configure appenders that represent a console or log files to which log statements are written.

    – Specify an appropriate file name for an appender.
    – Specify a log file size for an appender.
    – Associate a logger with multiple appenders.
    – Associate an appender with multiple loggers.

6

Configuration

When installation is finished, you must complete several tasks before you can enable users to start using the system. As system administrator, you are the individual who usually carries out such tasks, which are discussed in this chapter.

Note: The steps in this chapter assume you have created a empty database and have not created the demo database. This chapter does not apply to a demonstration installation.

The following topics are covered in this section:

► User management
► Organization and site setup
► Initial configuration and data import sequence

6.1 User management

Signing in using a default user ID

If you enabled security during the installation, user management takes place through the directory server you have configured to use with Maximo Asset Management. When first installed, Maximo Asset Management contains the following default user IDs:

► maxadmin
► maxreg
► mxintadm

The default password for each user ID is the same as the user name (for example, maxadmin is both the user name and default password).

> **Note:** User names and passwords are case sensitive. The default user names and passwords are lowercase.

To sign in, complete the following steps:

1. Open a browser window.

2. Navigate to the Maximo Asset Management logon URL - for example, http://hostname:<port>/maximo.

   The default port is 9080.

3. Enter the user name `maxadmin` (lowercase).

4. Enter the password `maxadmin` (lowercase) and click **Enter**. The software displays an empty Start Center.

Changing the user password

As a best practice, change the passwords for the default user IDs. To change the default passwords, complete the following steps:

1. Open the Users application.

2. From the List tab, select the user whose password you want to change.

3. From the Select Action menu, select **Change Passwords**. The Change Passwords dialog box is displayed.

4. Enter the new password in the **New Password** field.

5. Re-enter the password in the **Confirm New Password** field.

6. Click **OK**.

When you change the password of either the maxadmin user or the maxreg user, you also must change the password associated with that user in the maximo.properties file. You can do so by following these steps:

1. Navigate to MAXIMO\applications\maximo\properties.

2. Open the maximo.properties file using a text editor.

3. Search for the appropriate property and modify it as needed:

   – mxe.db.user for the database log on name
   – mxe.system.reguser for self-registering new users

4. Save your changes.

> **Note:** Any time you modify the maximo.properties file, you must rebuild and deploy a new maximo.ear file. You do so by accessing the application server, selecting the ear file, and clicking the deploy option. (The method may vary if you use a different application server; the steps in this Note apply to using WebSphere Application Server.)

You can change the default user names for the default user IDs by editing the maximo.properties file. Complete these steps:

1. Navigate to MAXIMO\applications\maximo\properties.

2. Open the maximo.properties file using a text editor.

3. Search for the appropriate property and modify it as needed:

   – mxe.db.password for the database logon password
   – mxe.system.regpassword for self-registering new users

4. Save your changes.

> **Note:** Any time you modify the maximo.properties file, you must rebuild and deploy a new maximo.ear file. You do so by accessing the application server, selecting the EAR file, and clicking the deploy option. (The method may vary if you use a different application server; the steps in this Note apply to using WebSphere Application Server.)

6.1.1 Organization and site setup

Before you can start working in Maximo Asset Management, it is imperative that you specify the name of your organization and the site for the organization. To ensure you do so, complete the steps in the sections that follow in strict

sequential order to create the organization, sites, and their appropriate dependencies.

> **Important:** You must create currency codes, and sets, configure the general ledger component, and create a clearing account prior to creating the organization and site. You must complete each step in the order presented in the following sections.

Create currency codes

A currency code is the measure of exchange in monetary terms. Specifying a currency code is required for your system to determine the currency you are measuring your financial transactions against. Determine the base currency that your organization utilizes before beginning the steps in this section.

You must define a currency code for an organization; complete these steps:

1. Open the Currency Code application by selecting **Go To** → **Financial** → **Currency Code**.

2. Click **New Row**.

3. Enter a currency code - for example, USD (United States dollar).

4. Click **Save**.

Disable validation options

We recommend you disable or uncheck the validation options to begin populating or importing data. If you have enabled validation options during the initial configuration, each entry you make is validated against a financial period, which is not required at this stage of the deployment. You can always revalidate these options at a later stage.

Follow these steps to disable the validation options:

1. Open the Financial application by selecting **Go To** → **Financial** → **Chart of Accounts**.

2. Access Select Actions.

3. Choose **Validation Options**.

4. Uncheck both **Validate GL Component Combinations** and **Validate Financial Periods**.

5. Click **OK**.

Create item and company sets

Defining item and company sets, you can share item and company records across organizations. To define these sets, complete the following steps:

1. Open the Sets application by selecting **Go To** → **Administration** → **Sets**.

2. Click **New Row**.

3. Enter a company set name - for example, COMPSET.

4. Enter COMPSET in the Type field.

5. Click **New Row**.

6. Enter an item set name, for example, ITEM1.

7. Enter ITEM1 in the Type field.

8. Click **Save**.

> **Tip:** Use simple, generic set IDs because you cannot alter them in the future.

Create a general ledger account component

A clearing account is an account that determines the cost center for multi-organization transactions. Before you assign your clearing account for your organization, you must create and define the general ledger account component. A general ledger account component is a structure that must be determined by your accounting staff.

Each general ledger account code consists of a number of distinct components (also called *segments*). In the Database Configuration application, you define the account code format using the GL Account Configuration dialog box. In the Chart of Accounts, you specify which components are valid for use in the system.

For easy identification, you can use delimiters to separate components when they are displayed. For example, you might use hyphens to separate components: 6100-400-SAF. The system writes account strings to the database in a concatenated format, with delimiters.

For any account code, you can define up to 20 components. Include a total of up to 254 characters or digits, not including delimiters (unless you choose to include the delimiters as part of the account code). Component Sequence Account components are displayed in a sequential format, with the leftmost component in the string representing the highest level. For example, four component levels are defined; the fourth is optional:

► Component 1 = Cost Center
► Component 2 = Activity

- Component 3 = Resource
- Component 4 = Element (optional)

To create a general ledger account component, complete the following steps:

1. Open the Database Configuration application by selecting **Go To → System Configuration → Platform Configuration → Database Configuration**.
2. Select **GL Account Configuration** from the Select Action drop-down menu.
3. Click **New Row**.
4. Enter a component name in the **Component** field - for example, Cost Center
5. Enter a numerical length for the component - for example, 5.
6. Enter a data type for the component - for example, ALN. (This is the type of data each component represents.)
7. Click **Save** and log out of Maximo Asset Management.

When you have completed setting up the GL Account Component, you *must* configure the database for your changes to take effect. Follow these steps to create a component:

1. Ensure you have created a backup of your database.
2. Make sure all users have logged off the system.
3. Document all changes.
4. Open the WebSphere administrative console and stop the MXServer application server.
5. Run the configdb.bat program from the following path:

   `C:\ibm\SMP\maximo\tools\maximo\configdb.bat`
6. Wait for your changes to take effect. Make sure this step has completed successfully, and then restart the MXServer. Log back on to Maximo, and you can now create general ledger accounts.

> **Note:** In our example, we configured the database after shutting down the Maximo application server. To perform a database configuration without shutting down your application server, refer to the *System Administrator Guide* (mam71_sys_admin_guide.pdf), available at:
>
> `http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.mam.doc_7.1/mam_welcome.htm`

Create an organization

To define the organization, complete the following steps:

1. Open the Organizations application by selecting **Go To** → **Administration** → **Organizations**.

2. Click the **New Organization** icon in the toolbar.

3. Enter an organization name in the Organization field - for example, MAIN.

4. Enter the base currency you defined in the Base Currency 1 field - for example, USD (see "Create currency codes" on page 138).

5. Enter the item set you defined in the Item Set field - for example, ITEM1.

6. Enter the company set you defined in the Company Set field - for example, COMPSET.

> Tip: Use simple, generic set IDs because you cannot alter them in the future. We recommend you not use spaces in between the organization and site IDs.

7. Enter the default item status of PENDING in the Default Item Status field.

8. Save your work.

Create a general ledger account

Before assigning the clearing account to your organization you have to create general ledger account to use as the clearing account. To create a general ledger account, complete the following steps:

1. Open the Chart of Accounts application by selecting **Go To** → **Financials** → **Chart of Accounts**.

2. Click on the name of your organization to select it. For example, click **MAIN**.

3. Select **GL Component Maintenance** from the Select Action drop-down menu.

4. Click **New Row**.

5. Add a GL Component value - for example, 1234 **-** and then click **OK**.

6. Click **New Row** on the GL Accounts for tab.

7. Select your general ledger account.

8. Click **Save**.

9. Open the Organizations application by selecting **Go To** → **Administration** → **Organizations**.

10. Click the organization name you created - for example, MAIN.

11. From the **Clearing Account** field, select the general ledger account you just created.

12. Select **Active**.

13. Click **Save**.

14. Assign your recently created general ledger account as the clearing account.

Create a site

To create your site or sites, complete the following steps:

1. From the Start Center, go to **Administration** → **Organizations**.

2. From the list window, select the organization you want to assign your site to.

3. Click the **Sites tab** to add a new site, and click **New Row**.

4. Specify the relevant site details. When complete, save your work.

Create default insert site

After you have created the organization and site in Maximo Asset Management, you must assign a default insert site to enter new records. To create a default insert site, complete the following steps:

1. Open the Users application by selecting **Go To** → **Security** → **Users**.

2. Search for "maxadmin" and select it to open the record for maxadmin.

3. Enter a the site you created earlier in the **Default Insert Site** field - for example, B901.

4. Enter a the site you created earlier in the **Storeroom Site for Self-Service Requisitions** field - for example, B901.

5. Click **Save**.

6. Open the WebSphere administrative console and restart the MXServer application server.

> **Note:** If you encounter an error message that indicates that the record is being updated by another user, log out as maxadmin and then log back on.

Signing out and signing in

When you make changes to a security group that your user ID is a member of, you must sign out and sign in again to view the changes. For example, although you granted the MAXADMIN group permission to create Start Center templates,

the actions are not visible until you sign in again. To view your changes, follow these steps:

1. Sign out as maxadmin.

2. Sign in as maxadmin.

6.1.2 Initial configuration and data import sequence

At this stage you have prepared the organization and site; from this point on, you can start populating the database. Although we did not write this guide to describe how to populate data into Maximo Asset Management, in Table 6-1 we briefly outline a recommended generic configuration and data import sequence for performing your initial data configuration.

Table 6-1 Generic configuration and data importation sequence

V7.1 module	Subapplication	Nature of work performed in this subapplication	Suggested entry mode
System Configuration	Database Configuration	Create new object attributes and relationships	Manual
System Configuration	Component Configuration	Configure GL component	Manual
Financial	GL Components	Create GL components	Manual
Financial	Chart of Accounts	Create GL codes	Manual
Financial	Currency Codes	Create currency codes	Manual
Administration	Item and Company Sets	Create item and company sets	Manual
Administration	Organizations	Create organization	Manual
Administration	Sites	Create sites	Manual
Administration	Calendars	Create calendars	Manual
Failurecodes	Failurecodes	Create failurecodes	Manual
Organizations	Tax Options	Assign tax types	Manual
Inventory	Measurement units	Define measurement units	Manual
Inventory	Conversion factors for measure units	Define conversion factors	Manual
Domains	Valuelists	Define ALN, NUM, SYNOMYM value lists	Manual

V7.1 module	Subapplication	Nature of work performed in this subapplication	Suggested entry mode
Organizations	Address Codes	Define addresses	Manual
Assets	Locations	Create systems and locations	Import
Purchasing	Company master	Create companies master records	Import
Purchasing	Companies	Create company records	Import
Administration	Classifications	Define classifications for assets, locations, items, etc.	Import
Inventory	Item Master setup	Create Item Master records	Import
Inventory	Storerooms	Create Storeroom locations	Import
Inventory	Inventory	Assign items to inventory	Import
Inventory	Tools	Create tools	Import
Inventory	Stocked Tools	Assign tools to inventory	Import
Assets	Meters/Meter Groups	Create meters and meter groups	Manual
Assets	Assets	Create assets	Import
Assets	Condition Monitoring	Define condition monitoring points	Import
Administration	Person groups	Define person groups	Manual
Administration	People	Define persons	Manual
SCConfig	Start Centers	Create new Start Center Templates (non-module)	Manual
Security	Security Groups	Create roles	Manual
Security	Users	Create and assign users to security groups	Manual
Administration	Crafts	Create crafts	Manual
Administration	Qualifications	Create qualifications for labor and crafts	Manual
Administration	Labor	Create labor records	Manual
Planning	Safety Plans	Create safety plans	Import
Planning	Precautions	Create safety plan precautions	Manual

V7.1 module	Subapplication	Nature of work performed in this subapplication	Suggested entry mode
Planning	Lock Out/Tag Out	Create safety plan lock out and tag out procedures	Manual
Planning	Safety Hazards	Create safety plan hazards	Manual
Planning	Job Plans	Create job plans	Import
Preventive Maintenance	Preventive Maintenance	Create preventive maintenance schedules	Import

Important: Table 6-1 is a generic example designed to guide the sequence of loading and populating the Maximo objects. You might not utilize all these modules. It is essential to follow the chronological sequence during your initial configuration.

The Workflow and Integration modules are not shown but can easily be implemented after these objects have been populated.

6.2 BIRT reports

Maximo Asset Management V7.1 is integrated with the Eclipse Foundation Business Intelligence Reporting Tool (BIRT). BIRT is an open source reporting system that integrates with Java or J2EE applications, such as Maximo Asset Management V7.1, to produce custom reports. BIRT utilizes XML report definitions to generate reports in PDF or HTML output. BIRT manages and displays the data from Maximo Asset Management V7.1 so users can immediately take action if necessary. User action may involve drilling down into reports to find a specific problem issue or analyzing the data for cost for regulatory purposes.

The following topics are covered in this section:

► Administering reports
► Configuring reports
► Running reports

6.2.1 Administrating BIRT reports

As the report administrator, you can specify the following:

► Availability of reports and how they open, run, and print
► Appearance of report titles and headings
► Security settings

This section provides a brief introduction to the Report Administration application. The look and feel of this application is similar to that of other applications launched from the Start Center.

You can open the Report Administration application using two methods: when initially administrating the report and after reports are defined. Follow these steps to open the Report Administration application:

1. Start the Report Administration application from the Start Center by selecting **Go To** → **Report Administration** → **Reporting** → **Report Administration**, as shown in Figure 6-1.
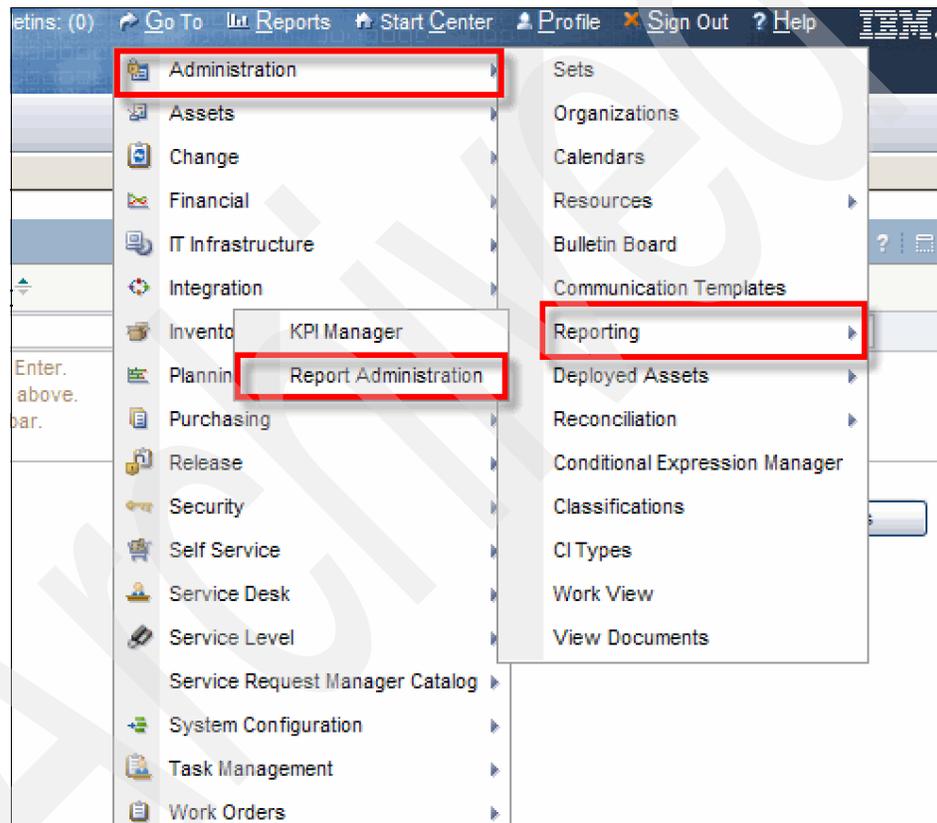
Figure 6-1   Report menu

6.2.2 Configuring BIRT reports

Several options on the Report tab enable you to configure a report. Those shown in Figure 6-2 with an (*) asterisk are required parameters.

Figure 6-2   Report configuration

The options for configuring BIRT reports are as follows:

► **Report Type**

Your choices are BIRT, Crystal, and Custom. By determining the report type and settings you register that report in the Maximo database.

► **Limit Records?**

The action limits the number of records against which a user can run a report. It prevents users from executing large queries, which can negatively impact performance. Use the Report Administration application to set record restrictions on reports. This feature applies to only reports without parameters. When you enable this option, you specify a number in the Max Record Limit option.

► **Use Where Clause?**

Enables current, selected, and user-defined parameters.

► **No Request Page?**

Disables Request Page for database updates.

► **Priority**

Numeric field used in report-queuing process.

► **Browser View?** and **Browser View Location**

The Browser View option enables you to create a shortcut. With the shortcut enabled, the user can click an icon on the application tool bar to open a report directly in the browser. When you select Browser View?, enter a value other than None in the Browser View Location field. This field determines the application tabs that have an active Browser View icon.

The following options are available for configuring the Browser View? and Browser View Location options:

– All - The Browser View icon is available on all tabs for the selected application.

– List - The Browser View icon is only available on the List tab for the selected application.

– Main - The Browser View icon is available on all tabs, except the List tab.

– None - The Browser View icon does not appear in the selected application. None is the default.

Click **Save Report** to apply the changes.

► **Direct Print?** and **Direct Print Location**

The Direct Print feature enables you to create a shortcut so a user can click an icon on the application tool bar to print the report. If you enable the Direct Print? option, you specify a printer in Direct Print Location.

► **Direct Print with Attachments?** and **Direct Print with Attachments Location**

The Direct Print with Attachments? feature enables you to create a shortcut so a user can click an application icon once (and select Yes in the Message dialog box) to print the report and any associated attached documents. When you enable the Direct Print with Attachments? option, you specify the location of the attachments.

► **Generate Request page**

Click the **Generate Request Page** button if you have not previously configured the report for Browser view. This option is available for all reports or at individual report level.

► **Preview**

You can check for the following items: The correct parameters, if any, appear to the user on the Request Page.

– The generated report opens with the correct data and format.

– The Request Page dialog box opens. The parameters displayed depend on the report that you select.

– Enter values in any required fields. Required fields have an orange asterisk (*) next to them.

From the parameters section (see Figure 6-3), you can define ad hoc user parameters. Note these parameters must also be specified in the report design itself.

Figure 6-3   Report parameters

6.2.3 Running BIRT reports

Follow these instructions to run a report. After you run a report, you can print the report, export data, and toggle the table of contents.

1. Open the Reports dialog box through one of the following methods:

    – From the Reports menu in the application tool bar, select an application - for example, **Labor** (see Figure 6-4).
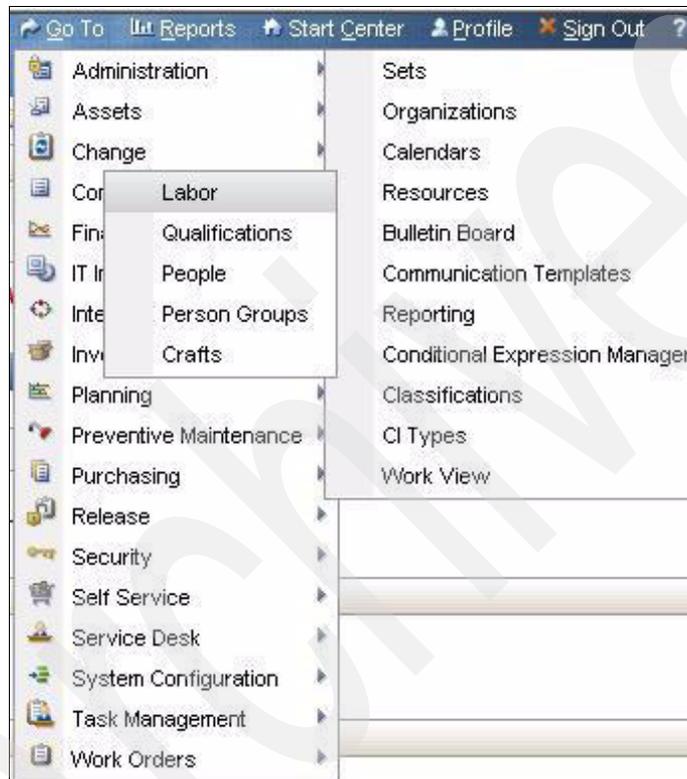
Figure 6-4   Report menu

– From the Select Action menu, select **Run Reports**. The Reports tab opens. The Reports to Run panel lists the available reports for the application (see Figure 6-5). Click the report that you want to run.
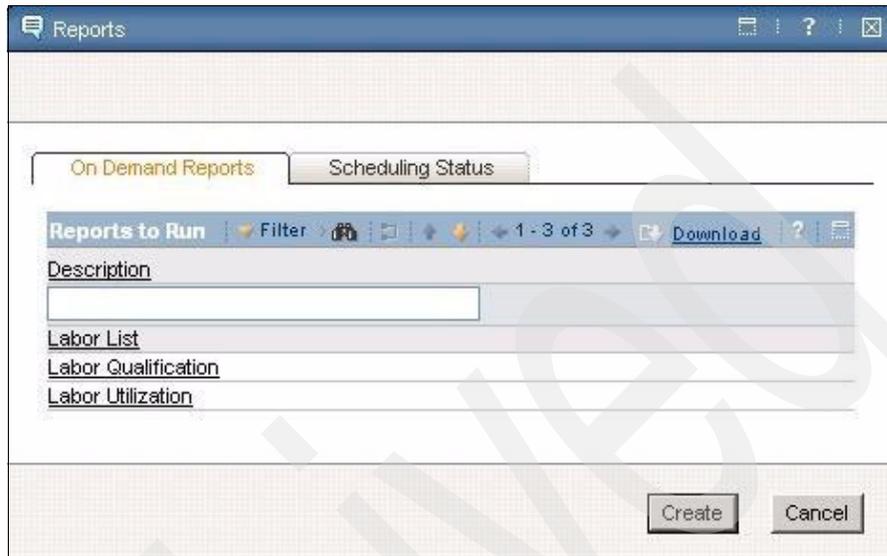
Figure 6-5   Selecting a report to run

2. Select the report you want to see - for example, Labor List. Enter the required parameters in the Request Page dialog box (see Figure 6-6).

Figure 6-6   Run Request

3. Click **Submit** to run the report. The report opens in your browser, as shown in Figure 6-7.
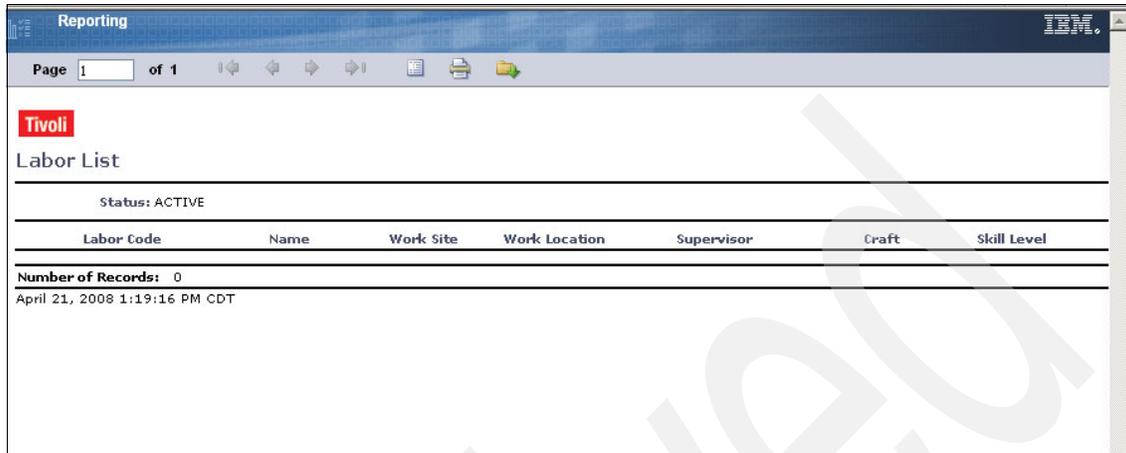
Figure 6-7   Incident report

► On the Reporting tool bar, perform any of the following actions:

– Click the Print Report as PDF icon to print the report.

– Click the Export Data icon to export the data in.CSV format.

– Click the Toggle table of contents icon to see the table of contents for your report. The report you select determines the table of contents.

To improve performance and to reduce load on the database server during working hours, it is possible to schedule report runs. Scheduled reports are then e-mailed when completed. The e-mail, in PDF format, can be sent to a single user or a group including subject and comments.

Part 4

Implementation

155

7
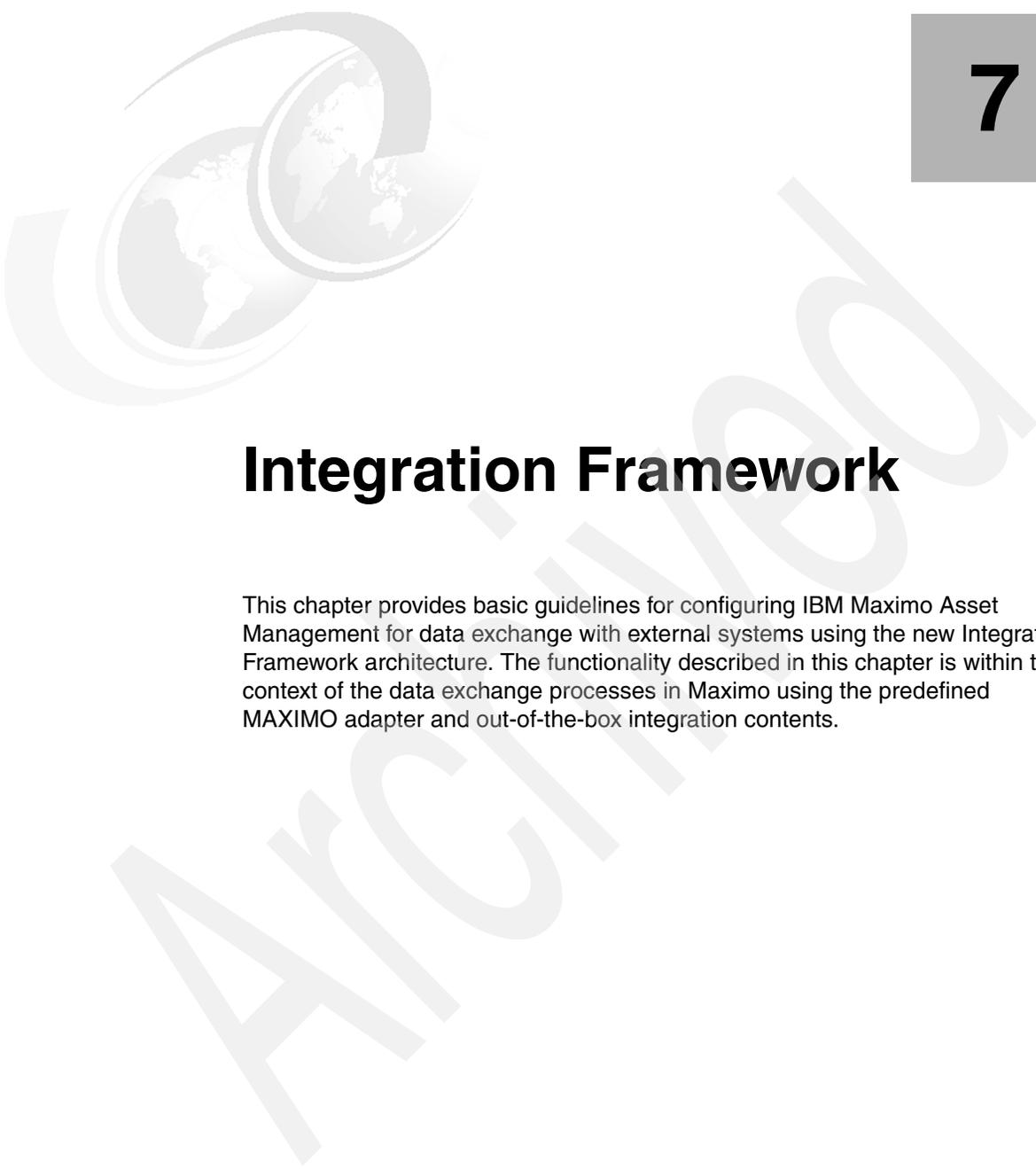
Integration Framework

This chapter provides basic guidelines for configuring IBM Maximo Asset Management for data exchange with external systems using the new Integration Framework architecture. The functionality described in this chapter is within the context of the data exchange processes in Maximo using the predefined MAXIMO adapter and out-of-the-box integration contents.

7.1 Integration Framework overview

This section provides an overview of Maximo Asset Management Integration Framework; predefined components are used to facilitate integration with external systems.

The section includes the following topics:

► Feature overview
► Integration changes from Maximo Asset Management V6.x

This section is not intended to replace detailed Maximo Integration Framework documentation. For a detail description of the Integration Framework capabilities, their associated configuration tasks, and inbound and outbound data processing, refer to the following document: *IBM Maximo Asset Management Integration Guide* at:

http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.mam.doc_7.1/mam_welcome.htm

7.1.1 Feature overview

Maximo Asset Management comes with the Integration Framework (formerly known as Maximo Enterprise Adapter - MEA) solution designed to enable you to quickly integrate and exchange data with external systems. The Maximo Integration Framework provides platform-level services definition and provisioning of Web services (J2EE). The Web services are built on the latest standards-based integration concepts and framework using service oriented architecture (SOA) technology.

The different components of an Integration Framework solution can be arranged in a layered "stack" so that each layer builds on the layers below. The object structure is the basis for applications and adapters to communicate. It is used to define message content encapsulating business objects. The processing layers operate on top of the object structure to transform and standardize messages between Maximo data exchange format and an external system's data exchange format. If using WebSphere Message Broker, the Java Message Service (JMS) queue sits on top of the processing layers that then is responsible for routing messages from the processing layers to the configured endpoint. If Message Broker is not configured, the endpoint operates on top of the processing layers to call the external systems' services. System functions such as security, metadata

management, and system management support the overall solution (see Figure 7-1).

• UI driven integration development and deployment

Define → Transform → Deploy

Define an Integration Service for Maximo Object(s)

Define transformation and processing logic

Define deployment parameters and systems

• Multiple integration options

– Web services
– JMS
– HTTP(S)
– Database
– File (XML or flat)
– Event or Batch/bulk processing
– Email and workflow connectivity

MAXIMO

Integral Part of Maximo

Synchronization of data

SOA (Service Oriented Architecture) Implementing WebServices

Extremely flexible customization

Modify existing

Add your own

External System

Figure 7-1 Integration Framework overview

The Integration Framework includes the MAXIMO adapter, which is part of Maximo base services (Tivoli's process automation engine) and is provided with Maximo Asset Management, the Change and Configuration Manager Database (CCMDB), Tivoli Service Request Manager (SRM) and Tivoli Asset Management for IT.

Furthermore, IBM provides additional add-on adapters to Maximo Asset Management for Oracle and SAP applications. The adapters can be implemented within the Integration Framework. This add-on to Maximo provides an end-to-end integration solution, thus enabling organizations to more quickly and easily connect Maximo Asset Management to their Oracle or SAP suite and leverage approved integration tools and processes on both ends.

The advanced integration functionality of the Integration Framework includes:

► A real-time engine for real-time inbound Web services queries.

► Synchronous and asynchronous JMS-based message exchange.

► Ability to build, transform, and customize message content.

► Means to configure, predefine, and create new integration definitions based on Maximo out-of-the box preconfigured integration contents, such as the following:

  – Object structures
  – Publish channels
  – Enterprise services
  – External system (EXTSYS1)
  – Adapter (MAXIMO)
  – Endpoints

► Multiple integration triggers, message format (XML, flat file, Integration table) and systems, which can be defined and managed.

► Real-time user notifications and alerts on integration issues.

► Rules-based message processing and routing through a rules engine. The rule processor can act on message values, destination, and system processing rules.

► Running as a native Maximo application server component for tighter alignment with enterprise service bus architecture and better performance.

7.1.2 Integration changes in Maximo V7.1 from Maximo V6.x

Table 7-1, Table 7-2 on page 161, and Table 7-3 on page 163 list common applications in and changes and enhancements to, respectively, the Integration Framework component from previous release versions.

Table 7-1   Changes to Integration Framework in Maximo V7.1 from Maximo V6.x

Changes	Properties
Maximo Enterprise Adapter (MEA) renamed to Integration Framework	A set of application components used to facilitate bidirectional data exchange between Maximo and external systems.
Integration object renamed object structure Integration Point removed and collapsed into the object structure	Object structures are used to define message content encapsulating business objects.

Changes	Properties
Interfaces split into outbound and inbound data exchange: ► Outbound renamed Publish Channels application. ► Inbound renamed Enterprise Services application.	► Publish Channels application is interface used for asynchronous outbound data exchange only. ► Enterprise Services application is interface used for both synchronous and asynchronous data exchange.

Table 7-2   New to Integration Framework applications in Maximo V7.1

New applications	Properties
Invocation Channels	► Synchronous outbound data exchange only. ► Similar to Publish Channels but it initiated via a Custom Action class (Java) and is synchronous (does not go through the outbound queue).
Object Structure Services	► Synchronous inbound data exchange only. ► These services rely entirely on object structure definitions and do not utilize exit processing available to enterprise services. ► Requires less configuration because no enterprise service or external system required. ► Each object structure supports five operations (create, update,delete, sync, and query). ► Query operation provides object structure as the response content. ► Create operation provides the key of the top MBO of the object structure as the response content.
Standards Services	► Synchronous inbound data exchange only. ► Created from annotated methods (such as changeStatus) in a Maximo Business Object (MBO) within an application service (such as PO). ► The inputs and outputs (if any) for these services are tied to the input and output parameters of the method.
Web Services Library	New integration application to support management and deployment of Web services for three service classes: ► Enterprise Service ► Object Structure Services ► Standards Services

New applications	Properties
Message Tracking	Async message audit trail for inbound and outbound messages: ▶ Inbound messages: Check whether an inbound message is received successfully into the queue and then whether it is successfully process into Maximo. ▶ Outbound messages: Display a list showing whether an outbound message is successfully pushed from queue into the endpoint.
Message Reprocessing	User interface front-end inbound async message correction and reprocessing, thus eliminating the need to access server to edit XML files.
▶ Logical Management Operations ▶ Integration Module ▶ Launch In Context	New Process Managers and Operation Management Products integration: ▶ Support assisted and automated approaches for Process Managers, such as Change and Release, to integrate with Operation Management Products, such as Tivoli Provisioning Manager. ▶ Maximo is fed updates of CIs, Operation Management Products, and their relationships with the IMIC adapter from TADDM (discovery data). ▶ Integration Framework supports the installation and configuration of Integration modules and Logical Management Operations. ▶ PMP processes (user interface, Workflow, Escalations) invoke Integration modules to perform actions (Logical Management Operations) on CIs via Operation Management Products. Example: Release Process Managers invokes an Integration Module to Distribute Software (Logical Management Operations) on a server (CI) using Tivoli Provisioning Manager (Operation Management Products). ▶ PMP application may also Launch (user interface) to an Operation Management Products application to support integration in an "assisted" model.

Table 7-3   Integration Framework component similar to Maximo V6.x

Applications	Description
End Points	Data exchange communication modes ► Web services ► HTTP and HTTPS ► EJB ► Flat file ► XML ► Integration tables
External Systems	Define systems integration with external systems.

This chapter is intended to be a quick reference for you to familiarize yourself with the data import and export processes in Maximo Asset Management V7.1. Therefore, the capabilities described in the following sections are discussed only within the context of the data import and export processes in Maximo using the predefined MAXIMO adapter, using Publish Channel or Invocation Channel for outbound data exchange, and using Enterprise Services for inbound data exchange. Object Structure Services, Standard Services, Logical Management Operations, Integration Modules, and Launch In Context are not covered in this chapter. Refer to *IBM Maximo Asset Management Integration Guide* for a description of the Integration Framework capabilities listed in Table 7-2 on page 161, their associated configuration tasks, and for details about inbound and outbound data processing.

> **Note:** You can access the *IBM Maximo Asset Management Integration Guide* at:
>
> http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?top ic=/com.ibm.mam.doc_7.1/mam_welcome.htm

Figure 7-2 provides an overview of the structure of the Integration Framework.
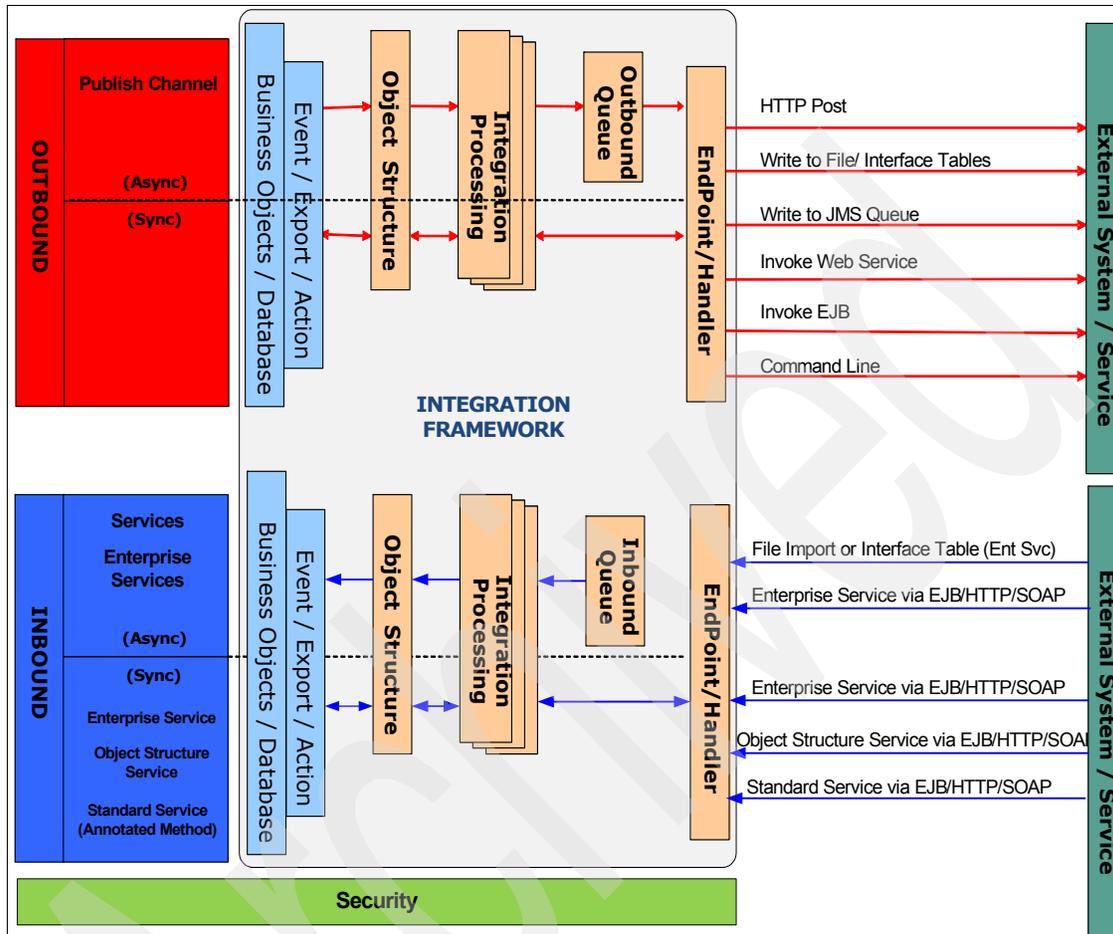
Figure 7-2   Integration Framework overview

7.2 Life cycle process for application integration

Figure 7-3 on page 165 depicts an inherent feature of the application integration process: its iterative nature by which you can refine the processes already in place and define and develop new ones. Iterations occur at two levels. The major level is the outer loop (as shown in Figure 7-3 on page 165), which involves defining business needs, implementing major programs, and evaluating their results. These processes are typically initiated by programs whose duration is measured in units of years, and require the involvement of multiple departments. Integrations also occur at the inner loop level. For each business function, there

may be multiple iterations along the inner loop. These iterations deal with the full design, application project implementations, assessment of results, refinements, and finally implementation.
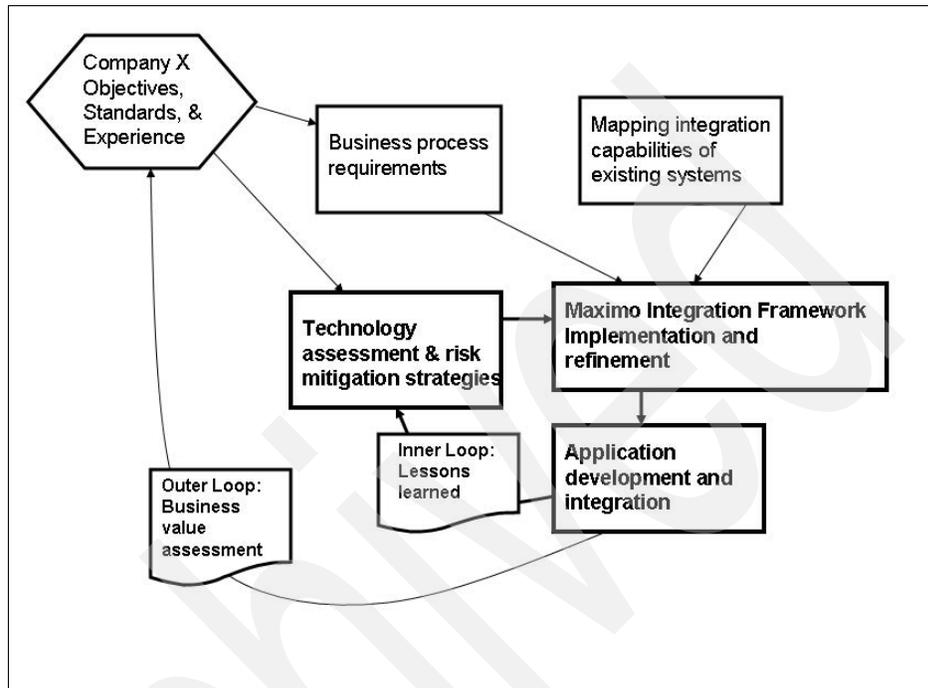
Figure 7-3 Life cycle process for Maximo integrating applications

7.2.1 Business process requirements

The business process requirements involve a top-down analysis that reviews the company's business process documentation and works with company's system users to define requirements that drive system integration priorities. This process occurs in steps; each step focuses on the most urgent business integration requirements. During this process, requirements are also sought that affect key design parameters of the overall integration system design.

7.2.2 Mapping existing system integration capabilities

This bottom-up analysis of the integration needs and capabilities for existing systems determines the best strategy for integrating existing systems to the Integration Framework. This process maps existing applications to abstract components and assesses integration alternatives for use in the development of interface adapters. This process is performed as new systems are procured.

7.2.3 Application development and integration

Applications in external systems are integrated as needed in separate, incremental steps. With each step, additional functionality and information becomes available to other authorized Maximo applications and users that previously interfaced with the Integration Framework, thereby improving the ability of the automation systems to more appropriately support business processes. Lessons learned (depicted as the inner loop in Figure 7-3 on page 165) about the strengths and weaknesses of the architecture are used to iteratively improve and evolve the integration configuration methodology of the Integration Framework.

7.3 High-level integration development steps

A project doing business process automation using the Integration Framework iterates through the following steps during the development process:

1. Document business process.
2. Identify information exchanges and define message types that are exchanged among Maximo Asset Management and an external system.

3. Extend the external system's application exchange information through the Integration Framework using a common language.

4. Extend the external system's overall message structure (for example, message type, message header expressed in according to either the Integration Framework specification or an external system's message-brokering product specification, and message payload expressed in XML).

5. Implement step 3 using the following Integration Framework processing layers:

   – Processing class
   – User exit class
   – XSL map file

6. Implement step 4 using the Integration Framework processing layers:

   – Processing class
   – User exit class
   – XSL map file.

7. Reuse or develop design patterns using pre-build Maximo components.

8. Design and implement components and adapters as needed.

9. Update the Integration Framework components configuration, then integrate and test it.

10. Apply process flow automation.

11. Perform acceptance test.

12. Deploy.

7.3.1 Recommended process for performing application integration

The process flow in Figure 7-4 summarizes the process recommended for performing application integration and automating business processes among Maximo Asset Management and external systems.
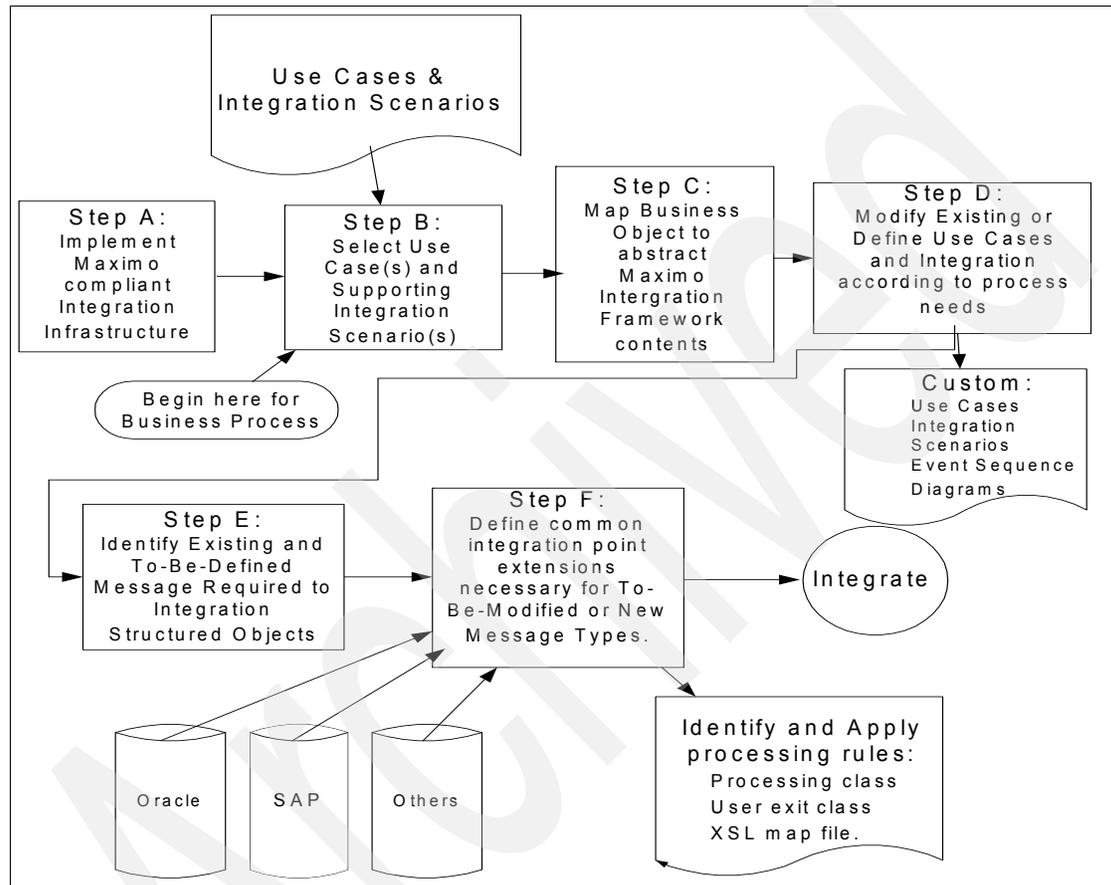
Figure 7-4   Recommended strategy for Maximo Asset Management integration

7.3.2 Project team roles

Project team responsibilities include high-level requirements, modeling, and creation of integration scenarios as described in the previous section. These goals can be accomplished by either the project team or by the project team

working in concert with an organization (for example, IT) charged with maintaining the corporate integration architecture (see Figure 7-5).
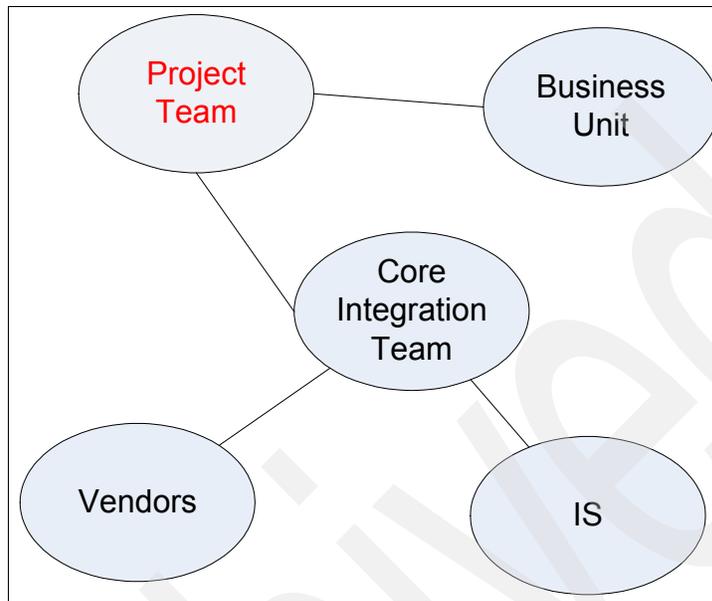
Figure 7-5   Project team integration roles

The project team responsibilities for a successful integration deployment are as follows:

Table 7-4   Integration project team responsibilities

- ► Define external system constraints.
- ► Define business drivers with business unit.
- ► Create high-level requirements.
- ► Provide timeline for overall project delivery.
- ► Check for consistency of high-level requirements.
- ► Create test plan.
- ► Create test scenarios.
- ► Work with core integration team and IS to identify hardware requirements. (Load and formatting data consumes CPU time and time on the application server and database server, which can cause long delays in the initial display.)
- ► Create integration scenarios.
- ► Initiate unit test collaborations.

> ► Create message exchange models using standards implemented by core integration teams.

7.4 Basic Maximo integration configuration checklist

After completing the checklist described in 7.3, "High-level integration development steps" on page 166, you can proceed with completing the following Maximo basic pre-configuration integration checklist.

Basic configuration tasks involve validating the following:

► System properties
► Queue and cron setup
► Web services or schema setup
► Security setup

7.4.1 Validating system properties

Verify the system settings properties listed in Table 7-5 by selecting **Go to** → **System Configuration** → **Platform Configuration** → **System Property**.

Table 7-5   System properties settings

System property	Setting descriptions
mxe.int.dfltuser	Specifies the default user ID used in all inbound integration transactions through the queue.
mxe.int.globaldir	Specifies the name of the directory to be used for all generated schema, WSDL, XML, and error files in predefined locations in the global directory. This property has no default value, and the noted files are located in the same directory as the application server default directory. If you specify an alternate location, it must be accessible from all system application servers.
mxe.int.expupdatesender	Specifies whether the Integration Framework writes the system identifier (the value of MAXVARS.MXSYSID) to the SENDERSYSID field when it generates an outbound transaction via the Data Export feature. The default value is 0 (false).
mxe.int.admintoemail	Specifies the system administrator e-mail address.
mxe.int.adminfromemail	Specifies the sender e-mail address.

System property	Setting descriptions
`mail.smtp.host`	To enable the system administrator to receive e-mail notification of errors, configure the name of the host running the SMTP server.

The System Properties window is shown in Figure 7-6.
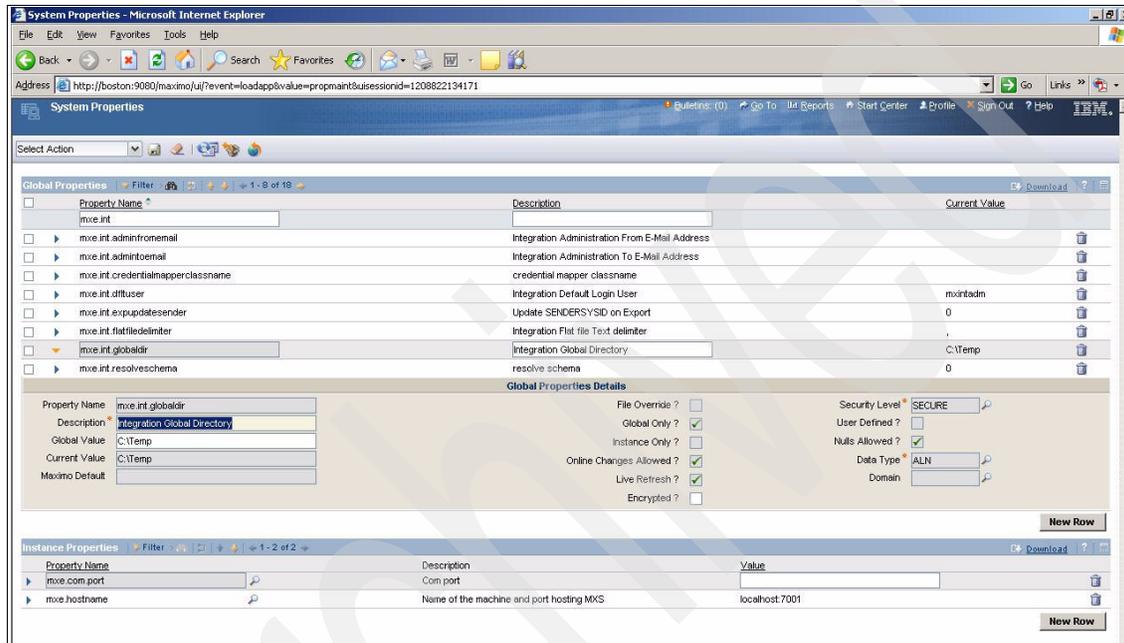
Figure 7-6   System Properties window

7.4.2 Validating queue and cron setup

To enable integration, the cron tasks must be enabled. Follow these steps to validate your queue and cron setup:

1. Verify Java Message Service (JMS) is installed on the application server by using the System Properties application (select **Go to** → **System Configuration** → **Platform Configuration** → **System Properties**). By default, JMS queues are enabled during system installations.

> **Note:** JMS is applicable only when using Enterprise Services and Publish Channels.

2. Ensure Java Message Service (JMS) is enabled:
   – cqin for continuous inbound queues
   – sqin for sequential inbound queues
   – sqout for sequential outbound queues

3. Add (if necessary) and modify the Queue dialog in the External Systems application. You can access this dialog from the Select Action menu. Here are some of the most common settings (see the *Maximo Asset Management Administrator Guide* for further detail).
   – User name
   – Password
   – Max Try Count
   – Compress
   – E-mail address

4. Turn on sequential queue CRON tasks from the Cron Tasks application (**Go to → System Configuration → Platform Configuration → Cron Task Setup** - Figure 7-7 on page 173).

   a. Enable the CRON task JMSQSEQCONSUMER - an instance for the inbound and outbound sequential queues.

   b. Activate instances and adjust the polling interval as needed (default is 30 seconds).

Figure 7-7   Cron Task setup

5. If you are using an interface table for inbound and outbound data exchange, enable the IFACETABLE cron task from the Cron Tasks application to regularly poll the MXIN_INTER_TRANS and MXOUT_INTER_TRANS queue tables for records to be processed.

6. Enable the FLATFILECO cron task from the Cron Tasks application if using flat files for inbound data import.

7. Enable the XMLFILECO cron task from the Cron Tasks application if using an XML file for inbound data import.

8. Turn on continuous Queue using Message Driven Beans (MDBs). Configuration is performed on the application server, not through Maximo Asset Management (except through the .xmi file).

   a. In the ibm-ejb-jar-bnd.xmi file, uncomment the MEA MDB section of the file to enable MDBs (requires a rebuild or redeployment of the EAR file).

b. Set the number of beans - that is, the maximum number of MDBs the system uses (default is 10).

c. Set the caching of messages - that is, the maximum number of messages the system receives from the message engine in a single batch.

d. Set the redelivery delay - that is, the amount of time that elapses from the point when a message encounters an error to when the message is reprocessed by the queue.

> **Note:** The redelivery delay is set for the whole messaging engine and impacts all queues. The setting of the redelivery delay should be less than the polling frequency of the cron tasks on the sequential queues; otherwise, they cannot poll data in the proper order (sequentially).

e. The error queue is an Application server feature. Messages that encounter errors can be redirected to a separate queue. This is not applicable for the sequential queue. If Error queues are implemented then a separate polling process or Message Driven Bean (MDB) is need to be configured for the Error queue messages to be processed.

7.4.3 Validating Web services and schema setup

Validate the Web services and schema setup from the System Property application (**Go to** → **System Configuration** → **Platform Configuration** → **System Property**). The properties for validating the Web services and schema setup are listed in Table 7-6.

Table 7-6   Web Services and schema setup

Web services/schema	Descriptions
mxe.int.uddiinqurl	Specifies the integration UDDI Registry Inquiry URL.
mxe.int.uddiname	Specifies the Integration UDDI Registry User ID.
mxe.int.uddipassword	Specifies the Integration UDDI Registry password.
mxe.int.uddipuburl	Specifies the Integration UDDI Registry Publish URL.
mxe.int.webappurl	Specifies the Integration Web application URL.
mxe.int.resolveschema	A value of true indicates the service schema does not reference other schema documents; all schema dependencies are resolved for service schemas. The default value is 0.

Web services/schema	Descriptions
`mxe.int.wsdlincludesschema`	When you set the property value to true, the WSDL files contain the complete service schema definition. The mxe.in.resolveschema must be set to true as well.
`mxe.int.wsdlnamespace`	Specifies the WSDL target namespace for the system Web services.
`mxe.int.xmlnamespace`	This property specifies the namespace used in the prolog section in the XML document.

7.4.4 Validating the security setup

Validate the security setup and mechanism to prevent unauthorized use of system components by securing the following:

► Integration queues
► EJB access
► HTTP servlet
► Web services
► Remote Integration APIs (MicService)
► Interface tables
► Outbound security
► Object-level authorization

Note: See the *Maximo Asset Management Integration Guide* for further details on enabling security:

`http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?top ic=/com.ibm.mam.doc_7.1/mam_welcome.htm`

7.5 Object structures

An object structure is the building block of the Integration Framework that enables integration applications to perform the following functions:

► Publish and query application data.

► Add, update, and delete application data through the following operations:

 – Sync
 – Create
 – Update
 – Delete
 – Query

► Import and export application data.

Each integration object consists of the following:

► Persistent
► Non-persistent
► User-defined field

Maximo Asset Management V7.1 includes various predefined object structures out of the box to facilitate the development of integration contents. In creating object structures, instead of having to create new components, you can duplicate and use the existing predefined components as a template.

> **Note:** If using object structures in interface tables or flat files, ensure that the column names in the object structure are unique. If duplicate names exist, modify the alias values for the duplicate columns. To determine whether conflicts are caused by duplicate fields, click **Select Action** and select **Add/Modify Alias**.

> **Note:** Interface tables require that the alias names for all columns included in the corresponding object consist of 18 or fewer characters.

Configuring Object Structures

Use the Object Structure tab in the Object Structures application (**Go To** → **Integration** → **Object Structures)** to perform the following configuration activities:

1. Create an object structure.

2. Add the system objects to the object structure.

3. Assign the processing classes (inbound or outbound) to the object structure.

4. Specify whether the object structure supports a flat file representation.

5. Restrict object structure supported operations to QUERY.

6. Use the Select Action menu in the Object Structures application to perform the following configuration activities:

   a. Configure the column aliases to resolve alias conflicts in the object structure.

   b. Include and exclude the system object columns from the object structure.

7.6 Publish Channels

Publish Channels are used to create and manage asynchronous outbound data flow from the Integration Framework to an external system (see Figure 7-8).
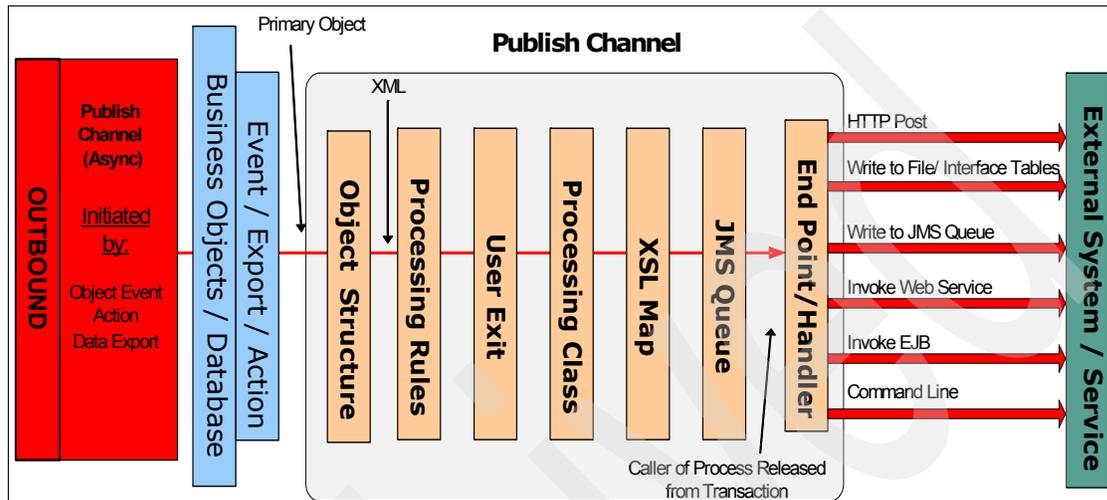
Figure 7-8   Publish Channels

Publish Channels provide the framework to:

- ► Transform object structures to service input
- ► Associate an endpoint
- ► Transform service response to object structures

Events that initiate Publish Channel processing are:

- ► Object events (insert, update, and delete)
- ► Application-initiated calls
- ► Data export

The Publish Channel can use the following processing layers to map the XML to the external system XML:

- ► Processing class
- ► User exit class
- ► XSL map file

The Publish Channel can apply specified processing rules to objects before it saves the objects. Processing rules are used to access and retrieve pertinent data from objects that are not included in the object structure.

You can override the behavior of predefined data processing that the enterprise service supports through the integration controls. The integration control behavior is implemented through enterprise service processing rules and Java processing classes.

From within Publish Channels, you have the option to change the interface table name for the service or channel.

Configuring Publish Channels

Use the Publish Channels tab in the Publish Channels application (**Go To** → **Integration** → **Publish Channels)** to perform the following configuration activities:

1. Create a Publish Channel.

2. Identify the object structure.

3. Assign the interface table name (if using an interface table).

4. Assign the processing classes.

5. Create the processing rules.

Use the Select Action menu in the Publish Channels application to perform the following configuration activities:

1. Create the integration controls.

2. Configure the Publish Channel integration controls and assign the default values.

3. Generate the schema files to describe the Publish Channel.

4. Enable and disable the listener for the outbound integration events

Note: Publish Channel configuration is required if you are working with external system and queue-based integration scenarios.

7.7 Invocation Channels

The Invocation Channels application is used to create and manage synchronous outbound data flow (that is, data flow that does not go through the outbound queue) from the Integration Framework to an external system (see Figure 7-9). The Invocation Channels application is similar to Publish Channel, but it is initiated with a custom Action class (Java).
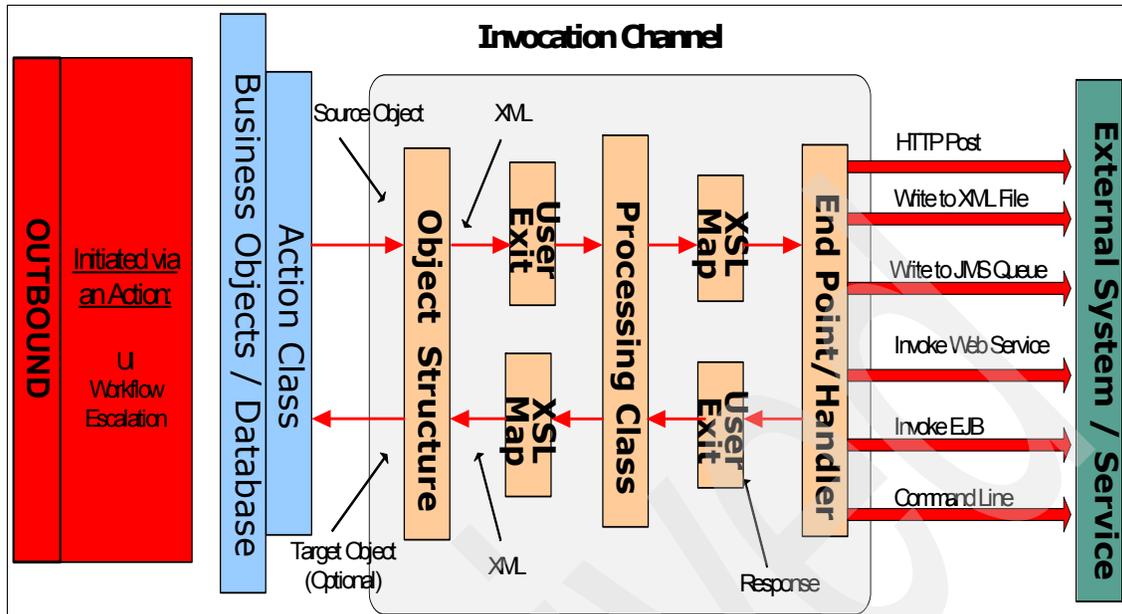
Figure 7-9   Invocation Channels

Configuring Invocation Channels

From the Invocation Channels tab in the Invocation Channels application (**Go To → Integration → Invocation Channels)**, perform the following configuration activities:

1. Create an invocation channel.

2. Identify the object structures for request and response processing.

3. Assign an endpoint.

4. Assign the processing classes.

Use the Select Action menu in the Invocation Channels application to generate schema files to describe the invocation channel.

7.8 Enterprise Services

Enterprise Services are used to create and manage both synchronous (without a queue) and asynchronous (with a queue) inbound data flow from an external system to the Integration Framework.

Table 7-7 lists supported protocols for Enterprise Services.

Table 7-7   Enterprise Services inbound messaging protocol

Inbound type	Protocol
Asynchronous (with queue)	► Flat files ► Interface tables ► EJB, HTTP, HTTPS, SOAP
Synchronous (without queue)	► EJB, HTTP, HTTPS, SOAP

For asynchronous messaging, message flows from the external system to the inbound queue (see Figure 7-10). Polling threads (through the Cron Tasks application) pull the message from the queue and send it to Enterprise Services processing layers.
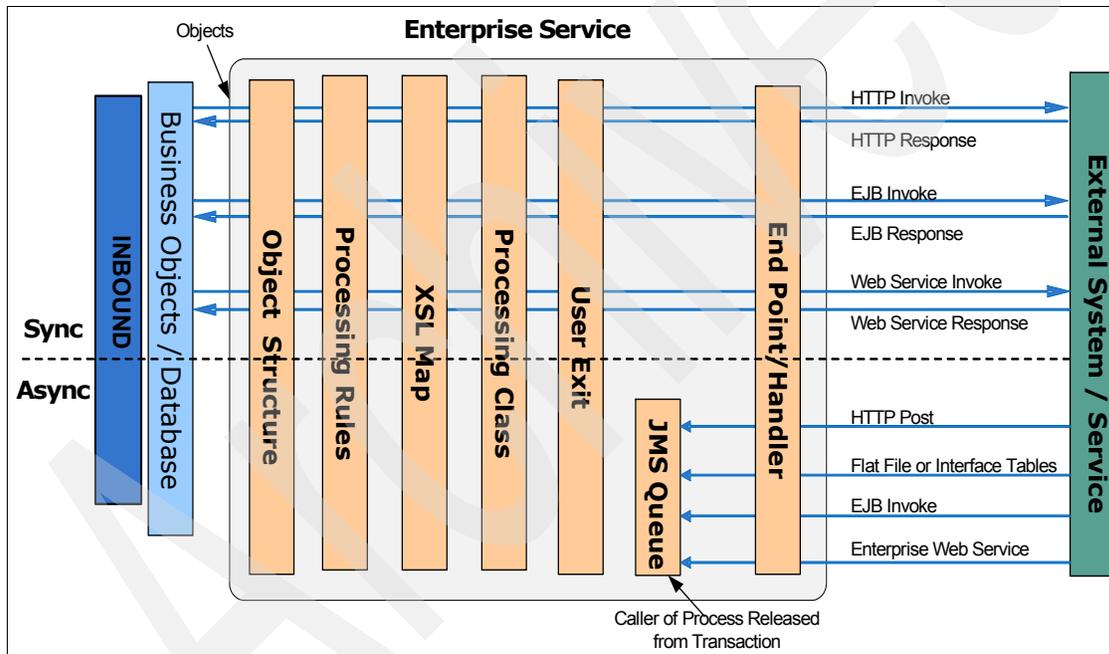
Figure 7-10   Enterprise Services

The Enterprise Services processing layers provide the following to map the XML to the object structure XML:

► Processing class
► User exit class
► XSL map file

You can apply processing to both the Enterprise Services configuration and Enterprise Services response configuration for query and create operations.

Configuring Enterprise Services

From the Enterprise Services tab in the Enterprise Services application (**Go To → Integration → Enterprise Services)**, perform the following configuration activities:

1. Create an enterprise service.
2. Specify the object structure.
3. Assign the supported operation.
4. Assign a multiplication control.
5. Assign an interface table name.
6. Identify whether you are using an external schema.
7. Assign a split tag that you use to process multi-noun messages.
8. Assign the processing classes (request or response).
9. Create the processing rules.

> **Note:** A fully qualified XPATH expression must be provided in the Split Tag field when you use an external schema. The XPATH expression indicates where the Integration Framework splits the multi-noun messages with an XML definition.

From the Select Action menu in the Enterprise Services application, perform the following configuration activities:

1. Create the integration controls.
2. Configure the Enterprise Services integration controls and assign default values.
3. Generate the schema files to describe the Enterprise Service.

> **Note:** Enterprise Services configuration is required when you are working with external system and queue-based integration scenarios.

7.9 External Systems

External Systems enables you to synchronize inbound and outbound data through Enterprise Services and Publish Channels, respectively.

Configuring External Systems

From the System tab in the External Systems application (**Go To** →
Integration → **External Systems)**, perform the following configuration activities:

1. Create an external system.

2. Identify the queues to the external system.

3. Assign an endpoint to the external system.

From the **Enterprise Services** and **Publish Channels** tabs in the External
Systems application, perform the following configuration activities:

1. Add Enterprise Services and Publish Channels to the external system.

2. Disable unused Enterprise Services and Publish Channels.

3. Specify the inbound JMS queue the Enterprise Services use.

4. Optional: Specify the endpoint that the Publish Channel uses to override the
   External Systems endpoint.

7.10 Endpoints

Outbound transactions (Publish Channels or invocation Channels) for your
business processes need endpoint values. The endpoints specify how the
transactions are delivered and the handler that routes the transactions to the
endpoint.

The predefined endpoints for the MAXIMO adapter are listed in Table 7-8.

Table 7-8   Endpoints

Endpoint	Handler	Description
MXXMLFILE	XMLFILE	XML File endpoint
MXIFACETAB	IFACETABLE	Interface table endpoint
MXFLATFILE	FLATFILE	Flat file endpoint
MXCMDLINE	CMDLINE	Command-line endpoint
TADDMEP	TADDMAS	TADDM authorization synchronization endpoint

7.11 Importing data

The following steps describe inbound processing that uses the data import feature. Before importing data, be sure to cross-check all fields and data for validity.

1. The Integration Framework checks that the external system and the Enterprise Services are valid and enabled.

    If you are importing flat files, the Integration Framework also checks that the Enterprise Service Object Structure supports flat structures. If the verification fails, the Integration Framework issues an error and does not process the message.

2. If the verification is successful, the Integration Framework identifies the inbound JMS queue that is assigned to the Enterprise Service and the external system.

3. The Integration Framework writes the message to the inbound queue. If the message contains multiple instances of a document - for example, if a single message contains ten person records - the application writes ten messages to the queue.

    If one of the referred records has a processing error, a single message-processing exception is identified, and none of the other messages that can be successfully processed are committed to the database.

4. The Integration Framework updates the message header with the external system and Enterprise Services names.

    You have the option to preview the data that you want to import to check the format and the data validity of a source file. A synchronous validation mechanism presents all the source file processing errors without committing data to the database.

The preview data process verifies that the data structure of the selected file complies with the integration XML or flat file definitions.

Data integrity testing

When data import is completed and the data has been moved between Maximo Asset Management and the external systems, you can test overall integration. We recommend you test the following when checking data integrity:

► Test system integration.
► Test the accuracy of data coming from external systems.
► Test the accuracy of data going into external systems.
► Test the scenario to ensure that the availability of the systems is not affected adversely.

For a data import example, see 8.4, "Integration Framework scenario" on page 234.

Integration Framework summary

The Maximo Asset Management Integration Framework provides platform-level services definition and provisioning of Web services in a service oriented architecture (SOA). It is capable of generating Web services for any of the Maximo business components and of hosting these Web services for external interactions - for example, an enterprise portal requesting performance metrics for a set of assets.

With the Integration Framework platform, one can connect Maximo Asset Management with other external systems using different integration mediums such as file systems, databases, message queues, and Web services. This enables bidirectional synchronous and asynchronous data exchange more easily and effectively.

Another benefit of the Integration Framework is the vast array of predefined out-of-the box integration libraries that support various integration scenarios between Maximo Asset Management and an external system, thus enabling quicker implementation with less complex integration procedures.

8

Deployment scenarios

This chapter describes how to use some of the new features in Maximo Asset Management in real scenarios to expedite your deployments. These scenarios are of a technical nature and require technical expertise.

Note: The features discussed in this chapter are not all of the new features released with Maximo Asset Management V7.1. However, they cover scenarios that pertain specifically to the deployment process in Maximo V7.1.

8.1 BIRT reports scenario

As a BIRT report administrator, you have received two reports from a developer. One of the reports is new, and another report is a modification of an existing report. As an administrator you want to deploy and test the reports. In addition you want to move the reports to the production environment after you have completed testing them.

The developer delivered the following two reports:

► New Maximo Asset Management report: valuable_asset.rptdesign

► Modified report: asset.rptdesign

8.1.1 Implementation steps

As the report administrator, you are responsible for the following high-level tasks:

► Importing reports into Maximo Asset Management
► Administering reports
► Running reports
► Logging messages

Importing reports into Maximo Asset Management

This section describes steps for importing a new or an existing report into Maximo Asset Management. In our example we import the reports from the command prompt. You can also import reports from the Report Administration console, albeit only one report at a time. To import more than one report at a time use the import as follows:

1. Move the reports to the proper location, which is the <maximo>\ reports\birt\reports\ASSET folder.

2. Configure the properties file.

3. Locate the properties reporttools.properties in <maximo>\ reports\birt\tools.

4. Configure the host name, port, user name, and password for the Maximo Asset Management server:

```
maximo.report.birt.hostname=<MAM server>
maximo.report.birt.port=<port>
maximo.report.birt.username=<user>
maximo.report.birt.password=<password>
```

5. Start the Maximo services.

6. Make sure you add new messages to the
   <maximo>\reports\birt\libraries\asset.properties file.

7. If this is a new report, add a section in the reports.xml file in <maximo>\
   reports\birt\reports\ASSET folder:

```
<report name=" valuable_asset.rptdesign ">
<attribute name="filename">valuable_asset.rptdesign</attribute>
<attribute name="description">Sample report One</attribute>
<attribute name="norequestpage">0</attribute>
<attribute name="detail">0</attribute>
<attribute name="reportfolder">ASSET</attribute>
<resources>
<resource>
<reference>tsd.properties</reference>
<filename>${libraryfolder}/tsd.properties</filename>
</resource>
</resources>
</report>
```

At this point, configuration is complete, and you are ready import your reports.

Running the Import command

To run the Import command from a command line, change the directory path to
<maximo>\ reports\birt\tools:

```
<maximo>\ reports\birt\tools\importreports app ASSET
```

The results of this command are shown in Example 8-1 on page 188.

Example 8-1   Successful import of reports

```
init:
     [echo] Using the following settings:
     [echo] hostname: localhost
     [echo] port: 7001
     [echo] ssl: false
     [echo] username: maxadmin

app:
     [echo] Importing Reports.
[importreports] Importing reports from: C:\harrieream\deployment\harrier\re
\birt\tools\..\reports
[importreports] Importing reports for app: ASSET
[importreports] Importing reports defined in: reports.xml
[importreports]     Importing report: asset_availability.rptdesign
[importreports]     Importing report: detailasset_fail.rptdesign
[importreports]     Importing report: drillasset_fail_tbl.rptdesign
[importreports]     Importing report: sumasset_fail.rptdesign
[importreports]     Importing report: asset_costrollup.rptdesign
[importreports]     Importing report: asset_costrollup_update.rptdesign
[importreports]     Importing report: assetmove_history.rptdesign
[importreports]     Importing report: asset_glaccount.rptdesign
[importreports]     Importing report: asset_po.rptdesign
[importreports]     Importing report: mgmt_sw.rptdesign
[importreports]     Importing report: asset_detail.rptdesign
[importreports]     Importing report: asset_subassembly.rptdesign
[importreports]     Importing report: asset.rptdesign
[importreports]     Importing report: asset_measurehist.rptdesign
[importreports]     Importing report: oeekpi_by_site.rptdesign
[importreports]     Importing report: oeekpi_by_location.rptdesign
[importreports]     Importing report: oeekpi_by_asset.rptdesign
[importreports]     Importing report: linear_work_history.rptdesign
[importreports]     Importing report: gaps_overlaps.rptdesign

BUILD SUCCESSFUL
Total time: 21 seconds
```

Administering reports

The information in the following sections describes key report administration tasks.

Generate request page

An important report administration task is generating a request page. Follow these steps:

1. Start the Report Administration application from the Start Center by selecting **Go To** → **Administration** → **Reporting** → **Report Administration** as shown Figure 8-1.
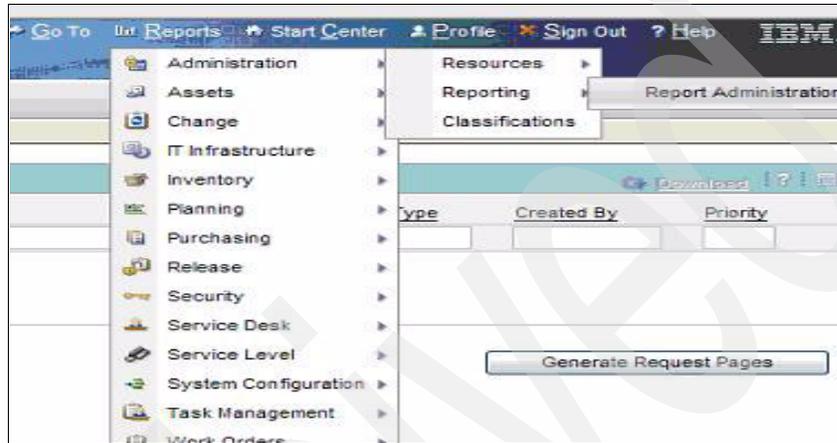
Figure 8-1 Report Administration menu

The Report Administration window is displayed (see Figure 8-2). The window is composed of three tabs: List, Report, and Security. To list only your reports, you can enter `asset` or enter the name of your report in the Report File Name field to filter this view for the desired reports.

Figure 8-2   Report List page

Select a report from the List tab and then click the **Generate Request Page** button (see Figure 8-3). On successful completion, you have the option to preview the reports.

Figure 8-3   Generate Request page

Set application security

Application security settings in the Report Application Security window (see Figure 8-4) enable you to set group security for all reports in a selected application. The MAXADMIN group has access to all out-of-the-box reports. You must set up group or report access to each individual application for new or customized reports.
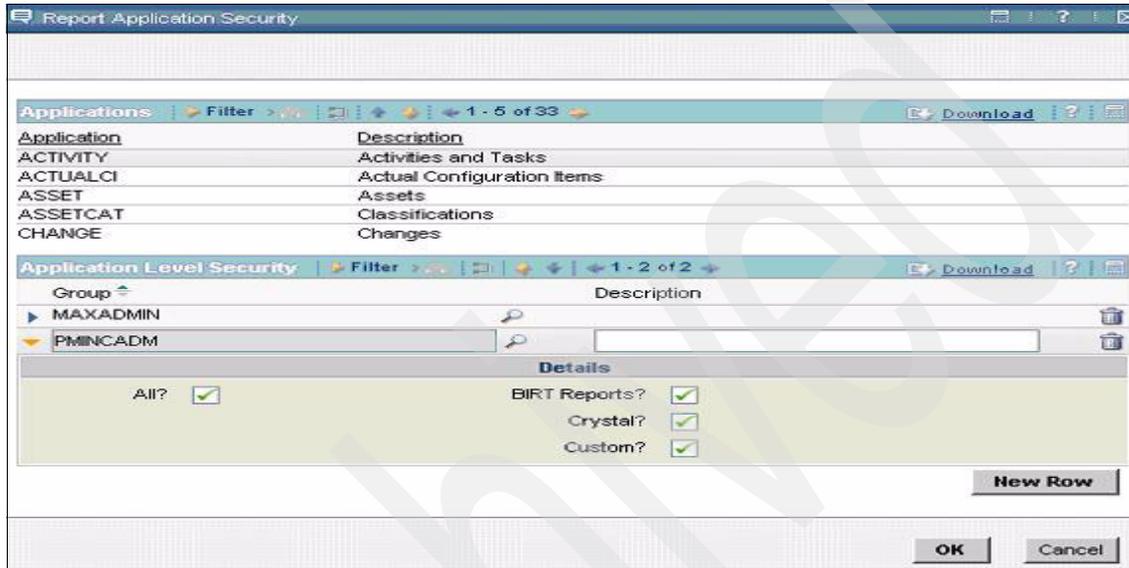
Figure 8-4   Report Application Security

Various options are available for setting report security. You can set security at the application or individual report level.

To access Group security, select **Action** → **View Group Security**. Click the lookup symbol, and select the desired group. You must set up application access for other groups individually.

Run reports

Follow these instructions to run a report. After you have run a report you have the options to print, export data, and toggle the table of contents.

1. From the Reports menu in the application tool bar, select an application. Doing so enables you to view added reports. In our scenario, we select **Assets** → **Assets**.

2. From the Select Action menu in the Assets application, select **Run Reports**.

   The Reports tab opens. The Reports to Run table lists the available reports for the application. The newly imported report as well other reports related to the Assets application are displayed.

3. Select the report you want to view - for example, Asset List. Enter the required parameters in the request page. Click **Submit** to run the report.

To improve performance and to reduce load on the database server during working hours, it is possible to schedule report runs. Scheduled reports are then e-mailed when completed. The e-mail, including subject and comments, can be sent to a single user or a group. The report is e-mailed in PDF format.

8.2 Conditional Expression Manager scenario

The following scenario demonstrates the use of the Conditional Expression Manager in defining conditional option access to an application.

In this scenario, our organization uses Maximo Asset Management to maintain various types of assets - namely IT, fleet, and production, which are owned by the IT, Fleet, and Production departments, respectively. The business structure of this organization allows these departments to view and update any asset. We want to ensure and maintain that each department has the right to create or update only its assets and not those of the other departments.

To achieve this we have to perform the following steps:

1. Create a condition.
2. Apply the condition.
3. Test the scenario.

Create a condition

The first step is to create a condition based on the scenario described earlier. To achieve this, perform the following:

1. Log on to Maximo Asset Management with an administrative account - for example, maxadmin.

2. Select **Go To** → **Administration** → **Conditional Expression Manager** (see Figure 8-5).

Figure 8-5   Conditional Expression Manager

3. Click **New Row**.
4. Provide the name of a condition - for example, FLEET_ASSET.
5. Enter a valid description.
6. The Type field by default should display EXPRESSION.
7. Click the **Condition Expression Builder** button.
8. Select Object as ASSET - The Asset Table.
9. Build the following expression:
   ```
   :assettype = 'FLEET'
   ```

10. Validate the expression by clicking **Test Expression** under the Miscellaneous section, as shown in Figure 8-6.
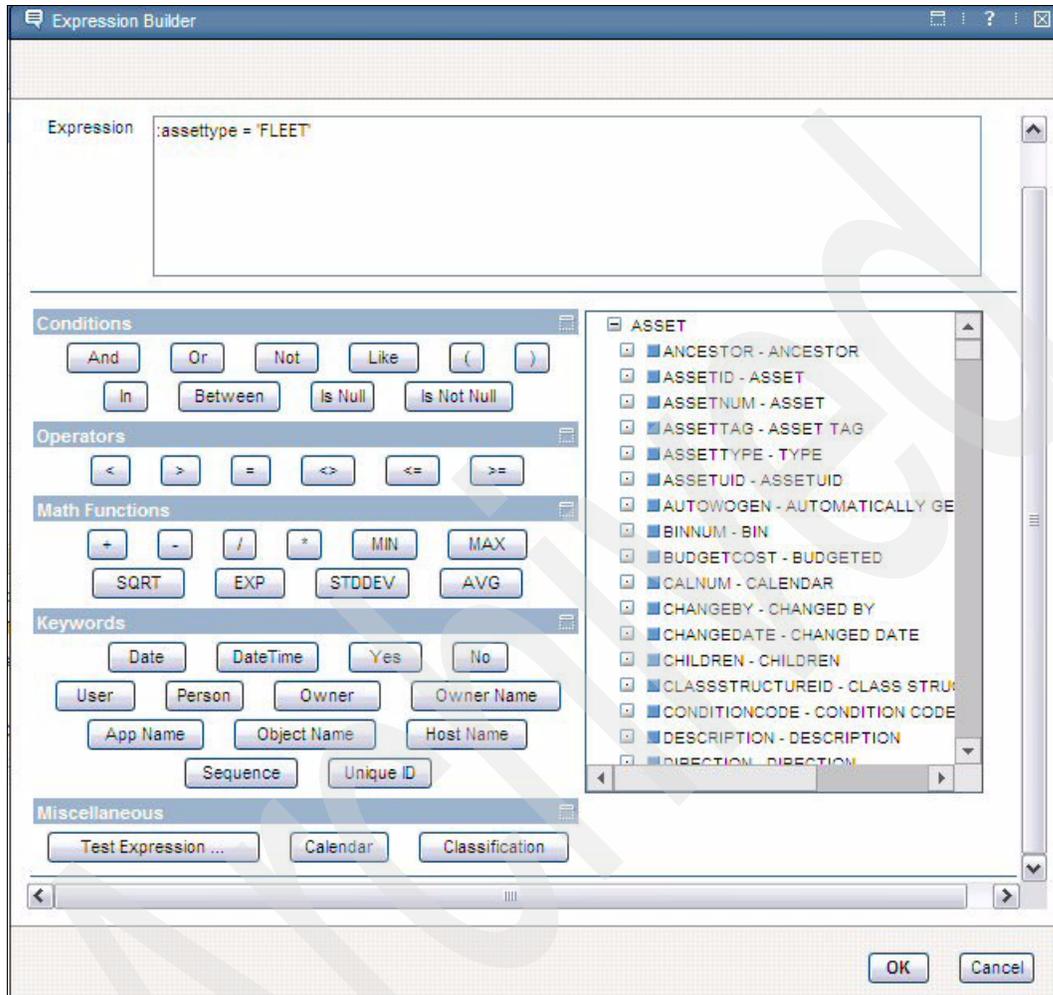
Figure 8-6   Expression Builder - Test Expression button under Miscellaneous

11.Click **OK** and Save the record. Figure 8-7 the window displayed when a condition has been successfully created.
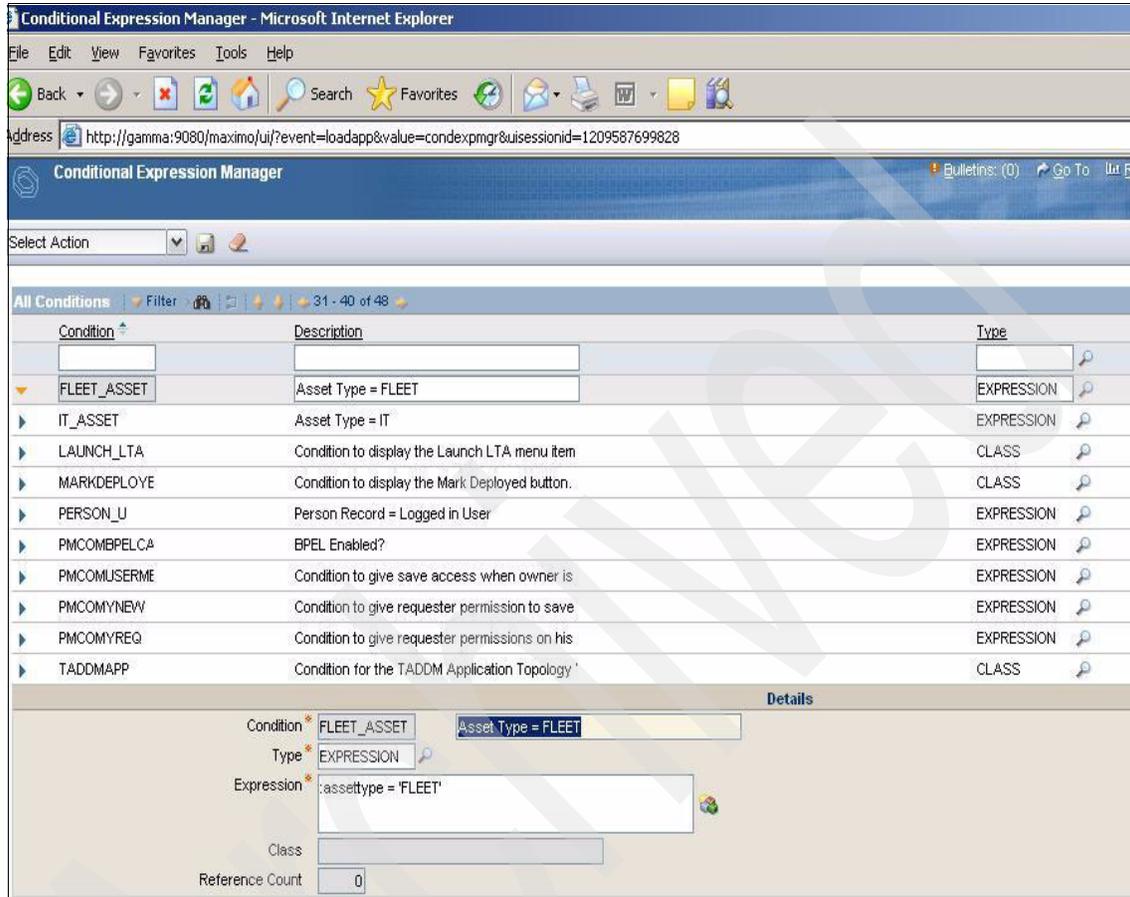
Figure 8-7   Condition created

> **Tip:** Ensure that the entries in the Condition and Expression fields are unique throughout the application.

Apply the condition

The next step is to apply the condition to the relevant security group and its associated users. Make sure you have created the relevant security group and user. To create a new security group, perform the following steps:

1. From the Start Center, select **Go To** → **Security** → **Security Groups**.

2. Create a new security group for FLEET assets. In our example, we name the security group FLEET _ASSET.

3. From the Sites tab, specify the sites applicable to this security group.

4. Access the Application tab to assign and determine access to the group.

5. Select the **Assets** application.

6. Grant access to the following options by checking the check boxes in the Options for Assets section:

   – Delete Asset
   – New Asset
   – Read access to Asset
   – Save Asset

7. Select the condition for Delete Asset as FLEET_ASSET.

8. Select the condition for New Asset as FLEET_ASSET (see Figure 8-8).
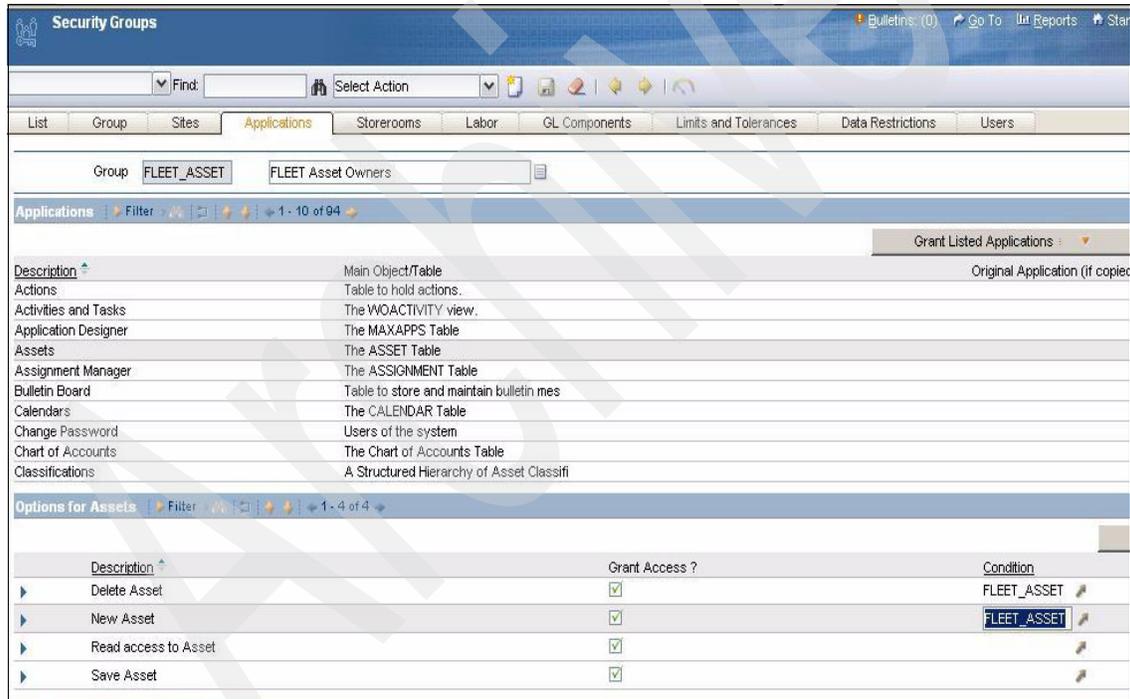
Figure 8-8   Conditional Option access

Important: Ensure that options are granted before selecting the condition.

9. Next, create a new user that belongs to the security group FLEET _ASSET. To do so, complete the following steps:

   a. From the Start Center, Select **Go To** → **Security** → **Users**.

   b. Create a new user. In our example, our new user name is DAVE. Assign the relevant user details and password. Do not save your changes yet.

   c. Click the **Groups** tab, select **New Row,** and assign the user to the group you have just created - in our example, FLEET _ASSET. Click **Save**

Test the scenario

At this point, we test our conditional expression. First, we log on to the application as the user David and perform the test cases described in this section.

> **Important:** To ensure the test is valid and properly filters the assets, ensure you have a few assets of type FLEET and a few non-FLEET assets in the Assets application.

► **Insert asset -** This case validates that users of a particular department or security group are able to create assets of a particular type only. For our example, the asset type is FLEET.

   a. Select **Go To** → **Assets** → **Assets**.

   b. Click **New Asset** icon.

   c. Give the Asset number - for example, 314159.

   d. Select Type as IT and click **Save**.

      A system message is displayed, declaring the Object asset is read-only.

   e. Change the Asset Type to FLEET and click **Save**.

      The record is saved.

► **Update Asset -** This case validates that users of a particular department or security group can update assets on any type.

   a. Select **Go To** → **Assets** → **Assets**.

   b. Select a non-FLEET type asset.

   c. Set the priority to 1 and click **Save**.

      The record is saved.

   d. Select a FLEET type asset.

   e. Set the priority to 2 and click **Save**.

      The record is saved.

If the tests complete successfully, the condition is applied to the user as specified.

Similarly, the Conditional Expression Manager can be used to configure conditional data restrictions and a conditional user interface using the Security Groups and Application Designer applications.

8.3 Migration Manager scenario

This scenario assumes you have already created a workflow in your development environment, and want to migrate it to your test acceptance or production environment.

> **Attention:** We strongly recommend that you have a robust change management process in place. The tool we describe in this section provides the technical capabilities to migrate and deploy these changes, but it is still important that you review test, approve, and document the changes before you save them.

In this section, we demonstrate a scenario using the Migration Manager, creating a new workflow process definition (along with related objects, such as roles, actions, and notifications) on a source environment and migrate the workflow and related objects to a target environment.

> **Note:** The example shown in this scenario is based on a Service Request workflow. We have borrowed this scenario from a soon-to-be-released IBM Redbooks publication on the Service Request Manager product. However, this scenario applies equally well to the Maximo Asset Management product.

We demonstrate how a new workflow process package definition is created or changed on the source environment, how the process of distribution is handled, and how content is deployed and validated on the target environment.

We duplicate and change an existing Migration group in this example to migrate only the specific objects we wish to migrate. We skip discussions of part of the package definition and dependent Migration groups, but we discuss and show each of the steps in the process.

The environment we use in our scenario is shown in Figure 8-9.
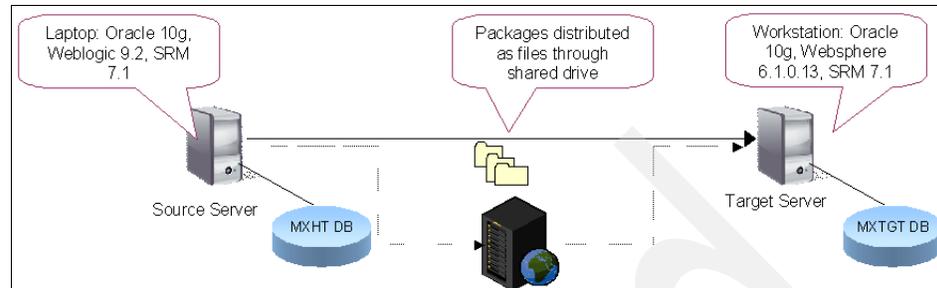
Figure 8-9   Package deployment scenario

To begin our examples, we have already logged on to both environments, describing the steps that follow this initial step.

8.3.1 Workflow example

If you are familiar with workflow, you know a workflow can consist of several different types of nodes, each with its specific functions and content. The workflow we want to migrate from source to target is a simple workflow for service requests, as shown in Figure 8-10.
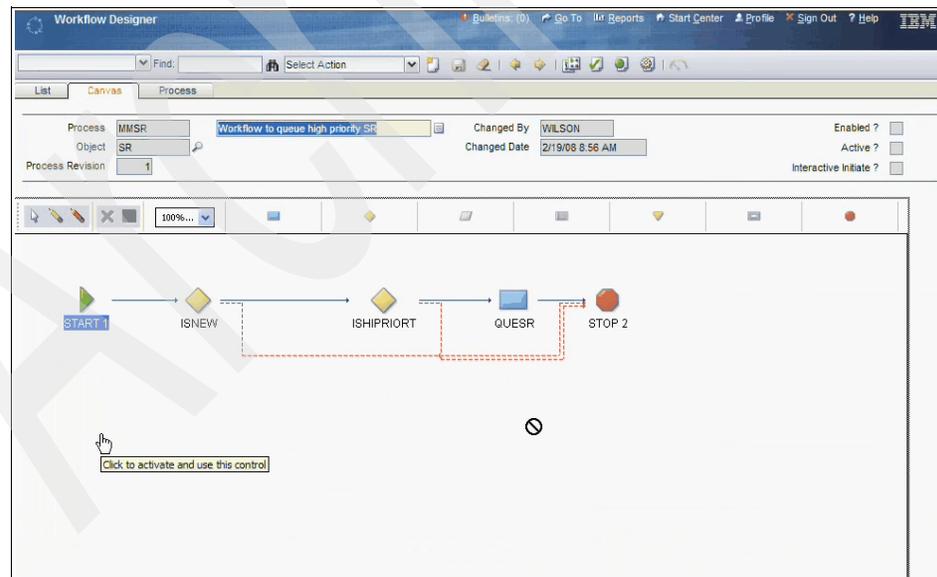
Figure 8-10   Example workflow

The workflow contains two condition nodes and one task node and is created to set the internal priority of a service request to 1, in case the request is created with a high (external) priority. The first condition node checks whether the SR is new, the second whether the SR is marked as a high-priority SR. The task node assigns and queues the SR to the Service Desk Manager, while a positive action on the node sets the internal priority to 1 and notifies the Service Desk Agent.

In this example, we define a task node with the role SDMGR to use for the assignment, as shown in Figure 8-11.

Figure 8-11   Example workflow role

On a positive result of the task node, we also define the action MMACT to set the SR property and notify the owner of the SR that the priority has been set. For this purpose, we use the role SDAGENT and the communication template MMNOTIF, as shown in Figure 8-12.

Figure 8-12   Example workflow action, role, and notification

To successfully migrate the workflow from source to target, we need to migrate not only the workflow, but also the related roles, actions, and notifications used in the workflow.

8.3.2 Package definition

To migrate the example workflow using the Migration Manager functionality, follow these steps:

1. From the Go To menu, select **System Configuration** → **Migration** → **Object Structures**, as shown in Figure 8-13.
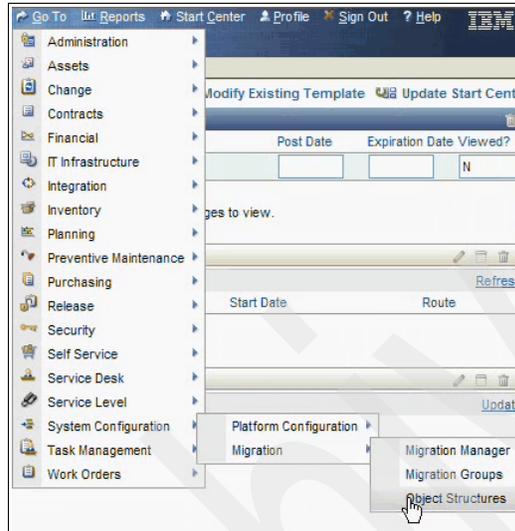
Figure 8-13   Migration menu - Object Structures

2. Search and select the DMWFPROCESS object structure. The object is shown in Figure 8-14.
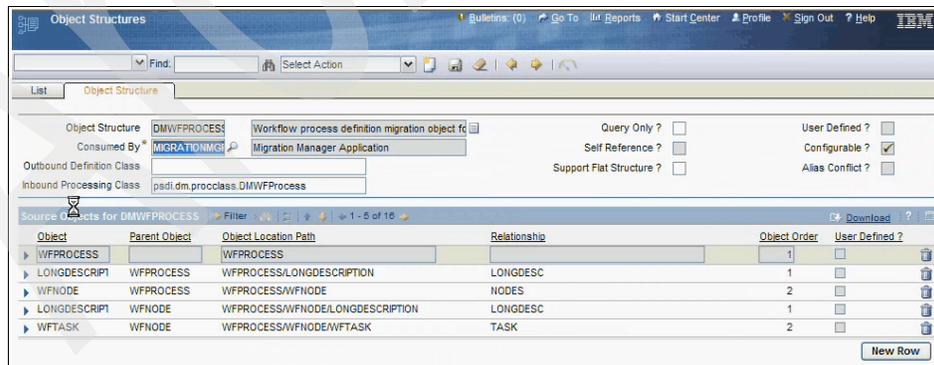
Figure 8-14   DMWFPROCESS Object Structure

In the Object Structure application, you find the source objects (MBOs) you want to migrate. These objects have a parent-child relationship you can find in the columns Object Order and Object Location Path. Because we use an out-of-the-box object structure for the workflow, the User Defined? field is not selected, and the file is read-only.

In this example, all workflow-related source objects are included, and an inbound Processing class psdi.dm.procclass.DMWFProcess is specified. Specifying this processing class ensures the imported objects are processed through the appropriate business logic.

We leave the object structure as it is and move on to the next step.

3. From the Go To menu, select **System Configuration** → **Migration** → **Migration Groups**, as shown in Figure 8-15.
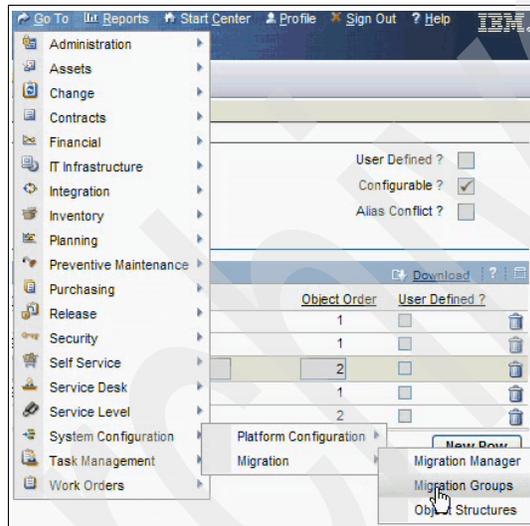
Figure 8-15   Migration menu - Migration Groups

4. Search and select the out-of-the-box BPM Migration Group. You find the group as shown in Figure 8-16.

Figure 8-16 BPM migration group

The BPM Migration Group contains all the migration objects we need to migrate, as identified earlier in this chapter. By default you find action, role, communication template, escalation, workflow, and E-mail Listener (inbound communication) objects in the BPM Migration Group.

Looking at dependencies, several other migration groups have dependencies with the BPM Migration Group, such as Data Dictionary, Application, and Resources. These migration groups with dependencies can be found in the section Dependency of the Migration Group application.

Figure 8-17 shows a representation of the Migration Group structure with the migration objects and migration group dependencies in the form of a tree.
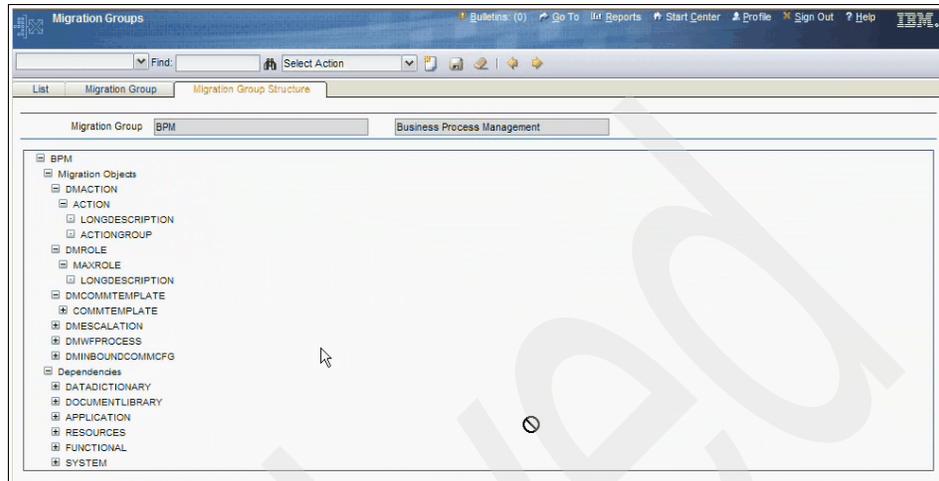
Figure 8-17   BPM migration group structure

5. Because we want to migrate only specific workflow artifacts and not all dependent objects, we duplicate the existing BPM group. From the Select Action menu, select **Duplicate Migration Group**.

6. We call the new migration group MYBPM, so enter MYBPM in the **Migration Group** field. We also know we have one workflow, one action, one notification, and two roles to migrate, but we are not interested in any dependent objects and escalations.

7. Remove the DMESCALATION migration object from the migration objects by clicking the recycle bin at the end of the record.

8. Remove all Dependency records by clicking the recycle bin at the end of each record.

9. Save the new MYBPM migration group by clicking the **Save** button (the little disk) in the button bar in your window. The MYBPM migration group should now look like the group shown in Figure 8-18.
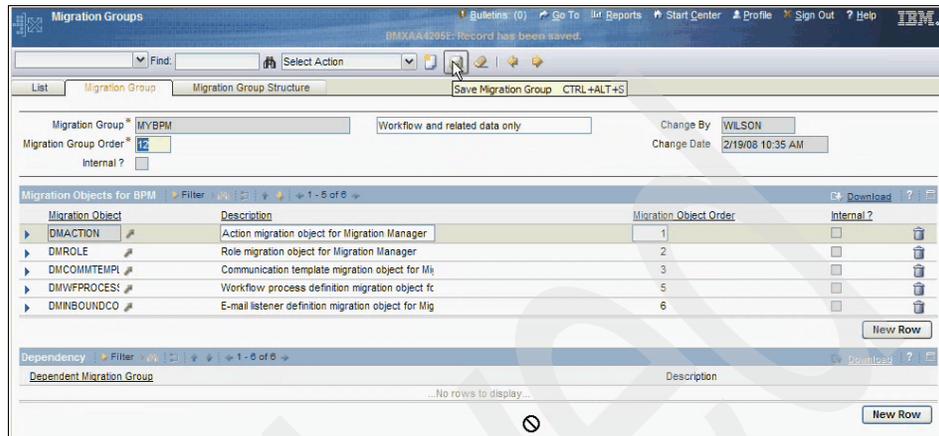
Figure 8-18   MYBPM migration group

We are now ready to create the package definition and start migration.

1. From the Go To menu, select **System Configuration** → **Migration** → **Migration Manager**, as shown in Figure 8-19.
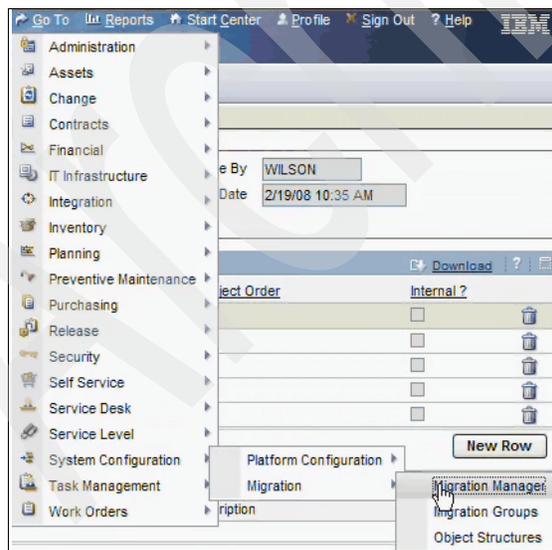
Figure 8-19   Migration menu - Migration Manager

2. Click the **New Record** button in the button bar on your window, as shown in Figure 8-20.

Figure 8-20   New package definition

A window is displays where you can create a new package definition, as shown in Figure 8-21.
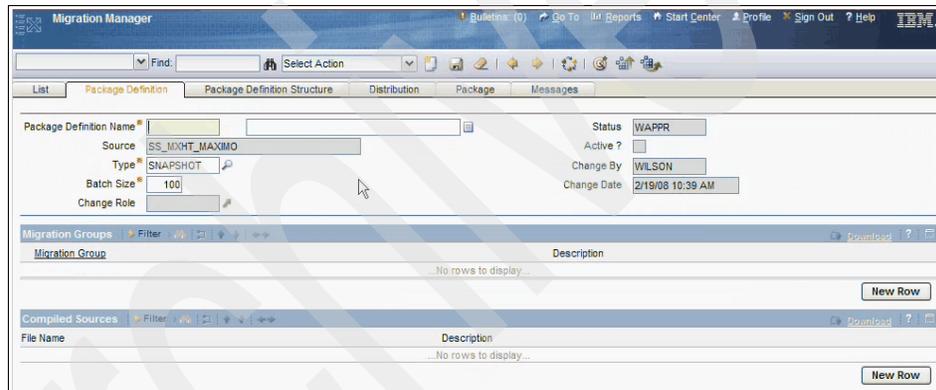
Figure 8-21   Package definition information

3. We name the package SRWF and describe it as SR priority workflow to indicate we want to migrate only the new SR priority workflow.

    The field Source is an identifier from the source environment, which ensures uniqueness of the package throughout the landscape. The source field is automatically generated and contains an identifier for the originating application server host name (in this example, SS), DB system identifier (SID) or instance name (in this example, MMHT), and schema name (in this example, MAXIMO), separated by underscores. Each package created in the source environment always contains the source identifier.

4. You can use the field Type to select the type of package, SNAPSHOT or CHANGE. In our example, we use a snapshot package because we want to migrate the entire workflow and related objects.

5. The field Batch is used to limit the load on your application server. You can set the number of XML records exported at one time, limiting memory usage. By default 100 records are exported at once, which we accept for our example.

6. In the section Migration Groups, you can select the groups representing the database content you want to migrate in the package. In our example, we select the MYBPM group.

7. In the section Compiled Sources, you can also select content outside the database, such as Java classes or properties files. In our example, we do not use any compiled sources for migration, so we leave this section empty.

8. After selecting the MYBPM migration group, we want to further specify which specific objects we want to migrate. We can do so clicking the **Where Clause** button, as shown in Figure 8-22.
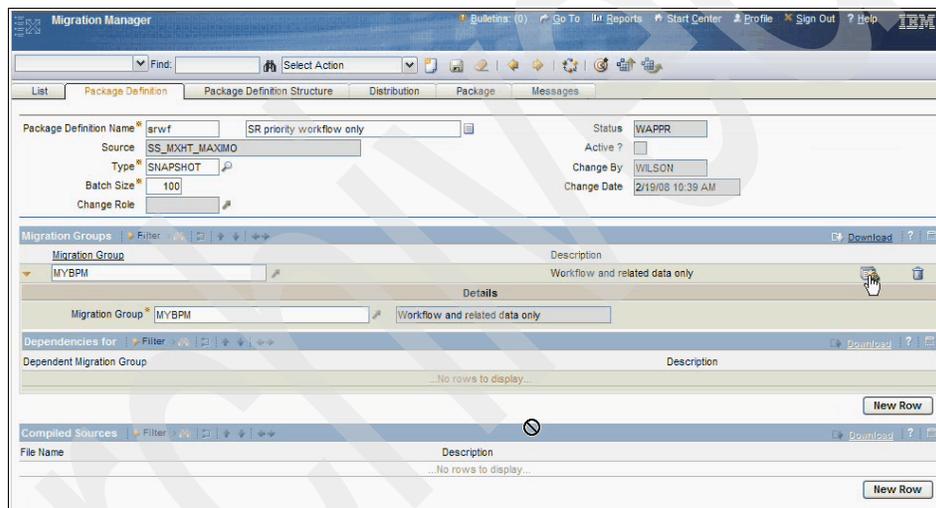
Figure 8-22   Package definition - Migration Group Where Clause

Figure 8-23 shows you the Set Where Clause dialog box, providing the migration objects and related MBOs you can use.
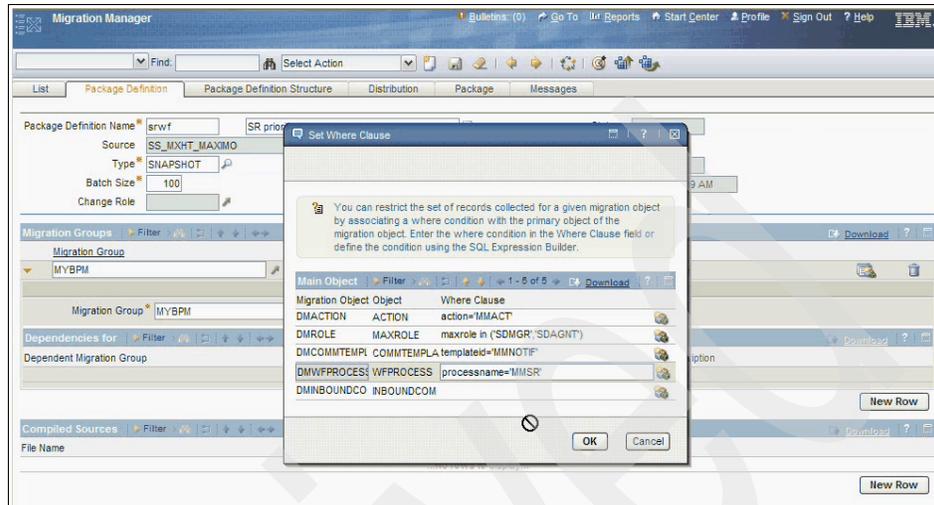
Figure 8-23   Package definition - Migration Groups Set Where Clause

We created a Where clause for the following migration objects, to make sure only the example workflow and related objects are migrated:

– DMACTION: The Where clause action='MMACT' ensures only the MMACT action, used in the workflow, is migrated as part of the package being created.

– DMROLE: The Where clause maxrole in ('SDMGR','SDAGNT') ensures only the roles used in the workflow are migrated as part of the package being created.

– DMCOMMTEMPLATE: The Where clause templateid='MMNOTIF' ensures only the MMNOTIF Communication template, used in the workflow, is migrated as part of the package being created.

– DMWFPROCESS: The Where clause processname='MMSR' ensures only the MMSR example workflow is migrated as part of the package being created.

To create a Where clause for any object, without thorough knowledge of SQL and database fields, you can use the SQL Expression Builder by clicking the rightmost button on the object line.

A SQL Expression Builder dialog box is displayed. In this dialog box, you can create the required SQL statement, as shown in Figure 8-24.
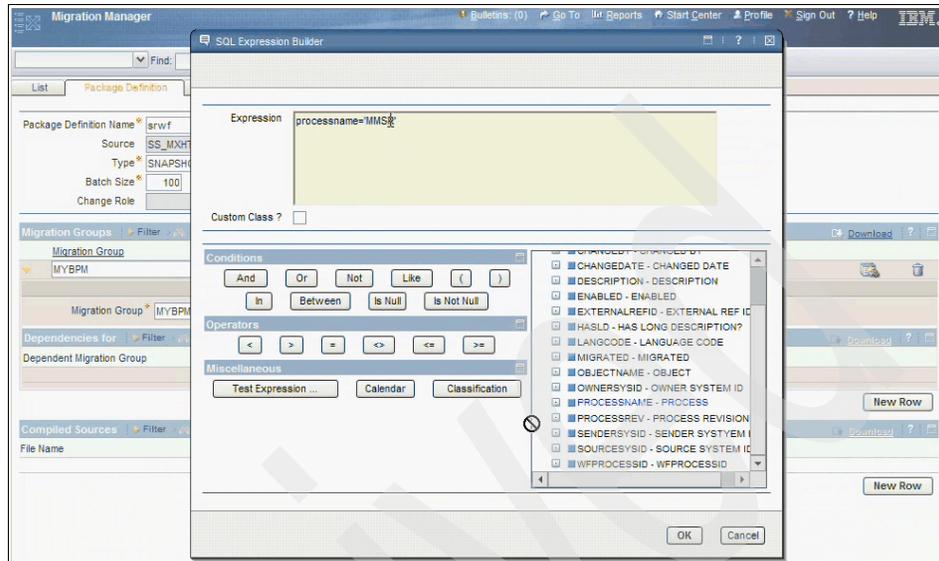
Figure 8-24   Package definition - Migration Group Where Clause Builder

You can select attributes for the specific object (MBO) and use conditions and operators to limit or specify the content. To ensure you have created the correct Where Clause, you can click the **Test Expression** button in the section Miscellaneous to verify the SQL statement.

Click the **OK** button on the SQL Expression Builder dialog box to use the SQL expression you created for the object. Repeat this step for each of the objects you want to further specify, filter, or manipulate.

Click the **OK** button in the Set Where Clause dialog box to use the Where Clauses created for all objects and continue.

9. Now save the new srwf package definition by clicking the **Save** button (the little disk) in the button bar in your window, as shown in Figure 8-25.
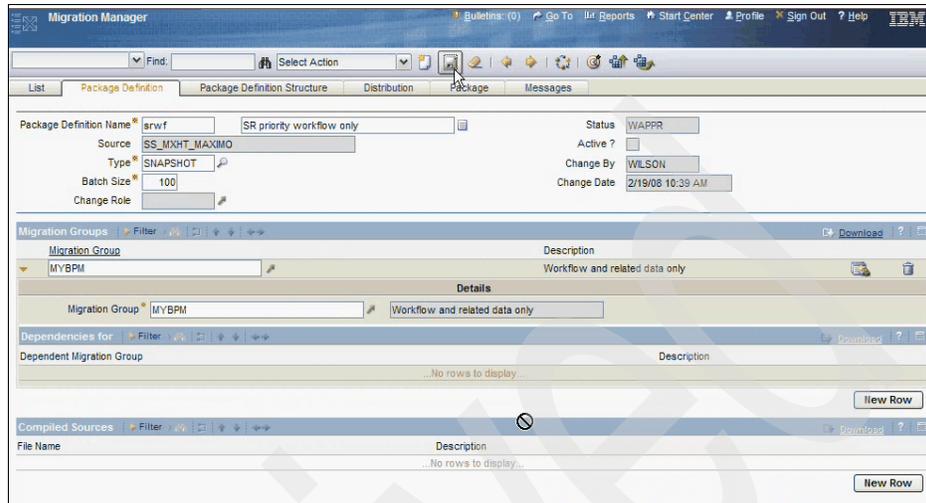
Figure 8-25   Save package definition

We now have to approve the package definition to continue. You can use this step to have a change manager, or other responsible role in the organization, review the package definition, approve the content, and continue.

10.Click the **Change Status** button in the button bar in your window, as shown in Figure 8-26. You can also use the Change Status action from the Select Action menu.
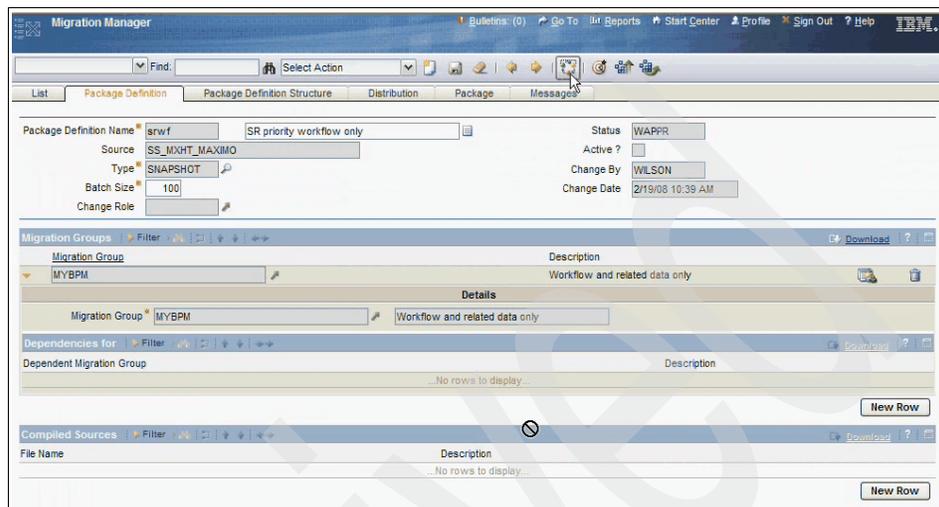
Figure 8-26   Change package definition status

A dialog box is displayed, as shown in Figure 8-27, where you can select the approved status.
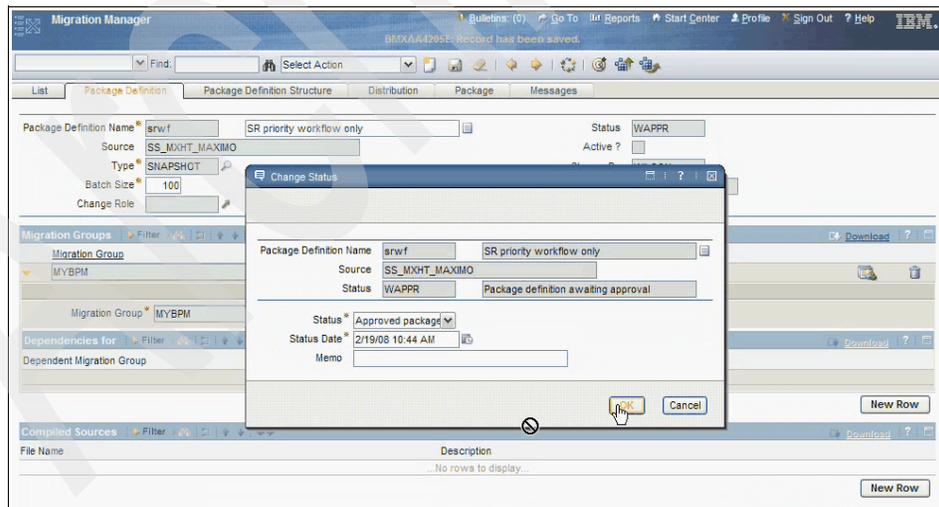
Figure 8-27   Package definition approved

11.Click the **OK** button to continue.

12. We now have to activate the package definition to be able to create a package for distribution. From the Select Action menu select **Activate/Deactivate Package Definition** to activate the package definition (see Figure 8-28).
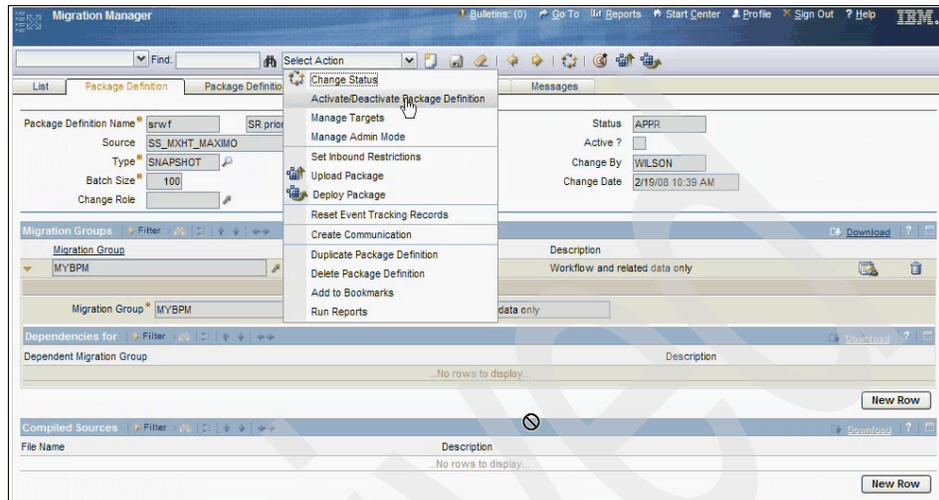
Figure 8-28   Activate package definition

A message in the menu bar states "Package Definition srwf has been activated." The Active check box in the window now contains a check mark.

13. You can view the package definition in a tree structure, using the Package Definition Structure tab. This view shows all package definition information in a parent-child relationship, as you can see in Figure 8-29.
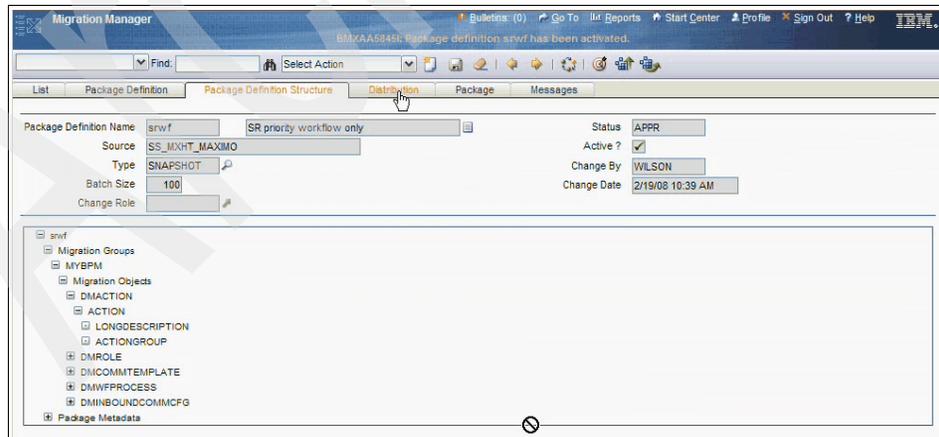
Figure 8-29   Package definition structure

14. The last step in package definition is defining all possible targets for the package. Click the **Manage Targets** button in the button bar in the window, as shown in Figure 8-30. You can also select the **Manage Targets** action from the Select Action menu.
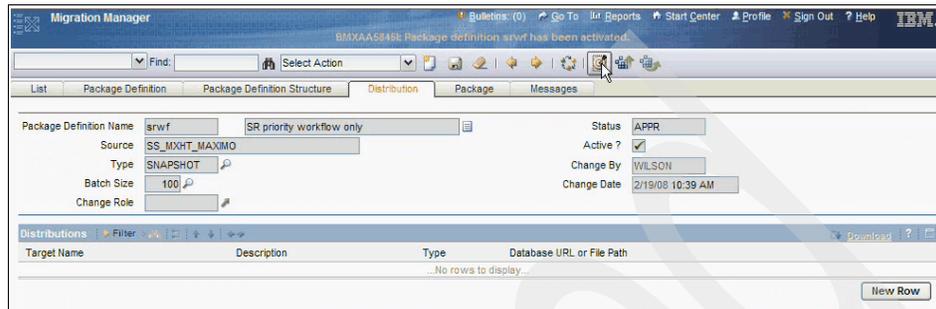
Figure 8-30   Manage package definition targets

A dialog box is displayed, as shown in Figure 8-31, where you can define all possible targets and distribution methods for these targets.
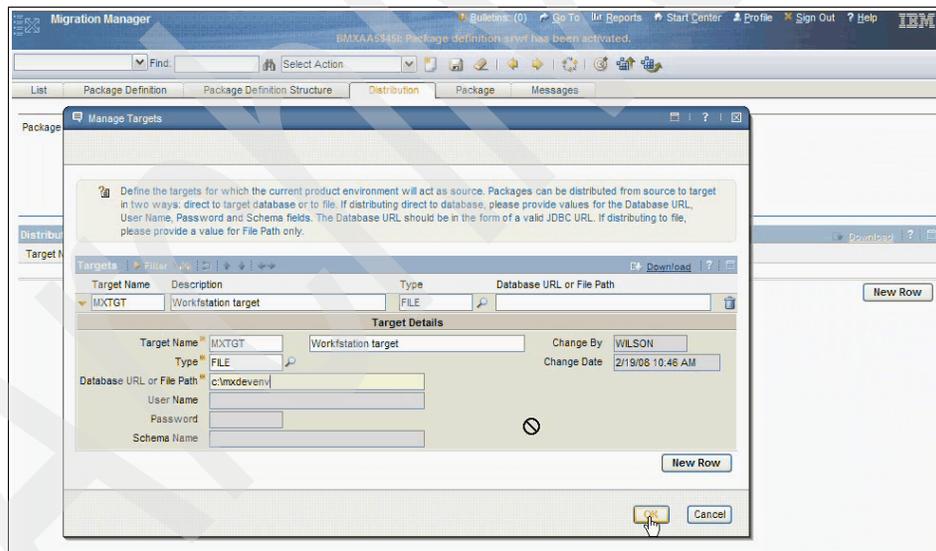
Figure 8-31   Possible package definition targets

In our example we use the MXTGT target system, using a file distribution. We store the physical package (package file) on the local operating system in directory c:\mxdevenv.

15. Push the **OK** button to continue.

16.We can now select the distribution target by clicking the **New Row** button in the Distributions section (see Figure 8-32).
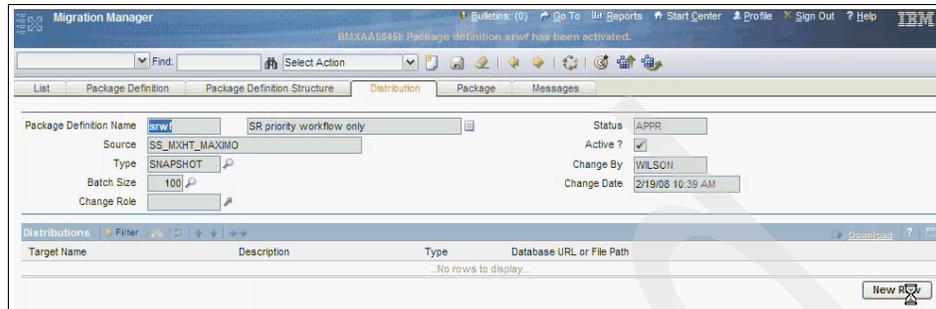
Figure 8-32   Select distribution target

17.Choose the MXTGT Target Name and save the package definition by pushing the **Save** button (the little disk) in the button bar in the window, as shown in Figure 8-33.
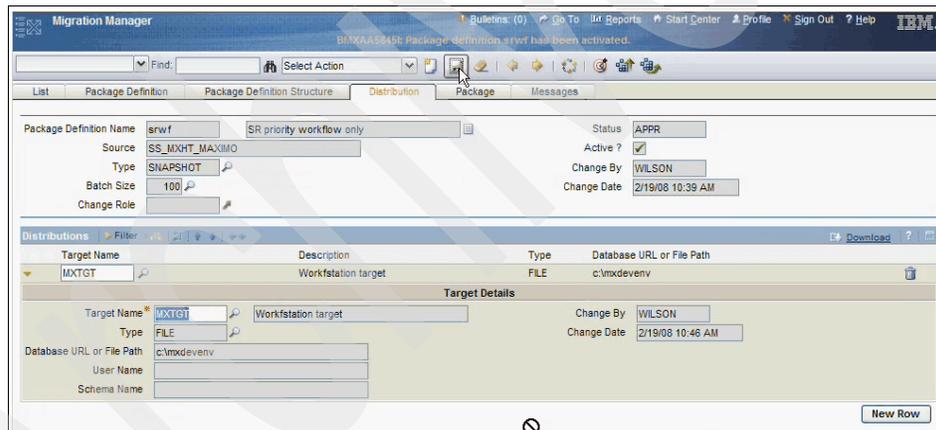
Figure 8-33   Save package distribution target

Package definition is now complete.

8.3.3 Package creation

At this point, we create a package to actually distribute and deploy. We remain in the Migration Manager application of the source environment.

1. Select the **Package** tab and click the **Create** button in the Packages section, as shown in Figure 8-34.
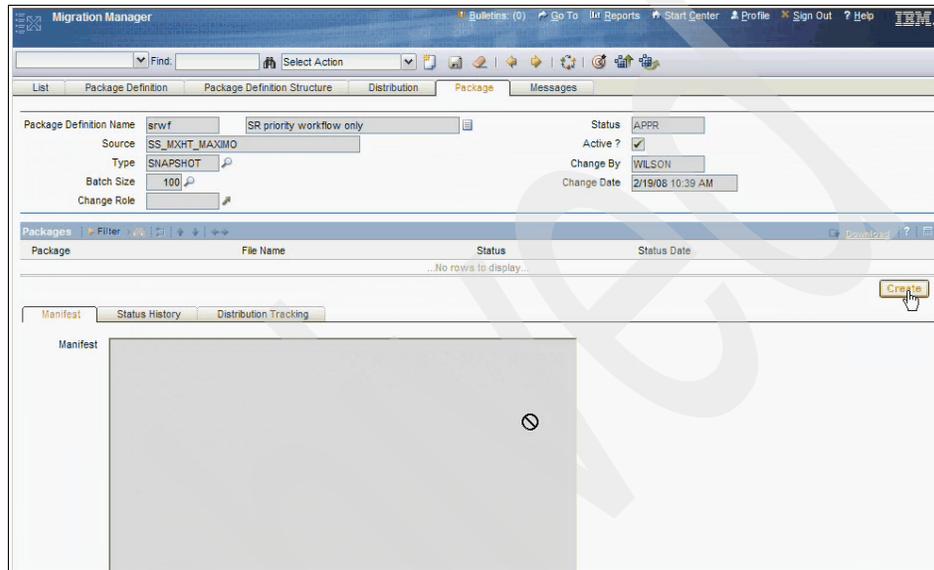
Figure 8-34   Create package

2. We first enter general information about the package in the dialog box, as shown in Figure 8-35.
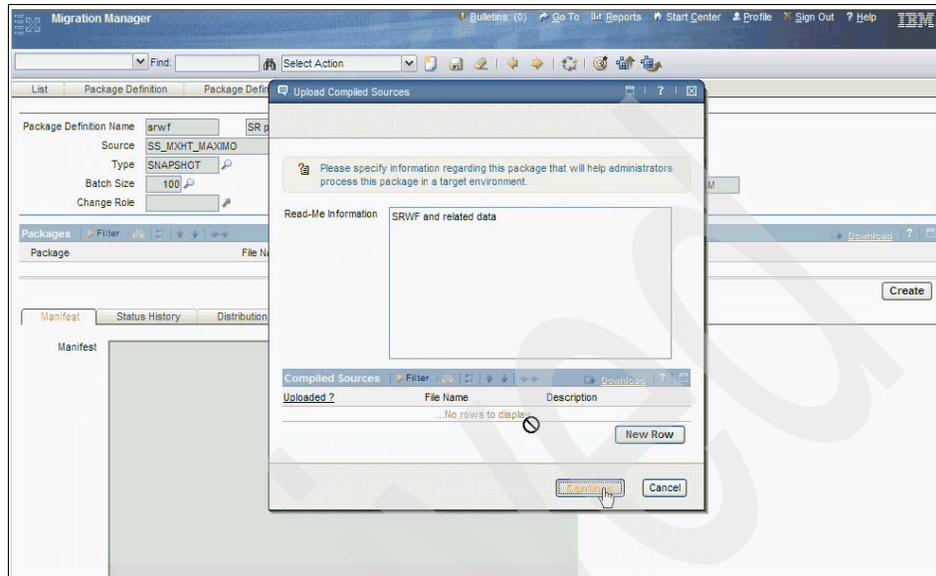
Figure 8-35   Package creation information

This information is part of the package manifest and is migrated to be used by a deployment manager to identify useful information about the package, before actually deploying the package.

In this step, we also have the option to select compiled sources (content outside of the database, which usually is a ZIP file containing the code and other files), which we skip in this example.

3. Click the **Continue** button in the dialog box to start package creation. A new dialog box is displayed, presenting the creation progress, as seen in Figure 8-36.
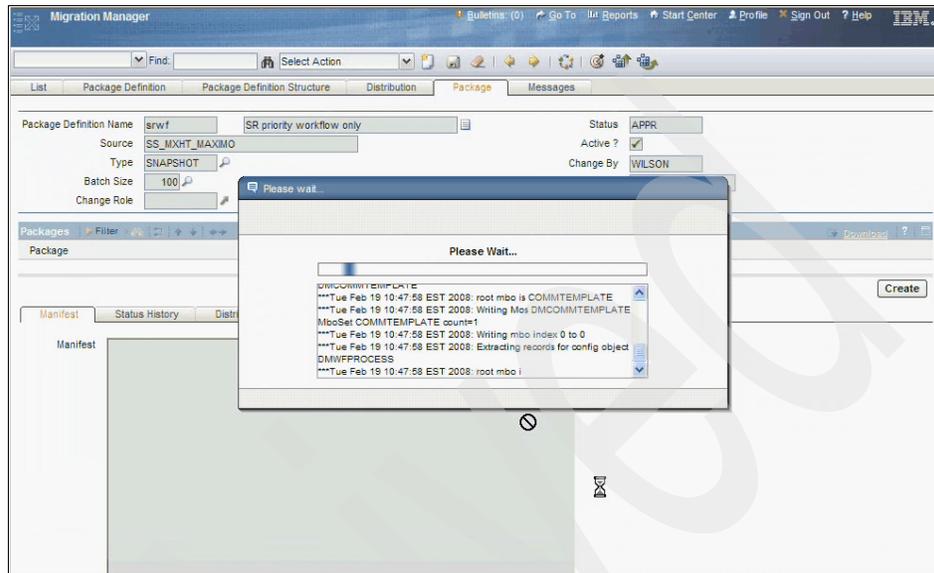
Figure 8-36   Package creation in progress

When creation is complete, a new package is visible in the Package section, as shown in Figure 8-37.
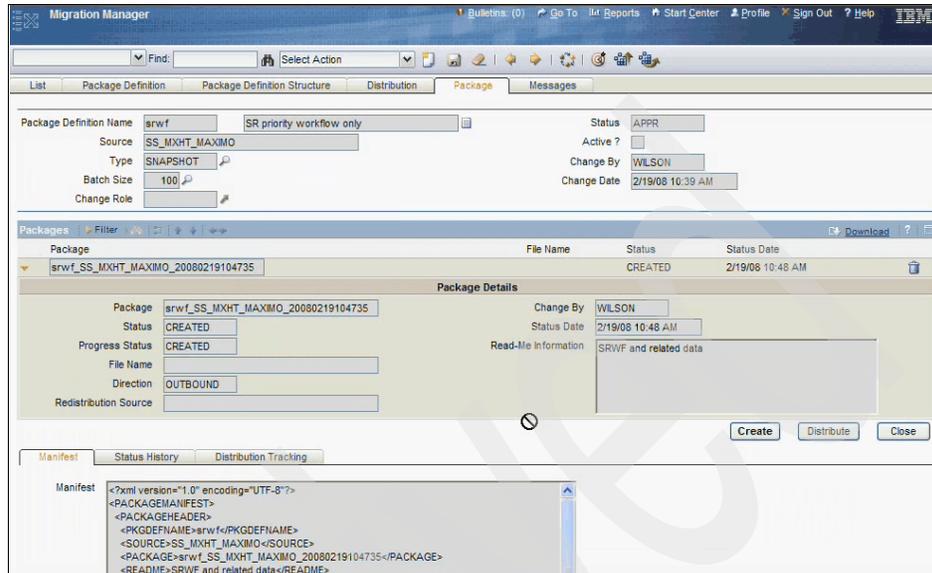
Figure 8-37   Package creation complete

Each package has a unique name, which in this example is srwf_SS_MXHT_MAXIMO_20080219104735. The name is built using the package definition name, the source identifier, and a time stamp of creation.

The status field provides the status of each package, which can also depend on the direction. In our example, the status is CREATED, meaning a package is created in the source database.

The direction of the package specifies whether the package is exported into the system you are logged on to (OUTBOUND) or imported in the system you are logged on to (INBOUND). Thus the package is specified as OUTBOUND in our example.

The Manifest tab shows general package information in XML format, such as the readme we entered earlier, source system version information, and package content.

At this point, the package is created, but it has only been put in a staging table of the database on our source environment. We have to distribute the package to actually have a physical file to import in the target environment.

8.3.4 Package distribution

At this point, we distribute the package by following these steps:

1. Click the **Distribute** button to start package distribution, as shown in Figure 8-38.
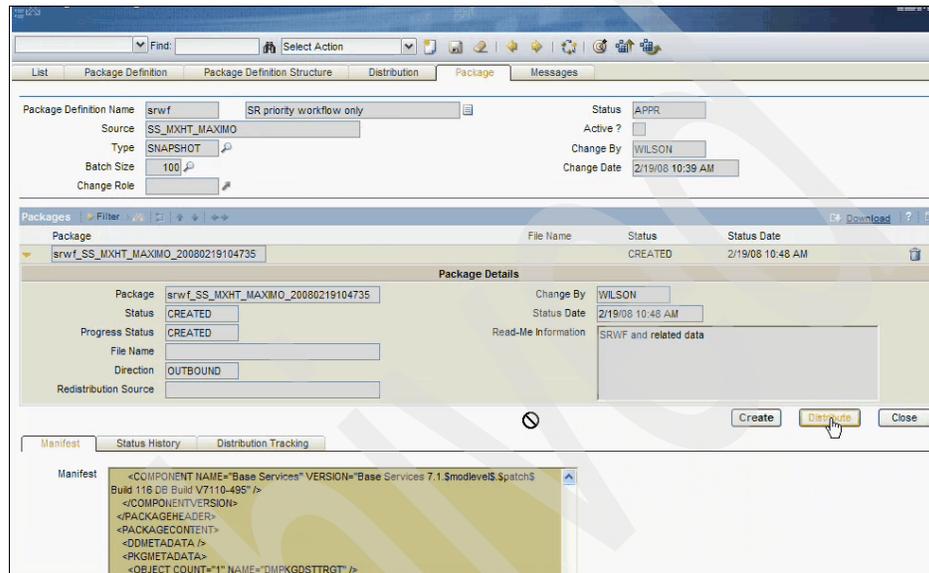
Figure 8-38   Package distribution

2. Select a target for distribution by clicking the appropriate check boxes, as shown in Figure 8-39.
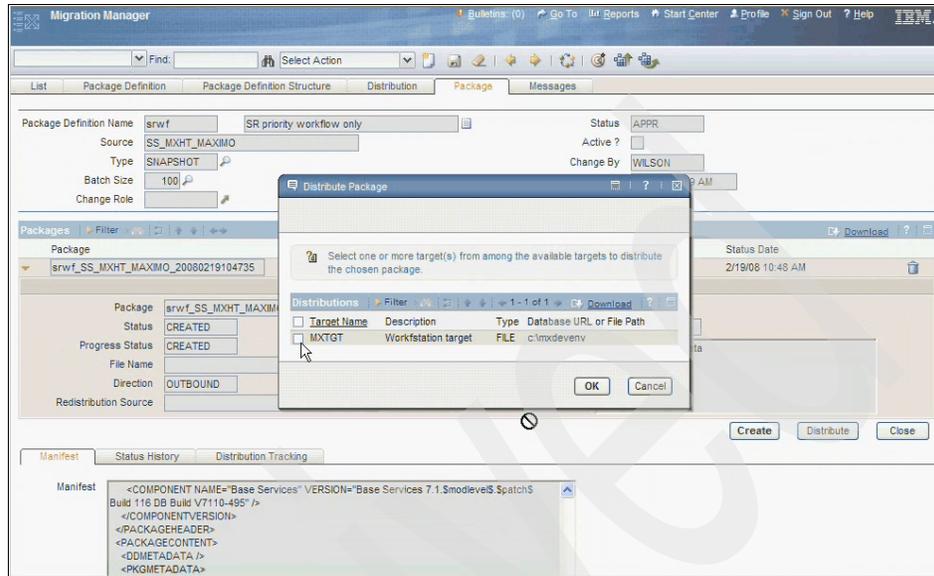
Figure 8-39   Package distribution targets

During package definition, we created only one possible target, exporting the package as a file to a directory on the source application server. You can have more targets to select from, each of a different type and with different export location, but we need only the MXTGT target for the example.

3. Click the **OK** button to continue. A dialog box is displayed that indicates distribution progress, as shown in Figure 8-40.
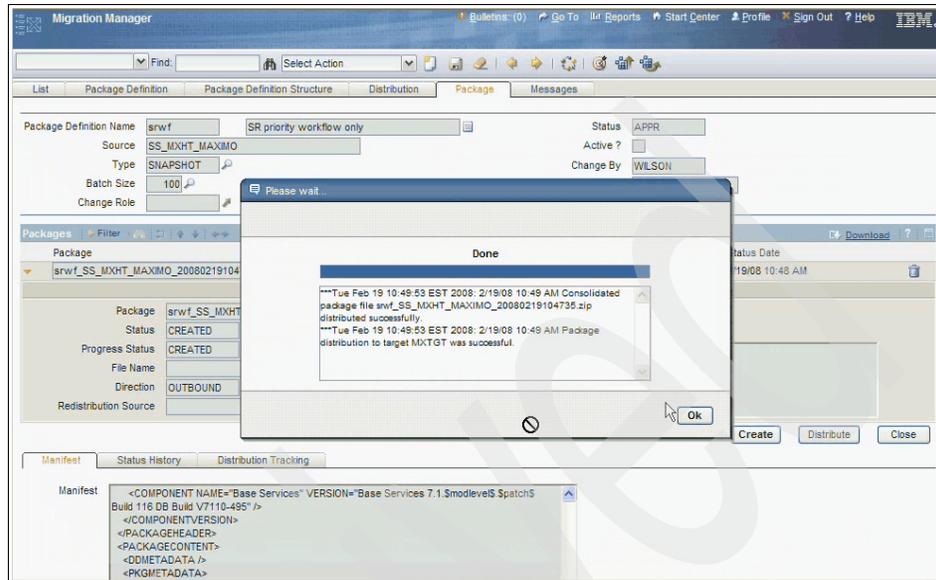
Figure 8-40 Package distribution in progress

To make sure the physical package is indeed created and stored in the correct location, we can verify whether the file is actually in the directory c:\mxdevenv on the source system. We can also download the file to use on the system we currently work on (it can be any system with a browser connection to the source environment), which we do. We download the file to verify the physical file actually exists, but doing so also provides the means to store the file on a shared location for import.

4. Click the **Download** button on the right of the package line, as shown in Figure 8-41.
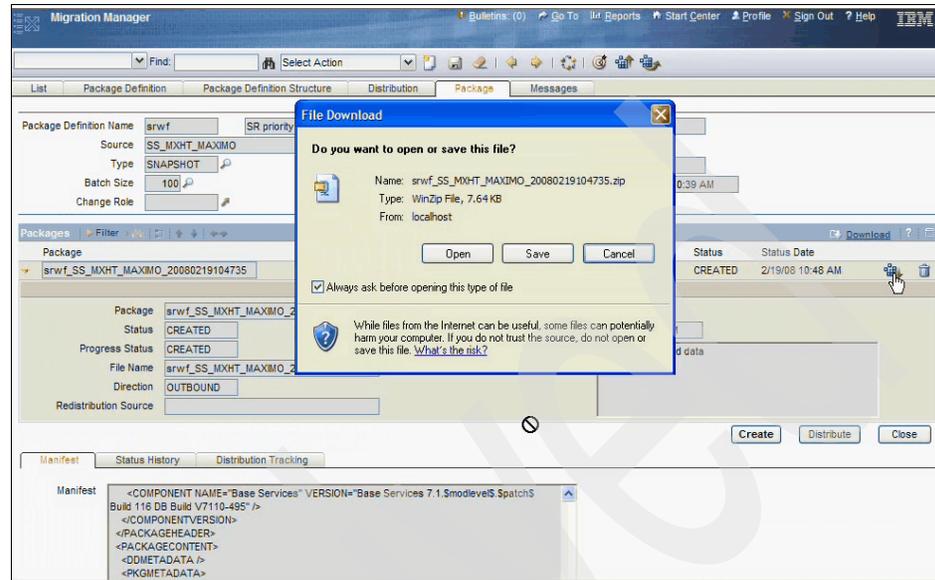
Figure 8-41   Package file download

5. A Save As dialog box is displayed, where you can select to open or save the file, or cancel the download. As you can see in Figure 8-42, the file name is the package name with a ZIP extension, which is also shown in the field File Name in the Package section of the window. We save the file on a shared file server.
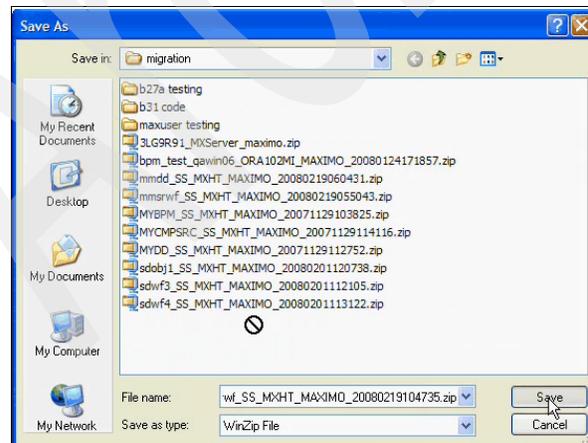
Figure 8-42   Package file download location

6. When the file download is completed, click the **Close** button to continue.

The messages tab of the Migration Manager application displays informational or error messages for each of the packages created, as you can see in Figure 8-43.
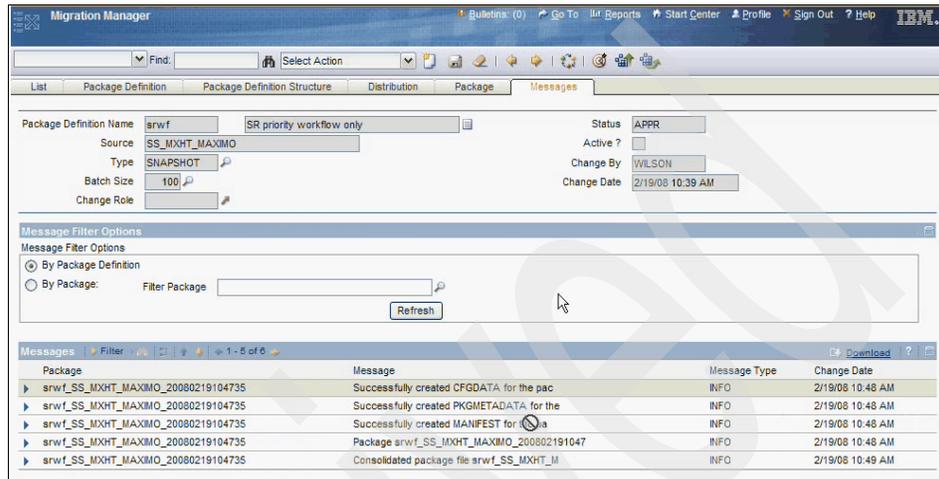
Figure 8-43   Package messages

If you experience any problems during the package creation process of Package Distribution, you can start troubleshooting at this point. Each message provides information about a specific step in the process for each package, including a time stamp for the event.

8.3.5 Package deployment

At this point you are ready to deploy the package.

1. Go to the target Tivoli Service Request Manager system and log on. From the Go To menu, select **System Configuration** → **Migration** → **Migration Manager**.

2. Click the **Upload Package** button, as shown in Figure 8-44.
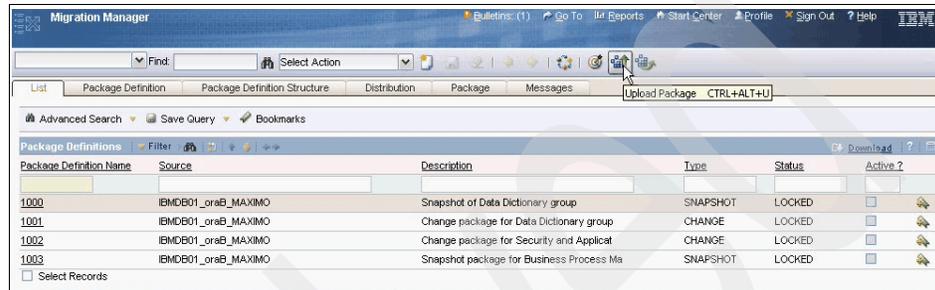
Figure 8-44   Upload package

3. In the dialog box that is displayed, select the file to upload from the location used to download the package, as shown in Figure 8-45.
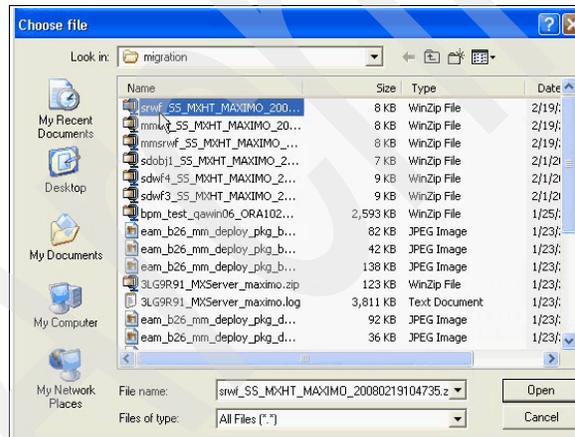
Figure 8-45   Select file to upload

4. In our example, we select the srwf_SS_MXHT_MAXIMO_20080219104735.zip file. Click the **Open** button on the dialog box to start the import.

   You should now see a message, as shown by the red notification on the menu bar in Figure 8-46, package srwf_SS_MXHT_MAXIMO_20080219104735.zip has been successfully uploaded.
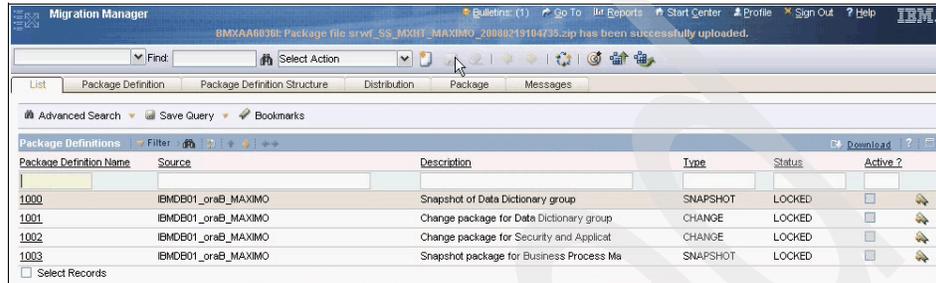
Figure 8-46   Package uploaded successfully

   At this point the package we distributed has been stored in the staging table of the target environment. We have to deploy the package to complete migration and be able to use the workflow.

5. Click the **Deploy Package** button, as shown in Figure 8-47.
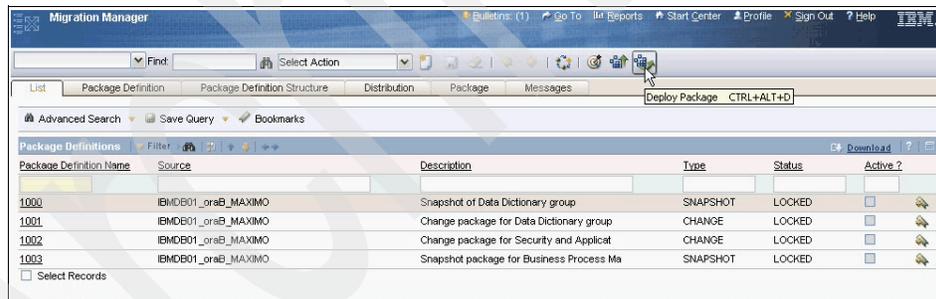
Figure 8-47   Deploy package

A dialog box is displayed, listing the packages available for deployment. You can select the package. In our example, we select the package srwf_SS_MXHT_MAXIMO_20080219104735 (see Figure 8-48).
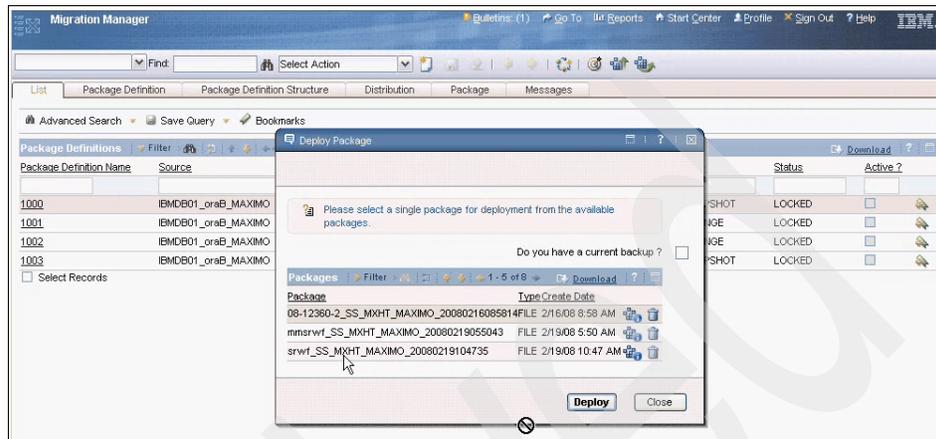
Figure 8-48   Select package to deploy

6. To view general information about the package prior to deployment, you can click the **Package Information** button, as shown in Figure 8-49.
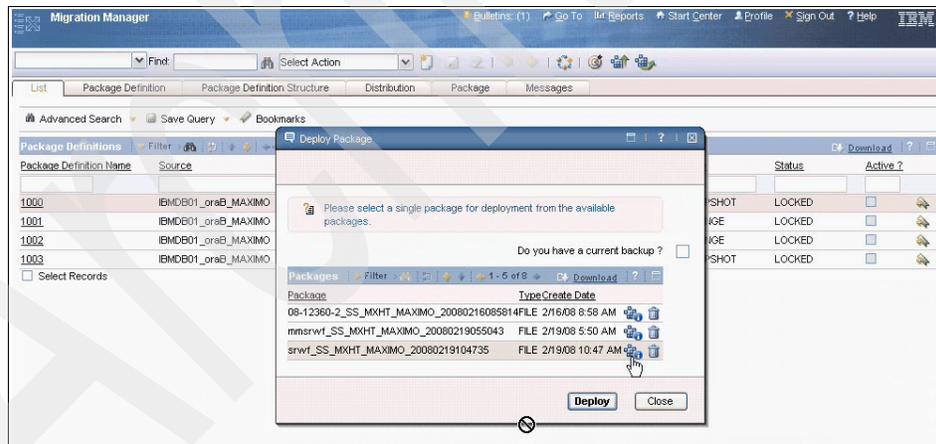
Figure 8-49   View selected package manifest

The package manifest is displayed, providing information about the content and source environment (Figure 8-50).
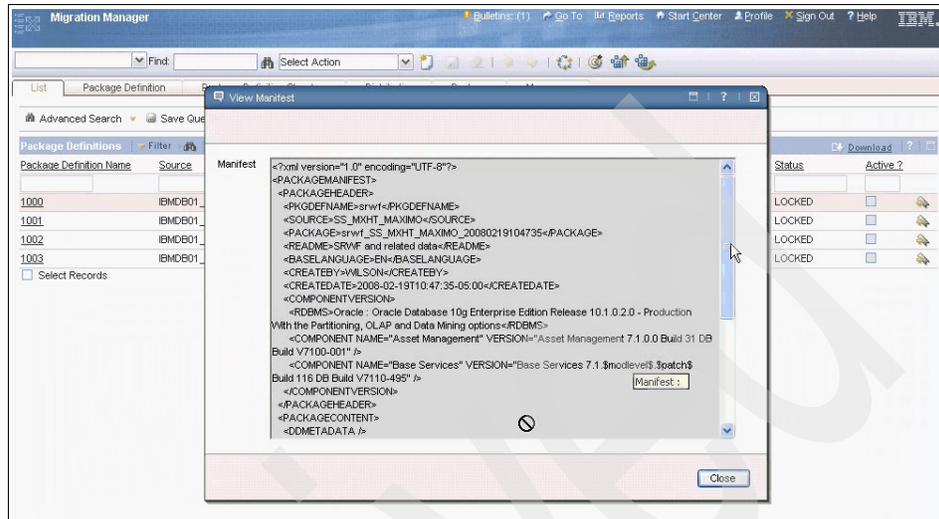
Figure 8-50   Package manifest

7. Click the **Close** button on the View Manifest dialog box to continue.

8. For safety and regulation compliancy, a check box is provided in the Deploy Package dialog box, prompting you to verify and register whether a recent backup has been made (see Figure 8-51 on page 230). It is important to have a recent (and possibly validated) backup of the entire environment. At the least, a backup of all database content is required, but if you also deploy compiled sources a file backup of the target application server is also necessary.
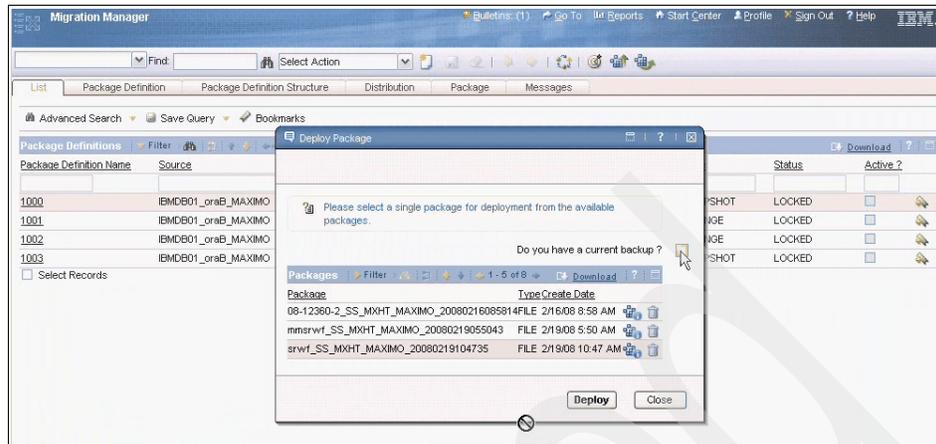
Figure 8-51   Predeployment backup

We do not expect the Package Deployment to fail, but in case problems occur, you still have a valid backup to restore, and you can continue working on the target system. The answer to the question you provide in the check box is stored in the database when deployment is started.

9. Click the **Deploy** button to start package deployment.

10. A new dialog box is displayed, prompting you for electronic signature authentication (see Figure 8-52). To continue, you have to provide the password for the user specified.
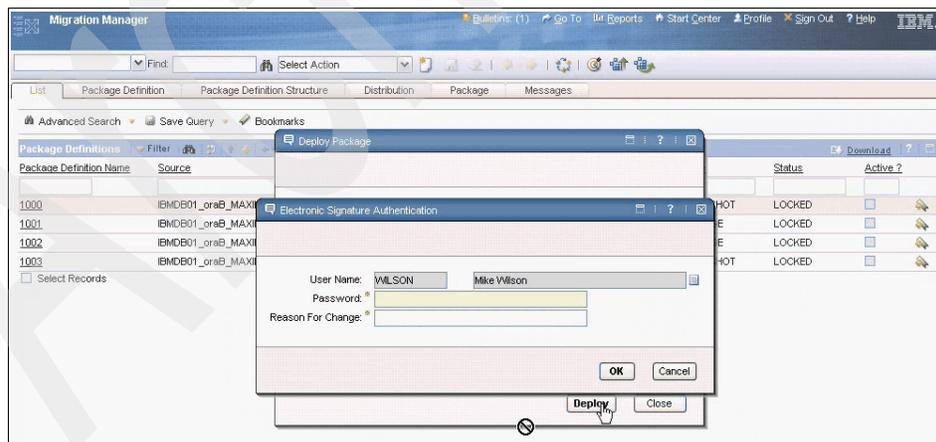
Figure 8-52   Package deployment authentication

11. Click the **OK** button to continue. At this point, the authentication is stored in the database.

12. A new dialog box is displayed, showing the progress of package deployment. If deployment is successful, the result is displayed as in Figure 8-53.
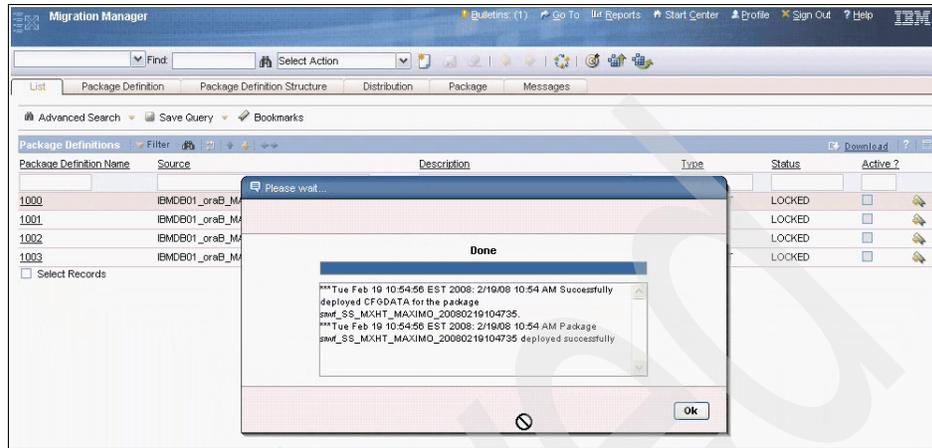
Figure 8-53   Package deployment successful

13.Click the **OK** button to continue.

In the Package tab of the Migration Manager application, you can find the status of the deployed package. In the Packages section, you view the package and status DEPLOYED (as shown in Figure 8-54). If you click the arrow to the left of the package name, more details are provided. The Direction field displays INBOUND because the package was uploaded and deployed in this environment.
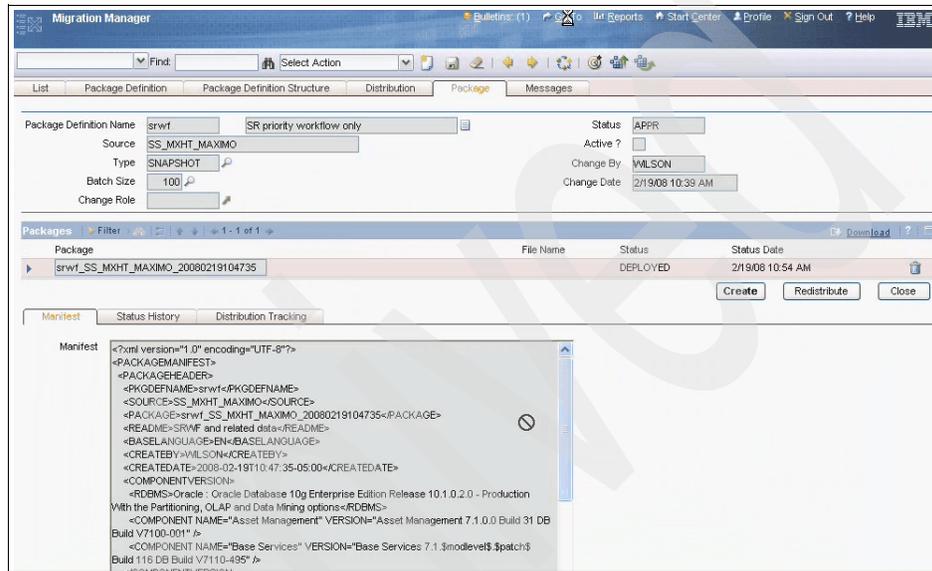
Figure 8-54   Package deployment status

14. At this point, we can verify whether the workflow and related objects were migrated successfully. We do so by validating the workflow. From the Go To menu, select **System Configuration** → **Platform Configuration** → **Workflow Designer**, as shown in Figure 8-55.
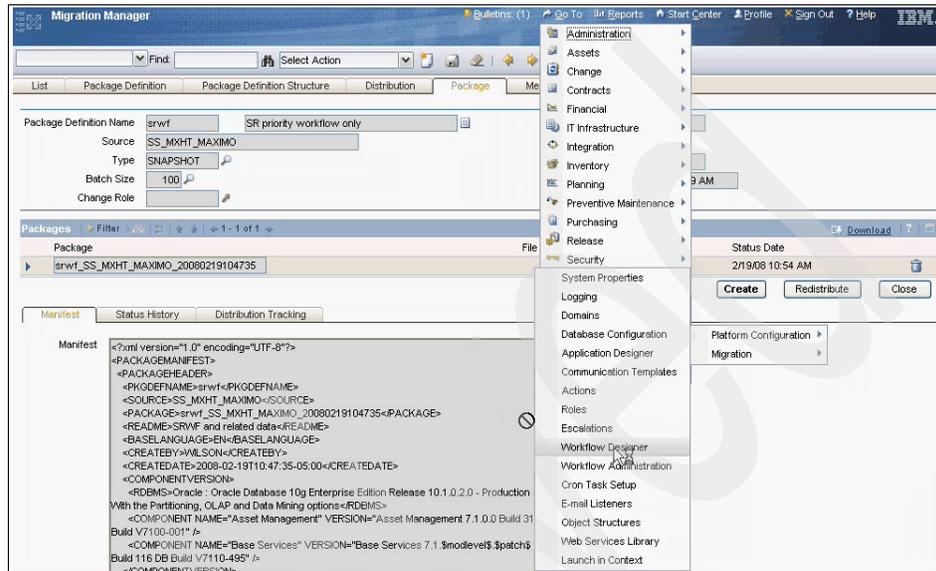
Figure 8-55   Workflow Designer

15. In the List tab we search for and select the MMSR workflow.

16. From the Select Action menu, choose **Validate Process**. If the validation returns successfully, as shown by the red notification on the menu bar in Figure 8-56, the MMSR workflow and all related objects we selected (roles, communication template, and action) are migrated successfully.
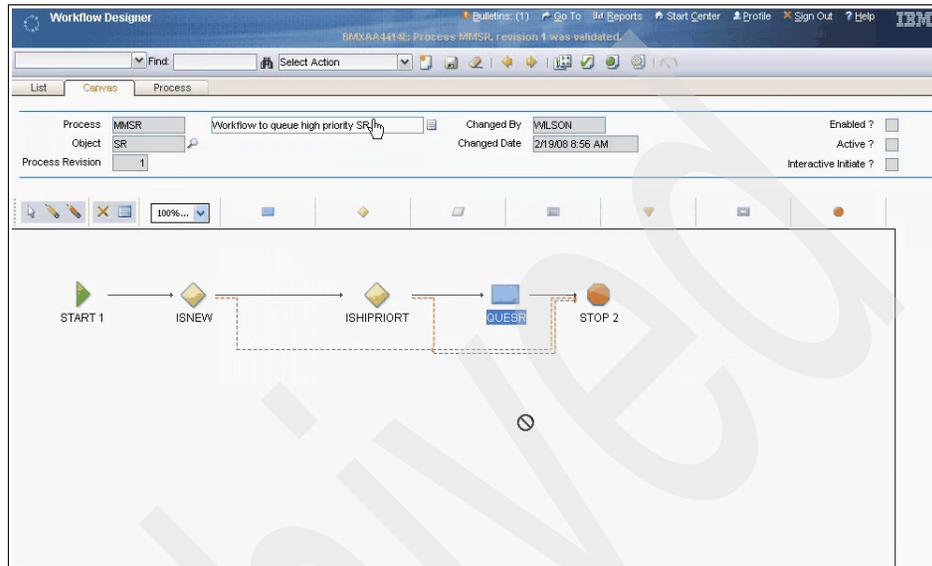
Figure 8-56   Object validation

We have successfully migrated configuration data from a source system to a target system, using a safe, automated, traceable, and reusable process.

8.4 Integration Framework scenario

This section covers data exchange between Maximo Asset Management and an external system using flat files, XML, and Integration tables.The topics discussed in this section are as follows:

► Asynchronous data exporting from Maximo to an external system using XML and flat file.

► Asynchronous data loading from external system to Maximo using Interface tables.

8.4.1 Basic configuration checklist prerequisite

Before initiating data import, complete the basic configuration tasks as outlined in 7.4, "Basic Maximo integration configuration checklist" on page 170.

8.4.2 Asynchronous data export using XML and flat file

This section describes asynchronous data export from Maximo Asset Management to an external system using XML and flat file. In describing this process, we use the MXITEM data object. The data export feature is exclusively available for the data synchronization (action = SYNC) interface for internal type adapters and interfaces based on non-merged integration objects.

The outbound process involves the following tasks:

1. Perform basic configuration.
2. Create the MXITEM2 object structure.
3. Create the MXITEMInte2 Publish Channel associate with the MXITEM2 object structure.
4. Create the EXTSYS2 external system.
5. Export data (XML and flat file).
6. Track messages.

Create object structure

In this example, we create MXITEM2, a duplicate of the predefined Maximo Asset Management MXITEM object structure.

1. Choose **Go To** → **Integration** → **Object Structures**.

2. Select the **Object Structures** tab.

3. Search for the MXITEM integration object (see Figure 8-57).
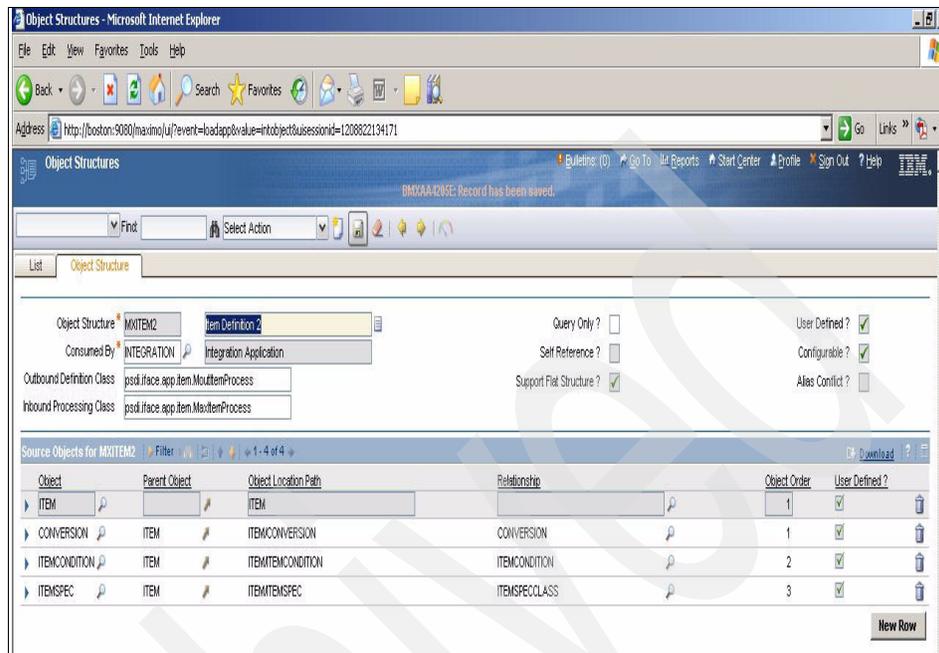
Figure 8-57   Object Structures

4. Duplicate the object structure. Click **Select Action** and then select **Duplicate Object Structure**.

5. Rename the new object structure (for example, MXITEM2).

> **Note:** Interface tables require all columns included in the corresponding object have an alias name of 18 or fewer characters.

6. Enable the **Support Flat Structure** check box.

7. To view conflicts that may exist due to duplicate fields, click **Select Action** and select **Add/Modify Alias** (see Figure 8-58).
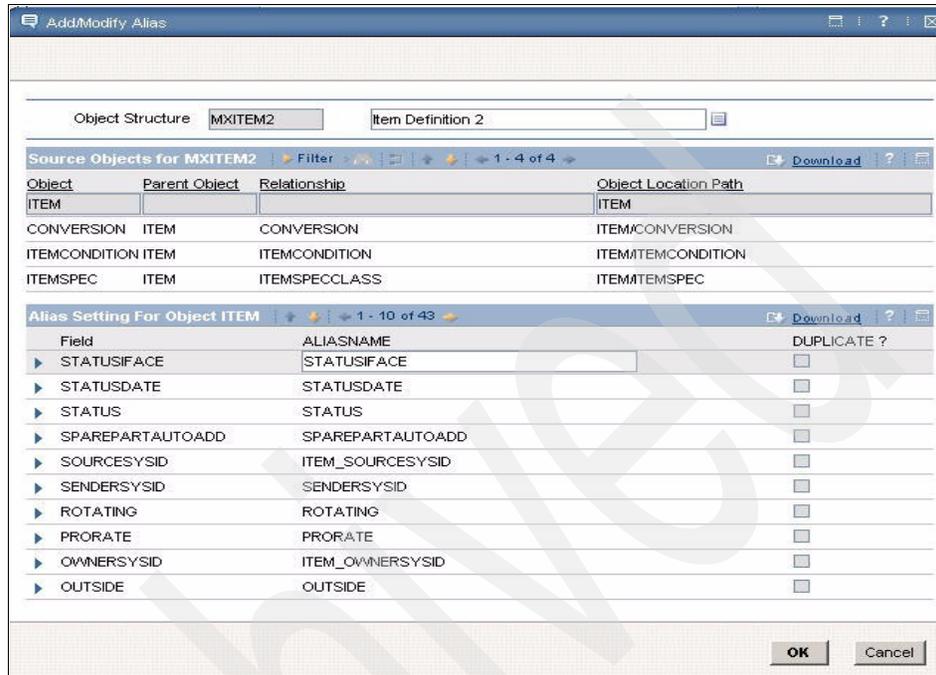
Figure 8-58   Object Structure- Add/Modify Alias

If a duplicate name exists, modify the alias value for the duplicate columns.

8. Persistent fields are always included and non-persistent fields are always excluded. To make the necessary changes, click **Select Action** and select **Exclude/Include Fields**, and make changes in the **Persistent Fields** tab or the **Non-Persistent Field** tab (see Figure 8-59).
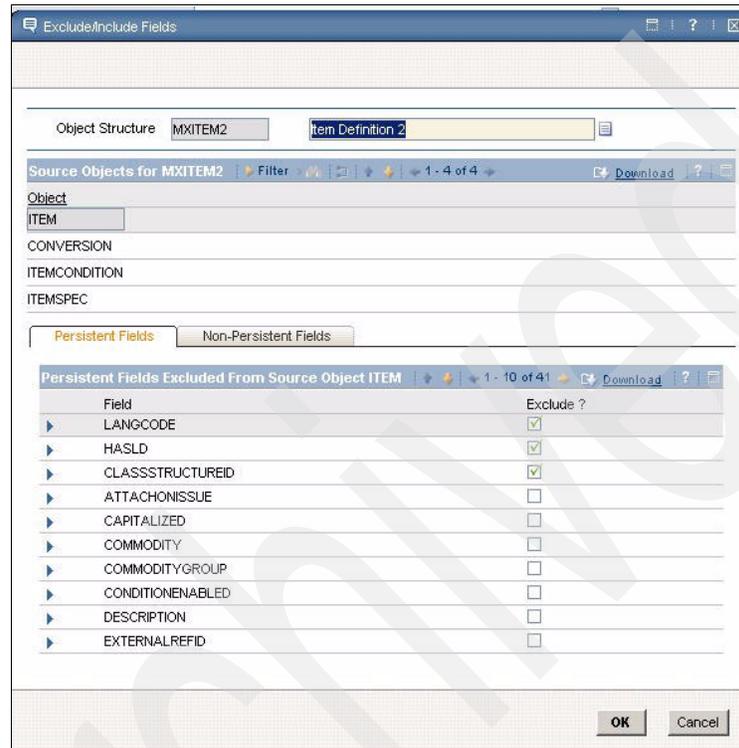
Figure 8-59 Object Structure - Persistent/non-persistent fields

9. Save the changes.

Create Publish Channels

Publish Channels is used to create and manage asynchronous outbound data flow from the Integration Framework to an external system.

In this section, we use the Maximo Asset Management pre-packaged MXITEMInte Publish Channel interface to associate with the MXITEM2 object structure example. Follow these steps:

1. Create Publishing Channel; choose **Go To** → **Integration** → **Publish Channels**.

2. Select the **Publish Channel** tab.

3. Enter the Publish Channel name (for example, MXITEMInte2).

4. Enter a brief description (for example, Item master 2).

5. Search for the object structure (for example, MXITEM2) associated with this channel in the Object Structure field. Choose **Select Value** from the Detail menu and populate the field.

6. Verify that Operation is set to Publish (see Figure 8-60).

Figure 8-60   Publish Channel creation

7. Enable Event Listener for the new interface by clicking **Select Action** and selecting **Enable Event Listener**.

8. Save your changes.

Create external system

In this section, we define a new external system, EXTSYS2, based on a duplicate of the predefined Maximo Asset Management EXTSYS1 interface. Follow these steps:

1. Choose **Go To** → **Integration** → **External Systems**.

2. Search for the EXTSYS1 system.

3. Select the **System** tab.

4. Duplicate the external system by clicking **Select Action** and select **Duplicate External System**.

5. Rename the new external system (for example, EXTSYS2).

6. Enable the external system by clicking the **Enabled** check box.

7. Specify the endpoint handler and click **OK** (see Figure 8-61).
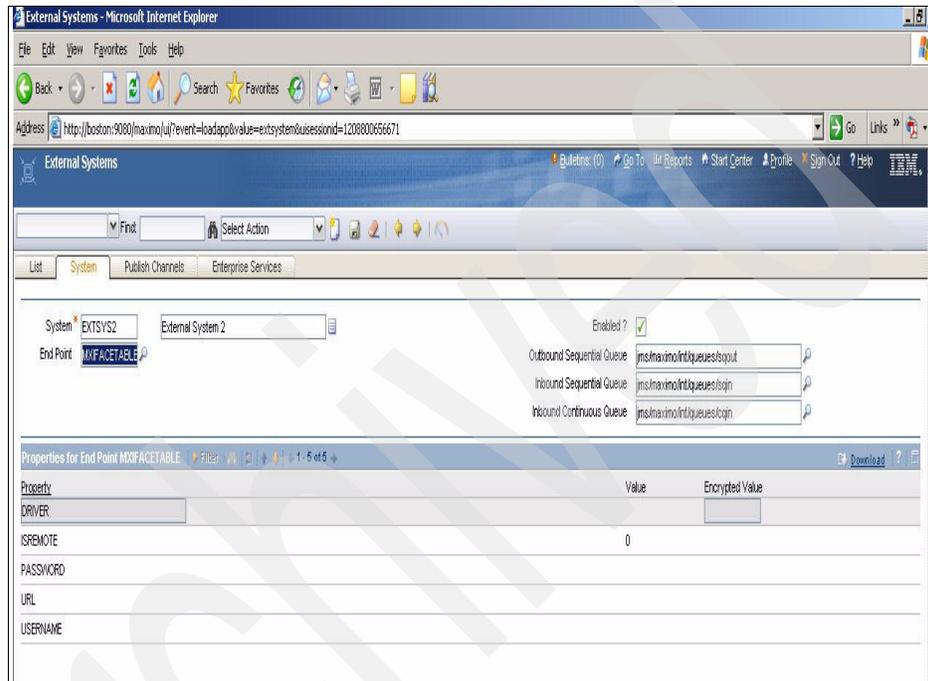
Figure 8-61   External systems creation

The options for adding or modifying an endpoint are listed in Table 8-1.

Table 8-1   Add or modify endpoint

Endpoint	Handler	Description
MXXMLFILE	XMLFILE	XML file endpoint
MXIFACETAB	IFACETABLE	Interface table endpoint
MXFLATFILE	FLATFILE	Flat file endpoint
MXCMDLINE	CMDLINE	Command-line endpoint
TADDMEP	TADDMAS	TADDM authorization synchronization endpoint

8. Verify the queue settings (see Table 8-2).

Table 8-2   Queue settings

Queue type	Settings
Outbound Sequential Queue	jms/maximo/int/queues/sqout
Inbound Sequential Queue	jms/maximo/int/queues/sqin
Inbound Continuous Queue	jms/maximo/int/queues/cqin

9. Select the **Publish Channels** tab (see Figure 8-62.

   a. Press the **Select Channel** button.
   b. Associate the Publish Channel interface with the new External System.
   c. Save your changes.
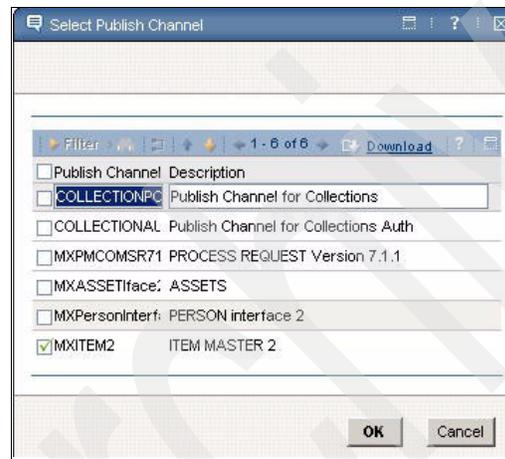
Figure 8-62   Mapping Publish Channels to external systems

10. Verify that data exists in the item object. If the database is empty, enter several values for testing purposes. Then, click the **Data Export** button to export data.

In the Data Export dialog box, enter an export condition if the filtering of exported records is required; otherwise leave bank and click **OK** (see Figure 8-63).

Figure 8-63   External Systems - Data Export Condition

If you chose flat file or XML in step 7 on page 240, the system generates a flat file or an XML file, respectively, in the global directory. The file generated from the export data is available in the %globaldirectory%/xmlfiles directory (see Figure 8-64).
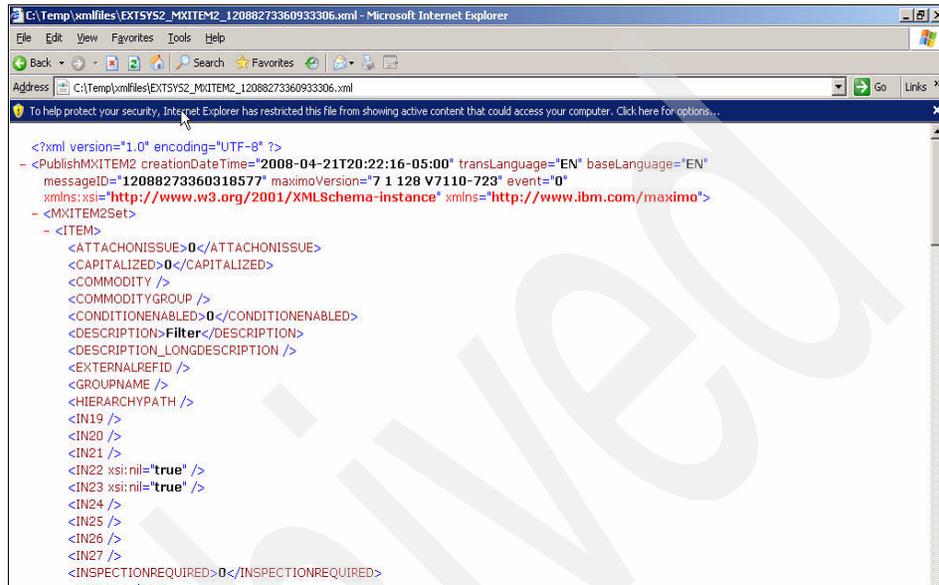
Figure 8-64   Sample XML data export

11. Verify message tracking for error messages.

    a. Select **Go To** → **Integration** → **Message Tracking** and click **enter**.

    b. Select the message from the Message Tracking application to view the message content (see Figure 8-65).
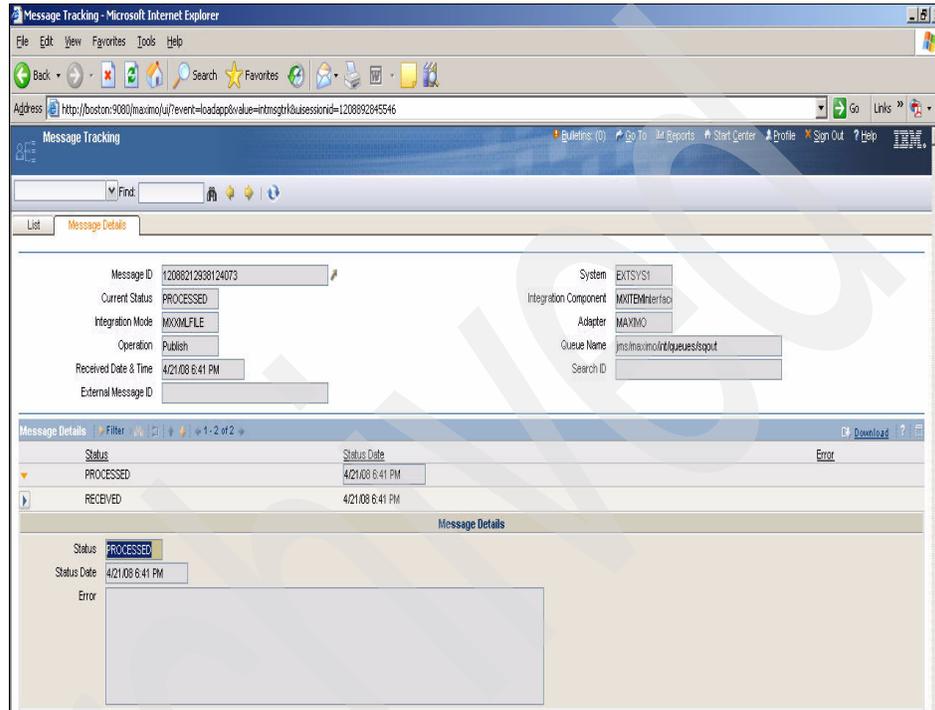
Figure 8-65   Message Tracking application - sample detail message view

12.If you encounter an error, use the Message Reprocessing application to correct and reprocess the message. Select **Go To** → **Integration** → **Message Reprocessing** (see Figure 8-66).
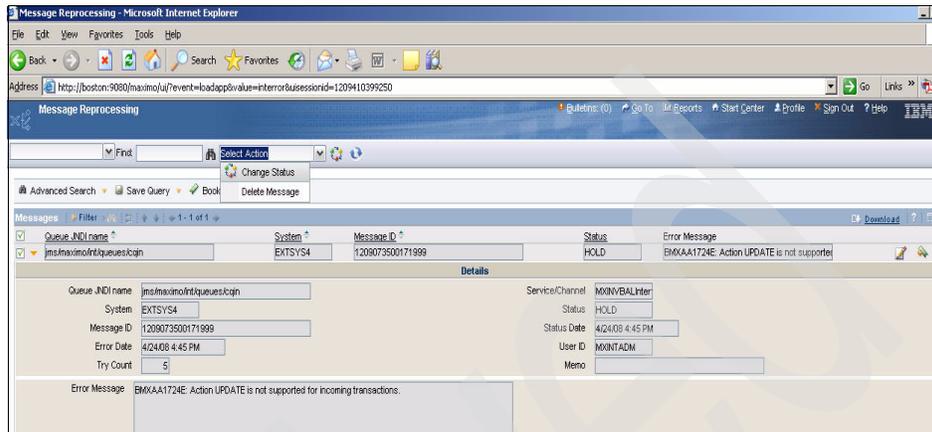
Figure 8-66   Message Reprocessing application

8.4.3 Data import using Interface tables

This section covers the necessary steps for importing data using Interface tables for the Asset object.

Assumptions

Prior to initiating the data import for the Asset object, verify that the following requirements have been met.

► Review the configuration settings as outlined in 7.4, "Basic Maximo integration configuration checklist" on page 170.

► We use the following prepackaged Integration Framework components in our example:

  – MXAsset object structure
  – MXASSETInterface for enterprise service
  – EXTSYS1 for external system.

Note: This section provides an example scenario. We recommend you perform these steps in a development environment first. Subsequently, create a duplicate system to maintain the state of the original external system.

► Verify the relevant cron tasks are active. Choose **Go To** → **System Configuration** → **Platform Configuration** → **Cron Task** and make sure the

following cron tasks are enabled and have the relevant inbound cron instance names defined and scheduled:

- – IFACETABLECONSUMER
- – JMSQSEQCON

Select Object Structure

Follow these steps to select the Object Structure:

1. From the Object Structure application, search for the MXASSET Integration Object. (Select **Go To** → **Integration** → **Object Structures**).

2. Verify the **Enable Flat structure** check box is enabled.

3. Verify no conflicts exist because of duplicate fields by selecting **Add/Modify Alias** from Select Action.

4. Save your results.

Define Interface table

Follow these steps to define the Interface table:

1. From the Enterprise Services application (**Go To** → **Integration** → **Enterprise Services**), search for the MXASSETInterface integration point. Verify the interface is associated with the proper Object Structure (for example, MXASSET).

2. Specify the Interface table name (for example, MXASSETInterface) in the Interface Table field.

3. Save your results.

Create Interface table

Follow these steps to create the Interface table:

1. Select the external system (for example, EXTSYS1) from which data is to be imported from the System tab in the External Systems application (select **Go To** → **Integration** → **External Systems**).

2. In the End Point field, select MXIFACETABLE.

3. Verify the **Enabled** check mark is selected.

4. From the Enterprise Service tab, click the **Select Service** button.

5. Associate the Enterprise Service with the external system by choosing it from the list and click **OK**.

6. After associating the Enterprise Service to the external system, you must create the interface table. To create the Interface table, choose **Go To** →**Select Action** →**Create Interface Table** to create the interface table.

Insert records in Interface table

Follow these steps to insert records in the Interface table:

1. Using a database query tool, import data records directly into the Interface table (for example, MXASSET) in the database. The following is an sample SQL script with an asset record:

```
Insert into MXASSET
(assetnum,as_description,as_location,as_itemsetid,
serialnum,as_orgid,as_siteid,transid,transseq,purchaseprice,
as_status,as_description_ld,failurecode,priority,vendor,
as_manufacturer,parent)
values('T60-105','IBM Lenovo
T60','BLDG905',NULL,NULL,'MAIN','AUSTIN','1','1','0','OPERATING',
NULL,NULL,NULL,NULL,NULL,NULL);
```

Important: The data imported is assumed to be correct and scrubbed, and all matching data to referential objects or domains is valid and correct.

2. Using a database query tool, from the back end, insert data into the MXIN_INTER_TRANS table. The following is a sample SQL script with a dummy record:

```
Insert into MXIN_INTER_TRANS (TRANSID,EXTSYSNAME,IFACENAME)values
(1,'EXTSYS1','MXASSETInterface');
```

Note: Before a record is inserted into the MXIN_INTER_TRANS table, make sure the relevant cron tasks are enabled.

3. At this stage the data that was inserted into the Interface table should have been successfully imported into the Asset object through the Maximo Asset Management business objects. You can confirm this by verifying that the data is imported from the Assets application.

4. Verify message tracking is enabled for error messages:

   a. Select **Go To** → **Integration** → **Message Tracking** and click **Enter**.

   b. Select the message from the Message Tracking application and view the message content.

<div align="right">

A

</div>

Manual configuring middleware

You can elect to automatically configure one or more Maximo Asset Management middleware components with the Maximo installation program. Alternatively, you can choose to manually configure one or more of the middleware servers. Manually configured installations involve configuring middleware components, the database server, the directory server, and the J2EE server, to work with Maximo before you use the Maximo Asset Management installation program.

This appendix provides information about the steps required to manually configure the middleware for use by Maximo Asset Management. Although we provide step-by-step instructions, we assume the reader has technical knowledge and experience with the middleware products. In this appendix our discussion is limited to middleware configuration based on Microsoft Windows 2003 and IBM WebSphere Application Server. The following topics are discussed:

► Manually creating the database
   – Oracle
   – DB2
   – Microsoft SQL Server
► Manually configuring the J2EE server
► Manually configuring Virtual Member Manager on WebSphere Application Server

249

- ► Manually configuring JMS queues
- ► Adding a server to the service integration bus
- ► Creating the service integration bus destination for the continuous inbound (CQINBD) queue
- ► Creating the service integration bus destination for the sequential inbound (SQINBD) queue
- ► Creating the service integration bus destination for the sequential outbound (SQOUTBD) queue
- ► Creating the continuous inbound (CQIN) JMS queue
- ► Creating the sequential inbound (SQIN) JMS queue
- ► Creating the sequential outbound (SQOUT) JMS queue
- ► Creating the JMS activation specification for the continuous inbound queue (CQIN)
- ► Manually creating a datasource for the persistence store
- ► Manually configuring the VMMSYNC cron task for Microsoft Active Directory
- ► Manually configuring IBM Tivoli Directory Server
- ► Manually configuring Microsoft Active Directory
- ► Rebuilding and redeploying Enterprise Archive (EAR) files

For more information regarding the manual configuration of middleware, refer to the installation guides *IBM WebSphere Application Server: Installation Guide* (mam71_install_was.pdf) *and BEA WebLogic Server: Installation Guide* (mam71_install_bea.pdf), both at:

http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=
/com.ibm.mam.doc_7.1/mam_welcome.htm

Manually creating database

If you chose not to have the Maximo Asset Management installation program automatically configure the database server, you must complete manual configuration of the database server before you use the Maximo Asset Management installation program. We describe the steps for manually creating the database for the Maximo Asset Management installation in this section.

Manually configuring DB2

Make sure you create user accounts that are operating system user accounts. Create operating system users using the user management tools available on the system. In addition, make sure the database instance is set to start automatically.

Manual database configuration properties

When creating the database, refer to Table A-1 to define your database attributes. The comments column is provided for your own notes regarding unique requirements for your environment.

Table A-1   DB2 database properties

Property or option	DB2 V9.1 value	DB2 V8.2 value	Comment
Database code set	UTF-8	UTF-8	
Port	50005	50005	
LOGFILSIZ	4096	4096	
APP_CTL_HEAP_SZ	1024	1024	
APPLHEAPSZ	1024	1024	
LOCKLIST	30000	30000	
LOGSECOND	4	4	
Tablespace	maxdata	maxdata	
Temporary tablespace	MAXTEMP	MAXTEMP	
Bufferpool	MAXBUFPOOL	MAXBUFPOOL	
Page size	32	32	
KB pages	4096	4096	
Create buffer pool	Immediately	Immediately	

Property or option	DB2 V9.1 value	DB2 V8.2 value	Comment
Extent and prefix size	> 200 MB < 2 GB		
Hard drive specification	SCSI	SCSI	
Dropped table recovery	Enabled	Enabled	
Database-managed Container name		CTGDAT	
Database-managed container type		File	
Database-managed container size		5000 MB	
Database-managed space container name		CTGTMP	
Database-managed space container type		File	
Database-managed space container size		3000 MB	

After setting the database configuration properties, follow these steps:

1. Grant permissions for the tablespace.

   Grant the user Maximo all privileges to the Maximo schema.

2. Create the schema.

   Create a schema that has the same name as the database user ID you have created and grant it all privileges.

3. Create the Maximo database.

   When running the maxinst utility to manually create the database, run the following scripts:

   To create a blank database:

   ```
   C:\ibm\SMP\maximo\tools\maximo>maxinst -e -sPRIMARY -tPRIMARY
   -iMAXIMO
   ```

   The -t and -s switches point to the table and storage space, respectively; -iMaximo is the switch and command used to create a blank database.

To create a database populated with sample data:

```
C:\ibm\SMP\maximo\tools\maximo>maxinst -e -sPRIMARY -tPRIMARY
```

The -t and -s switches point to the table and storage space, respectively. By default, the MAXDEMO database is created, as opposed to creating a blank database is using the -i switch.

Manually configuring SQL Server 2005

Assign the values listed in Table A-2 to the database when manually configuring the Microsoft SQL Server database properties.

Table A-2   SQL Server database properties

Property	Value	Comment
Suggested patch version	SP2 upwards	
Listener port	1433	
TCP/IP protocol	Enabled	
Full-text search	Enabled	
Authentication	Mixed Mode	
Tablespace	PRIMARY	
Storage space	PRIMARY	
Maximo user	MAXIMO	
dbowner	MAXIMO	Alternatively, the default user SA can also be used.

Tip: During our testing we had problems assigning the MAXIMO user as the database owner (dbo) on SQL Server 2005. If necessary, you can also use the default user SA as the MAXIMO administrator.

To create the Maximo user, run the following scripts:

```
sp_addlogin MAXIMO,MAXIMO go
sp_adduser MAXIMO,MAXIMO go
```

To create the database owner, run the following script:

```
sp_changedbowner MAXIMO
```

When running the maxinst utility to manually create the Maximo database, run the following scripts:

To create a blank database:

```
C:\ibm\SMP\maximo\tools\maximo>maxinst -e -sPRIMARY -tPRIMARY
-iMAXIMO
```

The -t and -s switches point to the table and storage space, respectively; -iMaximo is the switch and command used to create a blank database.

To create a database populated with sample data:

```
C:\ibm\SMP\maximo\tools\maximo>maxinst -e -sPRIMARY -tPRIMARY
```

The -t and -s switches point to the table and storage space, respectively. By default, the MAXDEMO database is created, as opposed to creating a blank database is using the -i switch.

Manually configuring Oracle

Assign the values listed in Table A-3 to the database when manually configuring Oracle database properties; these settings apply to a single schema database.

Table A-3   Oracle database properties

Property	Oracle 10g Value	Oracle 9 g Value	Comment
Suggested patch version	3	Release	
Shared memory management	Manual	-	
Shared pool	157286400	150	
Buffer cache	36000000	36	
Java pool	33554432	32	
Large pool	8388608	8	
PGA size	37748736	36	
Database character set	Unicode(AL32UTF8)	Unicode(AL32UTF8)	
nls_length_semantics	CHAR	CHAR	
open_cursors	1000	1000	
CTXSYS account	Leave unchecked	Leave unchecked	

Property	Oracle 10g Value	Oracle 9 g Value	Comment
Tablespace	maxdata (1000 MB autoextend)	maxdata (1000 MB autoextend)	
Temporary table space	maxtemp(1000 MB autoextend)	maxtemp(1000 MB autoextend)	
Maximo user	maximo	maximo	

To create the tablespace, run the following script:

```
Create tablespace maxdata datafile
'C:\oracle\product\10.1.0\oradata\maxdata\maxdata.dbf' size 1000M
autoextend on;
```

To create the temporary tablespace, run the following script:

```
create temporary tablespace maxtemp tempfile
'C:\oracle\product\10.1.0\oradata\maxtemp\maxtemp.dbf' size 1000M
autoextend on maxsize unlimited;
```

To create the Maximo user, run the following script:

```
create user maximo identified by maximo default tablespace maxdata
temporary tablespace maxtemp;
grant connect to maximo;
grant create job to maximo;
grant create trigger to maximo;
grant create session to maximo;
grant create sequence to maximo;
grant create synonym to maximo;
grant create table to maximo;
grant create view to maximo;
grant create procedure to maximo;
grant alter session to maximo;
grant execute on ctxsys.ctx_ddl to maximo;
alter user maximo quota unlimited on maxdata;
```

When running the maxinst utility to manually create the Maximo database, run the following scripts:

To create a blank database:

```
C:\ibm\SMP\maximo\tools\maximo>maxinst -e -sMAXDATA -tMAXDATA
-iMAXIMO
```

The -t and -s switches point to the table and storage space, respectively; -iMaximo is the switch and command used to create a blank database.

To create a database populated with sample data:

```
C:\ibm\SMP\maximo\tools\maximo>maxinst -e -sMAXDATA -tMAXDATA
```

The -t and -s switches point to the table and storage space, respectively. By default, the MAXDEMO database is created, as opposed to creating a blank database is using the -i switch.

Manually configuring the J2EE server

It is quite possible that an implementation might reside on an already existing WebSphere or WebLogic J2EE server.

> **Note:** With the implementation of WebLogic Server, the automated middleware option is not available. For instructions on how to install and deploy these servers, refer to the *BEA WebLogic Server* (mam71_install_bea.pdf), which is available at:
>
> http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.mam.doc_7.1/mam_welcome.htm

If you choose to not have the Maximo Asset Management installation program automatically configure the J2EE server, you must complete its manual configuration before you use the Maximo Asset Management installation program.

To configure the J2EE server on WebSphere Application Server prior to launching the Maximo Asset Management installation program, follow these steps:

1. Manually copy the keystore file from the IBM WebSphere Network Deployment deployment manager host to a temporary directory on the Maximo Asset Management administrative system where you are installing Maximo Asset Management:

    ```
    <WAS_home>/profiles/ctgDmgr01/etc/trust.p12
    ```

2. Launch the profile creation wizard.

3. Click **Next** in the Welcome dialog box.

4. Select the create a deployment manager option. Click **Next.**

5. Accept the default value or specify a profile name. Click **Next**.

6. Accept the default installation location. Click **Next**.

7. Accept the default values or specify the node name, host name, and cell name. Click **Next**.

8. Review the assigned port numbers. Click **Next**. Note the administrative port number; you will use this port number when invoking the console through a browser.

9. Select the **Run the Application Server as a Windows service**, and log on as a local system account. Click **Next**.

10. Click **Next** in the Profile summary dialog box.

11. Select the **Launch the First steps** console option. Click **Finish**.

12. Click the **Installation verification** link.

13. After installation verification completes, close the output window.

14. Use the launchpad command, and click the **Profile creation wizard** to open the First Steps window (if not open already).

15. Click **Nex**t in the Welcome dialog box.

16. Select Create a custom profile. Click **Next**.

17. Accept the default values or specify the appropriate information. Click **Next**.

18. Specify a unique profile name and select the **Make this profile the default** check box. Click **Next**.

19. Accept the default directory path. Click **Next.**

20. Specify a unique node name and the computer name (or IP address) of the machine where you are performing this installation. Click **Next**.

21. Review the port number listings. Click **Next**.

22. Click **Next** in the Profile summary dialog box.

23. Select the **Launch the First steps** console check box. Click **Finish**.Click **Exit**. If another First steps window is open, close it.

Manually configuring VMM on WebSphere Application Server

This section provides information for manually configuring Virtual Member Manager (VMM) to secure Maximo Asset Management. During the installation process, the Maximo Asset Management installation program provides the option of automatically configuring Maximo middleware. If you elected to have the Maximo Asset Management installation program automatically configure Maximo middleware, it performs Virtual Member Manager (VMM) configuration, among other tasks. If you elected to manually configure Maximo middleware for use with Maximo, you have to manually configure VMM. VMM provides the ability to

access and maintain user data in multiple repositories, and to federate that data into a single virtual repository.

The federated repository consists of a single named realm, which is a set of independent user repositories. Each repository may be an entire external repository or, in the case of LDAP, a subtree within that repository. The root of each repository is mapped to a base entry within the federated repository, which is a starting point within the hierarchical namespace of the virtual realm.

> **Note:** If you intend to configure VMM to use SSL with a federated LDAP repository, you must do so only after a successful Maximo Asset Management installation. If VMM is configured to use SSL with a federated LDAP repository prior to completing the Maximo Asset Management installation, the installation will fail.

To add an LDAP directory to the VMM virtual repository, you must first add the LDAP directory to the list of repositories available for configuration for the federated repository, and then add the root of baseEntries to a search base within the LDAP directory. Multiple base entries can be added with different search bases for a single LDAP directory.

The instructions provided in this section apply to IBM Tivoli Directory Server. If you are configuring VMM to use Microsoft Active Directory, substitute values you used in install/t_ccmdb_reusemsad.dita#t_ccmdb_reusemsad and install/t_ccmdb_manconfigMSAD.dita#t_ccmdb_manconfigMSAD where appropriate in this procedure.

You must modify the VMMCRONTASK as shown in install/t_ccmdb_configcrontask.dita#t_ccmdb_configcrontask.

> **Important:** Before you begin this procedure, make sure you have a wasadmin user created in your LDAP repository

Apply the values and properties as listed in Table A-4.

Table A-4 Tivoli Directory Server properties

Property	Value
Repository identifier	ISMITDS
Directory type	IBM Tivoli Directory Server Version 6
Primary host name	Enter host name or IP address of the IBM Tivoli Directory Server server.

Property	Value
Port	389
Support referrals to other LDAP servers	Ignore
Bind distinguished name	cn=root
Bind password	
Logon properties	Blank
Certificate mapping	EXACT_DN
Distinguished name of a base entry that uniquely identifies this set of entries in the realm	ou=SWG,o=IBM,c=US
Distinguished name of a base entry in this repository	ou=SWG,o=IBM,c=US
Realm name	ISMRealm
Primary administrative user name	wasadmin
Server user identity	Automatically generated server identity
Ignore case for authorization	Enabled
Person Account Entity type	PersonAccount
Person Account Base entry for the default parent	Enter ou=users,ou=SWG,o=IBM,c=US
Relative Distinguished Name properties	Enter uid
Group entity type	Group
Group Base entry for the default parent	Enter ou=groups,ou=SWG,o=IBM,c=US
Relative Distinguished Name	cn
OrgContainer Entity type	OrgContaine
OrgContainer Base entry for the default parent	Enter ou=SWG,o=IBM,c=US
Relative Distinguished Name	Enter o;ou;dc;cn
Enable administrative security	Enabled

Property	Value
Enable application security	Enabled
Use Java 2 security	Disabled

Performing post-installation tasks for the J2EE server

If you chose to have the Maximo Asset Management installation program automatically configure Maximo middleware, it performs J2EE server configuration, among other tasks. If you elected to manually configure Maximo middleware for use with Maximo Asset Management, you have to manually configure the J2EE server.

The following steps describe how to configure a WebSphere Application Server using the WebSphere administrative console:

1. From the Servers link in the tree view, click **Application servers**. Click **MXServer** in the main window (see Figure A-1).
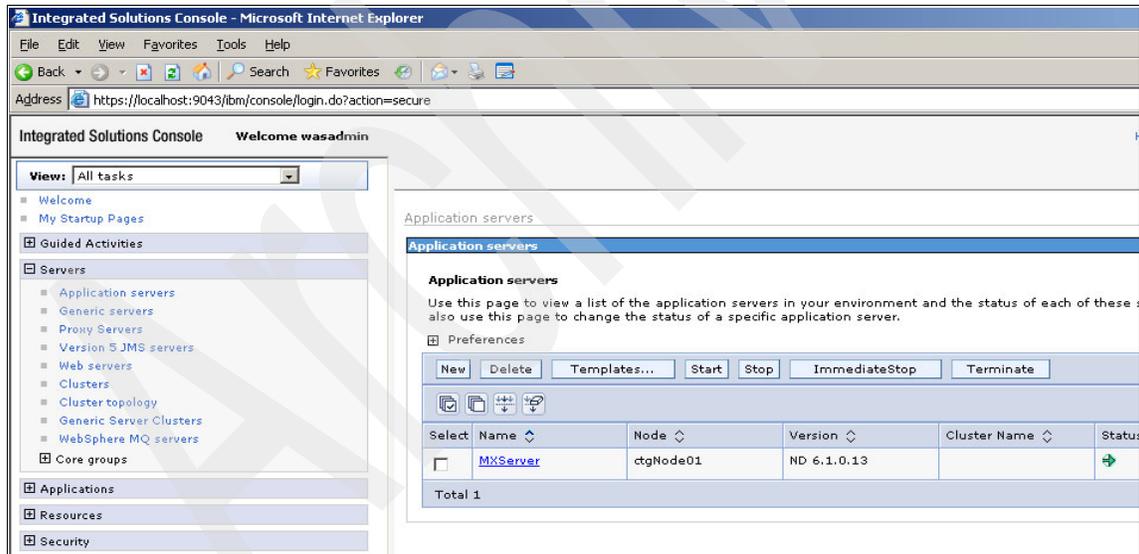
Figure A-1   MXServer

2. From the Server Infrastructure group, click the J**ava and Process Management** link (see Figure A-2).
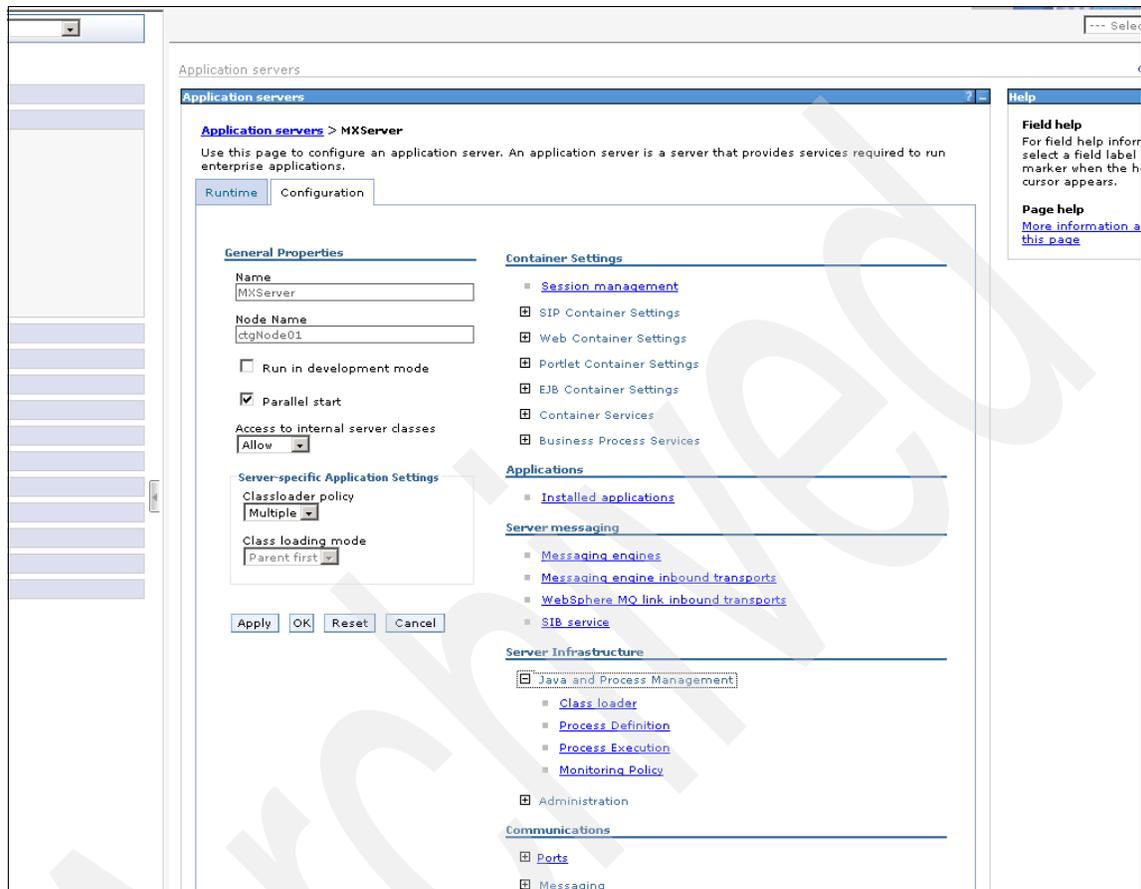
Figure A-2   Java and Process Management

3. Click **Process Definition**, then click **Java Virtual Machine** (see Figure A-3).

Figure A-3   Process Definition - Java Virtual Machine

4. Scroll down and enter 512 for **Initial Heap Size** and 1024 for **Maximum Heap Size** and click **OK** (see Figure A-4).
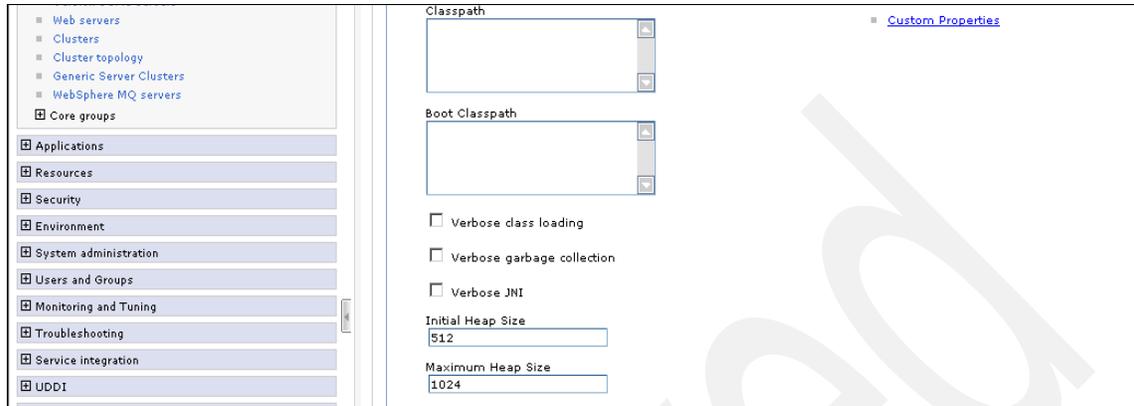
Figure A-4   Heap size definition

5. Click **Save** in the **Messages** panel (see Figure A-5).
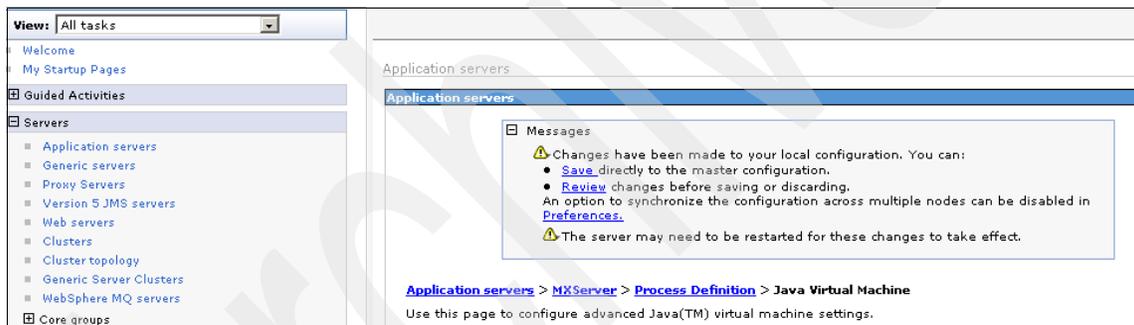
Figure A-5   Save heap size definition

6. Edit JVM™ memory settings for the deployment manager. From the **System administration** link in the tree view, click **Deployment manager** (see Figure A-6).
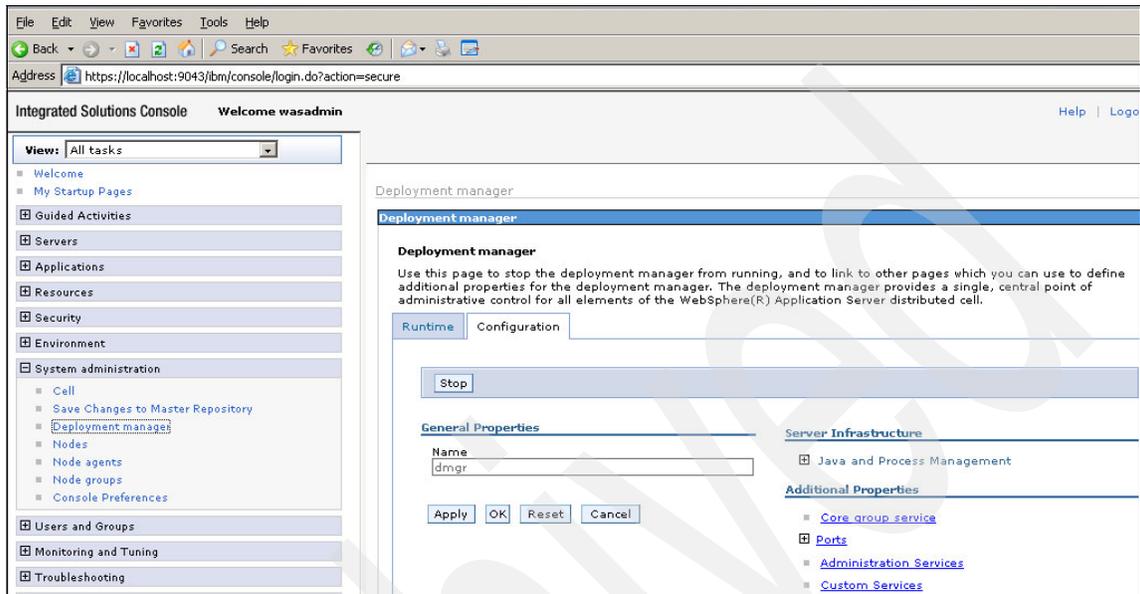
Figure A-6   Deployment Manager

7. From the Server Infrastructure group, expand the **Java and Process Management** link. Click **Process Definition** (see Figure A-7).
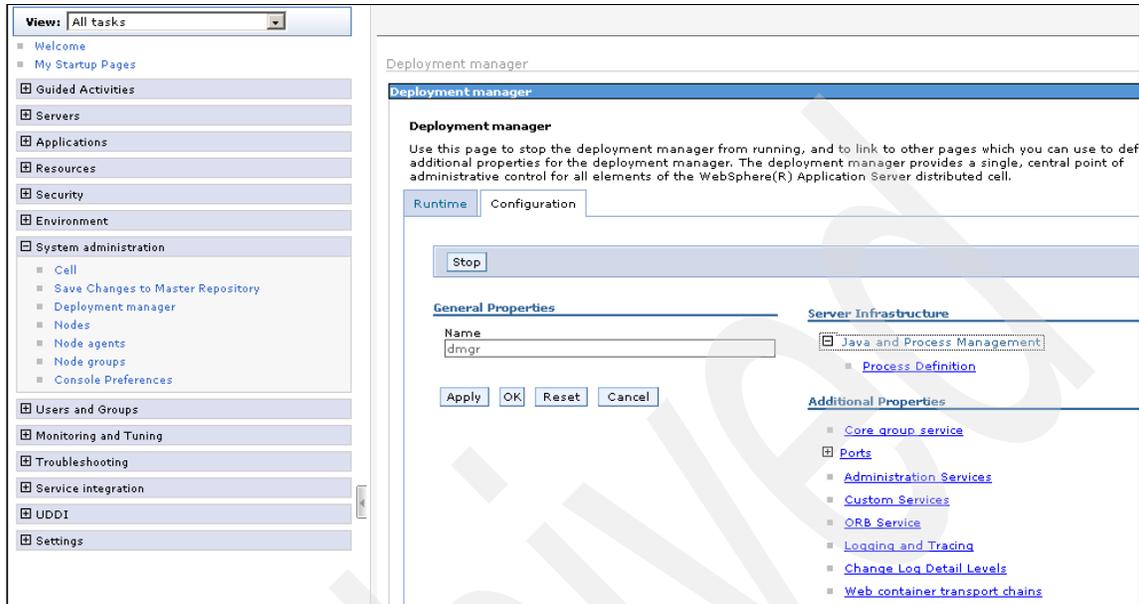
Figure A-7   Java and Process Management

8. Click **Java Virtual Machine** (see Figure A-8).
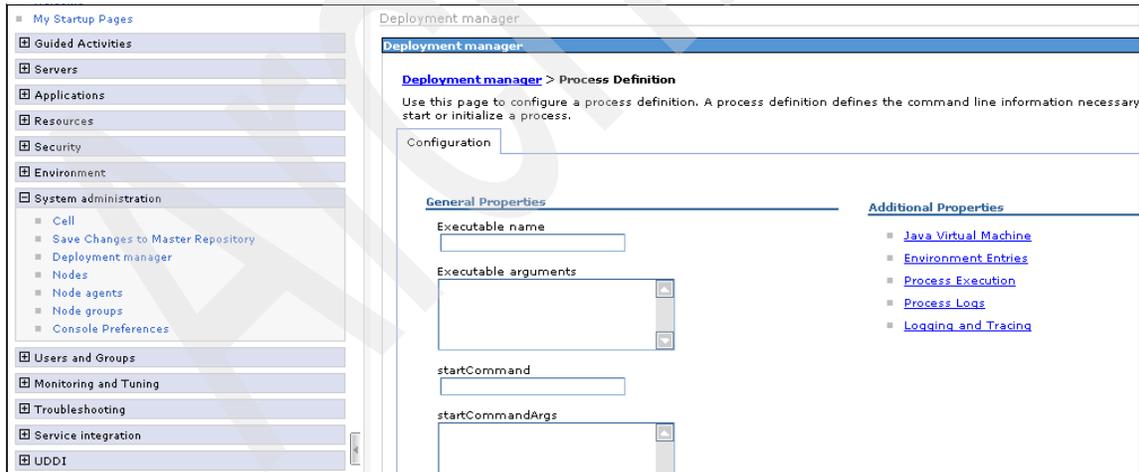
Figure A-8   Java Virtual Machine

9. Scroll down and type 512 for **Initial Heap Size** and 1024 for **Maximum Heap Size** and click **OK** (see Figure A-9).
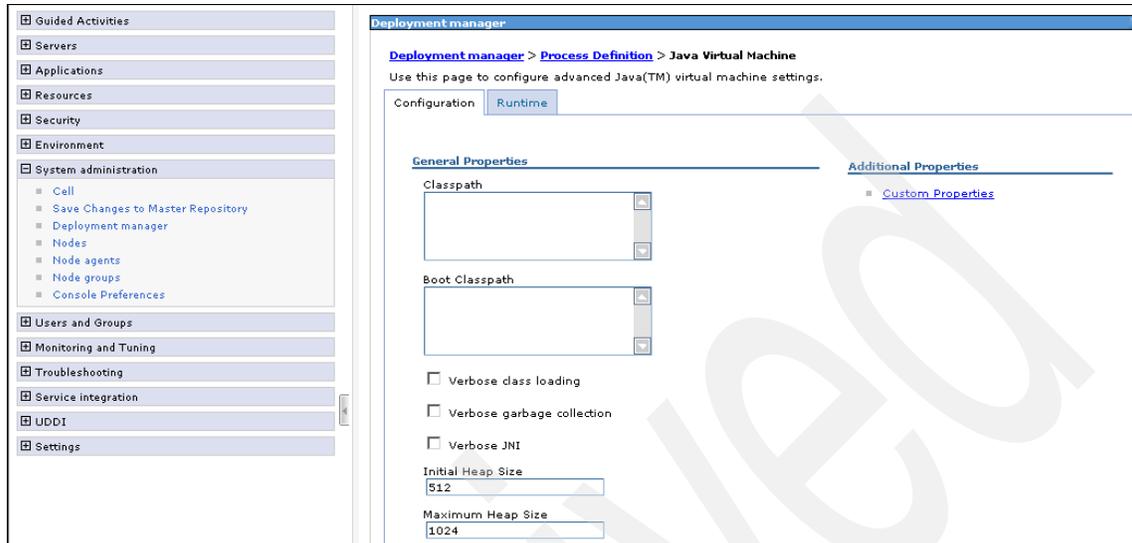
Figure A-9   Heap Size

10. Click **Save** in the Messages panel.

11. Identify the HTTP Transfer Port Numbers; select **Expand Servers** → **Application servers**, and click **MXServer** from the main window (see Figure A-10).
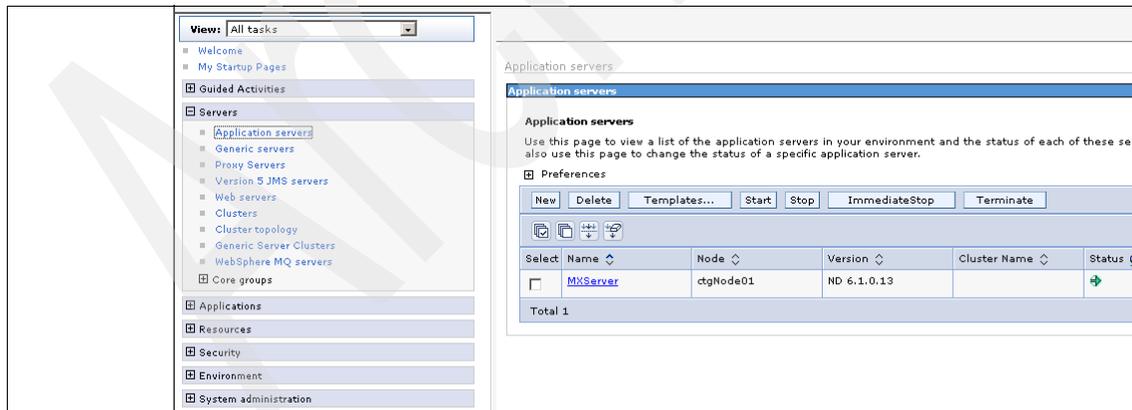
Figure A-10   MXServer

12. Open Web Container Settings and click **Web container transport chains** (see Figure A-11). Note the default port number as it appears with WC_defaulthost (9080). Click **Save**.
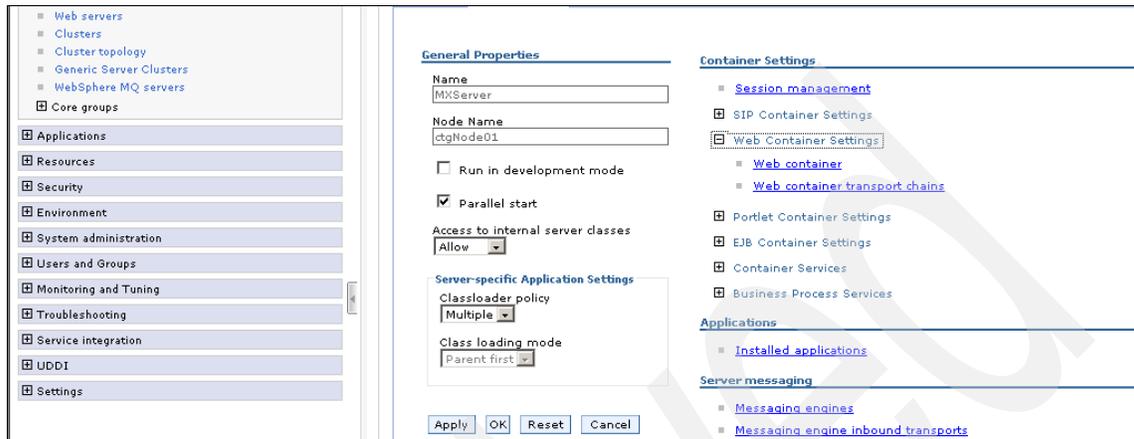
Figure A-11   Web container transport chains

13. Expand the **Environment** link from the tree view. Click **Virtual Hosts**.

14. Click **New**. In the General Properties section, type maximo_host in the Name field. Click **Apply** (see Figure A-12).
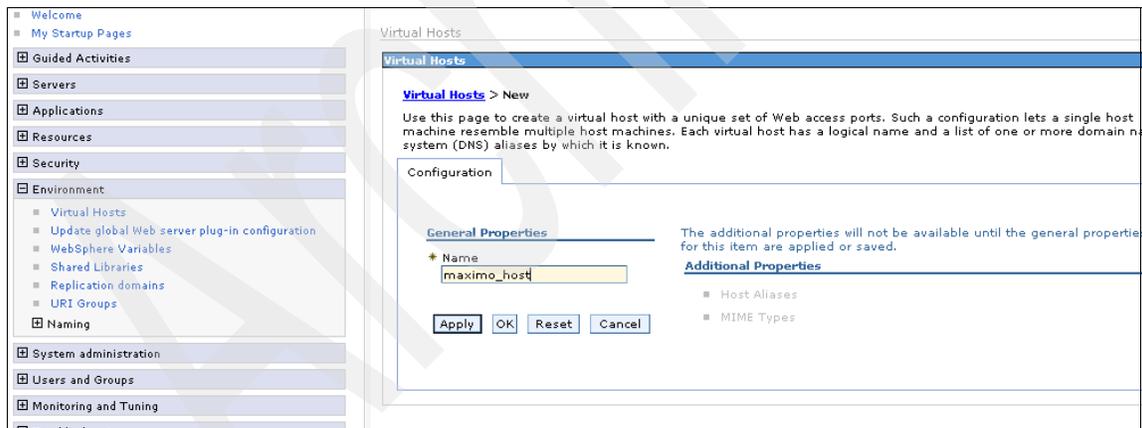
Figure A-12   maximo_host

15.Click **Save** (see Figure A-17 on page 270).
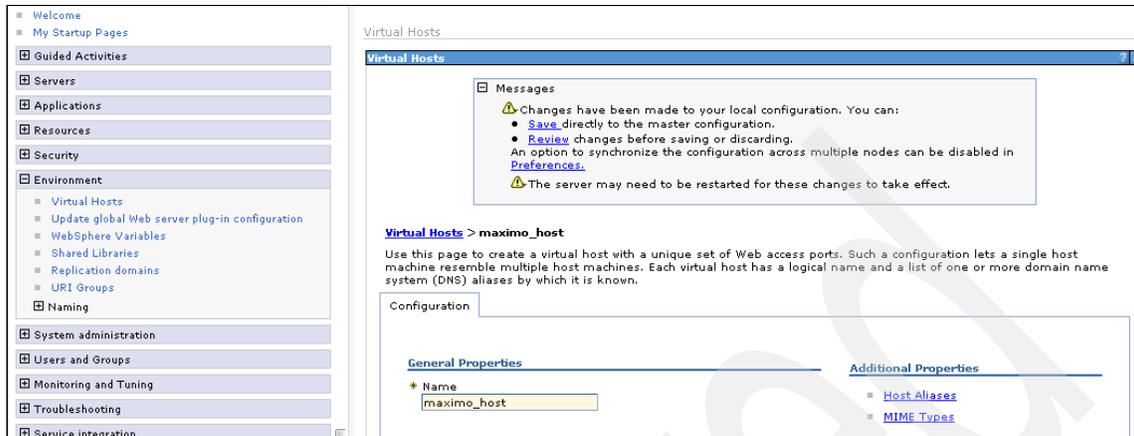
Figure A-13   Save maximo_host

16..From the Host Aliases link, click **New** (see Figure A-14).

Figure A-14   New host alias

17. Enter ***** (asterisk) in the Host Name field and type the HTTP port number, by default 80 (see Figure A-15).
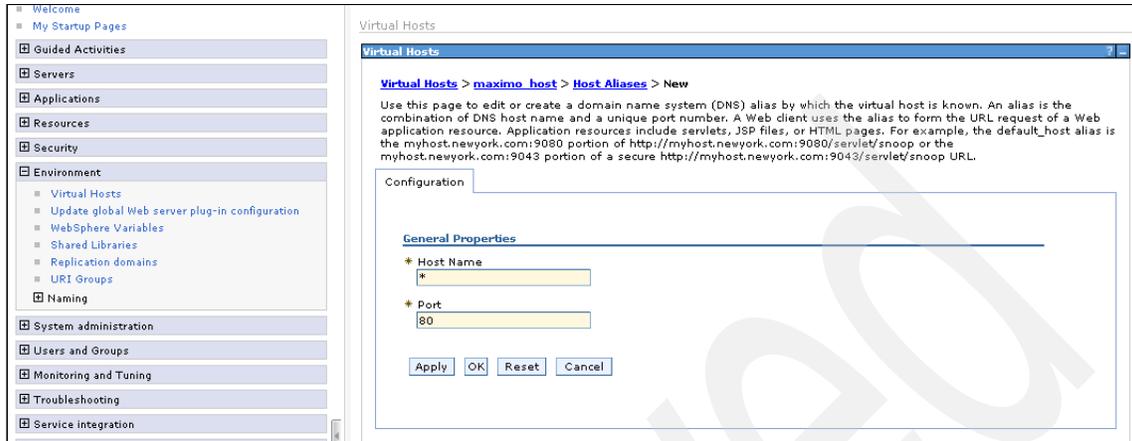
Figure A-15   HTTP port number

18. Apply the same procedure for ports: 9061 (see Figure A-16), 9443, 9080, and 9044.

> **Tip:** Make sure the port numbers provided in step 18 are not already in use because port conflicts cause problems during the startup process.
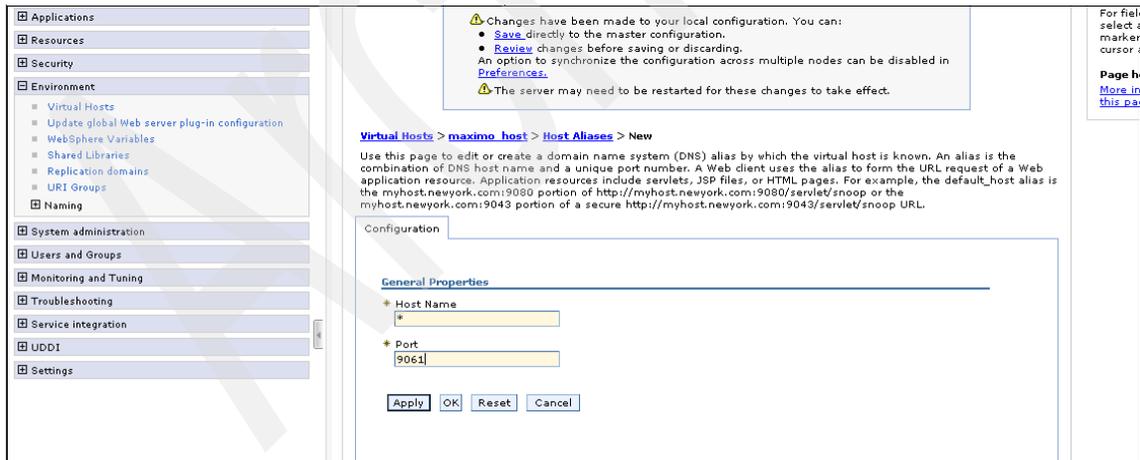
Figure A-16   Save HTTP port number

After configuring the ports, you view a list of the configured ports as shown in Figure A-17.

Figure A-17   Display HTTP port numbers

19. From the navigational breadcrumb trail, click **maximo_host**. Click **Apply** and then click **OK**.

Manually configuring JMS queues

> **Note:** This step is applicable only if your Maximo application includes the Integration Framework within its license agreement.

The following procedure provides steps for configuring JMS queues, which must be configured before deploying Maximo EAR files. During the installation process, the Maximo Asset Management installation program provided the option of automatically configuring Maximo middleware. If you elected for the Maximo Asset Management installation program to automatically configure Maximo middleware, it creates and configures JMS message queues, among other tasks. If you elected to manually configure Maximo middleware for use with Maximo Asset Management, you have to manually configure these message queues.

> **Tip:** After your queues have been set up, the queues cannot run unless they have been assigned to an enabled external system, and the relevant JMS queue cron tasks have been enabled. When the queues are running, monitor the effect of the system load of the active queues on the application's performance.

1. Ensure the Web server is started and log on to the WebSphere administrative console. Start the MXServer server by navigating to **Servers** → **Application Servers**, selecting **MXServer**, and then clicking **Start** (see Figure A-18).
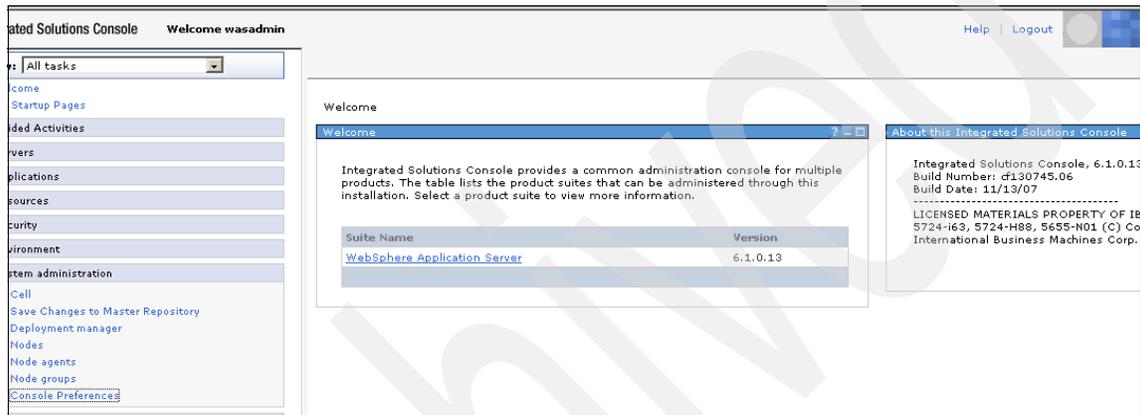
Figure A-18   MXServer

2. Select **System administration** → **Console preferences**. Select the **Synchronize changes with Nodes** option, and then click **Apply** (see Figure A-19).
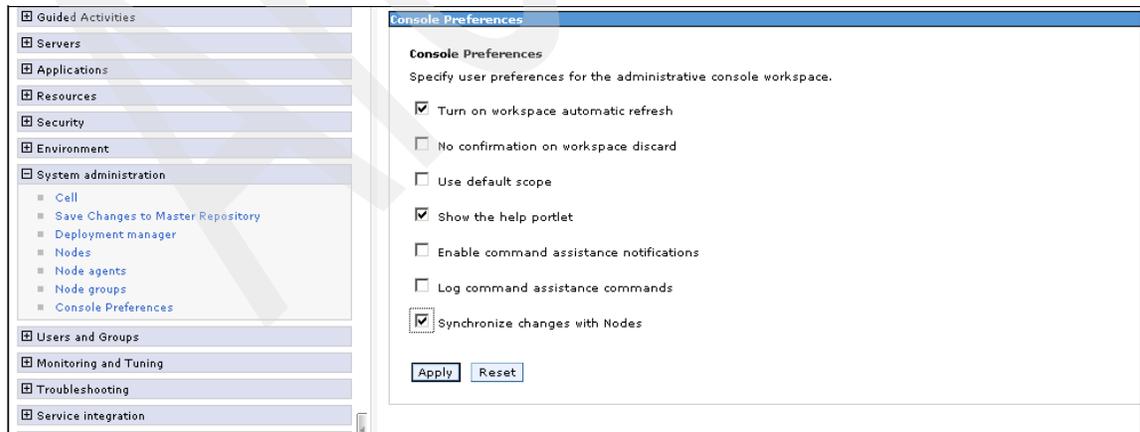
Figure A-19   Synchronize changes with nodes

3. Click **Service Integration** → **Buses** to open the Buses dialog (see Figure A-20). A bus is a group of interconnected servers and clusters that have been added as members of the bus.
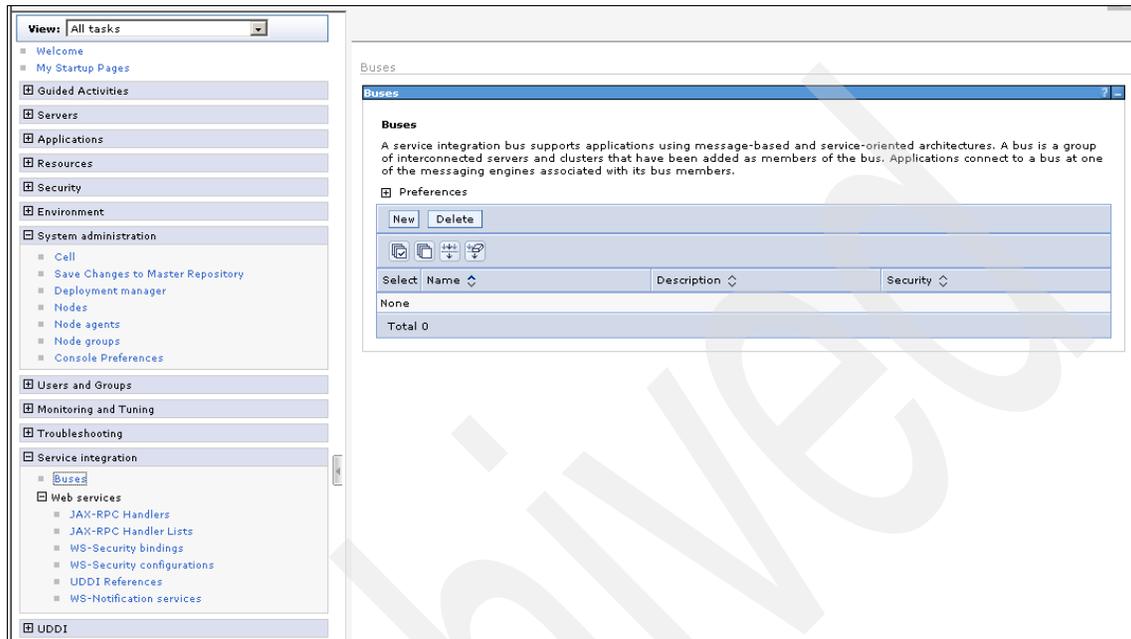
Figure A-20   Service integration

4. Click **New** and select **Buses** → **New**. A dialog box is displayed where you can add a new service integration bus. Deselect the **Bus security** check box (see Figure A-21). If you leave this box checked, intjmsbus inherits the Global Security setting of the cell.
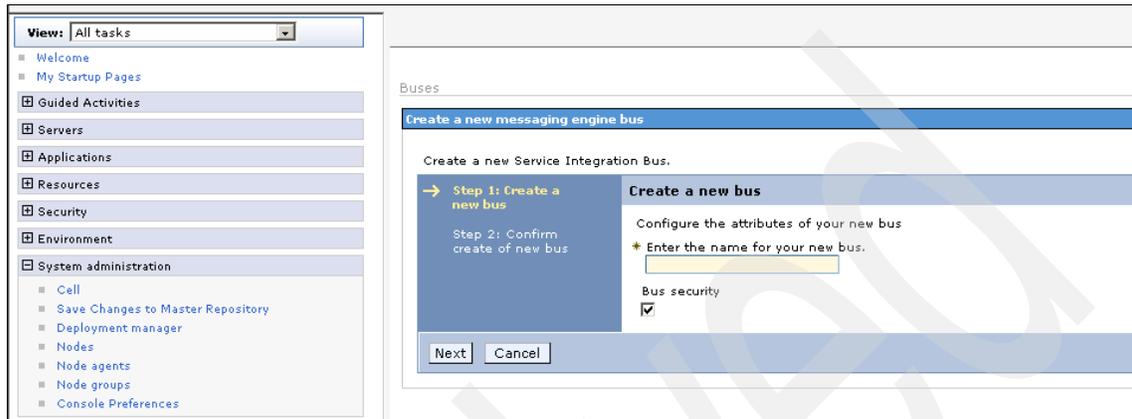
Figure A-21   New bus

5. Enter intjmsbus as the name of the new bus in the **Name** field. Click **Next** (see Figure A-22**)**.
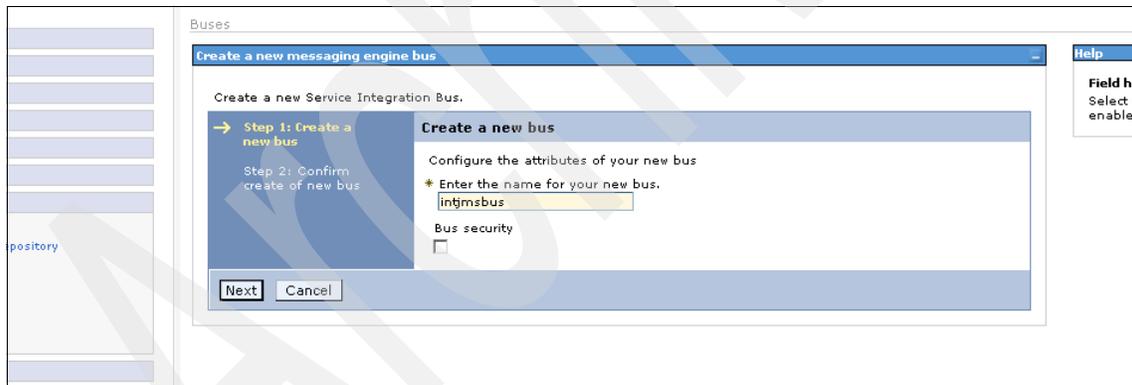
Figure A-22   intjmsbus

6. Click **Finish** (see Figure A-23).
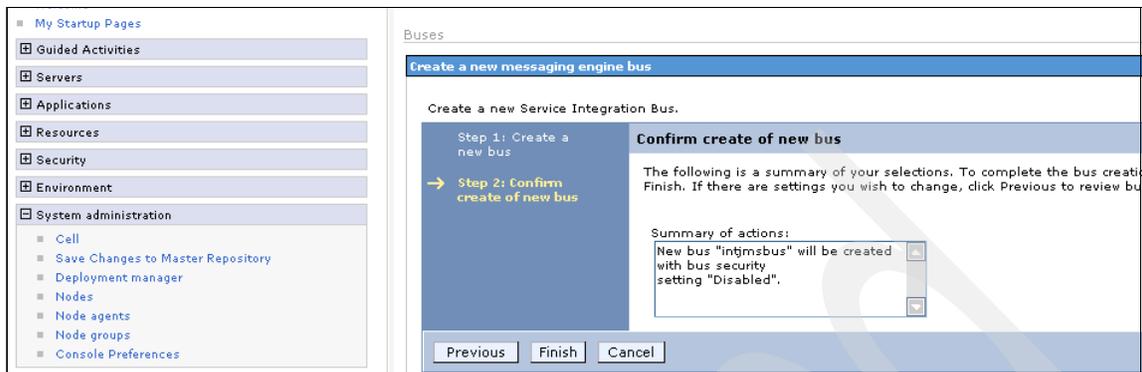
Figure A-23   Finish

This step propagates the JMS bus setup to the cluster configuration. Confirm that the build completed window displays the following:

▶ Bus name, for example, intjmsbus.

▶ Auto-generated, unique ID (UUID) - for example, 4BCAC78E15820FED.

▶ The Secure field is unchecked.

Click **Save** (see Figure A-24).
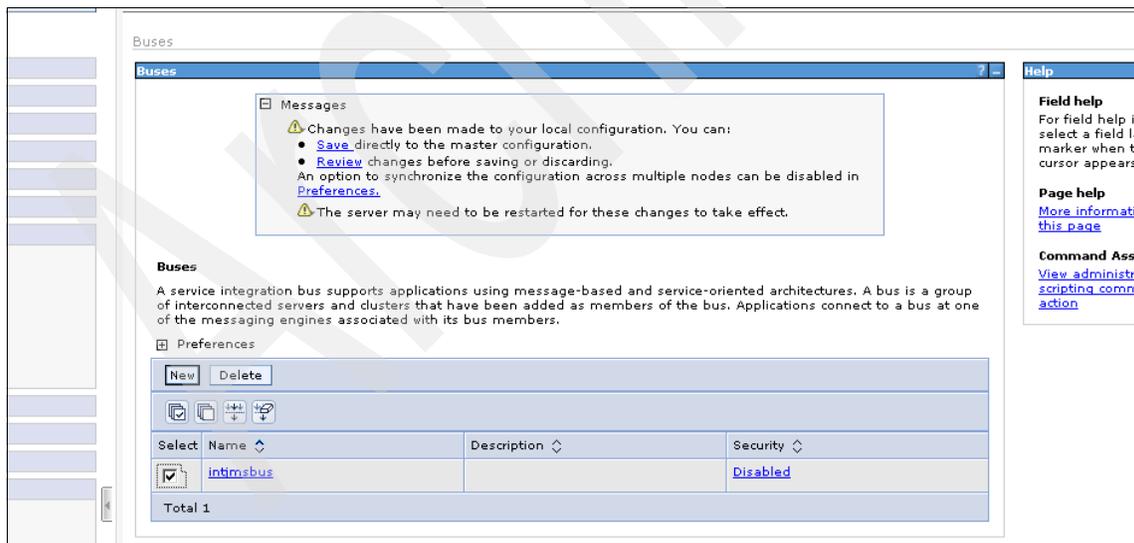
Figure A-24   Save bus

Adding a server to the service integration bus

Complete the following steps to add a server to the service integration bus:

1. From the WebSphere Administrative Console, click **Service Integration** → **Buses** to open the Buses dialog box.

▶ Click intjmsbus to open the Buses dialog box.

▶ Under Topology, click **Bus members**.

▶ Click **Buses** → **intjmsbus** → **Bus members** to access a dialog box where you click **Add** to open the Add a new bus member dialog box.

▶ Click the Server drop-down arrow, and select the server name **ctgNode01:MXServer** to add to the bus, and then click **Next** (see Figure A-25).

Figure A-25   Add new bus member

2. Check that the **File store** radio button is selected, and then click **Next** (see Figure A-26).
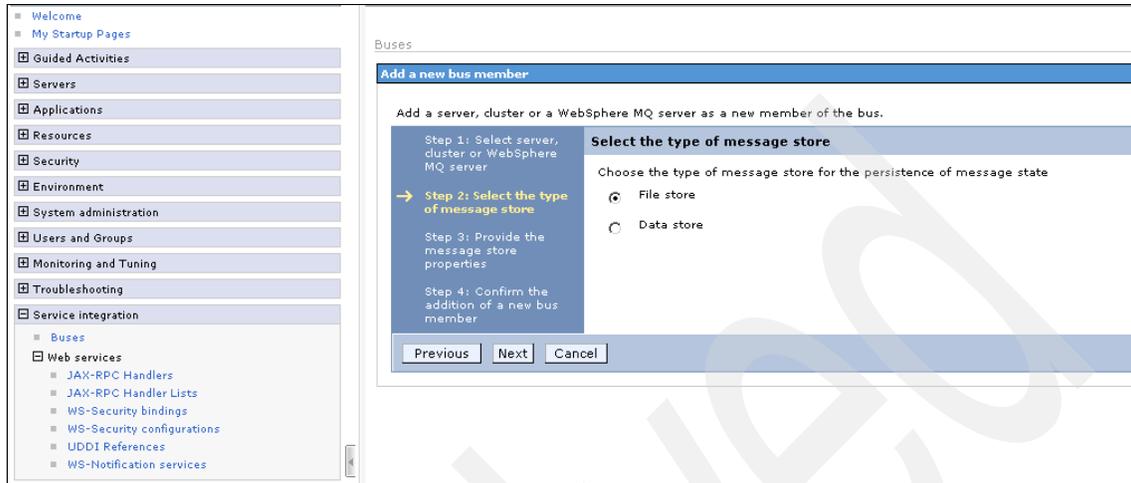
Figure A-26   File store

3. From the Provide the message store properties panel, click **Next** (see Figure A-27).
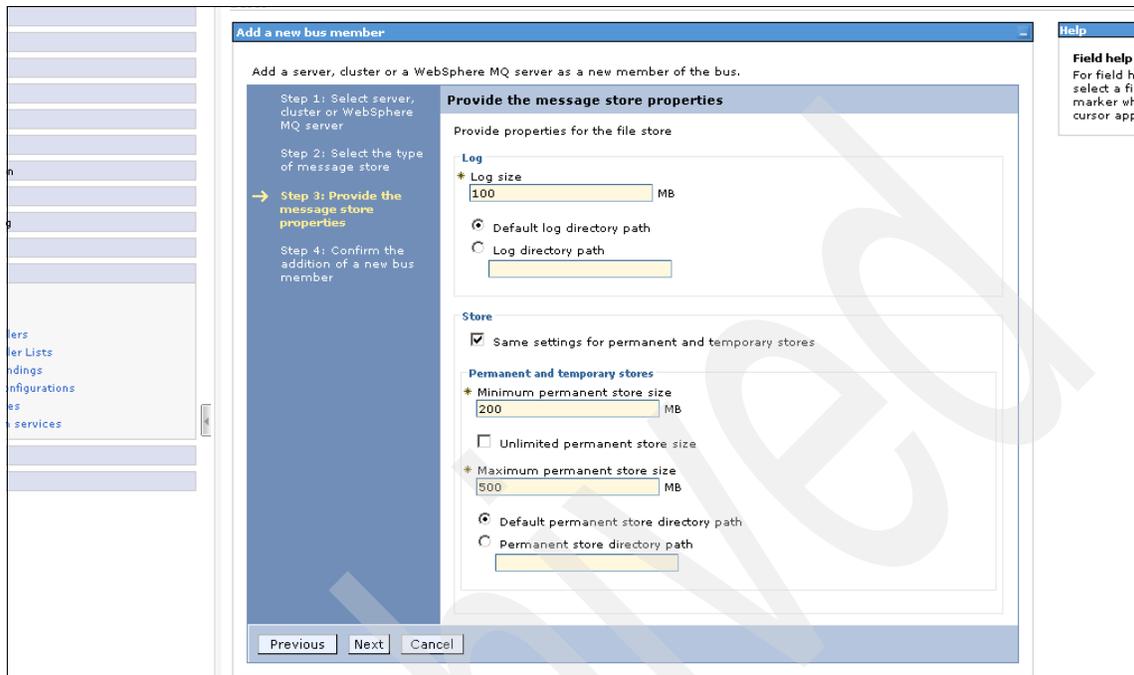
Figure A-27   Provide the message store properties

4. Click **Finish** (see Figure A-28).
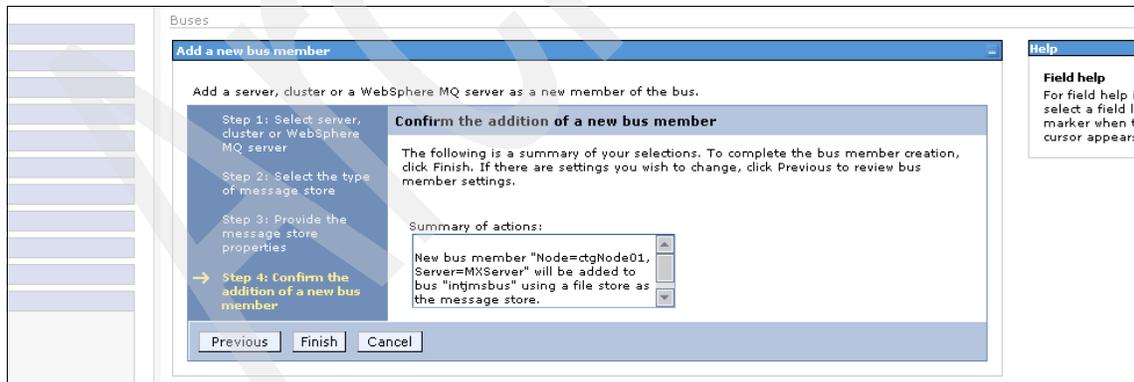
Figure A-28   Finish

5. Click **Save** (see Figure A-29).
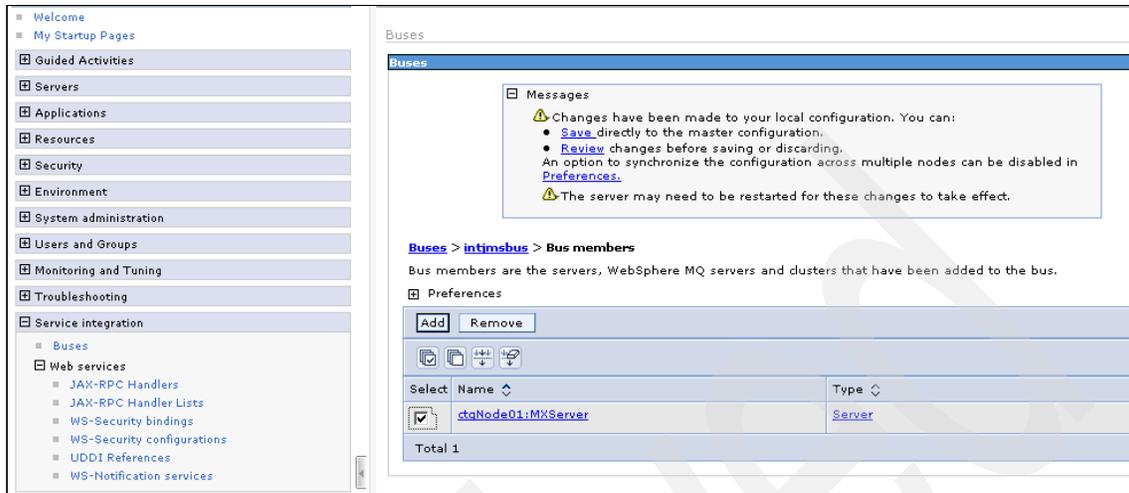
Figure A-29   Save changes

6. From Buses, select **intjmsbus**. Change the value of the High message threshold field to a minimum value of 500,000 messages, and click **Apply**, then **Save** (see Figure A-30).
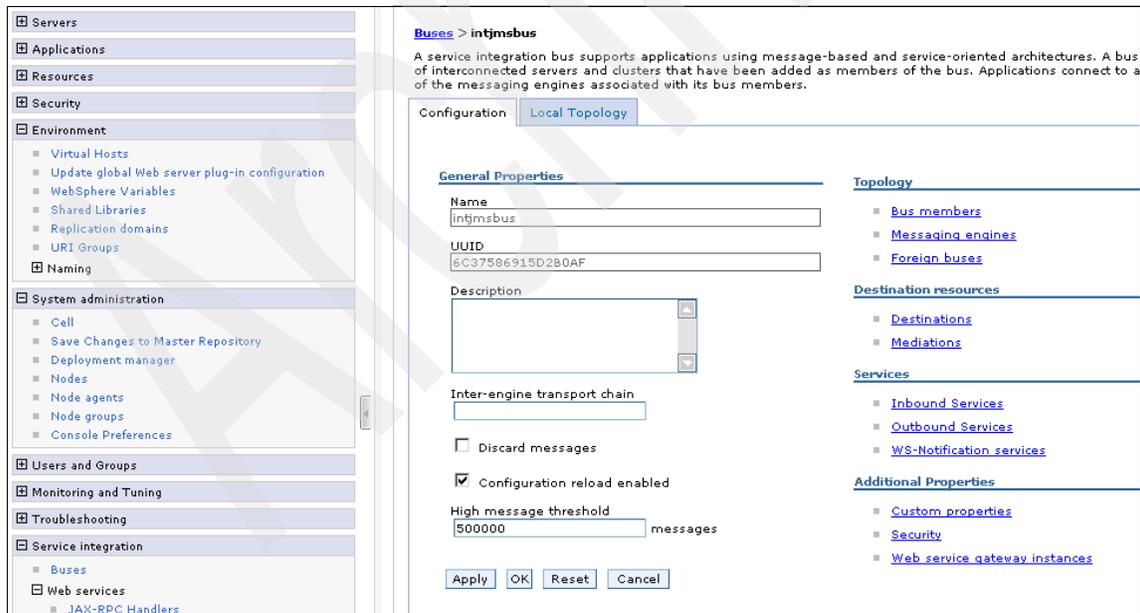
Figure A-30   High message threshold

> **Tip:** If the number of messages awaiting processing exceeds the high
> message threshold you set, the application server takes action to limit the
> addition of new messages in the processing queues. Depending on your
> message requirements, you may want to enter a message threshold value.
> You can determine an optimal message threshold setting by monitoring the
> messaging in/out queues and the impact of the message threshold setting on
> system performance. You might, for example, lower the threshold value if a
> value is degrading system performance. If you decide to change the high
> message threshold setting after the initial configuration, you must open the
> Additional Properties menu in the administrative console and change the
> threshold value for each child configuration.

Service integration bus destination for continuous inbound queue

To add a logical address for the continuous inbound bus destination queue
(CQINBD) within the JMS bus, complete the following steps:

1. From the **WebSphere Administrative Console**:

   a. Click **Service Integration** → **Buses** to open the Buses dialog box.

   b. Click **intjmsbus** to open the intjmsbus dialog box.

   c. Click **Destinations** under Destination resources, and select **Buses** →
      intjmsbus to access the Destinations dialog box (see Figure A-31). A bus
      destination - for example, CQINBD - is a virtual place within a service
      integration bus where applications can attach and exchange messages.
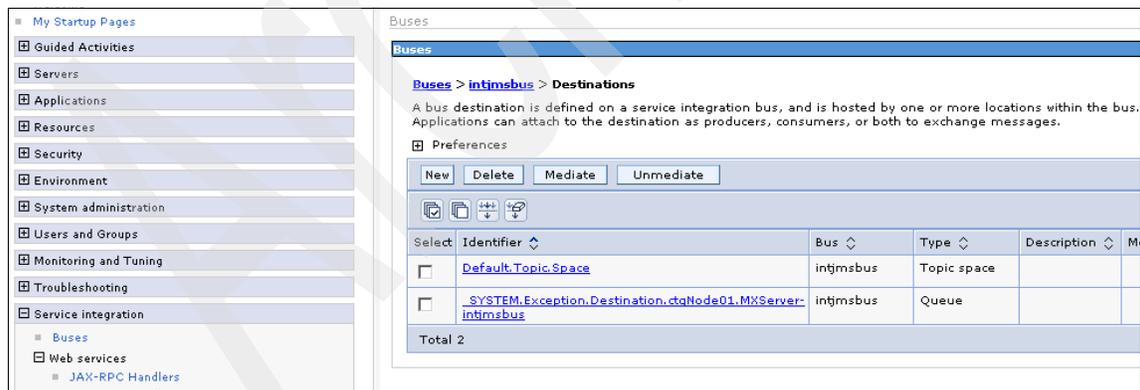
Figure A-31   Bus destination

2. Click **New** to open the Create new destination dialog box (see Figure A-32 on
   page 280). Leave Queue checked as the destination type, and click **Nex**t to
   open the Create new queue dialog box.
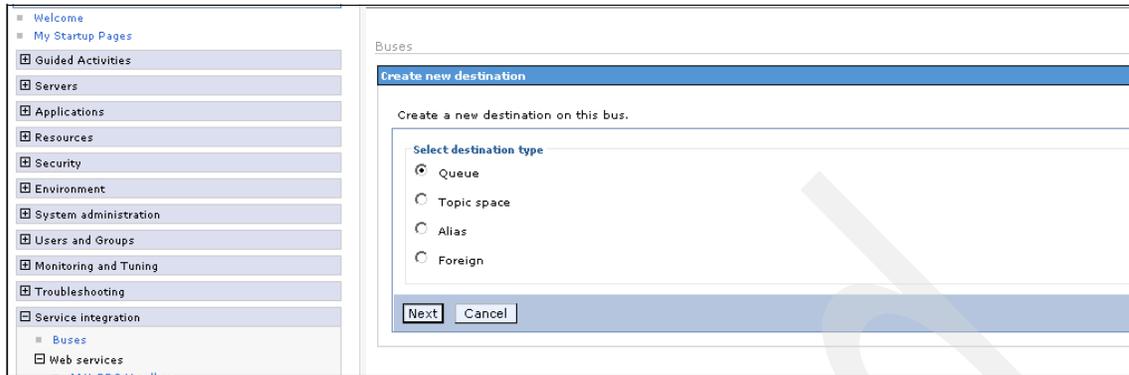
Figure A-32 New destination

3. Enter **CQINBD** in the Identifier field and Continuous Queue Inbound in the Description field, then click **Next** to open the Create a new queue for point-to-point messaging dialog box (see Figure A-33). Note that you must use the CQINBD value, and it must contain only uppercase letters.
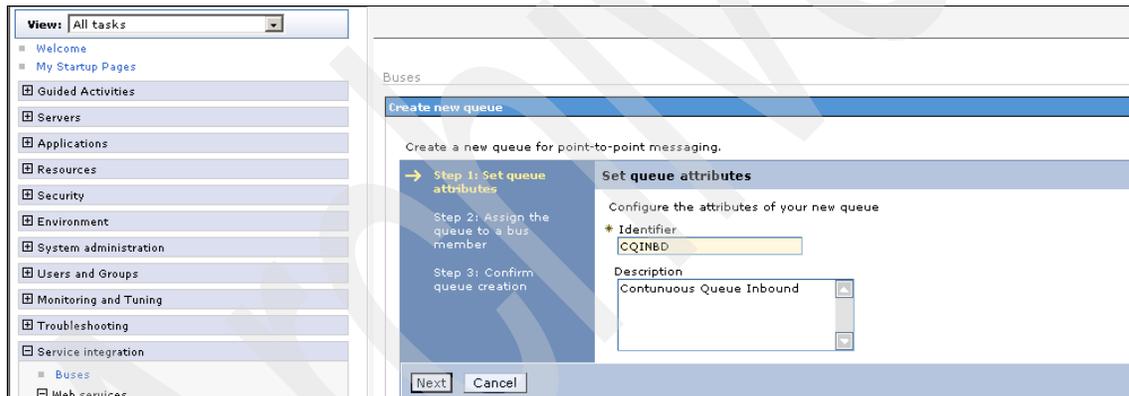
Figure A-33 Continuous Queue Inbound

4. Select the Bus Member pull-down and choose **Node=ctgNode01:Server=MXServer** as the bus member to store and process messages for the CQINBD bus destination queue (see Figure A-34). Click **Next** to open the Confirm queue creation dialog box.
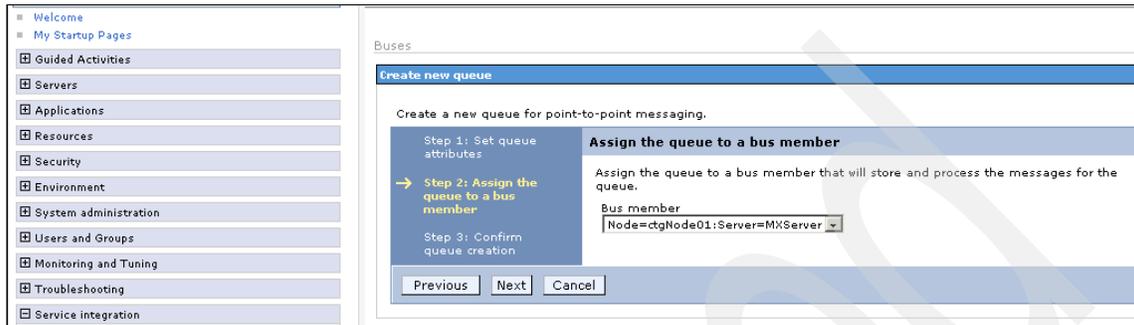
Figure A-34   Confirm queue creation

5. Review your selections, then click **Finish** to complete the creation of the CQINBD bus destination queue (see Figure A-35).
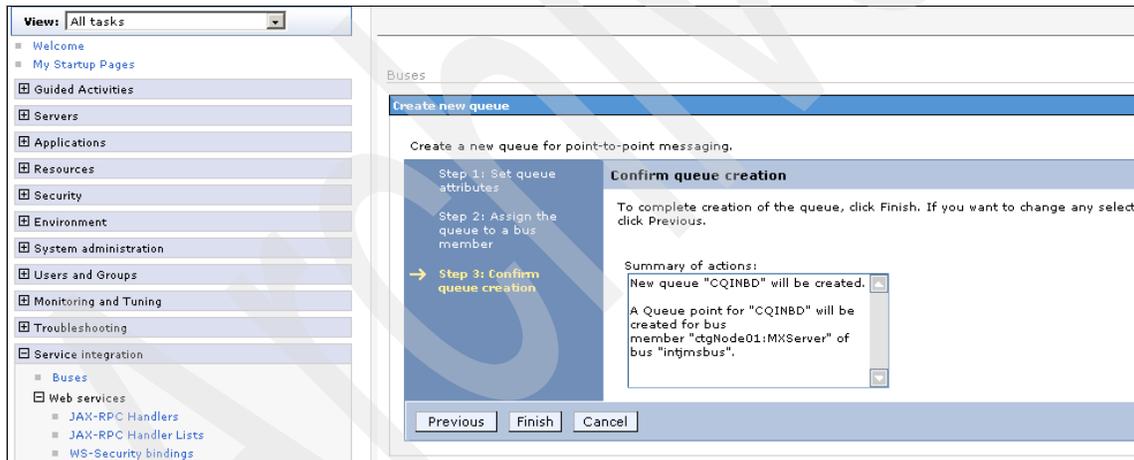
Figure A-35   Finish queue creation

6. Navigate the path **Buses** → **intjmsbus** → **Destinations**, then click **CQINBD** to open the configuration dialog box where you must make the following changes:

   a. Click **None** as the Exception destination value.

   b. Change the Maximum failed deliveries value to 1. This value represents the maximum number of times you want the system to process a failed

messaging attempt before forwarding the message to the exception destination.

   c. Click **Apply**.

   d. Click **Save**.

At this point, you apply the same sequence as applied to the continuous inbound (CQIN) JMS queue to both the sequential inbound queue (SQINBD - instructions provided in the section that follows) and outbound queue (SQOUTBD - see "Service integration bus destination for sequential outbound queue" on page 283).

Service integration bus destination for sequential inbound queue

To add a logical address for the sequential inbound bus destination queue (SQINBD) within the service integration bus, complete the following steps:

1. From the WebSphere administrative console, click S**ervice Integration** → **Buses** to open the Buses dialog box.

   – Click **intjmsbus** to open the Buses → intjmsbus dialog box.

   – Click **Destinations** under Destination resources to open the Buses →intjmsbus →Destinations dialog box. A bus destination - for example, SQINBD - is a virtual place within a service integration bus where applications can attach and exchange messages.

2. Click **New** to open the Create new destination dialog box. Leave **Queue** checked as the destination type, and click **Next** to open the Create new queue dialog box.

3. Enter **SQINBD** in the Identifier field and Sequential Queue Inbound in the Description field. Then click **Next** to open the Create a new queue for point-to-point messaging dialog box. Note that you must use the SQINBD value, and it must contain only uppercase letters.

4. Select the **Bus Member** pull-down and choose **Node=ctgNode01:Server=MXServer** as the bus member to store and process messages for the SQINBD bus destination queue. Click **Next** to open the Confirm queue creation dialog box.

5. Review your selections, then click **Finish** to complete the creation of the SQINBD bus destination queue.

6. Navigate the path **Buses** → **intjmsbus** → **Destinations**, then click **SQINBD** to open the configuration dialog box where you must make the following changes:

   a. Click **None** as the Exception destination value. Change the Maximum failed deliveries value to 1. This value represents the maximum number of

times you want the system to process a failed messaging attempt before forwarding the message to the exception destination.

b. Click **Apply**.

c. Click **Save**.

Service integration bus destination for sequential outbound queue

To add a logical address for the sequential outbound bus destination queue (SQOUTBD) within the service integration bus, complete the following steps:

1. From the WebSphere administrative console, click **Service Integration** → **Buses** to open the Buses dialog box.

   – Click **intjmsbus** to open the Buses →intjmsbus dialog box.

   – Click **Destinations** under Destination resources to open the Buses →intjmsbus →Destinations dialog box. A bus destination - for example, SQOUTBD - is a virtual place within a service integration bus where applications can attach and exchange messages.

2. Click **New** to open the Create new destination dialog box. Leave **Queue** checked as the destination type, and click **Next** to open the Create new queue dialog box.

3. Enter **SQOUTBD** in the Identifier field and Sequential Queue Outbound in the Description field, then click **Next** to open the Create a new queue for point-to-point messaging dialog box. Note that you must use the SQOUTBD value, and it must contain only uppercase letters.

4. Select the **Bus Member** pull-down and choose **Node=ctgNode01:Server=MXServer** as the bus member to store and process messages for the SQOUTBD bus destination queue. Click **Next** to open the Confirm queue creation dialog box.

5. Review your selections, then click **Finish** to complete the creation of the SQOUTBD bus destination queue.

6. Navigate the path **Buses** →**intjmsbus** →**Destinations**, then click **SQOUTBD** to open the configuration dialog box where you must make the following changes:

   a. Click **None** as the Exception destination value. Change the Maximum failed deliveries value to 1. This value represents the maximum number of times you want the system to process a failed messaging attempt before forwarding the message to the exception destination.

   b. Click **Apply**.

   c. Click **Save**.

Creating the JMS connection factory

You add a connection factory for creating connections to the associated JMS provider of point-to-point messaging queues.

To create a JMS connection factory, perform the following sequential steps:

1. From the WebSphere administrative console, click **Resources** →**JMS** → **Connection factories** (see Figure A-36).
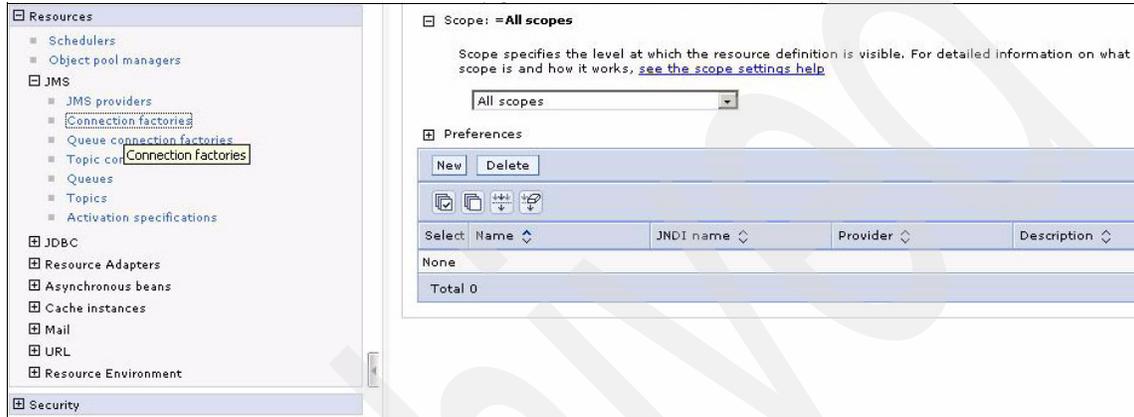
Figure A-36   Connection factories

2. From the Scope drop-down list. select **Cell=ctgCell01** (see Figure A-37).
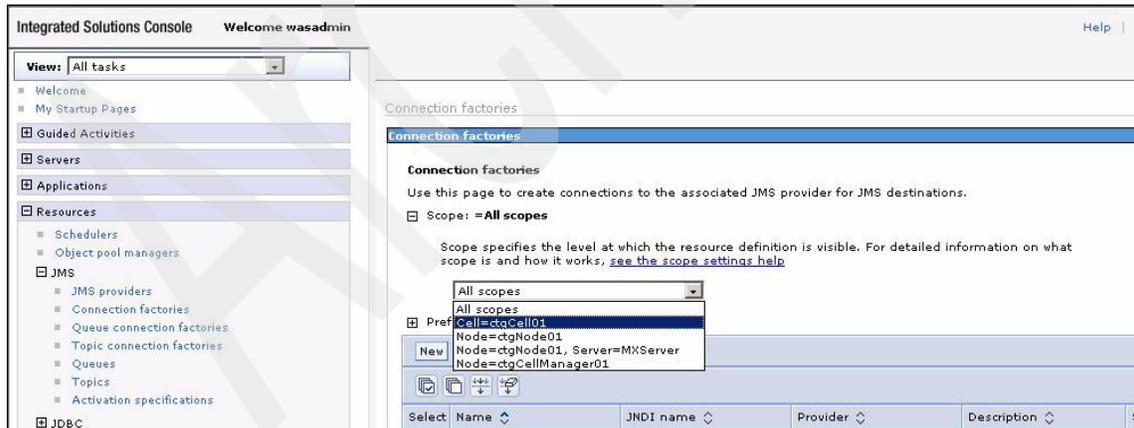
Figure A-37   Scope list

3. Click **New**.

4. Verify that the Default Messaging Provider is selected and click **OK** (see Figure A-38).
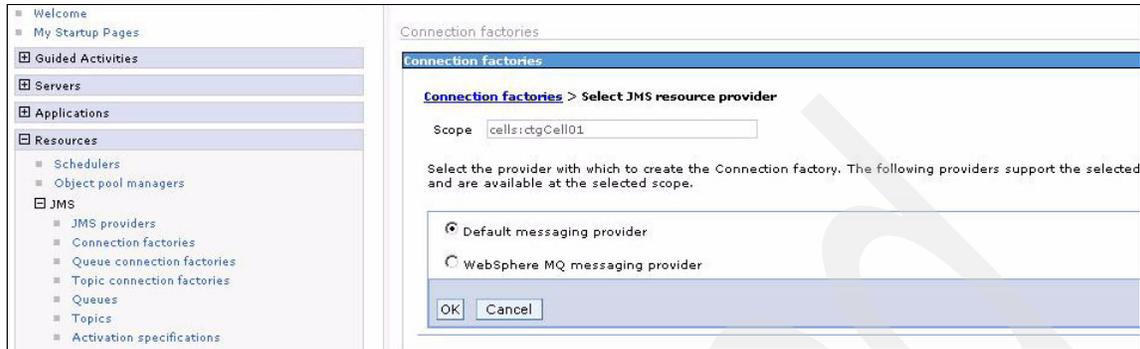
Figure A-38   New connection factory

5. Enter the following information, and then click **OK** (see Figure A-39):

   – Name: Enter **intjmsconfact**.

   – JNDI name: Enter jms/maximo/int/cf/intcf.

   – Bus name: Select i**ntjmsbus**.
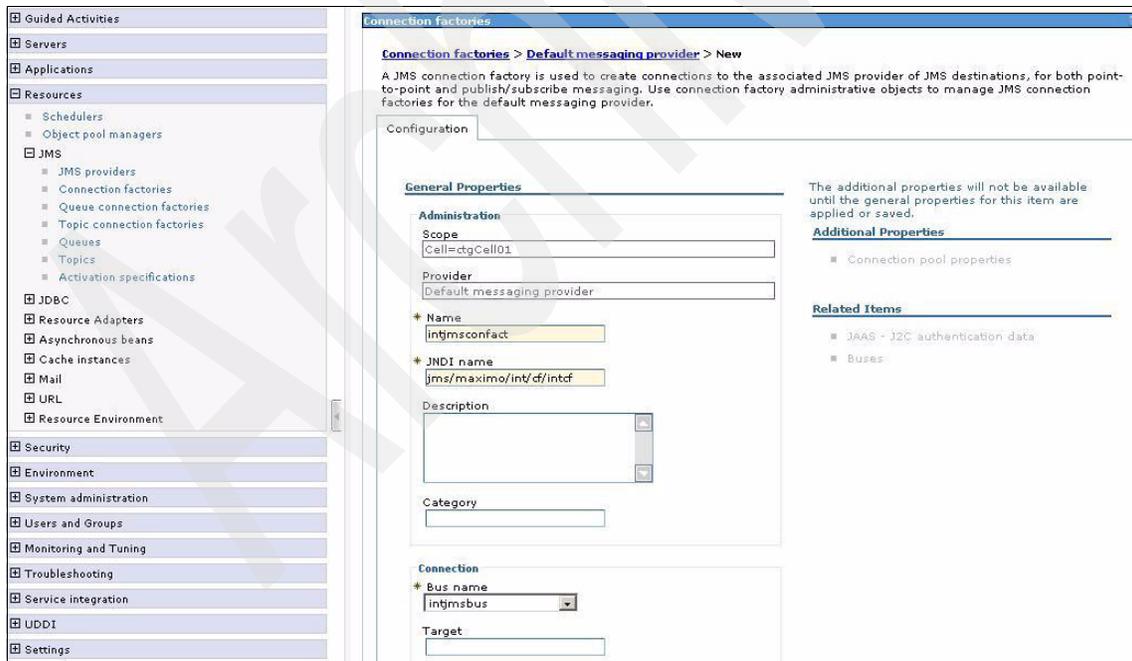
Figure A-39   intjmsconfact

6. Click **Save**.

Creating the continuous inbound (CQIN) JMS queue

You must create a JMS queue (CQIN) as the destination for continuous inbound point-to-point messages. Follow these steps:

1. From the WebSphere administrative console, click **Resources** → **JMS** → **Queues**.

2. From the Scope drop-down list, select **Cell=ctgCell01** and click **New** (see Figure A-40).
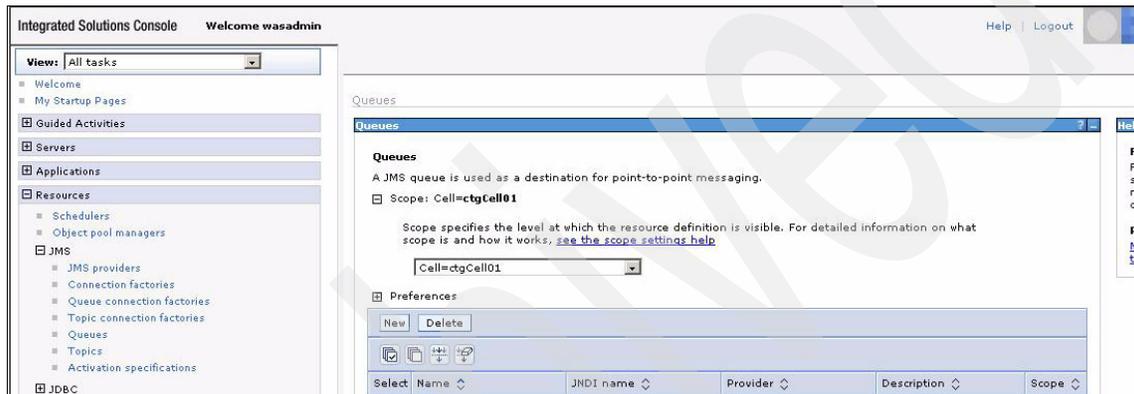
Figure A-40   Queues

3. Select Default messaging provider and click **OK** (see Figure A-41).
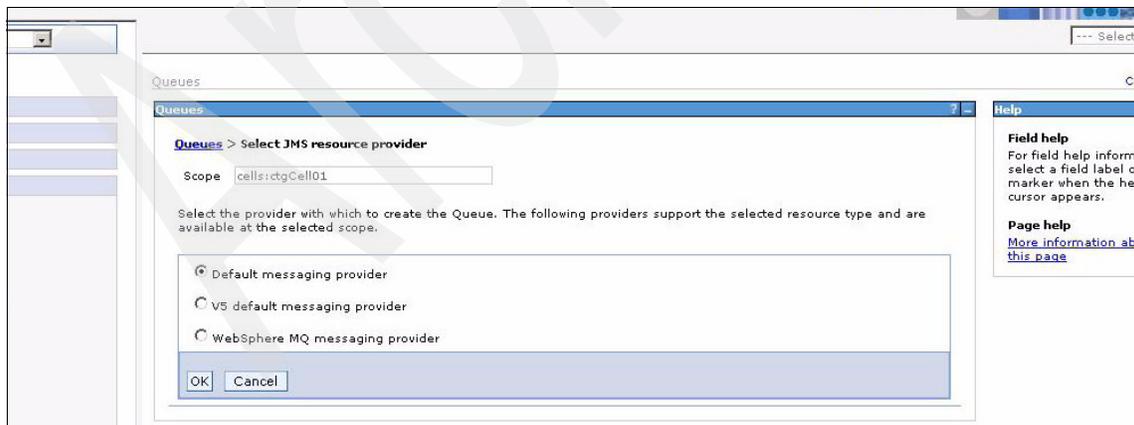
Figure A-41   Default messaging provider

4. Enter the following information, and click **OK** (see Figure A-42).

   – Name: Enter CQIN. (This value must contain only uppercase letters.)
   – JNDI name: Enter jms/maximo/int/queues/cqin.
   – Bus name: Select **intjmsbus**.
   – Queue name: Select **CQINBD**.

Figure A-42   CQIN

5. Click **OK**, then **Save**.

At this point, you can apply the same sequence as applied to the JMS activation of the continuous inbound (CQIN) queue to both the sequential inbound queue (SQIN) and outbound queue (SQOUT - refer to the sections that immediately follow).

Creating the sequential inbound (SQIN) JMS queue

From the WebSphere administrative console, click **Resources** → **JMS** → **Queues**.

1. From the Scope drop-down list, select **Cell=ctgCell01**.

2. Click **New**.

3. Enter the following information, and click **OK**.

   – Name Enter **SQIN**. (This value must contain only uppercase letters.)
   – JNDI name: Enter jms/maximo/int/queues/sqin.
   – Bus name: Select **intjmsbus**.
   – Queue name: Select **sqinbnd**.

4. Click **OK**, then **Save**.

Creating the sequential outbound (SQOUT) JMS queue

From the WebSphere administrative console, click
Resources →**JMS** →**Queues**.

1. From the Scope drop-down list, select **Cell=ctgCell01**.

2. Click **New**.

3. Enter the following information, and click **OK**.

   – Name: Enter SQOUT. (This value must only contain uppercase letters.)
   – JNDI name: Enter jms/maximo/int/queues/sqout.
   – Bus name: Select **intjmsbus.**
   – Queue name: Select **sqoutbd.**

4. Click **OK**, then **Save**.

Creating JMS activation specification for continuous inbound queue

You must activate the continuous inbound queue (CQIN) before it can receive messages. Complete the following steps to activate the CQIN queue:

1. From the **WebSphere Administrative Console,** click
   Resources →**JMS** →**Activation specifications** (see Figure A-43).
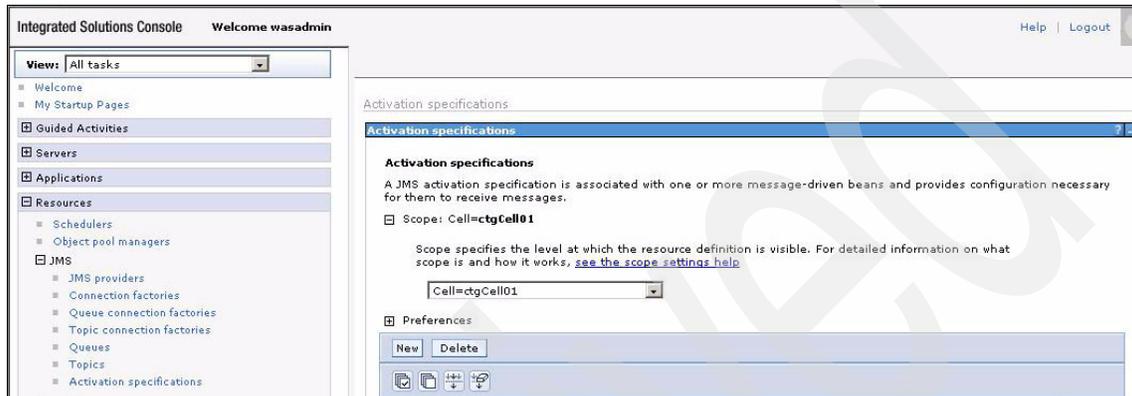
Figure A-43   Activation specifications

2. From the Scope drop-down list, select **Cell=ctgCell01** (see Figure A-44).
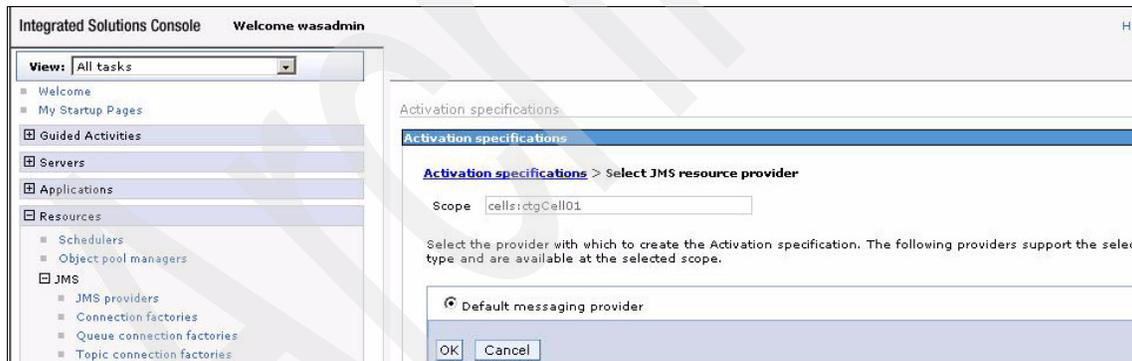
Figure A-44   Select JMS provider

3. Click **OK**.

4. Enter the following information, and then click **OK** (see Figure A-45).

   – Name: **intjmsact** - this value must only contain lowercase letters
   – JNDI name: **intjmsact**
   – Destination type: **Queue Destination**
   – JNDI name: jms/maximo/int/queues/cqin
   – Bus name: **intjmsbus**
   – Maximum concurrent endpoints: 10

Figure A-45   intjmsact

5. Click **Save**.

Make sure you stop all WebSphere Application Server-related processes and daemons.You must now restart these processes for the update to take effect. Start the bus member for the ctgNode MXServer intjmsbus if it is not started. If

you cannot start ctgNode MXServer intjmsbus, restart MXServer under
Servers →**Application servers**.

Manually creating a datasource for the persistence store

You have the option of having WebSphere Application Server use a DB2
database to store JMS messages. For more information about WebSphere
Application Server message storage, including the use of products other than
DB2, refer to the following Web sites:

► `http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=`
  `/com.ibm.websphere.nd.doc/info/welcome_nd.html`

► `http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=`
  `/com.ibm.websphere.pmc.nd.doc/tasks/tjm0035_.html`

If you chose to manually configure WebSphere Application Server, you have to
create a datasource to store JMS messages in a DB2 database. To create a
datasource for the persistence store, refer to the *IBM WebSphere Application
Server: Installation Guide* (mam71_install_was.pdf), which is available at:

`http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=`
`/com.ibm.mam.doc_7.1/mam_welcome.htm`

Manually configuring VMMSYNC cron task for Active Directory

To modify the VMMSYNC cron task for Microsoft Active Directory, complete the
following steps:

1. Log on to the Maximo console as maxadmin.

2. Navigate to Cron Task Setup and create a cron task with the properties listed
   in Table A-5.

Table A-5   Cron task properties

Property	Value
Crontask	VMMSYNC
Active?	Yes
Credential	Password for wasadmin in LDAP
GroupMapping	<basedn>ou=Groups,ou=SWG,dc=itsm,dc=com</basedn>
GroupSearchAttribute	cn

Property	Value
Principal	cn=wasadmin,ou=Users,ou=SWG,dc=itsm,dc=com
SynchAdapter	psdi.security.vmm.DefaultVMMSyncAdapter
SynchClass	psdi.security.vmm.VMMSynchronizer
UserMapping	\<basedn\>ou=Users,ou=SWG,dc=itsm,dc=com\</basedn\>
UserSearchAttribute	Uid

Manually configuring IBM Tivoli Directory Server

To configure IBM Tivoli Directory Server before launching the Maximo Asset Management installation program, you must create a new IBM Tivoli Directory Server instance. Make note of the properties specified in Table A-6.

Table A-6   IBM Tivoli Directory Server properties

Property/	Value	Comments
User	idsccmdb	
Encryption seed string		Minimum 12 characters
Instance description		
Server port number	389	
Server secure port number	636	
Admin daemon port number	3538	
Admin daemon secure port number	3539	
Administrator DN	cn=root	
Administrator password		
Database user name		
Database password		
Database name		
Database install location		
Character-set option	UTF-8/UCS-2	
Suffix (new)	o=IBM,c=US	

Property/	Value	Comments
LDIF DN (new)	ou=SWG,o=IBM,c=US	
LDIF DN (new)	ou=users	
LDIF DN (new)	ou=groups	
Suffix		

Important: Before you begin this procedure, ensure you have the following users and groups created in your LDAP repository:

► wasadmin
► maxadmin (usergroup MAXADMIN)
► mxintadmin (usergroup MAXADMIN)
► maxreg

Manually configuring Microsoft Active Directory

To configure Microsoft Active Directory prior to launching the Maximo Asset Management installation program, follow these steps:

1. Create a password policy as specified Table A-7.

Table A-7   Password policy properties

Policy	Value
Enforce password history	Not Defined
Maximum password age	Not Defined
Minimum password age	Not Defined
Minimum Password length	7 characters
Password must meet complexity requirements	Disabled
Store passwords using reversible encryption	Disabled

2. Create an organizational unit as specified in Table A-8.

Table A-8   Organizational unit properties

Type	Name
Organizational Unit	SW

3. Create the following security groups specified in Table A-9.

Table A-9   Security group properties

Group	OU	Scope
MAXADMIN	SWG	Global
MAXIMOUSERS	SWG	Global
MAXADMINPRE2K	SWG	Global

4. Create users and assign them to the groups specified in Table A-10.

Table A-10   Users and groups

First name	Full name	User login name	Password	Group
maxadmin	maxadmin	maxadmin	7 characters	MAXADMIN/MAXIMOUSERS
mxintadm	mxintadm	mxintadm	7 characters	MAXADMIN/MAXIMOUSERS
maxreg	maxreg	maxreg	7 characters	MAXIMOUSERS
wasadmin	wasadmin	wasadmin	7 characters	

Rebuilding and redeploying enterprise archive files

EAR files are archives that contain the required files to run an application. Maximo Asset Management uses the following two EAR files. Each contains one or more Web application modules (file names with .war extension):

► maximo.ear
► rmaximouiweb.war
► mboweb.war
► meaweb.war
► maximohelp.ear
► maximohelp.war

Table A-11 describes the various war.files.

Table A-11   WAR file names

File name	Definition
maximouiweb.war	Contains the Maximo UI-related JavaServer Pages. (.jsp files), Java classes, static HTML files, and static image files. The buildmaximoear.xml file contains information about the files in this module. This Web application uses the configuration details in the web.xml file, located in the <Maximo root>\applications\Maximo\Maximouiweb\webmodule\WEB-INF folder. This file also specifies the URL to access Maximo Help.
mboweb.war	Contains the Maximo business objects, Java classes, and dependent third-party Java classes. The build.xml file contains information about the files that are included for this module.
meaweb.war	The IBM Maximo Enterprise Adapter (MEA) enables Maximo to exchange data with other enterprise systems. Users create and maintain data in one system, and the MEA transfers it, which eliminates duplicate processing.
maximohelp.war	Provides the Maximo Help pages. The buildhelpear.xml file has information about all the files in this module.

The following is a list of file extensions that must be part of Maximo EAR files and deployed into the application server:

► .Class, .jar
► .properties
► .xml

Note: The .xml files listed are not .xml files applicable to the graphical representations set out in the Application Designer.

After you have changed one of the previously listed files, complete the following steps to rebuild and redeploy the EAR files:

1. Make sure all users are logged out of the Maximo Asset Management application.
2. Log on as system administrator to the administrative workstation where the Maximo Asset Management base service installation resides.
3. Stop all the relevant Maximo Asset Management services.
4. Back up the files you wish to change.
5. Change the files accordingly to your business requirements.

6. From the command prompt, point to the EAR file build command to rebuild your EAR file. Enter buildmaximoear.cmd as follows:

```
C:\ibm\SMP\maximo\deployment\default>buildmaximoear.cmd
```

7. Allow time for the EAR file to be rebuilt (the duration is usually no longer than two or three minutes).

8. Log on to the WebSphere administrative console, from **Applications** → **Enterprise Applications** (see Figure A-46). Select and uninstall the maximo.ear file, then click **Save**.
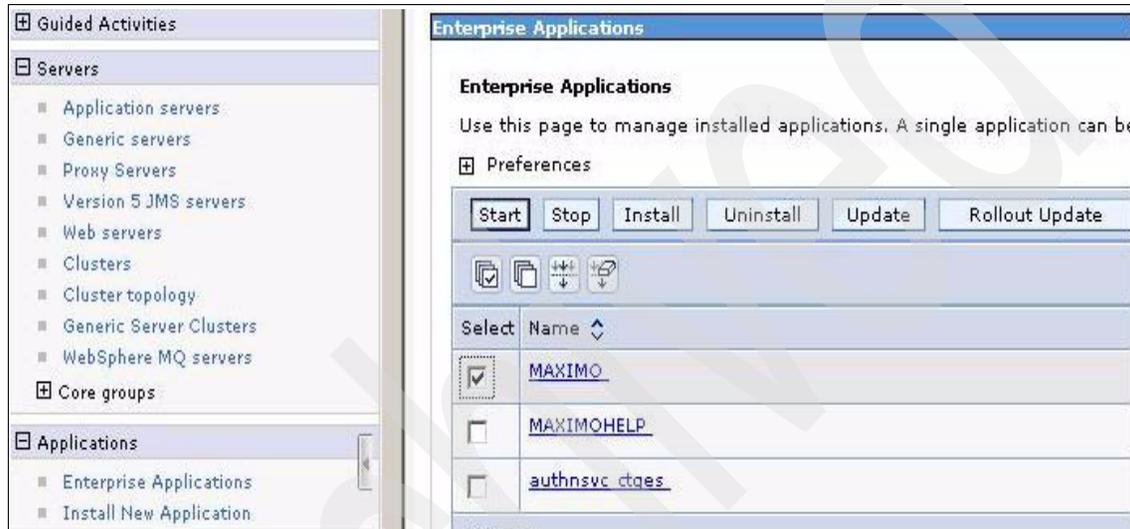
Figure A-46   Uninstall EAR file

9. As soon as you have uninstalled the old application, you have to reinstall the EAR files. From **Applications** →**Enterprise Applications**, click **Install New**

Application. You must point to your newly built EAR file to install the new application (see Figure A-47):

`C:\ibm\SMP\maximo\deployment\default\maximo.ear`
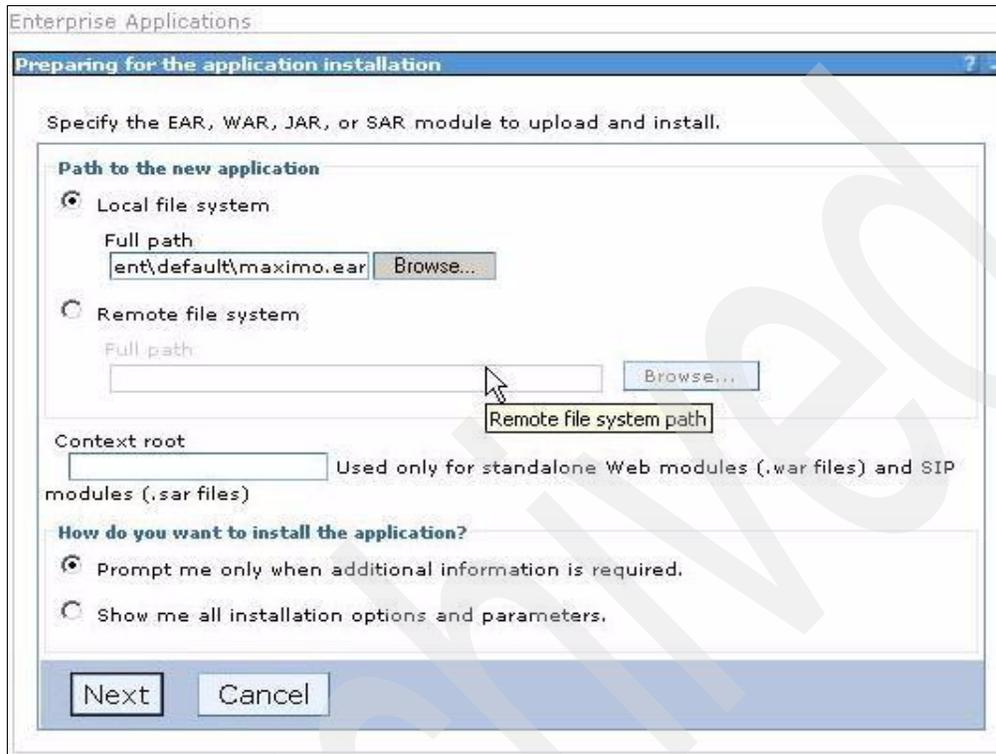
Figure A-47   Install new application

10.Click **Next** at Step 1: Select Installation options.

11.In Step 2: Map modules to server, make sure you apply and select both the targets and click **Next** (see Figure A-48).

```
WebSphere:cell=ctgCell01,node=ctgNode01,server=MXServer
WebSphere:cell=ctgCell01,node=ctgNode01,server=webserver1
```
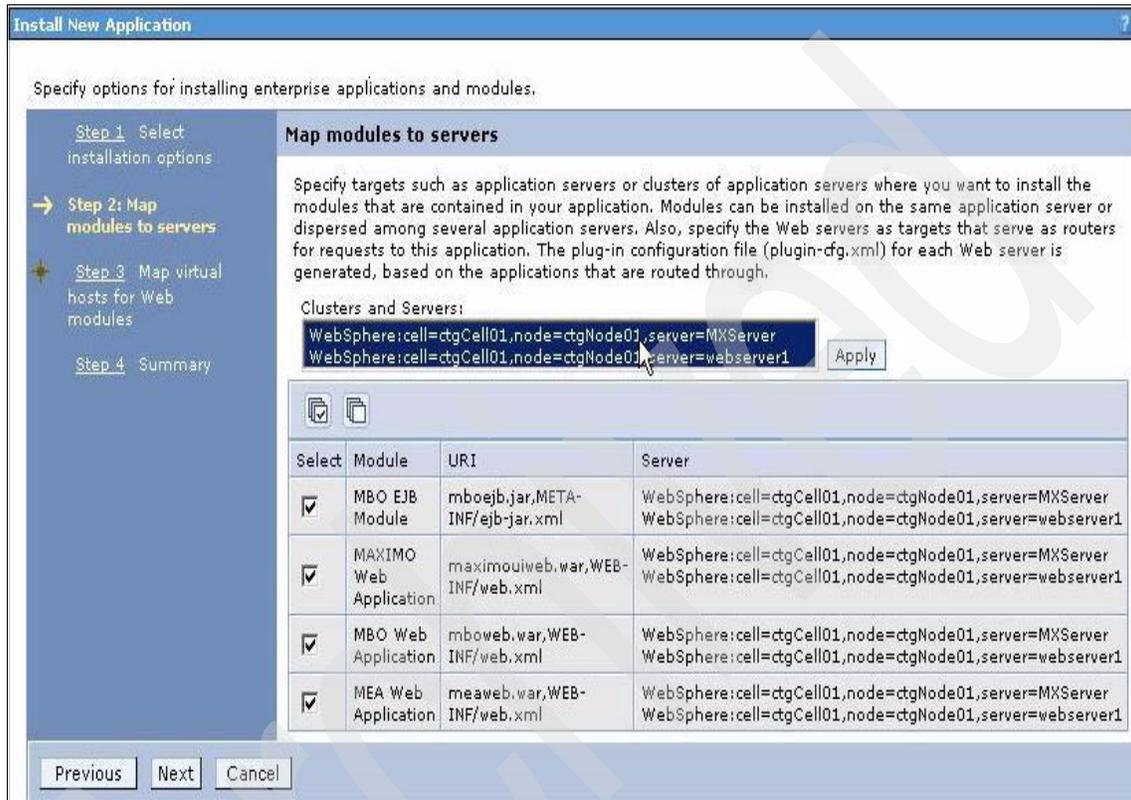

Figure A-48   Map modules to servers

12. Select and map your Maximo, Maximo Business Objects (MBO), and MEA Web applications to your virtual host, click **Next** (see Figure A-49).

Figure A-49   Map virtual hosts for Web modules

13. Verify the summary and click **Next**.

    The physical deployment takes place; allow a few minutes for the deployment to take effect.

14. After the deployment is complete, click the **Save directly to the master configuration** and restart the Maximo Asset Management application. At this point, your changes have taken effect.

Post-deployment considerations

The following section summarizes known recommendations that you can consider after the installation has been completed and Maximo Asset Management is running. We make these recommendations assuming you have installed and successfully configured Maximo Asset Management for daily usage. The recommendations are grouped into the following topics:

► Performance and system maintenance
► Usability
► Backlog management

Note: To gain a more detailed and comprehensive understanding of Maximo Asset Management, we recommend you attend Maximo Asset Management functional and technical training sessions.

Performance and system maintenance

The following are some considerations regarding system maintenance:

► Define a fixed maintenance window to perform routine maintenance on your deployment. Inform your users of the maintenance schedule. You can utilize the Bulletin Board within Maximo Asset Management to do so.

► Perform load balancing tasks on your servers and reassess your thread counts and your socket and memory settings in your application deployment on a regular basis. Consult the WebSphere Application Server and WebLogic Server documentation for further information.

► Application servers use threads to listen for and process requests from the browser. A thread represents a work item or task. Application servers (WebSphere Application Server and BEA WebLogic Server) have default settings for number of threads. WebSphere Application Server has a default minimum thread size of 5. A WebLogic Server has a default thread count of 25 (when started in production mode). You can, and often should, revise these values to improve system performance. To optimize performance, try to determine the right number of threads.

► If the thread count is too high, you have too many simultaneous requests to the CPU, and servicing the requests becomes inefficient. If the thread count is too low, the processor is not fully utilized. The queue grows, and the workload accumulates. The correct number of threads varies from site to site, but an optimal number of threads can often be found in the 40 to 55 range. The number depends on the volume of front-end transactions (user load) and back-end transactions (nonuser load).

► Use the WebSphere Application Server and WebLogic Server monitoring features to set and monitor thread usage and queue times for your application servers.

► To buffer database growth and performance, implement indexes to speed up search requests within commonly used modules - for example, Purchasing and Work Order Tracking. A database administrator can monitor growth and sizing and tune the database accordingly.

► Frequently consult Maximo Asset Management user groups and knowledge bases for issues and tips about tuning, performance, and best practices.

► Conduct regular audits and database integrity checks to find any backend database malpractices or possible bugs and errors.

► Before implementing any application change, make sure you have a change management procedure and release management protocol in place. In addition, make sure that change requests are approved by senior management and that they match your business processes. Change requests

are best released in phases in the deployment - for example, first the development environment, then training, then acceptance, and only then to the production environment. Utilize tools such as the Migration Manager to maintain identical deployment platforms.

► Review updates and new patch releases in development environments before attempting to implement these changes in acceptance and production environments.

► If you are experiencing technical issues with the Maximo Asset Management application, copy the Maximo database and restore it in a test environment. From there, enable debugging using the Logging application. Avoid running the Logging application in the production environment; logging may hinder the performance of the application server.

► Monitor for hung processes and applications. Processes and applications that are not responding use memory and can affect the performance of the client workstation. A workstation can have hung processes or applications that the user might be unaware of. Use system tools, such as the Windows Task Manager, to check for and end hung processes and applications.

► Use only one active network link. If a user has both a wireless network link and a LAN link active, it can cause system performance issues. Limit users to one active network link.

Usability

You can minimize user workload and ease their daily routine by implementing the following:

► Aim to "minimize clicks" by hiding fields wherever possible to save user effort. For example, run cron tasks to autogenerate the reordering of items, scheduling preventive maintenance, and generating reports.

► Facilitate the auto-numbering of records where possible; this provides proper and accurate referencing and ease in searchability.

► Add valid field defaults to your user interfaces to pre-populate fields that must be filled in.

► Facilitate accurate data integrity and consistency by requiring the appropriate fields to meet your business and reporting processes.

► Implement a standard, simple naming convention to populate your domains and value lists; keep the values simple, concise, and minimized.

► Use classifications to assist in creating a standardized naming convention for asset, location, item, and ticket descriptions. Doing so makes the application more workable.

► When conducting training sessions, design usable "cheat sheets" and user guides that can be easily related to and reused in the future. Maintain concise, accurate training and minimize the amount of training required.

► When customizing Maximo Asset Management, make sure to conform to the standard naming convention - for example, do not add a label with a title such as "Equipment" if the Maximo standard is "Asset." Using standard naming conventions avoids user frustration and ambiguity.

► Nurture and establish super users to support the product internally.

► Perform constant system audits to monitor usage of Maximo Asset Management. Perform integrity checks to establish data input accuracy and convey these findings to management.

► Conduct regular refresher training sessions to assess any new findings and experiences in Maximo Asset Management that can be improved. Revise these findings to match your business requirements.

► Implement updates and fixes only if they pertain to actual areas of priority within the application.

► Limit the number of users who can run reports. Limit the number of reports that users can run. During peak business hours, limit report use to reports that users need for their daily work, such as Print Work Orders and Print POs.

► Monitor user-defined queries. Most of the WHERE clauses in Maximo queries are generated by individual users on the List tabs of Maximo Asset Management applications. This powerful feature of Maximo can produce a lot of inefficient SQL. You can improve the ease of use and convenience for users by setting the appropriate search types for database columns. Using appropriate search types also reduces the load on the database.

► In Maximo Asset Management V7.1, you can configure search options. You can change the default search type of WILDCARD to TEXT or EXACT. TEXT and EXACT searches can use indexes. You specify the search type for a database column in the Database Configuration application. You also can change the search type for groups of columns.

► You can prioritize Maximo Asset Management traffic over standard Web traffic. Contact your network administrator for details about prioritizing Maximo traffic.

Backlog management

Consider the following techniques to manage backlogs within Maximo Asset Management. You want to manage backlogs that can hinder the usability and the performance of your deployment:

► Implement reports and KPIs to monitor transactions that have remained unchanged within due course of time. Set all key performance indicators (KPIs) to retrieve their data from the KPI table. Use cron tasks to run KPI queries at reasonable intervals.

► Close off records that are completed or inactive.

► Monitor items that are bound by time frequencies - for example, assets with outdated meter readings or incorrect averages might have preventive maintenance schedule assigned to them. If the meter readings have not been entered, the preventive maintenance schedules might not generate; on the other hand, if the meters are incorrect, the preventive maintenance schedules may generate too often.

► Conduct workshops and user sessions to brief users on backlogs and escalate these backlogs if necessary.

C

Summary of modules and applications

This appendix provides a summary of Maximo Asset Management modules and applications.

Important: This book was written before Maximo Asset Management V7.1 had been finalized. Therefore, the official product documentation should be used to validate the list of modules and applications.

Table C-1 summarizes the modules and applications that make up Maximo Asset Management.

Table C-1   Maximo Asset Management modules and applications

Module and sub-module	Application	Description
ADMINISTRATION	Organizations	Set up the organizations and sites to be used within Maximo.
	Classifications	Create classifications and establish classification hierarchies for Items, Assets, Locations, Work Orders, etc.
	Bulletin Board	Create, post, and view messages as well as broadcast information to users of the Maximo system.
	Communication Templates	Create and manage generic communication templates that Maximo users can leverage to standardize frequently used e-mail communications (also known as notifications).
	Calendars	Indicate working time for equipment, craft, and labor records for an organization and its associated sites.
	Sets	Create a framework for sharing item and company (vendor) data across multiple organizations.
	Work View	Make queries available for display in the Result Set portlet of a user's Start Center.
	CI Types	Specify which configuration item types (CI types) you want to use when you import actual configuration items from the IBM Tivoli Application Dependency Discovery Manager product.
	Conditional Expression Manager	Create and maintain a library of conditions. In other applications, such as Application Designer and Security Groups, you select from predefined conditions to set up conditional behavior.
REPORTING	KPI Manager	Create key performance indicators (KPIs) to track critical performance variables over time.
	Report Administration	Create reports, generate and preview request pages, add parameters, display reports as toolbar icons, e-mail reports, or specify a schedule for running a reports.
RESOURCES	People	Maintain records of people.

Module and sub-module	Application	Description
	Person Group	Maintain person groups. A person group consists of people who may or may not be workers.
	Crafts	Define and maintain craft records.
	Labor	Define and maintain labor records.
	Qualifications	Create qualifications and certification requirements for qualifications.
ASSETS	Assets	Store asset numbers and corresponding information such as parent, location, vendor, up and down status, and maintenance costs for each asset.
	Locations	Enter and track locations for assets, and organize these locations into logical hierarchical systems or network systems.
	Meters	Add or modify meter definitions. Meter definitions include names for the meters as well as sets of attributes that describe the meters.
	Meter Groups	Define a logical grouping of meters to exist in a meter group. Meter groups represent a collection of meters that are used together multiple times.
	Condition Monitoring	Define unlimited measurement points for assets, and specify alarm limits and associated work to be performed after reaching those limits. Create and view measurement-point records to define acceptable meter readings for a characteristic or gauge type of meter on an asset or location.
	Features	Define the physical features of the linear assets that you manage with the Maximo Linear Asset Management product option. A physical feature is a physical object, such as a guard rail or mile marker, that you associate with one or more linear assets.
	Failure Codes	Build and display failure hierarchies, which help you construct accurate histories of the failures that affect your assets and operating locations.
CHANGE	Activities and Tasks	Plan, review, and manage activities and tasks. When you create an activity, you initiate the work process and create a historical record of work being performed.

Module and sub-module	Application	Description
CONTRACTS	Purchase Contracts	Create, modify, and view contracts with outside vendors.
	Lease Rental Contracts	Define the overall terms and conditions of the lease or rental agreement between a vendor and a client regarding one or more assets.
	Labor Rate Contracts	Define multiple labor rates for specific crafts, skills, and optionally, labor records. Within the Labor Rate Contracts application, you can manage outside labor and the corresponding rates.
	Warranty Contracts	Maintain one or more assets for an outside service provider for a fixed fee or for regularly scheduled payment over a time period. Track warranty information for multiple assets or locations by time or meter.
	Master Contracts	Associate many contract types for a particular vendor. A Master Contract defines the relationship with a vendor and contains terms and conditions that apply to the contracts created and listed under it.
FINANCIAL	Currency Codes	Define currency codes and specify which codes can be used in Maximo Asset Management.
	Exchange Rates	Set up exchange rates for converting currencies.
	Chart of Accounts	Establish general ledger account fields with definitions equivalent to those used with the rest of your financial data processing system.
	Cost Management	Generate project cost information to track the financial resources that are required to complete a project and manage budgets more effectively.
IT INFRASTRUCTURE	Configuration Items	Define, create, and manage relationships among configuration items according to relationship rules that you specify in the Relationships application. A configuration item is any component of an information technology infrastructure that is under the control of configuration management.

Module and sub-module	Application	Description
	Relationships	Create definitions and rules for relationships. A relationship describes the dependency or connectivity between configuration items using basic terms such as "runs on," "installed on," and "contained." You use these relationships in the Configuration Items application to associate configuration items with each other.
	Collections	Group configuration items, assets, and locations so that they are easier to access and handle in other applications. Instead of selecting from a list of all configuration items, assets, or locations, you can specify a collection to obtain a list of records in the collection.
INTEGRATION	Object Structures	Create, view, modify, and manage the processing logic of an object structure. An object structure is the common data layer that the Integration Framework uses for all outbound and inbound application data processing. An object structure consists of one or more sub-records that develop XML content from a particular object.
	Publish Channels	Create and manage Publish Channel records that are used as the pipelines for exporting data from the Integration Framework to an external system.
	Invocation Channels	Create invocation channel records and, through service oriented architecture (SOA); enable the use of external services for processing data from multiple data sources.
	Enterprise Services	Create enterprise service records that are used as the pipelines for importing data from an external system to the Integration Framework.
	Web Services Library	Create and manage Web services that can be used by external applications to query or send transactions to the Integration Framework. You can also generate schema and Web Service Description Language (WSDL) files for any Web service that you deploy.

Module and sub-module	Application	Description
	End Points	Create and manage endpoint records. An endpoint identifies a location and the processing logic of data publication and service invocations. By using the defined endpoint handler, you can identify how to route outbound data to a specific location. You can also define which data format the Integration Framework or Deployment Manager components will use through the handler.
	External Systems	Create and manage external systems. Any business application that sends data to the system or receives data from the system is an external system. You can use External Systems to synchronize external data through an endpoint (location), and synchronize internal data through an external source.
	Logical Management Operations	Create and manage logical management operations. A logical management operation is the common data layer that defines an action that the system takes on an operational management product.
	Integration Modules	Create and manage Integration modules. The Integration Modules application provides a mechanism for a process management product - such as a Change or Release - to call an external operational management product.
	Launch in Context	Create and manage launch entries that open, in the same or a different browser session, an application that is external to the system.
	Message Tracking	Track and view the processing history of queue-based inbound (Enterprise Services) and queue-based outbound (Publish Channels) messages. When you enable message tracking, the Integration Framework writes processed messages to the Maximo database.
	Message Reprocessing	Manage and view integration transaction messages that are flagged with an error.
INVENTORY	Item Master	Define items that stocked in your Storerooms. You group these items in an item set, which can then be shared by the organizations using that item set.
	Service Items	Define and manage purchased services.

Module and sub-module	Application	Description
	Tools	Manage information about the tools used to perform work. Tools are typically non-consumable items for which you charge an hourly rate.
	Stocked Tools	Manage existing tools in Storerooms.
	Inventory	Enter, display, and update information about each inventory item.
	Issues and Transfers	Issue or transfer items from Storerooms, or return items to Storerooms.
	Condition Codes	Create and maintain a master list of condition codes for a particular item set.
	Storerooms	Add and maintain information about Storeroom locations, as well as view the items stocked within a Storeroom.
PLANNING	Job Plans	Create a detailed description of how a job is to be performed.
	Routes	List related work assets that are considered "stops" along an inspection or maintenance route.
Safety	Hazards	Define hazards that exist in the workplace and related safety precautions.
	Precautions	Define safety precautions to prevent hazards in the workplace.
	Lock Out/Tag Out	Create a detailed description of how to take work assets out of service or how to place them back in service, to ensure a safe work environment.
	Safety Plans	Create a detailed plan of how to safely service assets or locations.
PREVENTIVE MAINTENANCE	Preventive Maintenance	Create, modify, and view preventive maintenance plans for work assets. PM records are templates for work orders or for other PMs.
	Master PM	Create and modify master PMs, which are templates for other PM records.
PURCHASING	Purchase Requisitions	Ask the purchasing department to order materials or services.
	Purchase Orders	Purchase materials or services from an internal supplier or an external vendor.

Module and sub-module	Application	Description
	Receiving	Receive materials into inventory and record the receipt of services.
	Invoices	Record invoices and match them against purchase orders and receipts for approval.
	Request for Quotations	Request and manage vendor quotations.
	Companies	Manage data about manufacturers, vendors, and other companies that do business with you.
	Company Master	Create company master records that belong to a particular company set.
	Search Catalogs	Enables "punch-out" capability from Work Orders and Desktop Requisitions.
RELEASE	Activities and Tasks	Plan, review, and manage activities and tasks. When you create an activity, you initiate the work process and create a historical record of work being performed.
SECURITY	Security Groups	Grant access to sites, applications, and menu options. A user is assigned to one or more groups to gain access to the system.
	Users	Add and manage Maximo users.
SELF SERVICE		
Service Requests	Create Service Request	Self-service users use this application to create new service requests.
	View Service Request	Self-service users can view existing service requests.
Desktop Requisitions	Create Requisition	Self-service users create new purchase requests.
	View Requisition	Self-service users view purchase requests.
	View Templates	Administrative users design purchase requests from existing templates.
	View Drafts	Self-service users can view draft purchase requests.
SERVICE DESK	Activities and Tasks	Plan, review, and manage activities and tasks. When you create an activity, you initiate the work process and create a historical record of work being performed.

Module and sub-module	Application	Description
	Service Requests	Create, view, and resolve service requests from clients.
	Ticket Templates	Create and manage generic ticket templates that Service Desk environments can leverage to standardize common or high-volume service requests, incidents, or problems.
SYSTEM CONFIGURATION		
Platform Configuration	Actions	Manage the administrative functions of creating actions and action groups within Escalations.
	Roles	Manage roles within Maximo Asset Management.
	Communication Templates	Create and manage generic communication templates that users can leverage to standardize frequently used e-mail communications (also known as notifications).
	Database Configuration	Create or modify the objects and attributes used by Maximo Asset Management applications.
	Application Designer	Create new applications (clones and custom applications) or tailor the pages of existing applications.
	Escalations	Automatically monitor critical processes across your enterprise. The primary goal of Escalation Management is to ensure that critical tasks are completed on time, such as those defined in service-level agreements (SLAs).
	Cron Task Setup	Manage cron tasks. Cron tasks are behind-the-scene jobs set to run automatically and on a fixed schedule.
	Domains	Maintain lists of defined values that appear in drop-down lists (sometimes referred to as value lists).
	Logging	Manage log settings and configure log files.
	System Properties	Manage system properties and their values used by various product components.

Module and sub-module	Application	Description
	Web Services	Create, modify, and delete Web services. You also can generate schema and Web Service Description Language (WSDL) files for any Web service that you deploy. External applications can use Web services to query or to send transactions to the Integration Framework.
	Workflow Administrator	View and modify assignments within the workflow, escalation, and service-level agreement processes.
	Workflow Designer	Use this graphical application to create a series of decision paths for records to flow through, called workflow process.
	E-mail Listener	Receive and process incoming e-mail messages. This application can monitor multiple e-mail accounts to retrieve messages, and it supports embedded and normal message attachments.
	Launch in Context	Create and manage launch entries that open, in the same or a different browser session, an application that is external to the system.
	Object Structures	Create, view, modify, and manage the processing logic of an object structure. An object structure is the common data layer that the Integration Framework uses for all outbound and inbound application data processing. An object structure consists of one or more sub-records that develops their XML content from a particular object.
Migration	Migration Manager	Define, create, distribute, and deploy packages. Packages are used to transfer and deploy the many configuration changes possible with the Maximo Asset Management configuration tool set from one environment to another (that is, from a development environment, to test environment, to production environment).
	Migration Groups	Create groups of configuration objects and link related (dependent) groups to the objects that you create. You group configuration objects to ensure that all related configuration data is collected from source environments and distributed to target environments.

Module and sub-module	Application	Description
	Object Structures	Create, view, modify, and manage the processing logic of an object structure. An object structure is the common data layer that the Integration Framework uses for all outbound and inbound application data processing. An object structure consists of one or more sub-records that develops their XML content from a particular object.
TASK MANAGEMENT	Activities and Tasks	Plan, review, and manage activities and tasks. When you create an activity, you initiate the work process and create a historical record of work being performed.
WORK ORDERS	Work Order Tracking	Plan, review, and approve work orders for assets and locations.
	Labor Reporting	Report the type and total number of hours of work that was performed by external contractors or internal employees.
	Quick Reporting	Report work on open work orders or small jobs.
	Activities and Tasks	Plan, review, and manage activities that can initiate the maintenance process and create a historical record of work being performed.
	Assignment Manager	Dispatch urgent work and schedule labor to planned work requirements.
	Service Requests	Create, view, and resolve service requests from clients.
START CENTER	Layout and Configuration	Administrator users can modify and configure the layout of the portlets displayed on the Start Center.
	All Custom Applications	Enables access to all user-created custom applications.
	Favorite Application Setup	Administrators can define and edit the list of applications displayed in a Favorite Application portlet.
	Forgotten Password	E-mail users their current passwords from the Login application.
	Inbox/Assignments Setup	Administrators can define and edit which columns are displayed in the Workflow assignments inbox on the Start Center.

Module and sub-module	Application	Description
	KPI Graph Setup	Administrators can define and edit a KPI graph-style portlet to display on the Start Center.
	KPI List Setup	Administrators can define and edit a KPI list-style portlet to display on the Start Center.
	Change Password	Change a user's password.
	Quick Insert Setup	Administrators can define and edit the list of applications displayed in a Quick Insert portlet.
	Result Set Setup	Administrators can define and edit the query used and columns displayed in a Result Set portlet displayed on the Start Center.
	Start Center	The Start Center is an initial page displayed once users log on to Maximo Asset Management. It provides a dashboard-like starting point that can display various portlets including KPIs, Result Sets, Quick Insert, Favorite Applications, and Inbox/Assignments.
	User Self Registration	New users can register themselves as Maximo Asset Management users with a temporary password.

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

For information about ordering these publications, see "How to get Redbooks" on page 319. Note that some of the documents referenced here may be available in softcopy only.

► *WebSphere Application Server V6 System Management & Configuration Handbook*, SG24-6451

► *Maximo Asset Management Essentials V7.1 Implementer's Guide*, SG24-7645

Online resources

These Web sites are also relevant as further information sources:

► Maximo Asset Management online product documentation

    http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.mam.doc_7.1/mam_welcome.htm

► *Best Practices for Maximo System Performance*, a white paper available at:

    http://www-1.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&uid=swg27011545&loc=en_US

How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks, at this Web site:

    ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

IBM

Redbooks

Deployment Guide Series: Maximo Asset Management V7.1

(0.5" spine)
0.475" <-> 0.875"
250 <-> 459 pages

Deployment Guide Series: Maximo Asset Management V7.1

IBM®

Redbooks®

Learn about Maximo Asset Management V7.1

Plan for a deployment

Study sample usage scenarios

This IBM Redbooks publication is for IBM Business Partners, field personnel, and clients who want to plan a deployment of IBM Maximo Asset Management V7.1. We provide information to assist in the deployment and initial configuration of the environment, but we do not cover general product usage. This book assumes the reader has access to product documentation and some knowledge of previous versions of Maximo Asset Management or other Maximo products that utilize the same underlying framework.

Our intention is not to replace or reproduce the product documentation, but rather to provide useful, supplementary information to assist the reader in understanding deployment planning and processes for Maximo Asset Management. Some content in this book provides information helpful to those who are just becoming familiar with the Maximo product set. In other sections, we assume the reader has deeper technical knowledge and experience.

In most solution deployments, a team of individuals with different skill levels is required to successfully plan and deploy Maximo Asset Management. For various members of the deployment team, different sections of this book are valuable resources for performing their tasks.