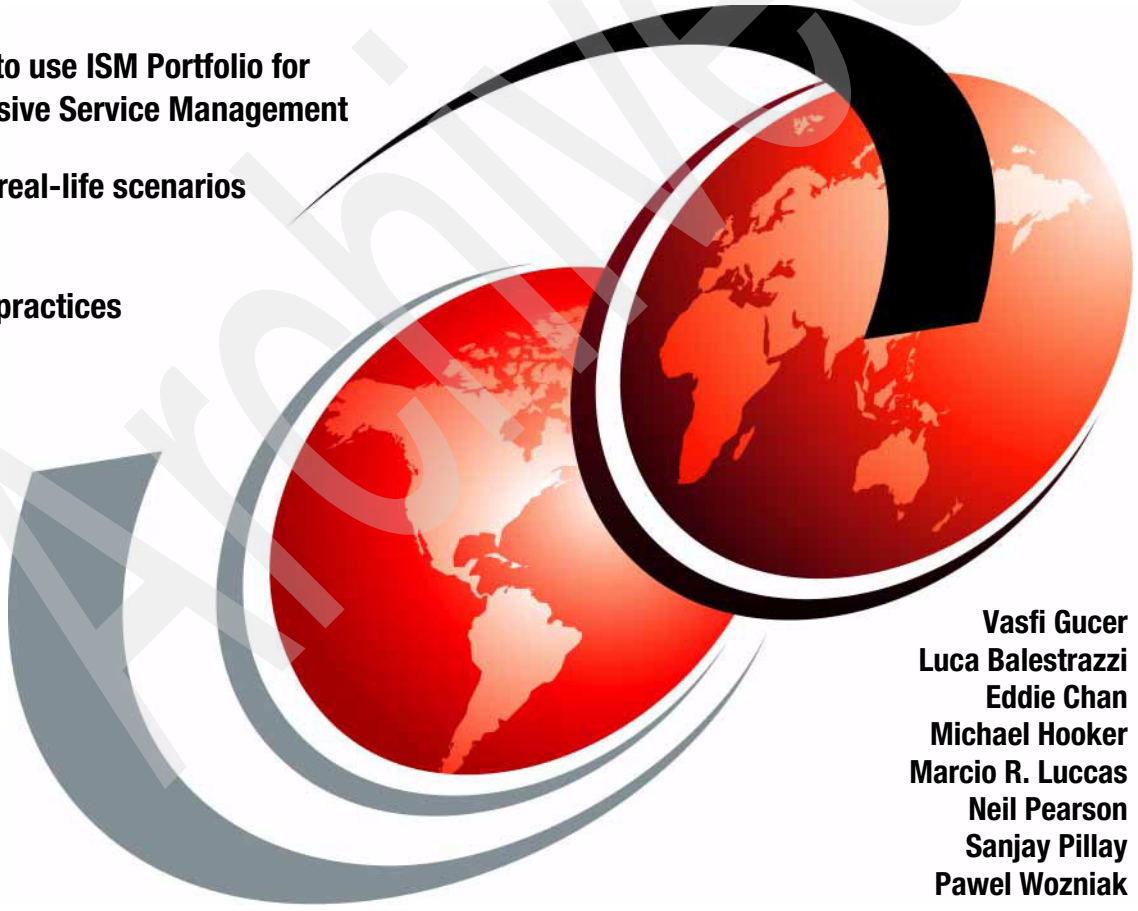
Tivoli software

IBM

End-to-End Service Management Using IBM Service Management Portfolio

- Learn how to use ISM Portfolio for comprehensive Service Management
- Implement real-life scenarios
- Learn best practices



Vasfi Gucer
Luca Balestrazzi
Eddie Chan
Michael Hooker
Marcio R. Luccas
Neil Pearson
Sanjay Pillay
Pawel Wozniak

Redbooks

ibm.com/redbooks

International Technical Support Organization

End-to-End Service Management Using IBM Service Management Portfolio

February 2009

SG24-7677-00

Note: Before using this information and the product it supports, read the information in “Notices” on page xv.

First Edition (February 2009)

This edition applies to IBM Tivoli Service Request Manager 7.1, IBM Tivoli Change and Configuration Management Database V7.1.1, and IBM Tivoli Release Process Manager V7.1.1.

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 about the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

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

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

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

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

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

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

COPYRIGHT LICENSE:

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

Trademarks

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

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

AIX®
Build Forge®
CICS®
DB2®
GDPS®
Geographically Dispersed Parallel Sysplex™
HACMP™
IBM Process Reference Model for IT™
IBM®
Informix®
Lotus Notes®
Lotus®
Maximo®
Netcool®
Netfinity®
NetView®
Notes®
OMEGAMON®
OS/390®
Parallel Sysplex®
Purify®
Quickr™
Rational®
Redbooks®
Redbooks (logo) ®
Sametime®
System Storage™
System z®
SystemView®
Tivoli Enterprise Console®
Tivoli®
TotalStorage®
WebSphere®
z/OS®

The following terms are trademarks of other companies:

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.

J2EE, Java, JavaScript, JDBC, JVM, Power Management, Solaris, Streamline, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel, 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

IBM® Tivoli® Service Request Manager, IBM Tivoli Change and Configuration Management Database, IBM Tivoli Asset Management for IT, IBM Tivoli Release Process Manager, and IBM Tivoli Business Continuity Process Manager are key components of the IBM Service Management strategy. This IBM Redbooks® publication presents scenarios of the combined usage of these products for implementing a complete, end-to-end Service Management solution.

We start by introducing Information Technology Infrastructure Library (ITIL®) based Service Management, followed by a discussion about how to design your ITIL-based Service Management solution using the Tivoli products.

We have included several scenarios that will help you understand how these products work together in a real customer environment.

This book is a major reference for IT specialists working on implementing ITIL-based Service Management solutions using the Tivoli products. IT managers, IT architects, and pre-sales technical specialists will also benefit from the information presented in this book.

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.

Vasfi Gucer is an IBM Certified Consultant IT Specialist at the ITSO Austin Center. He started working for the ITSO in January 1999, and has been writing Redbooks publications ever since. He has more than 15 years of experience in teaching and implementing Systems Management, networking hardware, and distributed platform software. He has worked on various Tivoli customer projects as a Systems Architect and Consultant. Vasfi is also a Certified Tivoli Consultant and an ITIL V2/V3 Manager.

Luca Balestrazzi is a Certified IBM IT Architect in Global Technology Services in Italy. He has five years of experience in the RNSL Rome Laboratory in service support, two years of experience in service support for SystemView® and Netfinity® products, and 10 years of experience with Tivoli Area of Network and Systems Management products. He also has experience with IBM Service Management (ISM) family products. He has a degree in Economics and is ITIL

v3 certified. Luca is working as a Delivery IT Architect on Change and Configuration Management Database (CCMDB) Projects in Italy.

Eddie Chan is a Certified IT Architect in IBM Hong Kong. He has 20 years of IT experience covering Systems Management for mainframe and open systems, IT strategic planning, enterprise IT architecture design, IT optimization, and strategic IT project implementation. He has used the IBM Process Reference Model for IT™ to design the IT processes for clients, and received much practical experience in providing Service Management for clients. He is currently the lead Technical Solution Architect for IBM Strategic Outsourcing (SO) in Hong Kong and provides infrastructure solution design for SO customers.

Michael Hooker joined IBM in 1997 as an Enterprise IT Specialist in the Tivoli brand. He is currently a Service Management Solution Architect Specialist in the U.S. West Region. He has over 22 of experience in the IT profession, and a Bachelor of Science degree in Computational Applied Mathematics. He is an IBM Certified IT Specialist and is ITIL Certified. He has covered the large IBM accounts for several years. His responsibilities have included leading the development of system management strategy, acting as a Tivoli Adviser to customers, and assisting with high-level designs for large-scale customers and complex opportunities. He is very knowledgeable about the Tivoli portfolio, Enterprise System management, and architecting solutions with Tivoli and other vendors.

Marcio R. Luccas is an IT Service Management Consultant in IBM Brazil. He has more than 12 years of experience working in Information Technology (IT) services, with seven years of experience as an IT Specialist focusing on Asset Management solutions and three years of experience as an IT Project Manager. He has a bachelor's degree in mathematics, focusing on informatics, with a post graduate degree in information technology. He is an Peregrine Implementation Consultant certified in ServiceCenter and is ITIL Foundations certified and an instructor.

Sanjay Pillay is ITIL certified, and a SUN Certified Enterprise Architect for J2EE™ technology and Certified Business Component Developer for J2EE. He has a bachelor's degree in electronics. He has 16 years of IT experience in the banking, insurance, Service Management, and inventory and Asset Management sectors in the United States, 12 years of which have been with IBM product development. He is currently a key development member of CCMDB Change and Release Process Managers in Austin, TX, and is on the Advanced Deployment Program assisting new customers with their implementation of CCMDB.

Neil Pearson is a Senior IT Specialist working in technical sales at IBM United Kingdom. He has over 17 years of IT industry experience, mostly in Enterprise Systems Management. He joined Tivoli Systems in 1997 as a Technical

Consultant and worked in the IBM Software Group Lab Services until 2007, during which time he successfully engaged with over 120 customers worldwide. He is currently responsible for the early product introduction of Tivoli's ISM portfolio. His experience includes Tivoli Business Automation and the ISM family of products. He is ITIL certified and is best known worldwide for his skills in SAP® Systems Management.

Pawel Wozniak is an Maximo® Technical Consultant at IBM UK. He has four years of experience as a Technical Consultant in Maximo solutions, focusing recently on CCMDB, TSRM, and the release process. He has a degree in computer science with a focus on the network systems, with a post graduate degree in GIS.

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

Arzu Gucer
International Technical Support Organization, Austin Center

Pandian Athirajan, John Cook, Scott Dickerson, Steve Hochstetler, Melanie Gurda, Edson Manoel, CJ Paul, Tom Tuning, Edward Whitehead
IBM USA

Leucir Marin Junior
IBM Brazil

Georg Ochs
IBM Germany

The team would like to express special thanks to Jeremias Werner from IBM Germany for his contributions to the IBM Tivoli Business Continuity Process Manager scenario described in Chapter 8, “Scenario 4: IT Service Continuity Management using IBM Tivoli Business Continuity Process Manager” on page 211.

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

1

Introduction to Service Management

In this chapter we introduce Service Management and discuss key concepts such as Information Technology Infrastructure Library (ITIL), IBM Service Management, and IBM Tivoli Unified Process (ITUP).

This chapter has the following sections:

1.1 Implementing processes based on ITIL

The Information Technology Infrastructure Library is a set of concepts and policies for managing information technology (IT) infrastructure, development, and operations. ITIL gives a detailed description of a number of important IT practices with comprehensive check lists, tasks, and procedures that can be tailored to any IT organization.

ITIL is a framework of best practices, not a methodology, as it only describes what needs to be done. ITIL does not provide guidance on how to implement the processes, so each company chooses the best way to fit ITIL to its requirements.

A key mind set when implementing processes based on ITIL is *adopt and adapt* is to *adopt* ITIL as a common language and reference point for IT Service Management and *adapt* ITIL best practices to achieve business objectives.

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

The Service Management processes selection should be done carefully, taking into consideration the relationship among all processes and not only the cost perspective and implementation complexity of individual processes.

A successful implementation of IT Service Management should consider:

- Aligning with business needs (that is, business driven, not technology driven).
- Improving staff awareness about business goals.
- Adapting to the culture of the organization. This adaptation should be done when defining the roles, responsibilities, tools, processes, procedures, tasks, and so on. After IT Service Management is implemented, it should be rigorously followed.
- Having processes easily changed as necessary.
- Having its processes clearly defined, documented, and available.
- Having its main processes integrated with each other.
- Integrating with external suppliers.

- Properly training and communicating to all people who will use or provide IT services.
- Having its inputs measurable and repeatable.
- Having IT process tool supported and customized to fit the processes defined.
- Having clearly measurable and repeatable key performance indicators.
- A successful IT Service Management implementation should result in improved IT customer satisfaction, better resource utilization, and improvement of customer perception of IT service quality.

1.2 ITIL V3 and components

ITIL V3 is the evolution of service life-cycle management. Figure 1-1 represents the five components of ITIL V3.

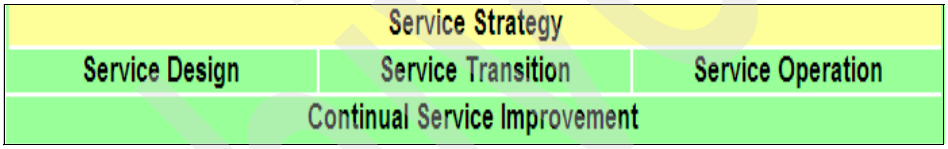


Figure 1-1 ITIL V3 components

ITIL V3 components are:

- Service Strategy

  The Service Strategy ITIL V3 book describes how business and IT align. This includes keeping the service life cycle focused on the needs of the business.

  The Service Strategy processes are:

  - Strategy generation
  - Adaptive processes for customers, services, and strategies
  - IT Financial Management
  - Service portfolio
  - Demand Management
  - Risk Management

- Service Design

  The Service Design, ITIL, Version 3 book discusses designing of IT services conforming to best practice, and including design of architecture, processes, policies, documentation, and allowing for future business requirements.

The Service Design processes are:

- Capacity Management
- IT policies, architecture, portfolios, and service models
- Information Security Management
- IT Service Continuity Management
- Service re-design
- Availability Management
- Supplier Management

- Service Transition

  The Service Transition, ITIL, Version 3 book describes how to manage and control changes to the infrastructure.

  The Service Transition processes are:

  - Service Asset and Configuration Management
  - Quality assurance
  - Service release and deployment planning
  - Acquire, build, test release
  - Deployment, decommission, and transfer
  - Knowledge Management and service knowledge system
  - Service release, acceptance, test, and pilot
  - Change Management
  - Performance and risk evaluation
  - Testing

- Service Operation

  The Service Operation ITIL, Version 3 book discusses how to achieve the delivery of agreed-upon levels of services both to users and customers.

  The Service Operation processes are:

  - Operation of services from end-to-end
  - Monitoring and management of events
  - Fulfillment of service requests monitoring and Event Management
  - Problem Management
  - Request fulfillment

- Continual Service Improvement

  The goal of Continual Service Improvement is to align and realign IT services to changing business needs by identifying and implementing improvements to the IT services that support the business processes.

  The Continual Service Improvement processes are:

  - Service improvement
  - Measurement and control
  - Process assessment and analysis
  - Service assessment and analysis
  - The business case for improvement
  - Service measurement

1.3 What is IT Service Management

IT service is a discipline for managing information technology systems, focusing on the customer‘s needs to improve business.

IT Service Management provides for the effective and efficient delivery of IT services in support of changing business needs, providing faster guidance, and addressing technology updates to support business challenges.

Implementing IT Service Management requires the optimal intersection of people, process, information, and technology, as shown in Figure 1-2.

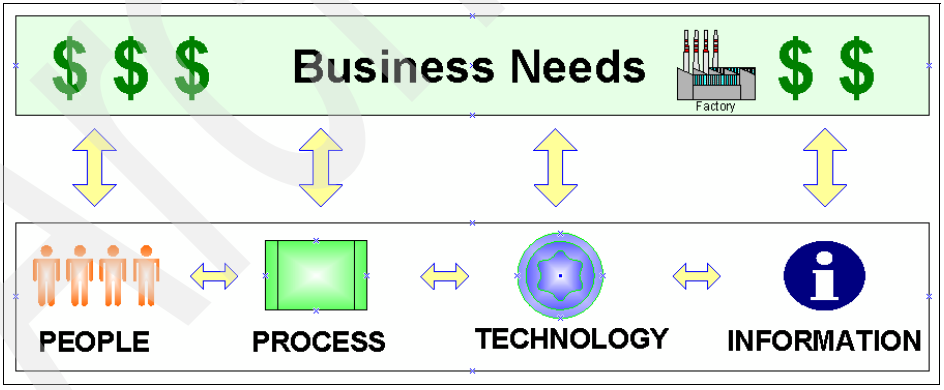


Figure 1-2 IT organization with Service Management

1.4 ITIL-based Service Management

ITIL based Service Management represents *the state of the art* of management information technology focus on business improvement. It shows us how the relationship between all components, such as people, technology, process, information, and business needs work together to support business challenges and improvement.

Figure 1-3 shows an IT organization environment with ITIL-based Service Management implemented.

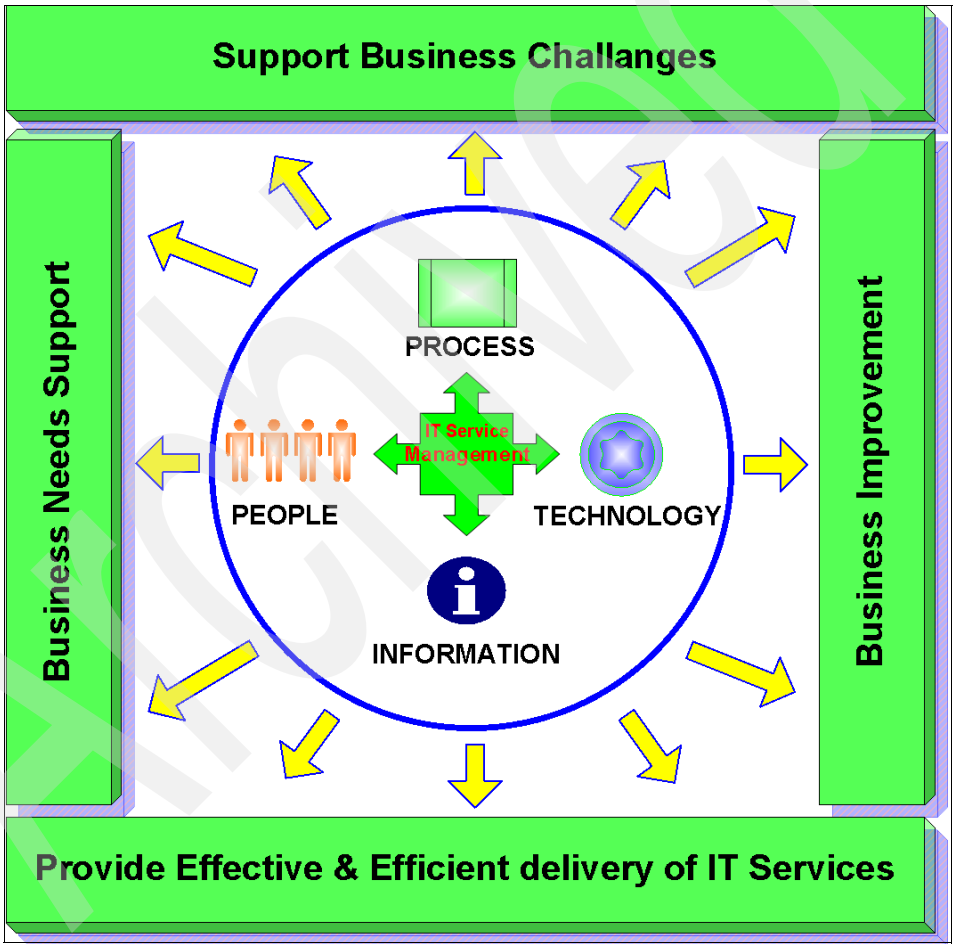


Figure 1-3 IT organization with ITIL-based Service Management Implemented

When all of these components come together, they can make IT more efficient and effective.

1.5 IBM and ITIL

IBM initially contributed to ITIL with its Systems Management concept *yellowbooks* and continues to contribute as a developer, reviewer, and user of ITIL.

IBM contributed in many ways to ITIL Version 2, including authoring, quality reviews, project management, and additional support through the IT Service Management Forum. The focus of Version 2 was on process management practices required to enable Service Management. The ITIL service support and delivery publications contain significant contributions from IBM.

The ITIL application management book, co-written by authors from IBM and other companies, is the basis for the life-cycle concept in ITIL Version 3. It lays the basic groundwork for how to integrate Service Management practices throughout the solution life cycle.

IBM supports the development of updates and refreshes to industry-accepted best practices, including supporting the ITIL Advisory Group through quality reviews and other briefings. Thought leaders also serve on the ITIL Advisory Group and other working groups to contribute as the need arises. From a strategic outsourcing perspective, ITIL is requested by many IBM clients all around the globe. Companies that are implementing improvements to their Service Management capabilities consider ITIL a good place to start.

IBM Tivoli Service Request Manager is aligned with ITIL best practices to support ITSM processes. Built with support for incident and problem management, change and Release Management, and Service-Level Management, it is a part of a single platform that combines asset and services management. IBM implements processes based on ITIL through IBM Tivoli Unified Process (ITUP).

1.6 What is IBM Tivoli Unified Process

IBM Tivoli Unified Process *is a roadmap for delivering ITIL-based IT Service Management* using existing Tivoli and IBM solutions. ITUP links actual product names and capabilities with ITIL-defined roles, responsibilities, and processes, and complements IBM Rational® Unified Process (a logical method of application development) to provide a mechanism and a philosophy for

customers to align their IT organization and processes along business service guidelines, and develop and implement applications with additional manageability factors built in to take full advantage of these new concepts.

Examples that you'll find in ITUP are:

- Workflows.
- Roles.
- Information (work products by name). Also not described in ITIL.
- Products (tools) that help implement aspects of the process.

Figure 1-4 shows the ITUP components.

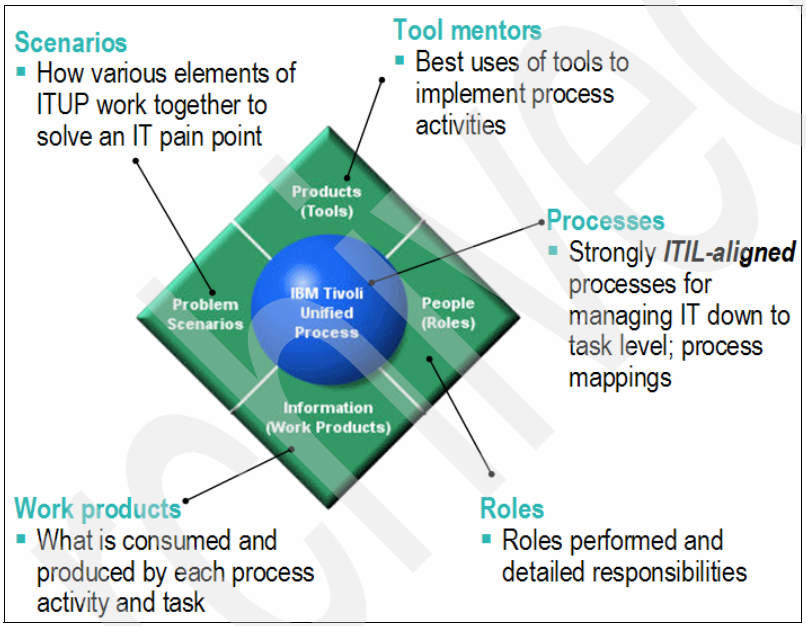


Figure 1-4 ITUP overview

For more information go to:

http://www-01.ibm.com/software/tivoli/governance/servicemanagement/itup/tool.html

That site provides a brief overview of components and features and how they interact with each other in the overall IBM IT Service Management strategy. ITUP is considered by IBM to be a definitive guide on how to implement ITIL in a modular yet comprehensive fashion using IBM solutions available today.

ITUP Composer

IBM Tivoli Unified Process Composer (ITUPC) is a tool that allows for an implementation of the ITUP framework. ITUP Composer is the product version of ITUP. It is an ideal starting point for organizations looking to implement IT Service Management best practices and document their operational model.

You can then use the Composer tool to customize, extend, and publish content to document your organization's operational processes.

ITUPC is shipped with Change and Configuration Management Database (CCMDB) Version 7.1 as IBM Rational Method Composer (RMC). RMC is the tool that enables the development, customizing, and publishing of methods and processes.

Table 1-1 summarizes the differences between ITUP and ITUP Composer.

Table 1-1 Differences between ITUP and ITUP Composer

Feature	ITUP	ITUP Composer
Industry best practices	Y	Y
Process-level information	Y	Y
Activity-level information	Y	Y
Tool use guidance	Y	Y
Task-level information	Y	Y
Content customization		Y
Content creation		Y
Content publishing		Y

Figure 1-5 shows ITUP and ITUP Composer components.

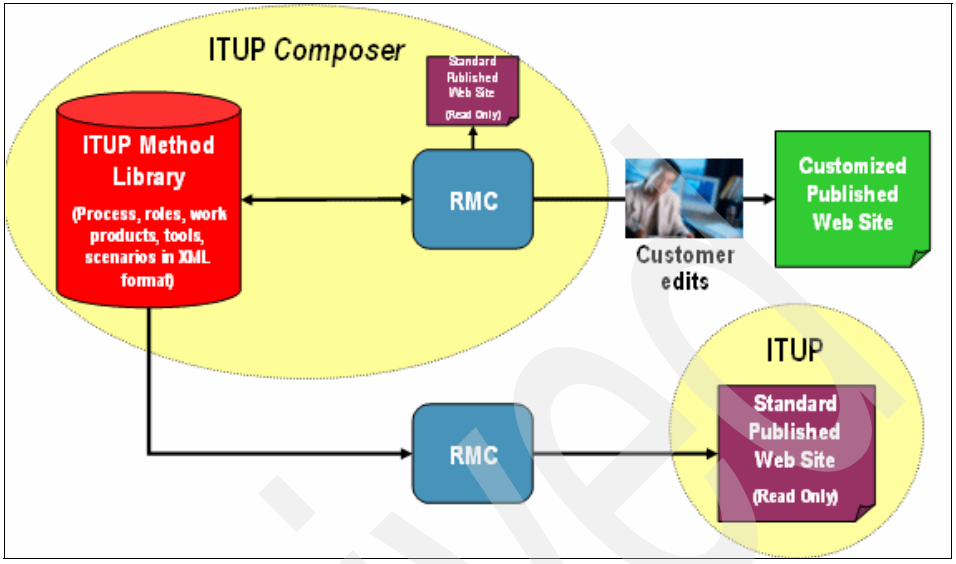


Figure 1-5 ITUP and ITUP Composer components Illustrated

ITUPC enables you to easily use the out-of-the-box process model as the foundation for an operational model that *meets your organization's requirements* and can serve as a unifying guide for all IT staff.

1.7 Benefits of using ITUPC

The benefits of using ITUPC are:

- It provides detailed documentation of IT Service Management processes based on industry best practices that speeds up the customization process.
- It simplifies maintenance with local copies of process guides that are soon out of date.
- It is easy to access, as it can be published in HTML format.
- It is a good candidate to become the single source of published ITSM process definitions and related documentation.
- It can be used with version control systems like CVS and Rational Clear case.
- ITUPC is linked with process implementation tools like Tivoli CCMDB.

1.8 What is IBM Service Management

IBM Service Management encompasses the management processes, tactics, and best practices needed to deliver business services.

IBM Service Management is about developing, deploying, and managing services that help reduce IT and operations costs through automated processes and more effectively manage compliance.

IBM Service Management lets you pull critical components, such as *people, processes, information, and technology,* together with an array of tightly integrated solutions that can be viewed as three interconnected layers:

- IBM Process Management
- IBM Services Management platform
- IBM Operational Management

These solutions are based on IBM and industry best practices, such as the IT infrastructure Library, Control Objectives for Information and related technology (COBIT), and enhanced Telecom Operations Map (eTOM), helping users to ensure that IT and operational processes are consistently designed, automated, and executed and are auditable for compliance adherence.

IT Service Management is the optimal intersection of people, processes, information, and technology, as shown Figure 1-6.

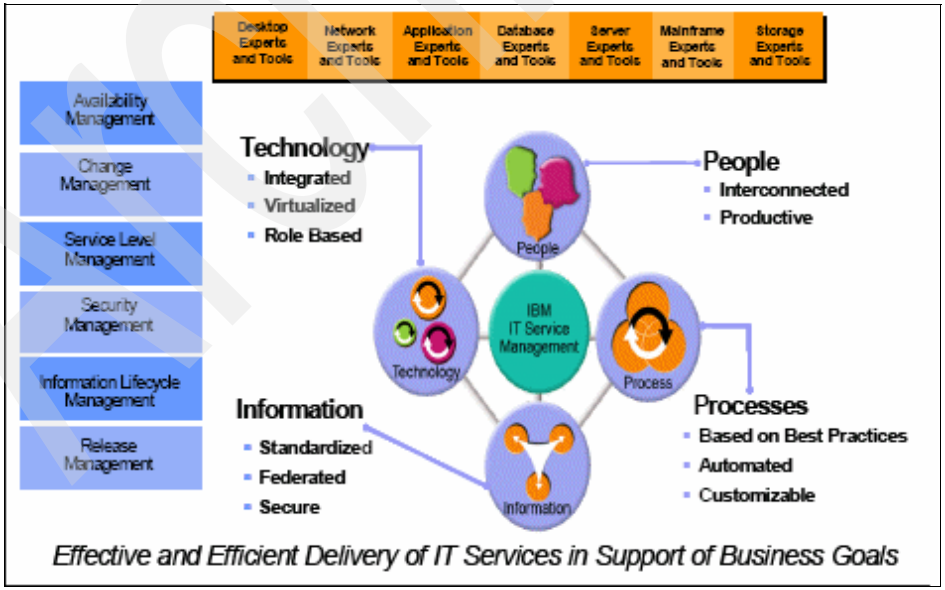


Figure 1-6 IBM Service Management interactions

Tools used to implement ITIL-based IBM Service Management (ISM)

ITUP is aligned with ITIL V3. IBM has a portfolio of tools that can help you implement IT Service Management processes that will fit the business needs of your organization.

Table 1-2 identifies which tools can be used to implement the ITIL Version 3 process.

Table 1-2 ITIL V3 processes mapped to tools

Process	Tool
Access Management	► IBM Tivoli Access Manager1 ► IBM Tivoli Federated Identity Manager ► IBM Tivoli Directory Integrator ► IBM Tivoli Directory Server ► IBM Tivoli Identity Manager ► IBM Tivoli Provisioning Manager
Asset Management	► IBM Tivoli Asset Management for IT ► IBM Tivoli Configuration Manager ► IBM Tivoli Contract Compliance Manager ► IBM Tivoli License Compliance Manager ► IBM Tivoli License Compliance Manager for z/OS® ► IBM Tivoli Network Manager ► IBM Tivoli Service Request Manager ► IBM Tivoli Usage and Accounting Manager
Availability Management	► IBM Tivoli Business Service Manager ► IBM Tivoli Data Warehouse ► IBM Tivoli Enterprise Console® ► IBM Tivoli Monitoring ► IBM Tivoli Netcool® OMNIbus ► IBM Tivoli Service Level Advisor
Capacity Management	► IBM Tivoli Capacity Process Manager ► IBM CICS® Performance Analyzer ► IBM Rational Performance Tester ► IBM Tivoli Business Service Manager ► IBM Tivoli Data Warehouse ► IBM Tivoli Intelligent Orchestrator I ► IBM Tivoli Network Manager ► IBM Tivoli Performance Analyzer ► IBM Tivoli Performance Modeler for z/OS ► IBM Total Storage Productivity Center I ► IBM WebSphere® Studio Workload Simulator for z/OS and OS/390®

Process	Tool
Change Management	► IBM Tivoli Business Service Manager ► IBM Tivoli Change and Configuration Management Database ► IBM Tivoli Service Request Manager
Configuration Management	► IBM Tivoli Change and Configuration Management Database ► IBM Tivoli Configuration Manager ► IBM Tivoli NetView® IBM Tivoli Network Manager
Data and Information Management	► IBM Tivoli Storage Process Manager ► IBM Tivoli Storage Manager ► IBM TotalStorage® Productivity Center
Event Management	► IBM Tivoli Enterprise Console ► IBM Tivoli Business Service Manager ► IBM Tivoli Compliance InSight Manager ► IBM Tivoli Data Warehouse ► IBM Tivoli Netcool OMNIbus ► IBM Tivoli Network Manager ► IBM Tivoli System Automation
Financial Management	► IBM Tivoli Usage and Accounting Manager ► IBM Tivoli Asset Management for IT ► IBM Tivoli Compliance InSight Manager ► IBM Tivoli License Compliance Manager ► IBM Tivoli License Compliance Manager for z/OS
Incident Management	► IBM Tivoli Service Request Manager ► IBM Tivoli Asset Management for IT ► IBM Tivoli Composite Application Manager for Response Time Tracking ► IBM Tivoli Composite Application Manager for SOA ► IBM Tivoli Enterprise Console ► IBM Tivoli Monitoring ► IBM Tivoli Netcool OMNIbus ► IBM Tivoli OMEGAMON® XE and DE ► IBM Tivoli System Automation
Information Security Management	► IBM Tivoli Security Operations Manager ► IBM Tivoli Access Manager ► IBM Tivoli Identity Manager ► IBM Tivoli Provisioning Manager ► IBM Tivoli Security Compliance Manager

Process	Tool
IT Service Continuity Management	► IBM Tivoli Business Continuity Process Manager ► Geographically Dispersed Parallel Sysplex™ (GDPS®) ► IBM Tivoli Storage Manager ► IBM Tivoli Storage Process Manager ► IBM Tivoli System Automation ► IBM TotalStorage Productivity Center
Knowledge Management	► IBM Tivoli Service Request Manager1 ► IBM Lotus® Connections ► IBM Lotus Notes® ► IBM Lotus Quickplace ► IBM Lotus Quickr™ ► IBM Lotus Sametime®
Monitoring and Control	► IBM Tivoli Monitoring ► IBM System Storage™ SAN Volume Controller ► IBM Tivoli Business Service Manager ► IBM Tivoli Composite Application Manager for Response Time Tracking ► IBM Tivoli Enterprise Console ► IBM Tivoli Intelligent Orchestrator ► IBM Tivoli Netcool OMNIbus ► IBM Tivoli Netcool Performance Manager for Wireless ► IBM Tivoli NetView ► IBM Tivoli Network Manager ► IBM Tivoli OMEGAMON XE and DE ► IBM Tivoli Service Level Advisor ► IBM Tivoli Storage Manager ► IBM Tivoli System Automation ► IBM Tivoli Workload Scheduler ► IBM TotalStorage Productivity Center ► IBM WebSphere Business Monitor

Process	Tool
Problem Management	► IBM Tivoli Service Request Manager ► IBM Tivoli Composite Application Manager for Response Time Tracking ► IBM Tivoli Composite Application Manager for WebSphere ► IBM Tivoli Data Warehouse ► IBM Tivoli Decision Support for z/OS ► IBM Tivoli Enterprise Console ► IBM Tivoli Monitoring ► IBM Tivoli Netcool OMNIbus ► IBM Tivoli Network Manager ► IBM Tivoli OMEGAMON XE and DE
Release and Deployment Management	► IBM Tivoli Release Process Manager ► IBM CICS Configuration Manager ► IBM Rational Build Forge® ► IBM Tivoli Composite Application Manager for WebSphere ► IBM Tivoli Configuration Manager ► IBM Tivoli Provisioning Manager
Request fulfillment	► IBM Tivoli Service Request Manager ► IBM Tivoli Access Manager
Risk Management	► IBM Tivoli Risk Manager ► IBM Tivoli Compliance InSight Manager
Service Asset and Configuration Management	See Asset Management, Configuration Management in this table
Service Catalog Management	IBM Tivoli Service Level Advisor
Service-Level Management	► IBM Tivoli Service Level Advisor ► IBM Tivoli Composite Application Manager for Response Time Tracking
Service Portfolio Management	► IBM Rational Portfolio Manager
Service validation and testing	► IBM Rational Functional Tester ► IBM Rational Manual Tester ► IBM Rational Performance Tester ► IBM Rational Purify® ► IBM Rational Robot ► IBM Rational Test Realtime
Supplier Management	IBM Maximo Online Commerce System
Transition planning and support	IBM Lotus Notes

Table 1-3 shows us the relationships between ITIL V3 books and the processes listed in Table 1-2 on page 12.

Table 1-3 Relationship between processes and ITIL V3 books

Process	ITIL V3 book
► Financial Management ► Risk Management ► Service Portfolio Management	Service Strategy
► Availability Management ► Capacity Management ► Data and Information Management ► Information Security Management ► IT Service Continuity Management ► Service Catalog Management ► Service-Level management ► Supplier Management	Service Design
► Asset Management ► Change Management ► Configuration Management ► Knowledge Management ► Release and Deployment Management ► Service Asset and Configuration Management ► Service validation and testing ► Transition planning and support	Service Transition
► Access Management ► Event Management ► Incident Management ► Monitoring and Control ► Problem Management ► Request fulfillment	Service Operation

IBM Service Management is a new way to align your organization and all its related functions with your business.

2

IBM Service Management products

IBM Service Management (ISM) is a comprehensive and integrated approach for Service Management, integrating technology, information, processes, and people to deliver service excellence and operational efficiency and effectiveness for traditional enterprises, service providers, and mid-size companies. This chapter discusses IBM Service Management products and has the following sections:

2.1 IBM Service Management

In the fall of 2007, the IBM Systems Journal provided a series of papers focused on the IBM Service Management strategy and related technologies and solutions. Some of the content from this section was extracted and paraphrased from the papers presented in the IBM Systems Journal. This IBM Systems Journal is available at:

http://ww.research.ibm.com/journal/sj46-3.html

In this section we describe the architecture of the Tivoli Service Management products. We cover:

- IBM Service Management overview
- IBM Service Management architecture
- Mapping of Tivoli products to IBM Service Management
- Tivoli process automation engine

2.1.1 IBM Service Management overview

Quality service delivery requires Service Management. Service is an offering, function, or activity delivered to an internal or external customer that may contribute revenue and profit or fulfill a critical mission of an organization, and is the output created through the use of an organization's human, intellectual, financial, and physical assets. Service Management encompasses the management processes, tactics, and best practices needed to deliver business services. IT, operations, and line-of-business services all require Service Management.

IBM Service Management is a comprehensive and integrated approach for Service Management integrating technology, information, processes, and people to deliver service excellence and operational efficiency and effectiveness for traditional enterprises, service providers, and mid-size companies. ISM is developed with the vision of meeting the following criteria for effective Service Management:

- *Breadth* of management across the entire service infrastructure
- *Scalability* requirements of the world's most demanding service infrastructures
- *Service context* to understand how to configure, optimize, and prevent disruptions
- *Integration* of technology, processes, and information to the correct people

- *Flexibility* and modularity to adapt for business, organizational, and technology convergence
- *Visibility* of end-to-end services, processes, and transactions
- *Automation* of processes and workflows fundamental to service delivery and support
- *Best practices* to provide rapid value and improve efficiency

ISM products and services cover four key areas, as shown in Figure 2-1.

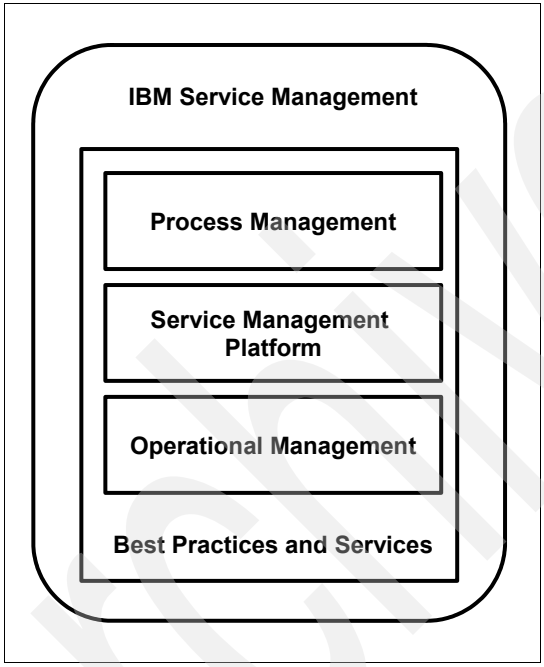


Figure 2-1 IBM Service Management

Process Management enables increased team performance, coordination, and collaboration. There are Process Manager products with the capabilities to:

- Enable consistent process execution.
- Align with best practices.
- Provide role-based visualization and control.
- Integrate IBM and third-party operational management tools into and across IT and business processes.
- Enforce and audit change and compliance.

The *Service Management platform* integrates visibility and control across people, process, technology, and information domains. IBM Service Management Platform delivers:

- Service visualization
- Data integration and federation
- Automation

Operation management products provide:

- Infrastructure management
- Role-based visualization and control
- Automation of tasks, workflows, and processes
- Open, standards-based products and tools

Best practices enable a modular approach for incremental execution and values through:

- Design, build, and run services provided by IBM Global Services
- Proven process models, standards, and best practices
- Standards-based build to manage toolkits
- Process Model for IT (PRM IT)
- IBM Tivoli Unified Process (ITUP)
- IBM Service Management Adoption Model: A step-by-step roadmap to improve Service Management
- Support for the implementation of Information Technology Infrastructure Library (ITIL), Control Objectives for Information and related Technology (COBIT), Enhanced Telecom Operations Map (eTOM), and other process models
- IBM Service Management Partner Ecosystem: An initiative that enables partners to drive innovation together around a common platform, the Change and Configuration Management Database (CCMDB), that delivers greater holistic value to customers
- Open Process Automation Library (OPAL): A collaborative portal available to partners and customers to exchange established workflows and process management tools

2.1.2 IBM Service Management architecture

IBM Service Management architecture is a service-oriented architecture designed to automate and simplify the management of business services. The

ISM architecture is the common architecture adopted by IBM Service Management products. The ISM architecture illustrated in Figure 2-2 on page 22 comprises four major components:

- A user interface that represents a portal-based integration of user interfaces (UIs) for user interactions and collaboration (labeled *portal-based user interfaces*)
- A process layer that includes the entity labeled *process runtime and services* and the entity labeled *Service Management solutions*
- An information layer represented by the Configuration Management Database (CMDB)
- An operational-management-technologies component represented by the entities labeled Operational Management products (OMPs) and the adjacent Integration Modules

Figure 2-2 also shows tooling, a collection of tools to create and modify processes, UIs, and data. The ISM architecture consolidates current IT components and management functions along several dimensions. All user interactions are consolidated at the portal. The roles and responsibilities of the users are integrated with the defined processes, and activities (subprocesses) are integrated with configuration data from the information layer. The architecture takes advantages of software middleware and industry standards for portal, workflow, and data federation.

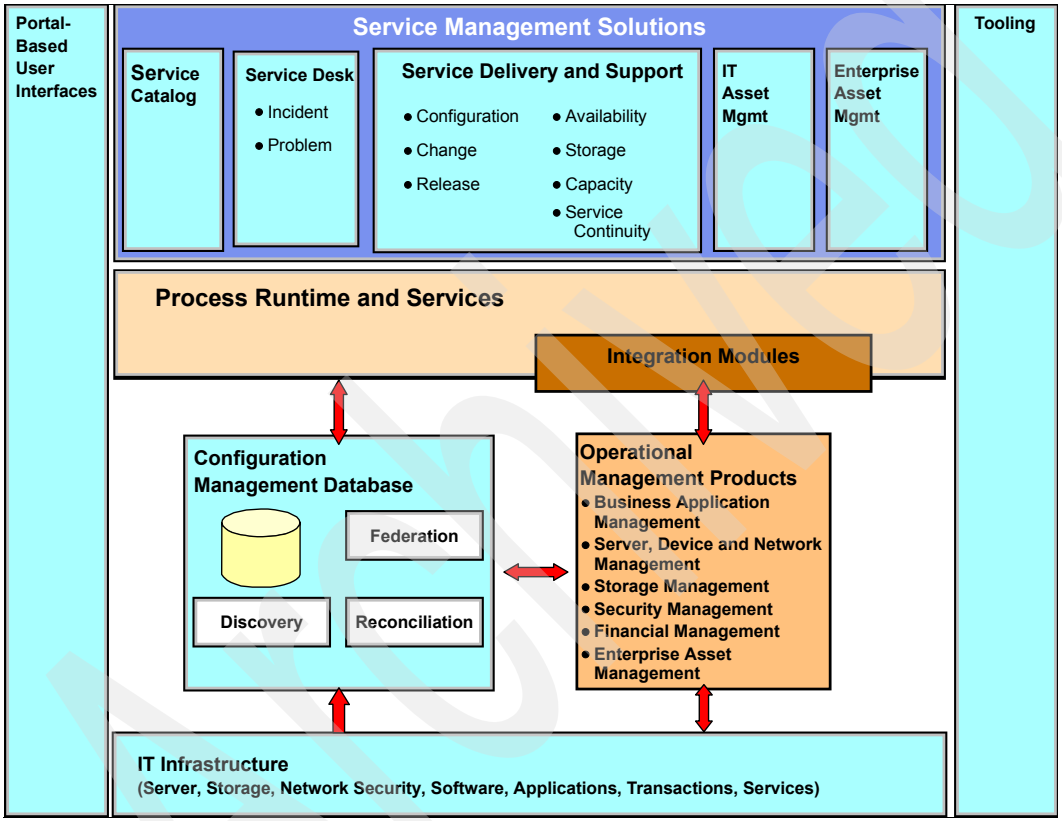


Figure 2-2 IBM Service Management architecture

Integration of Service Management processes with OMPs

Tasks performed as part of ISM processes leverage IBM OMPs and third-party products for task automation, thereby improving the overall efficiency of Service Management. Monitoring, event infrastructures, provisioning, distribution, availability, workload management, replication, backup, and security are among the pervasively deployed OMPs. For example, the deployment of a large-scale software update may utilize IBM Tivoli Provisioning Manager for Software to automatically distribute this software update to large numbers of desktops based

on a schedule. The ISM architecture allows the definition of logical management operations (LMOs) that provide an interface between the Service Management process and the OMPs that carry out the operation. A Web-services-based service-oriented architecture (SOA is used to implement these interfaces. This allows a loose coupling between the process and the OMP that provides the function, thus allowing an implementation to exploit best-of-breed OMP technology while maintaining process consistency.

To enable this loose coupling, the LMO interface is implemented by using an Integration Module. The Integration Module performs two key functions:

- It implements one or more calls to one or more OMPs by using the native interfaces of the OMPs, which could include command-line interfaces or application-programming interfaces (APIs).
- It maps the call arguments (provided by the process and based on the CMDB resource model) to arguments that are understood by the OMP. For example, a globally unique identifier (GUID) used by the process and CMDB to identify a server may need to be mapped to an object identifier that is used by the IBM Tivoli Provisioning Manager for Software to internally identify the same server.

The evolution and automation of processes will require additional LMOs and implementations. The ISM architecture supports the installation and configuration of these Integration Modules to interact with specific processes and tasks.

The ISM architecture enables client IT process transformation based on business process workflow tools, information integration technology, and operational management technology. The architecture provides a way for clients to transform their existing processes to incorporate best practices and gradually automate processes such as provisioning, orchestration, and problem determination. It provides access to accurate information that described the authorized and discovered states of the IT resources, commonly known as Configuration Management data. The ISM solutions are based on IBM and industry best practices, such as the ITIL, COBIT, and eTOM.

The process layer hosts solutions based on the concept of Service Management processes (also referred to as Process Managers or PMs). The Service Management processes are integrated with operational management technologies and the CMDB. The Service Management processes and related tooling incorporate a set of best practices that may be modeled and customized to support existing processes. Selected tasks within these processes may be progressively automated through operational management tools, directly reducing IT management costs in a manner consistent with organizational responsibilities. The integration of these tasks with the Systems Management technology (implemented through OMPs) is accomplished through the use of a service-oriented architecture.

The information layer, which includes the federated CMDB, provides automated application discovery and detailed views of system, software, and service topologies. Open interfaces provide ease of integration with process, data sources, and automation technology. Information about IT resources, topology, and relationships are often dispersed throughout operational registries used by management tools. Without including this information in a federated database, a logical view of all the IT resources and their respective relationships and dependencies is not available. This logical view is critical to improving the efficiency and effectiveness of processes. For example, to understand the impact of a change request and to implement a successful Change Management process, information about the current state of resources, the relationships to business applications, the service level objectives, the compliance policies, and the dependency on other resources are all critical information aspects. The integration of the Service Management processes and the CMDB with operational management technologies forms the core of theISM architecture.

2.1.3 Mapping of Tivoli products to IBM Service Management

In this section we look closer at how the Tivoli products are mapped to the IBM Service Management strategy and architecture. First we use a simple model, as illustrated in Figure 2-3, to show a basic flow of dependencies from business to technology:

- The business processes that are used to run the business and create revenue.
- The IT services that are required to support the business processes.
- These services are provided by the IT organization through the execution of standardized process.
- The processes consist of defined activities.
- The activities are further decomposed into distinct tasks.
- Technology or tools can be used to facilitate or perform these tasks.
- The tools are built on a set of technologies.

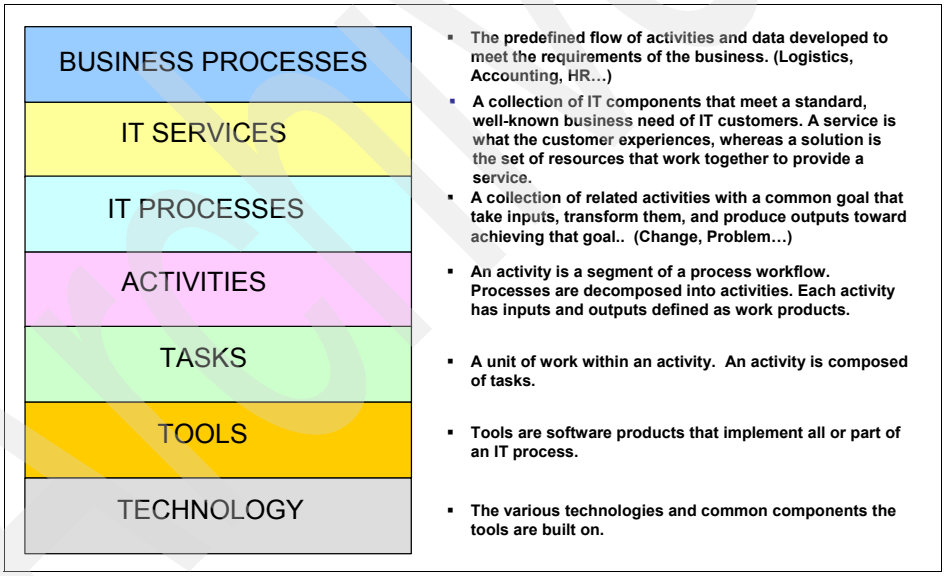


Figure 2-3 ITSM model: Business to technology

Figure 2-4 puts into perspective some of the many pieces of intellectual capital and products relative to ISM strategy. There are various levels of related business and process models that in turn drive the design of various products.

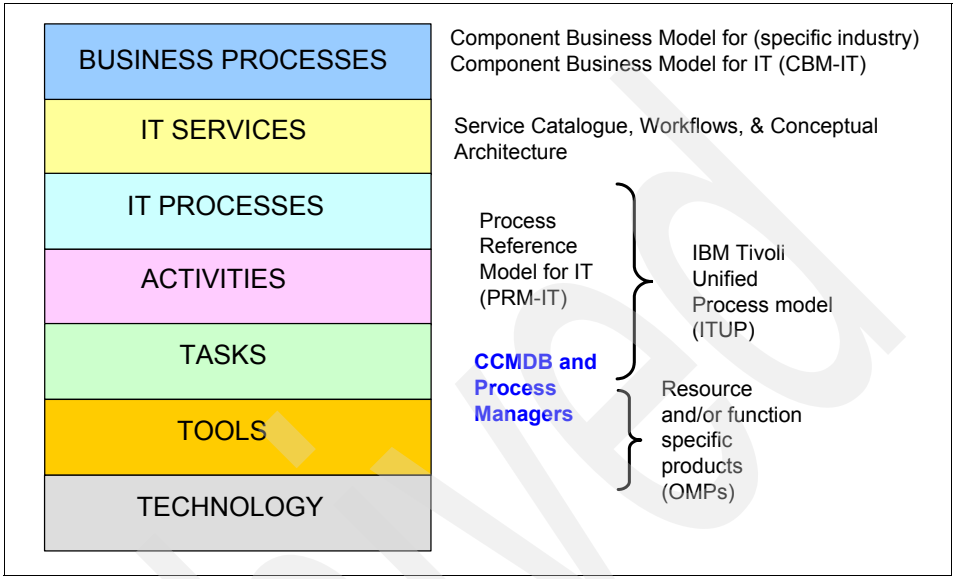


Figure 2-4 IBM intellectual capital and product overview

When we map out the ISM products on the ISM architecture, as shown in Figure 2-5, you will see that a large portion of the logical components is provided by the IBM Tivoli Change and Configuration Management Database V7.1 (CCMDB V7.1).

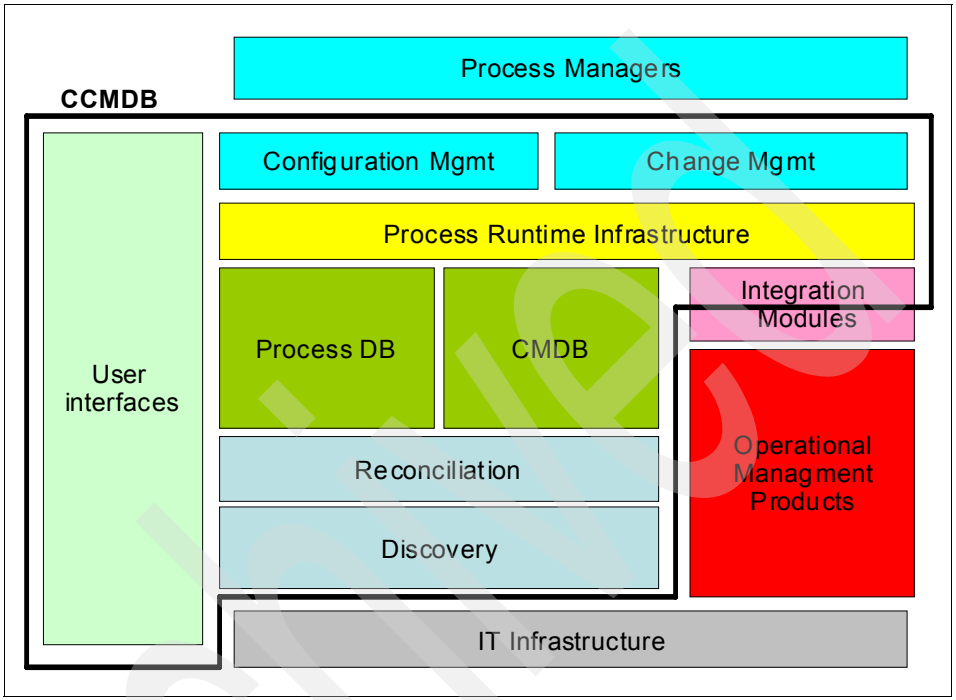


Figure 2-5 Logical components of ISM products

CCMDB is the foundation for the ISM strategy. It is the foundation for core ITIL process solution deliverables like *Configuration and Change or Release Management*. These process solutions provide best practice implementations of core ITIL processes.

The CCMDB provides a shared infrastructure as well as a set of foundation services used by different ISM process solutions, and includes the *configuration and Change Management* processes that provide core management capabilities needed in an IT environment.

In addition, the CCMDB incorporates a consistent data model and data layer implementation and includes a framework for discovery of resources and its relationships.

The other ISM Process Managers based on CCMDB V7.1, available on May 16, 2008, are:

- IBM Tivoli Service Request Manager (TSRM) V7.1 delivers user service request management through Service Desk and Service Catalog components. Service desk offers day-to-day management of incidents and problems. Service Catalog enables users to obtain IT services through published service offerings.
- IBM Tivoli Release Process Manager (TRPM) V7.1.1 is a productivity tool to manage, audit, and coordinate release tasks to be completed in the correct order by the correct people. It provides an executable process flow to enable business and IT people to plan, schedule, and implement new releases into the infrastructure.
- IBM Tivoli Asset Management for IT (TAMIT) V7.1 is an IT Asset Management solution that can track diverse IT assets across their entire life cycle.
- IBM Tivoli Business Continuity Process Manager (TBCPM) V7.1 enables organizations to operate best practice disaster recovery processes.

2.1.4 Tivoli process automation engine

Starting from the V7.1 products, the Tivoli Service Management products installer will install the product-specific applications on a common foundation called Tivoli's process automation engine.

- Tivoli's process automation engine provides the user interface, configuration services, process workflow runtime and services, and the common data system.
- Tivoli's process automation engine is more than what Maximo was in previous Maximo products. It includes install solutions, common services needed for ISM, and other services.

> **Note:** The Tivoli process automation engine used to be called Base Services, or Tivoli Process Automation Platform.

- Any product that has the Tivoli process automation engine as its foundation (including CCMDB, TAMIT, and TSRM) can be installed with any other product that has the Tivoli process automation engine. The installer will detect that the Tivoli process automation engine is already installed and just enable the additional applications and features of the product being installed.
- Every product ships with a specific version of the Tivoli process automation engine. The versions must match up if you are installing more than one product.

- The Tivoli process automation engine is the foundation of the following products:
  - Tivoli Change and Configuration Management Database 7.1

    Includes Tivoli process automation engine 7.1
  - Tivoli Change and Configuration Management Database 7.1.1

    Includes Tivoli process automation engine 7.1.1
  - Tivoli Service Request Manager 7.1

    Includes Tivoli process automation engine 7.1.1
  - Tivoli Asset Management for IT 7.1

    Includes Tivoli process automation engine 7.1.1
  - Enterprise Asset Management and Maximo Asset Management 7.1

    Includes Tivoli process automation engine 7.1.1
- Once you have installed one of these products, you can install any of the others on top of them. There is a different install option on each product's launchpad that you must choose if you want to install the product on top of another product. With this option, the Tivoli process automation engine will not be reinstalled. The new products applications will be enabled, as well as the existing applications.

For more information about the Tivoli process automation engine refer to section 2.3.3 of the book *Implementing IBM Tivoli Service Request Manager V7.1 Service Desk*, SG24-75799.

Now we put the Tivoli process automation engine and the Tivoli process automation engine-based products on the perspective of ISM architecture. Figure 2-6 shows the Tivoli process automation engine integrated portfolio.

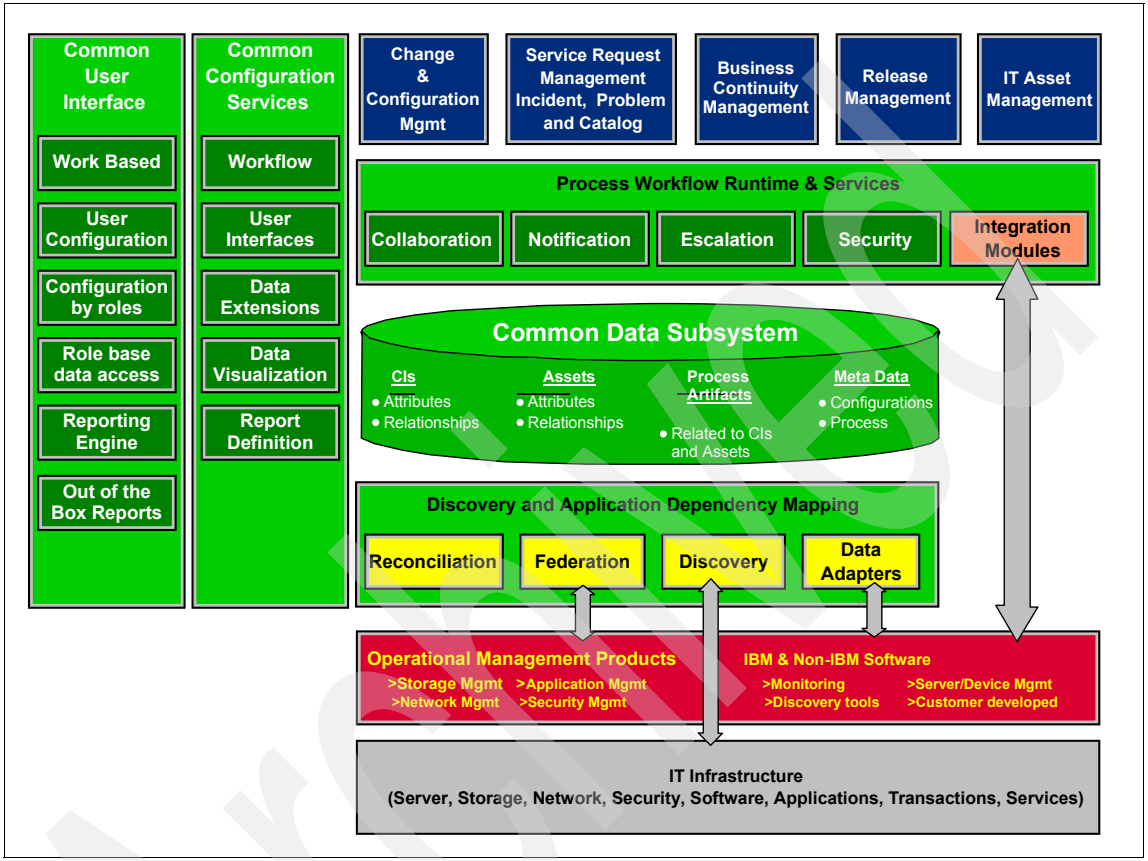


Figure 2-6 Tivoli process automation integrated portfolio

The following sections provide an overview of IBM Service Management products. Further information can be found in the IBM Service Management information center and IBM United States Software Announcement 208-241 in the URLs specified below:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp

http://www-01.ibm.com/common/ssi/index.wss

2.2 IBM Service Management products overview

In this section we provide an overview of the benefits and features of the following IBM Service Management products:

- IBM Tivoli Change and Configuration Management Database V7.1.1
- IBM Tivoli Service Request Manager V7.1
- IBM Tivoli Release Process Manager V7.1.1
- IBM Tivoli Asset Management for IT V7.1
- IBM Tivoli Business Continuity Process Manager V7.1

2.2.1 IBM Tivoli Change and Configuration Management Database (CCMDB) V7.1.1 overview

IBM Tivoli Change and Configuration Management Database is a platform for storing deep, standardized enterprise data. By integrating, automating, and optimizing data, workflows, and policies, it helps to align the ongoing management of IT infrastructure with business priorities.

- This scalable platform for the implementation of successful ITIL-based Service Management initiatives simplifies architectural complexity and reduces incident and problem management costs.
- A non-intrusive, agent-free discovery solution provides fast, automated application discovery, deep configuration detail, enterprise-class security, and easy integration with other data sources. Also included in the CCMDB is the critical relationship data that enables users to associate business configuration information directly to the physical IT environment.
- The discovery provided includes a variety of computer systems, network devices, applications, middleware, and databases.
- Using the portal interface, users can create, assign, monitor, notify, act upon, and report on change requests and configuration items.
- Using the data integration capability, users can make the most of existing investments in Operational Management products from IBM and other vendors to provide a consolidated view of the IT infrastructure.
- The creation of custom discovery library adapters and Integration Modules is possible with the toolkit included.
- Enforcing policies and tracking changes throughout the users' organization helps customers in their systems for compliance with internal and regulatory requirements.

Change Management and Configuration Management

These are at the core of any Service Management strategy. CCMDB enables you to standardize and share information that integrates people, processes, information, and technology for real business results. It includes an open, federated CMDB data model and the ability to automate process execution and it provides visibility to architectural complexity and helps reduce incident and problem management costs.

The Tivoli Change and Configuration Management Database helps benefit the users' business by:

- Delivering efficient, cost-effective management solutions by integrating IT processes, data, and people, and automating Operational Management product use
- Leveraging automated discovery, application mapping, and visualization capabilities to facilitate an extensive view of attributes and relationships between configuration items and supported business services
- Facilitating internal and regulatory compliance by enforcing policies as well as tracking and recording changes across your organization
- Employing best-practice Change Management processes with impact assessment and visibility of schedules to reduce business impact
- Helping ensure that configuration data is current through change and configuration process management
- Visualizing all critical intelligence regarding the infrastructure through data consolidation and federation capabilities

Through the integration, automation, and optimization of data, workflows, and policies, CCMDB helps align the ongoing management of your IT infrastructure with your business priorities, while helping to reduce and eliminate the impact that organizational complexity has on managing the infrastructure.

Tivoli Change and Configuration Management Database features:

- J2EE architecture enables flexibility, scalability, and access anywhere.
- Rich tooling allows configurations to be preserved version to version, and configurations stored as metadata in the database.
- Agentless, credential-less discovery of infrastructure elements.
- Deep configuration detail discovery from Wintel to System z®, including application relationship discovery.

- Common, unified platform to enable management of all Service Management activities, from Service Desk to Asset Management for IT.
- A Service Catalog is included.
- Support for Linux® on System z.

Figure 2-7 depicts the logical components related to CCMDB.

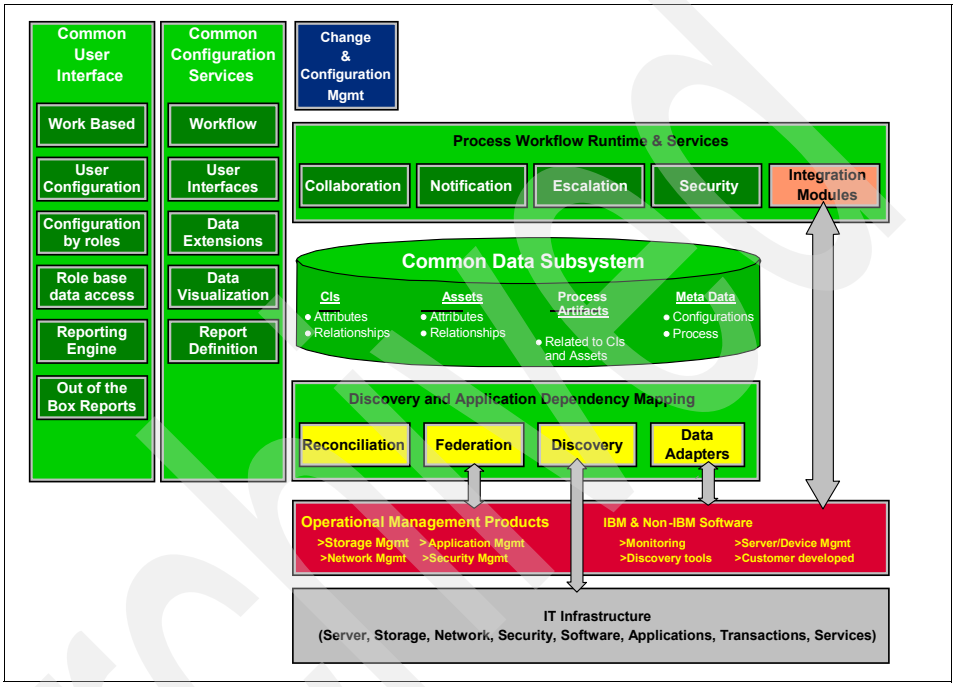


Figure 2-7 CCMDB logical components

For Change Management, the following processes and applications are enabled:

- Processes: Change process
- Applications enabled:
  - Changes
  - Change implementation schedule
  - Change window
  - Activities and tasks (Tivoli process automation engine core application)
  - Process requests (shared application)

For Configuration Management, the following processes and applications are enabled:

- Processes:
  - CI control (update) process
  - CI audit process
- Applications enabled:
  - Configuration items (shared application)
  - Actual configuration items
  - Process requests (shared application)
  - Configuration processes
  - Relationships (Tivoli process automation engine core application)
  - CI life cycles
  - Collections (Tivoli process automation engine core application)
  - CI types

For more information about the CCMDB product refer to *Deployment Guide Series: IBM Tivoli CCMDB Overview and Deployment Planning*, SG24-7565, which provides a more detailed overview of the CCMDB product and information related to planning and installing the product.

New features in CCMDB V7.1.1

The following new features are available in CCMDB V7.1.1:

- Support for new platforms
  - You can install and run CCMDB on a J2EE server running on SuSE Linux 9.0 Enterprise Server System z.
  - You can install and run CCMDB on a J2EE server running on a 64-bit Windows® machine.
  - You can use the middleware installer to install middleware on AIX®.
- IP V6 support

  You can install and run CCMDB on a network that uses Version 6 of the Internet Protocol, either V6 alone or a mixed network with IP V6 and V4.
- Migration manager

  The migration manager function helps you to move your configuration from one environment to another. For example, you might want to deploy CCMDB applications into a newly created cluster, or, perhaps you have deployed CCMDB into a test or development environment and, after customizing your deployment, you would like to migrate CCMDB to your production environment without using the CCMDB installation programs again. Migration

Manager enables you to define, create, distribute, and deploy packages of migration objects.

- Asset support in Change Management application
  - You can use the Change Management applications to work with assets as well as configuration items.
  - An asset is any component or item in an infrastructure that is under the control of Asset Management.
  - Assets can be used to model the end-to-end life cycle of a managed entity. Assets can be used to manage both financial aspects and operational aspects. You must create an asset for an entity if it has financial aspects such as purchasing, contracts, and inventory or some specific operational aspects. In other words, you can perform financial and operation activities with assets.
  - Each asset has several characteristics:
    - A classification, or type, that indicates what kind of item it is.
    - Attributes, which vary by classification and describe the characteristics of the individual asset.
    - Relationships, which indicate how the asset is related to other assets or configuration items.

Note: Refer to 4.6.1, “Assets and CIs in Change Management” on page 106, for more information about assets and CI integration.

- New task scheduler application

  The task scheduler application, part of Change Management, enables you to identify appropriate times to perform implementation tasks as part of a change. This application is also integrated with the change implementation schedule to resolve configuration item (CI) change window schedule conflicts.

For further information about the CCMDB, go to the product Web page:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.ccmdb.doc_7.1.1/ccmdb_welcome.htm

2.2.2 IBM Tivoli Service Request Manager (TSRM) V7.1 overview

In today's volatile, ever-changing IT environment, the Service Desk delivers critical support to the entire organization by keeping key business systems and services available and reliable. As technology becomes increasingly complex, problem resolution becomes more time-consuming, skill requirements increase,

and costs to maintain quality services escalate. In the face of tighter budgets and fewer resources, prioritization and responsiveness are the keys to maximizing the availability of business-critical IT services.

IBM Tivoli Service Request Manager combines the Service Desk and Service Catalog capabilities on top of a common process automation platform to provide a seamless, unified solution for all aspects of service requests, enabling a *one-touch* IT experience.

Tivoli Service Request Manager enables a unified solution with complementary products such as IBM Tivoli Asset Management for IT, IBM Maximo Asset Management, and IBM Tivoli Change and Configuration Management Database (CCMDB), facilitating a seamless approach to problem and Incident Management, change and Configuration Management, IT Asset Management, and Enterprise Asset Management.

Highlights of IBM Tivoli Service Request Manager can help:

- Streamline™ ITIL-based incident and problem management processes for more rapid service restoration.
- Increase the availability of critical IT services.
- Standardize and drive consistency and repeatability in IT service delivery with IT Service Catalog offerings.
- Optimize productivity of Service Desk personnel and increase user satisfaction.
- Align IT operations with your line of business through Service-Level Management.
- Assign and track SLA compliance to your service requests.
- Associate cost to your IT service offerings and manage consumption.
- Add asset, change, and Configuration Management functionalities to the Service Desk as a seamlessly integrated solution on a common platform.

Service desk

The Service Desk component of Tivoli Service Request Manager encompasses a broad variety of features that enable a single point of contact to automate incident and problem management. Built-in features streamline Service Desk functions and configure workflows and escalation across your organization, while a searchable knowledge base delivers fast answers to help desk agents. Additional features include:

- Dashboards that provide real-time performance views
- Out-of-the-box contents such as workflows, templates, KPIs, queries, and reports

- Remote diagnostics capability
- Instant messenger support
- Survey management capability
- Migration of configuration settings from test environment to production
- Integration with computer telephony and interactive voice response product

For more information about IBM Tivoli Service Request Manager Service Desk you may refer to *Implementing IBM Tivoli Service Request Manager V7.1 Service Desk*, SG24-75799.

Service Catalog

The Service Catalog component of IBM Tivoli Service Request Manager allows users to select services directly from a catalog, helping to lower the cost of providing services. Options can range from simple user services such as password reset to more complex services such as provisioning a server or upgrading an application environment. This versatile catalog also reflects the terms of any associated SLAs, rating and billing terms, and contractual agreements.

Figure 2-8 depicts the logical components related to TSRM.

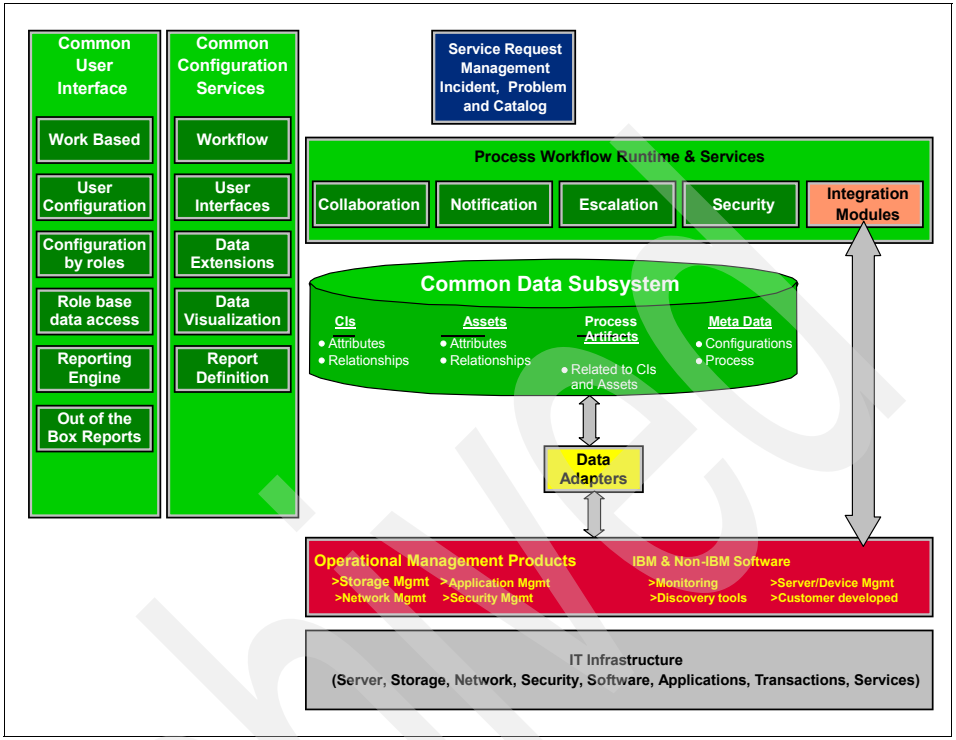


Figure 2-8 TSRM components

The following processes and applications are enabled by TSRM.

Processes:

- Service request management
- Incident Management
- Problem Management
- Service-Level Management

Applications enabled:

- Service Catalog
  - Service order management
    - View catalog requests
    - Catalog purchase requisitions
    - Catalog orders

  - Service Inventory
    - Catalogs
    - Service fulfillment
    - Fulfillment options
    - Offerings
- Service desk
  - Priority matrix
  - Solutions
  - Incidents
  - Problems
  - Service requests
  - Ticket templates
  - Activities and tasks
  - Process requests
  - Object-based search
  - Global search
- Service requests
  - Create service requests
  - View service requests
  - Search solutions
- Service-Level Management
  - Service level agreements
  - Service groups

For more information about IBM Tivoli Service Request Manager Service Catalog, you may refer to *Implementing IBM Tivoli Service Request Manager V7.1 Service Catalog*, SG24-76133.

For further information about IBM Tivoli Service Request Manager, go to the product Web page:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.srm.doc_7.1/srm_welcome.htm

2.2.3 IBM Tivoli Release Process Manager (TRPM) V7.1.1 overview

IBM Tivoli Release Process Manager puts users in control of their software and related hardware releases to protect the enterprise IT environment and committed service levels. It takes a holistic view of a change to an IT service to facilitate successful deployment in production. It manages, audits, and coordinates simple and complex release tasks to be completed in the correct order by the correct people. This portal-based solution helps users implement a

Release Management process faster with a set of best-practice ITIL flows, and coordinates and manages releases throughout the life cycle through an easy-to-use interface. It allows users to assess impact on their IT infrastructure and business-critical functions before they release, and manages software releases as a consistent and repeatable ITIL-based process to reduced time to implement and increased efficiency and productivity of the users' staff.

Highlights:

- The ability to perform impact analysis on assets
- The Task Scheduler application that contains analytics to examine a variety of criteria in order to suggest possible scheduling options
- CCMDB on System z platform support

  This means that the release Process Manager will work on System z platform-related configuration items (CIs) within the CCMDB.
- Support for Linux on System z

  This means that the release Process Manager will work with System z CIs within the CCMDB.

The Definitive Software Library Application allows you to add and remove software images or import from Tivoli Provisioning Manager for Software and Tivoli Configuration Manager. The deployment application enables you to distribute packages from the Definitive Software Library to a set of CIs within the infrastructure. The CIs may be managed by Tivoli Provisioning Manager or Tivoli Configuration Manager. In addition, the deployment application provides visibility to all releases in plan through the change implementation schedule, enabling you to identify other changes or releases that may cause conflicts.

Benefits

IBM Tivoli Release Process Manager V7.1.1 can help with the following IT functions:

- Automate the planning, approval, and implementation of releases to help users increase efficiency and effectiveness of running the business of IT.
- Focus on protecting the live environment and its services through the use of formal procedures and checks.
- Manage, audit, and coordinate simple and complex release tasks to be completed in the correct order by the correct people.
- Send automatic notifications when tasks are coming due or are overdue.
- Has easily customizable process flows to meet users' unique process requirements.

- Integrate Systems Management tools into the Process Manager, allowing users to increase processes at their own pace.
- Includes metrics with out-of-the-box reports focused on key performance indicators.

Figure 2-9 depicts the logical components related to TRPM. Note that CCMDB is a prerequisite of TRPM.

The following processes and applications are enabled by TRPM:

- Processes: Release
- Applications enabled:
  - Releases
  - Definitive software library
  - Deployments
  - Change implementation schedule
  - Activities and tasks
  - Process requests

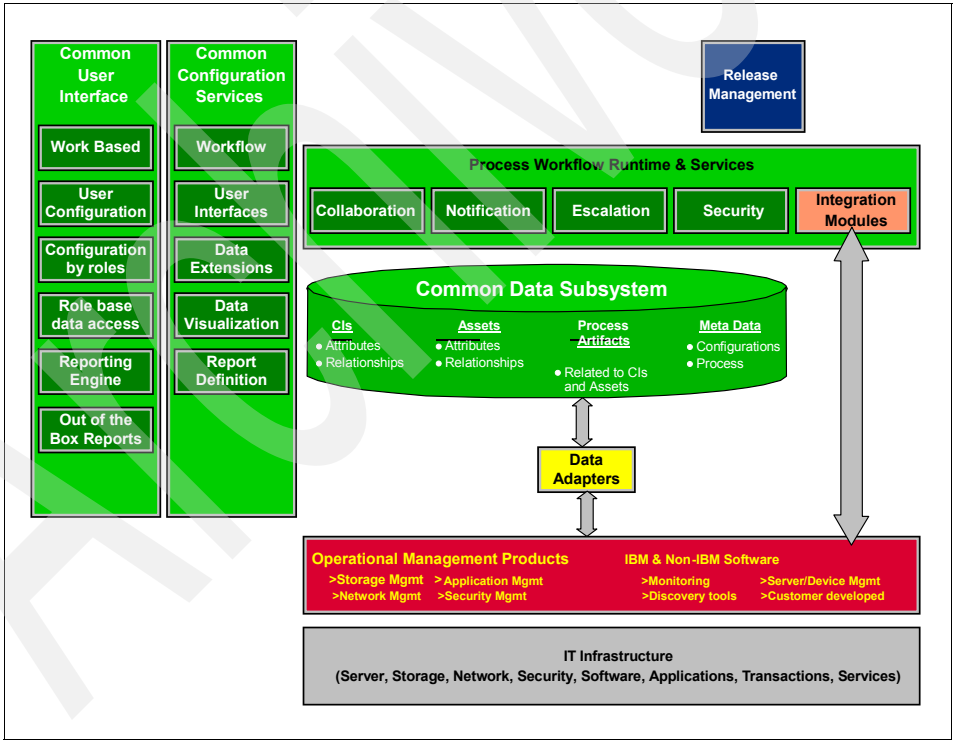


Figure 2-9 TRPM components

What is new in TRPM V7.1.1

In addition to all of the functionality provided in IBM Tivoli Release Process Manager 7.1, Release Process Manager 7.1.1 provides three new built-in job plans that you can use to install a database, install middleware, or build a new server. In addition, this product version enables you to use the Deployments application to deploy a number of Tivoli Provisioning Manager-supported logical device operations (LDOs), in addition to the software deployments supported by the previous version.

New job plans

The three new Release Process Manager job plans are aligned with similarly named and structured job plans provided in IBM Tivoli Service Request Manager 7.1 and in IBM Tivoli Change and Configuration Management Database (CCMDB) 7.1.1. The job plans are shipped as content in each of the three products, and are included by default when the products are installed. In Service Request Manager, the job plans are used within the service catalog component. In CCMDB, the plans are used within the Change Management component.

The alignment of these job plans across the three Process Managers (Service Catalog, Change, and Release) extends your flexibility in carrying out database installations, middleware installations, and server builds in your data center. Some of the available options are:

- Use a job plan to fulfill the operation strictly within the originating Process Manager. For example, Service Catalog might receive a request for a database installation. You can use the Service Catalog work plan provided for this purpose and execute the activities and tasks exactly as outlined in the Service Catalog job plan (which is part of the work plan). Similarly, you can carry out a request that originates in change or release entirely within the originating Process Manager, with no interaction among Process Managers.
- Starting with the Service Catalog work plan, you can submit a change process request. This moves the operation into change, where the request is fulfilled using the corresponding change job plan. In effect, you are turning the database installation over for execution within change. In the Service Catalog, you simply wait for the change process request to complete.
- Determine whether a deployment is needed to fulfill a request. For example, a database installation might be sufficiently complex that a formally scheduled deployment is required. In that case, the change manager could submit a request for a release. The database installation release job plan is then used to fulfill the request.

Each job plan contains nested, activity-level job plans. To view details about the job plans for Release, Change, and Service Catalog, click **Go To** → **Planning** → **Job Plans** and filter on PMREL (for release), PMCHG (for change), or PMSC (for

Service Catalog). A list of the job plans for that Process Manager is displayed, and you can click a job plan to view full information for that job plan.

You can also view more information about the release job plans by clicking **Reference** → **Content** → **Job Plans** in the navigation section of the Release Process Manager information center.

Table 2-1 lists the new release job plans and their corresponding change and Service Catalog job plans.

Table 2-1 New Release job plans

Release job plan	Change job plan	Service Catalog job plan
Middleware installation and configuration: PMRELMW	PMCHGMW	PMSC_0010C
Database installation and configuration: PMRELDB	PMCHGDB	PMSC_0007C
Server build: PMRELSB	PMCHGSB	PMSC_0006C

New classifications associated with the new job plans

Each of the new job plans that is provided for this product version has an associated release classification. If the Service Request Manager 7.1 Service Catalog component is installed, the classification that is provided by the Service Catalog is used for the change and release job plans that are applied to that request. If the Service Catalog is not installed, the change and release job plans use their own classifications for the associated job plans.

For example, if you open a process request in Service Catalog to perform a database installation, select the Service Catalog classification that is associated with the database installation job plan. Change and Release use this same classification, along with the corresponding job plan. If the process request is opened in Change, the Change classification is used. If the request is opened in Release, the Release classification is used.

Each classification has associated attributes. To view details about the classifications for Release, click **Go To** → **Administration** → **Classifications** and filter on RM. A list of the classifications is displayed and you can click a classification to view full information for that classification.

You can also view more information about Release classifications by clicking **Reference** → **Content** → **Release attributes** → **Release classifications** in the navigation section of the Release Process Manager information center.

Table 2-2 lists the three new release classifications that correspond to the new job plans.

Table 2-2 Release classifications

Release job plan	Corresponding classification	Attributes
Middleware installation and configuration: PMRELMW	RMMWINSTALL	Channel name, database name, installation name, queue name, table name
Database installation and configuration: PMRELDB	RMDBINSTALL	Name, installation name, table name
Server build: PMRELSB	RMSVRBLD	Channel name, database name, installation name, queue name, table name

Tivoli Provisioning Manager LDO deployments

One of the strengths of Tivoli Release Process Manager is its integration with Tivoli Provisioning Manager. Release Process Manager 7.1 provided the ability to pull Tivoli Provisioning Manager software modules into the Definitive Software Library (DSL) and deploy the software using the Deployments application.

In Version 7.1.1, you can use the Deployments application not only to deploy software modules but to perform a wide variety of logical device operations (LDOs) that are supported by Tivoli Provisioning Manager. For example, you can deploy network devices, perform bare metal installations, and extend file systems.

In order to enable this new functionality, Release Process Manager 7.1.1 provides a new deployment type, TPMOPERATION. Complete instructions for creating and implementing a Tivoli Provisioning Manager LDO deployment are provided in the "Working with deployments" section of the information center.

For further information about the Tivoli Release Process Manager, go to the product Web page:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.rpm.doc_7.1.1/rpm_welcome.htm

2.2.4 IBM Tivoli Asset Management for IT (TAMIT) V7.1 overview

IBM Tivoli Asset Management for IT helps users manage your IT asset life cycle efficiently and effectively. It can enable them to optimize sourcing and utilization, reduce asset costs, and improve service levels. Part of the same SOA as Tivoli Service Request Manager, Maximo Asset Management, IBM Tivoli Change Management, and IBM Tivoli Release Process Manager, it can help users achieve additional operational

Business benefits

IBM Tivoli Asset Management for IT is designed to help users achieve the following:

- Lower total cost of ownership by standardizing on the Tivoli process automation engine for asset and Service Management
- Greater efficiencies through system consolidation
- Visibility and control over your IT assets by knowing exactly what you have, where it is, who is using it, and what it costs
- Support for regulatory compliance initiatives (Sarbanes-Oxley, Basel II)
- Ease of use and business agility with role-based user interface, configurable KPIs, real-time dashboards, and built-in flexible tools for database configuration and workflow and application design
- Time and cost savings with user configurations changed as needed
- Less license compliance risk and fewer fines associated with unlicensed software
- Part of a unified solution to manage all critical assets that drive the business

Figure 2-10 depicts the logical components related to TAMIT.

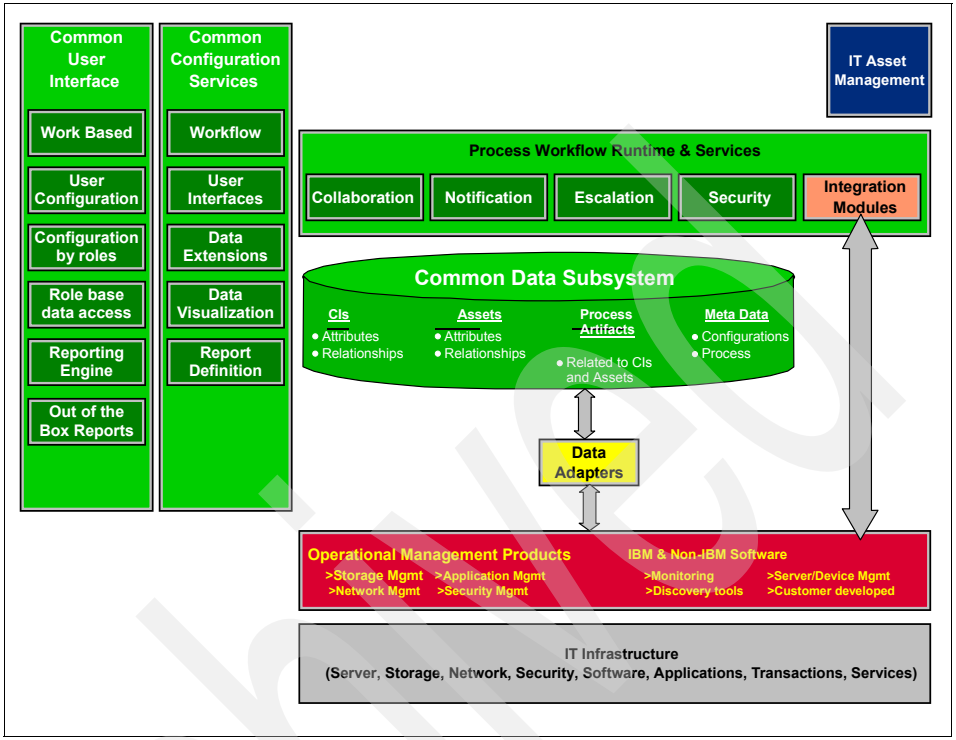


Figure 2-10 TAMIT components

The following processes and applications are enabled by TAMIT:

- Processes: Life cycle
- Applications enabled: The following applications are enabled with IBM Tivoli Asset Management for IT:
  - Desktop requisitions
  - Purchase orders and requisitions
  - Assets
  - Reconciliation
  - Inventory
  - Deployed assets
  - Software license view
  - Contracts

For more details about the TAMIT V7.1 refer to the IBM Redbooks publication *IBM Tivoli Asset Management for IT Portfolio Overview*, SG24-73766.

You can also refer to the following Web site:

http://www-01.ibm.com/software/tivoli/products/asset-management-it/

2.2.5 IBM Tivoli Business Continuity Process Manager (TBCPM) V7.1 overview

For effective IT Service Continuity using Tivoli software, a Process Manager is now available that leverages the infrastructure provided by CCMDB.

IBM Tivoli Business Continuity Process Manager provides a range of features to support the implementation of IT Service Continuity processes.

Note: Note that IT Service Continuity Management supports the overall Business Continuity Planning process and is not dealt with in isolation. Business Continuity Management is concerned with managing risks to make sure that an organization can continue operating at all times, at least, at a predetermined minimum level.

Administrators can use the out-of-the-box configuration to implement a ready-made business continuity solution or modify the supplied configuration to match their business requirements.

The key features of Business Continuity Process Manager include:

- A set of out-of-the-box components supported by the Tivoli process automation engine framework. After installing the product and providing site-specific configuration details, administrators can use the supplied components to implement a ready-made business continuity solution. Included in the set of components is a business continuity workflow that guides business continuity personnel through a sequence of well-defined disaster recovery processes.
- Integration with the Change and Configuration Management Database (CCMDB).
- Integration with the CCMDB common request mechanism. When a service request with a business continuity classification is accepted, an IT Service Continuity work order is automatically generated.

- Integration with specific IBM Tivoli Operational Management products (OMPs). These integrations enable business continuity personnel to do the following:
  - Launch OMPs direct from the Business Continuity Process Manager user interface to access OMP information. Business continuity analysts may do this during outage analysis activities to gather information about the impact of an outage. This console-launch feature reduces the time required for business continuity analysts to complete their analysis and draw their conclusions.
  - Initiate the execution of OMP console commands and scripts (for example, to enact a recovery option) direct from the Business Continuity Process Manager user interface.

Figure 2-11 shows the Business Process Manager architecture overview.

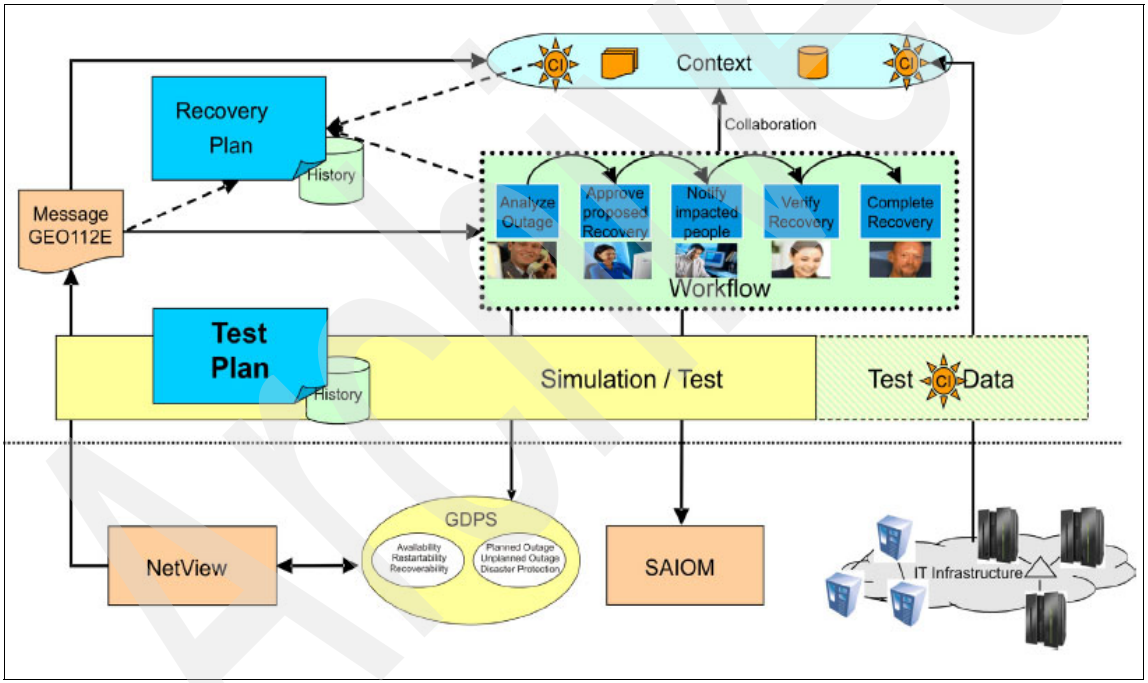


Figure 2-11 Business Continuity Process Manager architecture overview

TBCPM 7.1 provides out-of-the-box integration functionality with the OPMs described in Table 2-3.

Table 2-3 : Functions provided by business continuity integrations

Integration	Features
IBM Tivoli GDPS	Intercept GDPS GEO112 write-to-operator messages and convert them to IT Service Continuity work orders. Supply the necessary GEO113A reply messages to continue GDPS processing.
IBM Tivoli NetView	Trigger the execution of NetView scripts.
IBM NetView Web Application	Launch a GDPS/NetView console.
IBM Tivoli System Automation Application Manager	Launch System Automation Operations Console.
IBM Tivoli System Automation for z/OS	Launch Tivoli Enterprise Portal.
IBM Tivoli System Automation for Integrated Operations Management (SA IOM)	Issue business continuity communications using the alerting methods supported by SA IOM instead of or in addition to the CCMDB notification mechanism.

For further information about the Tivoli Business Continuity Process Manager, go to the product Web page:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.bcpm.doc/welcome.htm

2.3 Integration of IBM Service Management products

In this section we discuss:

- Integration requirements
- Integration categories
- Integration benefits
- Integration technologies
- Common integrations for IBM Service Management products

2.3.1 Integration requirements

Within the world of IT you will hardly find any solution that covers all the business requirements of your company, depending of your company size and business needs. Because of the specific functionality and complexity of, for example, financial processes, IT processes, and their alignment, integration has become a necessity.

The integration of ITIL or COBIT-like best practice processes with OMPs is a critical enabler for improving the efficiency of process activities and tasks. OMPs, such as monitoring, Event Management, provisioning, and license management products, allow Service Management processes to be applied to large-scale resource domains while minimizing repetitive labor cost and resulting errors.

2.3.2 Integration categories

The ISM platform supports three kinds of integration:

- User interface integration
- Functional integration
- Data integration

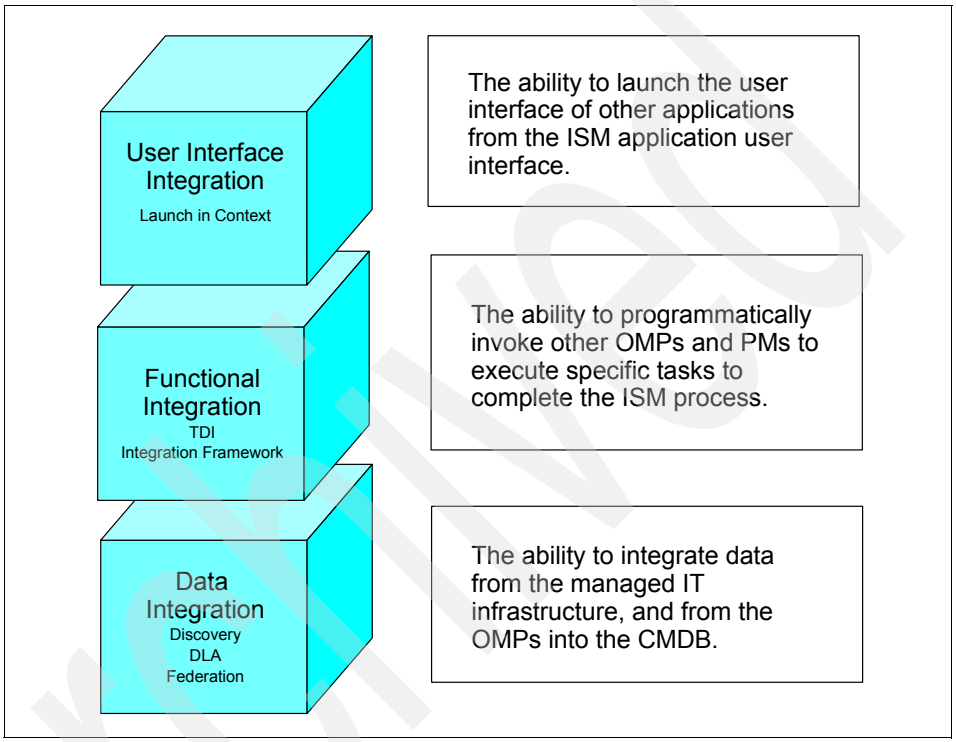


Figure 2-12 Integration categories

User interface (UI) integration

User interface integration is used when you need to invoke an application to perform something manually or need a role to make some judgement based on seeing something. Manual tasks are often performed by launching OMPs and interacting with their UIs. Examples of these include:

- While deploying a change that is requested on a server (as part of a Change Management process), a Change Deployer may want to see detailed information about this server maintained in the data center model of Tivoli Provisioning Manager.
- To identify the failing component (as part of an Incident Management process), a Service Desk analyst may need to launch into Tivoli Business Services Management.

The user interaction with the OMP UI can be optimized (for example, redundant input and the number of panels navigated can be minimized) by launching the OMP UI in the context of one or more CIs for which the task is being performed. Launching into the OMP UI should not only display the appropriate contextual view, but also pass contextual information about the CIs for which the OMP is being launched.

Launch-in-context (LIC) is provided as a general mechanism in the ISM architecture. launch-in-context can be used between Service Management processes and OMPs, directly between OMPs, between different processes, and between processes and the CMDB.

For more information about launch-in-context refer to Chapter 11, “Launch-in-context,” of *Deployment Guide Series: IBM Tivoli CCMDB Overview and Deployment Planning*, SG24-7565.

Functional integration

Functional integration (also called process integration) enables process tasks to programmatically invoke OMPs to execute specific tasks. In conformance with an SOA approach, the invocations of OMPs are implemented by using an LMO, an abstract logical interface that is loosely coupled with the specific APIs provided by an OMP. As mentioned earlier, LMOs provide a service abstraction of the OMPs and a degree of transparency from versions, instance, and location. The architecture enables the Integration Module that implements an LMO (that is, binds the LMO to a specific OMP) to be separately developed, installed, and configured within the platform. This provides the opportunity to create an ecosystem of OMP vendors to be integrated with the ISM platform.

The Integration Module architecture includes the following key aspects:

- An Integration Module is defined as a Web service, using standard Web Services Description Language (WSDL) interface definitions.
- An Integration Module implements one or more LMOs by providing the binding and functional mapping between the syntax and semantics of the LMOs to the interfaces provided by the target OMP.
- To accomplish the binding and functional mapping, the Integration Module translates the process reference to a CI into corresponding identifiers for one or more resources as understood by the OMP. This is accomplished by interacting with the CMDB, which maintains the mapping between the CMDB CI identifier and the resource identifier for each OMP that can manage the resource.
- Each Integration Module is registered with the CMDB with appropriate information about the location of the OMP, the LMOs it supports, and the set or collection of CIs for which it supports the LMOs.

Use of an SOA to implement OMP integration is a key differentiating aspect of the ISM architecture. We expect that the increasing maturity and automation of Service Management will drive the definition of standardized LMOs for the various Service Management process domains.

We cover several scenarios in this book describing the functional integration.

Data integration

Data integration is used to load the data into the CMDB. The CMDB supports the following mechanisms to populate and maintain the data in these repositories:

- Data discovery: Information about CIs can be discovered directly from the IT infrastructure through sensors.
- Data adapter: Import mechanisms bring in data from other sources (for example, management tools, spreadsheets) that may already have discovered information about the CIs.
- Data federation: Enables logical access to CI data from another repository.
- Data reconciliation: Ensures that discovery of data originating in multiple sources of data (such as distinct monitoring and provisioning management tools) about the same CI results in a single system of record in the CMDB.
- User input and application programming interfaces: These interfaces create, read, update, and delete data in the CMDB.

For more information about data integration, you can refer to the IBM Redbooks publication *Integration Guide for IBM Tivoli Service Request Manager V7.1*, SG24-7580. That book presents several scenarios for TSRM data integration.

2.3.3 Integration benefits

In general, the advantages of integration with third-party solutions or any other external solution can be:

- Availability of information for assisting the roles of the Service Management processes to make a decision.
- Leveraging on existing investment on hardware, software, skills, and implementation effort.
- Enable incremental implementation of new functionality and new automation process when justified.
- Increasing the automation in the end-to-end process.
- Having one primary location where data is stored and maintained, while you still have the option to exchange and use the data in other solutions.
- Reducing costs as exchanging data via automated integration requires less manual interaction.
- Reducing the number of errors as exchanging data via automated integration requires less manual interaction.
- Enforcing data integrity as exchanging data via automated integration takes care of data synchronization.

When implementing a synchronization solution, the result is an environment where shared data looks the same for all consuming applications. This is because changes are propagated throughout the synchronized network of systems, molded in transit to fit the needs of each consumer. Each data source is kept up-to-date, maintaining the illusion of a single, common repository. Each application accesses its data in an optimal manner, utilizing the repository to its full potential without creating problems for the other applications.

2.3.4 Integration technologies

The key technologies available are:

- *Launch-in-Context (LIC):* A functionality provided by the Tivoli process automation engine to launch-in-context from the ISM Web interface to either the Discovered CI space in the discovery/TADDM environment or to an external Operational Management product (OMP). Launching into the TADDM environment requires a GUID as a reference identifier. You can launch into either the Domain Manager or Java™ Client in either a Domain or an Enterprise Discovery Server. As part of the base services, the CCMDB V7.1 provides a *launch-in-context* application to define the launch points to different target systems and launch reference points.

- *Discovery Library Adapter (DLA):* A program that copies data from an Operational Management product, converts it to IdML, and stores it in books in the discovery library.
- *IBM Tivoli Integration Composer (ITIC)*: This is used for the mapping of TADDM data (Discovered CIs) into CCMDB data (Actual CIs).
- *Federation Service* is used to create a link in the CI data that is kept inside the CCMDB to external data sources without physically copying the external data into the CCMDB. IBM provides federation capabilities within the DB2® subsystem in case you are federating to another DB2 or Informix® database. In case your external data source is different from DB2 or Informix, you must install the WebSphere Federation Server component on top of the DB2 system that is keeping your CCMDB data.

  Note: Federation capability is dependent on the database used for CCMDB installation. If, for example, the CCMDB uses Oracle® as the database, a federation capability is available from CCMDB to external Oracle databases.

- *Maximo Enterprise Adapter (MEA) integration framework* allows you to expose any object (MBO) (which encapsulates database structures) in order to exchange data with external systems. The MEA integration framework not only allows you to import or export data into the CCMDB database (for example, to load authorized CIs from a flat file, XML file, or interface table), but it also allows you to link your CCMDB system and processes into an overall business flow. For example, you can link your Change Management process to an external procurement system in case the change requires the purchase of some IT equipment.
- *Integration Module (IM)* is a specialized implementation of the MEA integration framework. An Integration Module is specialized code that interfaces with an external system in order to call an operation within an external system and receive a synchronous or asynchronous response. The Tivoli Provisioning Manager (TPM) or the Tivoli Configuration Manager (TCM) products are examples of an OMP. You can call an OMP, such as TPM, to perform a Logical Management Operation (LMO), such as deploy software. TRPM is shipped with Integration Module to integrate with TPM or TCM to enable you to associate the software package CIs in the Definitive Software Library (DSL) with the software image stored in TPM or TCM.
- *IBM Tivoli Directory Integrator (TDI)* is a truly generic data integration tool that is suitable for a wide range of problems that usually require custom coding and significantly more resources to address with traditional integration tools. It is designed to move, transform, harmonize, propagate, and synchronize data across otherwise incompatible systems. Tivoli Directory

Integrator can be used in conjunction with the deployment of integration with the IBM Tivoli Service Request Manager product to provide a feed from multiple Service Desk Systems like HP Service Desk, Remedy Service Desk, as well as to function as a custom adapter to integrate with network monitoring tools like Tivoli Enterprise Console and Netcool Omnibus.

- *Business Process Execution Language (BPEL)* provides a language for the formal specification of business processes and business interaction protocols. If you have an environment hosting BPEL flows, for example, based on WebSphere Process Server technology, the CCMDB V7.1 predefined process flows (Change and Configuration Management) provide Web service interfaces in order to call specific activities from an external process engine hosting BPEL workflow. The detailed task flow is still maintained in the CCMDB process environment.

Refer to Chapter 7 of the IBM Redbooks publication *Deployment Guide Series: IBM Tivoli CCMDB Overview and Deployment Planning*, SG24-7565, for the integration technologies available to CCMDB. Implementation guidelines for the integration technologies can be found in the IBM Redbooks publication *IBM Tivoli CCMDB Implementation Recommendations*, SG24-7567. Integration for TSRM is also covered in details in *Integration Guide for IBM Tivoli Service Request Manager V7.1*, SG24-7580.

2.3.5 Common integrations for Tivoli Service Management products

In this section we list some of the common integrations for CCMDB, TAMIT, and TSRM using the integration technologies mentioned in 2.3.4, “Integration technologies” on page 54. Some of these can be found on the IBM Tivoli Open Process Automation Library (OPAL) Web site at:

`http://catalog.lotus.com/wps/portal/topal`

Common integrations for CCMDB:

- TADDM integration via new and existing DLAs
- Change integration with Release to TCM and TPM
- Loading authorized CIs directly into Tivoli process automation engine, using TDI
- Importing Actual CIs from TADDM (This is all part of CCMDB, but some see this as an integration.)
- MS® Project adapter to allow management of Change work orders from Project (coming soon)

Common integration for TAMIT:

- ITIC adapters that load deployed assets
  - Tivoli Application Dependency Discovery Manager 7.1
  - Tivoli License Compliance Manager for Distributed 2.3
  - Tivoli License Compliance Manager for z/OS 4.2
  - System Center Configuration Manager 2003 (SMS)
  - Alteris Inventory Solution 6.5
  - HP Openview Inventory Manager 3.1
  - Centennial Discovery 2006/2007
  - Tivoli Provisioning Manager (TPM)
  - Tivoli Configuration Manager (TCM) 4.2, 4.2.3
  - Netcool Precision 3.7
- Use Integration Framework (MEA) to import (authorized) assets from:
  - SAP
  - Peoplesoft
  - HP Asset Manager

Common integration for TSRM:

- OMP integration
  - Tivoli Monitoring
  - Netcool
  - Tivoli Identity Manager
- Event generators
  - Tivoli Enterprise Console (TEC)
  - NetView on Z (OPAL)
  - Omnibus (OPAL)
- Other Service Desks
  - HP Service Center
  - Remedy
  - Infoman (OPAL)
- Telephony (CTI)

  Genesys

2.3.6 Operational Management products

Operational Management products automate tasks to address application or business service operational management challenges. These products help optimize the performance and availability of your business-critical applications, along with the supporting IT infrastructure. They also help ensure the confidentiality and data integrity of your information assets while protecting and maximizing the utility and availability of your business data.

The Tivoli products related to Service Management can be categorized into four areas:

- Performance and Availability Management: By proactively managing both the performance and the availability of key IT assets, customers can use them to help identify and then resolve potential problems before they can affect the health of their business computing systems.
- Storage management: An integral part of your IT computing landscape, which when managed properly can help ensure the availability of key business-critical data and help control costs.
- Security management: An essential component that will aid in protecting users, networks, data sources, IBM Business Partners, and applications from threats wherever the source or whatever the intent.
- Infrastructure and business management: Key critical components that provide the foundation for process and Service Management to allow applications, networks, databases, and computing infrastructure components to work together in an integrated manner.

By using the integration technologies as mentioned in 2.3.4, “Integration technologies” on page 54, you can create your own integration scenarios to automate the ISM processes. A simple example can be found in Figure 2-13. More integration scenarios can be found in Chapter 6, “Scenario 2: Fixpack deployment” on page 153, and Chapter 7, “Scenario 3: Process level integration with an external Service Desk system” on page 183.

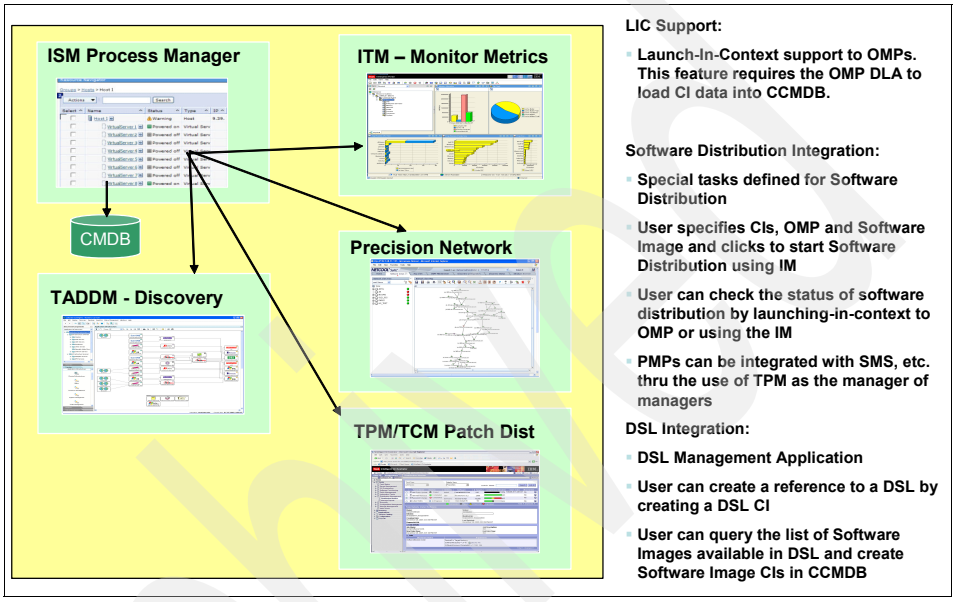


Figure 2-13 CCMDB integration scenario

For the purpose of this book we provide a brief description of the OMPs used in our scenarios:

- *IBM Tivoli Monitoring (ITM)* provides monitoring for essential distributed system resources to detect bottlenecks and potential problems, and to automatically recover from critical situations. ITM provides you with the IBM Tivoli Enterprise Portal (TEP) user interface to give you a single point of control for managing the resources that your applications rely on, including a range of operating systems, servers, databases, platforms, and Web components. TSRM can use the LIC technology to launch the TEP, for example, to enable the incident owner to perform resolution.
- *Tivoli Enterprise Console (TEC)* is an Event Management software system that collects, consolidates, and correlates events from a variety of event sources across the managed network, initiating automated corrective action when appropriate, in order to reduce the number of events that require human intervention to a manageable size. Events are consolidated or correlated by filtering redundant or low-priority events, discarding duplicate events,

discarding secondary events (events caused by other events) where appropriate, and automatically closing a problem event when the related recovery event occurs. It has preconfigured rules that provide best-practice Event Management. It enables comprehensive management that even accepts events from non-Tivoli products/systems. TSRM can be integrated with TEC using the Integration Framework (also called MEA or Maximo Enterprise Adapter) and TDI technologies, for example, to enable automatic creation and closure of incident tickets.

- *Tivoli Netcool/Omnibus (Omnibus)* provides similar functionalities as TEC in terms of Event Management. It delivers real-time, centralized monitoring of complex networks and IT domains. It provides bi-directional integration of Omnibus with TSRM and keeps the incidents synchronized with the related events within Omnibus. Similar to the TEC, Omnibus integration uses TDI. The Omnibus Probe and Gateway are also required to manage the Event Management workflow.
- *Tivoli Business Service Manager (TBSM)* provides real-time dashboards to enable IT operations and lines of business to have visibility into the availability, performance, and integrity of critical business services, and associated key performance indicators and service level agreements. With this real-time knowledge operations staff can prioritize response to problems according to the greatest impact on the business. TBSM collects events and dependency information from a wide variety of sources such as Omnibus, TEC, and TADDM. TBSM provides a Discovery Library Reader to import discovered data into its own data stores to support the creation of service models.
- *Tivoli Provisioning Manager (TPM)* automates manual tasks of provisioning and configuring servers and virtual servers, operating systems, middleware, applications, and storage and network devices acting as routers, switches, firewalls, and load balancers.

  Tivoli Configuration Manager (TCM) has the software distribution capability that enables you to rapidly and efficiently deploy complex mission-critical applications to multiple locations from a central point. TPM/TCM can be integrated with TRPM using the Integration Module technology, for example, for software distribution.
- *Tivoli Identity Manager (TIM)* provides a secure, automated, and policy-based user management solution to address key business issues. Tivoli Identity Manger helps enterprises set up new accounts and passwords quickly for employees and customers, including the ability for users to reset and synchronize their own passwords. It also allows enterprises to improve visibility into security management operations and quickly produce reports for auditors with predefined reports and audit events. TSRM can be integrated with TIM using the Integration Framework technology, for example, to automate the password reset.

3

Planning your IBM Service Management infrastructure

This chapter contains best practice information for planning for scalability and high availability of an IBM Service Management Solution based on the following products:

- IBM Tivoli Service Request Manager (TSRM)
- IBM Tivoli Change and Configuration Management Database (CCMDB)
- IBM Tivoli Asset Management for IT (TAMIT)
- IBM Tivoli Release Process Manager (TRPM)
- IBM Tivoli Application Dependency Discovery Manager (TADDM)
- IBM Tivoli Provisioning Manager (TPM)

This chapter does not cover installation details of these environments. These can be found in the Tivoli products manuals and in IBM Redbooks publications *IBM Tivoli CCMDB Overview and Deployment Planning,* SG24-7565, and *IBM Tivoli CCMDB Implementation Recommendations,* SG24-7567.

3.1 Understanding the IBM Service Management environment

In our sample IBM Service Management (ISM) environment we have included the following products:

- IBM Tivoli Service Request Manager V7.1
- IBM Tivoli Asset Management for IT V7.1
- IBM Tivoli Change and Configuration Management Database V7.1.1
- IBM Tivoli Release Process Manager V7.1.1
- IBM Tivoli Application Dependency Discovery Manager V7.1
- IBM Tivoli Provisioning Manager V5.1.1

Important: Note that we have included IBM Tivoli Provisioning Manager because it is a prerequisite of IBM Tivoli Release Process Manager. Currently, TPM does not utilize the services of Tivoli process automation engine, but TPM V7.1 will be running on top of Tivoli process automation engine and will be using all services of this platform as the other Tivoli process automation engine products. This means tight integration on process, data, and GUI level.

We discuss best practices for sizing of this environment. We also discuss implementation topology, high availability, and scalability considerations.

Figure 3-1 on page 63 shows our sample ISM environment. Note that in addition to the products discussed above, this environment has other products such as Tivoli License Compliance Manager and HP Service Center. We have included these to show you the integration capabilities of the ISM solution. This is one of the most complex topologies that you might see in a client site. It this important to understand that not all clients will be using all these products. For example, a client can start with a basic Information Technology Infrastructure Library (ITIL)-based configuration solution using CCMDB and then later add additional products like Release Process Manager and Tivoli Provisioning Manager to implement an ITIL-based Release Management process.

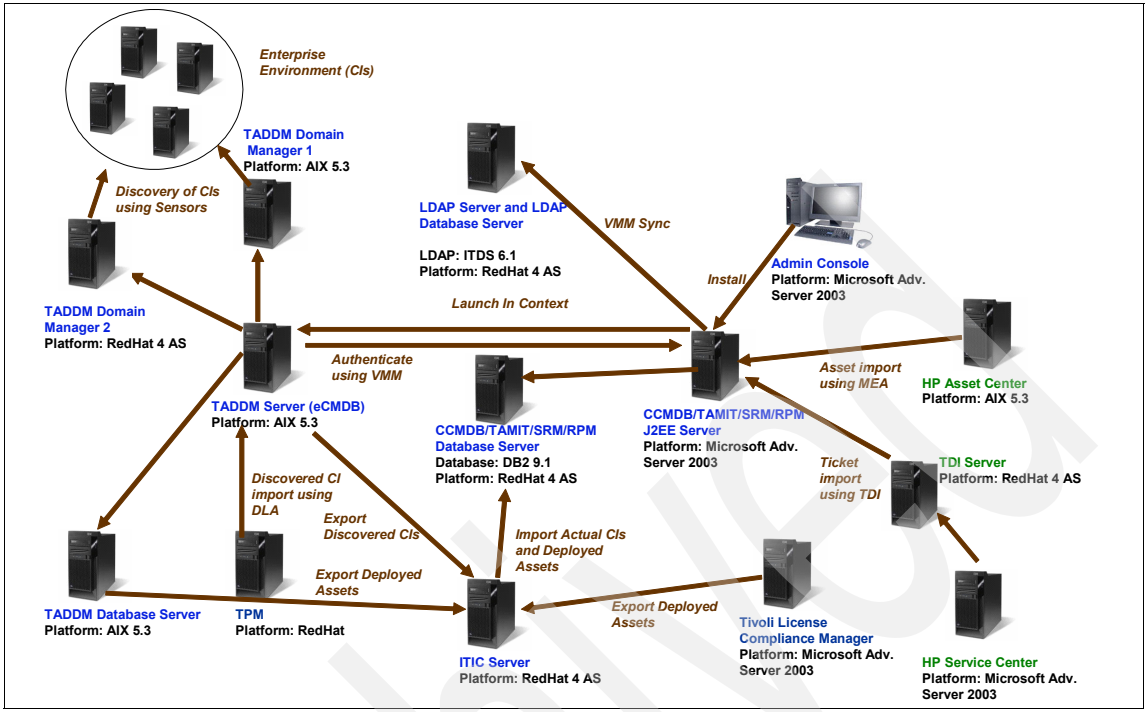


Figure 3-1 ISM environment

To better explain this complex topology and show you the interaction between these components we break down the overall topology into sub components, such as:

- TADDM topology
- Tivoli process automation engine topology
- Tivoli process automation engine and TADDM combined topology

3.2 Topology considerations

In this section we discuss topology considerations.

3.2.1 TADDM topology

Figure 3-2 shows the TADDM topology. Note that the system that is labeled *WebSphere with VMM and ESS Server* system is the same system that was labeled *CCMDB/TAMIT/SRM/RPM J2EE Server* in Figure 3-1 on page 63. The J2EE services are provided by WebSphere.

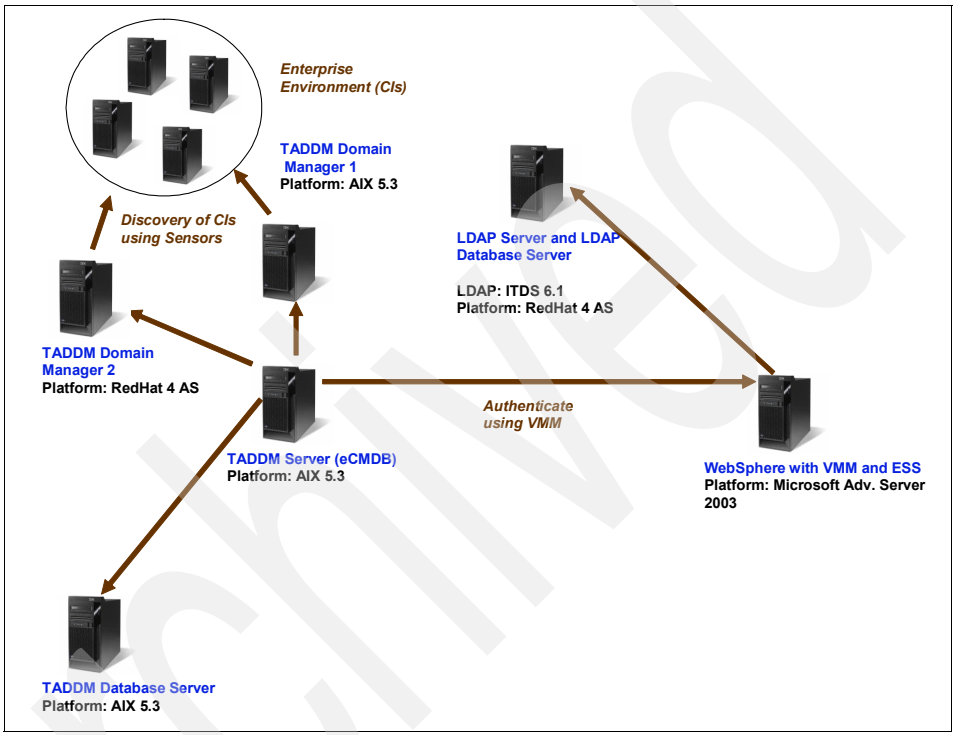


Figure 3-2 TADDM system topology

In the ISM solution, all interactions between the applications and the LDAP Directory Server flow through the Virtual Member Manager (VMM), a component of WebSphere. VMM's common interface masks the differences of the LDAP provider implementation. The LDAP maintains user and group entries as well as the user-to-group relationships. The relationship defines which user is a member of which group. In addition, passwords for users are maintained in the LDAP implementation.

Note: Although some of the Process Manager products (PMPs) such as TSRM can also be run without LDAP. LDAP is a prerequisite for CCMDB, so we use LDAP in this sample integrated ISM environment. If, for example, you want to use only TSRM and not CCMDB, you have the option of not using LDAP.

Extended Security Services (ESS), on the other hand, is used for launch-in-context, for example, when a user tries to launch-in-context from a Process Manager product into the Discovered CI space of the TADDM environment. After TADDM receives the token through the URL of a launch-in-context operation, it validates the token by calling the Extended Security Services. This provides single sign-on within the ISM solution.

TADDM in ISM solution is a prerequisite of CCMDB and is used to discover the CIs and their relationships. This is called Discovered CI space. The promotion from Discovered to Actual CI is managed through the Integration Composer, which we discuss in 3.2.2, “Tivoli process automation engine topology” on page 66.

Figure 3-2 on page 64 shows two TADDM Domain Managers. These are optional components in a TADDM environment, but when used increase the scalability of the discovery process. This is discussed in more detail in “Using TADDM Domain Managers to increase scalability” on page 77.

3.2.2 Tivoli process automation engine topology

Figure 3-3 shows the Tivoli process automation engine topology. Let us see the individual components that make up this topology.

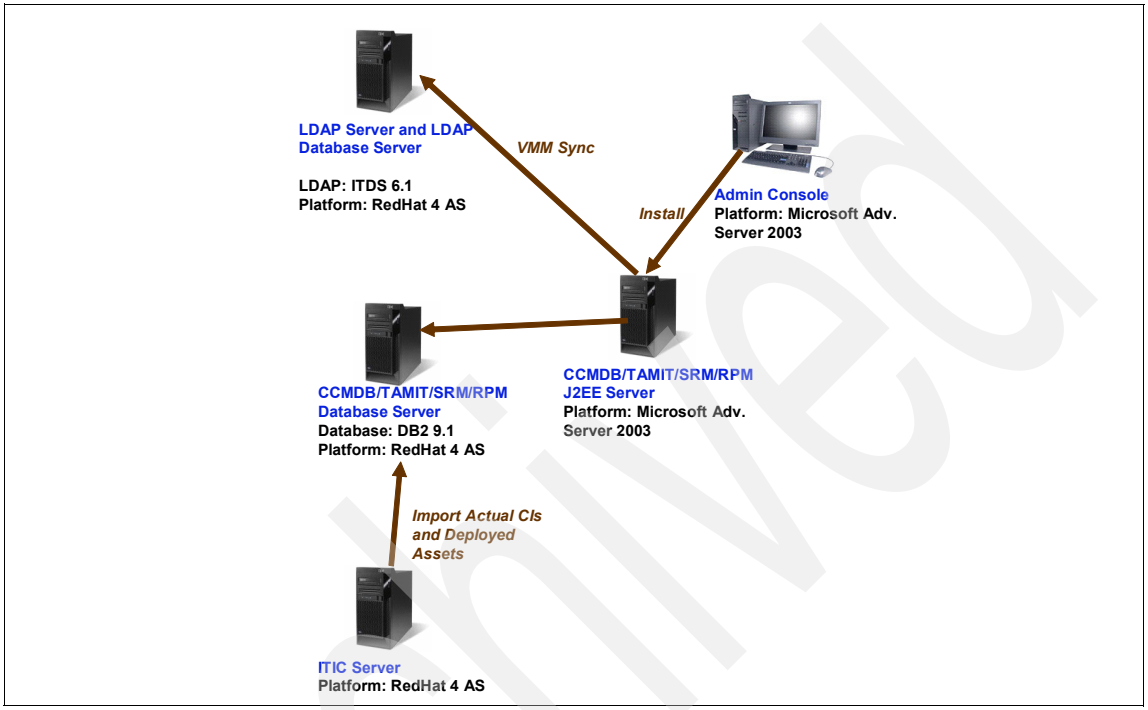


Figure 3-3 Tivoli process automation engine topology

As discussed above, LDAP maintains the user and group entries as well as the user-to-group relationships and user passwords. ISM solution can either use IBM Directory Server, which is the default, or Microsoft® Active Directory® as a directory server implementation. IBM provides the IBM Directory Server within the PMP product packages, like CCMDB or TSRM. Note that if an LDAP server that is supported by ISM products is available in the client environment, then it would make sense to use this server, rather than deploying a separate one.

Note that VMMSync is a cron job that synchronizes the Tivoli Directory Services users and groups with the Tivoli process automation engine users and groups. If the LDAP was Microsoft Active Directory, then LDAPSync would be the cron job.

Important: It is possible to use more than one LDAP server (such as one Microsoft Active Directory and one IBM Directory Server) in this environment, but this does not increase high availability, since with the current VMM implementation if one of the LDAP servers goes down, the authentication mechanism will stop working until the that LDAP server comes back up.

The admin console that is used for installing the components must be a machine running Windows.

The next component is the database server for the ISM solution. Note that DB2 and Oracle are supported for CCMDB, TAMIT, TSRM, and TADDM. TSRM and TAMIT support Microsoft SQL Server® as well. Refer to release notes for individual products for the latest information about supported version and release levels. Similar to our directory server discussion, if a database that is supported by ISM products is available in the client environment (for example, in a storage area network), then it would make sense to use this database, rather than deploying a separate database.

Another key component is the J2EE application server environment, which hosts PMPs and provides run-time services. Currently, CCMDB supports only IBM WebSphere Network Deployment V6.1 FP 13. TSRM and TAMIT also support BEA WebLogic V9.2.2.

The Integration Composer (ITIC) is a data migration component. Its primary purpose in the solution environment is to transfer discovered data into the Actual CI space of the CCMDB database. The Integration Composer (ITIC) is a Java application that can either run on a system of its own, on the same box as the CCMDB J2EE Application Server, or on the TADDM server or another box.

Note: In the current version of the CCMDB, the Integration Composer supports data migration from only one TADDM environment.

3.2.3 Tivoli process automation engine and TADDM combined topology

Figure 3-4 shows the combined Tivoli process automation engine and TADDM topology. Note that Tivoli Asset Manager for IT ITIC adapter pulls data directly from the TADDM database, which means that TADDM does not need to be running for this operation to proceed.

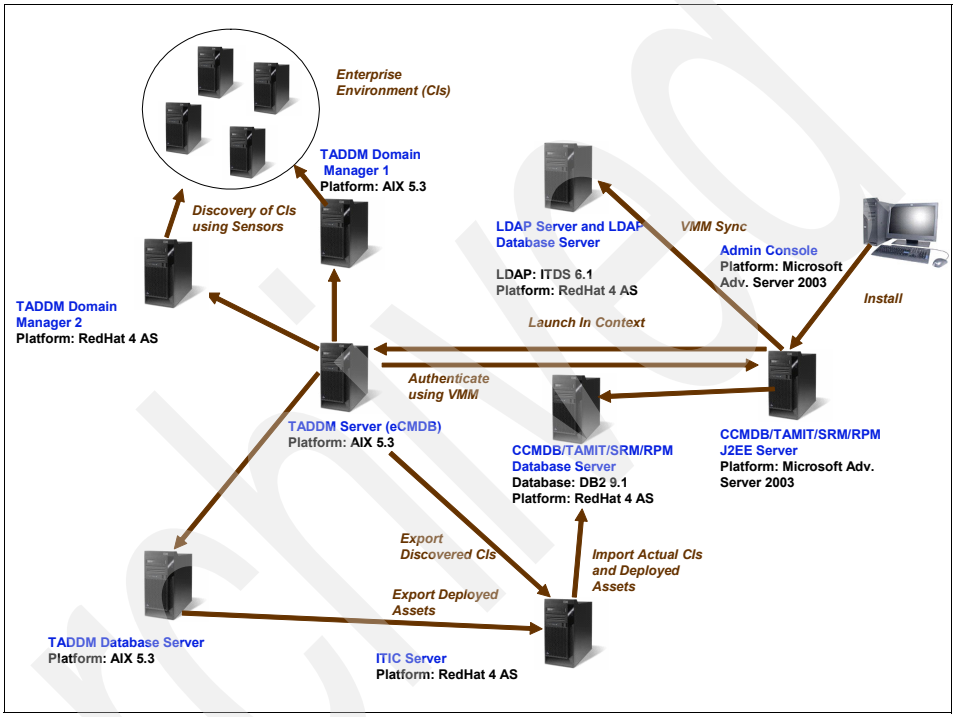


Figure 3-4 Tivoli process automation engine and TADDM topology

The CCMDB ITIC Adapter, on the other hand, uses the TADDM Java APIs to access the data from the TADDM database, so TADDM should be running during this operation.

Extended Security Services must be configured on both the WebSphere server and TADDM server for launch-in-context to work.

3.2.4 ISM overall solution topology

Figure 3-5 shows the ISM overall solution topology. We can see that TPM data can be loaded into TADDM using a Discovery Library Adapter (DLA). This integration also works with Tivoli Configuration Manager (instead of TPM) as well. DLAs for these and other products can be found and downloaded from the Open Platform Automation Library (OPAL) at:

http:///www.ibm.com/software/tivoli/opal

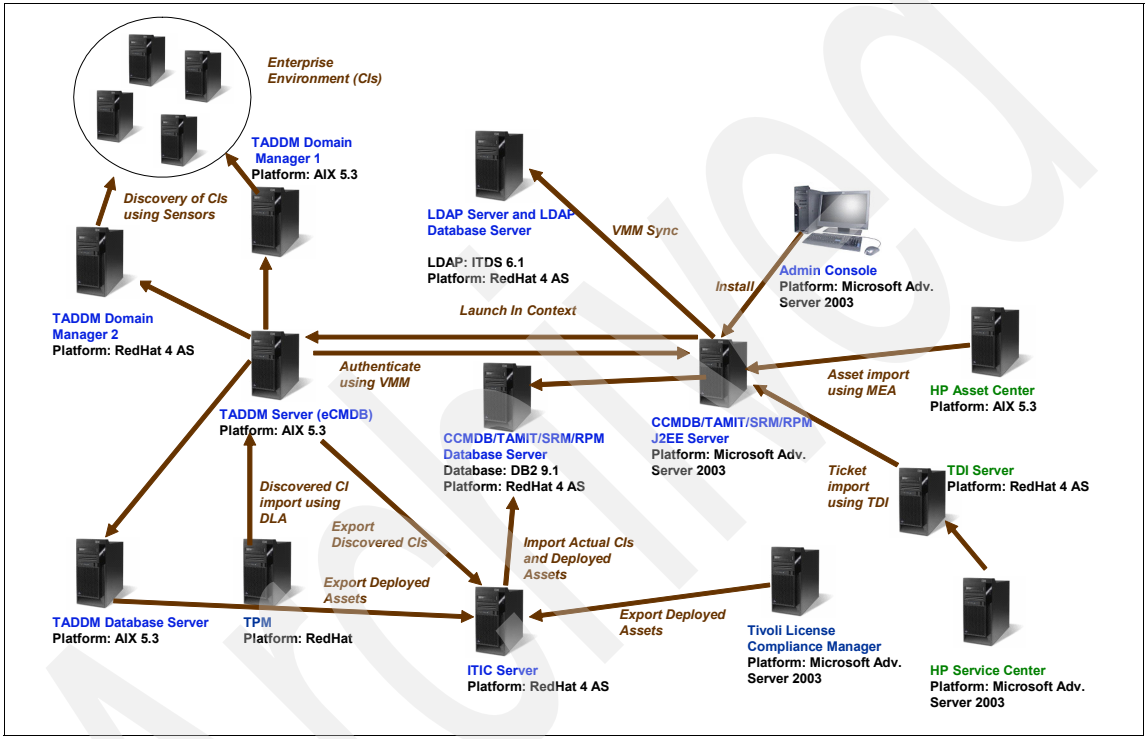


Figure 3-5 ISM overall solution topology

Tivoli Directory Integrator (TDI) is used to integrate a third-party product such as HP Service Center with the ISM solution. In this case, ticket information is imported to TSRM.

Figure 3-5 also shows a typical usage of Maximo Enterprise Adapter (MEA). In this case, MEA is used to import assets from HP Asset Center to TAMIT.

Finally, Tivoli License Compliance Manager is used to export Deployed Assets to ITIC, from where they imported to TAMIT.

It is important to understand that ISM solution lends itself to many integration scenarios, and the ones shown in Figure 3-5 on page 69 are only a subset of them.

3.3 Scalability and availability considerations

This section discusses the scalability and availability considerations for the overall ISM solution.

3.3.1 ISM products scalability considerations

In this book we do not provide absolute numbers of required system configuration and resources, since these can differ a lot depending on the platform being used and environmental circumstances. Therefore, we will give you some guidelines.

The ISM design allows you to scale for discovering and maintaining a high number of systems as well as satisfy a high number of user requests to work with the data that has been discovered within Service Management process implementation like Change or Configuration Management.

The Tivoli process automation engine is implemented with multiple Java Virtual Machines (JVMs) on WebSphere. When the user starts a session against JVMs, the session will be a *sticky session,* so the user keeps the session to the same JVM™ until the session is finished.

We recommend *horizontally* scaling the number of JVMs, since allocating more than 1.5 GB RAM to a JVM will not increase the performance in the same ratio.

As a general rule of thumb, adding 50 concurrent users to the system requires that you add approximately 1 CPU and 1 GB of RAM of physical resources to the system. You also should consider a logical application server (JVM) instance to the system for each 50 users. We also recommend reserving one CPU and 1 GB of RAM for the operating system.

Note that these numbers do not take into account MEA requirements. If the MEA is involved with moderate to small activity, then one JVM should be added and the MEA and cron task transactions should be separated from the user JVMs. If a high level of activity is expected then the MEA should have its own JVM and cron tasks should have its own JVM as well.

Some consideration must be taken regarding the number of concurrent users. For example, Release Management users use the system at different time

frames from the other processes that are normally executed during office hours. This means that the number of concurrent users can be picked up considering maximum peak number in a 24-hour time frame.

Below is an example of a concurrent users estimation for a hypothetical large company. Some of the indicated processes are not included in this book's scenarios but constitute a good example for number estimate.

The peak number of concurrent users is during time frame 8:00 a.m. and 8:00 p.m., so users of Release Management are not added to the maximum access number because users are connected during the night in the time frame 8:00 p.m.–8:00 a.m.

Table 3-1 Example for estimations for concurrent users

	Number of users	Concurrency factor	Concurrent users	Time frame
Incident Management	120	70%	84	7/24
Problem Management	200	15%	30	7/24
Configuration Management	16	25%	4	7/24
Change Management	111	15%	17	Usually 8:00 p.m.–8:00 a.m., sometimes outside business hours
Release management	23	80%	19	Usually outside business hours
Service-Level Management	20	20%	4	8:00 p.m.–8:00 a.m.
Capacity Management	10	30%	3	8:00 p.m.–8:00 a.m.
Availability Management	4	80%	3	8:00 p.m.–8:00 a.m.
Asset Management	30	25%	8	8:00 p.m.–8:00 a.m.
Total concurrent users			153 (excluding Release Management)	

Anticipating that these numbers will increase by 10% over the next year and adding a buffer, we can size this system for 200 concurrent users.

The following two examples should give you a rough understanding of what is needed to scale the ISM environment—one for 200 and the other for 400 concurrent users:

- Intel®-based environment, 200 concurrent users, separation of application and database server:
  - One application server: Four CPUs (Intel Quad), 8 GB RAM
  - Four JVM Instances (four logical application server instances)
  - Database server: Two CPUs, 4 GB RAM
- UNIX®-based environment, 400 concurrent users, two physical application server nodes, separation of application and database server, designed for high availability:
  - Two application servers: Eight CPUs, 16 GB RAM (each)
  - Sixteen JVM instances (16 logical application server instances) spread over the two physical nodes
  - Database server: Eight CPUs, 16 GB RAM

3.3.2 High availability

The goal of a high-availability approach is to prevent single points of failure in the system. The requirement is to always provide the ability to respond to user requests or perform back-end operations like discovery in the environment.

Some high availability considerations of the ISM solution are:

- Use a cluster of HTTP servers behind a load-balancing system in order to receive incoming user requests. In case one of the Web servers is down, the load is spread by the load balancer to the remaining systems.
- Use a cluster of logical application server instances to host the J2EE application. You can spread the load to multiple logical application servers on one physical system or application server instances distributed over multiple physical systems. In case one of the application servers is down or in maintenance, the remaining application servers can take over the load. With respect to high availability, you should at least have two physical systems in your application server cluster in case one system is down or in maintenance.
- LDAP directory server implementations make use of techniques like replication and referral. Replication is the technique to copy data from the master to several subordinate servers. The IBM Directory server even makes use of a concept referred to as *peer-to-peer replication*, which allows you to

define multiple masters. You can replicate the data between those instances. These techniques not only allow you to scale the environment, but also prevent single point of failures.

- For the database system use an external high-availability solution like Tivoli Systems Automation for Multiplatform, High Availability Cluster Multi-Processing (HACMP™), VERITAS Cluster, or a solution recommended by the vendor of the database that you are using in your implementation. As for many customer environment, CCMDB is implemented in a environment where a database solution, for example, using SAN technologies, is just available.

In our example the only component of the overall ISM solution that can be regarded as a single-point-of-failure is the Integration Composer and administration server. In a medium size environment, we could think about Integration Composer and administration server on a Windows Cluster in a Enterprise Environment.

3.3.3 TADDM scalability considerations

This section discusses general guidelines that can be used to estimate the CPU, memory, and disk space requirements for TADDM implementation. These are general guidelines and other factors, such as number of users, will impact server utilization. These guidelines assume that the TADDM and database server are separated, which is the recommended configuration for production environments.

Server sizes

For the purposes of sizing, the following categories of TADDM servers will be used (based on server equivalents, or SEs):

- Small: Up to 1,000 SEs
- Medium: 1,000–2,500 SEs
- Large: 2,500–5,000 SEs
- Enterprise > 5,000 SEs

Domain Manager Server

Domain Manager server is an instance of TADDM (including discovery, analytics, and DB).

CPU requirements (2 GHz (minimum), 3 GHz (or faster) recommended):

- Small: Two processors
- Medium: Three processors
- Large: Four processors

Memory requirements:

- Small: 4 GB
- Medium: 4 GB
- Large: 6 GB

Disk requirements:

- 5 GB minimum: Includes product install and additional space for log files,
- Additional space may be required for DLA books, additional logging requirements, and so on.

Domain Manager Database Server

The database that a domain server uses to store topology and configuration data, which is populated using sensors, DLAs, or the TADDM API. Most customers, particularly large enterprise clients, keep their databases, including TADDMs, on a separate database server.

CPU requirements (2 GHz (minimum), 3 GHz (or faster) recommended):

- Small: One processor
- Medium: Two processors
- Large: Two processors

Memory requirements (minimum):

- Small: 1 GB
- Medium: 2 GB
- Large: 3 GB

Disk requirements

Database disk space requirements include space for the following components:

- System catalog
- Tables
- Indexes
- Logs
- Temp space (for sorts, joins, and so on)
- Backup space

Disk space and disk drive requirements for a database server are not just simply a function of disk capacity. Considerations must be given for I/O operations as well.

- Disk drive requirements: 2 (minimum), 3 (or more) recommended.
- Disk space requirements: See “Initial disk space calculation” on page 76 to calculate estimated disk space requirements.
- Initial disk space is required for the database logs. You must create the TADDM schema: 160 MB.

Enterprise Manager Server (eCMDB)

An instance of the eCMDB is used to link together one or more domain servers.

- CPU requirements: 2 GHz (minimum), 3 GHz (or faster) recommended, Enterprise 4 processors.
- Memory requirements: Enterprise 8 GB
- Disk requirements:
  - 5 GB minimum, including product install and additional space for log files.
  - Additional space may be required for DLA books, additional logging requirements, and so on.

Enterprise Manager Database Server

The database that the eCMDB uses to store topology and configuration data, which is populated using synchronization with one or more domain servers. Most customers, particularly large enterprise clients, keep their databases, including TADDMs, on a separate database server.

- CPU requirements: 2 GHz (minimum), 3 GHz (or faster) recommended, Enterprise 4 processors
- Memory requirements: (minimum), Enterprise 4 GB
- Disk requirements: Database disk space requirements include space for the following components:
  - System catalog
  - Tables
  - Indexes
  - Logs
  - Temp space for sorts, joins, and so on
  - Backup space

Disk space and disk drive requirements for a database server are not simply a function of disk capacity. Consideration must be given for I/O operations as well, as shown in the following list:

- Disk drive requirements: 2 (minimum), 3 (or more) recommended.
- Disk space requirements: See “Initial disk space calculation” on page 76 to calculate estimated disk space requirements.
- Initial disk space required for the database logs: You must create the TADDM schema: 160 MB.

Initial disk space calculation

The following formulas can be used to estimate the initial amount of disk space required for TADDM implementation. These estimates are based on Level 3 discovery type data. Depending on the breadth and depth of data in a customer environment, the disk space requirements can change. Estimates can be based on the number of configuration items (CIs) or server equivalents.

- CI No: Number of configuration items

  About 4,000 bytes per CI
- SE: Number of server equivalents
  - 200 CIs per SE
  - About 800,000 bytes per SE
- CI-RDS: Amount of raw disk space for CIs (without overhead)

  CI No x 4,000
- SE-RDS: Amount of raw disk space for SEs (without overhead)

  SE x 800,000
- TDS: Total disk space (with overhead)
  - Use one of the following formulas:
    - CI-RDS x 1.75
    - SE-RDS x 1.75
  - Includes overhead for temp space, and so on
- CHS: Change history disk space
  - This is the amount of space by which the database will grow weekly, over and above the initial disk allocation, depending on the frequency of discovery.
  - TDS x 1.1 (10%).

Note: The space requirements will increase when additional data is discovered or loaded, or if the versioning feature of TADDM is used.

The following are a disk space calculation examples:

- Example large domain - CI:
  a. CI No: 1,200,000
  b. CI- RDS: 4,800,000,000
     CI No x 4,000
  c. TDS: 8,400,000,000
     RDS x 1.75
  d. CHS: 840,000,000
     TDS x 1.1
- Example large domain - SE:
  a. SE No: 5,500
  b. SE- RDS: 4,400,000,000
     SE No x 800,000
  c. DS: 7,700,000,000
     SE-RDS x 1.75
  d. CHS: 770,000,000
     TDS x 1.1

Using TADDM Domain Managers to increase scalability

In large environments a single TADDM Server cannot effectively manage the discovery of all CIs. In these situations, TADDM supports a feature known as Enterprise CMDB (eCMDB). In this configuration one TADDM Server is set up as an enterprise server, the other TADDM Servers are then configured in the enterprise server as domain server. Domain servers operate as standard TADDM Servers and can run discoveries, display topologies, and run reports. This is a scalability option that can be used to map to organizational or geographical boundaries.

TADDM Server is also referred as the Domain Manager. Depending on CI and organizational or geographical boundary considerations, it is possible to use one or multiple TADDM Servers (Domain Managers).

Figure 3-6 is an example of an Enterprise Domain Manager configuration.

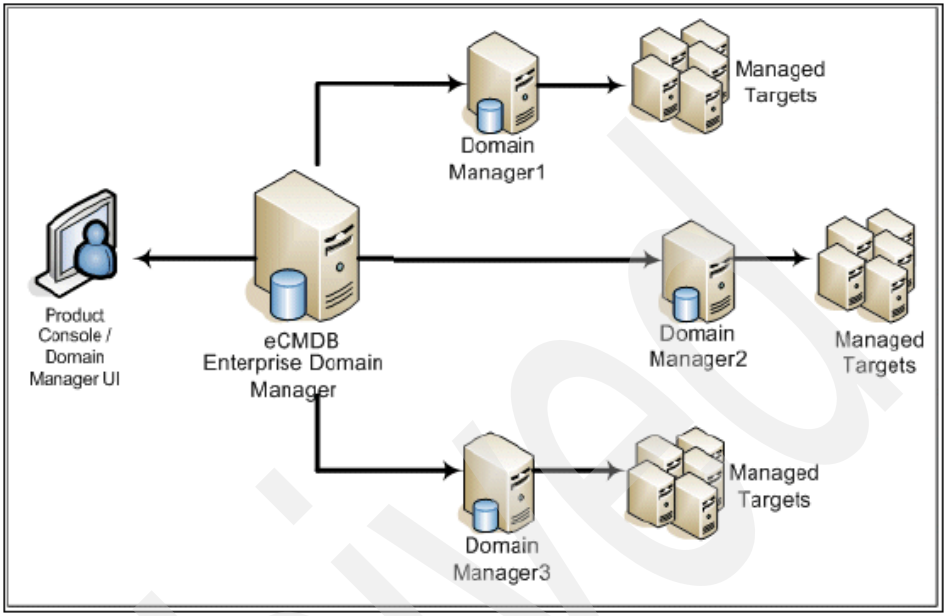


Figure 3-6 Enterprise Domain Manager

From the TADDM Enterprise Domain Manager it is possible to display information about components and business applications within an interconnected graph and display a hierarchical view of the discovery data.

Synchronization is the method by which the eCMDB server pulls in information from the domain servers. Synchronization can be run on demand or on a scheduled basis. The first synchronization for a domain is always a full synchronization. Scheduled synchronization are always incremental. If a discovery is running at the domain the synchronization will fail.

Each Domain Manager maintains CI discovery data in its database. The Enterprise Domain Database synchronizes the discovery information of TADDMs defined in the domain.

The eCMDB database uses the same schema as the remote domains. This means that the version must be kept the same.

In the default mode, all information is synchronized from Domain Managers to Enterprise Data Manager. The configuration file that manages this is on the Enterprise Domain Manager is $COLLATION_HOME/etc domainquery. The

domainquery file defined in the eCMDB server contains the attributes that must be copied on the eCMDB server.

It is possible to define that only a small set of top-level information is synchronized from Domain Managers to Enterprise Domain Manager. This is called *shallow mode*. This reduces the amount of information that needs to be held at the Enterprise Domain Data. The definition of data that can be synchronized is defined in the $COLLATION_HOME/etc/domainquery.shallow file on the TADDM Enterprise Domain Server. The file can be customized.

In our environment the path of files domainquery and domainquery.shallow is /opt/IBM/cmdb/dist/etc.

The domainquery file indicates to the eCMDB server which attributes will be copied locally to the eCMDB server. The deep synchronization is in the default mode. This includes a single line with SYNC_ALL_ATTRS, meaning that all CI attributes are retrieved and synchronized to the eCMDB.

```
[root@TADDM etc]# cat domainquery
SYNC_ALL_ATTRS
[root@TADDM etc]#
```

Figure 3-7 Default configuration of domainquery file

The domainquery.shallow is the file that can be configured to synchronize between Domain Managers and Enterprise Domain Database. It is provided with a default configuration that can be modified based on your requirements. This allows transferring of only CI information required for promotion onto the Enterprise Domain Database.

Configuration items discovered have many attributes and relationships. In some cases it is not required to manage and control every attribute on every CI.

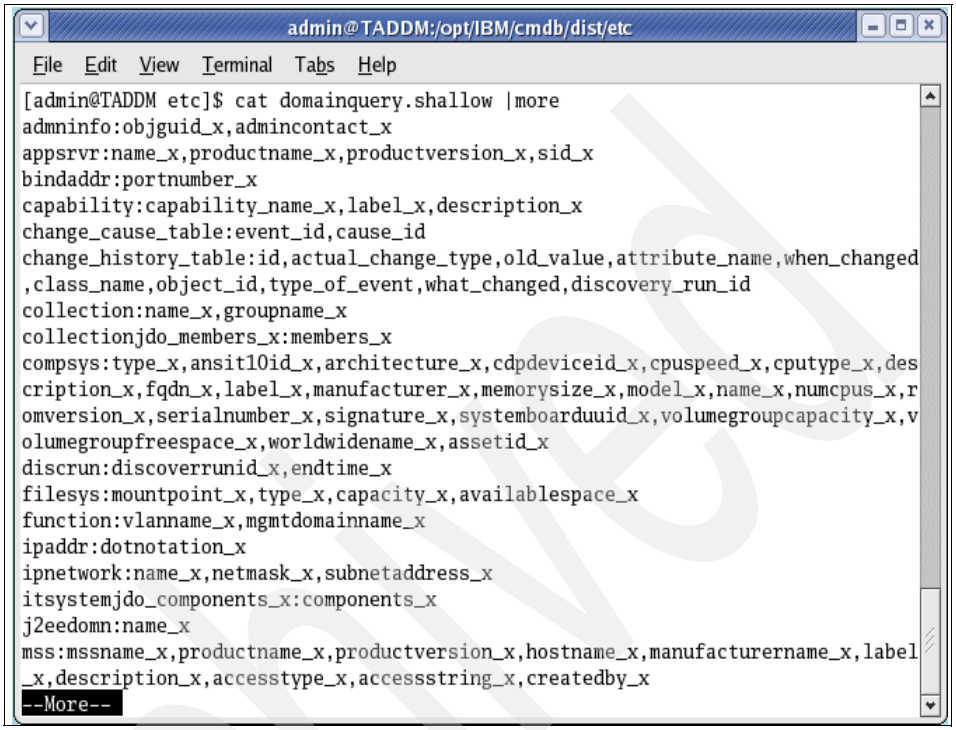


Figure 3-8 An example of domainquery.shallow configuration

Tivoli Provisioning Manager (TPM) scalability considerations

The scalability of Tivoli Provisioning Manager and Tivoli Manager for Software is based mainly on the following configuration options:

- Hierarchical software: This configuration requires the installation of remote depots, added to distribute software packages, for example, in every branch office. This is the architecture where the package from the Definitive Software Library is sent to the distributed depot and from the depot to every target of the distribution.
- Peer-to-peer configuration: This configuration does not require an additional depot server in the customer environment, but uses peer-to-peer technology. In this way the Tivoli Common Agent works as a miniature depot server. This means that it can hold copies of distributed files in a cache and act as sources for this file during downloads by their neighbors.

The software repository of TPM (in a Release Process Manager Configuration called the Definite Software Library) is generally is defined on the same system

on which TPM is installed. If a large amount of software packages need to be defined in a large environment, a storage area network can be used as a dedicated storage area or software package.

Tivoli Provisioning Manager starting from V5.1.1 supports read-only Lightweight Directory Access Protocol. This means that TPM uses the information in the user registry to authenticate users, but it cannot modify the user registry because it works in read-only mode. The default installation sets up authentication based on user accounts defined on the operating system. This type of configuration should be used only for evaluation purposes. We strongly recommend using an LDAP binding and using the LDAP defined at the client environment if supported by TPM and CCMDB solution.

4

How Asset Management and Configuration Management work together

This chapter explains the concepts of Asset Management and Configuration Management and how they works together to improve quality of services and fit business needs.

This chapter discusses the following topics:

4.1 Asset Management

Asset Management encompasses the entire asset life-cycle management, from request until the retire phase. Asset Management's focus is to control financial and regulatory information of assets. An asset is obtained from a supplier through a procurement process.

ITL definition of an asset: Any resource or capability, such as management, organization, process, knowledge, people, information, applications, infrastructure, and financial capital.

Figure 4-1 represents an asset life cycle.

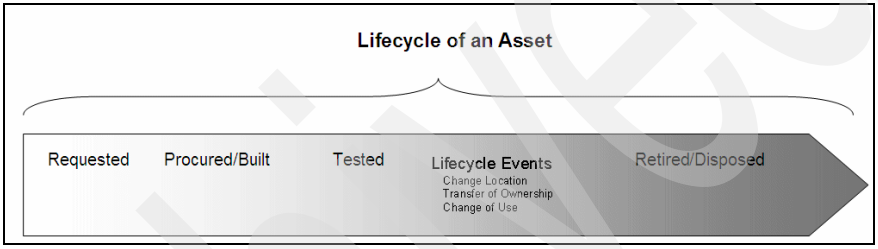


Figure 4-1 Asset life cycle

Asset Management manages asset information from initial creation (such as receipt from suppliers), through all life-cycle events (such as change of location, transfer of ownership, change of use) until eventual retirement or disposal. This includes identifying, collecting, maintaining, controlling, and reporting inventory and financial information about information technology (IT) assets throughout their life cycle.

Asset Management also includes:

- License Management (including software license compliance)
- Lease and maintenance administration of each asset
- Inventory Management (includes physical components and specifications)
- Allocation of available assets to meet approved requests
- Physical logistics (such as transportation) of assets
- Retirement of outdated assets

- Triggering requisition for the procurement of additional assets (for example, if a policy of maintaining minimum inventory stock levels for a standard, frequently needed asset item is in place)
- Financial life cycle of assets (including valuation)

The following processes *are not* included in the Asset Management:

- Risk Management.
- Contract and Supplier Management (including procurement and Supplier Management).
- Configuration Management (logical relationships).
- Managing the security of an asset (Facilities Management, Security Management).

4.2 Configuration Management

Configuration Management enables and IT organization to identify, control, maintain, and account for all configuration items (CIs) in the IT infrastructure.

Configuration items are IT organization components that are under control of Configuration Management. All of the following are candidates for CIs:

- Request for change
- Service level agreement
- Procedures
- Hardware
- Software
- Formal documentation (such as process documentation)
- Buildings
- People
- Other items that must be controlled

ITL definition of a CI: Any component that must be managed in order to deliver an IT service.

Information about each CI is recorded in a configuration record within the Configuration Management System and is maintained throughout its life cycle by Configuration Management. CIs are under the control of Change Management.

The Configuration Management Database (CMDB) holds all CI information and their attributes, as well as the relationship between CIs and cross-link CI information to all ITSM process.

IBM Service Management (ISM) process solutions provide a product named Change Configuration Management Database (CCMDB) solution. CCMDB incorporates a consistent data model and data layer implementation and includes a framework for discovery or resources and its relationships. The IBM CCMDB solution provides an Information Technology Infrastructure Library (ITIL)-aligned implementation of a Configuration Management Database.

4.3 Intersection between assets and configuration items

There is an intersection between these concepts at the operation phase. Assets have a life cycle that spans procurement/development through operations and retirement. Once an asset is in operation within the IT infrastructure, it may also be considered as a configuration item.

Configuration Management focuses on configuration items in the IT infrastructure (during the operation phase) and their relationships. Figure 4-2 describes this scenario.

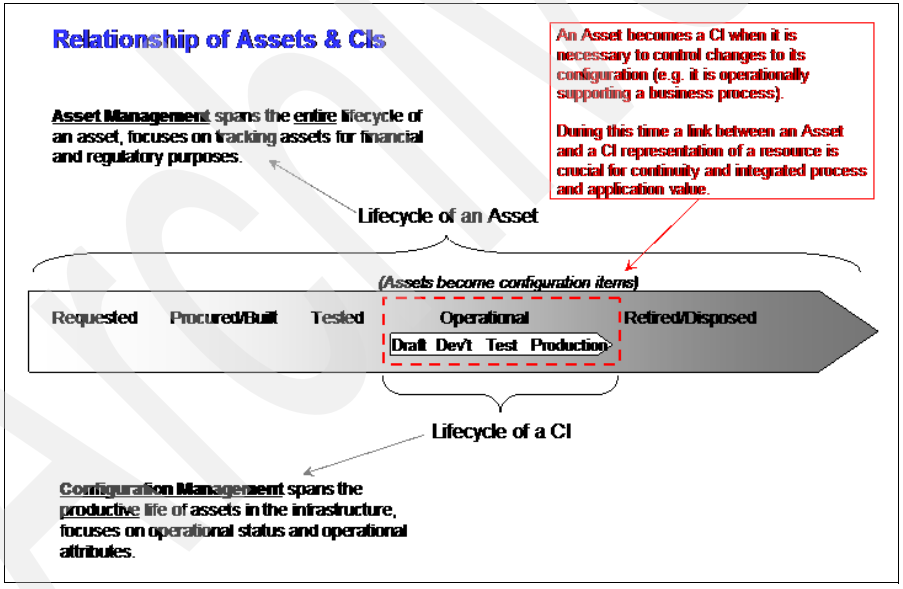


Figure 4-2 Assets become configuration items

After a useful lifetime, this CI is removed from operational status and becomes part of a set of spares or possibly retired and disposed. Thus, an asset has a longer life cycle than a CI.

When using assets only:

- There is no requirement to implement Configuration Management, so you do not need to view CIs and their relationships.
- You do not need asset relationships in any of their process implementations (including change, incident, problem).
- You do not need all of the class types (from the Common Data Model in TADDM) that CIs provide.

When using CIs only:

- You want to manage operational assets/CIs, not full the asset life cycle.
- You do not need license compliance.

Assets and CIs according to ITIL V3: ITIL V3 CMDB has been redefined to become a Configuration Management System (CMS) that contains several federated CMDBs. It is also now part of a new process: Service Asset and Configuration Management (SACM). The SACM process replaces the Configuration Management process in V2 with a broader scope. For example, financial accounting aspects of managing assets were considered separate from Configuration Management in V2. ITIL V3 combines all V2 Configuration Management tasks with some of the V2 Financial Management tasks to form the new Service Asset and Configuration Management (SACM) process. So asset and CI integration is more emphasized in ITIL V3.

4.3.1 CI considerations

In CCMDB 7.1.1, you import Actual CIs from IBM Tivoli Application Dependency Discovery Manager (TADDM). TADDM is the discovery component for CCMDB 7.1.1 and a prerequisite of CCMDB.

Actual CIs use the Common Data Model that is imported from TADDM.

For Authorized CIs we strongly recommend that you create your own authorized CI hierarchy (also called separate Authorized CI Hierarchy or Authorized Data Model) and show which classifications are parent/child in your IT environment, instead of using the same hierarchy for both Actual CIs and Authorized CIs (called dual class hierarchy). Using dual class hierarchy might produce too much (and unnecessary) detail for an effective Change and Configuration Management process. Figure 4-3 shows the relationship between Actual CIs and Authorized CIs.

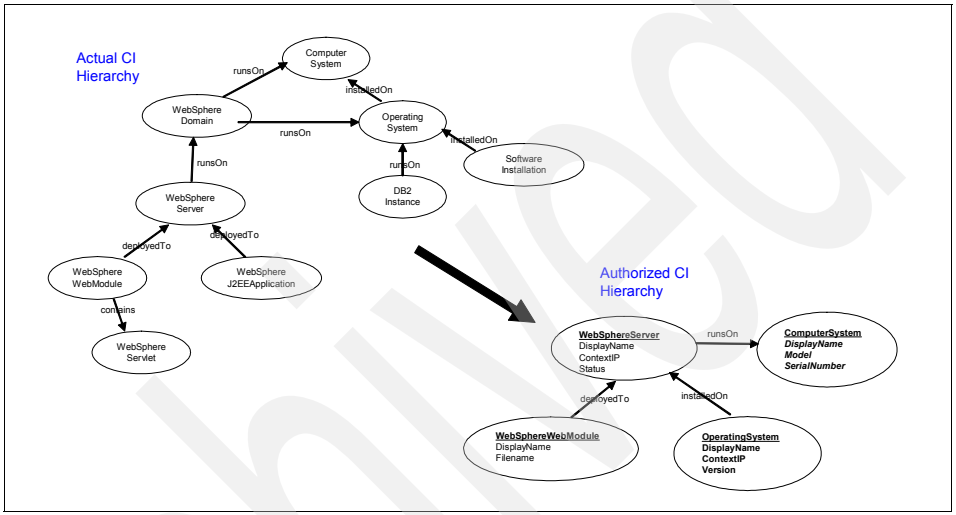


Figure 4-3 Actual to Authorized CI Hierarchy

Note: More details about TADDM can be found in *IBM Tivoli Application Dependency Discovery Manager Capabilities and Best Practices*, SG24-7519.

4.3.2 Asset considerations

Authorized assets are imported from one source of data (or created manually), and then the life-cycle process is used to keep them up to date. Deployed assets are imported from another source of data.

There is no permanent link between a deployed asset and an asset. They are linked using a link rule when you compare them using a reconciliation task.

There are three types of deployed assets:

- Computers
- Network devices
- Network printers

There is no relationship between assets or deployed assets.

4.4 Process layer view assets and CIs

In this section we discuss the process of asset and CI linking. But first we provide a conceptual overview of the CCMDB Data Layer.

The CCMDB Data Layer contains three data spaces that hold CIs, process artifacts, and relationships between these objects to provide everything from a dependency mapping of the discovered environment to a specification of authorized CIs. It defines the specific aspects and characteristics of CIs that you wish to tightly control and manage.

The CCMDB supports the Tivoli Common Data Model (CDM) across all three data spaces. The CDM is a logical information model that is used to support the sharing of consistent data definitions and the exchange of data between Tivoli management products, concerning managed resources and components of a customer's business environment.

Figure 4-4 shows a diagram that reflects the three CI data spaces of the CCMDB 7.1 solution.

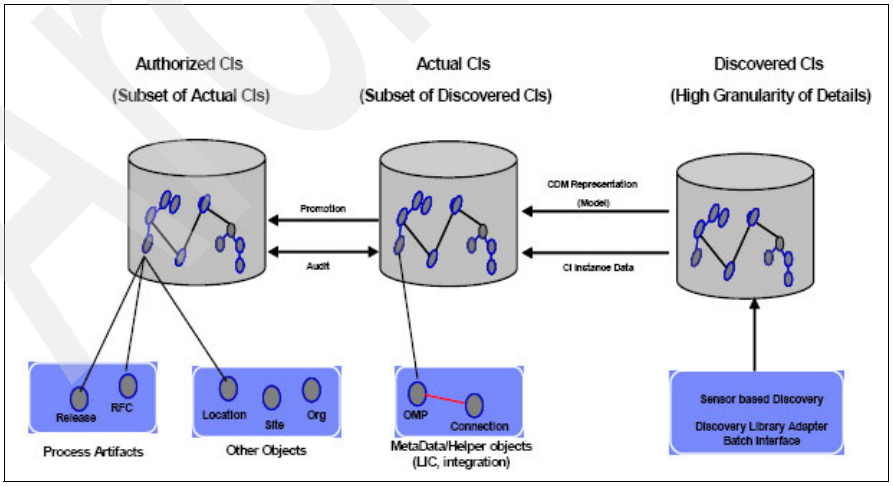


Figure 4-4 CCMDB 7.1.1 Data Layer

We now describe how assets and CIs are organized using this structure represented above. In Figure 4-5 we show the relationship between assets and CIs.

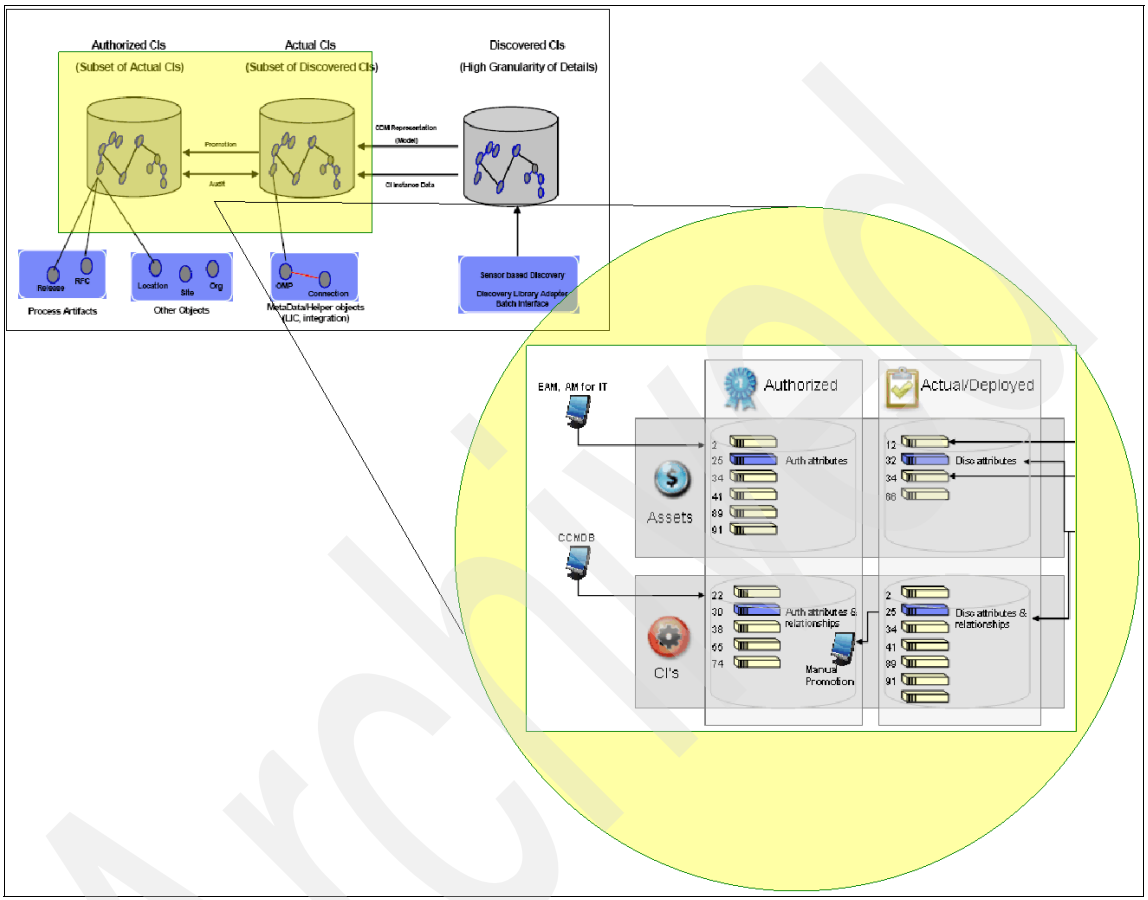


Figure 4-5 Asset and CI relationship

In Figure 4-6 we show the elements responsible for the complete asset and CI management process.

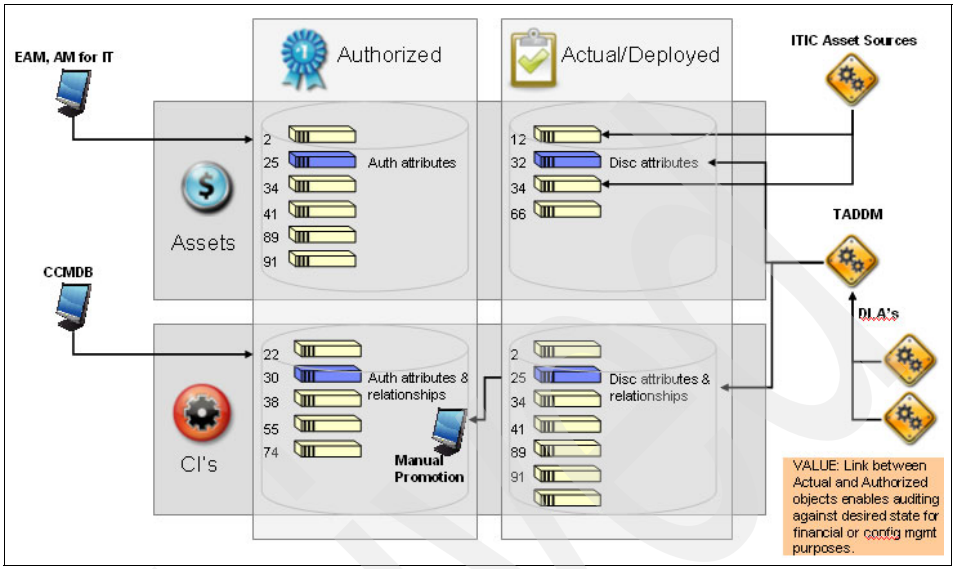


Figure 4-6 Asset and CI relationship: Process overview

In the next figures we explain in more detail each step of the asset and CI relationship management process. Figure 4-7 represents the scenario above, focusing on TADDM (for discovery).

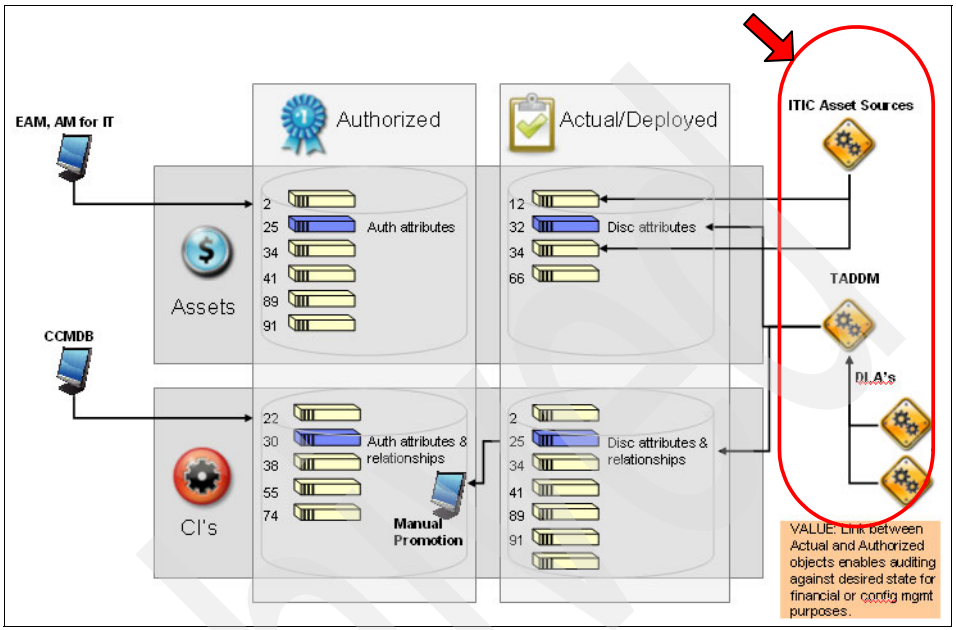


Figure 4-7 Asset and CI relationship: TADDM (discovery phase)

In the discovery phase we have a tool named TADDM responsible for discovering the elements of the IT infrastructure. TADDM discoveries collect configuration information about your IT infrastructure, which includes deployed software components, physical servers, network devices, virtual LAN, and host data used in a runtime environment.

The Discovered CIs data space of the CCMDB contains information discovered from the heterogeneous IT environment. This includes CIs and relationships discovered using sensor discovery and those loaded through Discovery Library Adapters (DLAs).

These discovery capabilities provide an accurate dependency mapping between CIs and include many different types of relationships, such as logical, physical, and application topologies.

To optimize the breadth and depth of information that TADDM gathers, there are some setup tasks that are required within TADDM and your environment. You must provide the following required information for discoveries to run in your environment:

- The discovery scope: Typically a valid IP range, a subnet, or a specific component, the discovery scope signifies the span of the discovery process.
- Access lists: Access lists specify the read-only access credentials needed to discover and query the components for their appropriate configuration attributes and dependencies.
- Schedule: TADDM's discovery process can be executed on demand, as part of a schedule, or driven by externally triggered events

Discovering your environment is an interactive process. A full understanding of your IT infrastructure is built up over successive discoveries.

IBM Tivoli Integration Composer (ITIC) is responsible for providing a filter that determines which CI instances are brought over into the Actual CI data space. For each CI Type specified, all instances of that CI Type in the Discovered CI space are copied into the Actual CI space.

ITIC considerations:

- In production environments, run ITIC on a separate machine from TADDM and Tivoli process automation engine for best performance.
- Make sure that you only activate the CI types that you need to import, and ensure that the depth is set to the lowest value that brings over all the data that you need.
- Always schedule ITIC imports, TADDM discoveries, and TADDM bulk loads at separate times and where possible during non-business hours.
- Remember that the first time that ITIC runs it might take a long time. Subsequent runs (delta runs) are much quicker.

At this point it does not matter whether collected elements are identified as assets or CIs.

Note: For more details about TADDM see *IBM Tivoli Application Dependency Discovery Manager Capabilities and Best Practices*, SG24-7519.

Figure 4-8 represents our scenario focusing on the actual/deployed phase.

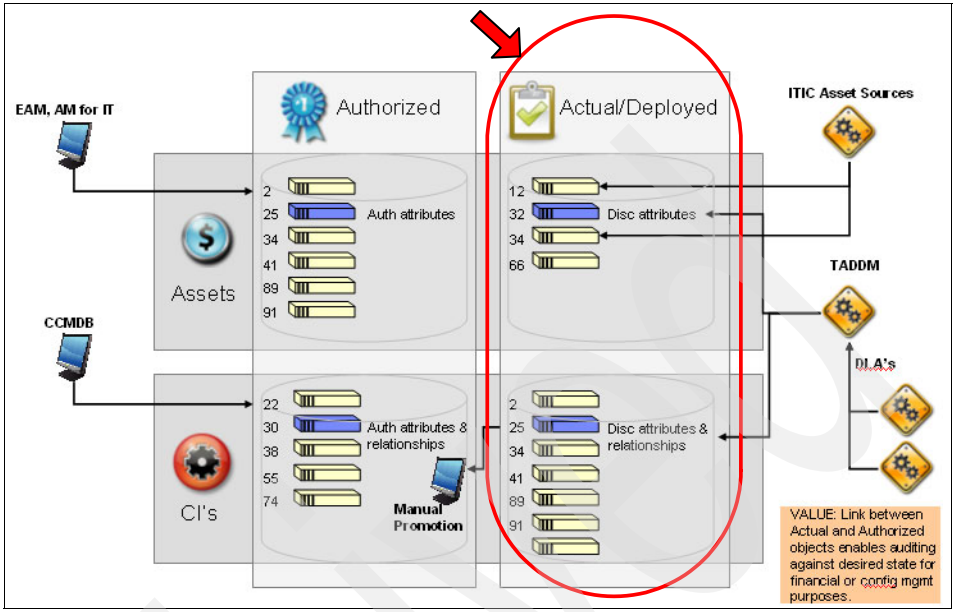


Figure 4-8 Actual data space

In the actual/deployed phase, we use ITIC to defined a filter to bring CI data from the discovery data space (shown in Figure 4-4 on page 89) to the Actual CI data space.

The data copied into the actual data space is a sub-set of the data existing in the discovery space. We use the ITIC filter to determine how deep the level of data should be imported.

The actual data space holds data from the real TI environment. This data can be used to make a comparison between the actual data space and the authorized data space (called reconciliation or audit).

Figure 4-9 represents our scenario focusing on the authorized phase.

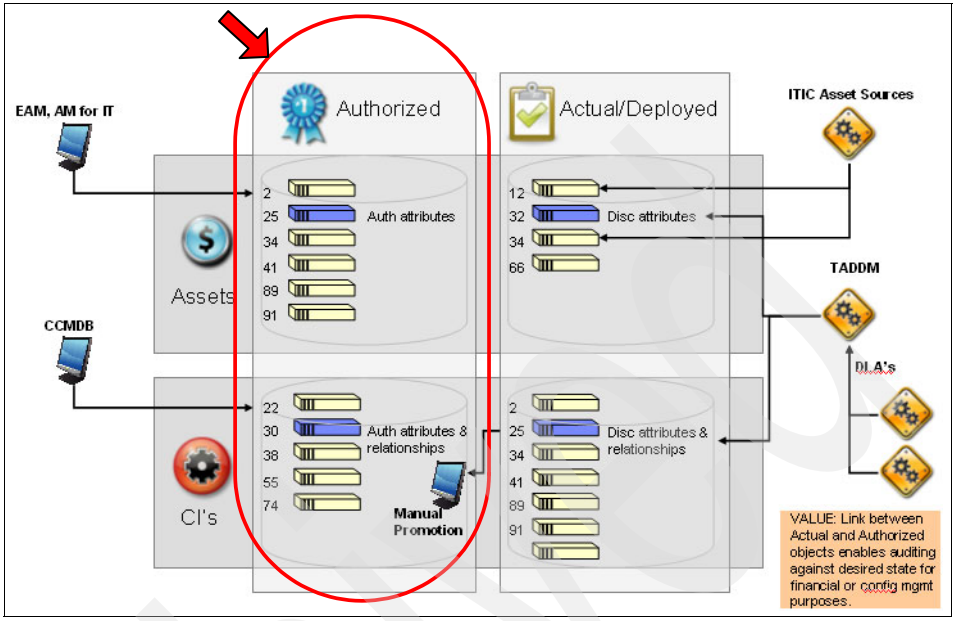


Figure 4-9 Authorized data space

We can specify which CIs should be copied from the Actual CI data space into the Authorized CI data space. The process of creating authorized CI instances from Actual CI instances is called *promotion*.

In the authorized data space, the CI data is managed by Service Management processes such as Configuration Management, Change Management, Release Management.

In the authorized space, the asset data is also managed by the Asset Management and Financial Management processes.

An Authorized CI is typically a simple definition with a small number of relationships and attributes. A computer system CI can be quite complex with a large number of attributes, and a fair number of relationships with other objects. However, an Authorized CI for a computer system would contain only the attributes/values and relationships to objects that you care to bring under change control.

All CI types are grouped within the Common Data Model (CDM) into entities that correspond to items in the real world.

Note: CDM is a representation of CMDB entities, their relationships, and their semantics.

Each CI type in the CCMDB contains:

- Hierarchy

  Classes in the CDM are arranged into a single-inheritance hierarchy, which enables attributes to be shared among classes.

- Attributes

  At the most basic level of granularity, for example, an attribute for an operating system is version.

- Relationships

  Relationships are links between two CIs, for example, a link between an operation system and a computer system.

- Naming rules

  These define the ways in which each CI type is identified, in order to foster identification of entities in the CMDB. This consistency is accomplished by using naming roles.

Note: Find more details about the authorized phase see the *Deployment Guide Series: IBM Tivoli CCMDB Overview and Deployment Planning*, SG24-7565.

CI linked to asset: Process layer view

Figure 4-10 shows the links (see arrows) between CI and the asset in the authorized and actual phases.

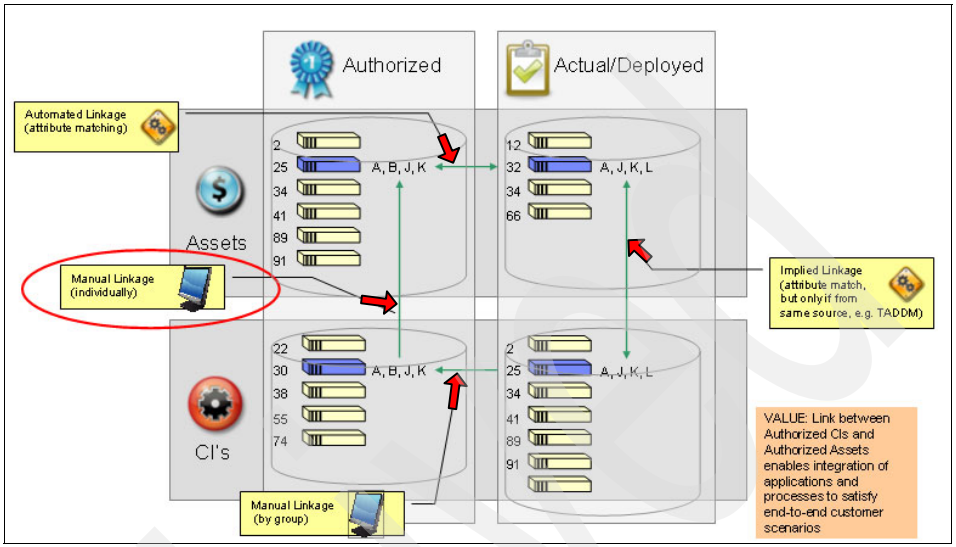


Figure 4-10 CI and asset relationship

The link between the CI and the asset allows you to implement end-to-end Information management for the elements that comprise your IT infrastructure. How can we can we create this link? We discuss that in 4.5, “How to link an asset to a CI” on page 100.

Figure 4-11 shows the asset and CI aspects.

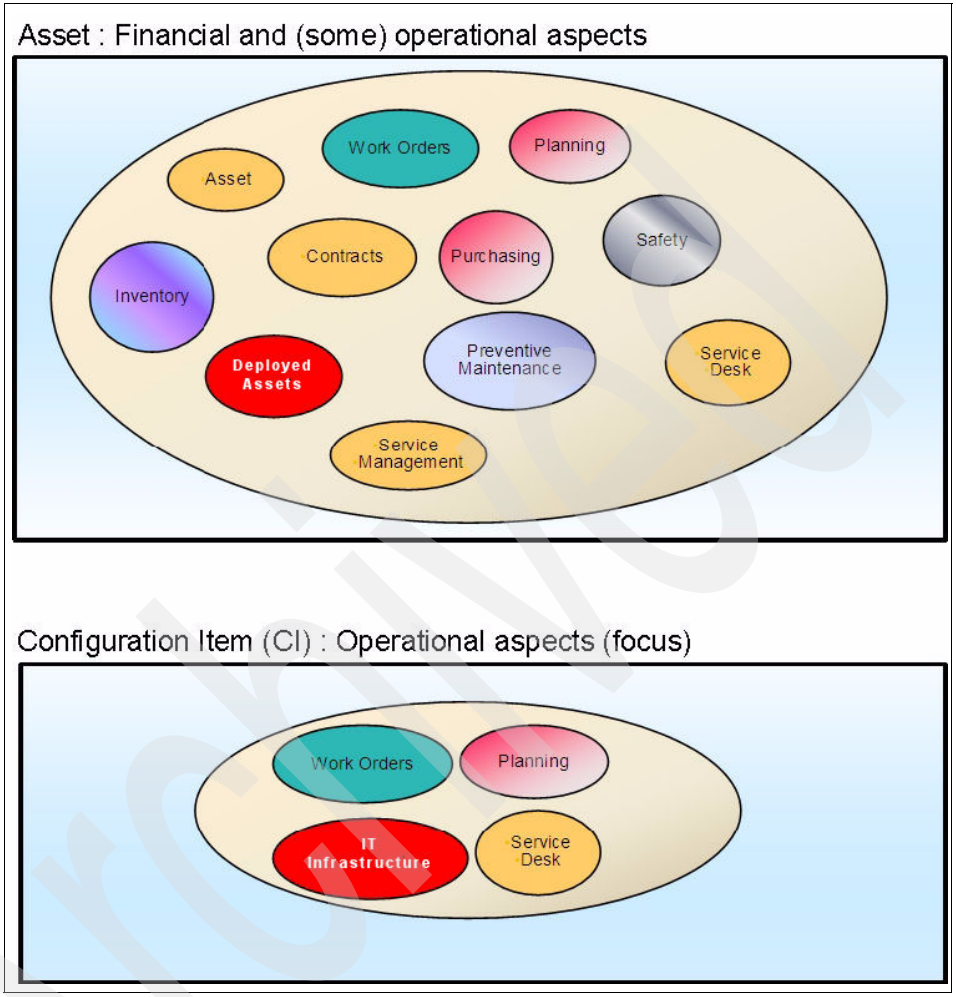


Figure 4-11 Asset and CI aspects

Assets can be used to model the end-to-end life cycle of a managed entity. They are used to manage both financial aspects and some operational aspects.

You must create an asset for a *managed thing* if:

- It has financial aspects, for example, purchasing, contracts, and inventory.
- It has some specific operational aspects:
  - Meters
  - Failure codes
  - Preventive maintenance
  - Safety

You can manage some operational aspects of a managed entity by creating it as an asset:

- Incidents, problems, service requests, change, release
- Comparing discovered versus managed attributes using assets and deployed assets
- SLAs
- Auditing discovered and managed attributes

CIs provide the following features:

- A vast number of CI types and their relationships can be discovered by TADDM and imported into CCMDB 7.1.1. This data provides support for impact analysis that can aid in Change Management, Incident Management, Release Management, and Problem Management processes.
- Configuration Management can be used to reconcile CIs and Actual CIs. A customer can define which CIs and attributes are to be reconciled. Automated tasks can be scheduled for these audits and reports can be generated.
- CIs can be used for Configuration Management and operational management, but not for Financial Management.
- The Change and Release process can provide automation of implementation tasks, for example, software distribution tasks. Infrastructure services are provided so that customers can develop their own automation.
- Change Management provides impact analysis.
- Incident and Problem Management processes provide Launch-in-Context to TADDM, which helps in their resolution.
- The Configuration Management process helps audit and control updates to CIs.

By combining these two elements (assets and CIs) with ISM, you can implement a complete end-to-end Service Management Solution.

4.5 How to link an asset to a CI

In this section we provide several figures that shows you how to link an asset and a CI using the configuration items and assets applications.

1. Log in to CCMDB (Figure 4-12).

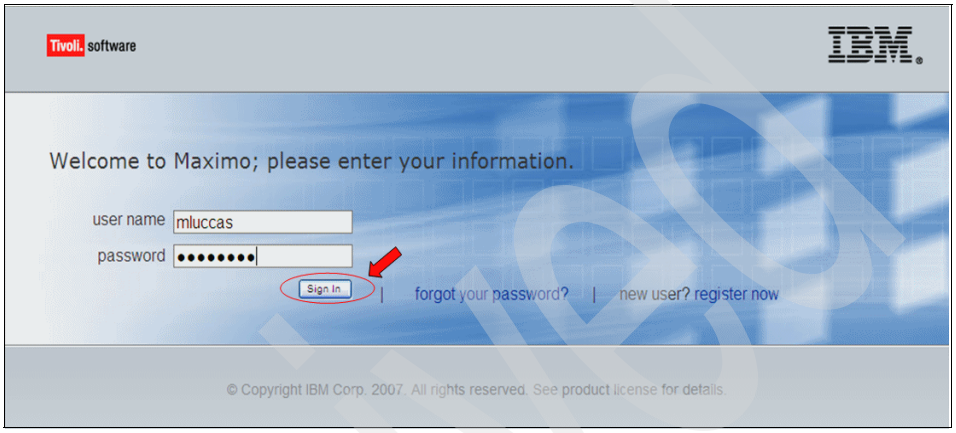


Figure 4-12 Linking asset and CI (1 of 9)

2. Select **Go To** → **IT Infrastructure** → **Configuration Items** (Figure 4-13).

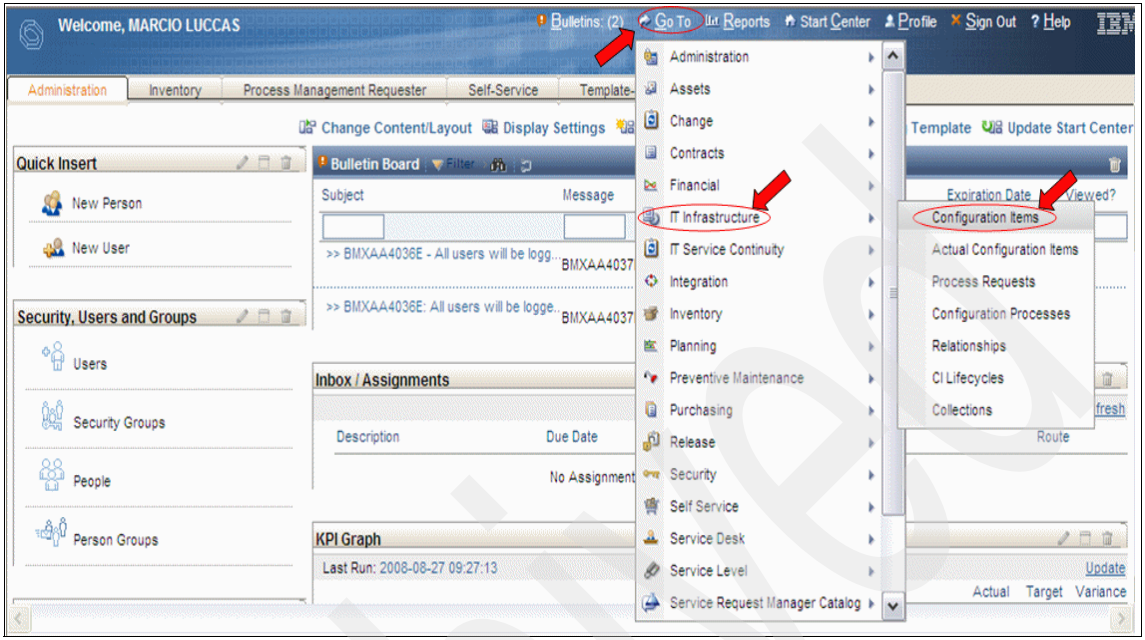


Figure 4-13 Linking asset and CI (2 of 9)

3. Locate the configuration item that you want to link to an asset. See Figure 4-14 and Figure 4-15 on page 102.

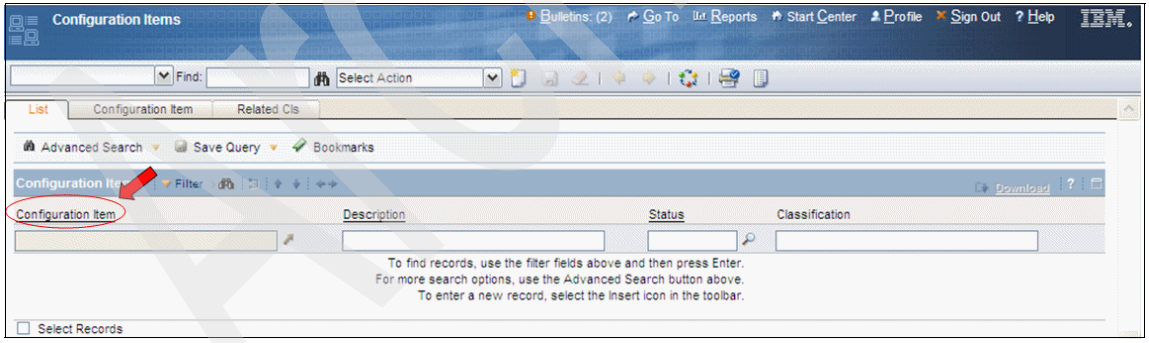


Figure 4-14 Linking asset and CI (3 of 9)

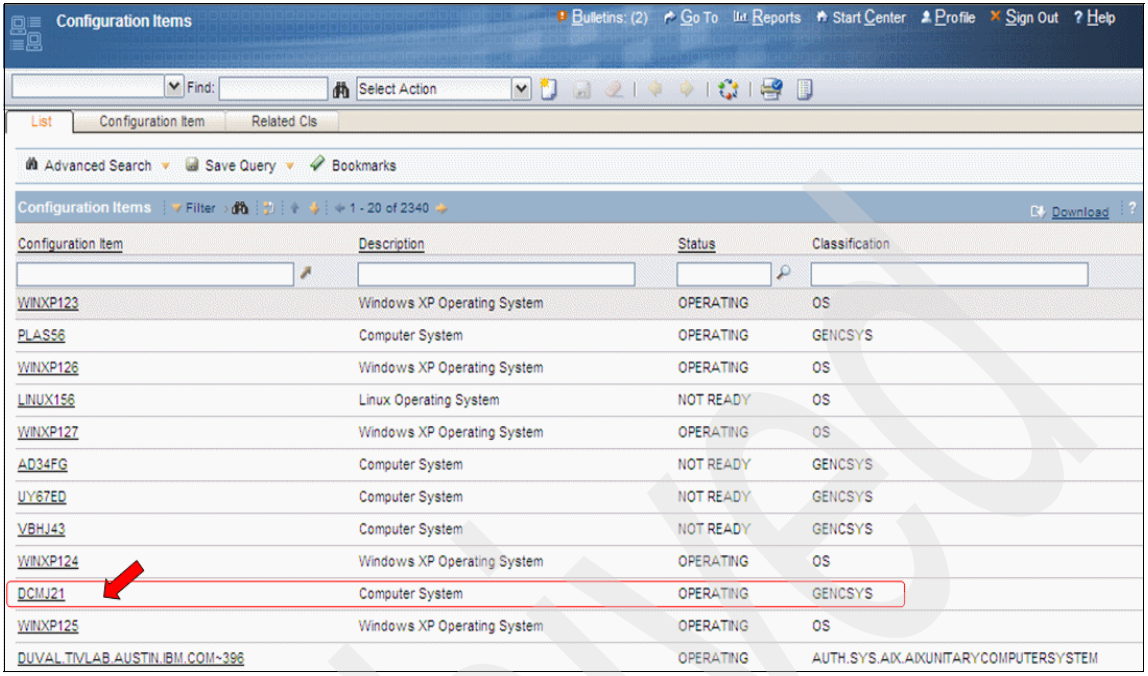


Figure 4-15 Linking asset and CI (4 of 9)

4. Now we need to go to the asset information. Click the arrow icon next to the Asset field and select **Go To Asset**. This launches the assets application in context. See Figure 4-16.

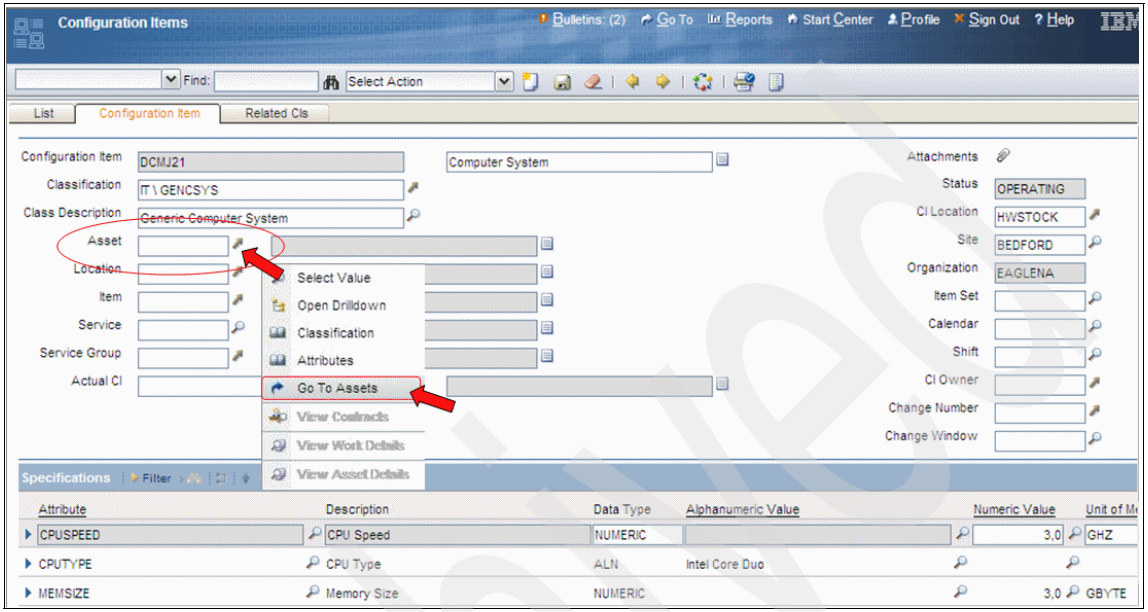


Figure 4-16 Linking asset and CI (5 of 9)

5. In the assets application, locate the asset that you want to link to the CI. See Figure 4-17 and Figure 4-18 on page 104.

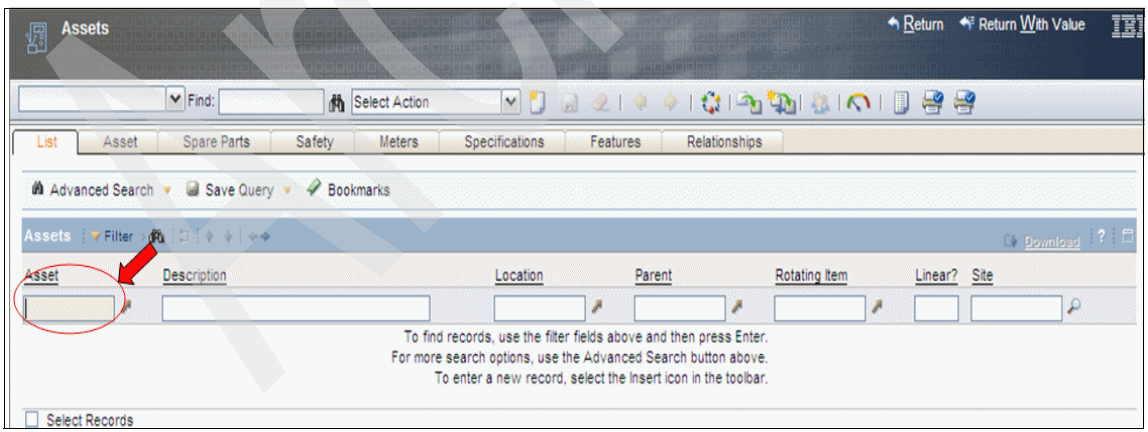


Figure 4-17 Linking asset and CI (6 of 9)

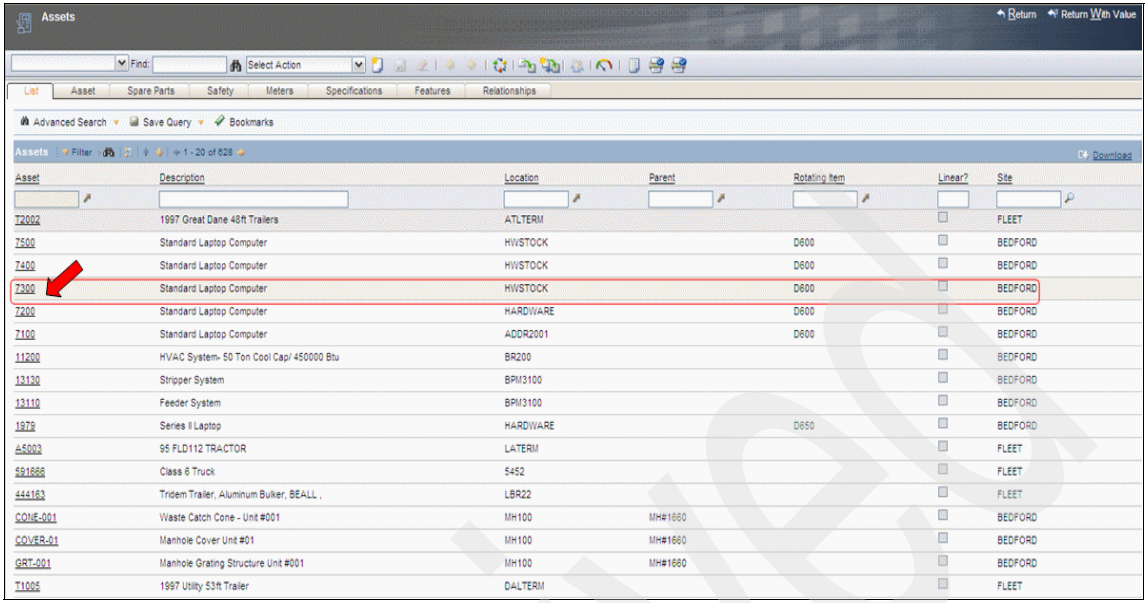


Figure 4-18 Linking asset and CIs (7 of 9)

6. Click **Return with Value** (Figure 4-19).

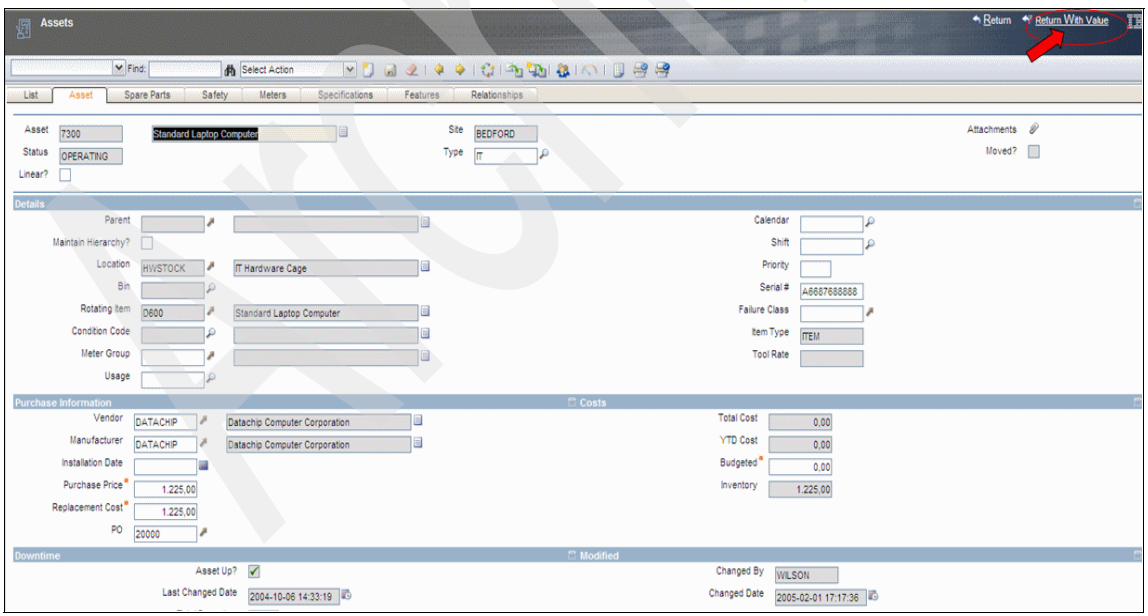


Figure 4-19 Linking asset and CIs (8 of 9)

7. In Figure 4-20 you can see that the CI DCMJ21 is linked to the asset 7300. Save this record.

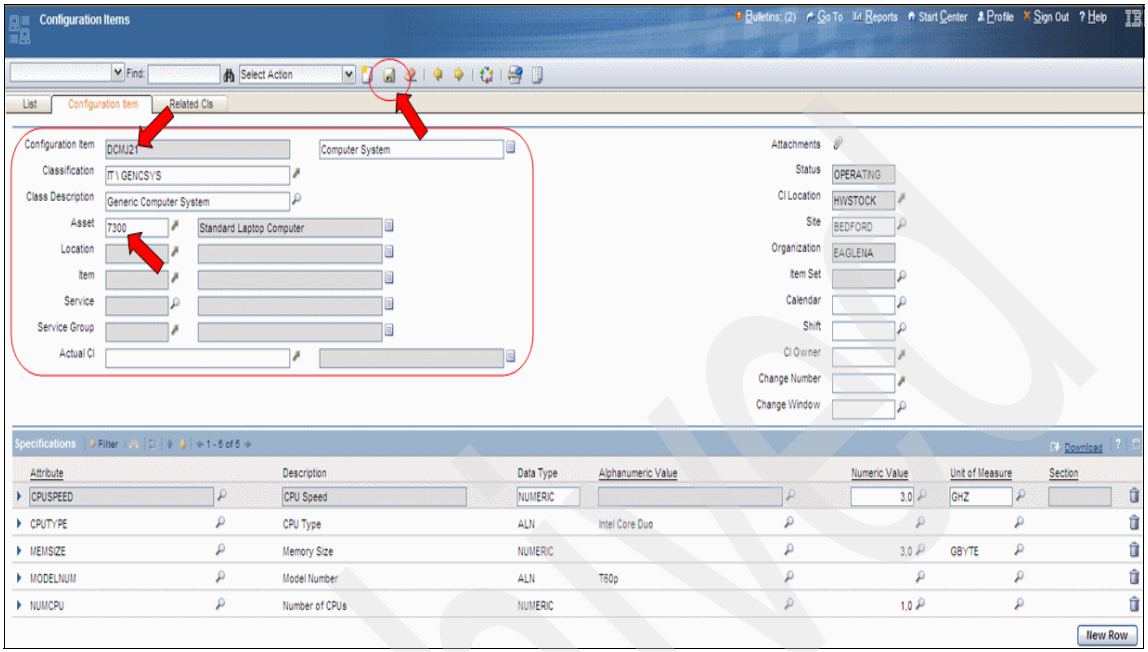


Figure 4-20 Linking asset and CIs (9 of 9)

In this section we showed you how to link a CI (CMJ21) to an asset (7300). In the following sections we show you how you can use this capability to link an asset and CI information in change, release, and Service Desk modules.

4.6 Relationship between an asset and CI

Assets can be used to model the overall life cycle of a managed entity. During certain asset life-cycle states, an asset can be managed as a CI when it is operationally supporting a business process. To manage an asset as a CI, you should link the CI to the asset using Configuration Management in CCMDB, as shown in 4.5, “How to link an asset to a CI” on page 100.

Figure 4-21 shows you the Asset Management life cycle and Configuration Management relationship.

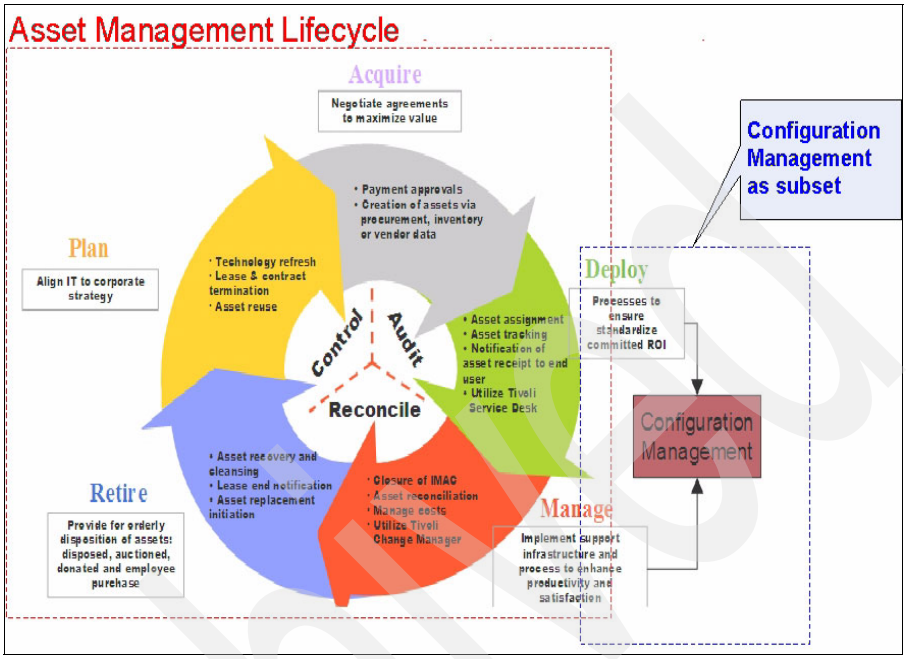


Figure 4-21 Asset Management life cycle and Configuration Management

CCMDB and TAMIT provide best-of-breed integration between Asset Management and Configuration Management. Both CCMDB and TAMIT can be installed on a single set of middleware and share the same application server, database, users, policies, workflow, and so on. All TAMIT and CCMDB applications can run in a single user interface. It is very easy to navigate from assets to CIs and vice versa.

4.6.1 Assets and CIs in Change Management

Change Management allows a change process to be targeted at a CI or an asset. When a CI is linked to an asset, the change process is automatically populates the asset field if the target is a CI and vice versa.

The following figures show us how this linkage works in the Change Management module. To show you this functionality, we will use the same CI and the same asset that were linked in the previous section: CI = DCMJ21, mobile computer and Asset = 7300 mobile computer.

Note: Only CIs are managed under Configuration Management. Therefore, to manage an asset through Configuration Management, it needs to be linked to a CI.

1. Access the Change Management module as shown in Figure 4-22.

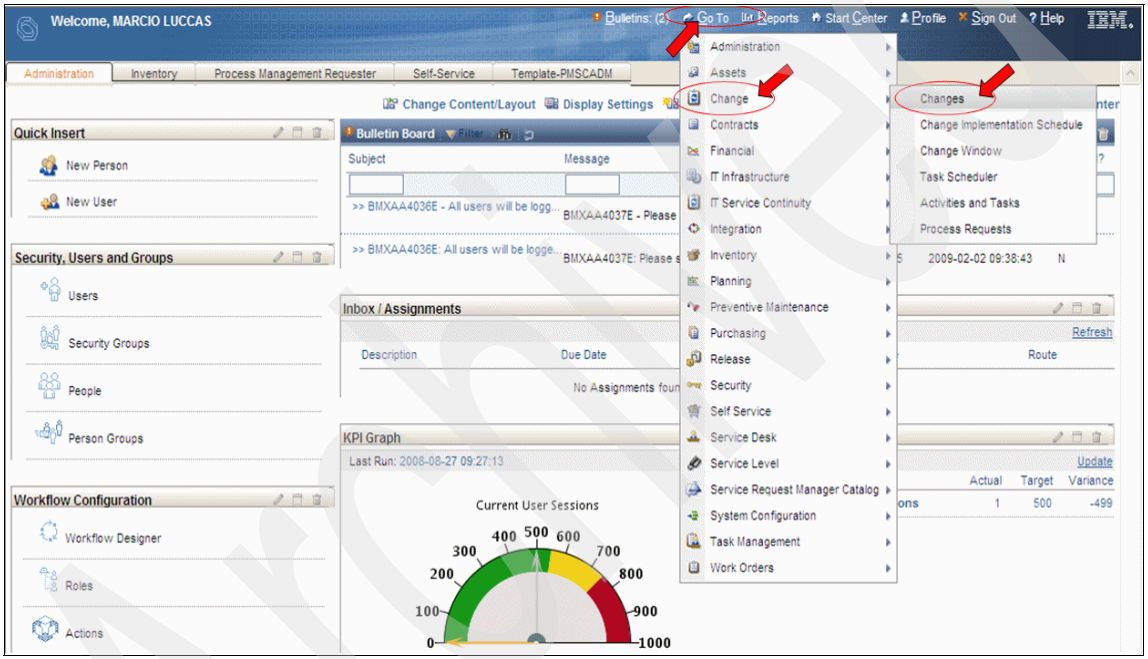


Figure 4-22 Asset and CI in Change Management (1 of 5)

2. Click the button shown in Figure 4-23 to create a new change.

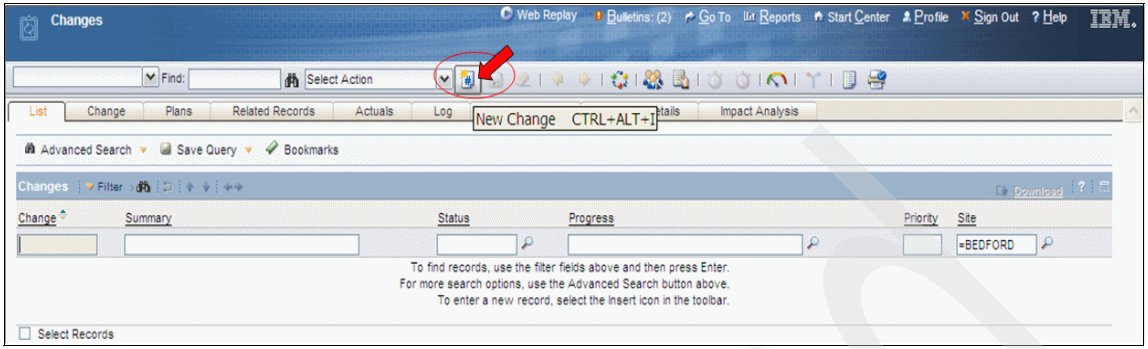


Figure 4-23 Asset and CI in Change Management (2 of 5)

3. Select a configuration item, as shown in Figure 4-24 and Figure 4-23.

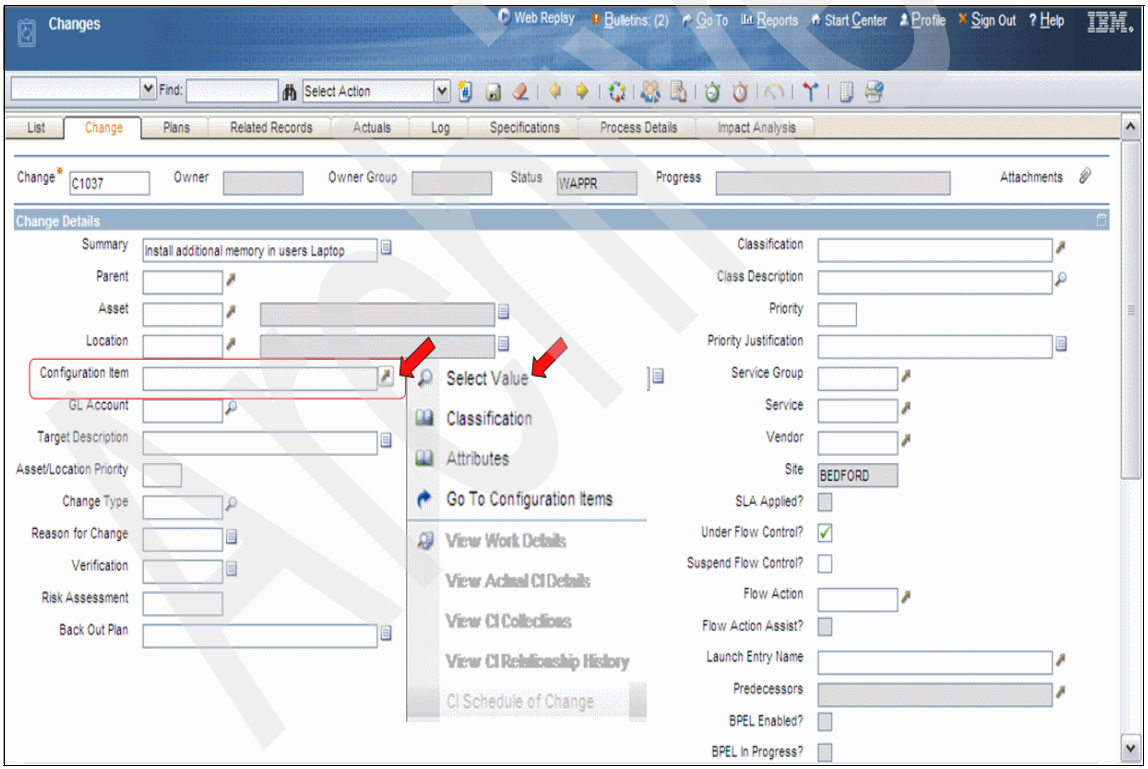


Figure 4-24 Asset and CI in Change Management (3 of 5)

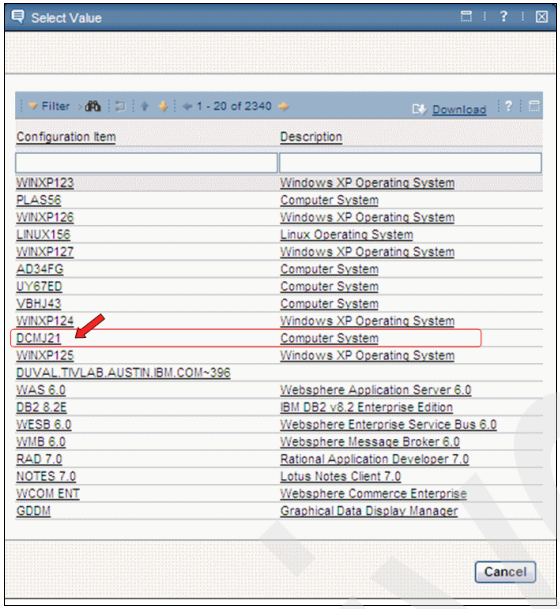


Figure 4-25 Asset and CI in Change Management (4 of 5)

4. Figure 4-26 shows that the asset and the corresponding CI are automatically linked in the Change Management module. You can now save and submit the request, as shown in Figure 4-26.

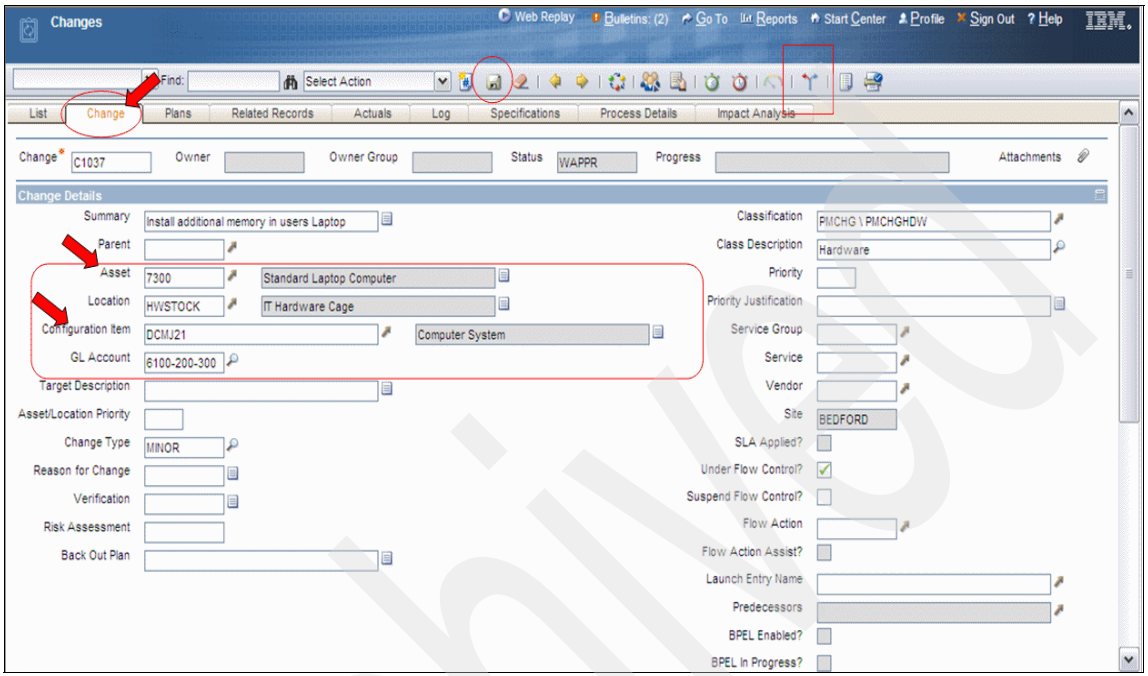


Figure 4-26 Asset and CI in Change Management (5 of 5)

4.6.2 Assets and CIs in Release Management

Release Management allows a release process to be targeted at a CI or an asset. When a CI is linked to an asset, the release process automatically populates the Asset field if the target is a CI and vice versa.

In the following figures we show you how this functionality works in Release Management.

1. Figure 4-27 shows how to access the Release Management tool.

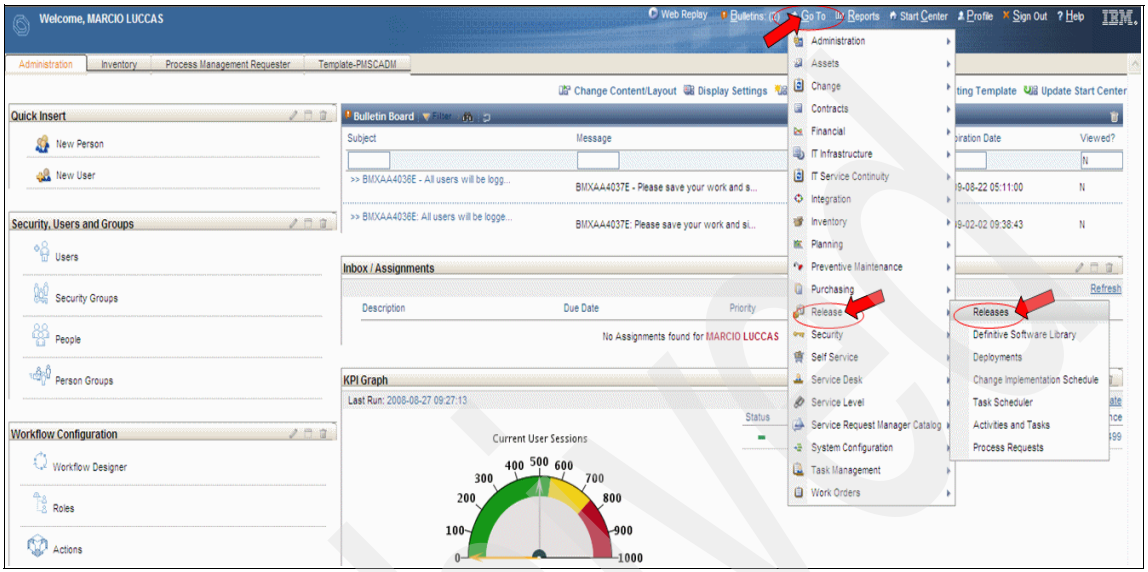


Figure 4-27 Asset and CI in Release Management (1 of 5)

2. Create a new Release request as shown in Figure 4-28.

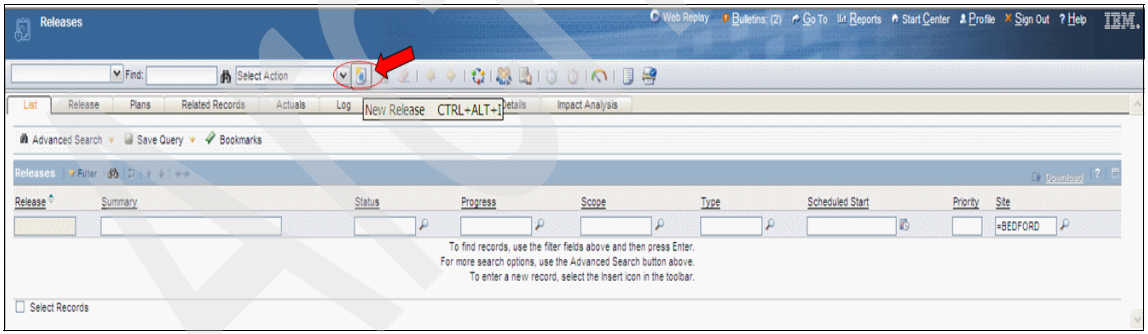


Figure 4-28 Asset and CI in Release Management (2 of 5)

3. Locate CI. See Figure 4-29 and Figure 4-29.

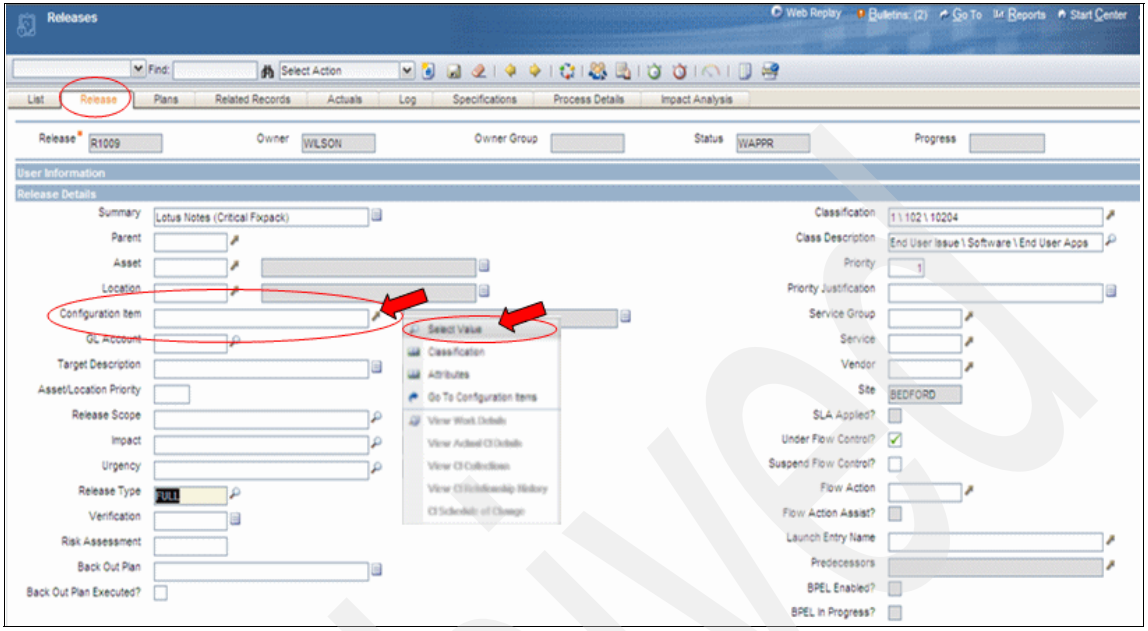


Figure 4-29 Asset and CI in Release Management (3 of 5)

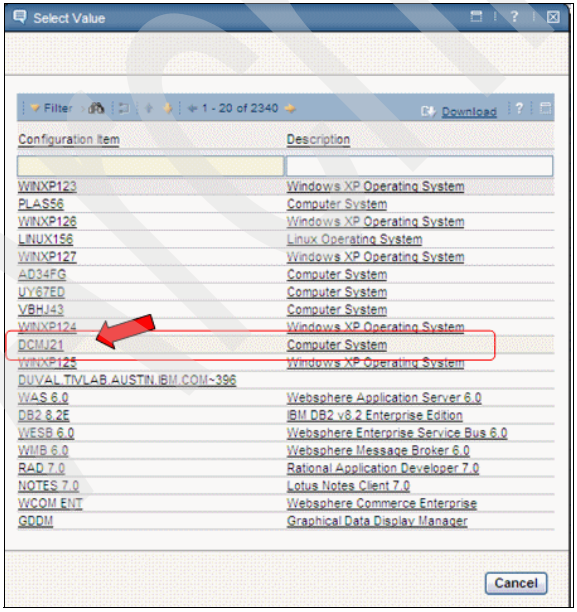


Figure 4-30 Asset and CI in Release Management (4 of 5)

4. Figure 4-31 shows that the asset and the corresponding CI are linked in the Release Management tool.

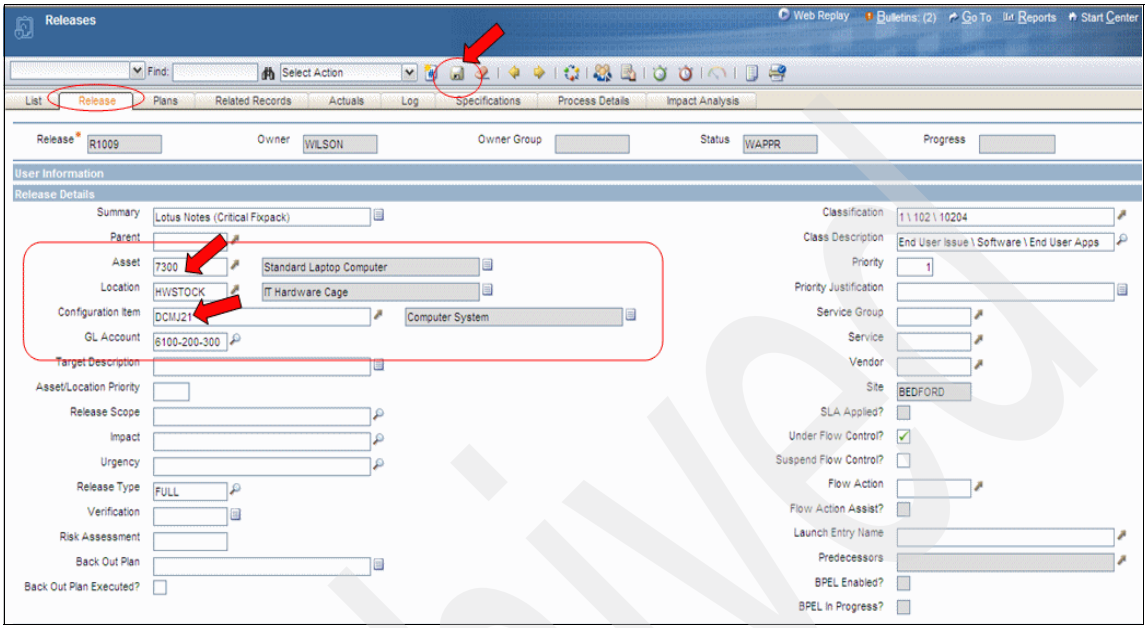


Figure 4-31 Asset and CI in Release Management (5 of 5)

4.6.3 Assets and CIs in Service Desk

Assets and CIs can be managed through Problem and Incident Management, as well. Note that to be able to import assets or CIs from TADDM (to be used in Service Desk) you must also install CCMDB (for CIs) and TAMIT (for assets).

1. Figure 4-32 shows you how to access the Service Desk Tool (Incident Management module).

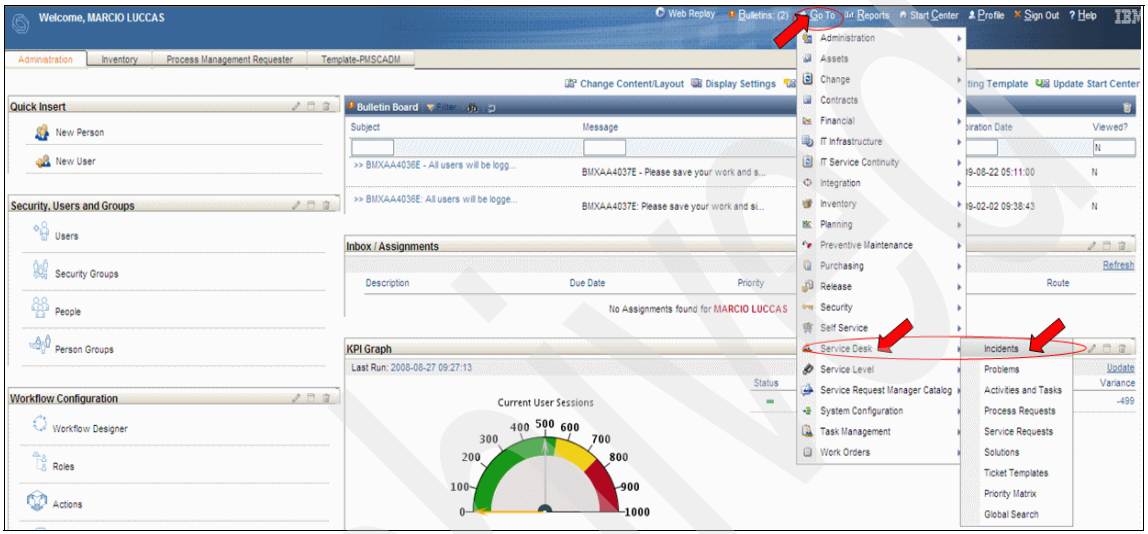


Figure 4-32 Asset and CI in Service Desk Module (1 of 5)

2. Create a new incident, as shown in Figure 4-33.

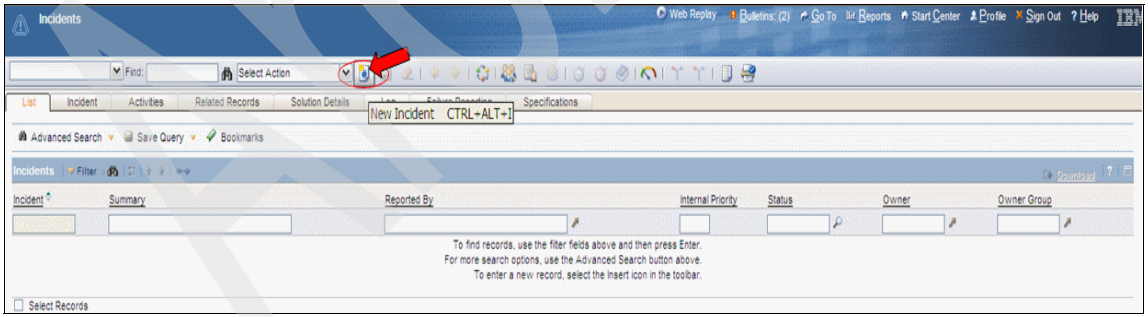


Figure 4-33 Asset and CI in Service Desk Module (2 of 5)

3. Choose an asset, as shown in Figure 4-34. Note that you can search for a CI or an asset. The result will be the same once they are linked.

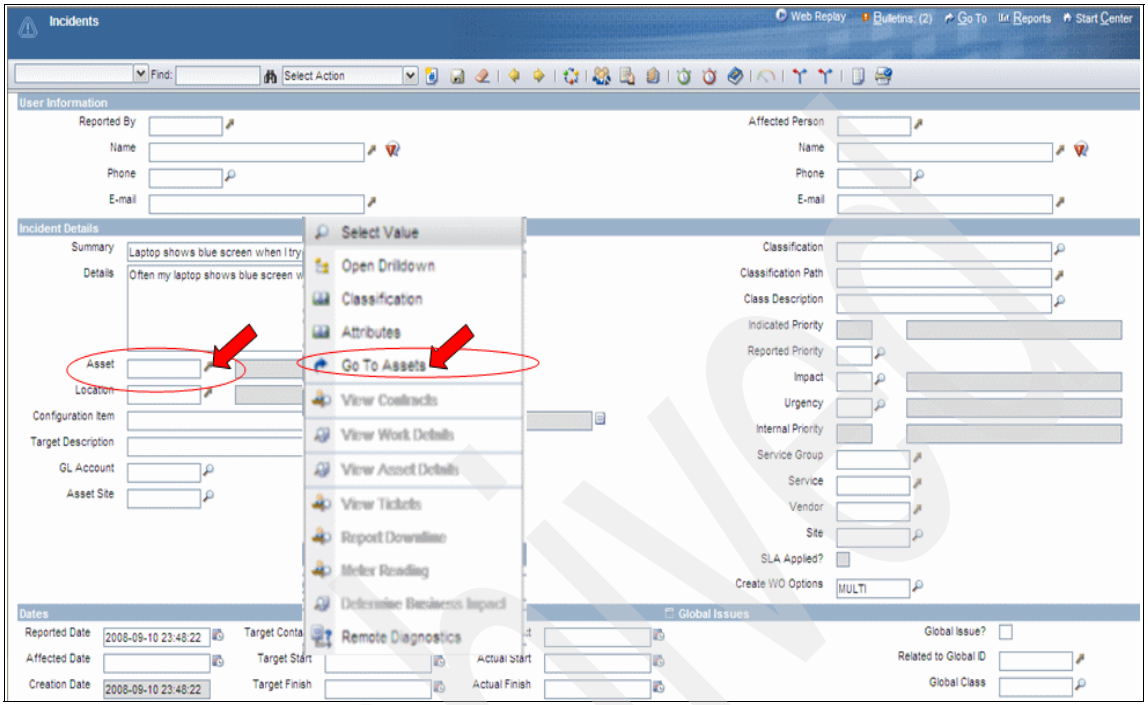


Figure 4-34 Asset and CI in Service Desk Module (3 of 5)

4. Figure 4-35 shows you that we entered the value regarding the asset that we are looking for. In this example, the asset chosen is 7300.

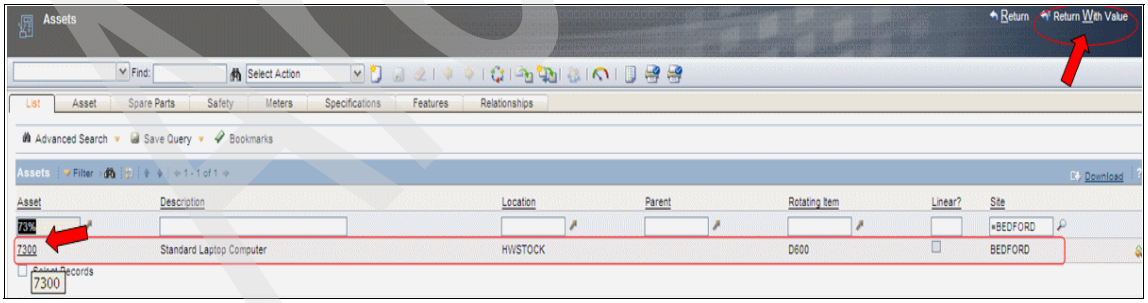


Figure 4-35 Asset and CI in Service Desk Module (4 of 5)

5. After the asset is found, we select the **ReturnWithValue** option.

Figure 4-36 shows that the asset and CI are linked in the Incident Management tool.

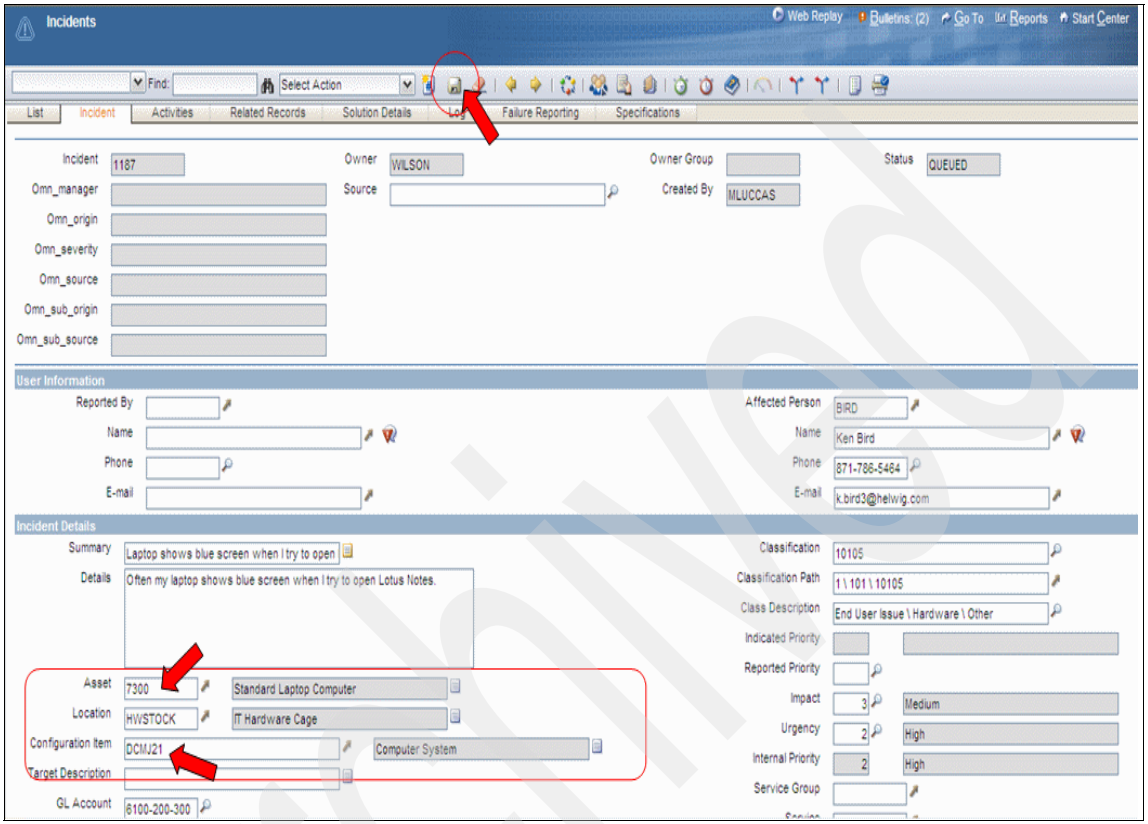


Figure 4-36 Asset and CI in Service Desk Module (5 of 5)

6. Save the incident record.

5

Scenario 1: Incident, Problem, and Change Management

This chapter describes a typical scenario for Incident, Problem and Change Management integration and shows the powerful integration capabilities of IBM® Service Management Products.

This chapter contains the following sections:

5.1 Lab environment

Figure 5-1 shows the lab environment used for all the scenarios in this chapter.

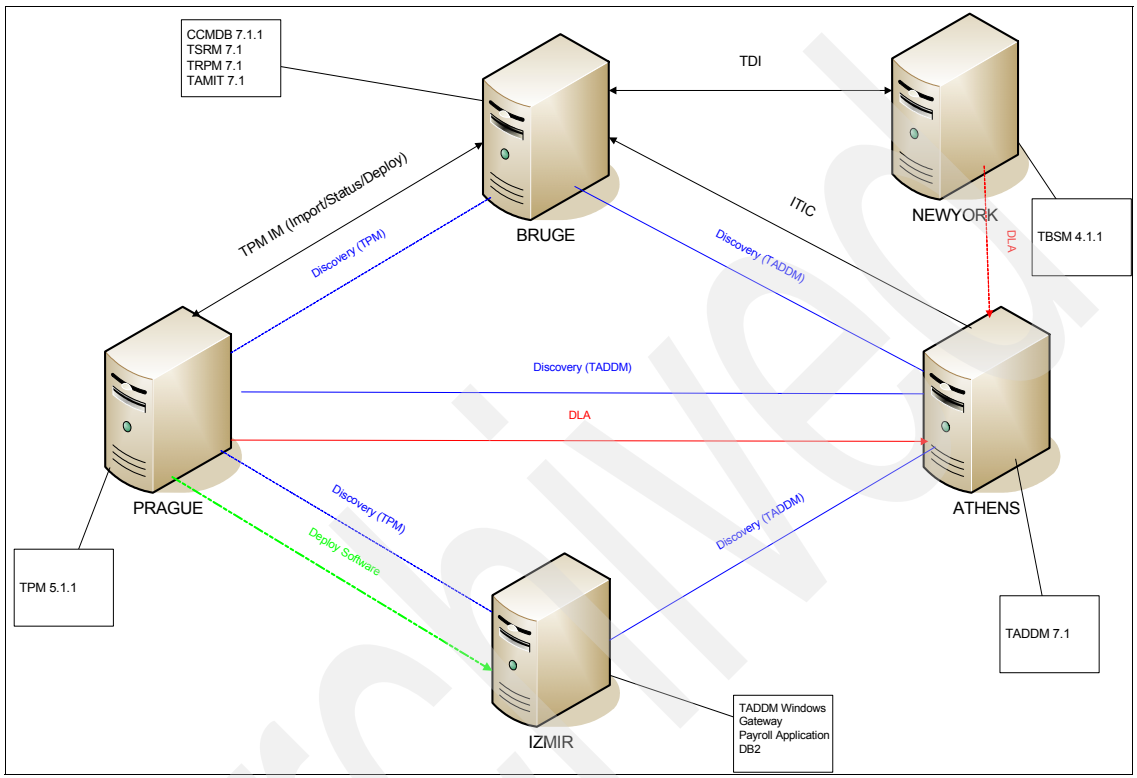


Figure 5-1 Lab environment

5.2 Installed components

The installed components on each system are described below:

- BRUGE
  - SRM Incident Management 7.1.0.00 Build 20080505D DB Build V710-59
  - Common PMP 7.1.1.2 Build 20080618D DB Build V710-40
  - SRM Computer Telephony Interface 7.1.0.00 Build ${env.BUILD_NAME} DB Build V710-03
  - SRM Service Desk Demo Data 7.1.0.00 Build 20080505D DB Build V710-13

- SRM Service Catalog 7.1.0.1 Build 20080624D DB Build V7101-08
- Release PMP 7.1.1.00 Build 20080429D3 DB Build V710-35
- SRM Search 7.1.0.00 Build 20080505D DB Build V710-10
- SRM Survey Management 7.1.0.00 Build 20080505D DB Build V710-07
- SRM Service Desk 7.1.0.1 Build 20080624D4 DB Build V7101-06
- SRM Problem Management 7.1.0.00 Build 20080505D DB Build V710-59
- Config PMP 7.1.1.2 Build 20080619D DB Build V710-29
- Release PMP 7.1.1.00 Build BUILD DB Build V710-35
- IM TPM 7.1.1.00 Build 20080503D DB Build V710-08
- Asset Management 7.1.0.0 Build 52b DB Build V7100-001
- SRM Service Request Management 7.1.0.1 Build 20080624D4 DB Build V7101-02
- Base Services 7.1.1.2 Build 10 DB Build V7112-10
- SRM Solution 7.1.0.00 Build 20080505D DB Build V710-20
- SRM Service Desk Content - Classification 7.1.0.00 Build 20080505D DB Build V710-03
- Business Continuity PMP 7.1.0.00 Build 20080508D DB Build V710-06
- Change PMP 7.1.1.2 Build 20080619D DB Build V710-28
- SRM Instant Messenger 7.1.0.1 Build 20080624D4 DB Build V7101-02
- Service Desk Integration MEA 7.1.0.00 Build BUILD DB Build V710-14
- SRM Service Desk Content - Best Practices 7.1.0.00 Build 20080505D DB Build V710-49
- Toolbox Web Replay 7.1.1.00 Build local DB Build V710-04

- ATHENS

  TADDM 7.1

- PRAGUE

  - TPM 5.1.1
  - TADDM Windows® Gateway

- IZMIR

  Payroll Application
  DB2® 9.1
  Payroll reporting App
  Used as target for Change Management

- NEWYORK

  TBSM 4.1.1

5.3 Emergency Request for Change (RFC) scenario

This scenario describes basic process of a full Incident Management cycle starting from incident creation until the deployment phase. It is designed to give the user a picture of the entire incident-problem-change process with deployment functionality. Later in the next scenario release functionality is covered, based on the problem created here. All workflows that will be shown for incident-problem management are based on ITUPv3, which is ITIL3 and IBM best practices. The process flow for Change Management is based on job plans and tasks and will be described later in this document.

5.3.1 Process flow

Figure 5-2 shows the process flow for this scenario.

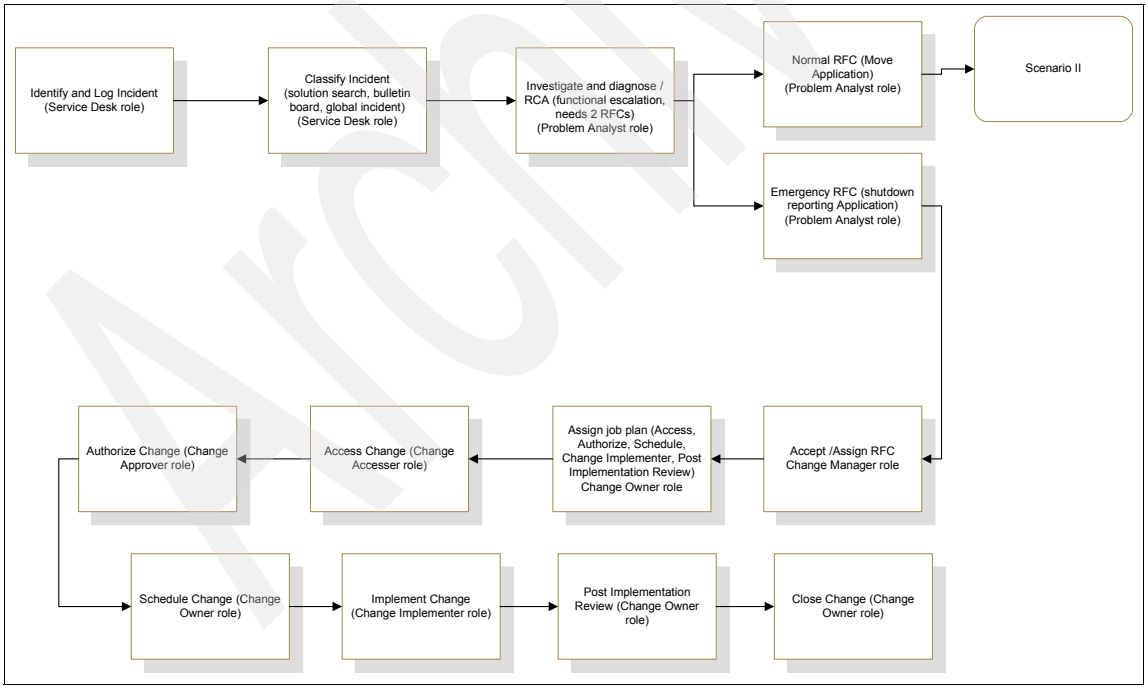


Figure 5-2 Process flow

5.3.2 Implementation steps

The implementation steps of this scenario are:

1. Tommy Reading is working with the payroll application. Today he cannot perform his normal day-to-day activities because the application is running very slowly and very often he gets a time-out exception that the operation took too much time to execute. Because it is critical for him to work normally, he decides to open a service request in TSRM. He logs in and creates a service request with basic information. See Figure 5-3 and Figure 5-4.

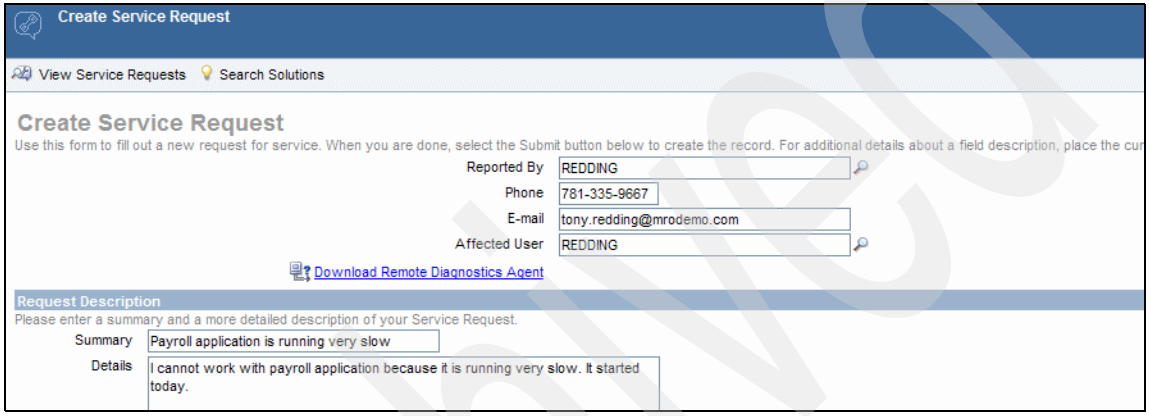


Figure 5-3 Service request

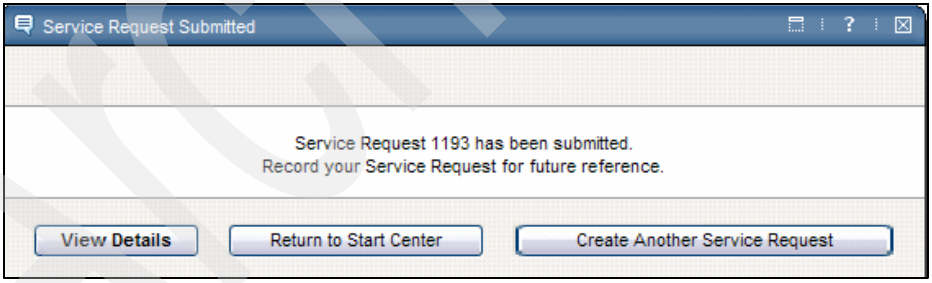


Figure 5-4 Confirmation that the service request has been created

Note: In standard TSRM self-service users have access to create service request applications in the self-service module. It is a best practice to revoke it and just leave access to search solutions. This will force users to search for existing solutions prior to creating service requests because there is an option to create a service request from the search solution application.

2. Hallie Moroles (an incident analyst) sees that there is a new service request created and she starts working on it. By clicking it she is redirected to the service request application where she can perform her analysis. See Figure 5-5 and Figure 5-6.

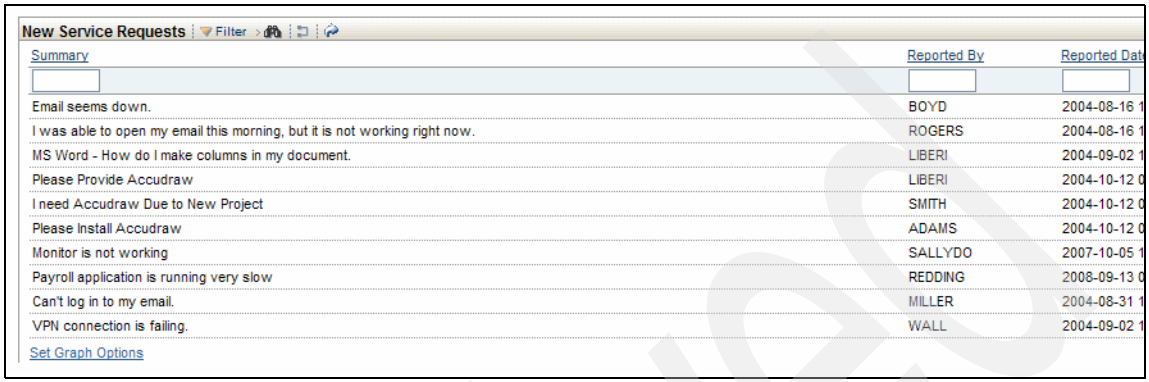


Figure 5-5 Portlet with service requests assigned to Hallie Moroles

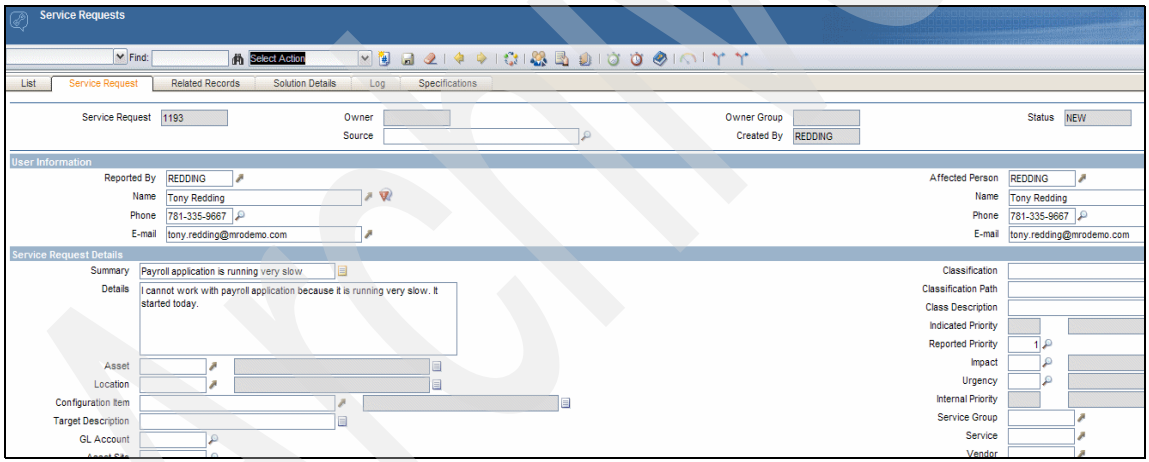


Figure 5-6 Service request record

3. Hallie applies site to a ticket (SITEID is a basic criteria for selecting SLAs and ticket templates) and then applies ticket template (PAYROLL) and SLA. After applying the ticket template, fields like Classification, Internal Priority, Service Group, and Service are copied from the template. At the end she specifies the CI (payroll app) that is involved in this service request. See Figure 5-7 through Figure 5-9 on page 124.

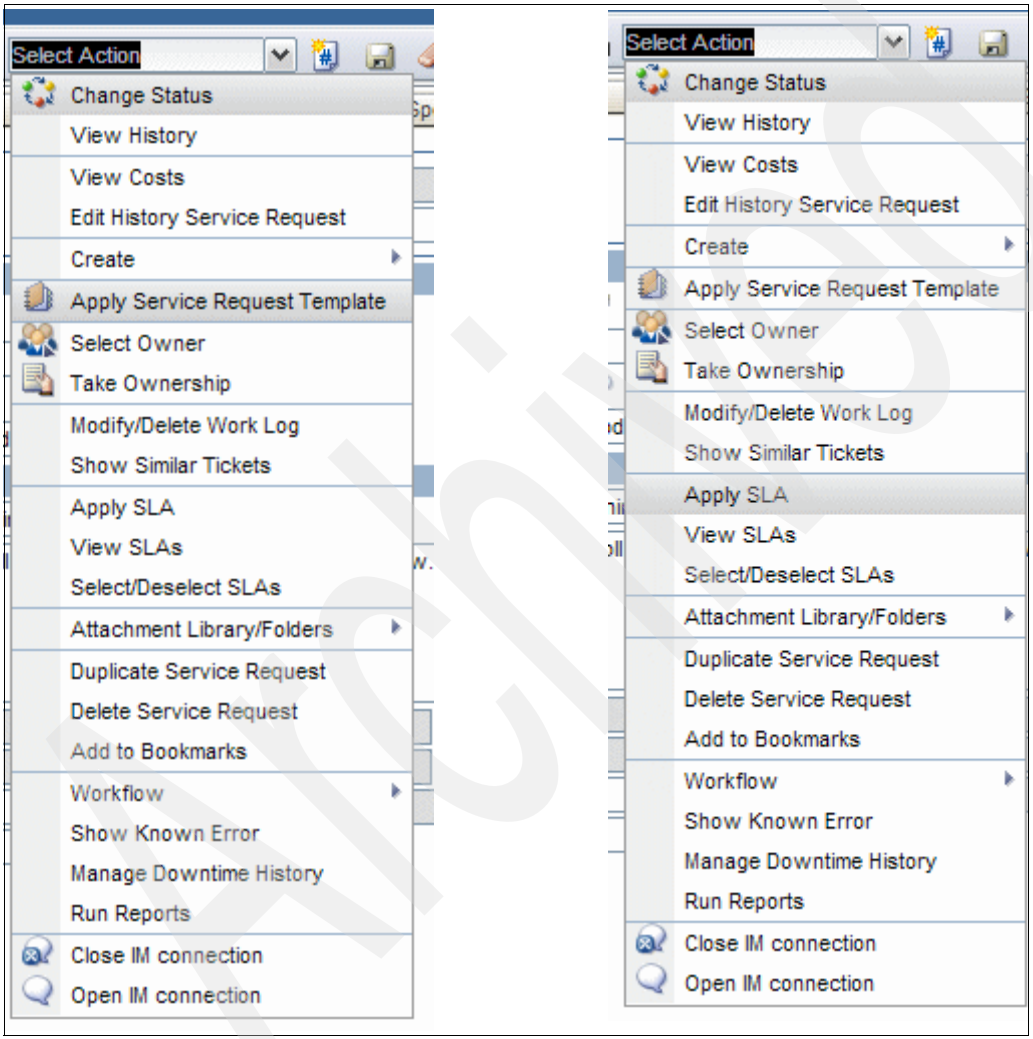


Figure 5-7 Applying service request template and SLA

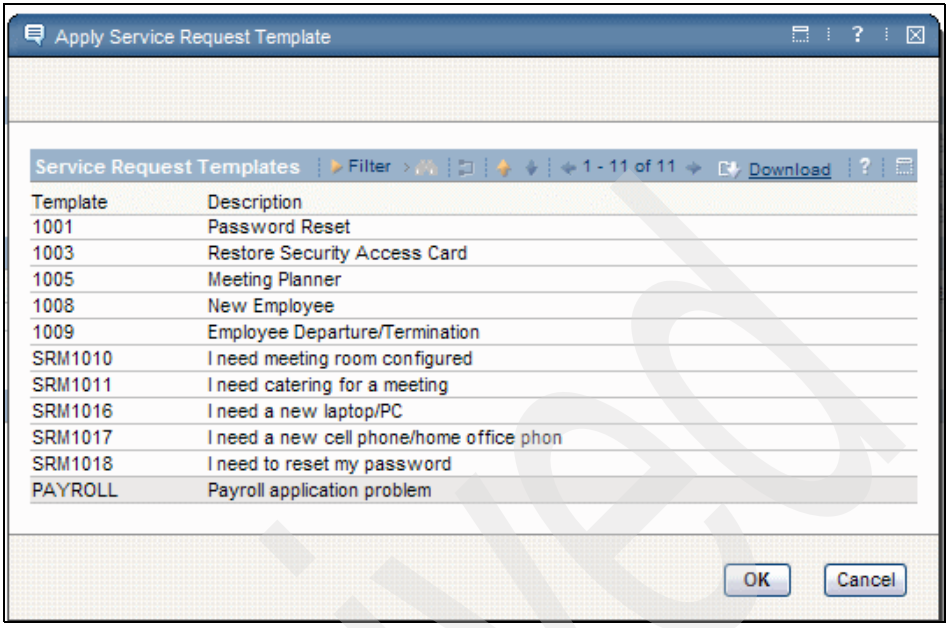


Figure 5-8 Service request templates: Ticket templates

Figure 5-9 Notification that SLA has been applied

Note: The PAYROL ticket template and EMERGENCY SLA have been created in TSRM to meet this scenario's requirements. Normally, you would create ticket templates and SLAs before you start using the product in production, or you would import the data from an external source. Note that using ticket templates and SLAs is not required, but recommended.

4. After applying the SLA and ticket template, Hallie starts working on the service request using workflows provided( ). We use a workflow called SR_ITUPv3. As a first step she needs to put all required information, such as source (in our case self service), description, siteid, and priority. See Figure 5-10 and Figure 5-11.

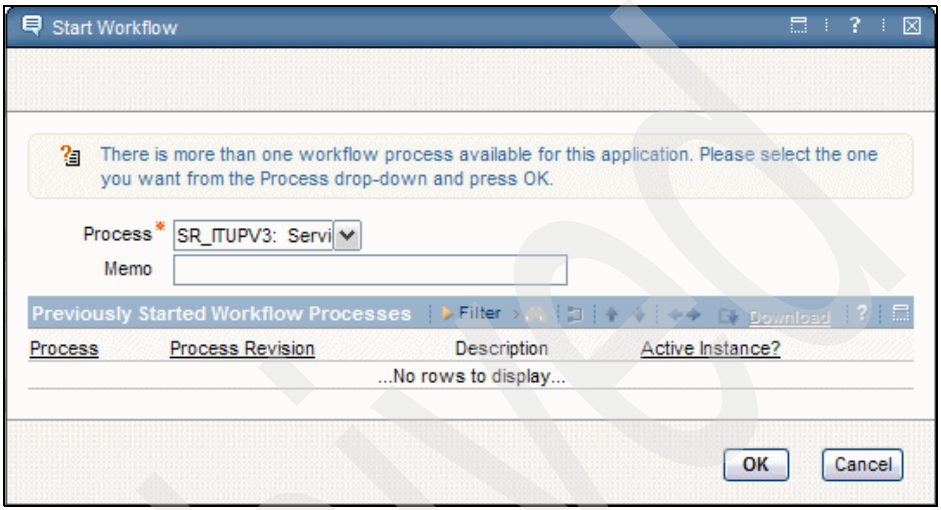


Figure 5-10 Selecting workflow process

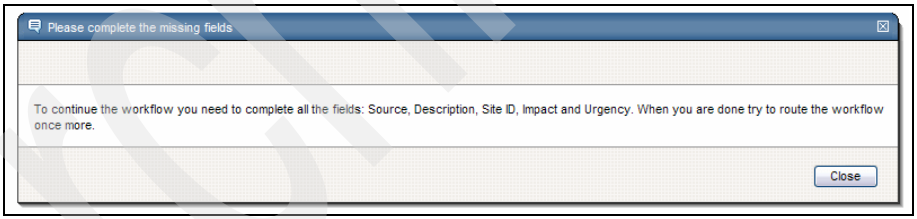


Figure 5-11 Information about necessary actions for next step of workflow

Note: To define priority you must specify impact and urgency. The priority matrix application contains business rules about how to calculate priority based on those fields.

Note: If your company decides to use just one workflow, deactivate the rest of the workflows for this application in the Workflow Designer application. You will not see the dialog with the selection of all available workflows. Only the one that you want to use starts.

5. Hallie enters all required information and routes the workflow again( ).
6. Hallie assigns herself to be the owner of this service request and routes the workflow again.

   The next step in the flow is to find available solutions for it. A dialog window with the search solution application pops up. See Figure 5-12.

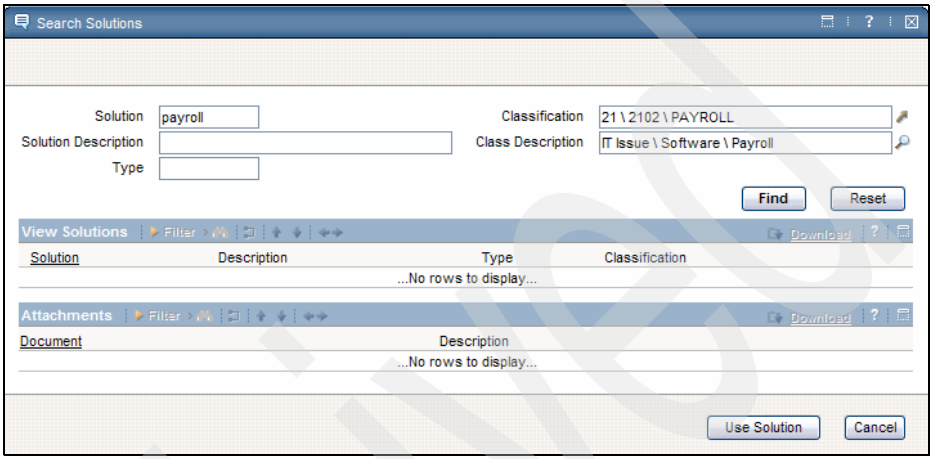


Figure 5-12 Searching knowledge base for suitable solutions

7. Unfortunately, no solution can be found. As a next step Hallie is required to communicate to the requestor about the status of the service request resolution. She communicates to Tony that the solution cannot be found and that it will become an incident. She uses the Communication Template dialog window that is show as soon as she routes the workflow again. See Figure 5-13 and Figure 5-14 on page 128.

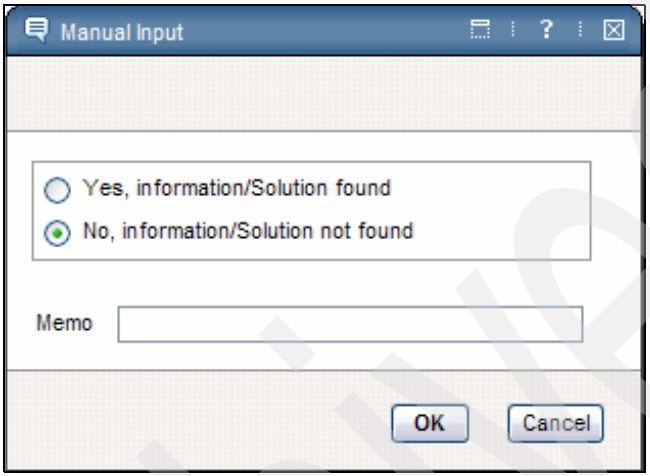


Figure 5-13 No solution found

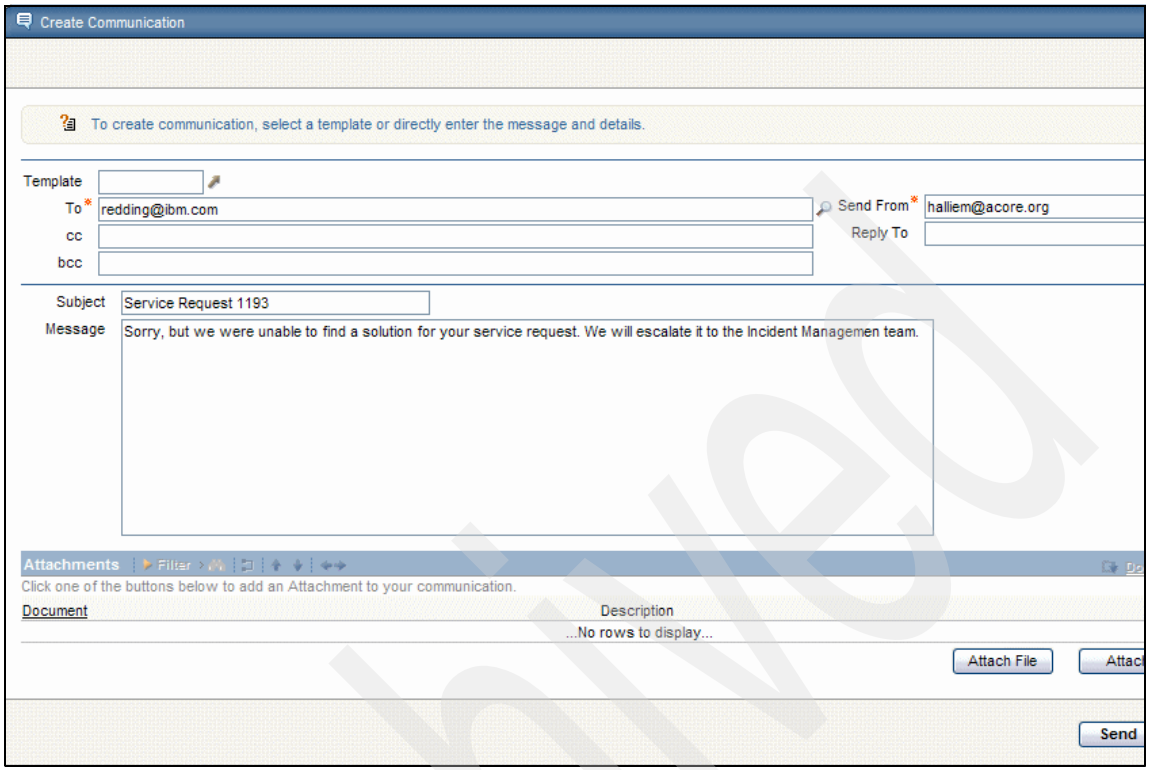


Figure 5-14 Sending e-mail to Tony

8. Hallie creates an incident and routes the workflow again to stop the service request process flow. Now there is an incident related to the service request reported by Tony, and incident analysts should take care about next step in the process flow. See Figure 5-15 and Figure 5-16.

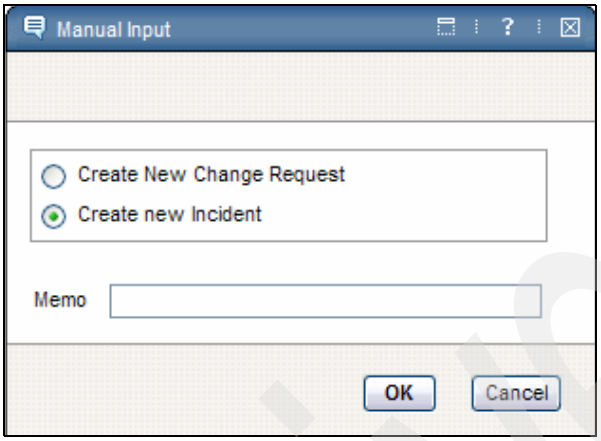


Figure 5-15 Creating incident from service request

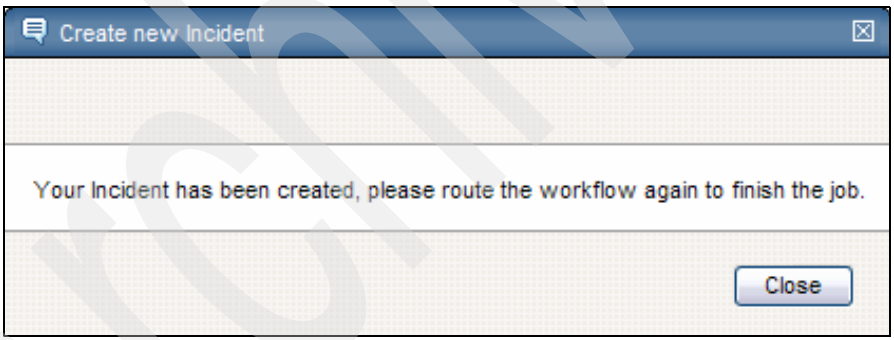


Figure 5-16 Notification that incident has been created

9. Sophie C. Allen (an incident analyst) logs in to TSRM and sees that there is an incident assigned to her because there is a record in the My Open Incidents portlet in her Start Center. She opens this incident and starts the workflow (ITUPv3). The first steps are to perform incident identification:
   a. Review it.
   b. Acquire additional data if required.
   c. Contact the requestor if required.
   d. Identify incident CIs.

Because Sophie needs additional information she contacts Tony. See Figure 5-17 through Figure 5-20 on page 131.

Figure 5-17 Information about next steps in Incident Management

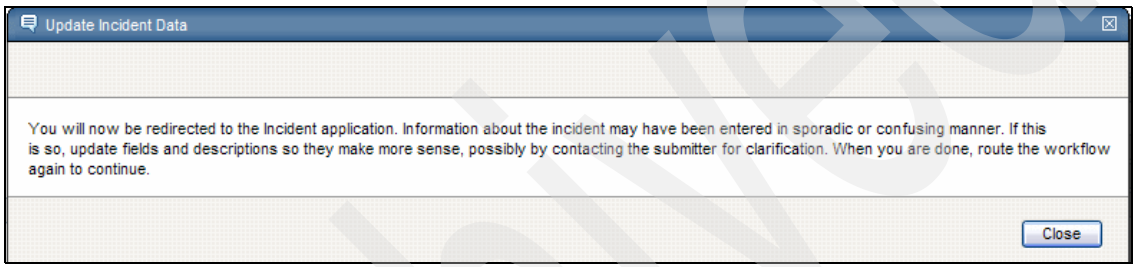


Figure 5-18 Information about next steps in Incident Management

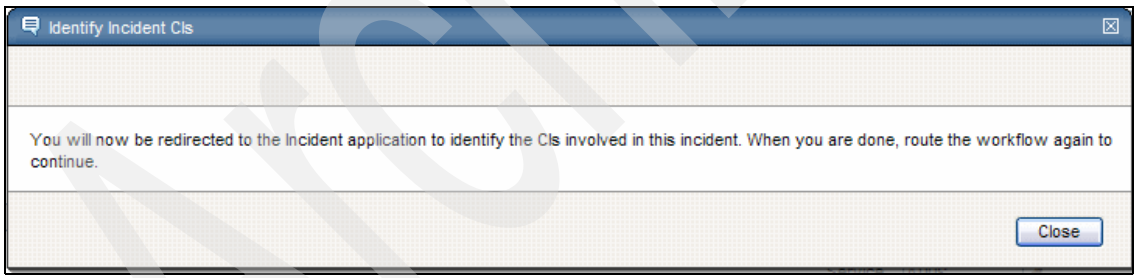


Figure 5-19 Information about next steps in Incident Management

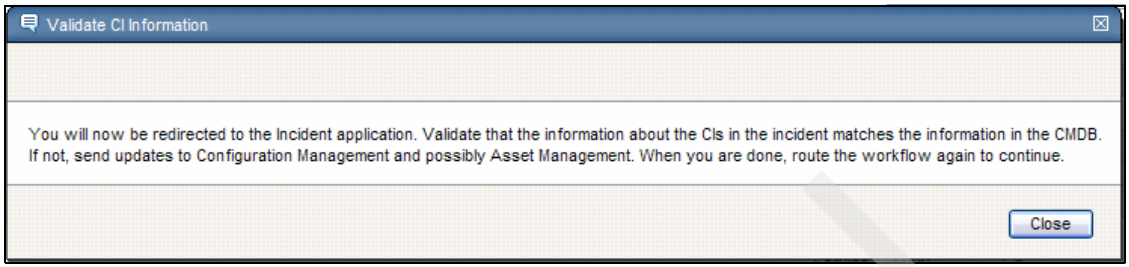


Figure 5-20 Information about next steps in Incident Management

Note: At this time it is also possible to apply ticket templates with predefined values for fields like classification. In our case Sophie just categorize the incident without applying the ticket template. Keep in mind that ticket templates are specific for each application (SR, INCIDENT, PROBLEM).

10. At this point Sophie is presented with the dialog to specify an asset (Figure 5-21). She was working with CI all the time and she does not want to specify an asset at this point.

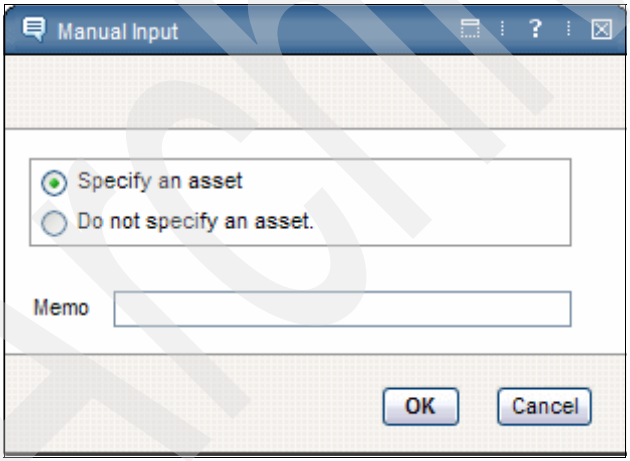


Figure 5-21 Decision point to specify an asset and work with Asset Management

Note: To learn more about how CIs and assets work together refer to Chapter 4, “How Asset Management and Configuration Management work together” on page 83.

11. Sophie routes the workflow again to search for similar tickets using the action Show Similar Tickets. As she finds that there are some similar tickets, she decides to relate them to the current incident and make it a global issue. See Figure 5-22 through Figure 5-26 on page 134.

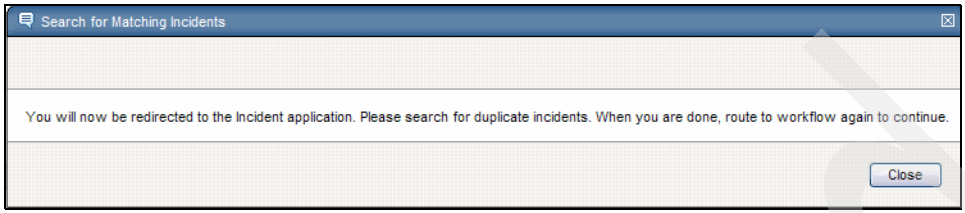


Figure 5-22 Dialog informing about next step: Searching for duplicate incidents

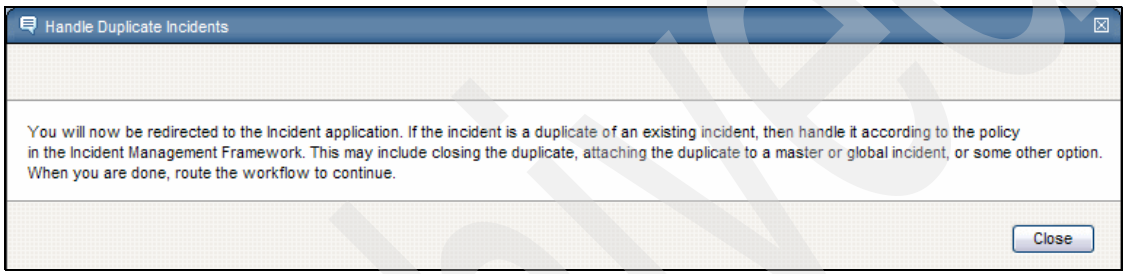


Figure 5-23 Help information about what to do with duplicate tickets

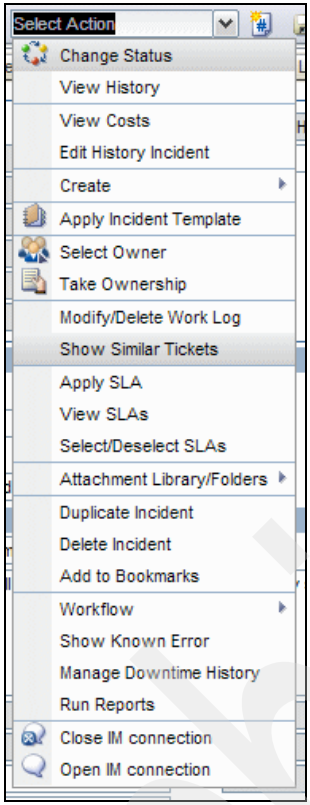


Figure 5-24 Using Show Similar Ticket action

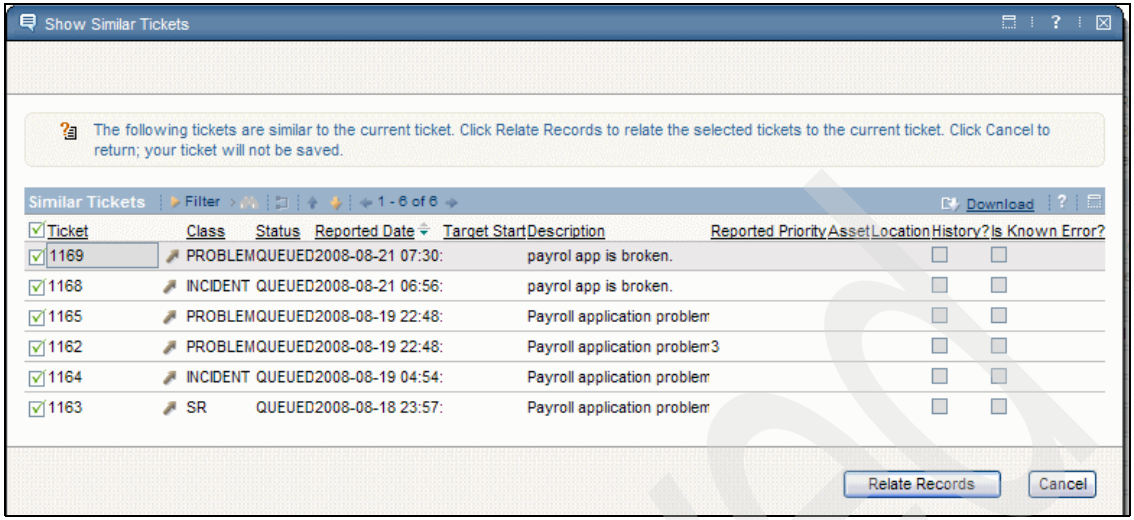


Figure 5-25 Similar tickets found

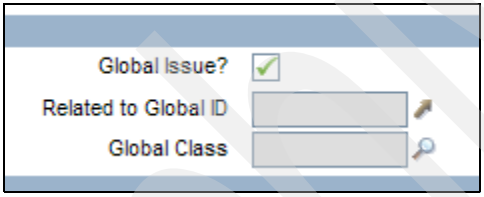


Figure 5-26 Setting incident to be global issue

12. Define both impact and urgency to calculate priority. As the incident is critical, impact and urgency must be both 1. See Figure 5-27 and Figure 5-28 on page 135.

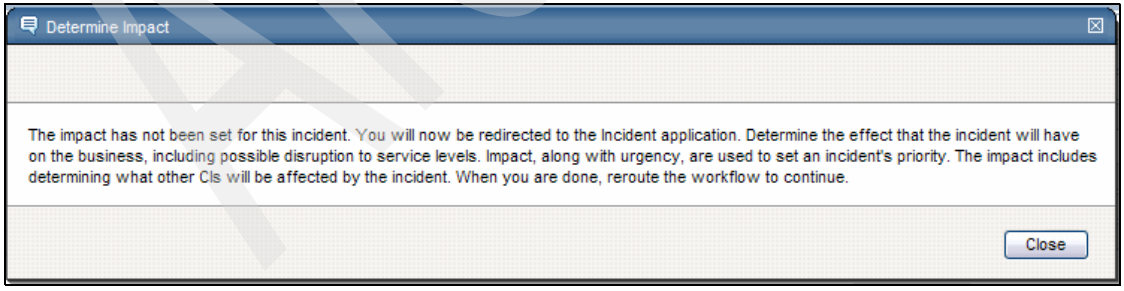


Figure 5-27 Information about next step: Setting impact and urgency

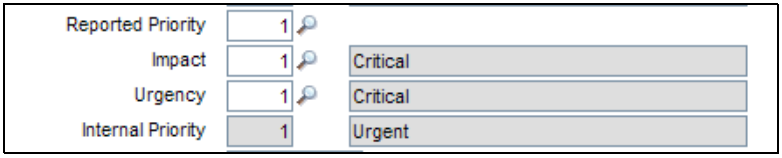


Figure 5-28 Impact and urgency, and Internal priority based on those values

13. Sophie tries to find a solution for this from matching tickets, but there is no solution available. She decides to assign someone else to handle this incident (Scot Motika, Tier 1 member) and routes the workflow again to finish working with this incident. See Figure 5-29 through Figure 5-31 on page 136.

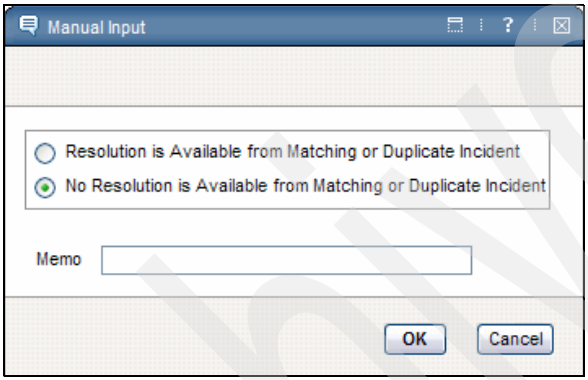


Figure 5-29 No resolution available

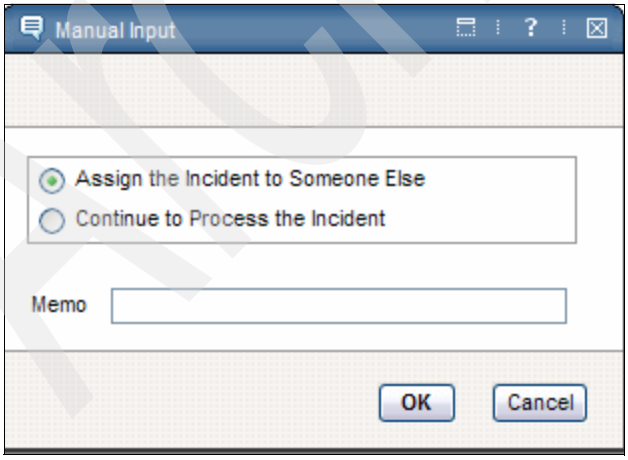


Figure 5-30 Assigning someone else to the incident

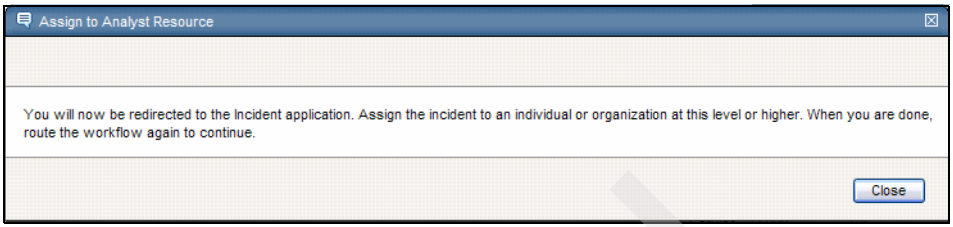


Figure 5-31 Hint to assign someone with higher level of competencies

14. Scot Motika logs in and opens the payroll incident that is accessible for him from his Open Incident's portlet in the Start Center. He performs standard checks (the same that Sophie did). After standard checks Scot is asked to perform additional check to understand the root cause of the incident. He decides to check all dependencies in the TADDM by using the launch-in-context option (Select Action menu). He finds out that there was another application installed on the same server that may be the problem. He creates a work log and routes the worklfow again. See Figure 5-32 and Figure 5-33 on page 137.

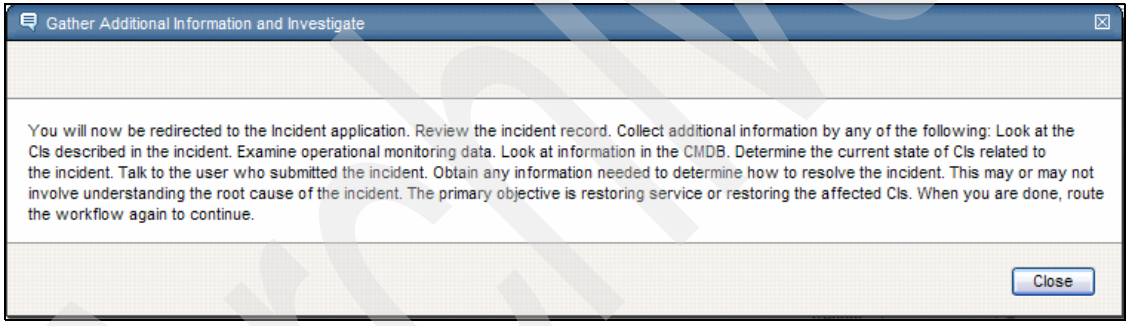


Figure 5-32 Look into CMDB to find more information about the CI

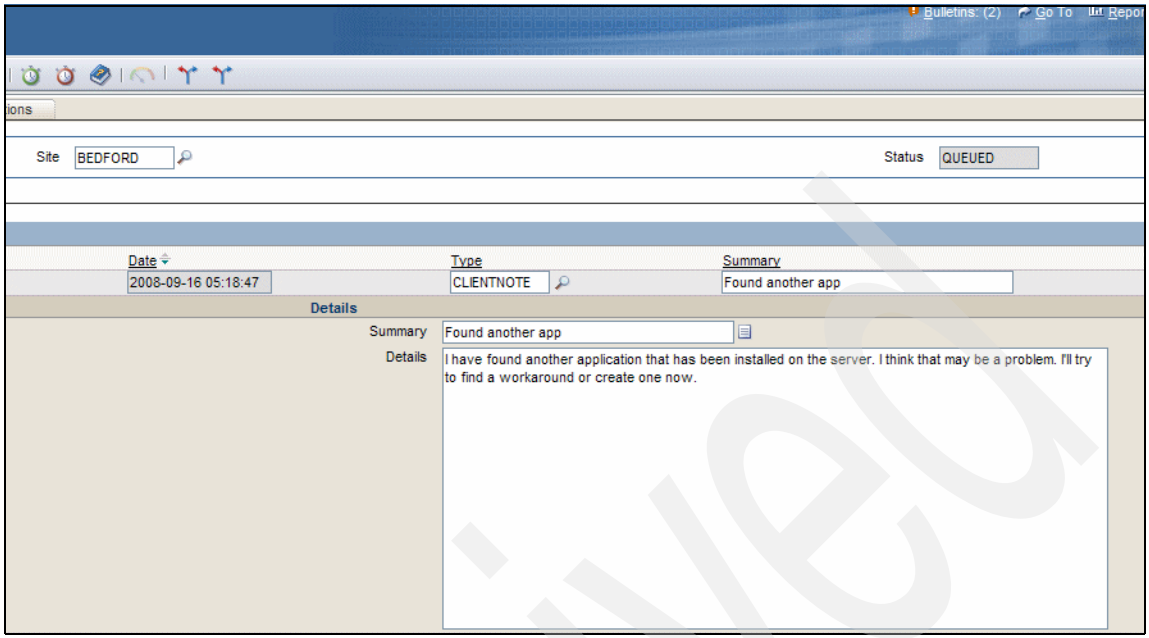


Figure 5-33 Work log created

15. Scot attempts to find a fix or workaround. Unfortunately, he is not able to find one, so he tries to create a workaround. Again, it is not possible to create it, so he decides to report a major incident that requires root cause investigation, and decides to create a problem. See Figure 5-34 through Figure 5-36 on page 138.

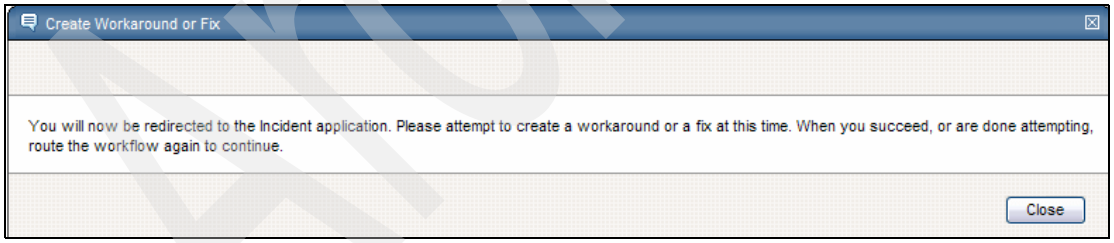


Figure 5-34 Next step is to create or find a workaround

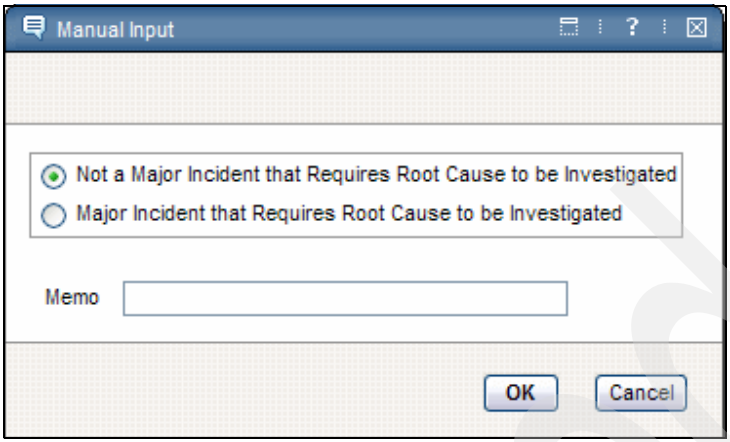


Figure 5-35 Select major incident option

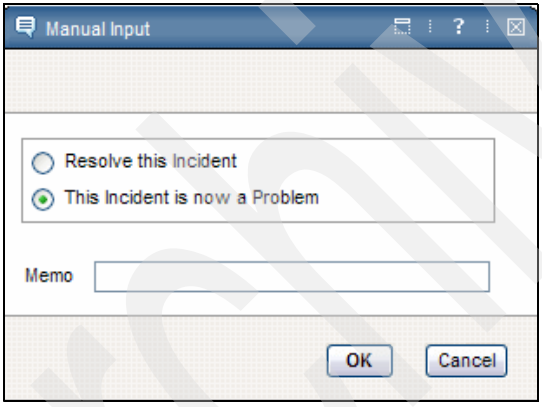


Figure 5-36 Incident becomes a problem

16. Philip Campbell (a problem manager) logs into TSRM and checks whether there are any new problems without an owner. Because there is one, he opens this problem and assigns someone to start working on it. He uses the select owner action and filters for people who are members of the problem analyst group. He finds Granger and assigns him to handle the problem, as shown in Figure 5-37

Note: Every time you assigned an owner ticket changes its status to QUEUED.

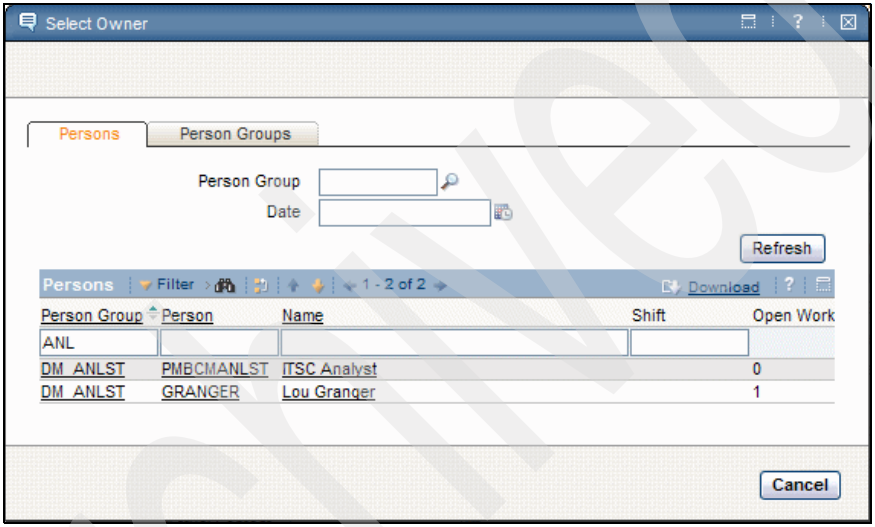


Figure 5-37 Assigning analyst to handle problem

17. Lou Granger (a problem analyst) sees that there is an open problem assigned to him. He opens the problem, and after reviewing all information, he starts working on it using workflow (ITUP_v3 Problem Management). He is asked to select the strategy with which to handle the problem. He tries to understand the major incident by reviewing the incident for which the problem was created. He is automatically redirected to the Related Record tab, from which he can open the original incident. See Figure 5-38 through Figure 5-40 on page 141.

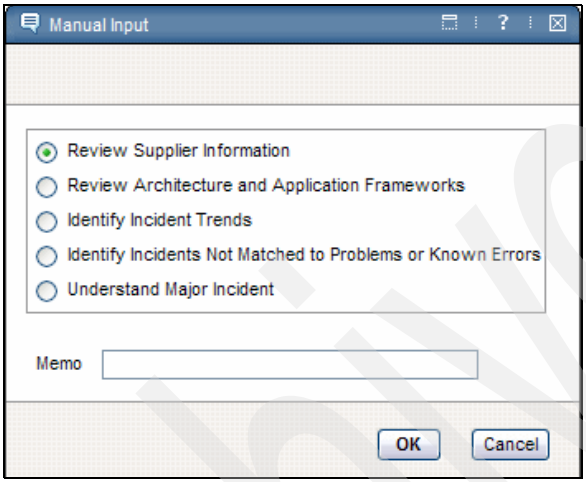


Figure 5-38 Lou selects Understand Major Incident strategy

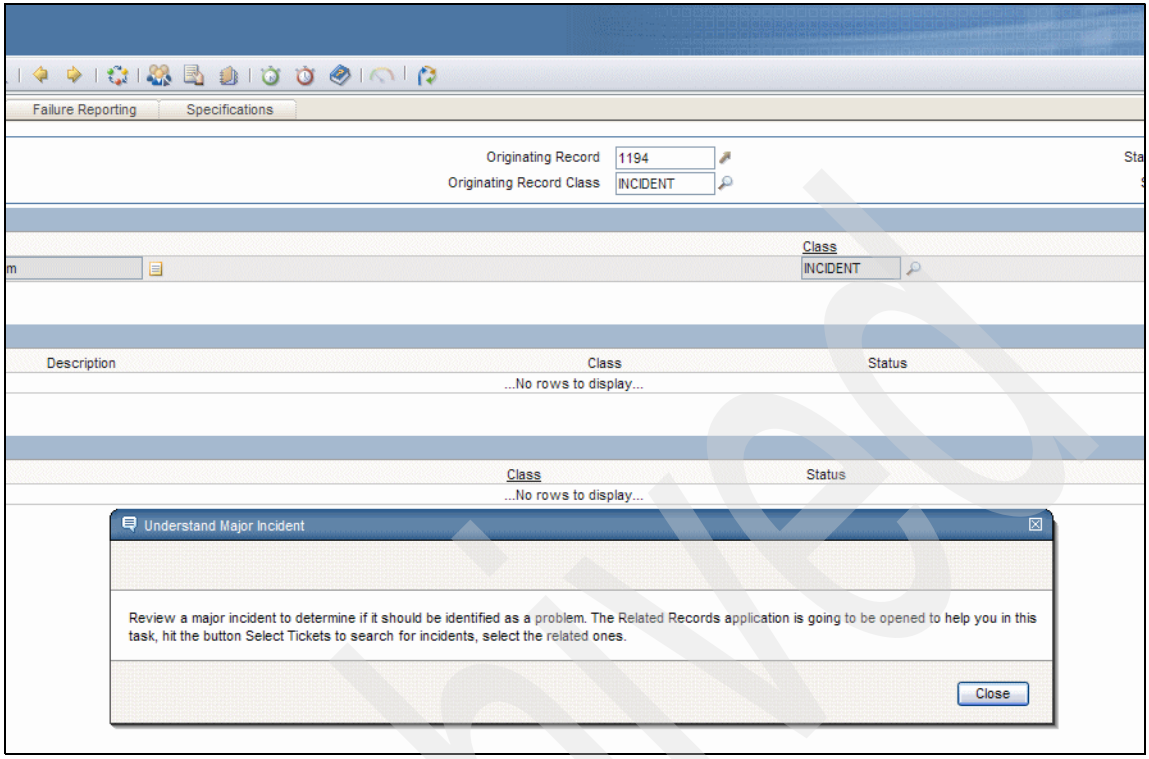


Figure 5-39 Lou is transferred to Related Record tab of problem to open originating incident to analyze related records

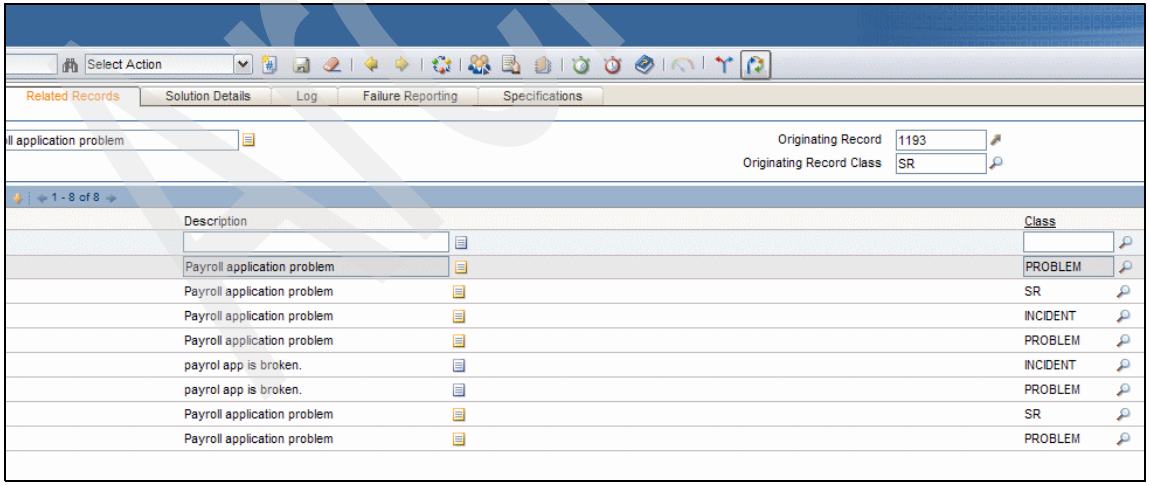
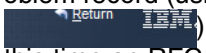


Figure 5-40 List of all records related to this incident

18. Lou opens the original incident and from there opens the CI that this incident is reported against. Then he opens TADDM using launch-in-context and checks for all dependencies. He finds another application installed on the server so he goes back to the problem record (using the return function located on top right of the panel 

) and routes the workflow to log improvement opportunity. At this time an RFC is automatically created. Lou is asked to go to the change record to fill in all required information. See Figure 5-41 through Figure 5-43 on page 143.

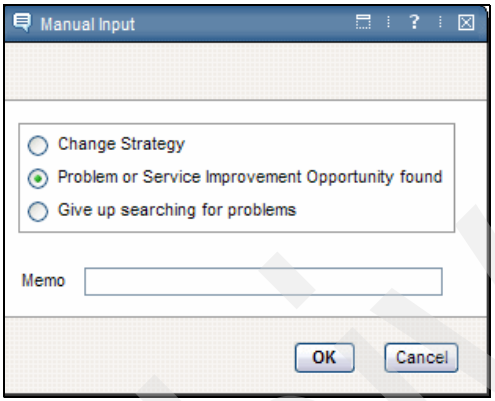


Figure 5-41 Found improvement opportunity

Figure 5-42 Notification that the change record has been created

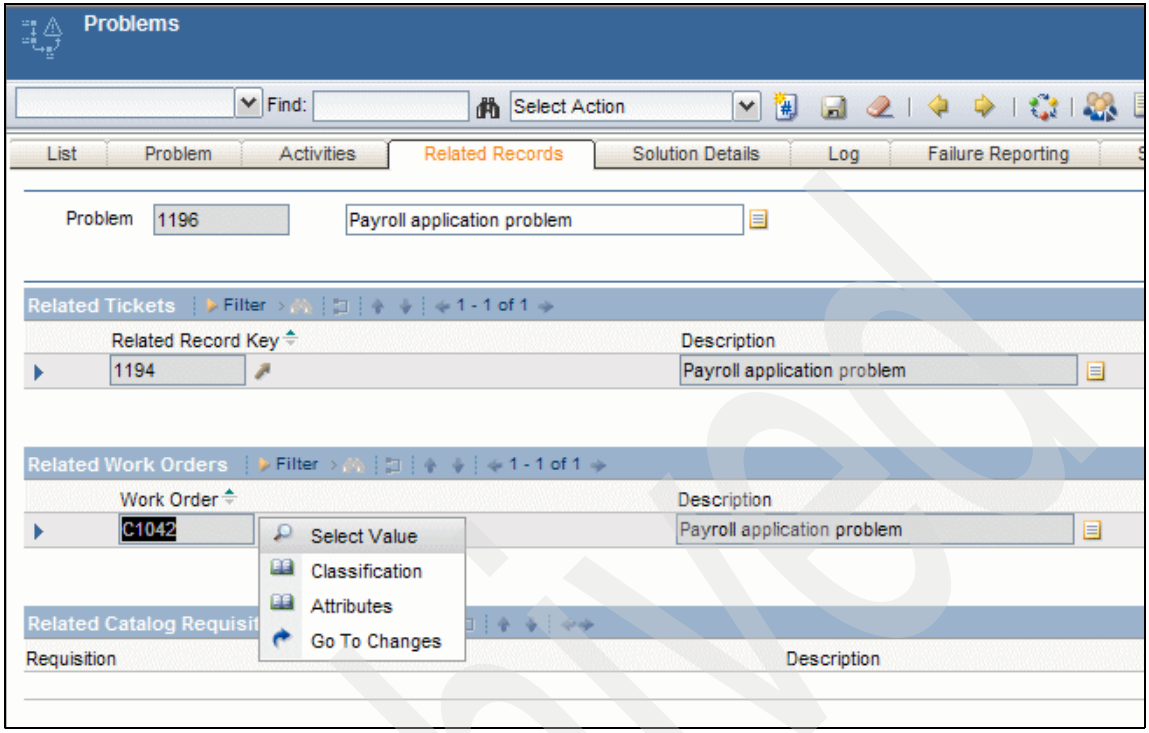


Figure 5-43 Change created: Using go to changes link Lou goes to the newly created change record

19. Lou opens the change record and works on it. After applying all necessary information (like description, CI, priority, and so on) he submits it to Change Management and returns to the problem application.

20. Lou tries to describe symptoms and define the impact on the IT infrastructure by assigning impact and urgency to the problem. He documents the cause of the problem by creating a work log, as shown in Figure 5-44.

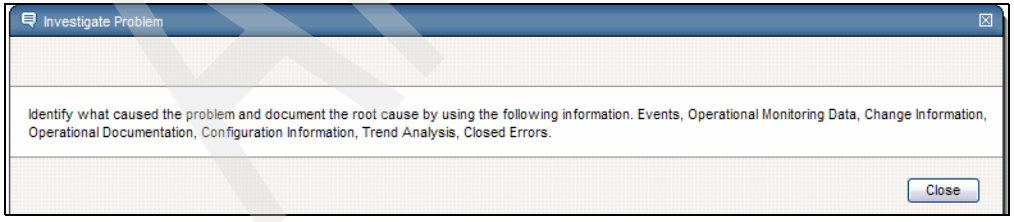


Figure 5-44 Documenting the root cause

21. Lou acknowledges that he found the root cause of the problem. He does not want to set it as a known error, so he continues working with the problem. He

decides that this problem must be resolved (that the cost and complexity of implementing the change is acceptable). See Figure 5-45 and Figure 5-46.

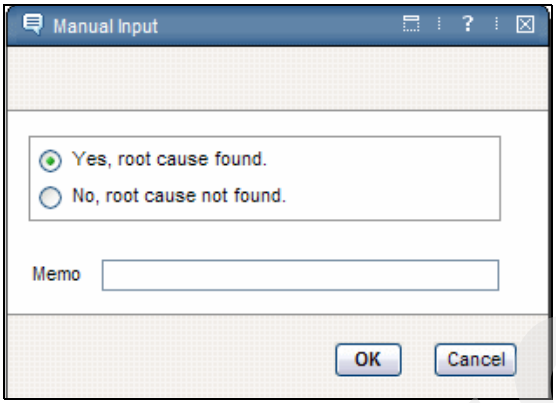


Figure 5-45 Found the root cause

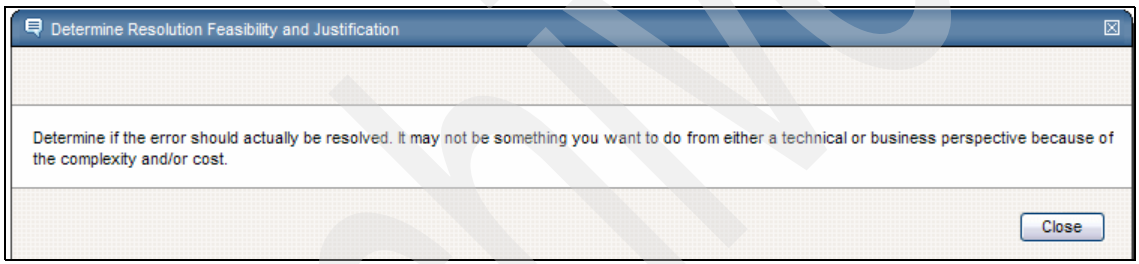


Figure 5-46 Deciding whether error can be easily resolved taking into consideration complexity and costs

22. Lour decides that there is no feasible solution and sets the status to pending to wait for a change to be implemented, as shown in Figure 5-47.

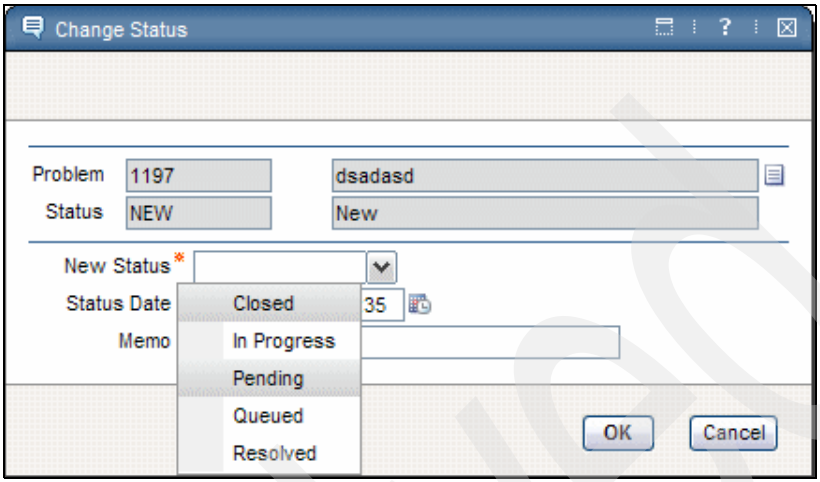


Figure 5-47 Set status to pending

23. Kazmier (a change manager) logs in and starts working on new change record that he has in his Start Center. He reviews the change and assign the owner (Homer) to continue working on the change.

24. Homer (a change owner) notices that there is a new change assigned to him. He goes to the change record and reviews it. After identifying that this is urgent, he sets the priority to 1 and applies an emergency job plan. Once the emergency job plan is applied, there are activities and tasks created for this change that are accessible from the plan tab. The first task is started as soon as he sets the change to INPRG (in progress). See Figure 5-48 and Figure 5-49 on page 146.

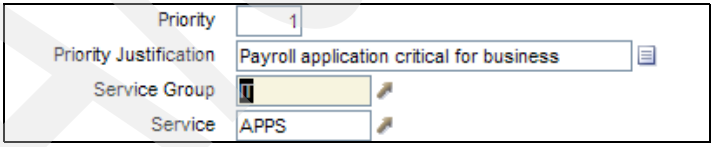


Figure 5-48 Setting priority to 1

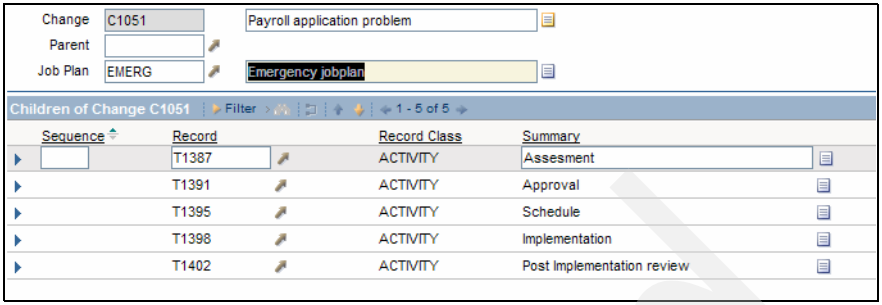


Figure 5-49 Applying emergency job plan: Below are listed activities applied to change after applying job plan

Note: Job plans may vary depending on the organization. For this scenario an emergency job plan has been created. It consists of five phases (assessment, approval, schedule, implementation, post-implementation review). Each phase was modified to meet emergency change request requirements. Because there are lots of steps to complete the change, only the most important are described.

Assessment phase

The assessment phase includes three tasks to be completed:

- Preliminary assessment
- Business impact assessment
- Setting change status to assessed

Once all tasks are completed, the change automatically goes to another phase, which is the approval phase, and starts the first tasks in the list. Two first tasks have workflow connected, which jumps the user from the change record to impact analysis application to allow the user to perform assessment easily.

1. Georg Bednorz checks his My Work portlet (Figure 5-50 on page 147) and searches for new tasks that must be completed by him. He can find one, called preliminary assessment. By clicking it he is redirected to the activities and tasks application to the specific task. There is a assisted workflow connected to the task, so when he clicks **Start Assisted Workflow**(Start Assisted Workflow) the workflow will redirect him to the change record where he can perform preliminary analysis. He checks whether CI is specified, classification is set, change schedule is defined, and so on. After reviewing the change record he sets the status of the task to complete to automatically start the following tasks.

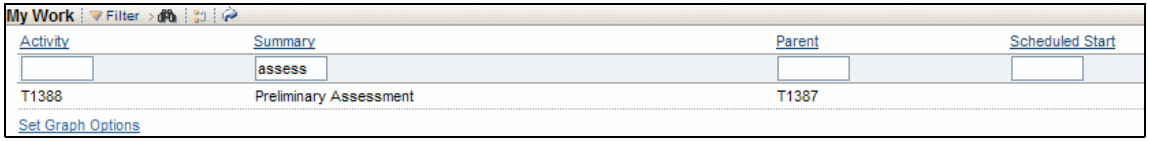


Figure 5-50 My Work portlet of Georg Bendorz

2. Leo Esaki checks his My Work portlet (Figure 5-51) and sees that there is one new task for him to do. He opens this task and clicks **Start Assisted Workflow**, which redirects him to the change record on the Impact Analysis tab. After reviewing the record and identifying any business impacts, he creates two assessments (financial and operational) in the Business Assessment Results sub tab. Then he completes the task. At this time the next task (setting change status to assessed) will be set to INPRG, but because there is an flow action connected to this task, setting change progress to assessed and setting task status to completed is done automatically. See Figure 5-52 on page 148 and Figure 5-53 on page 148.

Figure 5-51 My Work portlet of Leo Esaki

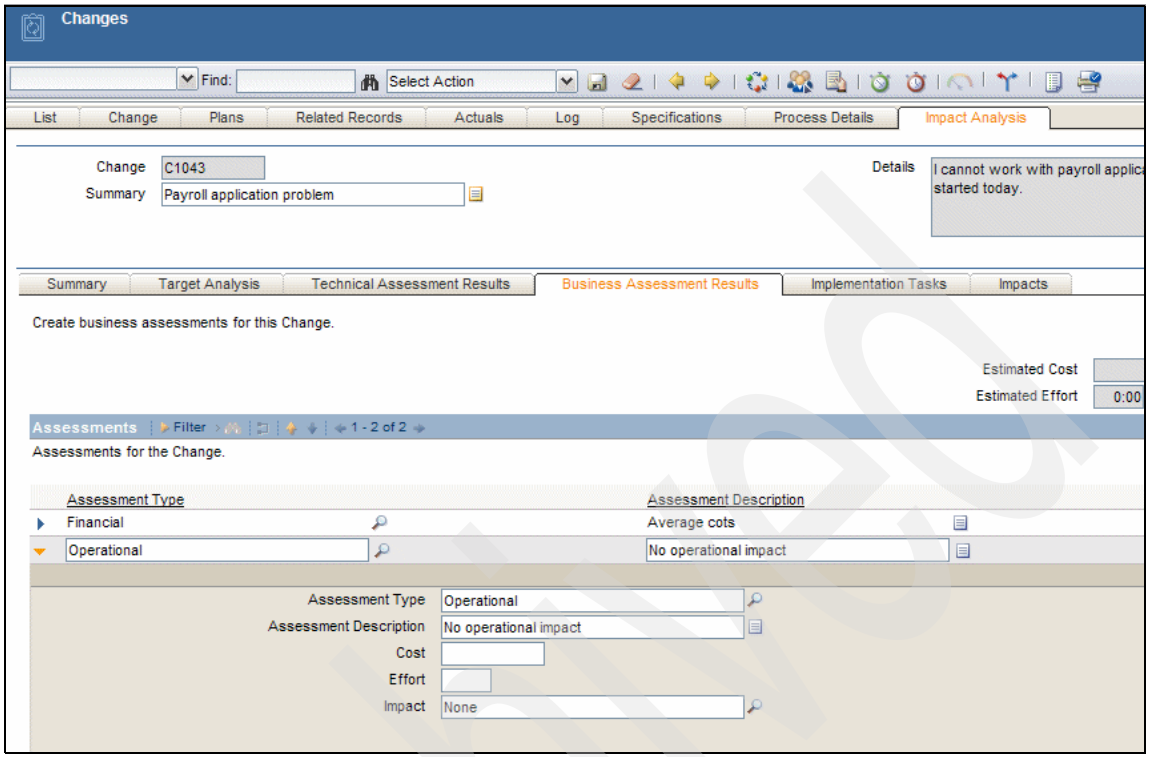


Figure 5-52 Two assessments made by Leo (financial and operational)

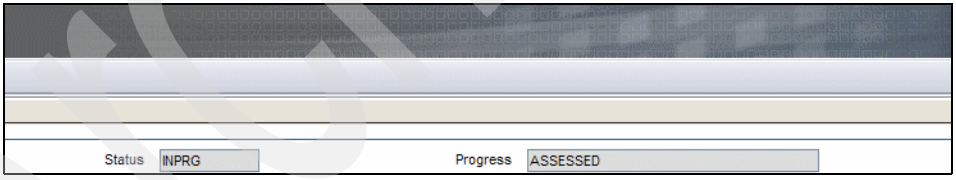


Figure 5-53 Change progress is automatically set to ASSESSED

Note: Check the job plan task functionality (flow controlled flag, flow action flag, and predecessors field) to learn more about how to manipulate the process flow using tasks, flow actions, and assisted workflows.

To learn more about the assessment phase and tools that are available to assess the change, go to Chapter 9, “Change impact assessment” on page 235.

Approval phase

The approval phase consists of three processes:

- Business approval
- Impacted CI approval
- Update Change Progress to approved (This task will be completed automatically when both approvals are completed.)

In the approval phase, business approval and Impacted CI approval have been defined to be at the same level, which means that those approvals can be done in parallel. Only the business approval phase will be described.

Note: The approval phase (as well as all other phases) can be customized to mirror the real-life Change Management process. We can define many templates with different phases and tasks (that is, for emergency changes we do not require many approvals, but for standard changes we require more). There are standard job plans delivered with the Change and Configuration Management Database, so check whether there is a need to create or customize already existing job plans or whether those plans are enough.

1. Alex Mueller opens a new task that he has in his My Work portlet. He goes to the change record to check the business assessment details and decides to approve the business aspect of the change by completing the task.
2. When all tasks are completed change progress is automatically set to approved and the scheduling phase takes place, as shown in Figure 5-54.

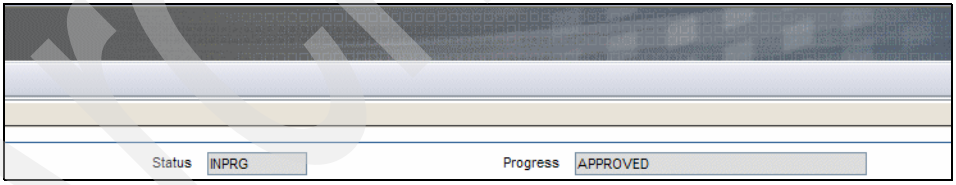


Figure 5-54 Change progress is set to approved

Schedule phase

The schedule phase (for this *emergency* job plan) has two tasks:

- Create implementation tasks and assign targets.
- Schedule implementation tasks and assign owners.

The first phase is to create implementation tasks from implementation notes, which may be created in the assessment phase. Implementation tasks are only tasks that will be scheduled in the scheduler application and only ones that will be able to change CI attributes.

After implementation tasks are created, the change owner (in this case Homer) will schedule them based on selected criteria:

1. Homer logs in and checks his work. He can see that there is one task assigned to him (create implementation tasks). He does not see a task for scheduling implementation tasks, as this task waits for creation of implementation tasks to be completed (predecessors functionality). He opens a task and using assisted workflow he is redirected to change record to the Impact Analysis tab to Implementation tasks subtab. He creates implementation tasks from Notes using the Create Task from Implementation Node button and then goes back to the task and sets the status to completed.
2. The next task for him is to schedule implementation tasks. He goes back again to the tasks application, searches for all implementation tasks for his change, and schedules them.

Implementation phase

Depending on the selected job plan/tasks and input from impact analysis (there is a possibility there to define tasks for implementation that will be created later), a various number of tasks will be created here. In the scheduling phase we have defined a task to assign owners to each one, so at the end of the schedule phase each task will have assigned owner. This is the real task that must be performed within the IT infrastructure.

Note: We can define a workflow or use integration framework to, for example, call a procedure on an external system. There is one action that is worth mentioning here, move/swap/modify. There we can manually change values of the CI attributes (or asset or location).

After all implementation tasks are done, an action to set the implementation phase status to complete is invoked, allowing the post implementation review phase to start.

Post implementation review

In the post implementation review phase we have defined three tasks

- Conduct change review.
- Update change status to closed.
- Update change progress to completed.

The idea of this phase is to force Change Manager to check whether the change was successful and if there are no more errors.

Kazmier logs in and looks for new tasks. There is one called conduct change review. He opens the task and from there goes to the change record. He uses

communication functionality to send e-mail to Homer (the change owner) to check the result of implementing the change. After he gets a response that everything was okay, he completes the task by setting the status to completed. It automatically closes the change and sets change progress to completed.

6

Scenario 2: Fixpack deployment

This chapter can be considered the continuation of the scenario that was discussed in 5.3, “Emergency Request for Change (RFC) scenario” on page 120. In this chapter we pick up where we left off in the previous chapter and describe the process that allows us, starting from a change request, to deploy software automatically on target systems.

This process is managed through the release application that is used to manage large-scale deployments of authorized versions of software to release targets. Release contents include the tasks, scheduling, and people or groups that are involved in the operation. The flow of activities can be customized to ensure that the release tasks are executed in the correct order by the authorized people.

This chapter contains the following sections:

6.1 DSL repositories

A Definitive Software Library (DSL) is a secure compound in which the definitive, authorized versions of software package configuration items (CIs) are stored and protected. A DSL consists of one or more software libraries or file-storage areas, referred to as repositories. The Definitive Software Library application can be used to create, edit, and manage DSL repositories.

Note: *Definitive Software Library* is a term used in Information Technology Infrastructure Library (ITIL) V2. ITIL V3 uses the term *Definitive Media Library* to describe a secure library where definitive authorized versions of all media CIs are stored and protected.

The DSL contains master copies of installable software. Each software image contains binaries and installation scripts. Often, a single software image consists of multiple software products. It is possible to manage software images manually, using the DSL application, or, if you have Integration Modules (IMs) installed for IBM Tivoli Provisioning Manager or IBM Tivoli Configuration Manager or both, you can discover the software images that are stored on these Operational Management products (OMPs).

Either of these OMPs enables us to deploy a software image to a set of targets. However, the process control provided by Release Process Manager adds auditable steps like approvals. In addition, the Change and Configuration Management Database (CCMDB) discovery process discovers both the systems that you deploy to and the configuration items in your environment. This product helps you identify the impacts of your distribution and adds significant audit, compliance, and control capabilities.

After you create a repository, you can specify software images that are associated with the repository for distribution. A repository that has an associated Operational Management product can be used as a source for automated software distribution tasks. Repositories that are not associated with an OMP can be used as store rooms from which you can select packages for manual distribution tasks.

Selection for the Definitive Software Library is from the CCMDB Main Panel: **GoTo** → **Release** → **Definitive Software Library**, as shown in Figure 6-1.

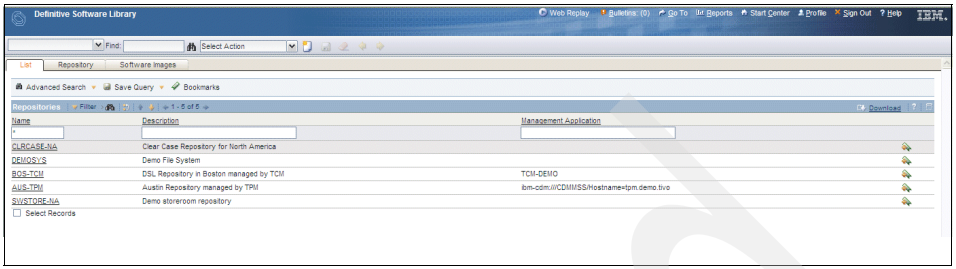


Figure 6-1 Definitive Software Library panel

Figure 6-2 shows the Software Library defined in our environment whose definition is *Austin Repository Managed by IBM*. These definitions point to Tivoli Provisioning Manager.

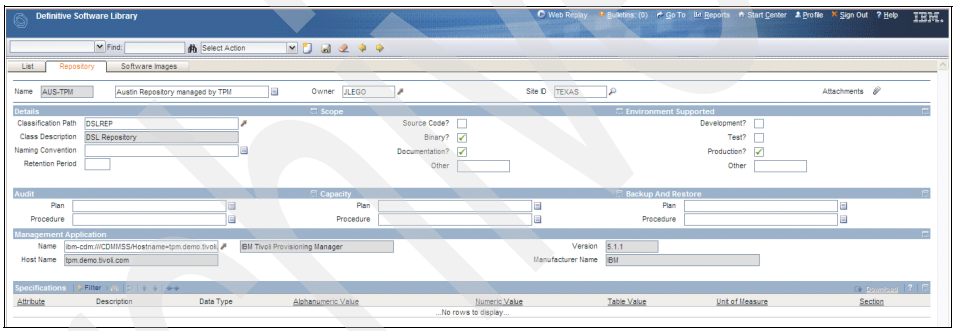


Figure 6-2 AUS-TPM Definitive Software Library: Repository

When a Definitive Software Library repository is created, it is necessary to populate the repository with references to one or more software image CIs. The CIs that have been selected to reference in a repository are either registered to the Configuration Management Database (CMDB) or stored on an Operational Management product. You can select CIs directly from an OMP that was defined as the management application for the repository, select CIs directly from CMDB, or create a new CI. In our environment OMP is the Tivoli Provisioning Manager. See Figure 6-3.

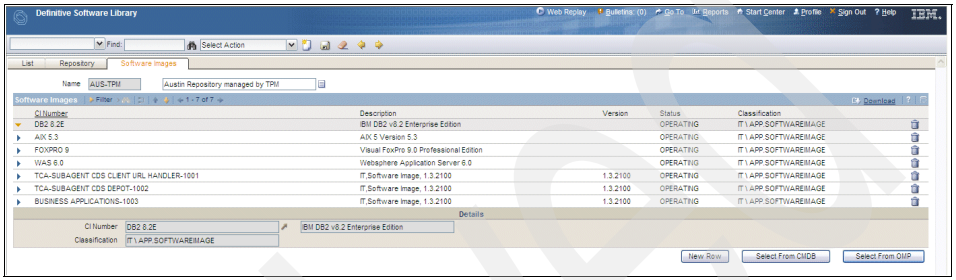


Figure 6-3 AUS-TPM Definitive Software Library: Software Images

Discovering and integrating OMP CIs

The following steps are required to discover the OMP servers that are available in the environment and the steps required to bring the discovered OMP configuration items into the product environment.

1. Run the discovery library adapter (DLA) for each of the OMPs from which you want to pull software images. For example, if you have IBM Tivoli Provisioning Manager for Software, run the DLA for this OMP. If you have IBM Tivoli Configuration Manager, run the DLA for that OMP. For more information about running a DLA, see the *Tivoli Application Dependency Discovery Manager (TADDM) Discovery Library Adapter Developer's Guide* at:

   `http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.taddm.doc_7.1/cmdb_dladevguide.pdf`

   After you run the DLA on an OMP, all of the managed systems for that OMP are registered as configuration items in your product environment. For example, you might discover an IBM Tivoli Provisioning Manager installation in Austin and another installation in Tokyo. Each of these systems would be defined as a CI.

2. Use the IBM Tivoli Integration Composer to move the OMP managed system CIs into the Maximo database.

   IBM Tivoli Integration Composer is described in the *IBM Tivoli Integration Composer System Administrator's Guide*. For instructions on using IBM Tivoli Integration Composer to move CIs into the Maximo database, see the *IBM*

Tivoli Integration Adapter for Tivoli Application Dependency Discovery Manager *Implementation Guide* at:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.ccmdb.doc_7.1/reference/adapter_ibmbook.pdf

Communication setup

To allow communications between CCMDB and OMP systems:

1. From the CCMDB main menu select **GoTo** → **Integration** → **End Points**.
2. Click the New End Point icon to open a view in which is possible create a new endpoint for the OMP.
3. In the Name field, enter a name for the endpoint. Optionally, you can type a description of the endpoint in the field beside the Name field.
4. Click the Select Value icon beside the Handler field and select **TPMHANDLER** or **TCMHANDLER** from the list, depending on whether the OMP for which the endpoint is created (Tivoli Provisioning Manager or Tivoli Configuration Manager).
5. Specify the port for the OMP server, the host name, and the password and user name that are used to log on to the OMP server for which you are creating the endpoint.

   In our environment the OMP Server is a Tivoli Provisioning Manager Server and 8777 is the IP port used for Web services. See Figure 6-4.

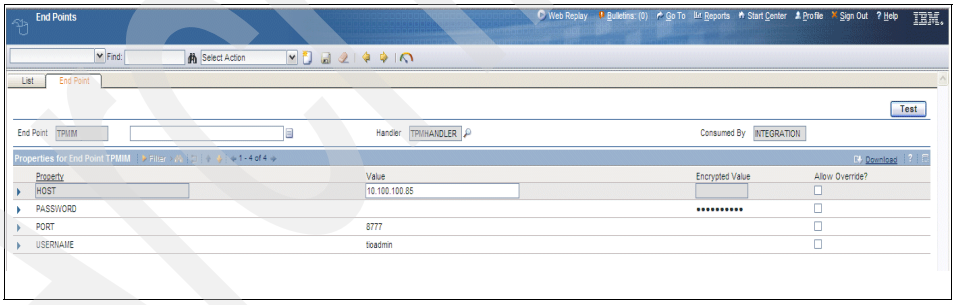


Figure 6-4 End points communication

6. Use this task to link the newly created endpoint with the discovered OMP and to specify the logical management operations (LMOs) that are supported for this OMP.

Link new endpoint with the discovered OMP

To link the endpoint with the OMP and specify supported operations:

1. From the main menu select **GoTo** → **Integration** → **Integration Modules**.
2. Click **Name** to list the Integration Modules that are installed in the environment.
3. Click the name of the Integration Module for the OMP that you are configuring. For example, click **TPMIM** if you are working with IBM Tivoli Provisioning Manager.
4. Open the **Operational Management Products** tab and click **New Row** in the Operational Management Products for OMPIM section.
5. In the new row, perform the following steps:
   a. Click in the Operational Management Product column and select the OMP that you are synchronizing with the environment.
   b. Click beside the End Point field and select the endpoint created in the end points application.
6. In the Logical Management Operations section, select the LMOs that this OMP must carry out, and click **Select Operations**. After these operations are specified, the new endpoint is associated with the OMP, and the OMP can be used for the operations that have been specified.

   It is possible to define a new LMO at any time to enable additional functionality from an OMP. To do so, it is necessary create an XML file that supports the new functionality, and go to the end points application to define an endpoint for the LMO, and specify the new LMO in the Integration Modules application.

   Figure 6-5 shows the Integration Modules window.

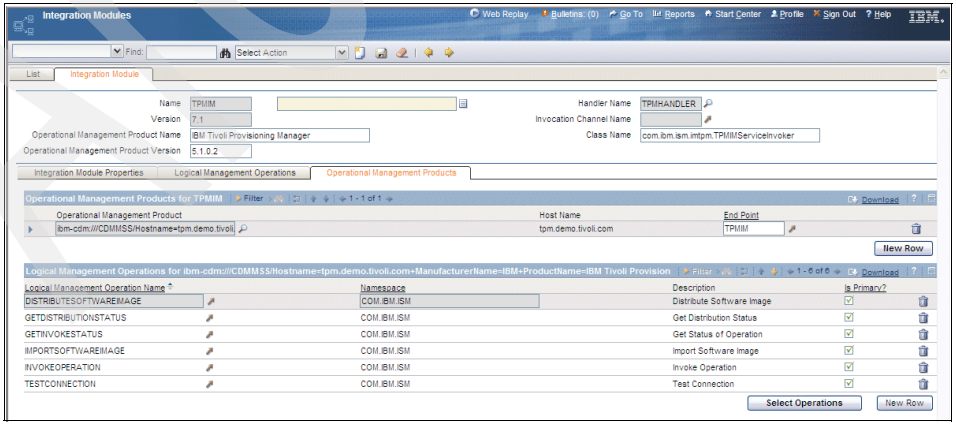


Figure 6-5 Integration Modules

6.2 IBM Tivoli Release Process Manager overview

IBM Tivoli Release Process Manager leverages best-practice ITIL process. Main functionalities of IBM Tivoli Release Process Manager are:

- Software rollout plans
- Software release tracking
- Agreements about content and a rollout plan for a release, according to the change manager process
- Assignment of activities and tasks based on roles.
- Schedule of multiple rollouts that have target configuration items in a single release
- Reports provided to show for all Releases, release progress, release status, and the CIs that are effected to this release.

IBM Tivoli Release Process Manager is integrated with Operation Management Products:

- IBM Tivoli Configuration Manager
- IBM Tivoli Provisioning Manager

Tivoli Release Process Manager requires the installation of:

- Discovery Library Adapter (DLA)
  - Tivoli Configuration Manager Discovery Library Adapter if TCM is the product used for software delivery
  - Tivoli Configuration Manager discovery library if TPM is the product used for software delivery
- Release Process Manager product
- Integration Module

  This component is installed on the CCMDB Server. As for the DLA, the Integration Module is available for Tivoli Configuration Manager and for Tivoli Provisioning Manager.

Installation should be in this order:

1. Discovery Library Adapter
2. Release Process Manager product
3. Integration Module

Post-installation steps are required to ensure that these Operational Management products are fully synchronized with the Tivoli Process Execution Engine database. It is necessary to run a discovery operation to discover the

OMP servers in the environment, move the managed system CIs into the environment, create associated endpoints in the Maximo database, and link the endpoints to the OMPs.

After that these post-Integration Module installation tasks are executed and all of the software packages that are stored on the OMP systems are available for distribution. It is possible to define the software repositories from which to pull the packages and select a specific software image as the source for each deployment that must be carried out.

These operations can be performed only if an Integration Module for each OMP server from which you want to deploy software is installed. In addition, the Discovery Library Adapter is installed for each OMP on the OMP server. For information about downloading DLAs, see the “Installing an OMP Integration Module” section of the Information Center:

http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.rpm.doc_7.1/rpm/t_rpm_installing_im.html.

These tasks must be performed after Integration Module installation and each time an OMP is added to the environment. Any OMP that is used to distribute software must be fully integrated and synchronized with the product environment.

In our environment we used IBM Tivoli Release Process Manager V7.1.1. Significant improvements are available with this version. One of them is the possibility of using Tivoli Provisioning Manager not only for the software distribution functionality provided with the software package editor interface, but also for supporting all logical device operations (LDOs). Another improvement available from Version 7.1.1 is the presence of three new built-in job plans that are available in the Release Process Manager component, service catalog for the Service Request Manager component and Change Management component. Each job plan contains nested activity-level job plans. To view, for example, job plans for release, from the main CCMDB menu select **Go To** → **Planning** → **job plans**, then filter with **PMREL**. See Figure 6-6 on page 161.

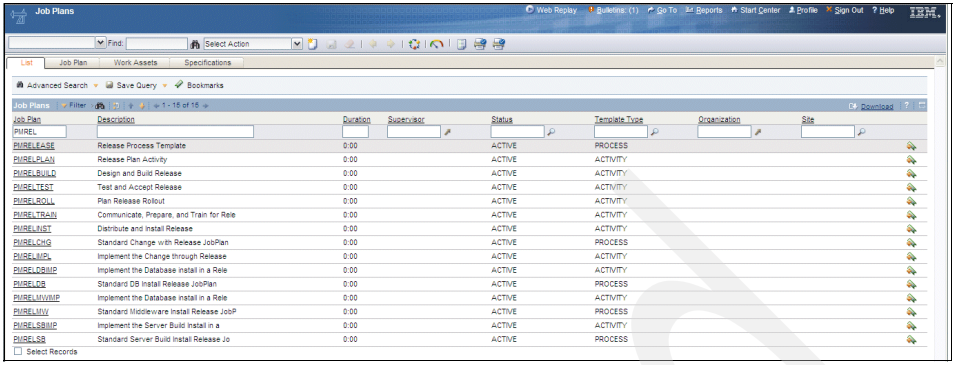


Figure 6-6 Release activity-level job plans

The product installation can be done using either the process solution command-line interface that is provided with CCMDB or the process solution installation wizard.

The installation wizard covers the main installation requirements but contains a subset of the installation operations that are available in the CLI program. The process solution command-line interface is useful, for example, when an unattended or silent installation is performed.

6.3 The release process

In Chapter 5, “Scenario 1: Incident, Problem, and Change Management” on page 117, our scenario was about:

- Identifying an incident
- Classifying the incident
- Creating an RFC
- Accepting and assigning the RFC
- Assigning a job plan
- Accessing the change
- Authorizing the change
- Implementing the change

Start from incident identification, then RFC creation starts a change that has an emergency flow. At the same time a new change is identified that requires us to implement a DB2 fix pack that resolves performance problems that arise on the payroll systems.

Release Process Manager V7.1.1 supports a release process that is closely aligned with the process defined in ITIL and ITUP processes.

The follow example is of a release process where the main phases are:

1. Plan.

   After a new release is created, you determine which changes will go into the release. Multiple changes can be added to a release. When you add a change to a release, the source and target configuration items for the change are also associated with the release. An impact analysis identifies dependencies that might affect additional related CIs across the data center. A basic task list is defined. You can apply a job plan that contains activities and tasks that meet the requirements of the release. Certain activities might be empty at this stage. For example, detailed rollout tasks are defined during the plan rollout phase. The overall structure of the release plan is the final goal of this stage.

2. Design and build.

   The installation scripts and packages to be deployed are designed and created during this phase. The software is not created, but it is packaged for deployment. Installation scripts and mechanisms, communication and education plans, and backout procedures are also developed.

3. Test and accept.

   During this phase, the release package is tested to ensure that is free of errors, and the release is reviewed to determine whether it can be accepted for deployment. Copies of the package are added to the Definitive Software Library. The Definitive Hardware Store is used for any hardware that is needed for deployment.

4. Plan rollout.

   Detailed plans, including release dates and deliverables, are created for rolling out the release to each site. For each site, delivery is scheduled for any new assets that are part of the release, and the release deployment is also scheduled. If training is needed, it is scheduled, as are any communications that are needed.

5. Communicate and prepare.

   During this phase, you ensure that all stakeholders, users, and support personnel are aware of the changes that will occur when the release is deployed. To ensure that a site is ready for a release, you might schedule site-specific testing, system shutdowns, reminders about the release, and so on.

6. Distribute and install.

   This is the phase where the distribution and installation occur. With CCMDB it is possible to verify the status of CIs that will be targets of release operations. If the component used for release deployment is Tivoli Provisioning Manager or Tivoli Provisioning Manager for Software, the release delivery can be the

deployment of a Logical Device Operation (LDO) or a software distribution package. If the component used for release deployment is Tivoli Configuration Manager, then the release delivery is a Software Distribution Package.

Figure 6-7 shows the release process.

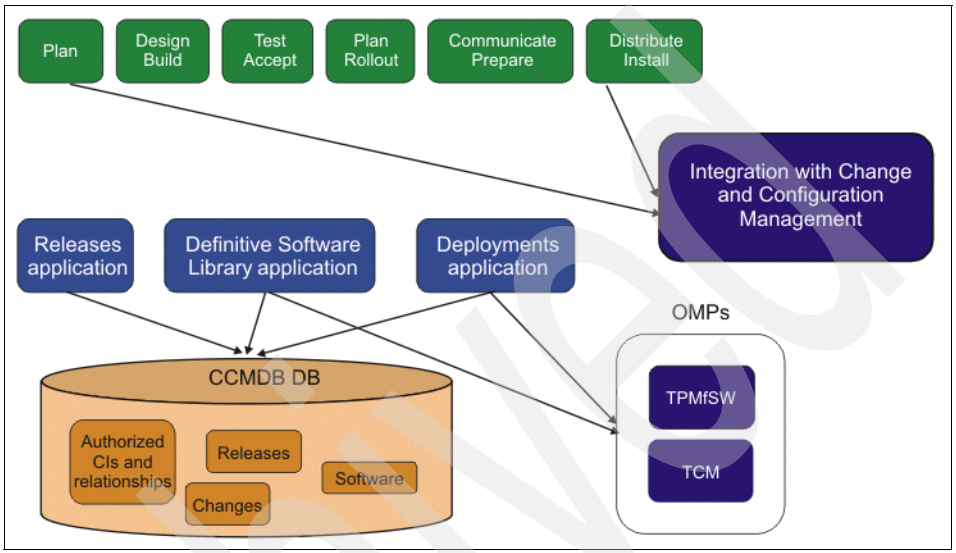


Figure 6-7 Release process

Creating a release

To create a release, it is necessary to specify some attributes, such as its classification, the type and scope of the release, the CIs that are source and target of the release, and so on.

1. To create a new release, from the CCMDB menu select **Go To** → **Release** → **Releases**.
2. Inside the release application choose the number button (#) on the right of Select Action window.
3. Add a short description of the release details in the Summary field.

4. Add a description in the Release Type field (Figure 6-8).

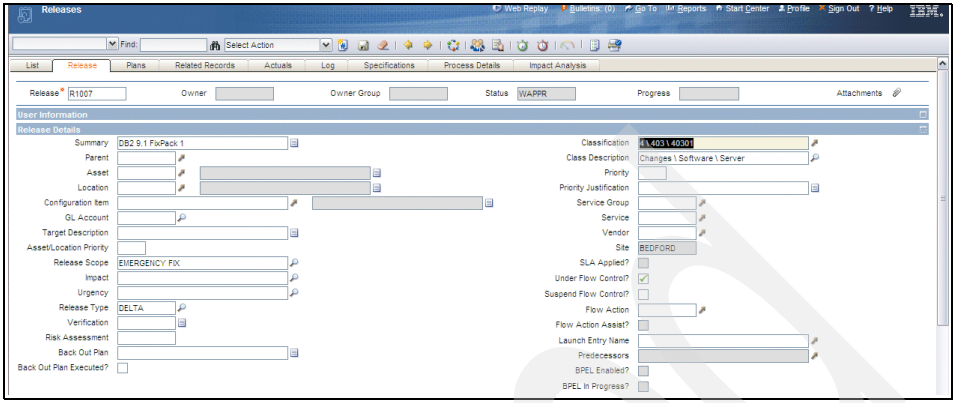


Figure 6-8 Releases application

5. Select **Classification** → **Classify** and you will see the classification tree, where it is possible to choose the classification value that best describes this release. We select **Classification changes\Software\Server** (Figure 6-9).

Figure 6-9 Classification tree

Autonumber setup information

In this section we see that every time an activity is created the prefix is T. When a change is created the prefix is C. When a release is created the prefix is R. This is the default customization of CCMDB. It can be changed based on your requirements and new prefixes can be added.

To do this select **GoTo** → **Administration** → **Organizations**.

In the Organizations panel select the **Organization** tab then choose **Select Action** → **Autonumber Setup**. Four options are now available:

- System Level
- Set Level
- Organization Level
- Site Level

Selecting **Organization Level** shows the autonumber name, the seed, and the prefix that are used. We can see the default prefixes for ACTIVITINUM, CHANGENUM, and RELEASENUM. See Figure 6-10.

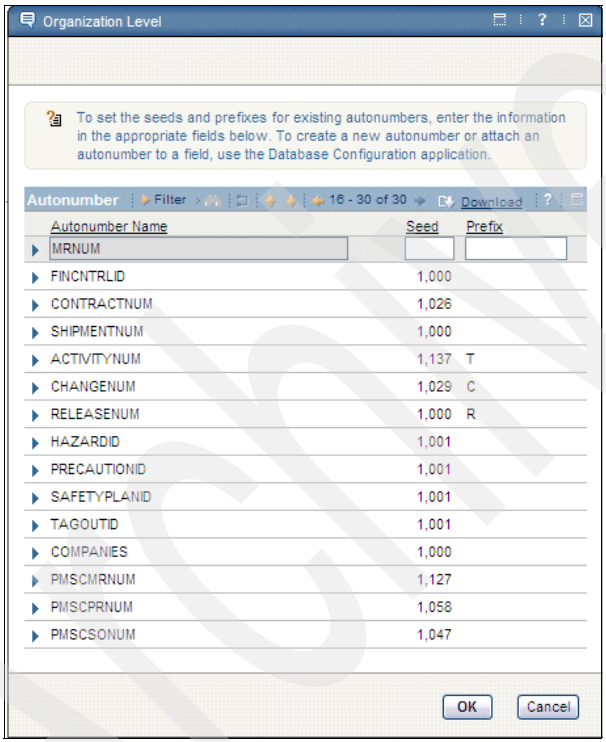


Figure 6-10 Organization Level prefixes

6.4 Change and release process interaction

The release process makes changes to one or more configuration items and CI changes are managed through the Change Management process.

The Change Management process and the Release Management process are tightly integrated.

Often simple changes are not associated with releases. For complex changes, such as a large-scale software deployment. The release process guarantees an ordered, , and auditable change.

A release must be associated with one or more approved changes. A change that is associated with a release cannot be completed until the release is completed.

Multiple changes can be associated with a release, but a change cannot be associated with more than one release.

Several operations are used to associate changes with a release. For example, it is possible create a change from within a release to handle CI changes that the release is planned to make. Additionally, a Change Manager can add approved changes to a release from within the changes application. This operation generates an *add change to release* request. An add change to release request can be accepted either within the releases application or from the ISM Request application. The request can also be rejected, and changes can be removed from a release.

The operations that are used to associate changes with a release are:

- Creating a RFC from a release

  This task can be used to create a Request for Change (RFC) from a release. When an RFC is created from a release, it is necessary to specify the RFC completion date that is required by the release. It is also required to specify any other information that defines the requested change.

- Adding changes to a release

  This task is used to request that one or more scheduled changes are added to a release. If an existing release is defined so that it can accommodate the change, you can specify this release. Otherwise, the change can be made available to any release.

- Responding to add change requests

  When a request is made to add one or more changes to a release, the requests are available for acceptance or rejection from within the releases application. If the request is to add the change to a specific release, the release owner of that release is notified, and he accepts or rejects the request. If the request makes the change available for any release, all release owners are notified, and one of them accepts the task of accepting or rejecting the request.

- Removing changes from a release

  It is possible to remove changes from a release in two ways:

  – From the change that has been added to the release
  – From the release to which the change was added

  When a change that was previously added to a release is removed, all of the configuration items or software packages that are associated with the change are also removed from the release.

- Responding to a remove change request

  When requests to remove changes from a release are submitted, the requests are available for acceptance or rejection from within the releases application. When a remove change request is accepted, the configuration items associated with the change are no longer associated with the release. When a request is rejected, the CIs retain their association with the release.

- Transferring a change to another release

  The Transfer Change to another Release dialog box is used to transfer one or more changes that are assigned to a release to a different release. It is possible to transfer a change to another release because the currently assigned release has been cancelled because another release is more appropriate for a change.

- Using process requests for change and release interaction

  Rather than using the change and release applications, it is possible to submit requests through the process requests application to add a change to a release or remove a change from a release. After one of these requests is submitted, the request is evaluated by a release owner and either accepted or rejected.

Example: Creating a RFC from a release

An RFC is requested from a release when the release requires changes to the IT environment. For example, memory must often be upgraded before a release can be deployed to an application server.

To create an RFC from a release:

1. If it is not already done, it is necessary perform these steps to open the Release that requires the RFC:
   a. Select **GoTo** → **Release** → **Releases** to go to the releases application.
   b. Select **Release** to display a list of the releases that are defined in you environment.
   c. Select the number of the release that requires the RFC.

2. Select **Create** → **Process Request** from the Select Action menu to open the process request application (Figure 6-11).

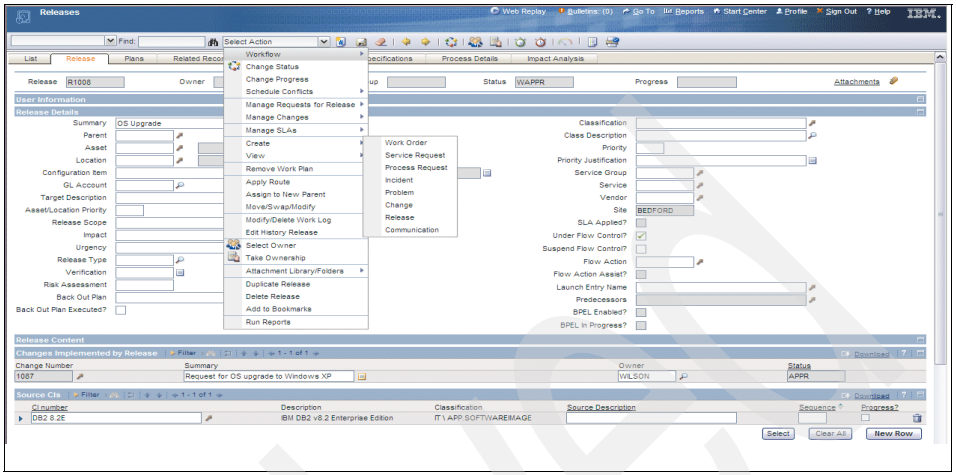


Figure 6-11 Creating a RFC from a release

3. Select the New Request icon number sign (#) and specify the following information to define the new process request:

   a. Type a short description of the requested change in the description field. The description should clearly identify the needed change so that other users can easily identity what is requested.

   b. Select the **Process Manager Type** field, select **Classify**, and click the blue box beside the classification value that best described the purpose of the request. The Classification and Class Description fields are populated based on the selection, and any attributes associated with the classification are displayed in the Classification Attributes section. Optionally, specific information can be entered to define the attributes.

c. Enter values in other optional fields as appropriate, then save the process request. See Figure 6-12.

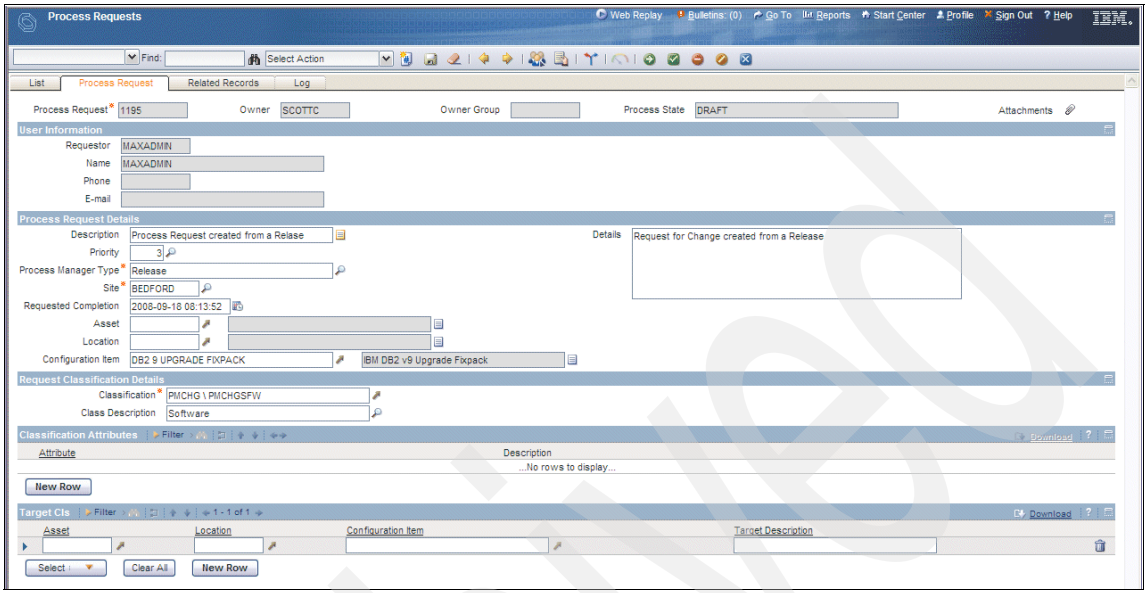


Figure 6-12 A request for change created from release

Example: Add a change to a release

This example shows how starting from the Changes panel we choose **Select Action → Release Requests**. As shown in Figure 6-13**,** available menus are:

- Add a specific Release
- Make available for any Release
- Remove from Release
- Cancel outstanding Requests

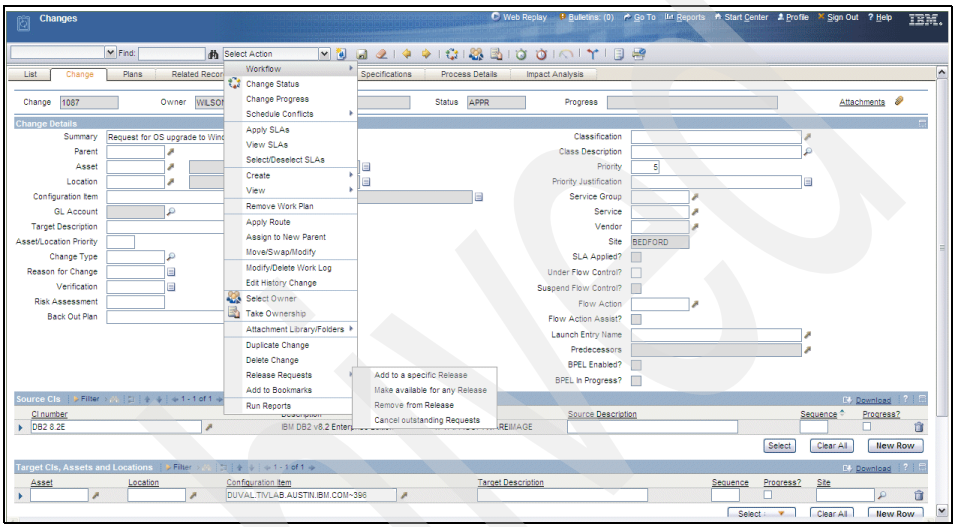


Figure 6-13 Available menus

We choose to add this change to a specific release, R1008, so by going to the Releases panel and selecting **R1008** it is possible to see that release R1008 has, for changes associated to the release, the change number 1087 (Figure 6-14).

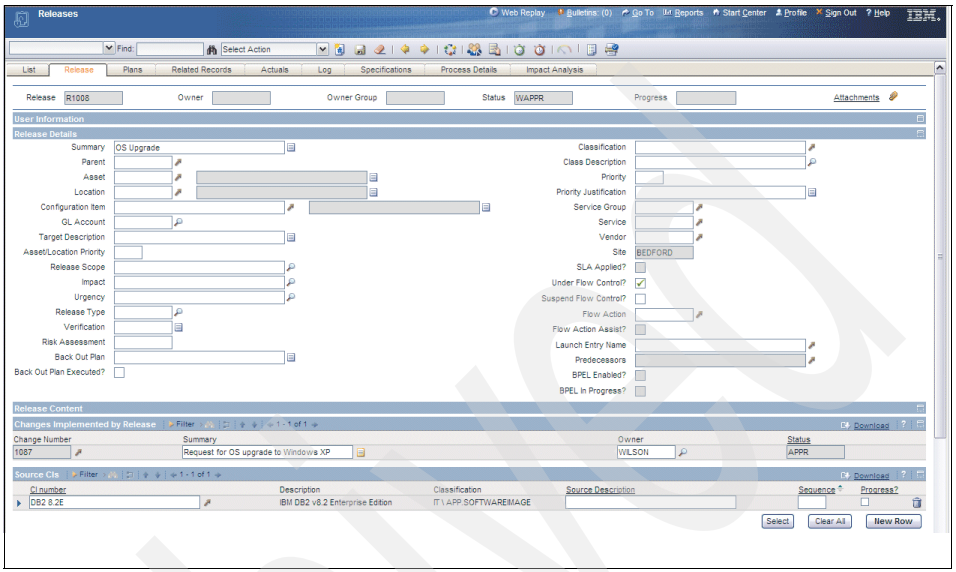


Figure 6-14 Changes implemented by release

6.5 Scenario: Fix pack deployment

Starting from the scenario in 5.3, “Emergency Request for Change (RFC) scenario” on page 120, during the investigate and diagnose phase two RFCs are opened:

- An emergency RFC
- A normal RFC

The normal RFC follows the path shown in Figure 6-15.

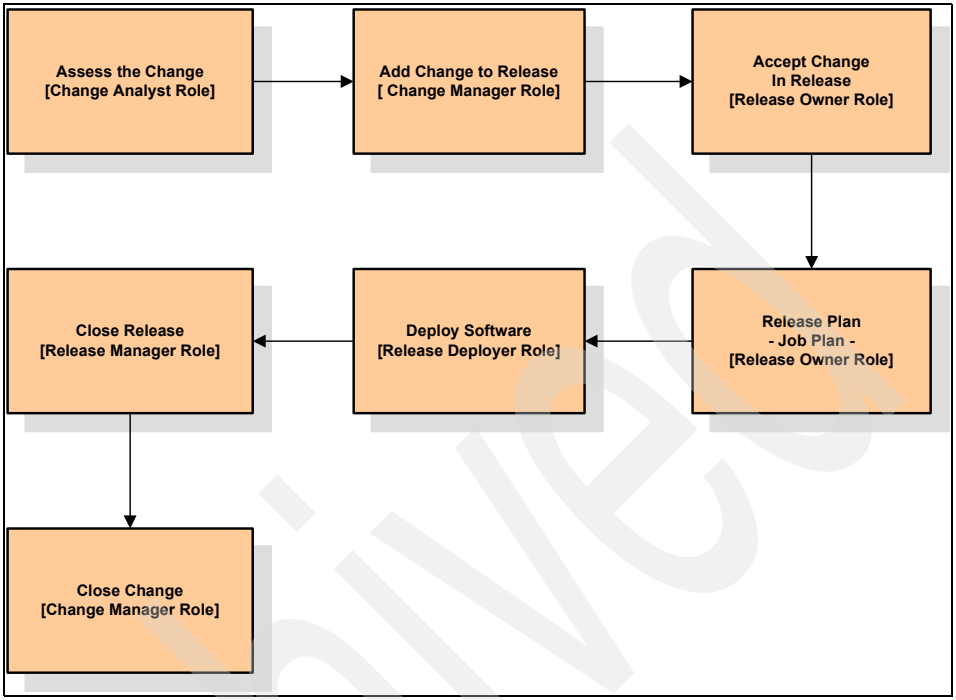


Figure 6-15 Flowchart of release process

In our scenario the change assessment is done by the change analyst. When approved, the change is added by the change manager to a release. The change in release is accepted by the release owner who defines a release plan. The software deployment is executed by the release deployer. After the release has been applied the release is closed by the release manager, then the change is closed by the change manager.

In the previous scenario the change was created by the change analyst (in our environment George Bednorz of the group PMCHANGEANALYST), then it was approved by Gerd Binnig (he is the change approver of group PMCHANGEAPPROVER). These steps were detailed in Chapter 5, “Scenario 1: Incident, Problem, and Change Management” on page 117.

After the change is approved it is assigned to the change owner. In our environment the change owner is Homer Simpson (the group is PMCHANGEOWNER). When Homer logs on toStart Center the SmartCenter he can see that a change was assigned to him. (When logged into the Start Center select **List View** in the All my active Changes window. Otherwise, the graphical view is displayed.) See Figure 6-16.

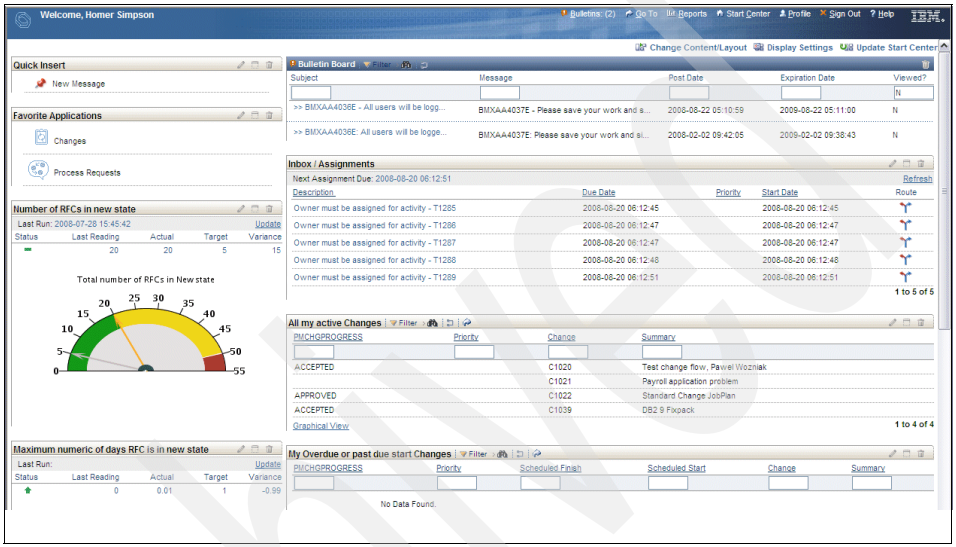


Figure 6-16 Start Center view of the change owner

When Homer logs on to Start Center he sees that there is change 1039 assigned to him that is related to a DB2 fix pack that must be installed on payroll systems.

As the change owner, Homer recognizes that a typical release process operation must be implemented. Adding the change to a release is required.

This scenario needs some configuration steps in the change process to have all authorization phases defined before the process is implemented. So what is required is the definition of a job plan and the creation of nested job plans. A sequence is also created, so only when all nested job plans are completed is master job plan also completed.

Figure 6-17 shows the parent job and nested job plans. During the environment preparation the status was in draft mode. When the configuration is complete the status of the job plan and its nested job plans must be set to active (Figure 6-17).

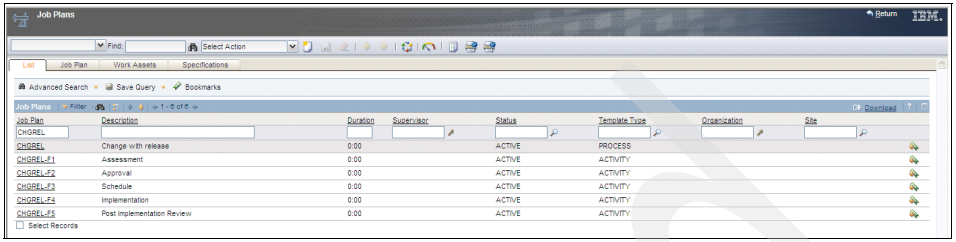


Figure 6-17 Job plan and nested job plans

For the change process the following job plan and nested job plans have been created (CHGREL):

- CHGREL-F1
- CHGREL-F2
- CHGREL-F3
- CHGREL-F4
- CHGREL-F5

The main job plan manages the overall process through the nested job plans. The phases are:

1. CHGREL-F1: Assessment
2. CHGREL-F2: Approval
3. CHGREL-F3: Schedule
4. CHGREL-F4: Implementation
5. CHGREL-F5: Post implementation review

Every nested job plan contains tasks necessary for phase completion. For example, for CHGREL-F1 three different tasks are defined:

- Preliminary assessment.
- Assess business impact.
- Update change progress to release.

For every task defined, the owner (and when available the flow action and the assisted workflow) is copied directly to the next application window.

When a predecessor is defined (such as for the update change progress to assessed task), it means that the predecessor must be completed first, so the next task goes to the inprogress state. See Figure 6-18.

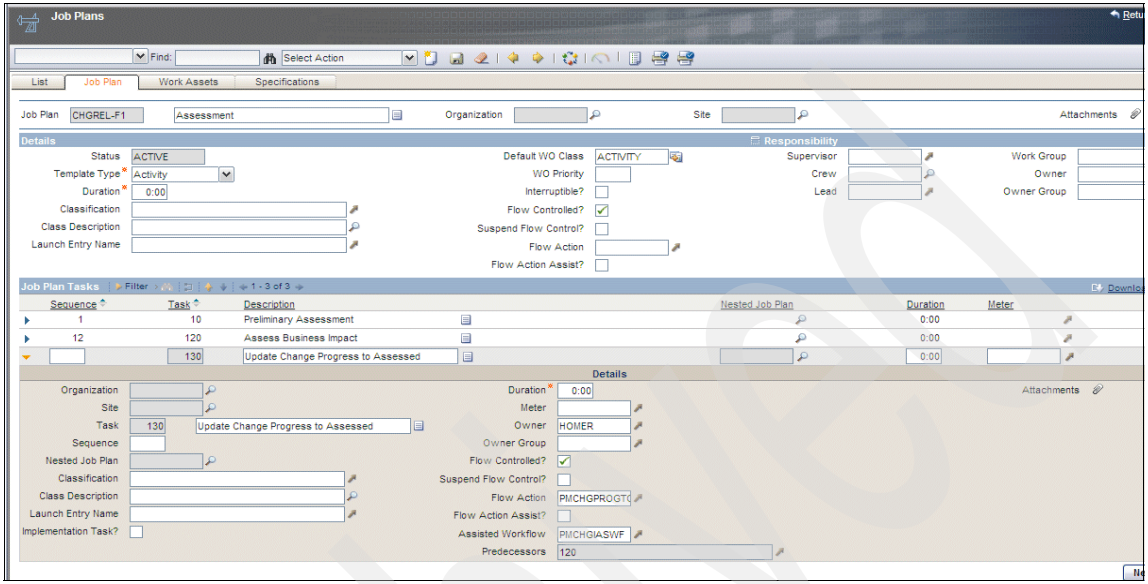


Figure 6-18 Nested job plan CHGREL-F1

Figure 6-19 shows the nested job plan CHGREL-F4 where at “wait for release tasks to be completed” task starts an escalation process.

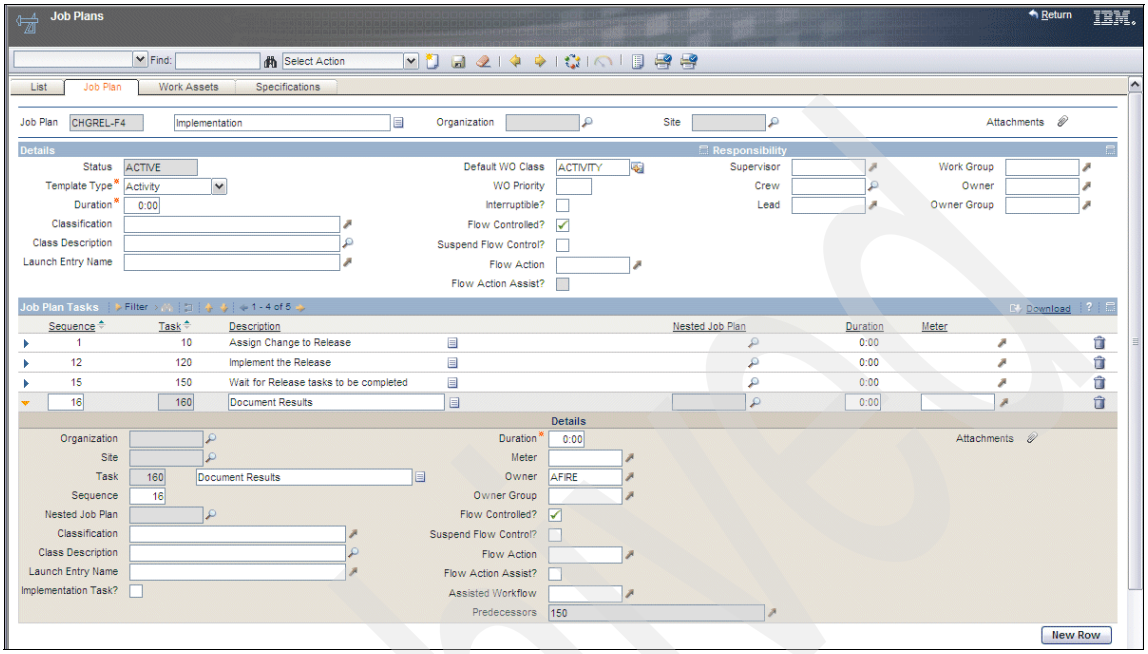


Figure 6-19 Nested job plan CHGREL-F4

The Release Process Manager provides five different predefined escalation processes. Theses can be accessed by selecting **GoTo** → **System Configuration** → **Platform Configuration** → **Escalations**.

We focus now on the release part of the process. We provide examples showing the interaction between change and release and how a change can be added to a release.

An *add change to release* request is typically generated when a change is too complex to be managed through Change Management alone. For example, a change might encompass numerous workstations that are to receive a major software update. These kind of changes are best accomplished through a planned release process.

If the change is added to a specific release, the owner of that release is notified. If the change is made available for all releases then all release owners are notified. Then the request is evaluated by a release owner and either accepted or rejected. If the change is available for any release and no current release

satisfies the requirements of the change, a release owner can create a new release to handle the change.

The change owner Homer assigns the release to Andrew Fire, who is the release deployer. See Figure 6-20.

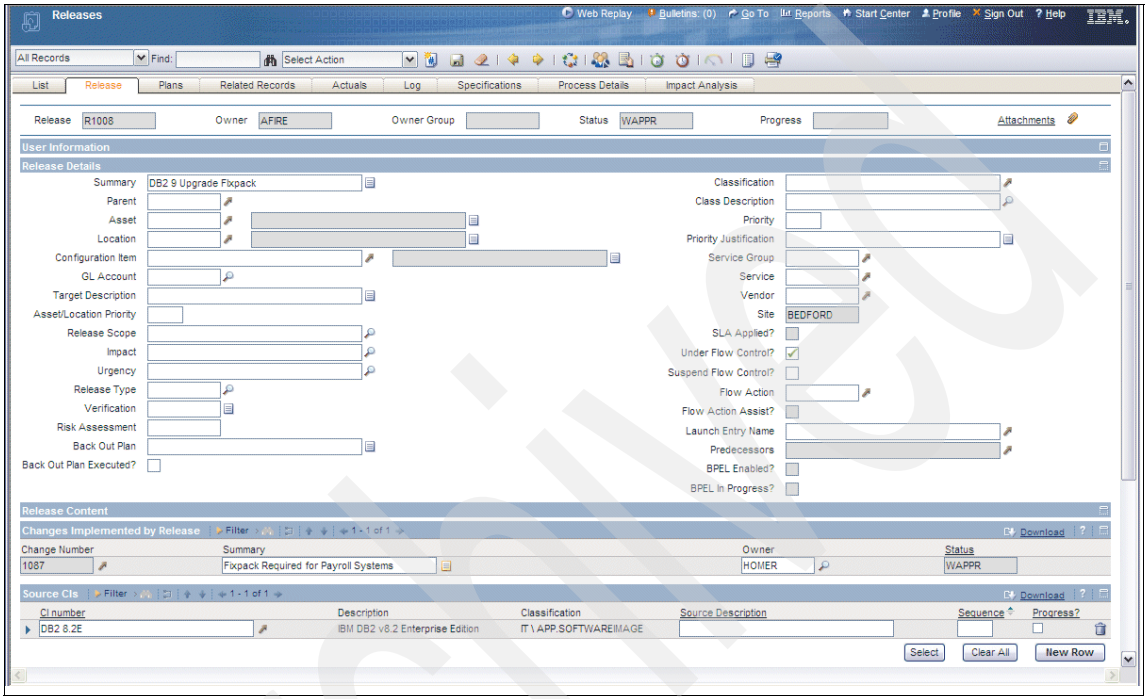


Figure 6-20 Change assignment to release

When Andrew Fire logs onto the SmartCenter he finds that a release has been assigned to him, as shown in Figure 6-21.

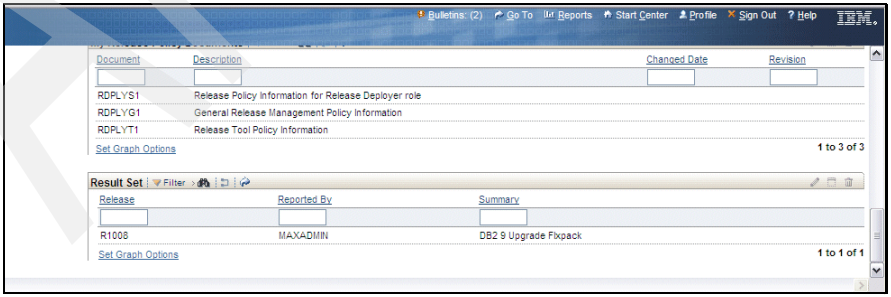


Figure 6-21 SmartCenter of Andrew Fire (release deployer)

In Figure 6-21 on page 177 we can see in detail which is the CI associated with the change.

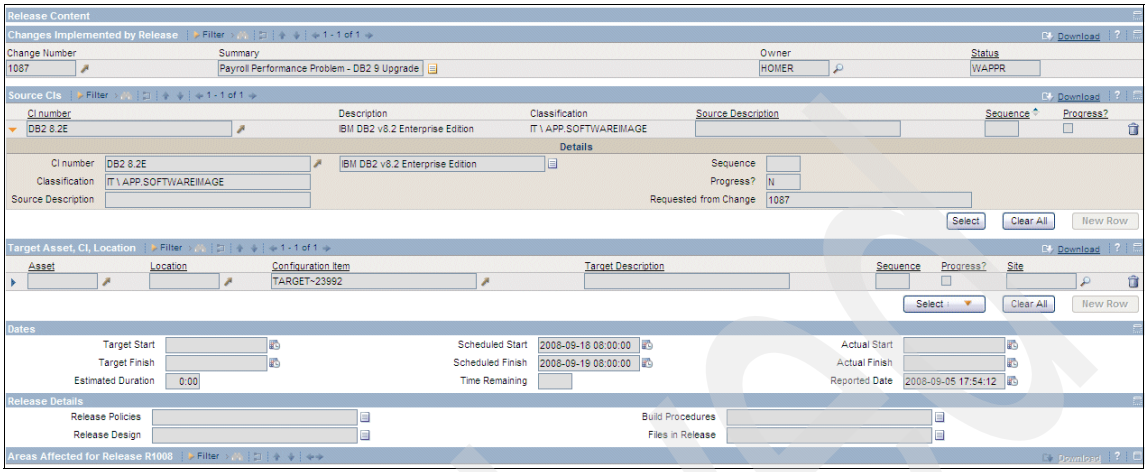


Figure 6-22 The release content, the CI number, the source CI

Deployment phase

The deployment of the release is executed through the page **GoTo** → **Release** → **Deployments**. From this window, the deployer of the release (in our environment, Andrew Fire) can deploy the software package (or LDO if TPM is used) on release targets.

In the Deployment window it is shown that the address of the Web services used to communicate with the Operational Management product in our configuration is TPM.

The connection to TPM is done through the following Web services:

```
ibm-cdm:///CDMMSS/Hostname=tpm.demo.tivoli.com+ManufacturerName=IBM+
ProductName=IBM Tivoli Provisioning Manager
```

In the same panel the OMP manufacturer name, the OMP version, the OMP host name, and the OMP product name are defined.

In the source information the CI name identifies the software package that will be installed on release targets, its description, and the definition of the Definitive Software Library.

In the target information fields the information related to targets is displayed. In our test environment only TARGET 23992 was available. In a release production environment, target information will include all the systems that are targets for the release process. See Figure 6-23.

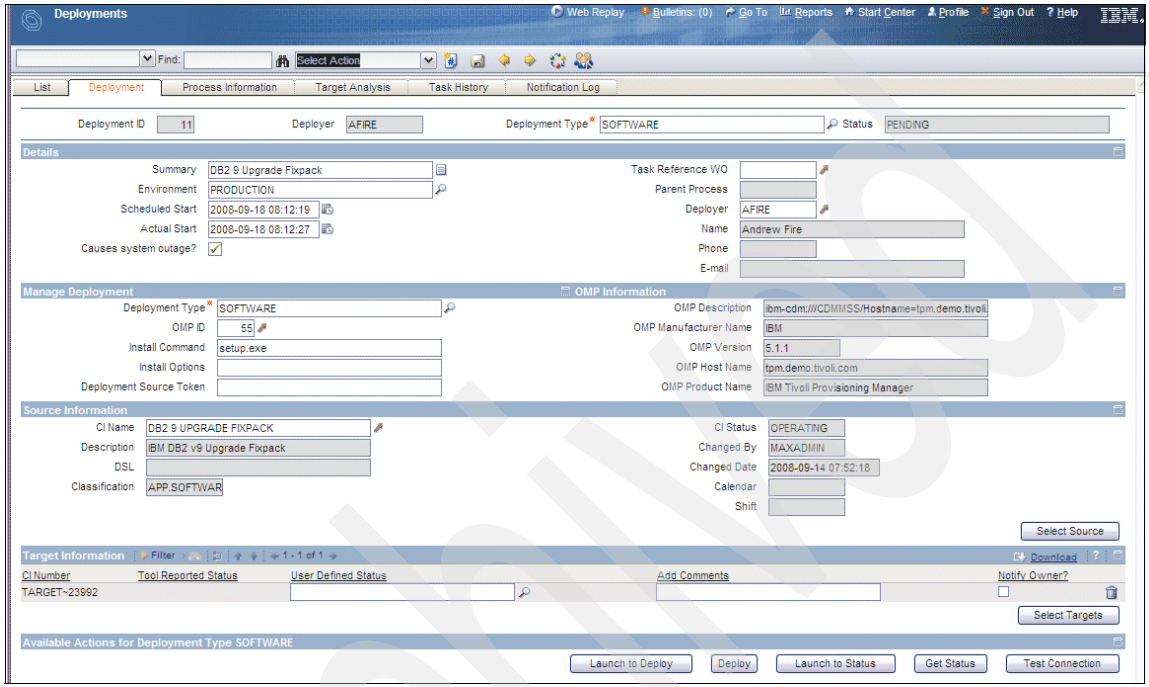


Figure 6-23 Deployment of the fix pack

At the bottom of the window the following functional buttons are available:

- Launch to Deploy: Launches the TPM or TCM Web client for that deployment
- Deploy: Kicks off a new deployment (This is an asynchronous call.)
- Launch to Status: Launches to the deployment application showing the status of the deployment
- Get Status: Makes a call to TPM or TCM to get the current deployment status and plugs it in the deployment status field in the deployment application
- Test Connection: Executes the test connection LMO and displays whether the TPM/TCM connection is successful

CCMDB makes available the launch-in-context (LIC) option that permits at any time the direct connection to displays of different components. In our environment the LIC is configured for TADDM, TPM, and TBSM. The Launch in

Context panel is available by selecting **Goto** → **System Configuration** → **Platform Configuration** → **Launch in Context**.

For TPM two LICs are available:

- PMRELGETSWDISTSTATUS: Launch to status on TPM.
- PMRELOPSSTATUS: Launch to TPM LDO status.

Figure 6-24 shows the status of TPM software distributions and LDOs.

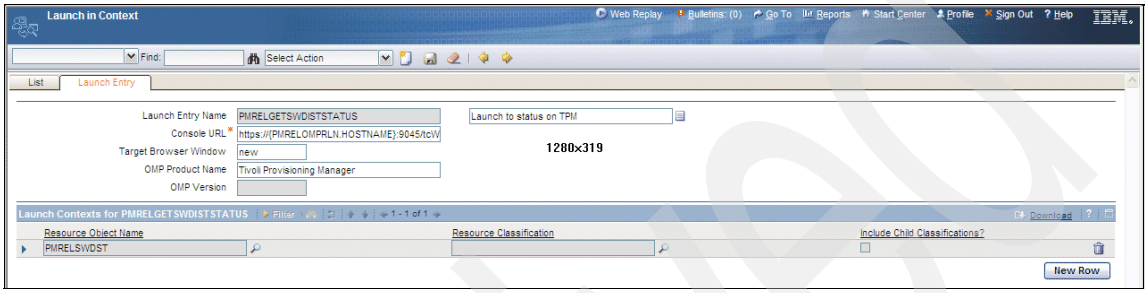


Figure 6-24 Launch in Context for TPM software distribution get status

The deploy action started the software distribution on the OMP system (in our environment, the Tivoli Provisioning Manager). The results of the software distribution are available in the CCMDB or directly on the TPM URL address by selecting **Automation** → **Workflow Status**, as shown Figure 6-25.

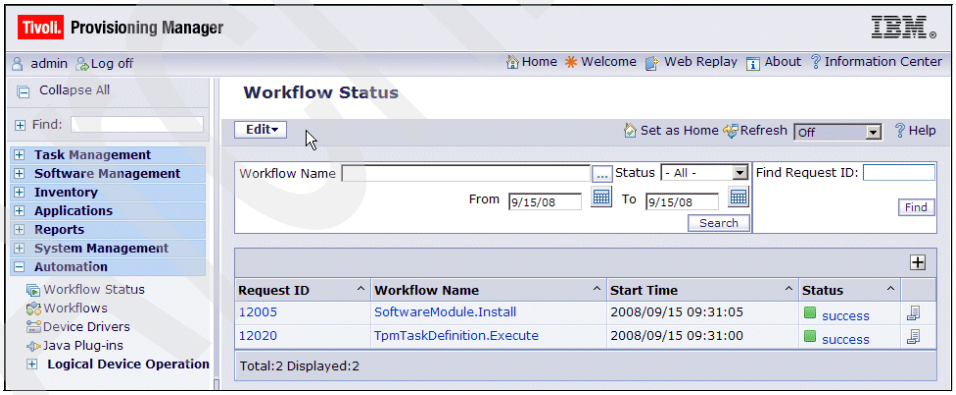


Figure 6-25 Distribution status on Tivoli Provisioning Manager

Escalations

Escalations enable you to automatically monitor critical steps in a change process. escalations ensure that critical tasks are completed on time, change the status of a record, or change the owner of a record.

The escalation process works in the background. The escalation PMCHREL is scheduled every hour and verifies whether the release deployment is completed. When this happens, the release status changes to COMP and the wait-for-release task also changes to the COMP status. The change that generated the release has a relationship with the release. This means that when the release is completed the change is closed.

The escalation time should be accurately set up to avoid the problem that arises when the default time for the associated escalations is set to a time period that is longer than appropriate.

The three escalations that affect wait-for-release tasks are PMCHGREL, PMCHGWAIT, and PMRELWOSTA.

Escalations panel selection is through **GoTo** → **System Configuration** → **Platform Configuration** → **Escalations**. See Figure 6-26.

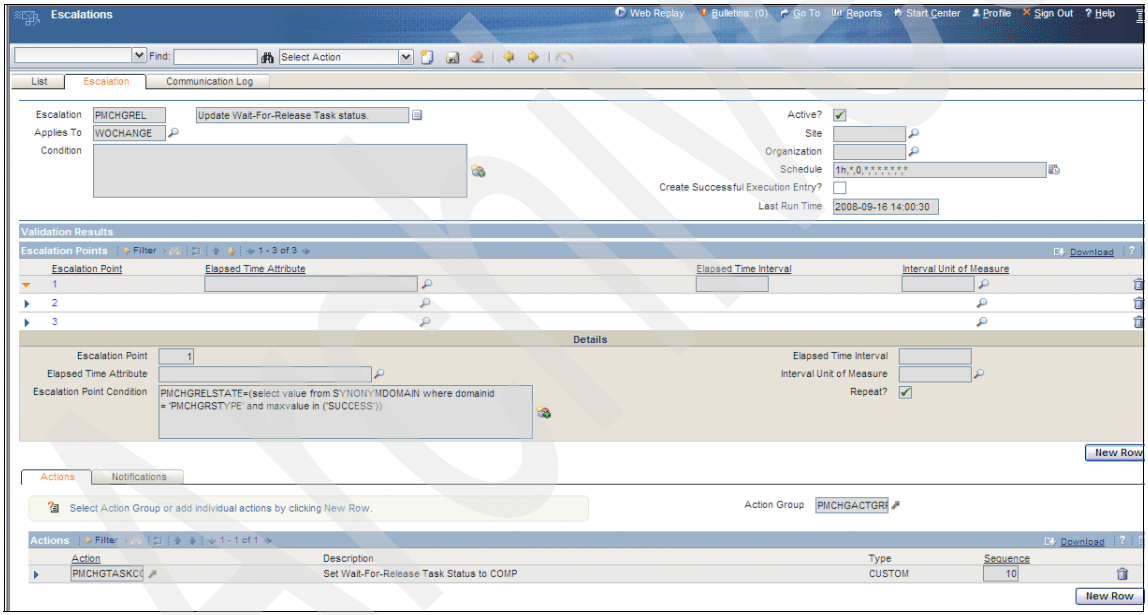


Figure 6-26 Escalations

7

Scenario 3: Process level integration with an external Service Desk system

The scenario described in this chapter shows the integration of IBM Service Management products with an external Service Desk system such as HP OpenView ServiceCenter. The rationale behind this scenario is that there are some organizations with a Service Desk system installed already, probably due to merger & acquisitions or some other historical decisions. Nevertheless, they would like to integrate the existing Service Desk system with the Tivoli Service Management products for the rich functionality and more automation.

This chapter has the following sections:

7.1 Scenario introduction

One of the challenges in today's IT environment is to integrate between systems. It is not a question to have integration, but question is how can we integrate, to what and how easily. Integration leads to less data redundancy, increases functionality of the IT system and makes process management more easy to handle- having whole picture of the process we can make right decisions - in opposite to knowing only about "your part" in it.

We have identified three models of integration: User interface integration, data integration and process integration. User interface integration, also called launch-in-context, is all about launching the user interface of other applications from the IBM Service Management (ISM) application user interface within context. Data integration is about moving it between IT environments so we can use the same data in many places where process integration is about using different tools to encapsulate the whole process and using the same data. We want to benefit from different tools and different functionality to improve decision making process and reduce cost of implementing new functionalities by reusing already created ones. In our case, we will deal with processes like Incident Management, Change Management, Release Management, etc. We will show Tivoli process automation engine capabilities to integrate with systems like IBM Netcool, HP Service Center, Change and Configuration Management Database (CCMDB) and TSRM to demonstrate process flow from discovering a problem using monitoring tools to releasing a fix pack. We will demonstrate different approaches to integration by using core integration functionality delivered with CCMDB, called Integration Framework, know also as Maximo Enterprise Adapter (MEA), as well as using Tivoli Directory Integrator as a general solution when integrating process on data level.

For more information about different levels of integration with IBM Service Management products, see 2.3.2, "Integration categories" on page 51.

7.2 Scenario details

For this scenario, the following products are used:

- IBM Tivoli Monitoring (ITM) V6.1 is used for infrastructure resources monitoring.
- IBM Tivoli Netcool/Omnibus (Omnibus) V7.1 is used for Event Management.
- IBM Tivoli Business Services Manager (TBSM) V4.1 is used for business level monitoring.

- IBM Tivoli Netcool/Omnibus gateway for HP ServiceCenter V6.0 is used as the integration tool to automate the generation/update of ticket in HP Service Center base on the status from Omnibus.
- IBM Tivoli Change and Configuration Management Database (CCMDB) V7.1.1 is used for change and Configuration Management.
- HP OpenView ServiceCenter (HP/SC) V6.1 is used for Service Desk.
- IBM Tivoli Directory Integrator (TDI) V6.1.1 is used as the integration tool for synchronizing the change records between CCMDB and HP OpenView ServiceCenter.

Process flow

Refer to the Figure 7-1, the process flow of this scenario is described as below. Please note that some steps as shown by the boxes below the dotted line are fully automated, and does not require human intervention.

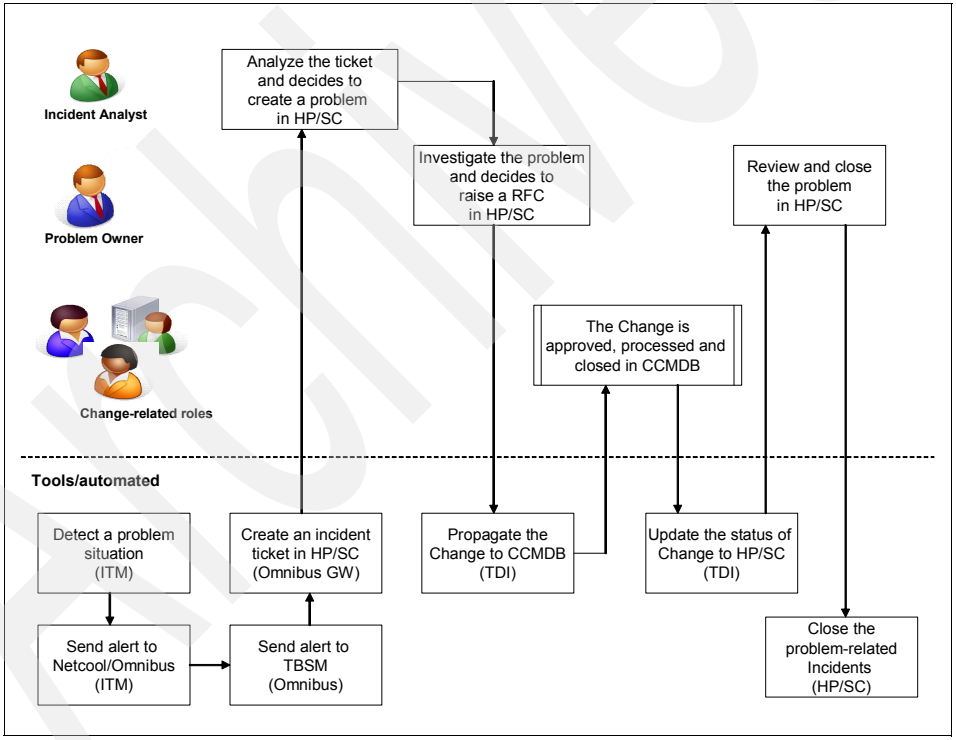


Figure 7-1 Process flow for integration with external Service Desk system

The process is:

1. ITM detects a problem, for example CPU utilization is too high for the business application server.
2. ITM sends an alert to Netcool/Omnibus.
3. Netcool/Omnibus sends an alert to TBSM.
4. Netcool/Omnibus also creates an incident ticket in HP/SC via the Netcool/Omnibus gateway for HP ServiceCenter.
5. After analysis of the incident ticket, the incident analyst decides to create a problem in HP SC.
6. After investigation of the problem, the problem owner raises a Request-for-Change (RFC) in HP/SC.
7. The RFC is propagated to the CCMDB as a Change via the TDI Assembly Line.
8. The Change is approved, processed and closed in CCMDB.

> **Note:** To avoid duplication with previous sections in this chapter, this is shown as a compound process here. Please refer to scenarios 1 and 2 for how this can be done.

9. The status of the Change is updated to HP/SC via TDI.
10. After reviewing the Change is closed, the problem owner close the problem in HP/SC.
11. The incident ticket in HP/SC is closed after the close of the problem.

7.3 Implementing the ITM/Omnibus/HP ServiceCenter Integration

In this section, we will explain the integration in our laboratory environment between IBM Tivoli Monitoring (ITM), Netcool/Omnibus and HP ServiceCenter. In the ITSO laboratory we installed TBSM 4.1 instead of just Netcool Omnibus on its own.

7.3.1 Integration environment

We have five servers in our integration environment for this scenario (Figure 7-2 on page 188):

- Server A is a MS Windows-based server running DB2 UDB Enterprise Server 8.2 and the ITM DB2 agent software.
- Server B is a MS Windows-based server running ITM 6.2 and Tivoli EIF Probe from the TBSM 4.1 installation images.
- Server C is a MS Windows-based server running TBSM 4.1 and Netcool Omnibus 7.1.
- Server D is a RedHat Linux-based server running the Netcool HP ServiceCenter Gateway program.
- Server E is a MS Windows-based server running the HP ServiceCenter server.

Note: The Netcool HP ServiceCenter Gateway program is not supported on the Windows platform. Solaris™, AIX, HP-UX, and Linux are supported.

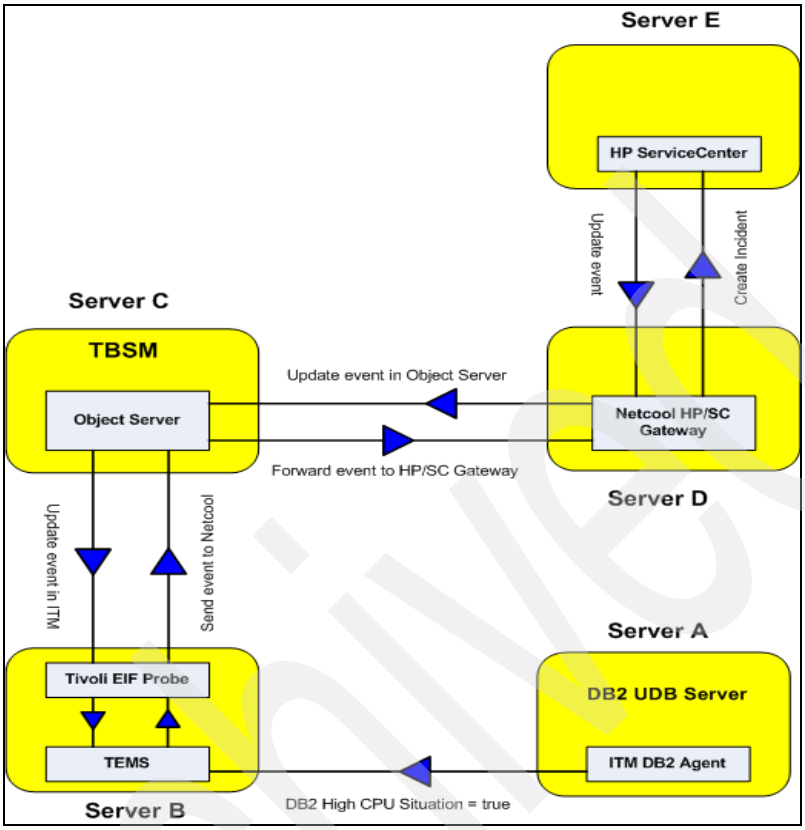


Figure 7-2 ITM and Netcool Omnibus environment

7.3.2 Implementation steps

To implement the integration solution:

1. Read the installation documentation that comes with the integration solution.
2. Check that the prerequisites are met.
3. Acquire the information required on the planning worksheet.
4. Install and configure ITM 6.x (TEMS/TEPS) (server B).
5. Install and configure ITM 6.x DB2 Monitoring Agent (server A).
6. Create a DB2 monitoring situation in ITM (server B).
7. Install and configure TBSM 4.x or Netcool Omnibus (server C).
8. Install and configure Tivoli EIF Probe (server B).

9. Install and configure Netcool HP ServiceCenter Gateway (server D).
10. Configure HP ServiceCenter for integration with Netcool Omnibus Gateway (server E).

> **Note:** TBSM does not have to be installed. You could install Netcool Omnibus on its own to achieve the same result.

For a much more detailed explanation of the integration between ITM and Omnibus, refer to the *IBM Tivoli Monitoring 6.x Installation and Setup Guide*:

http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/topic/com.ibm.itm.doc_6.2.fp1/itm_install.htm

In the following sections we provide more details about steps 6 and 8.

7.3.3 Create DB2 monitoring situation in ITM

In this step you must define a new ITM situation to monitor DB2. To do this:

1. Log in to the Tivoli Enterprise Portal as user SYSADMIN.
2. Click the Situation Editor icon. Within the Situation Editor click the Create new situation icon.
3. Create a new DB2 situation using the example shown in Figure 7-3.

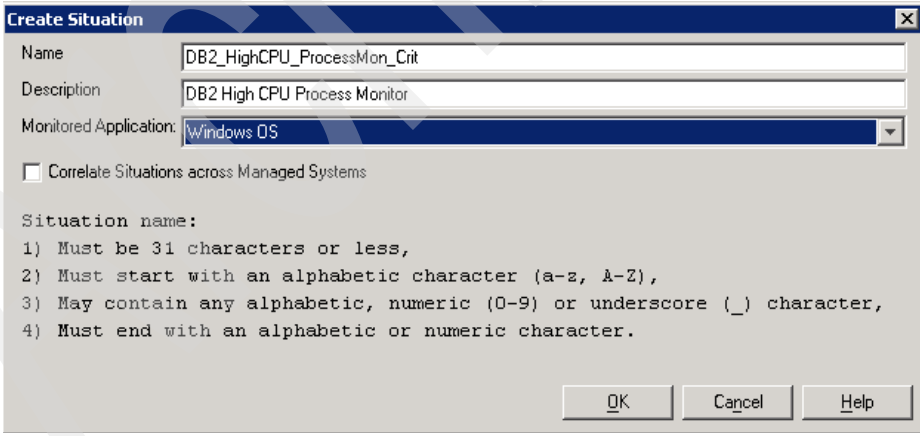


Figure 7-3 Create new DB2 situation

> **Note:** We used the Windows OS monitored application. Use the operating system running DB2 in your environment.

Click **OK** when done.

4. Select the conditions for this situation. We selected the **NT Process** attribute group and **% Processor Time** and **Process Name** attribute items, as shown in Figure 7-4.

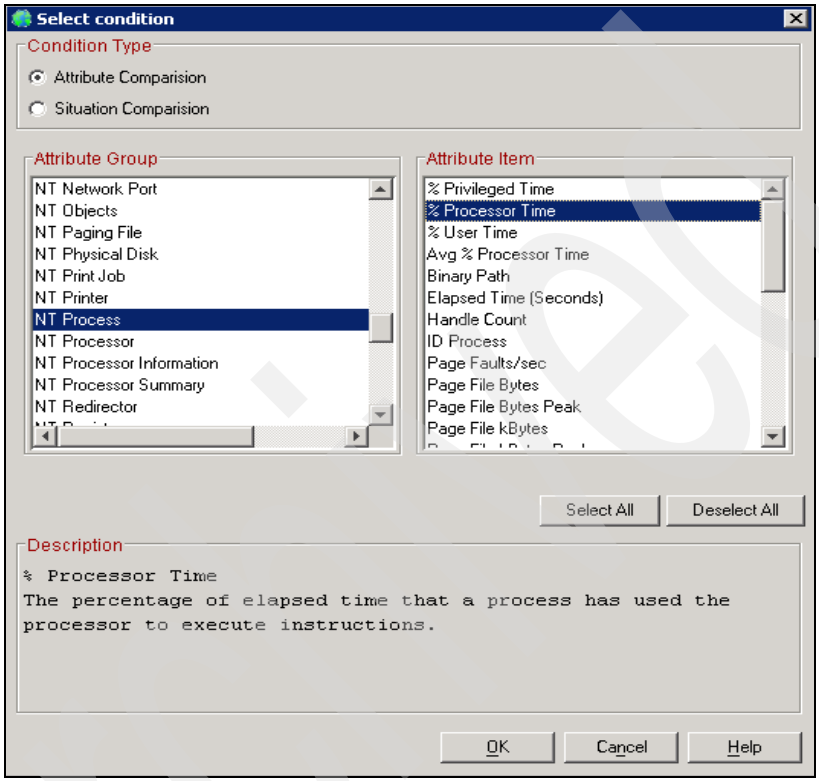


Figure 7-4 Situation condition

Click **OK** when done. You are now presented with the main situation editor panel.

5. Drag the **Process Name** attribute so that it is now the first column in the formula.

6. Configure the situation using Table 7-1 for a reference.

Table 7-1 Situation configuration data

Situation tab	Parameter	Example value
Formula	Process Name	= db2syscs.exe
	%Processor Time	> 65
	Sampling Interval	1 minute
	Run at Startup	Checked
Distribution	Assigned	Primary:CAIRO:NT
Expert Advice	Text or Advice Location	DB2 process is consuming high CPU time
EIF	Forward Events to EIF Receiver	Checked
	EIF Severity	Critical
	EIF Receivers	<Default EIF Receiver>

The DB2 high CPU situation should now be similar to ours, as shown in Figure 7-5.

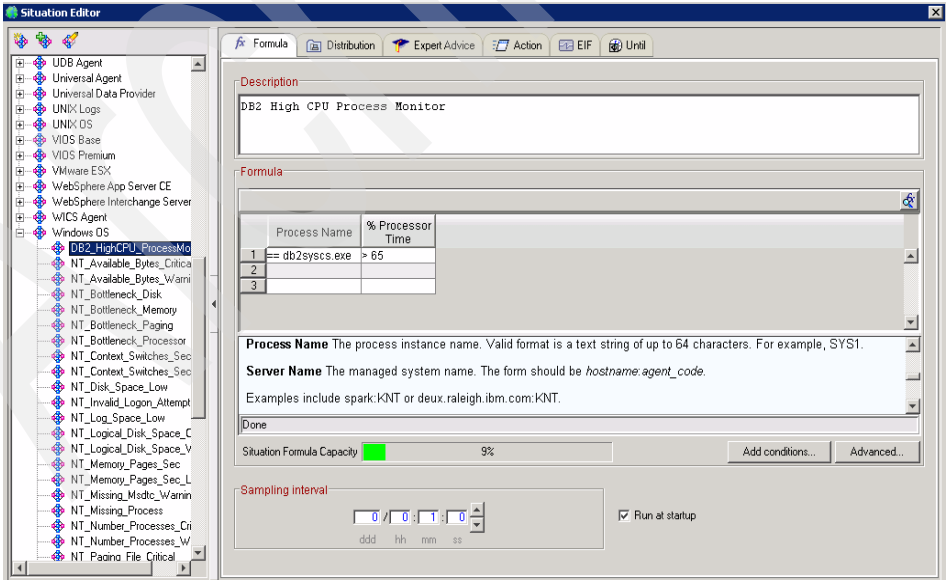


Figure 7-5 DB2 high CPU process situation

7. Click **Apply**, then **OK**.
8. Start the DB2 High CPU Process situation. To do this right-click the situation and select **Start Situation** to being monitoring this resource.

7.3.4 Configuring ITM for Tivoli EIF Probe

In this book we do not explain the installation of the Tivoli EIF Probe. Refer to the TBSM installation guide for the version that you are using for instructions on installing the probe. In this section we explain how to configure ITM for the Tivoli EIF Probe.

Refer to the latest *Tivoli Business Service Manager Installation Guide* at time of writing that explains the installation procedure:

http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.tivoli.itbsm.doc/installguide/bsmi_t_installing_eif_probe.html

Before you begin ensure that the Tivoli EIF Probe service is running and configured to listen on the default port 5530.

To configure ITM to send events to Netcool Omnibus via the Tivoli EIF Probe, as shown in Figure 7-6:

1. Run the Manage Tivoli Monitoring Services program. To do this for MS Windows, as in our case, run **Start** → **All Programs** → **IBM Tivoli Monitoring** → **Manage Tivoli Monitoring Services**.
2. Right-click over the Tivoli Enterprise Monitoring Server object and select **Reconfigure** from the list of menu options.
3. Check the **Tivoli Event Integration Facility** option, as shown in Figure 7-6.

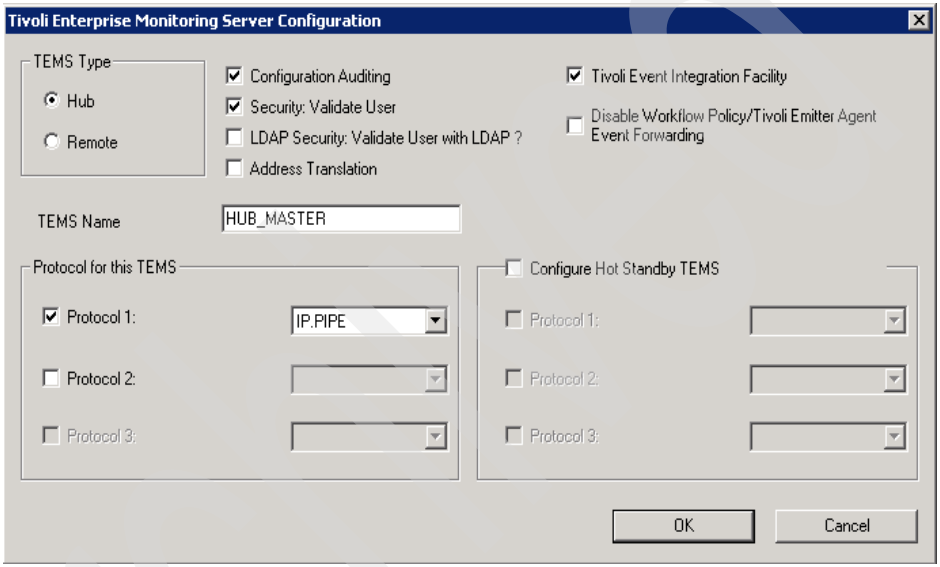


Figure 7-6 TEMS configuration

4. Click **OK** to continue.
5. In the Hub TEMS Configuration window, click **OK**.

6. You will now be presented with the Event Server window. Enter the host name of the TBSM/Netcool Omnibus server and the Tivoli EIF Probe listening port. The default port is 5530, as shown in Figure 7-7.

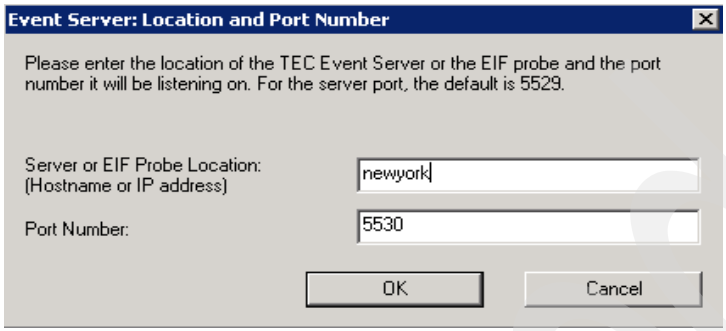


Figure 7-7 Event Server Configuration window

7. Click **OK** when done.
8. Restart the Tivoli Enterprise Monitoring Server to reflect the changes made. Reconfiguring the TEMS causes the service to stop.

7.3.5 Implementing Omnibus to HP ServiceCenter Integration

Before you begin this part of the scenario, ensure that the following components are running:

- DB2 High CPU Process situation
- ITM 6.2 TEMS/TEPS server
- Tivoli EIF Probe service
- TBSM or Netcool/Omnibus server

We received the event shown in Figure 7-8 in Netcool verifying that ITM sent the event to Omnibus via the Tivoli EIF Probe.

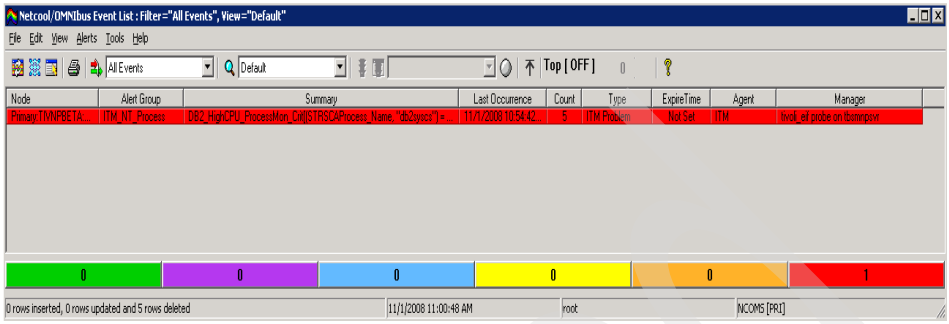


Figure 7-8 Omnibus event list

To complete the integration between Netcool/Omnibus and HP ServiceCenter for the purposes of creating an incident:

1. Verify that all software requirements have been met.
   - Netcool Omnibus version
   - HP ServiceCenter version
2. Create an operator named Omnibus in HP ServiceCenter.
3. Install and configure the Netcool/Omnibus HP ServiceCenter Gateway program on a supported platform. We used RedHat Linux in the ITSO laboratory for this purpose.
4. Create a gateway configuration G_PEREGRINE.conf file in the $OMNIHOME/etc directory.

Figure 7-9 shows a configuration file complete with gateway commands.

```
#
# Start up the reader - connect to the ObjectServer SCNCOMS
#
# This example does not use filtering
# START READER SERVER_READER CONNECT TO SCNCOMS;
#
# Start up the writer
#
START WRITER PEREGRINE_WRITER
(
TYPE = PEREGRINE,
REVISION = 1,
MODULE_REVISION= 4,
OPEN_MAP = PEREGRINE_OPEN_MAP,
UPDATE_MAP = PEREGRINE_UPDATE_MAP,
CLOSE_MAP = PEREGRINE_CLOSE_MAP,
JOURNAL_MAP= PEREGRINE_JOURNAL_MAP,
SCAUTO_SERVER = 'berlin.12690',
USER = 'Omnibus',
APPLICATION_PASSWORD = '',
FEEDBACK_SERVER = 'NCOMS',
FEEDBACK_FIELD = 'Location',
PROBNUM_FIELD_INDEX = 2,
SEPARATOR_CHAR = '^',
CLOSE_PROBLEMS = TRUE,
DATE_FORMAT = '%m/%d/%y %T',
COUNTERPART = SERVER_READER,
CONVERSIONS_TABLE = 'conversions.peregrine',
OPEN_ACTION_SQL = '/opt/Omnibus/gates/peregrine/open_action.peregrine.sql',
UPDATE_ACTION_SQL = '/opt/Omnibus/gates/peregrine/update_action.peregrine.sql',
CLOSE_ACTION_SQL = '/opt/Omnibus/gates/peregrine/close_action.peregrine.sql'
# OPEN_ACTION_SQL = 'C:\\Program Files\\Netcool\\OMNIbus\\gates\\Peregrine\\ open_action.peregrine.sql',
# UPDATE_ACTION_SQL = 'C:\\Program Files\\Netcool\\OMNIbus\\gates\\Peregrine\\ update_action.peregrine.sql',
# CLOSE_ACTION_SQL = 'C:\\Program Files\\Netcool\\OMNIbus\\gates\\Peregrine\\ close_action.peregrine.sql',
# CURR_SYS_SEQ_TABLE = 'custom.sys_seq',
# CURR_SYS_SEQ_NAME = 'peregrine',
# RW_MODULE_PING = TRUE );
#
# Add a route from the reader to the writer so the data is passed
#
# ADD ROUTE FROM SERVER_READER TO PEREGRINE_WRITER
#
# End of file
#
```

Figure 7-9 HP ServiceCenter Gateway configuration file

5. Create a gateway conversion table according to your needs.
6. Edit gateway action SQL files according to your needs.
7. Register HP ServiceCenter event types.
8. Create HP ServiceCenter mappings.
9. Configure HP ServiceCenter format control subroutines.

Refer to the following *Netcool/Omnibus Gateway HP Openview ServiceCenter Reference Guide* for an in-depth overview of the installation procedure and integration:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.netcool_OMNIbus.doc/hpscgw-pdf.pdf

7.4 Implementing the CCMDB and HP ServiceCenter integration

In this section we explain how we implemented the CCMDB and HP ServiceCenter integration in our laboratory environment. For a detailed explanation of the technical concepts behind this, refer to the *Integration Guide for IBM Tivoli Service Request Manager V7.1*, SG24-7580.

7.4.1 Integration environment

We have three servers in our integration environment (Figure 7-10 on page 198):

- Server A is a Linux-based server running the CCMDB and all middleware applications.
- Server B is an MS Windows-based server running the HP ServiceCenter server and client applications. HP Service Center Web Services also run on this server.
- Server C is a MS Windows-based server to run the TDI Config Editor.

Note: We run the TDI assembly lines interactively through the TDI Config Editor for the ease of control and problem determination. TDI provides the mechanism to run the TDI assembly lines as batch and auto-started tasks easily.

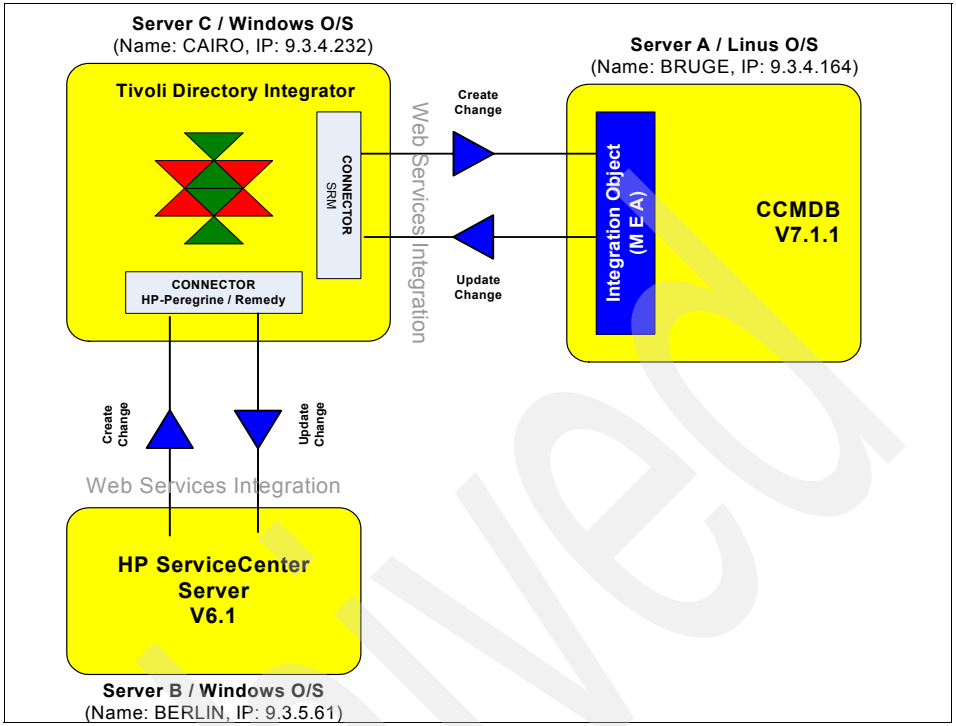


Figure 7-10 CCMDB and HP ServiceCenter Integration environment

7.4.2 Integration solution

TDI installation files and the HP ServiceCenter integration solution (in the form of a zip file) are shipped with the Tivoli Service Request Manager V7.1. If you have the CCMDB installation CD only, you can look for the integration solution on the IBM Tivoli Open Process Automation Library (OPAL) Web site at:

`http://catalog.lotus.com/wps/portal/topal`

The integration solution is not part of CCMDB or Tivoli Service Request Manager. The integration solution is just an example, with limited features and data fields, to demonstrate the capability of TDI in integrating CCMDB and HP ServiceCenter. Enhancement of the sample solution to support a production rollout would normally require services from IBM unless you have deep knowledge on CCMDB, HP ServiceCenter, and TDI.

7.4.3 Implementation steps

To do to implement the integration solution:

1. Read the installation documentation that comes with the integration solution.
2. Check whether the prerequisites are met.
3. Acquire the information required on the planning worksheet.
4. Install the Service Desk integration solution on CCMDB.
5. Configure the HP ServiceCenter.
6. Configure the Tivoli Directory Integrator.
7. Configure the application designer in CCMDB (optional).
8. Read the documentation to understand the assembly lines included in the integration solution.
9. Run the assembly lines.

In the following sections we provide more details about steps 4 to 7.

Install the Service Desk integration solution on CCMDB

In this step we change the CCMDB database and application. We recommend doing a backup of your middleware servers and the CCMDB administration workstation (a MS Windows-based workstation) before you start this step. Perform the tasks listed in Table 7-2.

Table 7-2 Tasks to perform

No.	Tasks	Notes
1	Install the Maximo Enterprise Adapter (MEA) and any other process solution included in the integration solution.	If you have installed Tivoli Service Request Manager, the MEA will be installed already. We recommend using the Process Solution Installation wizard on the CCMDB administration workstation to install the MEA and process solution because the wizard will perform validation before installation and it automates some steps in the installation. But if you are not using the CCMDB administration workstation for installation, you can still use the installation command-line interface (CLI).

No.	Tasks	Notes
2	Change the system properties.	You will update the system properties through the Web user interface. After you log on as Maximo administrator, click **Go To** → **System Configuration** → **Platform Configuration** → **System Properties**. You will need to check or update the global value of the global properties. After your update, click **Save**, then **Live Refresh**. Refer to Figure 5-3 on page 121.
3	Generate XML schemas.	You must navigate to the object structures application by clicking **Go To** → **Integration** → **Object Structures**, then select the **Generate Schema/View HTML** action. Refer to Figure 7-12 on page 201.
4	Update launch-in-context entries.	Updating launch-in-context (LIC) entries is an optional task. It is only needed if you want to launch HP ServiceCenter in the CCMDB application. Also, after you have created the LIC entries, you must use the application designer application to change the user interface of the CCMDB application to use the new LIC entries.
5	Verify configuration.	Make sure that you have verified the configuration before you proceed. It is difficult to perform problem diagnosis if you have not done the installation correctly.

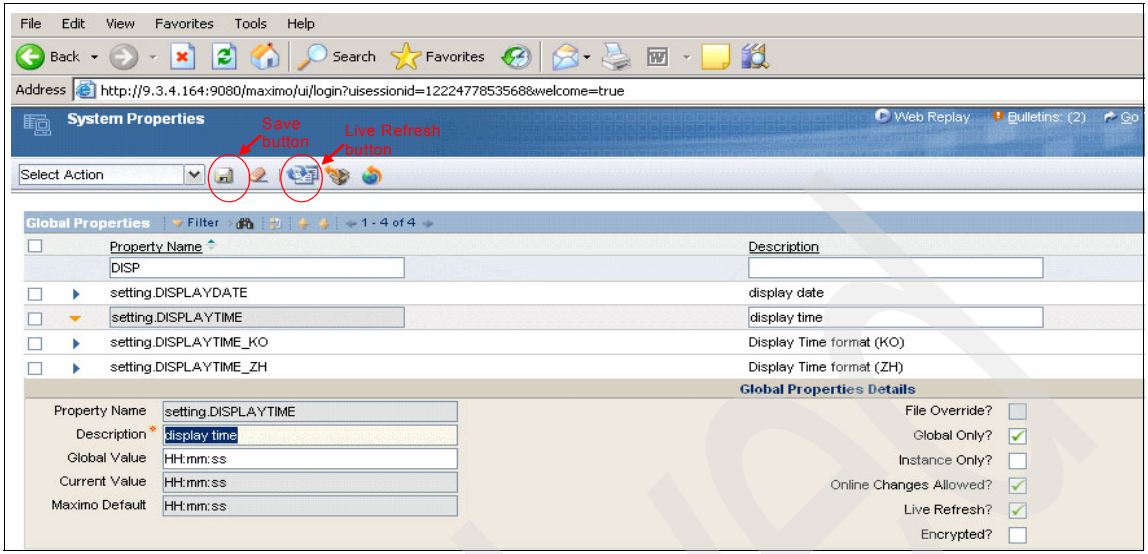


Figure 7-11 System Properties

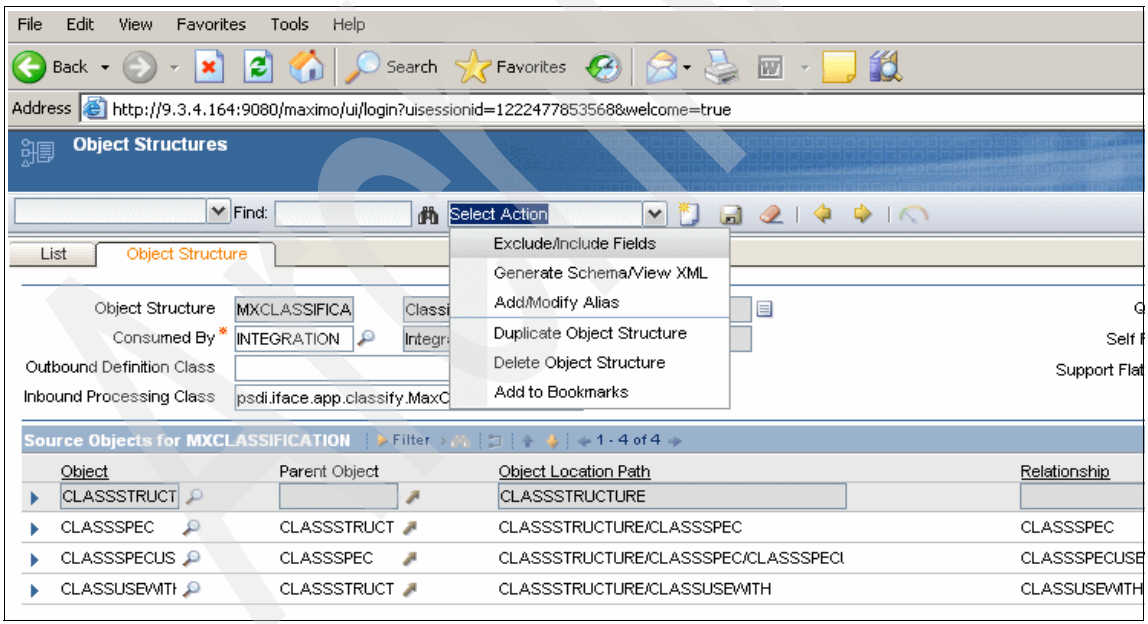


Figure 7-12 Object Structures

Configure the HP ServiceCenter

In this step perform the following tasks:

1. Set up the Web services server.
2. Create new fields.
3. Add Web services API database attributes.
4. Implement launch-in-context (optional).
5. Verify the creation and publishing of new fields.

After you have done the above tasks, you can view the WSDL files by typing the following text in browser:

```
http://<PEREGRINE_SERVER_HOSTNAME>:<PEREGRINE_SERVER_PORT>/<WSDL_FILENAME>
```

For example:

```
http://9.3.5.61:12670/IncidentManagement.wsdl
```

Configure the Tivoli Directory Integrator

In this step perform the following tasks:

1. Install the Maximo connectors.
2. Install the Peregrine connectors.
3. Install the assembly lines.
4. Configure the mxe.properties file.

Note: The integration solution uses a common properties file to store the default properties. You can use the TDI config editor to update the properties file:

1. Start the TDI config editor.
2. Open a TDI assembly line.
   a. Select **File** → **Open**.
   b. Chose an assembly line file, for example, C:\TDI\work\mxe.xml.
   c. Click **Open**.
3. Set the properties.
   a. In the navigation menu on the left hand side, click **Properties**.
   b. Select **MXE** from the Property Stores list (Figure 7-13 on page 203).
   c. Edit the properties value by double-clicking the Value field for each of the properties.
4. Select **File** → **Save**.
5. Select **File** → **Quit**.

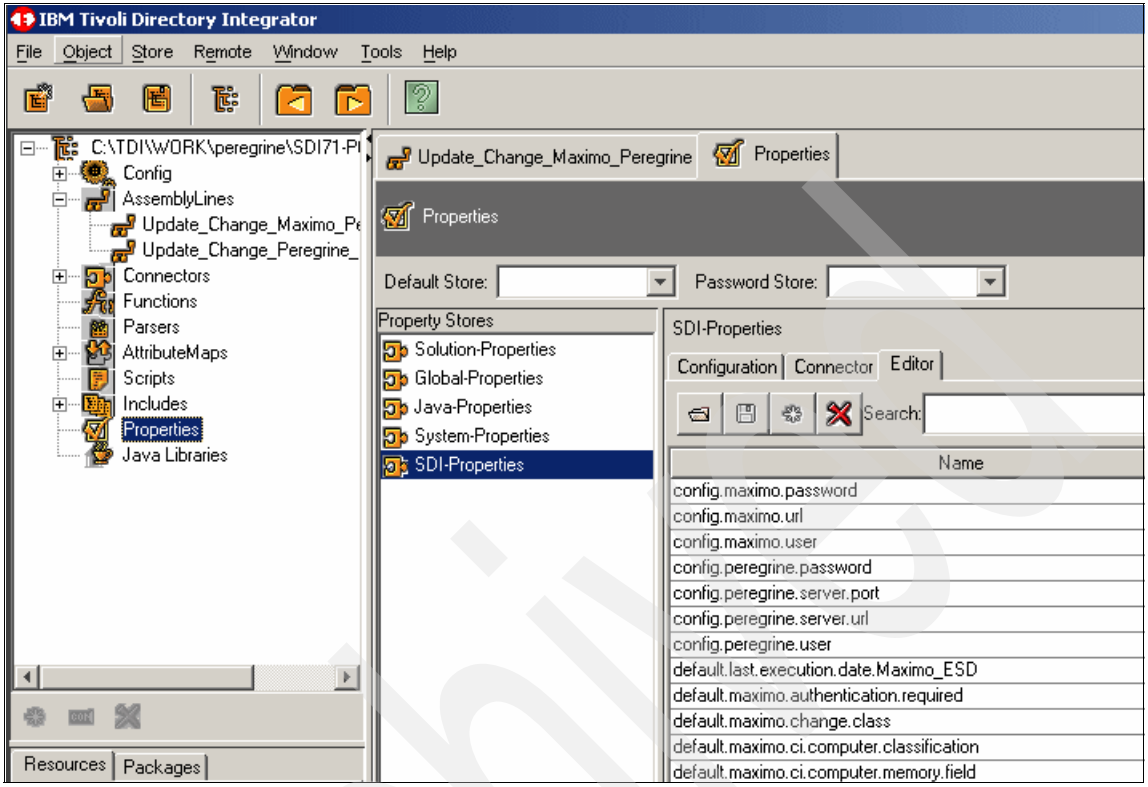


Figure 7-13 TDI property stores

Table 7-3 MXE Properties store in TDI

Property	Default value	Properties as per integration environment	Record your values
config.peregrine.password		Provide HP ServiceCenter user password for authentication.	
config.peregrine.server.port	12700	Provide HP Service Desk port number.	
config.peregrine.server.url	http://<server IP>	Provide Peregrine URL, for example, http://9.3.5.56.	
config.peregrine.user, value	falcon	Provide HP Service Desk user name for authentication.	
default.last.execution.date.Maximo_ESD	2008-01-01T06:00:00+00:00	Auto populated.	

Property	Default value	Properties as per integration environment	Record your values
default.maximo. authentication. required	TRUE	DEFAULT.	
default.maximo. callSystemId, value	PCALL	DEFAULT.	
default.maximo.error.e xcedent.size	FALSE	DEFAULT.	
default.maximo .incidentSystemId	PINC	Provide value Maximo Incident System ID.	
default.maximo.owner	EXT_SD	Provide value of owner ID, which should be a valid ID in person records.	
default.maximo.owner. group	G_EXT_SD	Provide value of owner group ID, which should be a valid ID in person groups records.	
default.maximo .password		Provide user password for authentication.	
default.maximo. problemSystemId	PPRO	DEFAULT.	
default.maximo.siteid	BEDFORD	Provide site ID for the tickets.	
default.maximo.url	http://<Server name or ID>:<por number>	Provide CCMDB application URL with port number, for example, http://9.3.4.164:9080.	
default.maximo.user	maxadmin	Use maxadmin as default or provide an integration user, which would be used for the activity.	
default.maximo.xml. character.validation	TRUE	DEFAULT.	
default.peregrine. assignment.group	DEFAULT	DEFAULT.	
default.peregrine. category	other	DEFAULT.	

Property	Default value	Properties as per integration environment	Record your values
default.peregrine. contact	FALCON, JENNIFER	Provide Peregrine user for the contact.	
default.peregrine. incident.assignment. group	DEFAULT	Provide Peregrine incident assignment group.	
default.peregrine. incident.category	other	DEFAULT.	
default.peregrine. incident.closure.code	Advice & Guidance	Provide Peregrine incident closure code.	
default.peregrine. incident.contact	FALCON, JENNIFER	Provide Peregrine incident contact name.	
default.peregrine. incident.problem.type	None	DEFAULT.	
default.peregrine. incident.product.type	None	DEFAULT.	
default.peregrine .incident.resolution.fix. type	Permanent	DEFAULT.	
default.peregrine. incident.severity	5 - Very Low	Provide Peregrine incident severity.	
default.peregrine. incident.site.category	Remote	Provide Peregrine incident category.	
default.peregrine. incident.subcategory	Client dependent	Provide Peregrine incident subcategory.	
default.peregrine.owner	BOB.HELPDESK	Provide Peregrine process owner, which is a valid Peregrine user.	
default.peregrine. problem.category	Other	Provide Peregrine problem category.	
default.peregrine. problem.initial.impact	Low	Provide Peregrine problem initial impact.	
default.peregrine. problem.problem.type	None	Provide Peregrine problem type.	

Property	Default value	Properties as per integration environment	Record your values
default.peregrine. problem.product.type	None	Provide Peregrine problem product type.	
default.peregrine. problem.subcategory	Client dependent	Provide Peregrine problem subcategory.	
default.peregrine. problem.type	None	Provide Peregrine problem type.	
default.peregrine. problem.urgency	4 - Low	Provide Peregrine problem urgency.	
default.peregrine. severity	5 - Very Low	Provide Peregrine problem severity.	
default.peregrine.site. category	Remote	Provide Peregrine problem site category.	
default.peregrine. subcategory	Client dependent	Provide Peregrine problem subcategory.	
default.problem. urgency	4 - Low	Provide Peregrine problem severity.	
delete.last.execution. date.Maximo_ESD	no	DEFAULT.	
maximumQueueSend Attempts	10	DEFAULT.	
queueStoreINIFile	pwstore_server.ini	DEFAULT.	

Configure the application designer

This step is optional. In our integration environment, we have modified the WOCHANGE object in the change application to add the display of the EXTERNALREFID attribute because the CCMDB will generate a new change number for the change created by TDI. The EXTERNALREFID contains the original change number in the HP ServiceCenter. Therefore, it is easier to cross-reference the change numbers in both system.

To configure the application designer:

1. Log into CCMDB as a user with administrator permissions.
2. Click **Go to** → **System Configuration** → **Platform Configuration** → **Application Designer**.
3. Select the application, for example, Change.

4. For safety's sake, save the application definition now by clicking the Export Application Definition icon. The XML of the application is displayed in a browser. Click **File** → **Save As** to save the XML in a safe location.
5. Select the tab to add the display, for example, Change.
6. Click the Control Palette icon.
7. Drag and drop a textbox to a location where you want to display the textbox. (What you see is what you get.)
8. Configure the textbox properties by right-clicking and selecting **Properties**.
9. Input the textbox properties. We input EXTERNALREFID for the attribute and External Reference ID in for the default label.
10. Click the Save icon after exiting the text properties.

Refer to Figure 7-14.

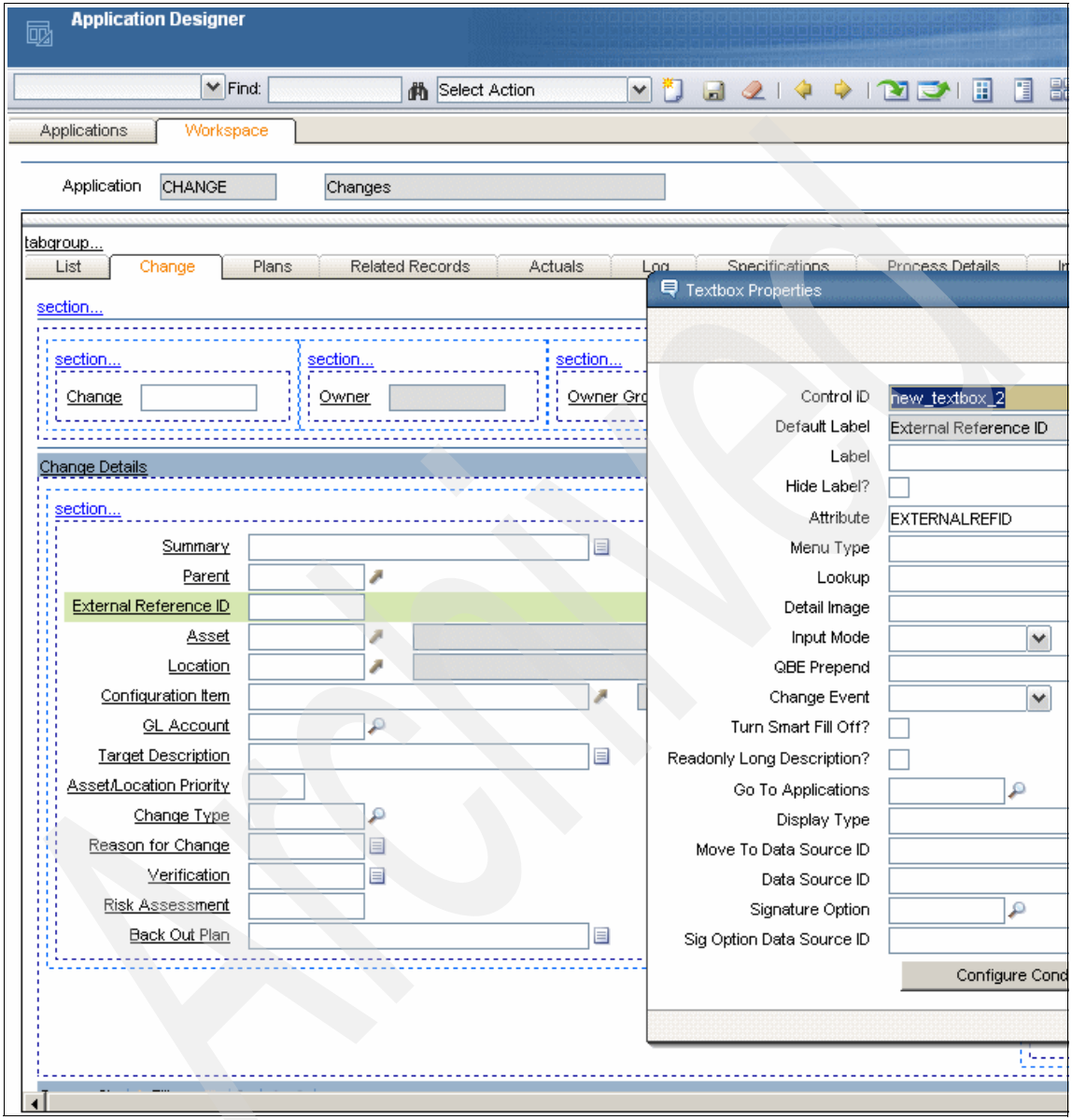


Figure 7-14 Application designer

Run the assembly line

To run an assembly line manually:

1. Start the TDI config editor.
2. Open the assembly line file.
3. Right-click the assembly line and select **Run** to run an assembly line.
4. After assembly line runs successfully you go on to CCMDB to see the change record propagated from HP ServiceCenter to CCMDB. Refer to Figure 7-15 to see a change record propagated from HP ServiceCenter.

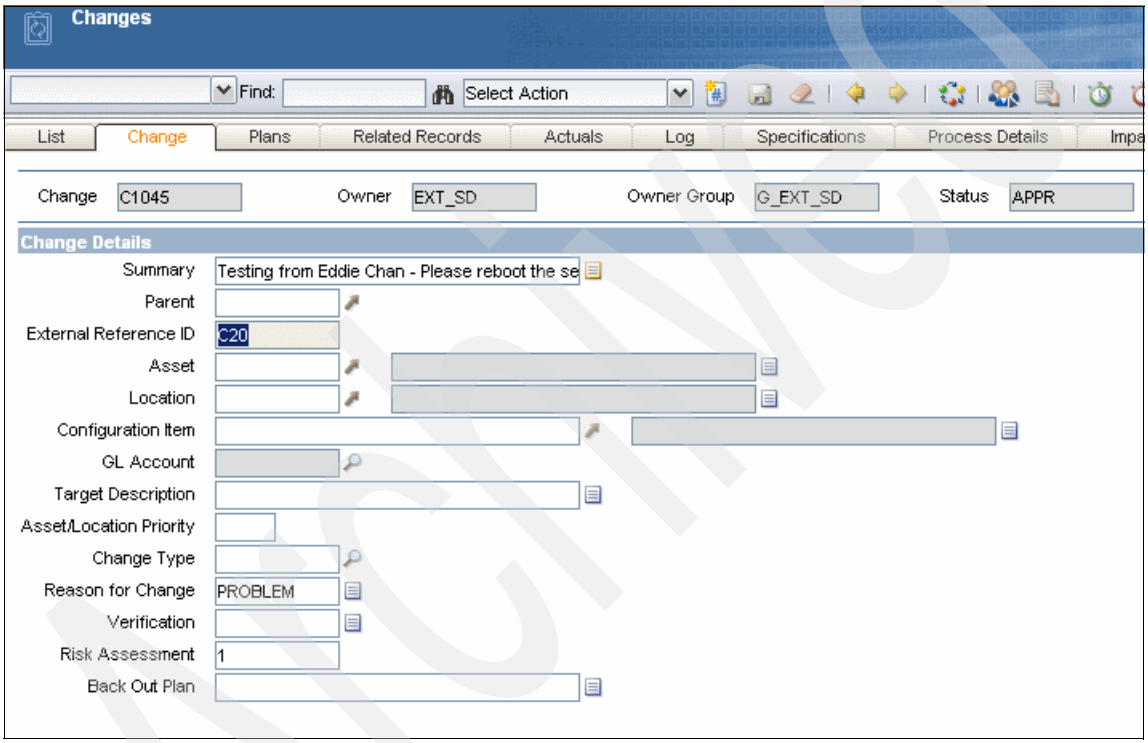


Figure 7-15 Change record in CCMDB

8

Scenario 4: IT Service Continuity Management using IBM Tivoli Business Continuity Process Manager

Tivoli Business Continuity Process Manager provides configurable processes to plan, test, and execute IT Service Continuity, leveraging the infrastructure provided by the Change and Configuration Management Database (CCMDB). In this chapter we present a scenario that involves creating a recovery and test plans and show you how IBM Tivoli Business Continuity Process Manager (TBCPM) can be used in a real-life outage situation to restore the services.

This chapter contains the following sections:

8.1 Scenario introduction

In this scenario we show you how to create a recovery plan to capture information about your IT infrastructure, the people, processes, and plans that you have in place. Then we use a test plan to simulate an outage situation. Finally, we walk through the workflow that IBM Tivoli Business Continuity Process Manager provides to see how outage situations are handled using TBCPM.

We assume that you have already collected the information described by Information Technology Infrastructure Library (ITIL) for IT Service Continuity planning. This information is needed to implement TBCPM, adapt the workflow, and create a recovery plan.

8.2 Scenario implementation

The person in charge of creating recovery plans is the IT Service Continuity (ITSC) manager, which is in line with the responsibilities that ITUP recommends for supervising the continuity process. To implement:

1. Log into CCMDB with a user account that has the ITSC manager role assigned (Figure 8-1).

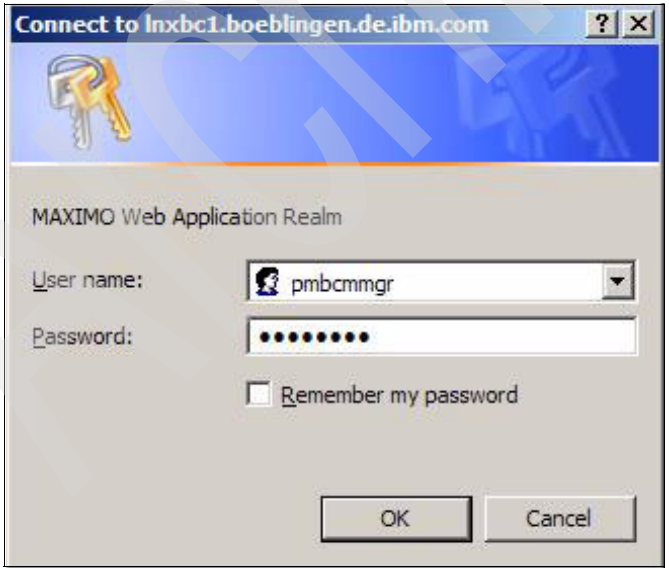


Figure 8-1 Login panel

2. Figure 8-2 shows the custom Start Center for an ITSC Manager. Among other things, it shows links to the applications provided by TBCPM:
   - The recovery plan application
   - The test plan application
   - The work order application

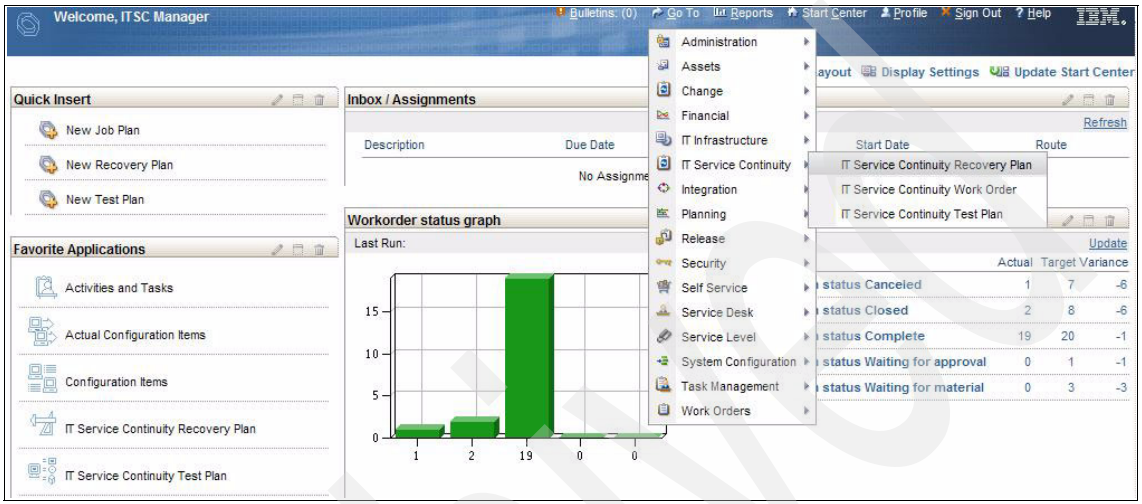


Figure 8-2 Custom Start Center for an ITSC Manager

3. Navigate to the recovery plan application and create a new recovery plan (Figure 8-3).

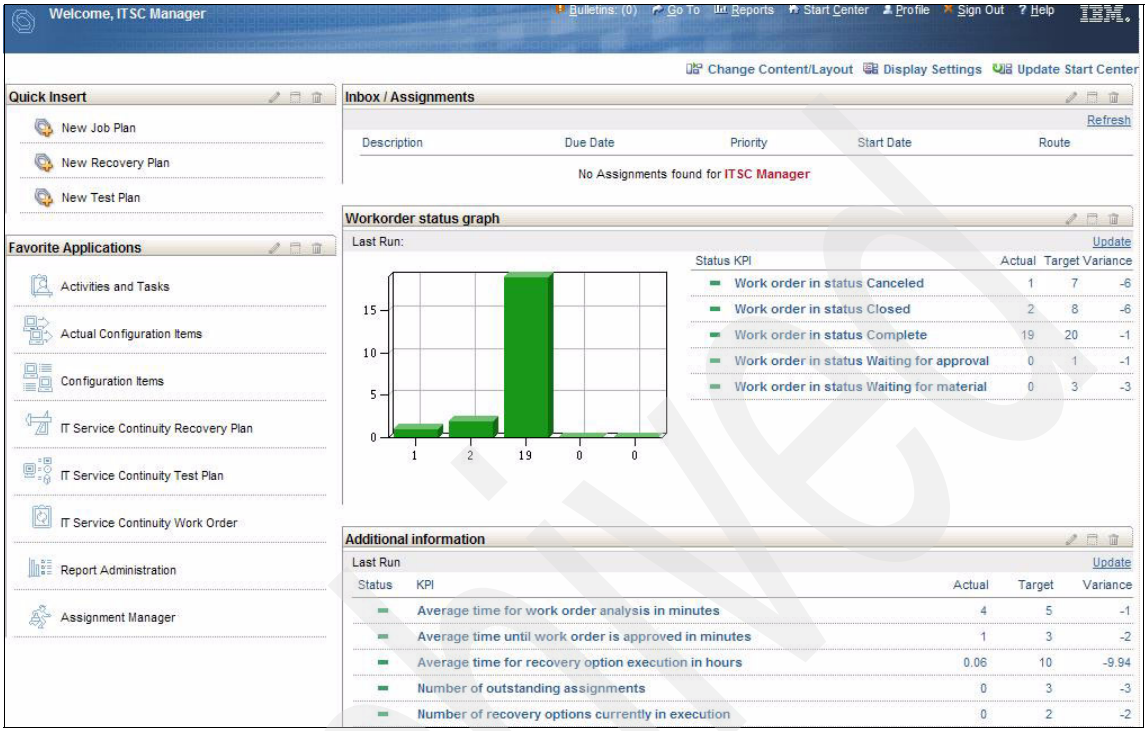


Figure 8-3 Recovery plan application

8.2.1 Create a recovery plan with auto-approval

In this step we create a recovery plan with auto-approval.

1. Specify basic details for the recovery plan. We name it DM_REC_S, give it a description, and specify the owner of the plan.

2. When this recovery plan is selected to provide the recovery procedure in a real outage situation, it might be necessary to get the decision explicitly approved. You can do this by specifying an approver role that gets the task assignment before the actual recovery procedure can be started. For this scenario we skip this feature, so we do not need to specify an approver role. Instead, we enable auto approval, as shown in Figure 8-4.

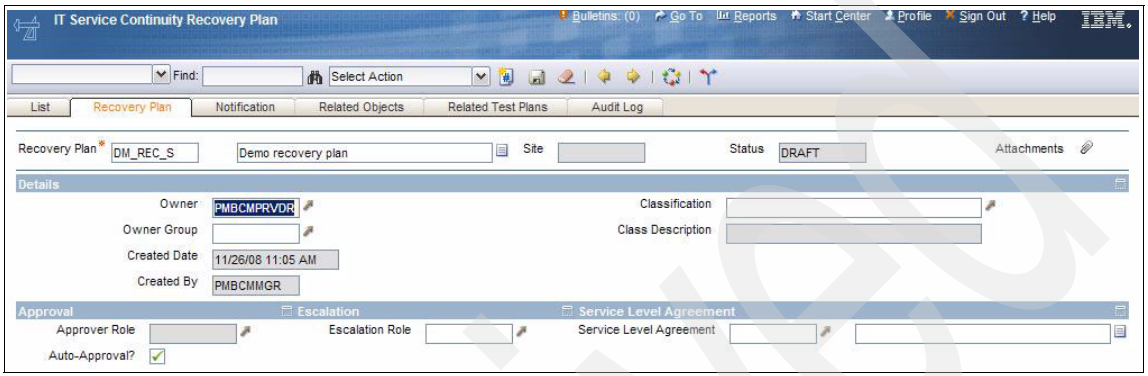


Figure 8-4 DM_REC_S recovery plan

3. To speed up the process of finding the correct recovery plan in an outage situation, the recovery plan can be associated with various objects such as configuration items that represent business systems, disks, hardware boxes, and so on. It can also be associated with specific locations or sites, as shown in Figure 8-5.

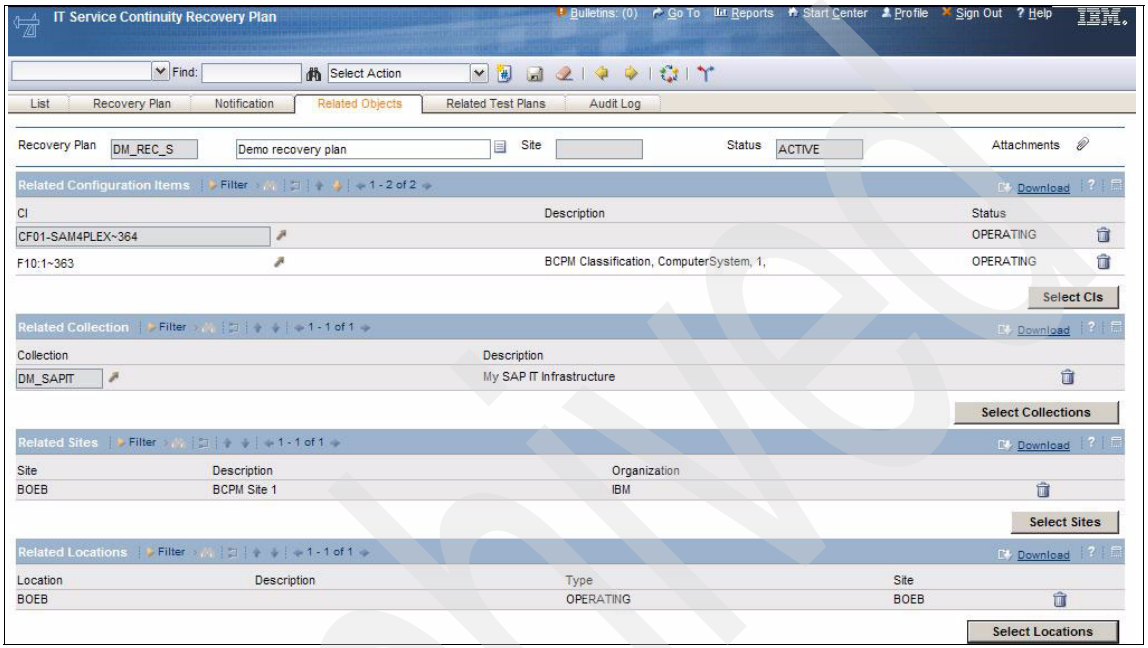


Figure 8-5 Recovery plan can be associated to various objects

4. A recovery plan can be classified according to its intended use, for example, for disk outages, or for system outages, and so on (Figure 8-6 and Figure 8-7 on page 217).

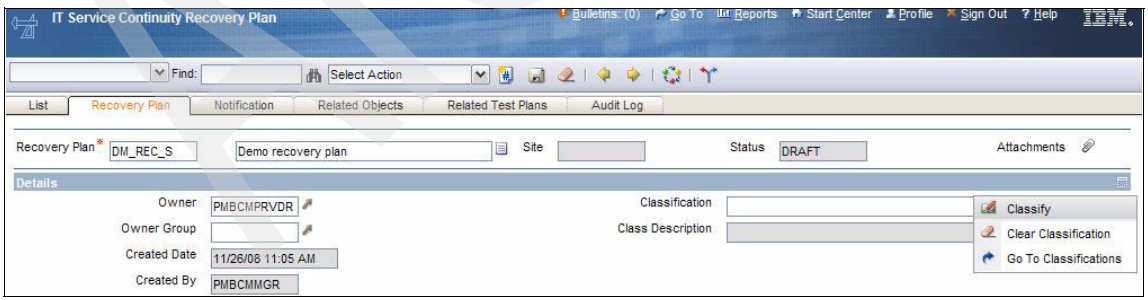


Figure 8-6 Selecting Classify

Figure 8-7 Classification

5. You can also specify recovery options that clarify what is actually done when a disaster has been declared, as shown in Figure 8-8. Recovery options can take the form of simple actions, complete workflows, or job plans. When using a recovery plan in a particular situation, the set of options available is the super-set of the options provided by the recovery plan in addition to the actions that are supplied by the Operational Management product in the outage event message. The recovery plan provides the execution context for all of these options.

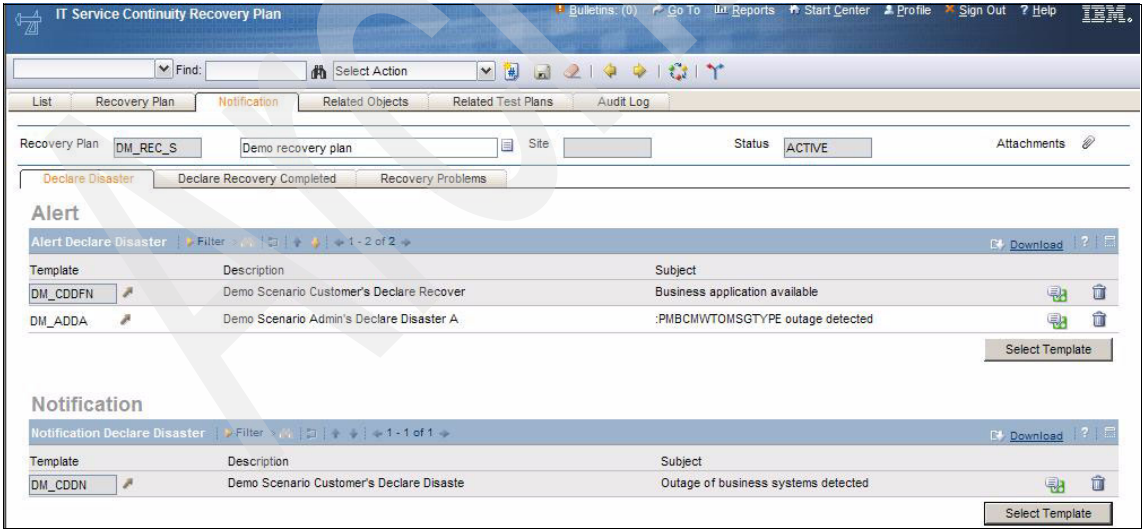


Figure 8-8 Specify recovery options

6. You may also want to specify communication templates for the recovery plan. The communication templates describe who is notified with what information at key moments such as when a disaster is declared. See Figure 8-9.

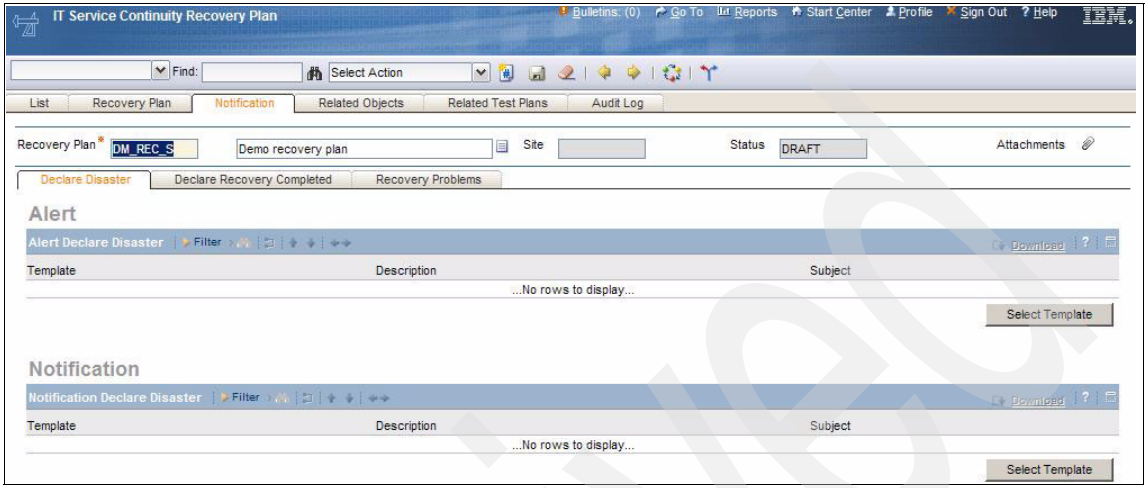


Figure 8-9 Specify communication templates for the recovery plan

7. Finally, activate and save the recovery plan, as shown in Figure 8-10.

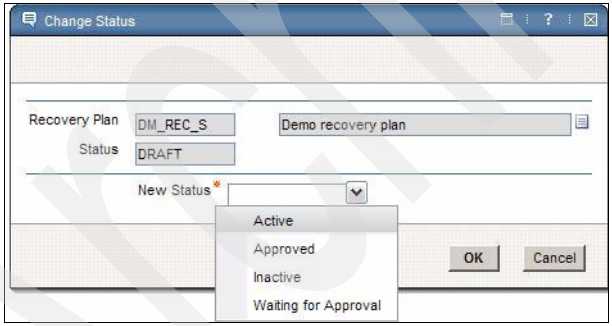


Figure 8-10 Activate and save the recovery plan

Figure 8-11 shows that the recovery plan is active.

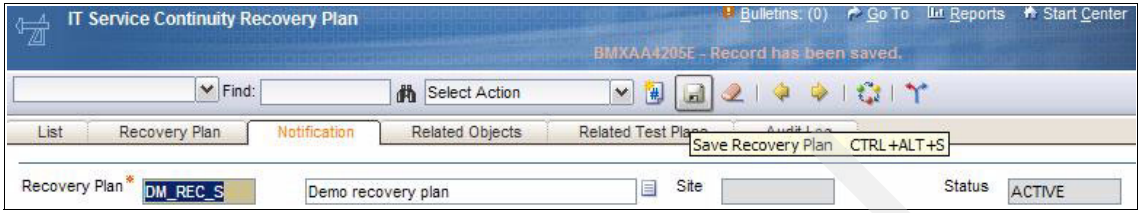


Figure 8-11 Recovery plan active

8.2.2 Simulate an outage scenario

When you have successfully created one or more recovery plans, you are ready to handle outage situations. However, to test and verify that the plans are correctly implemented and that the recovery procedures work properly, you may want to simulate events before real events come in. To do so:

1. Navigate to the test plan application to simulate an outage message (Figure 8-12).

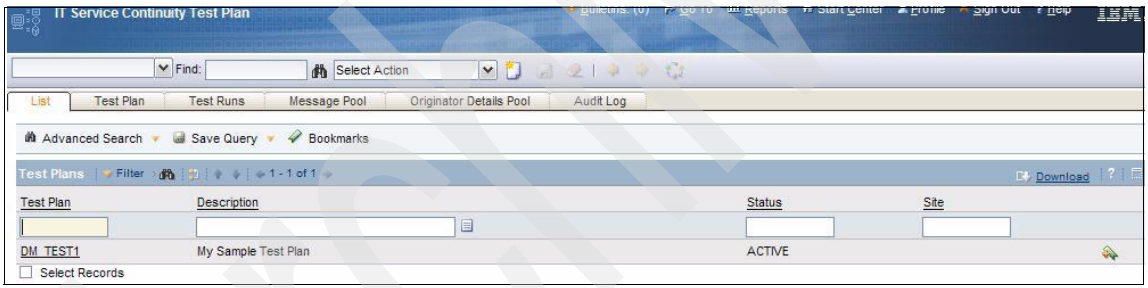


Figure 8-12 Test plan application

TBCPM ships with a set of demo data that can be used for training and demonstration purposes. Included in this demo data is a test plan named DM_TEST1. We used this test plan for our scenario (Figure 8-13).

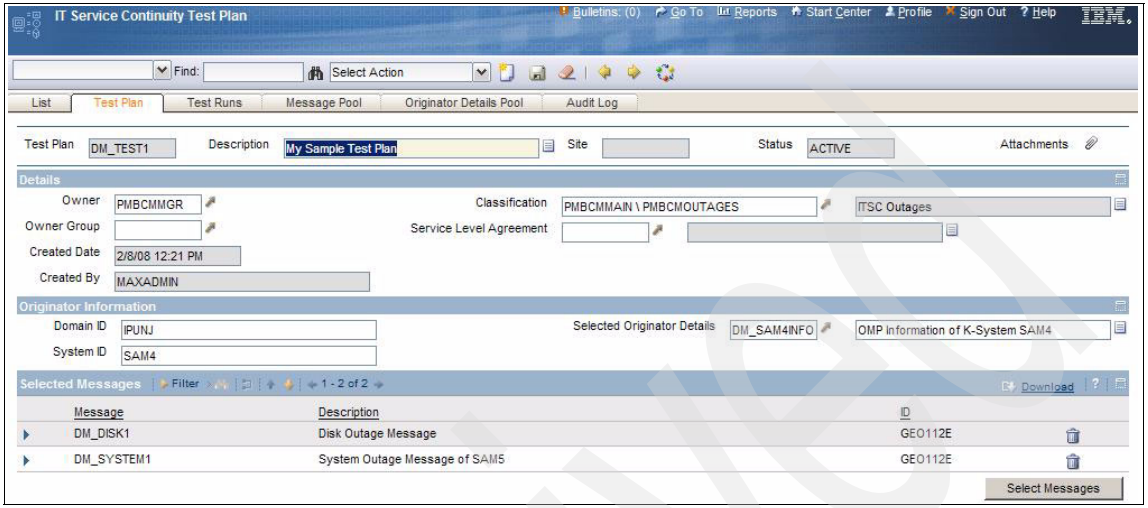


Figure 8-13 Select test plan

2. Also, several samples of possible trigger event messages have already been captured. You see all available messages on the Message pool tab. We use one of these messages—the system outage message DM_SYSTEM1—to trigger the TBCPM workflow, as shown in Figure 8-14.

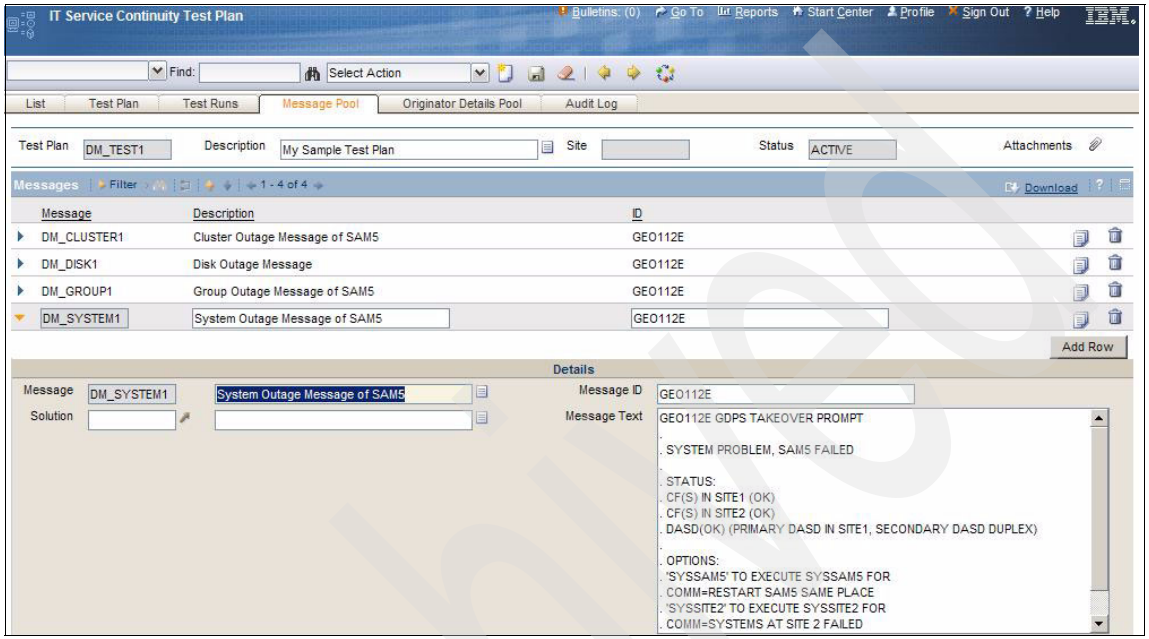


Figure 8-14 Trigger event messages

3. To run a test, a new test instance, DM_MYTEST, is created (Figure 8-15). In the test instance, you can specify the level of simulation. In this demo, all activities are going to be simulated, so none of the available check boxes should be selected.

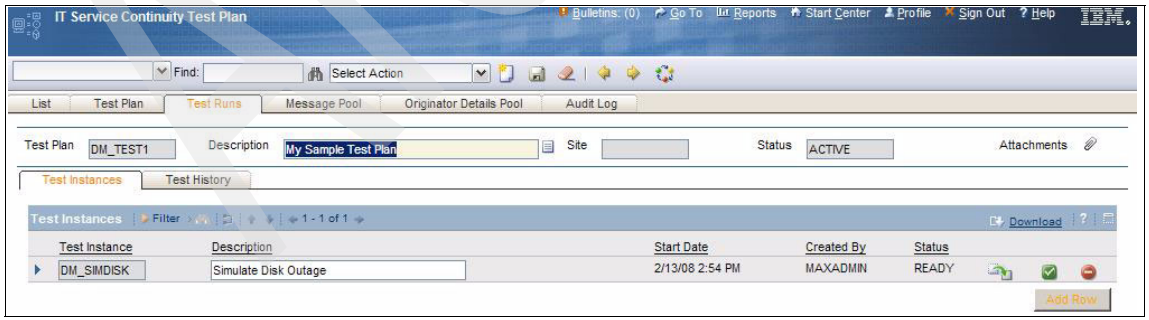


Figure 8-15 New test instance DM_MYTEST is created

4. To trigger a message, click the **Send Message** button and select one of the available messages, as shown in Figure 8-16. The message is sent to TBCPM as though it had been sent by the real infrastructure.

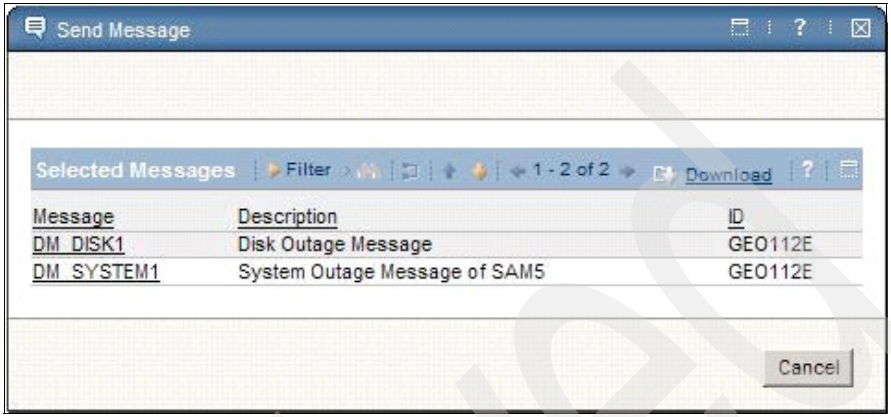


Figure 8-16 Send Message window

Important: Although TBCPM in its first release implemented out-of-the-box capabilities to support Geographically Dispersed Parallel Sysplex (GDPS), TBCPM is not restricted to nor requires GDPS. This scenario shows how a GDPS GEO112E message triggers the TBCPM workflow. However, the scenario could easily be adapted to any other system event or underlying software.

As soon as TBCPM receives the message, a work order is created, then the TBCPM workflow is initiated, and then the incoming message is automatically parsed and evaluated. As part of the evaluation process, information is retrieved from the message and is stored in the work order for later use. Information retrieved in this way includes the symptom (such as system, disk, cluster outage, and so on), the failing configuration item, if known, and the recovery options provided by GDPS in the GEO112E message.

Based on this information, some actions are performed automatically by the TBCPM master workflow. For example, notifications are sent out to the crisis management team, and the task is assigned to the correct subject matter expert based on the information retrieved from the actual message.

Based on this automatic analysis, a subsequent manual analysis step is initiated.

8.2.3 Analyze the outage

As mentioned, the ITSC analyst will have received an e-mail requesting an outage analysis task to be performed.

1. To see what this involves, let us now log in using the pmbcmanlst account.

   Figure 8-17 shows the predefined Start Center for the ITSC analyst role. As you can see, the ITSC analyst has a task assignment in the inbox. The task assignment is pre-filled with information that has been retrieved from the incoming message. The details in the task assignment show that a system outage triggered the workflow.

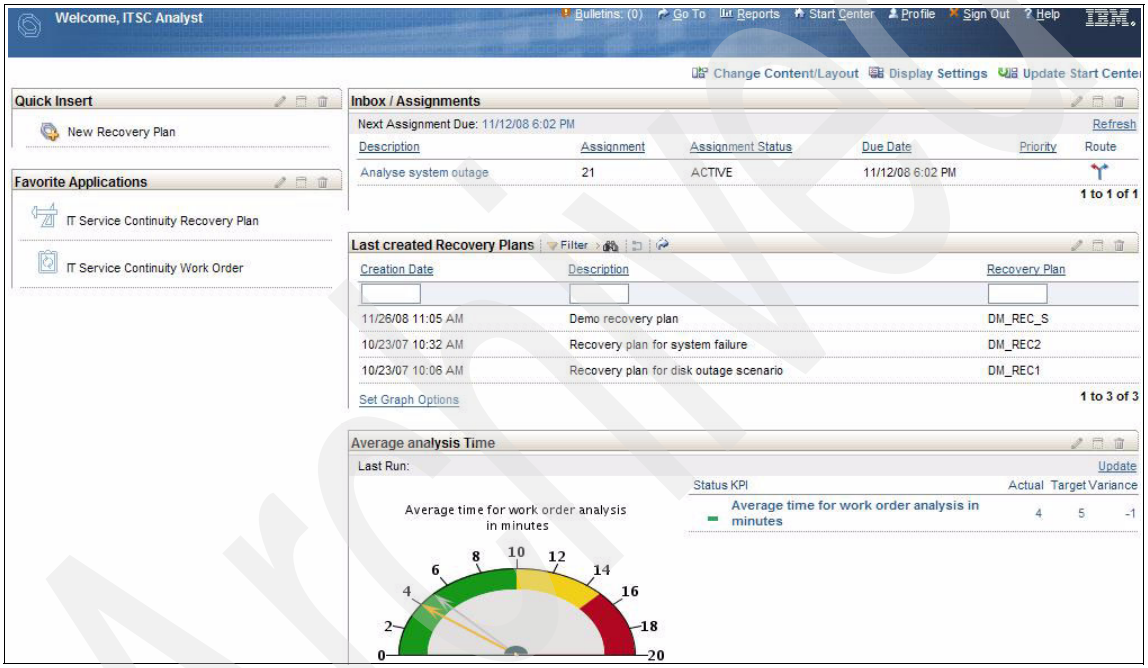


Figure 8-17 ITSC Analyst Start Center

2. Let us take a closer look at the work order and do some outage analysis. The TBCPM work order application (Figure 8-18) provides specific views for the different roles that work with it. In the current case, the analyst has access to all information and tools necessary to determine what is impacted, what the objectives and policies are, whether the outage represents a crisis, and so on.

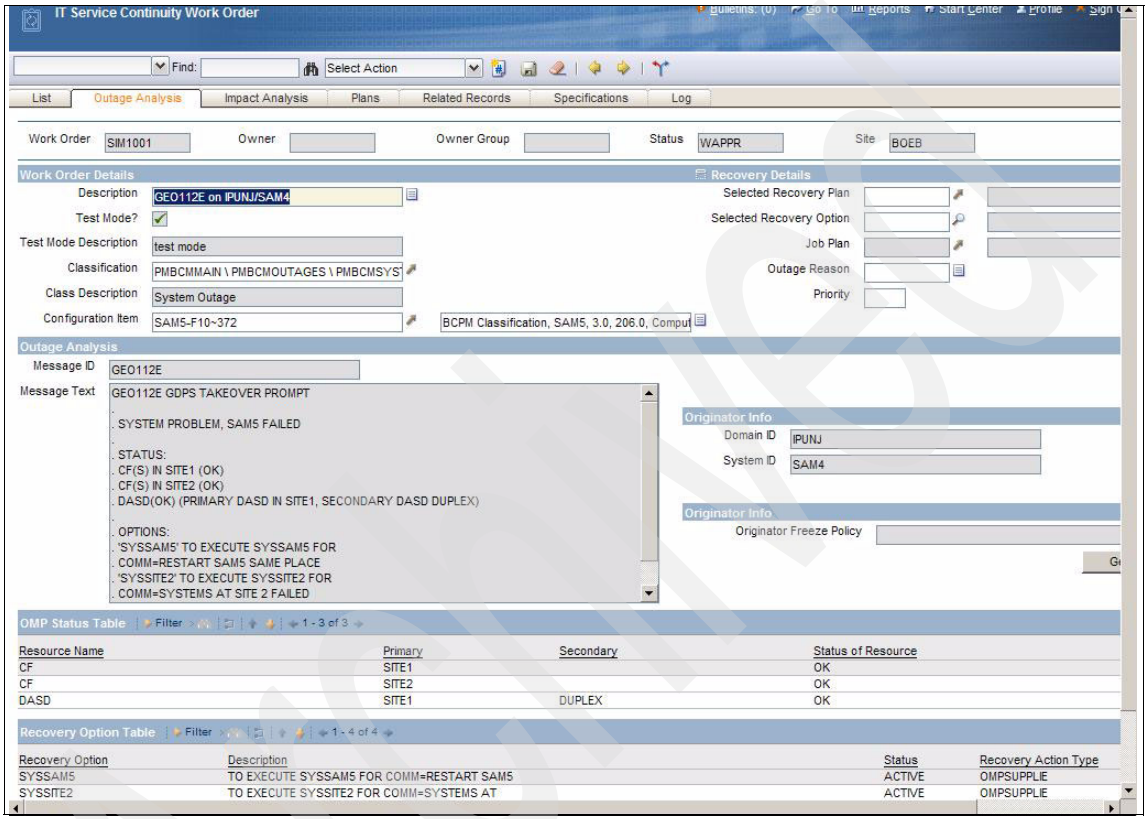


Figure 8-18 Work order application

3. The integrated tools are available via a group of launch commands on the Select Action menu, as shown in Figure 8-19.

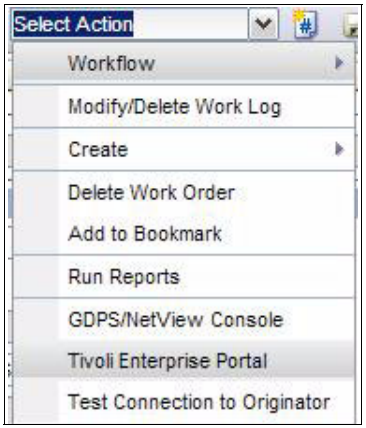


Figure 8-19 Select Action menu

4. Furthermore, detailed queries can be performed on the originating Operational Management product. When you click the **Get Originator Details** button, a set of properties is returned reflecting the current state. See Figure 8-20.

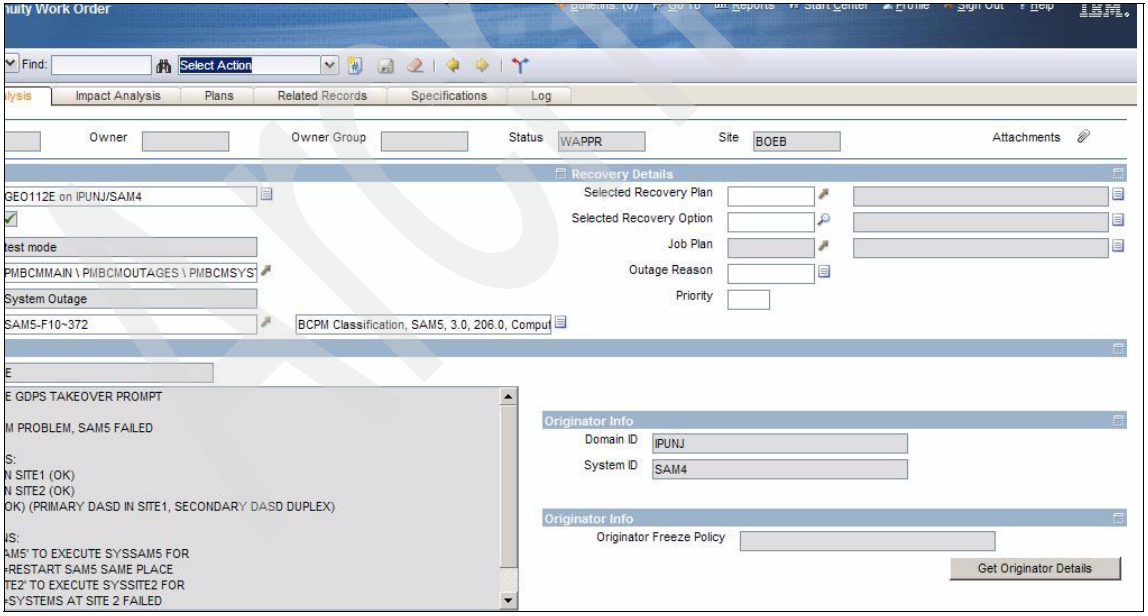


Figure 8-20 Queries on the originating Operational Management product

5. One of the most important pieces of information from GDPS is the freeze policy, as shown in Figure 8-21.

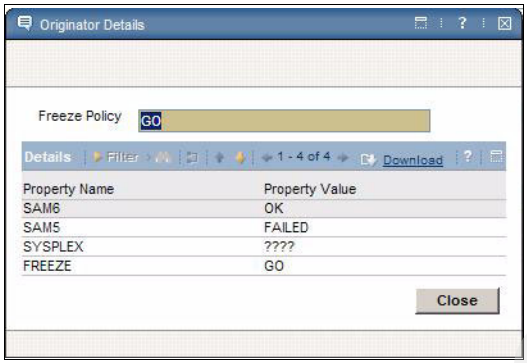


Figure 8-21 Freeze policy

6. The main goal of the analysis step is to find the most appropriate recovery plan for the current situation. Based on the information shown and more information retrieved from the systems using the tools that are available, the analyst process involves searching for and selecting a recovery plan. See Figure 8-22.

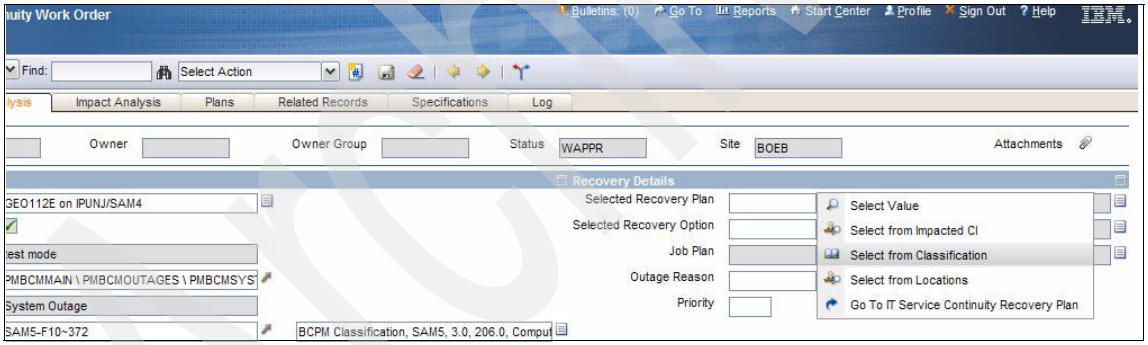


Figure 8-22 Searching for and selecting a recovery plan

7. You can filter recovery plans based on information that was entered when creating the plan (that is, during the planning phase). For example, since the current event is a system outage, let us only show recovery plans that are classified for system outages. See Figure 8-23.

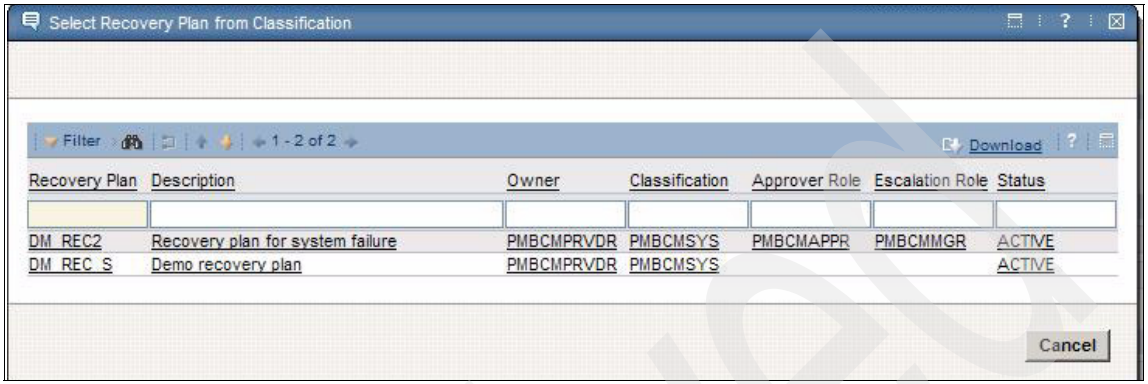


Figure 8-23 Recovery plans that are classified for system outages

8. After selecting a plan, we need to decide on the correct recovery option. For this scenario, we select one of the options offered by GDPS (Figure 8-24). This option involves simply replying to the message on the operator console. As you will see later in this scenario, this action is fully automated.

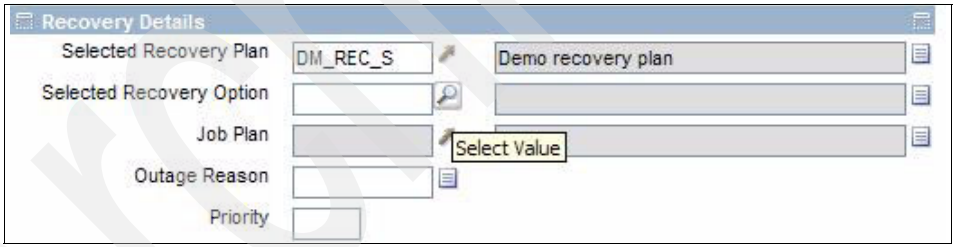


Figure 8-24 Recovery details

Figure 8-25 shows the selected recovery plan.

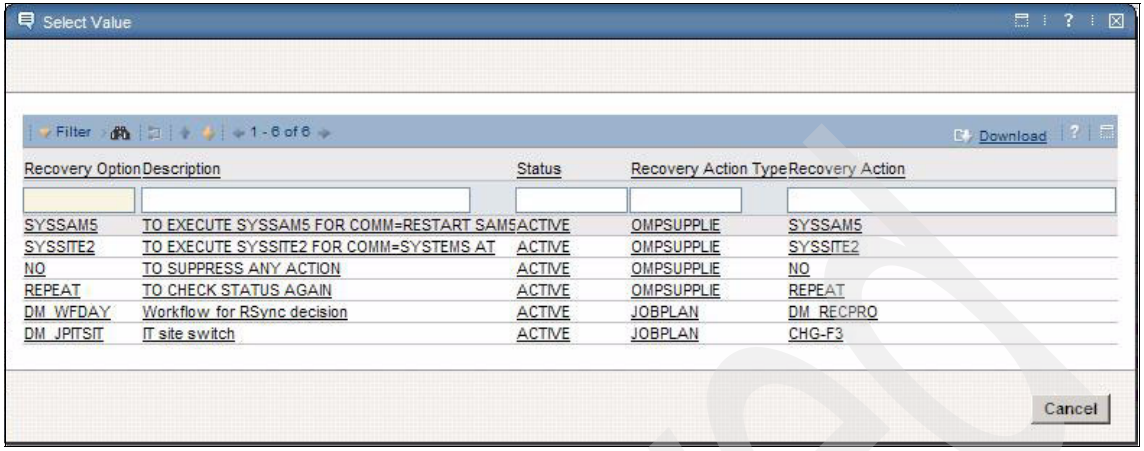


Figure 8-25 Selected recovery plan

9. After selecting the recovery plan and option, the analysis task is done. We now route the workflow for further processing by selecting the Route Workflow icon (Y shaped icon on the right-hand side of the menu in Figure 8-26).

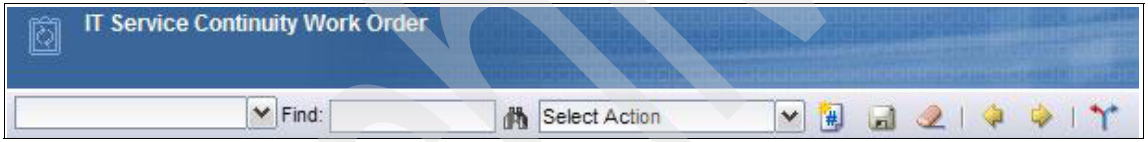


Figure 8-26 Routing the workflow

10. Click **Finish Analysis and Decide Next** (Figure 8-27).

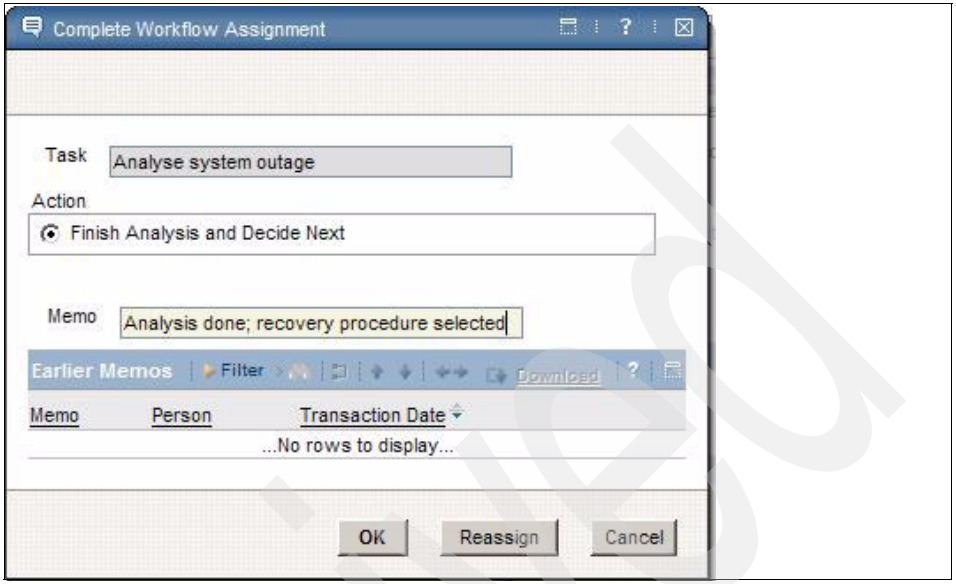


Figure 8-27 Complete Workflow Assignment

11. Click **Execute Recovery Plan** to start the recovery plan (Figure 8-28).

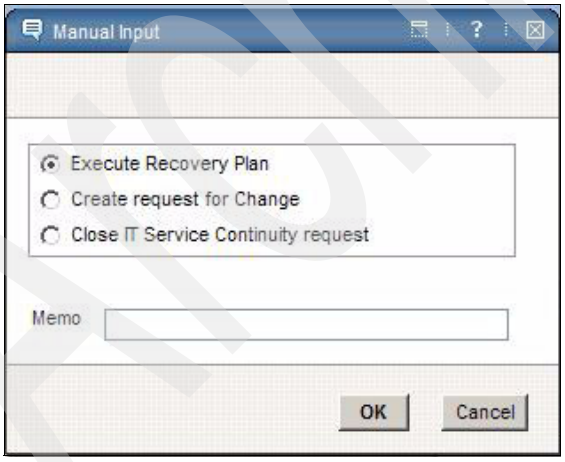


Figure 8-28 Execute Recovery Plan

8.2.4 Declare the disaster

The next role to get a task assignment is the ITSC manager, who declares the disaster and triggers the execution of the recovery procedure.

Upon receiving an e-mail with instructions for the task that must be performed, the ITSC manager logs into CCMDB.

Note: Typically, the ITSC manager does not need to do more than route the workflow. This action is effectively *pressing the emergency button*, which causes the recovery procedure to be started. Simultaneously, notifications and alerts are sent out as specified in the recovery plan to inform staff and users that their business application may not be available for a certain amount of time.

1. We start with logging onto ITSC Manager's Start Center. Figure 8-29 shows ITSC Manager's Start Center.

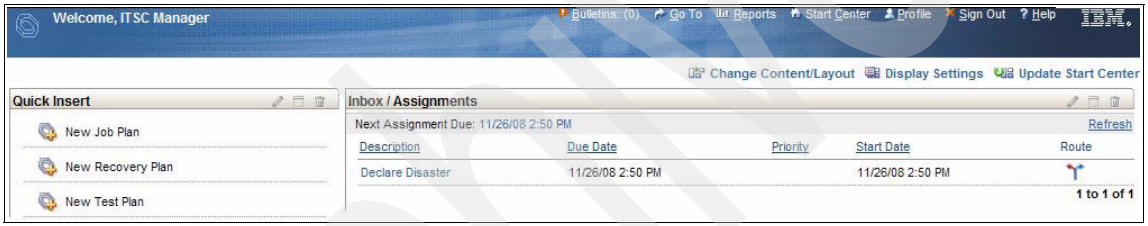


Figure 8-29 ITSC Manager's Start Center

2. In general, the recovery procedure may consist of several steps including further task assignments to subject matter experts handling specific tasks necessary for recovery. To keep this demo simple, we assume a fully automated recovery procedure that requires only that a reply to the GDPS message be sent. Figure 8-30 shows the Complete Workflow Assignment window.

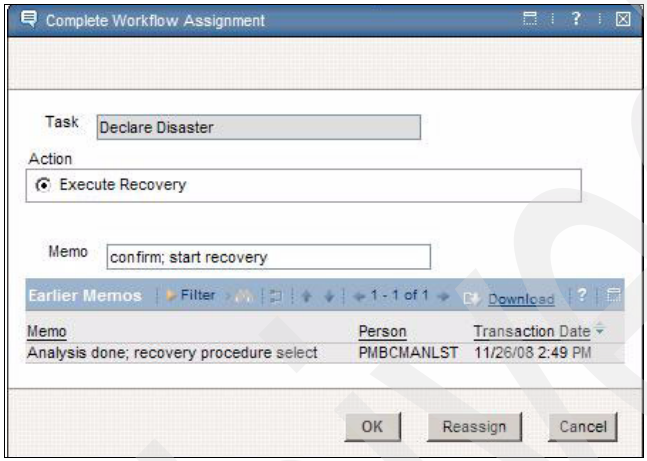


Figure 8-30 Complete Workflow Assignment window

8.2.5 Verify the recovery

The results must be verified after a recovery. Therefore, after the ITSC manager has declared a disaster and routed the workflow, the continuity provider receives a task assignment to perform this necessary verification. The continuity provider has the same capabilities as the ITSC Analyst and uses the information provided in the work order together with the integrated tools to evaluate the systems and decide whether the performed recovery procedure delivered the desired outcome.

1. We start with logging onto the continuity provider's Start Center. (Figure 8-31).

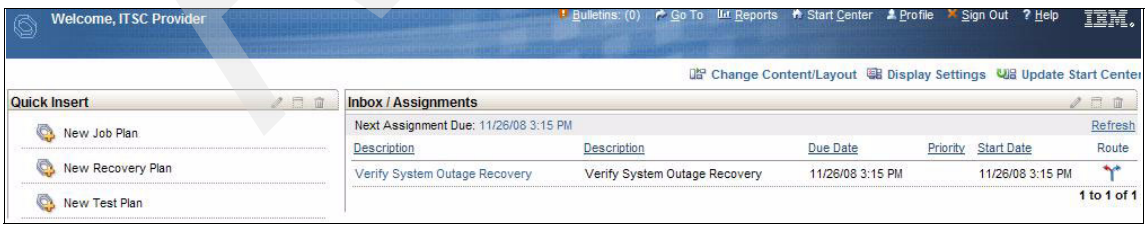


Figure 8-31 Continuity provider's Start Center

Figure 8-32 shows the details of the work order and the message text.

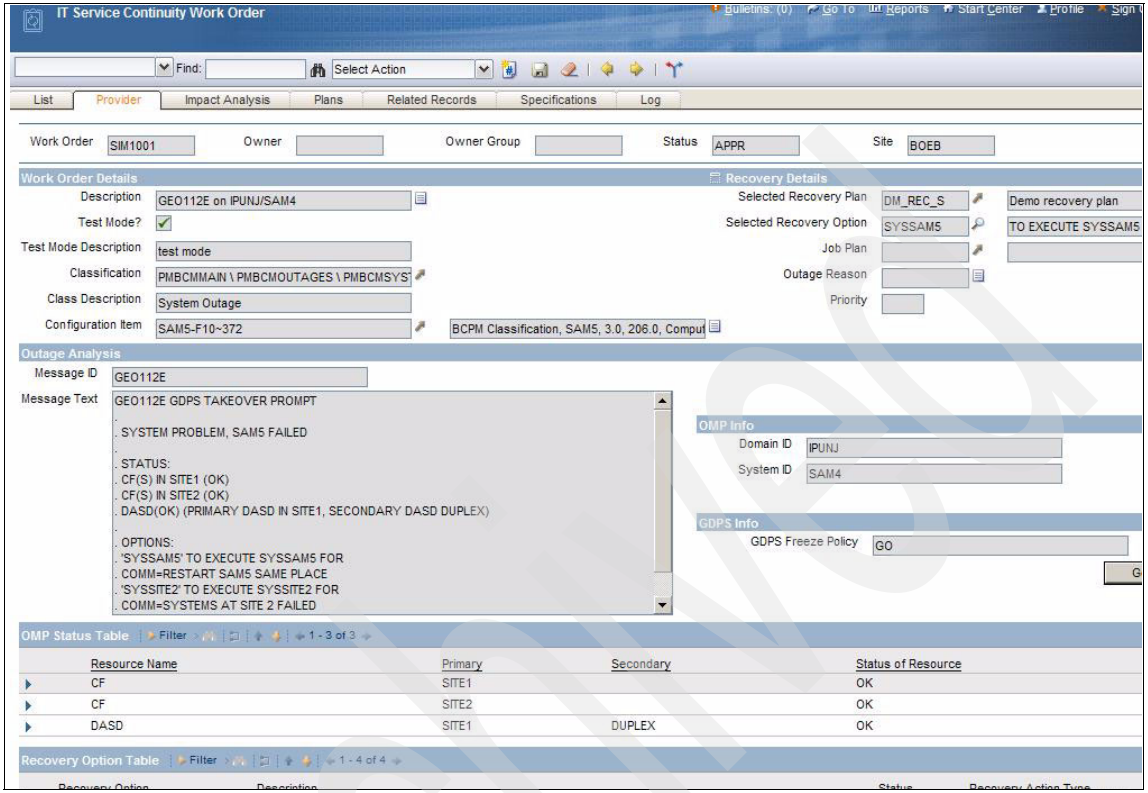


Figure 8-32 Work order

2. Complete the workflow, as shown in Figure 8-33.

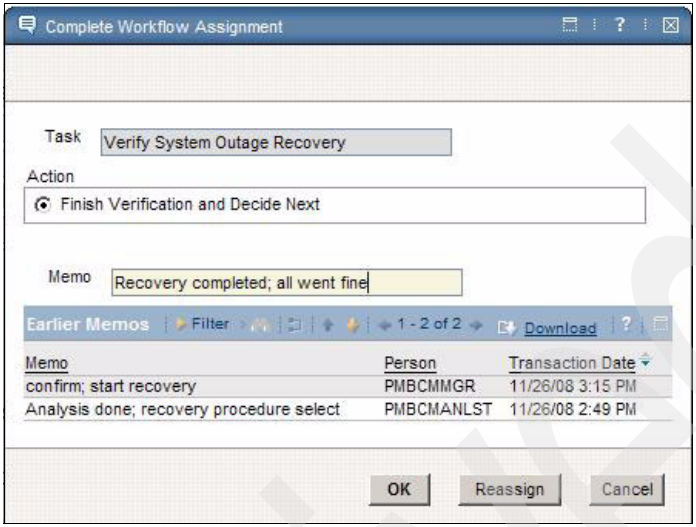


Figure 8-33 Complete Workflow Assignment window

3. Select **Close IT Service Continuity request** to close the request (Figure 8-34).

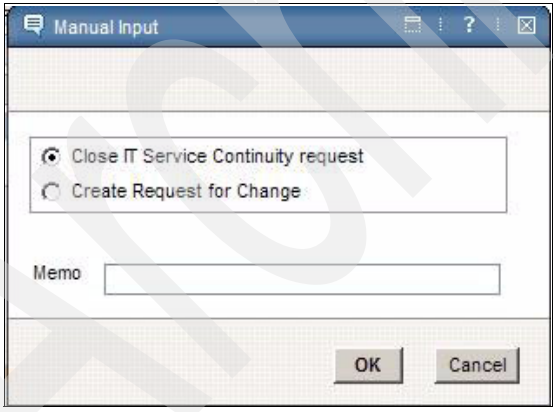


Figure 8-34 Close IT Service Continuity request

8.3 Conclusion

In summary, we have looked at how TBCPM supports the planning aspect by capturing information and creating recovery plans and options.

TBCPM supports testing and simulation through a tailorable simulation environment that allows you to run fire drills or training sessions without impacting or requiring a real environment. TBCPM also assists in handling outage situations by structuring and automating the required tasks via work management and a recovery plan.

You can use various reporting capabilities that are provided to show the availability of plans, the performance of recovery procedures, and much more. Such reports and key performance indicators can be used for auditing purposes as well as to improve existing processes.

Furthermore, detailed logs about test runs are available. Using the information from these logs, you may, for example, monitor test runs.

In conclusion, IBM Tivoli Business Continuity Process Manager makes you better prepared to handle disaster situations.

9

Change impact assessment

This chapter describes the tools that can easily be integrated with Tivoli Service Request Manager to assist analysts and operations staff in quickly determining the business impact of a change request, incident, problem, or service level violation.

We first look at the Service Management roles in which business impact analysis is a key element, and then the essential Tivoli tools that are available for integration with Tivoli Service Request Manager (TSRM) and Tivoli Change and Configuration Management Database (CCMDB).

This chapter concentrates of both Tivoli Business Service Manager and Tivoli Application Dependency Discovery Manager tools that can greatly help with change requests and incidents.

This chapter contains the following sections:

9.1 Change impact assessment

Changes in a business can arise for different reasons:

- Proactively, such as seeking new business opportunities
- Reactively, such as fixing a software bug to resolve a problem

Regardless of the reason, all changes must be managed. Managing changes means managing the potential risk that changes can introduce to your IT services that support your business processes. This is called change impact assessment or business impact assessment.

In order to perform a change impact assessment, the potential impact on the IT services of failed changes must be considered. Information Technology Infrastructure Library (ITIL) defines several generic questions, called the *seven Rs of Change Management*, that could help as a starting point for evaluating the impact of a change. These are:

- Who *raised* the change?
- What is the *reason* for the change?
- What is the *return* required from the change?
- What are the *risks* involved in the change?
- What *resources* are required to deliver the change?
- Who is *responsible* for the build, test, and implementation of the change?
- What is the *relationship* between this change and other change?

Organizations might have different forms to use for impact assessment and the information about who is responsibility for evaluating different types of changes must be clearly defined, but all of this is of little value if there are no proper tools to help determine the impact. The most important consideration is to understand the CIs that are affected with the change and dependencies between those CIs. In that respect, IBM Service Management products come into play. In the following sections we show you several ways to use these tools for impact assessment.

Important: Sometimes change impact assessment and business impact analysis are used interchangeably, but they are different concepts. Change impact assessment is used to assess the risks and impact of a proposed change, whereas business impact analysis (BIA) is a component of an organization's business continuance plan and used to reveal vulnerabilities to critical business processes and to develop strategies to minimize this risk. In this publication we focus on change impact assessment. See 9.6, “Business impact analysis” on page 271, for more discussion on business impact analysis.

9.2 Impact assessment tools

The most common integrated tools for business impact analysis that can be leveraged by Service Management personnel include:

- Tivoli Change and Configuration Management Database (CCMDB)
- Tivoli Application Dependency Discovery Manager (TADDM)

> **Note:** Note that TADDM is a prerequisite for CCMDB, but also can be installed as a standalone product.

- Tivoli Business Service Manager (TBSM)

Other related tools that can also be integrated with SRM include:

- IBM Tivoli Monitoring (Tivoli Enterprise Portal)
- Netcool Impact/Precision
- Tivoli Provisioning Manager (Release Process Manager)

Let us first discuss Service Management roles that would typically use TBSM and TADDM during day-to-day operations.

9.3 Service Management roles

ITIL describes various roles associated with change impact assessment. The following are the most common roles associated with determining the impact of an IT issue from the business perspective:

- Change analyst
- Incident analyst

Other Service Management roles associated with business impact include:

- Problem analyst
- Service level analyst

Refer to either the IBM Tivoli Unified Process or ITIL books for a more in-depth description of all Service Management roles.

9.3.1 Change analyst

The integration between Tivoli Service Request Manager and Tivoli impact assessment tools can provide change analysts or change assessors with a powerful decision-making solution when faced with *requests for change (RFC)* that affect business-critical IT services.

Change analysts are generally tasked with the following responsibilities (Figure 9-1):

- Provide subject matter expertise to understand the impacts of proposed changes.
- Provide business or technical information.
- Provide recommendations to change assessors.

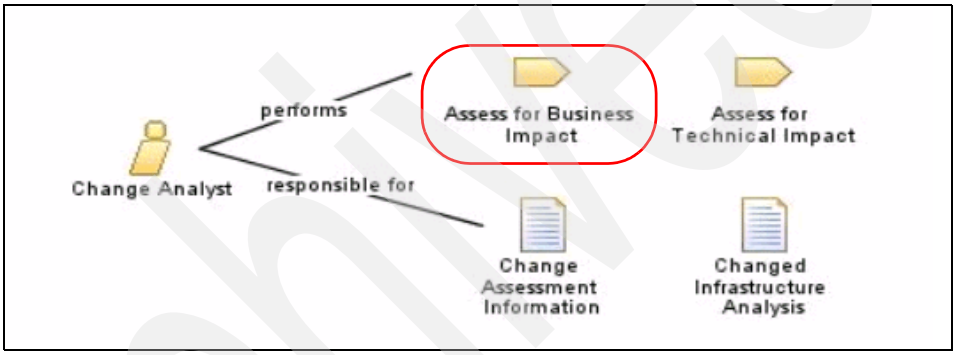


Figure 9-1 Change analyst role

Change analysts perform various tasks to assess business impact. ITUP describes the following tasks performed by a change analyst:

- Determine impact on business operations.
- Review whether the change meets business policies.
- Determine the impact on business resources.
- Review the business notification list.
- Review the priority and impact categorization.
- Analyze support and training plans.
- Review the business impact of rendition.

9.3.2 Incident analyst

Incident analysts also have a vital role to play for assessing the business impact of incidents reported. The incident analyst, being in most instances the second line (or higher) support professional, is the subject matter expert of one or more competency domains. This role is responsible for quickly providing a good analysis of an incident or a solution to it in order to restore the disturbed service as soon as possible. Incidents are typically assigned to the incident analyst by request fulfillment. See Figure 9-2.

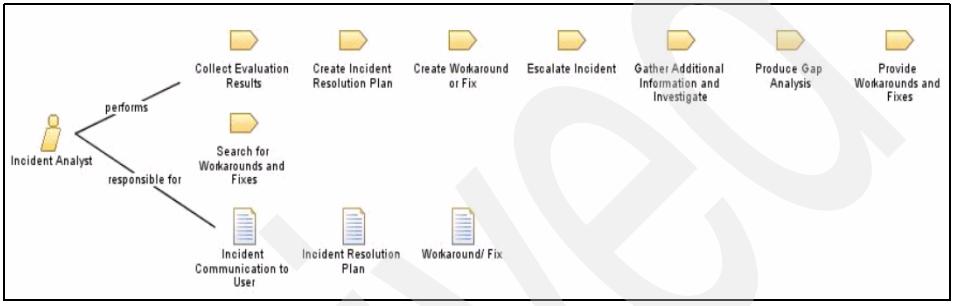


Figure 9-2 Incident analyst

ITUP describes the following tasks performed by an incident analyst:

- Performing incident determination
- Creating a workaround
- Initiating a change request
- Executing a workaround, if applicable
- Installing a permanent fix for the incident
- Executing a resolution, if applicable
- Updating the incident reporting system with resolution information
- Providing effective resolution to the incident in accordance with the priority service level
- Updating the closure portion of the ticket
- Identifying resolved incidents as candidates for inclusion in the operational documentation

9.4 Tivoli Business Service Manager

Tivoli Business Service Manager (TBSM) is IBM Tivoli's premier business Service Management solution. It takes Availability Management to the next level by enabling IT operations and lines of business to more effectively deliver against operational and business objectives.

Utilizing the TBSM real-time dashboards, analysts and operations staff have visibility into the availability, performance, and integrity of critical business services and associated key performance indicators and service level agreements. With this real-time knowledge analysts operations staff can prioritize their response to problems according to the greatest impact on the business.

The TBSM console service shown in Figure 9-3 is an example equities trading line of business service.

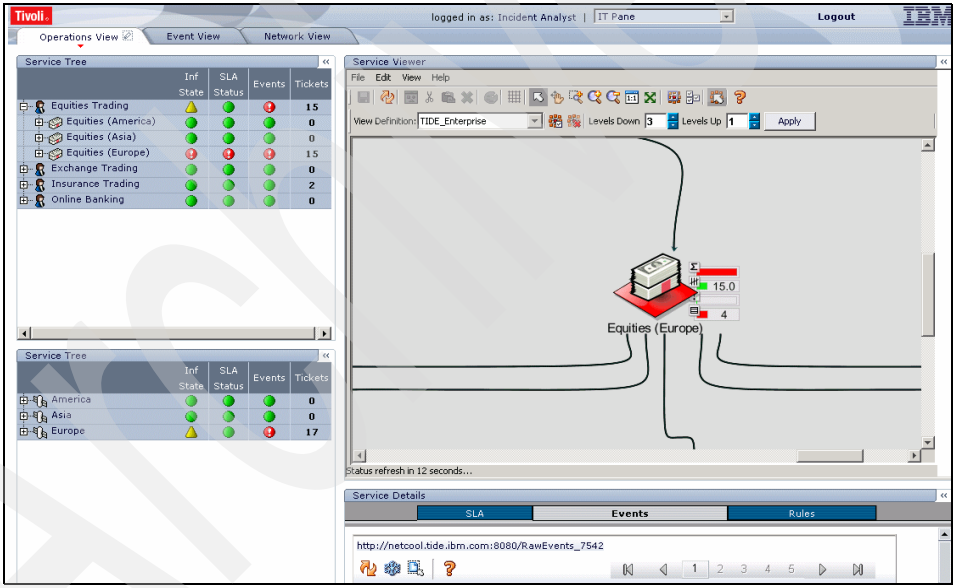


Figure 9-3 Equity trading TBSM LOB service view

TBSM provides several integration points that can be implemented in conjunction with Tivoli Service Request Manager:

- Incident creation (via Netcool Omnibus or Netcool Impact)
- Application link within incident or change applications
  - External hyperlink to TBSM Service view
  - Push button to TBSM Service view

Figure 9-4 shows an example external hyperlink implemented within the Business Assessment Results tab of a change request.

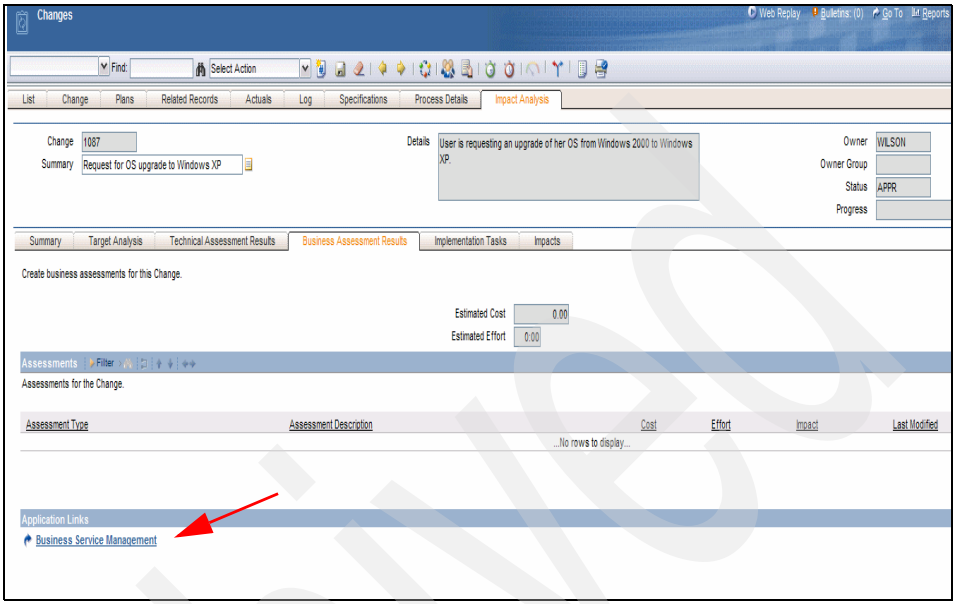


Figure 9-4 Example external hyperlink

The purpose of an external hyperlink from SRM is to quickly launch a TBSM line of business service view from the SRM user's Web browser. This enables the analyst to immediately assess the impact of a change or incident against defined IT services, on demand.

The result of selecting the TBSM hyperlink control within a change request is shown in Figure 9-5.

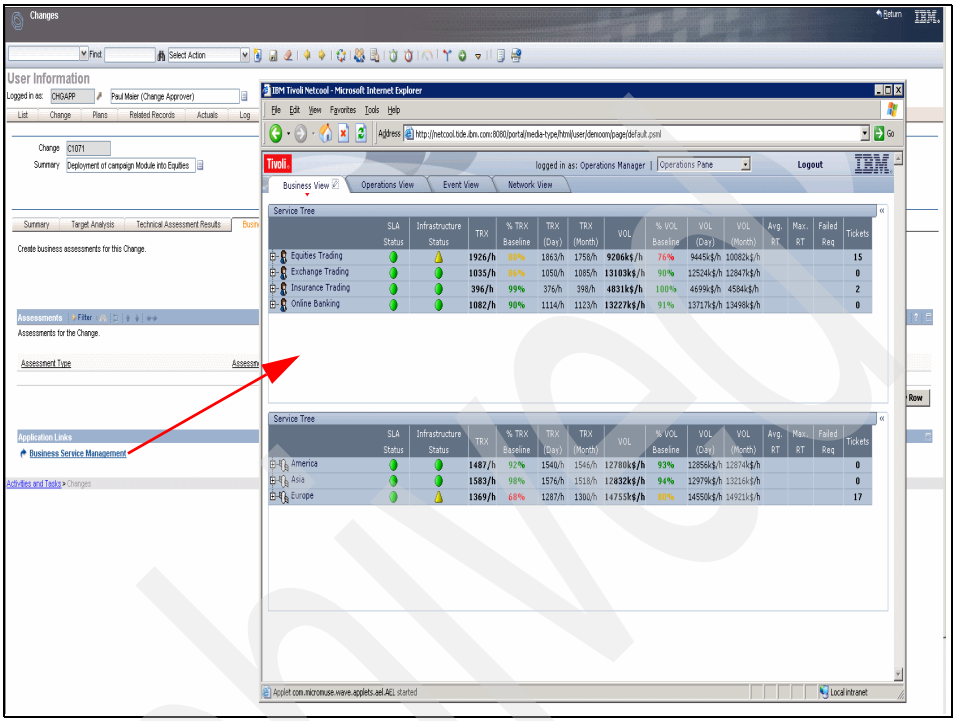


Figure 9-5 Result of external hyperlink

Note: The TBSM console can be set to auto-log in and display a service view using a defined user and password.

The Netcool Omnibus and Impact integrated solutions for SRM are available on the Tivoli OPAL Web site:

http://www-01.ibm.com/software/brandcatalog/portal/opal/results?catalog.catalogName=Tivoli+OPAL&catalog.searchTerms=&catalog.c=Software_IBM_TivoliServiceRequestManager&catalog.start=0

9.4.1 Implementing the External Hyperlink Control to TBSM

This section describes how to create a new hyperlink control, in this case, for the CHANGE application.

To create a hyperlink control that launches TBSM, the following tasks must be completed:

1. Start the SRM Application Designer tool.
2. Edit the CHANGE application.
3. Add a new section to the CHANGE application.
4. Create a hyperlink control.
5. Configure the hyperlink control.

Note: Before you begin, verify that the TBSM host name can be reached from a URL in your systems Web browser. For example, `http://<tbsm host>:8080`.

Complete the following steps:

1. Log in to SRM with a valid user such as maxadmin. You are presented with the Start Center for this user.
2. From the Start Center select **Go To** → **System Configuration** → **Platform Configuration** → **Application Designer**.
3. In the Application field, enter `CHANGE`. From the list of applications returned, click **CHANGE**, as shown in Figure 9-6.

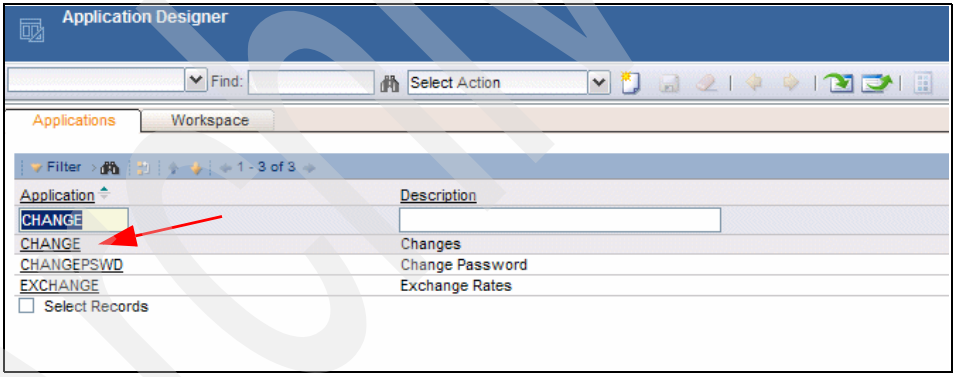


Figure 9-6 Application list

4. Within the CHANGE application, select the **Impact Analysis** tab, followed by **Business Assessment Results**, as demonstrated in Figure 9-7.

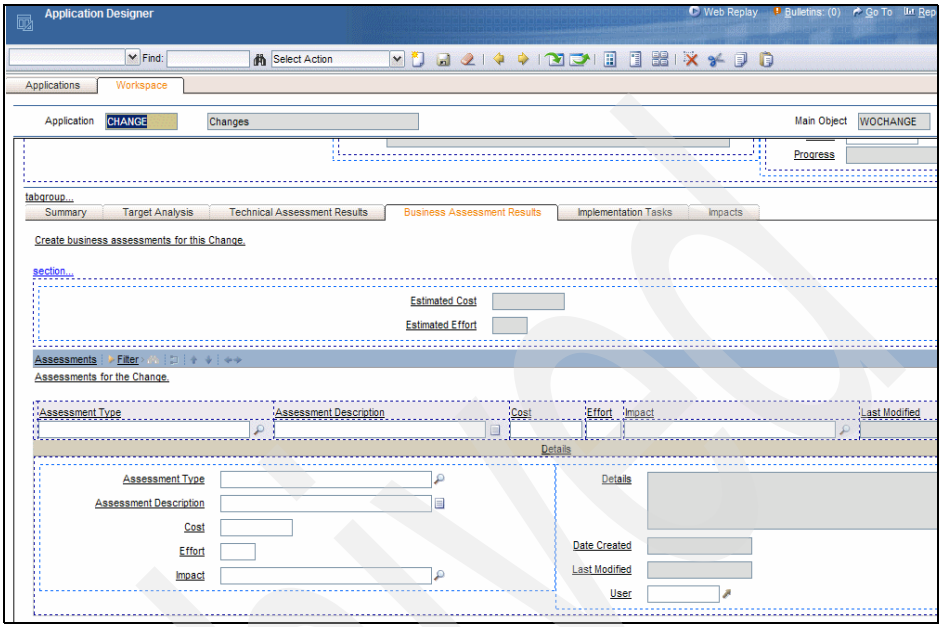


Figure 9-7 Change application

5. From the menu bar select the **Control Palette**, as shown in Figure 9-8.

Figure 9-8 Control palette

You are presented with a window showing a list of available controls (Figure 9-9).

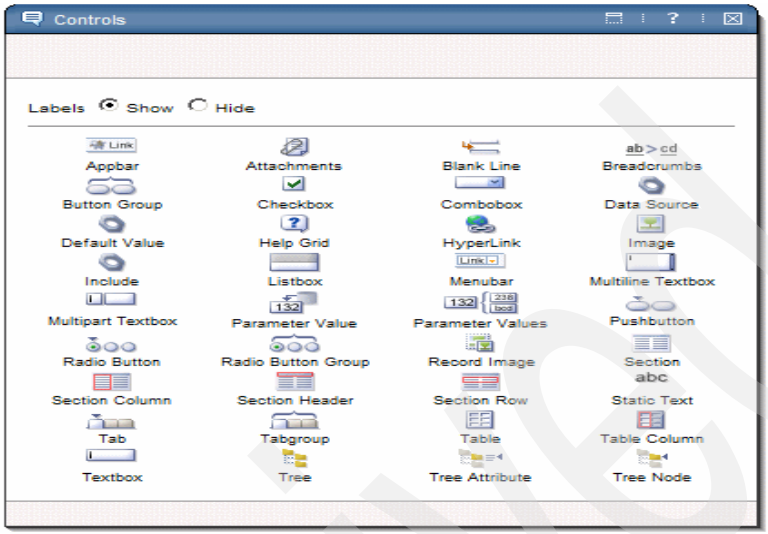


Figure 9-9 Controls

6. Drag the Section icon to beneath where buttongroup is displayed, as shown in Figure 9-10.

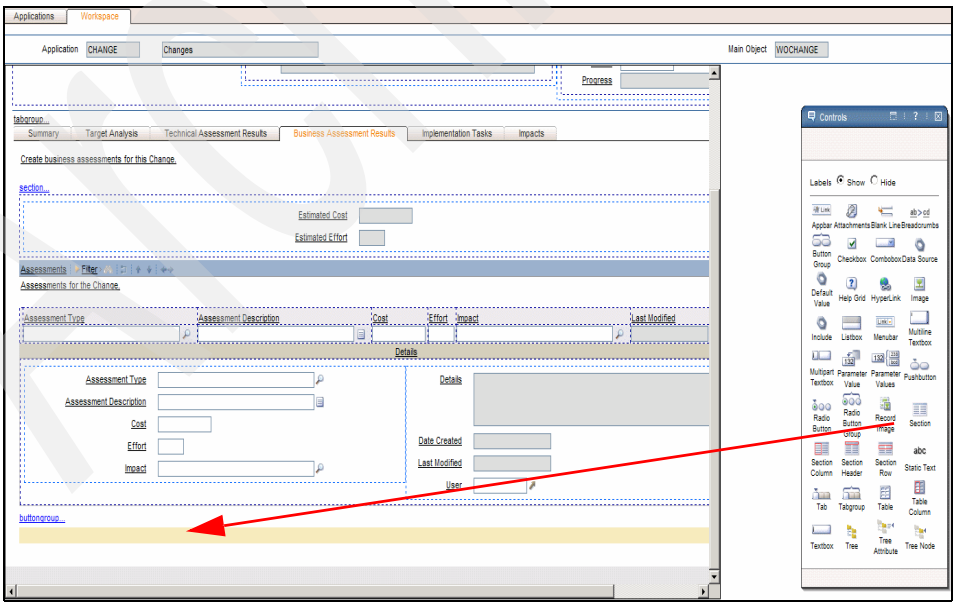


Figure 9-10 Create new section

Note: You may encounter an error if an invalid area of the panel is selected, such as the footer.

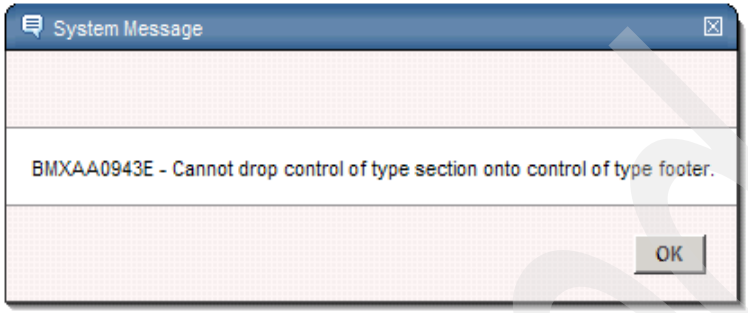


Figure 9-11 Control error message

You should now see a new section added to the application, as shown in Figure 9-12.

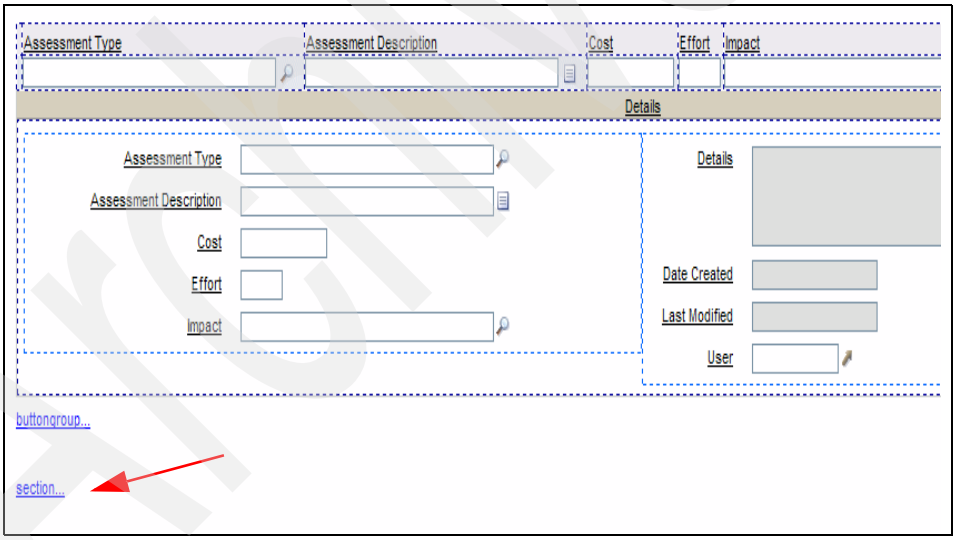


Figure 9-12 New section

7. Right-click over the new section and select **Properties**. Use Table 9-1 as a guide for configuring the properties. Only the label property must be setting.

Table 9-1 Section property table

Property	Value
Control ID	new_section
Label	Application Links
Show Border?	
Collapsed?	
Description	
Data source ID	
Business Object	
Parent Data Source ID	
Input Mode	
Listeners	
Signature Option	
Sig Option Data Source ID	

The new section properties should now be similar to what is shown in Figure 9-13.

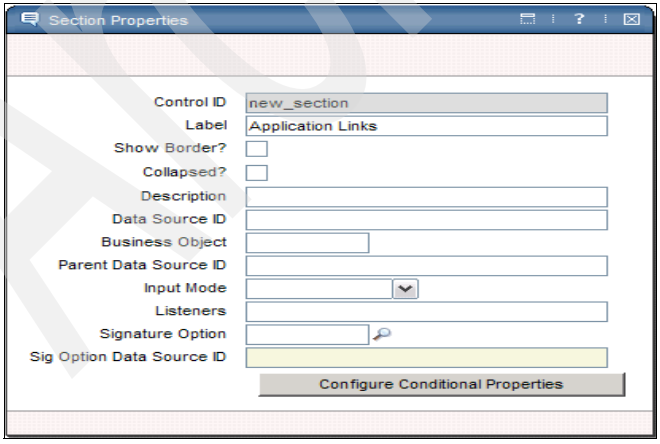


Figure 9-13 Section property window

8. Close the Section properties window and click the Save Application Definition icon, which is located on the top menu bar.
9. Drag the Hyperlink control icon over to the new section (Figure 9-14).

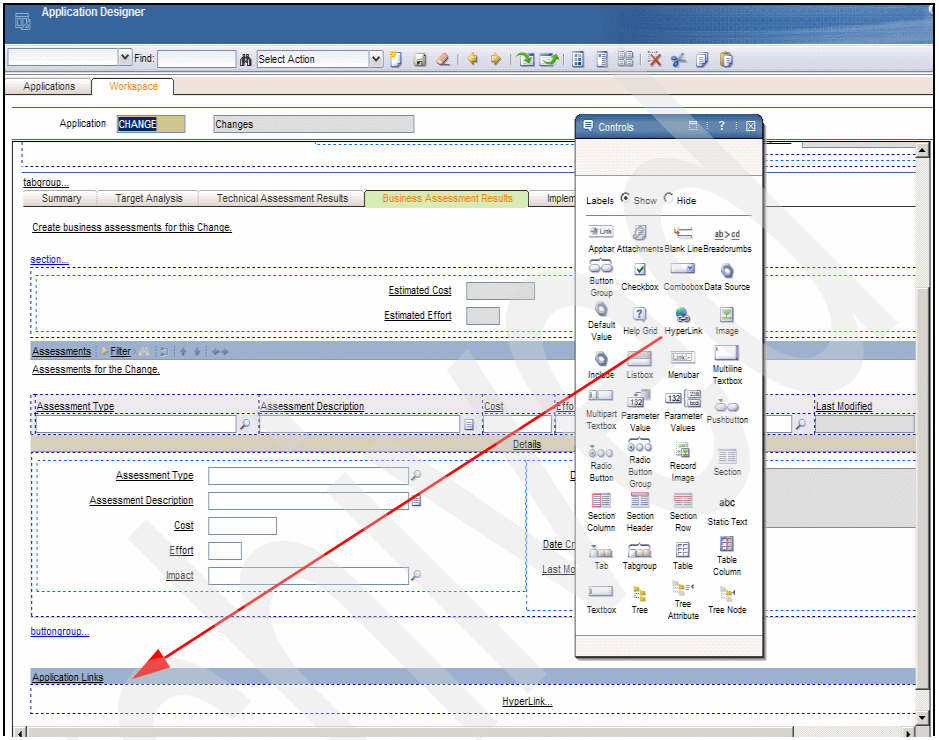


Figure 9-14 Create hyperlink control

You should now see the hyperlink control displayed in the center of the section, as shown in Table 9-2.

10. Close the Control palette.
11. Right-click over the Hyperlink control and select **Properties**. Configure the properties using Table 9-2 as a reference.

Table 9-2 Hyperlink property table

Property	Value
Control ID	new_hyperlink_2
Label	Business Service Management
Text Alignment	Left

Property	Value
Column	
URL	http://<host>:8080/portal?action=JLoginUser&request_url=http://<host>:8080/portal/media-type/html/user/<user>/page/default.psml&username=<user>&password=<password>
Image Filename	menu_icon_link.gif
Image Alignment	
Separator?	
CSS Classname	buttongrouplabel
Access key	
Control Target ID	
Event Type	
Event Value	
Signature Option	
Sig Option Data Source ID	

Note: The Control ID property is automatically set by SRM.

The URL property in this example is set to automatically log in and start the default.psml page configured for the user. For in-depth information about how to create pages, consult the *Tivoli Business Service Manager Scenarios Guide,* SC23-6043.

The Hyperlink Properties panel should now look similar to that shown in Figure 9-15.

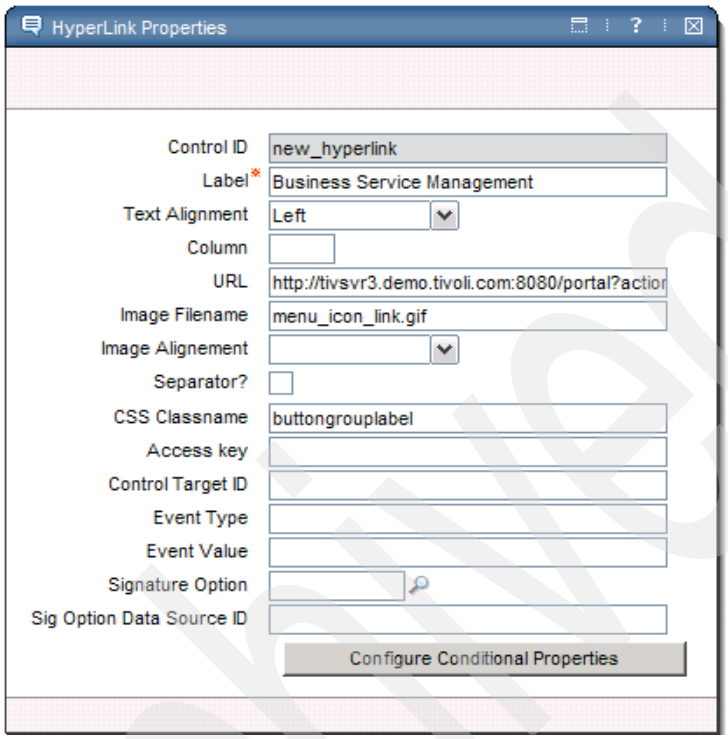


Figure 9-15 Hyperlink properties

12. Close the Hyperlink Properties window and click the Save Application Definition icon to save the changes. The TBSM hyperlink control is now created and ready to test, as shown in Figure 9-16.

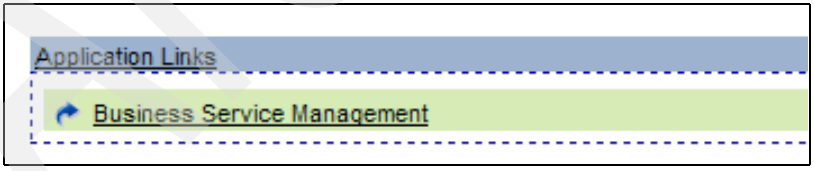


Figure 9-16 Business Service Management hyperlink

13. Quit the application designer by going to your Start Center.

9.4.2 TSBM 4.2 integration possibilities

At the time of writing this book, TBSM Version 4.2 was made generally available (GA). With this release you can further integrate with SRM to launch TBSM service views, which leverages the new *Tivoli Integrated Portal (TIP)*.

You can open a service directly from a URL, using this URL format:

```
https://tipserver:port/ibm/action/launch?pageID=unique page id
```

You can access information about service availability and service administration. When entering the URL, you must specify the name of the service, not the display name.

To open a service using a URL:

1. Enter a URL using the following format:

   `https://`*`tipserver`*`:`*`port`*`/ibm/action/launch?pageID=`*`unique page id`*

   Where *TIPserver* and *port* specify the location of the TBSM server that you want to access and *unique page id* is one of the following:

   – com.ibm.tbsm.navigationElement.desktop (This opens the Service Availability page.)
   – com.ibm.tbsm.navigationElement.serviceAdmin (This opens the Service Administration page.)

2. Include one or more of the following parameters on the URL:

   – &Service InstanceID
   – &Service InstanceName
   – &ManagedSystemName
   – &Guid
   – &MSSName (required)
   – &SourceToken (required)
   – &CDMClass (optional with &MSSName and &SourceToken, but recommended)

These parameters are searched for in the order listed. The first parameter that is found is used for context reference, and the remainder are ignored.

In addition, the Service Availability page supports the &View parameter, which enables the view definition that is used in the service viewer to be changed from the default. This parameter can have the value BusinessImpact or BusinessImpactAll. For example:

```
https://intwin4.tivlab.raleigh.ibm.com:16316/ibm/action/launch?pageID=com.
ibm.tbsm.navigationElement.desktop&ManagedSystemName=Primary:INTWIN3:NT
```

So what you could do now is navigate straight to the configuration item INTWIN3:NT in TBSM from the hyperlink control.

You can see in Figure 9-17 that the configuration item WINXP126:NT will be evaluated by the SRM attribute {CINUM} in the hyperlink control.

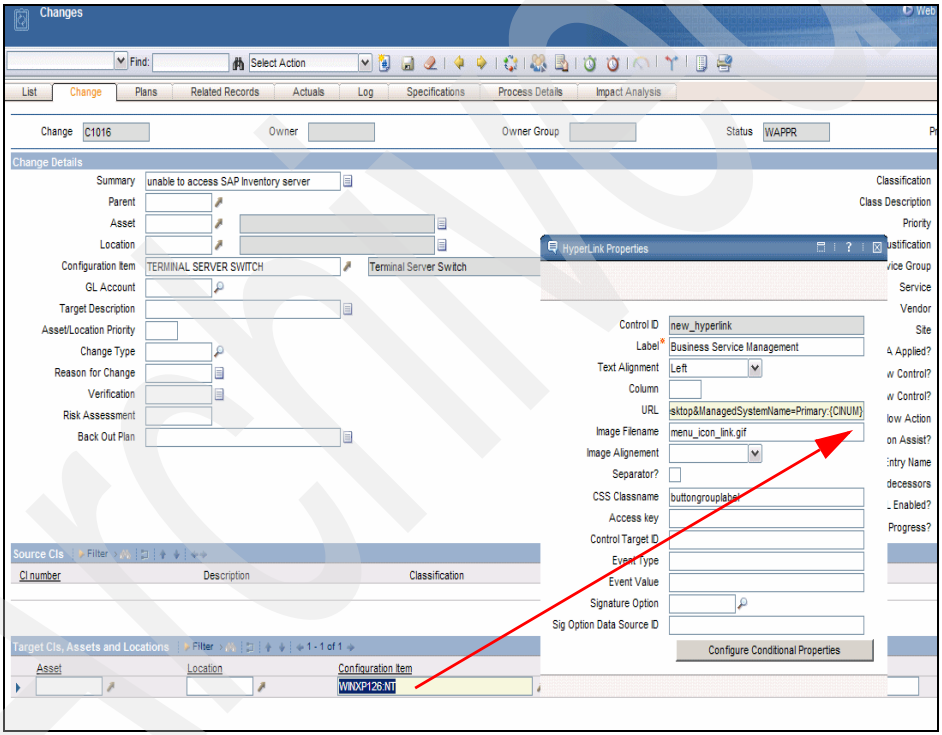


Figure 9-17 CINUM attribute

Refer to the TBSM 4.2 documentation at the link below to learn more about this release:

http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.tivoli.itbsm.doc/welcome.htm

9.5 Tivoli Application Dependency Discovery Manager

Tivoli Application Dependency Discovery Manager (TADDM) is a pivotal component of an IBM Tivoli-based Service Management solution in determining the impact of an IT service from a business perspective.

Impact analysis is a crucial part of evaluating a proposed change and ensuring that it is implemented within your company's or business unit's guidelines.

Figure 9-18 shows the TADDM main panel.

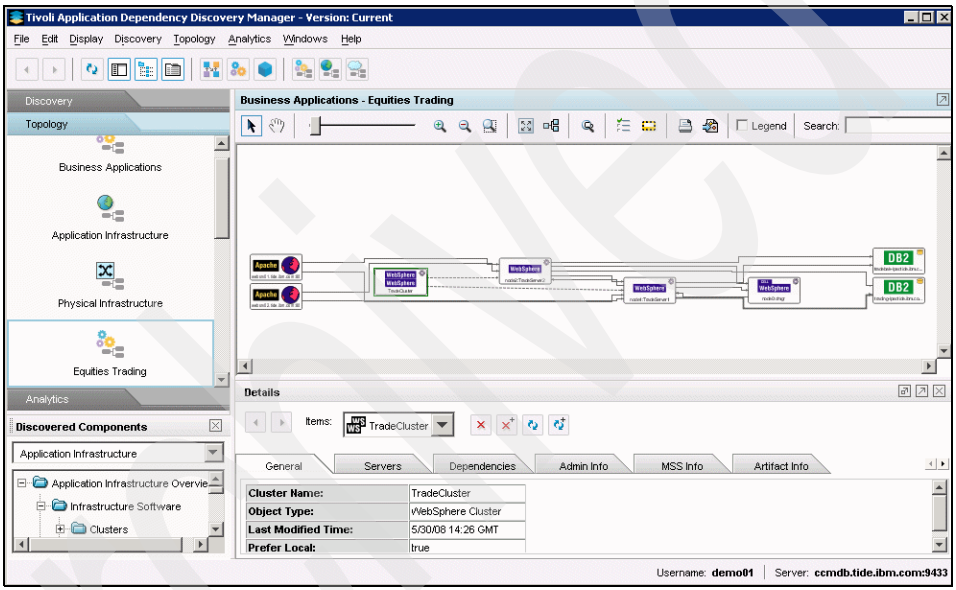


Figure 9-18 TADDM

TADDM is usually implemented in conjunction with CCMDB, so for Tivoli Service Request Manager, CCMDB 7.x must also have been installed and configured.

Generally speaking, the following integration points are most useful for impact analysis between SRM, CCMDB, and TADDM:

- Launch-in-context
- External hyperlink control
- Push button control

To better explain some of these integration possibilities, let us walk through an example Launch-in-Context scenario to assess a configuration item and the impact that it has.

We also show how to create a push button from within an Incident.

9.5.1 TADDM launch-in-context

The TADDM launch entries are created as part of a CCMDB 7.x installation, as shown in Figure 9-19.

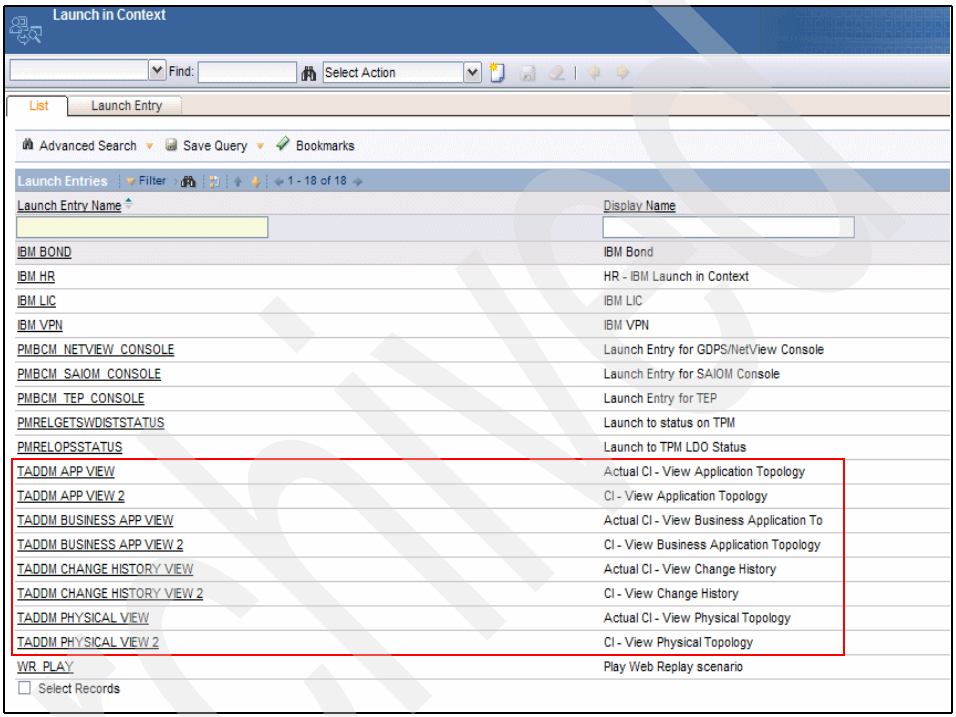


Figure 9-19 TADDM launch-in-context entries

To correctly configure TADDM launch entries within SRM/CCMDB:

1. Log in to SRM with a valid user such as maxadmin. You are presented with the Start Center for this user.
2. From the Start Center select **Go To** → **System Configuration** → **Platform Configuration** → **Launch in Context**.

3. You will be presented with the Launch in Context window, as shown in Figure 9-19 on page 254. Select a TADDM launch entry to configure, for example, TADDM APP VIEW.

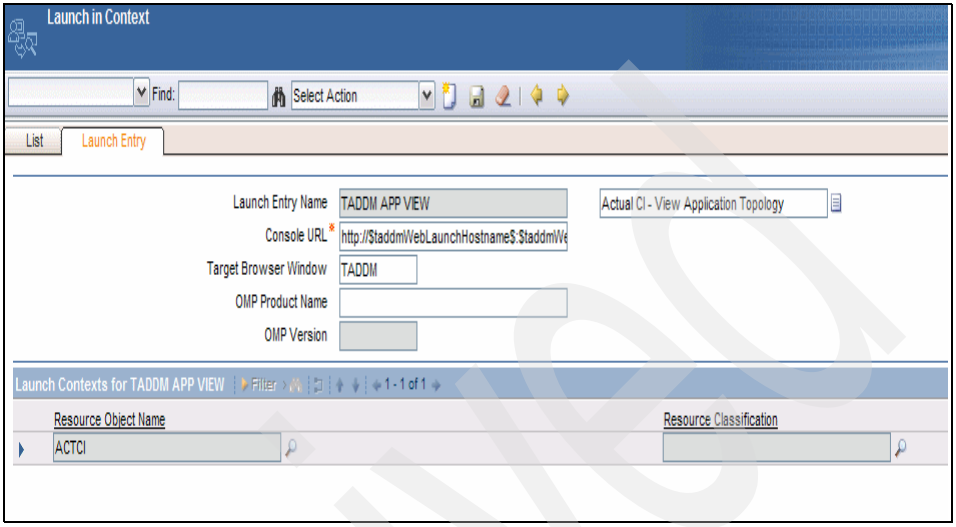


Figure 9-20 TADDM APP VIEW

The following syntax shows the URL format to use for launching in context that can be used for the Console URL property:

```
http://<TADDMHostname>:<TADDMPort>/<ContextRoot>/?<queryString>
```

The possible launch-in-context values are detailed in Table 9-3.

Table 9-3 nameValuePairs parameters and values

Parameter	Valid values
guid	Valid string representation of a GUID.
target	The following string values are valid: ► Existing ► New

Parameter	Valid values
graph	The following string values are valid: ► physicalinfrastructure ► applicationinfrastrure ► businessapplications ► app_software ► app_physical ► bus_svc_software ► bus_svc_phyiscal ► collection_relationship ► collection_physical
view	The valid string value is changehistory.
days_previous	A number.
console	The following string values are valid: ► Web ► Java

An example TADDM launch-in-context URL is:

```
http://tivsvr2:9430/cdm/servlet/LICServlet?default.port=9433&graph=
businessapplications
```

Note: For further information about launch-in-context possibilities refer to the *TADDM SDK Developer's Guide* at:

```
http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?
topic=/com.ibm.taddm.doc_7.1/SDKDevGuide/c_cmdbsdk_introducingsdk.
html
```

4. Modify the console URL string with a valid URL using the syntax just described in the previous step.

The Launch in Context entry for TADDM APP VIEW should now look something like what is shown in Figure 9-21.

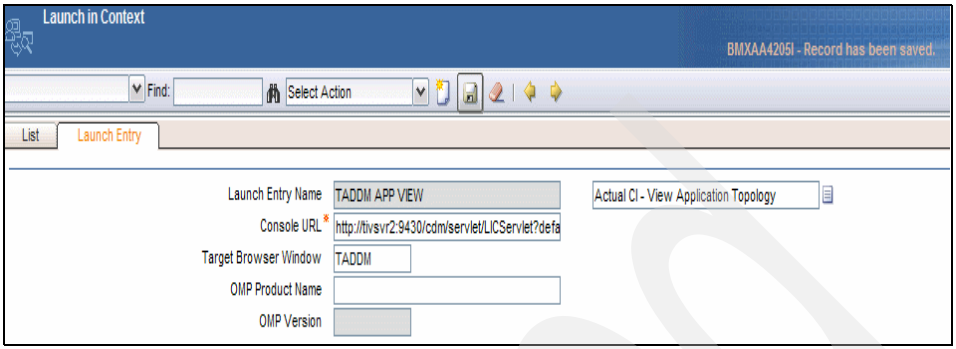


Figure 9-21 Launch in Context entry for TADDM APP VIEW

5. Click **Save**.
6. Repeat the same steps for all other TADDM launch-in-context entries that you intend to use with SRM.

9.5.2 Equities trading scenario

In this scenario, the SRM change request C1058 has been created for a new campaign module for the equities trading application, and is being assessed by the change analyst. See Figure 9-22.

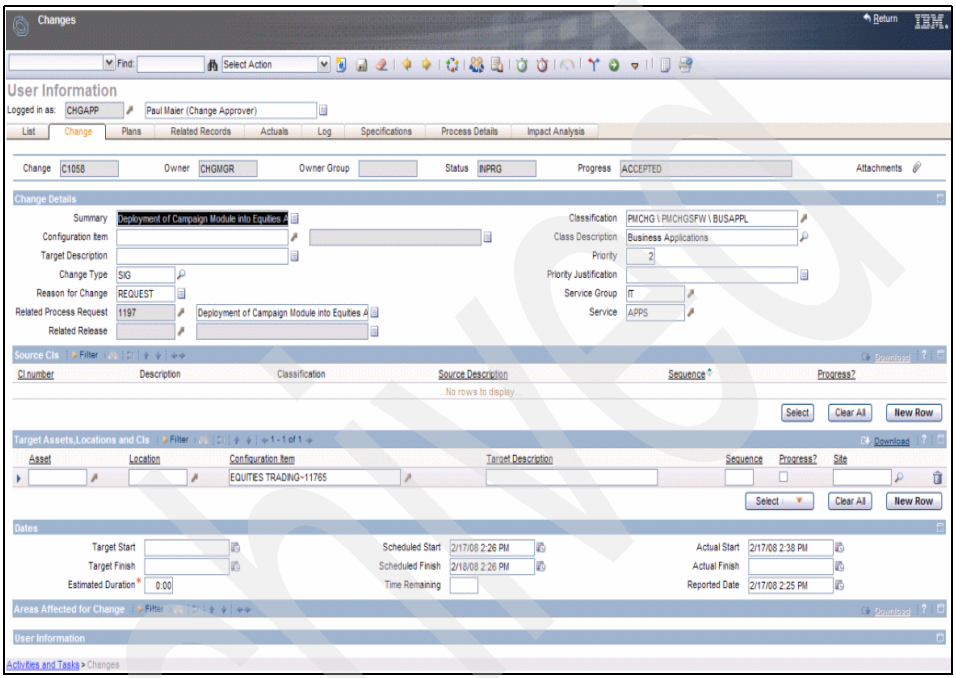


Figure 9-22 Equities trading change request

The steps are:

1. Select **Impact Analysis** from within the change request, as shown in Figure 9-23.

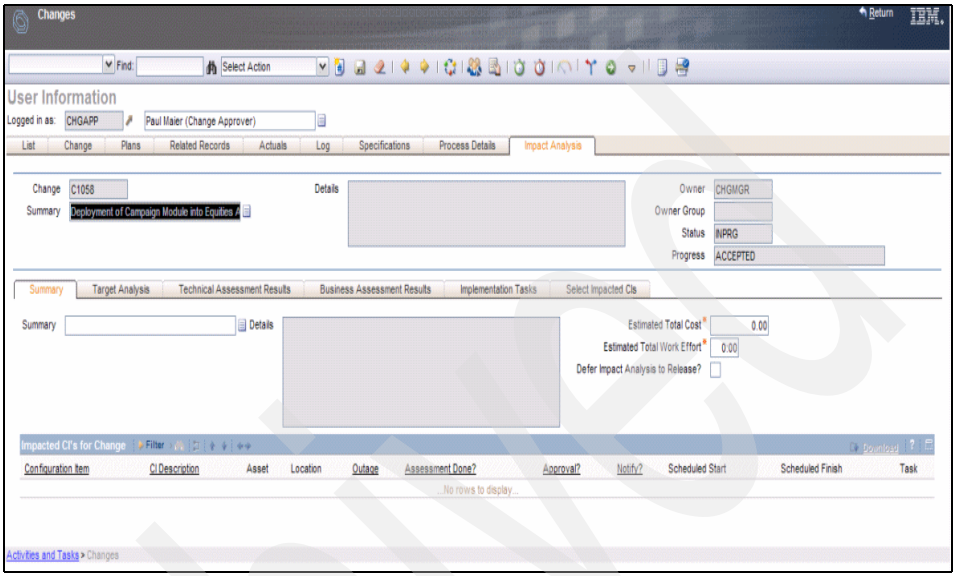


Figure 9-23 Impact analysis

2. Select the **Target Analysis** tab to reveal the configuration item (Figure 9-24).

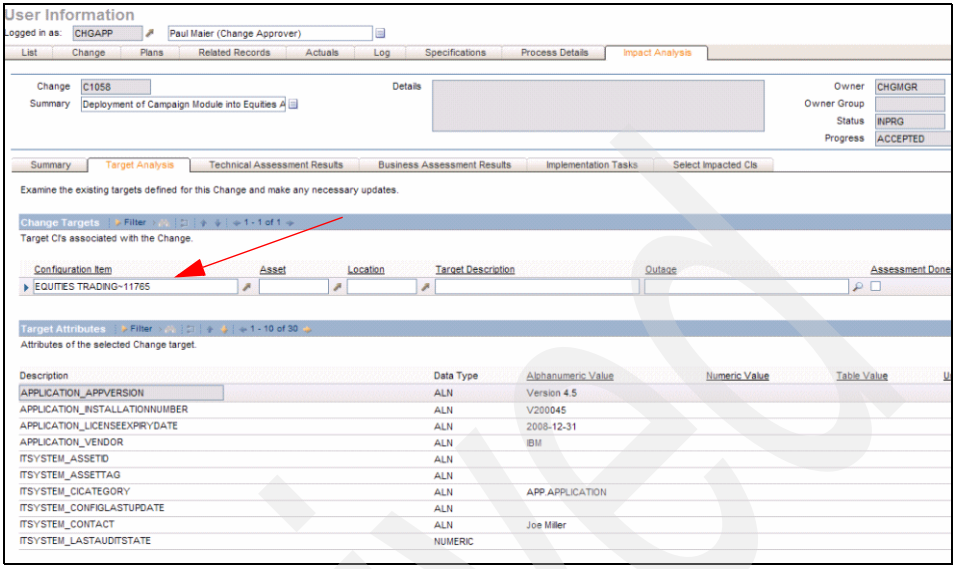


Figure 9-24 Target analysis

3. The next step is to click the arrow to the right of the configuration item and select **Go to Configuration Items**, as shown in Figure 9-25.

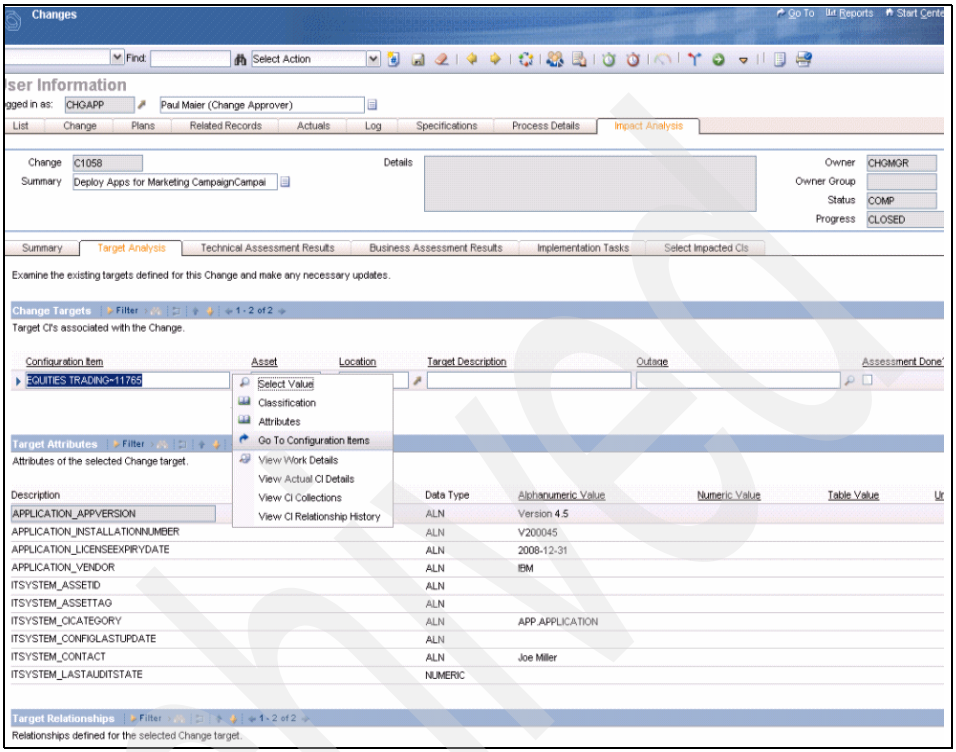


Figure 9-25 Target analysis configuration Items

4. From the Select Action menu, select **View Actual CI Topology → Business Application**, as shown in Figure 9-26.

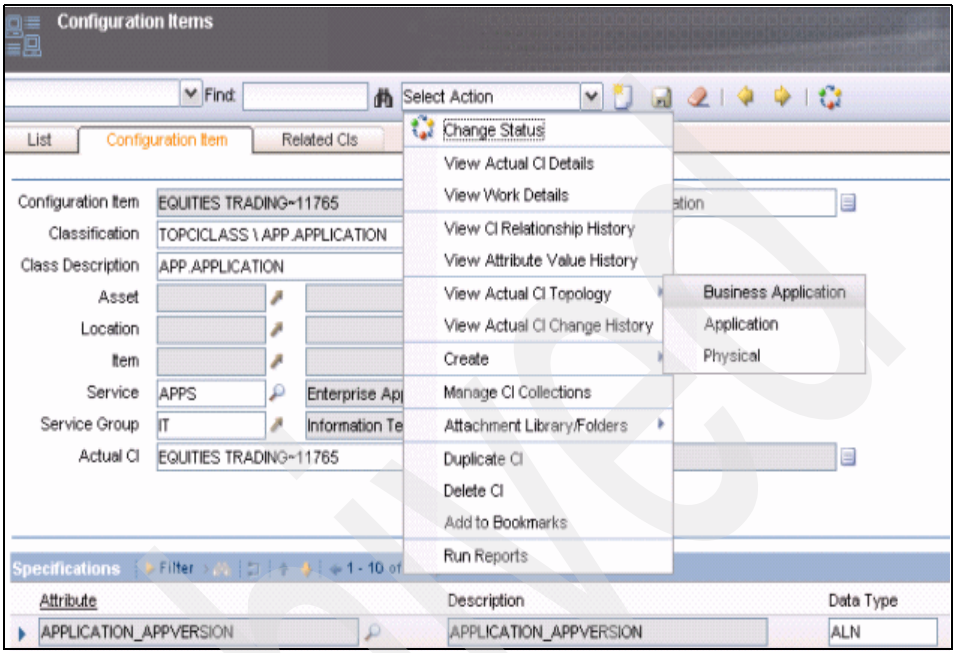


Figure 9-26 View Actual CI topology

Assuming that the TADDM launch-in-context entries have been correctly configured in SRM, you should now be presented with a TADDM Business Application view, as shown in Figure 9-27.

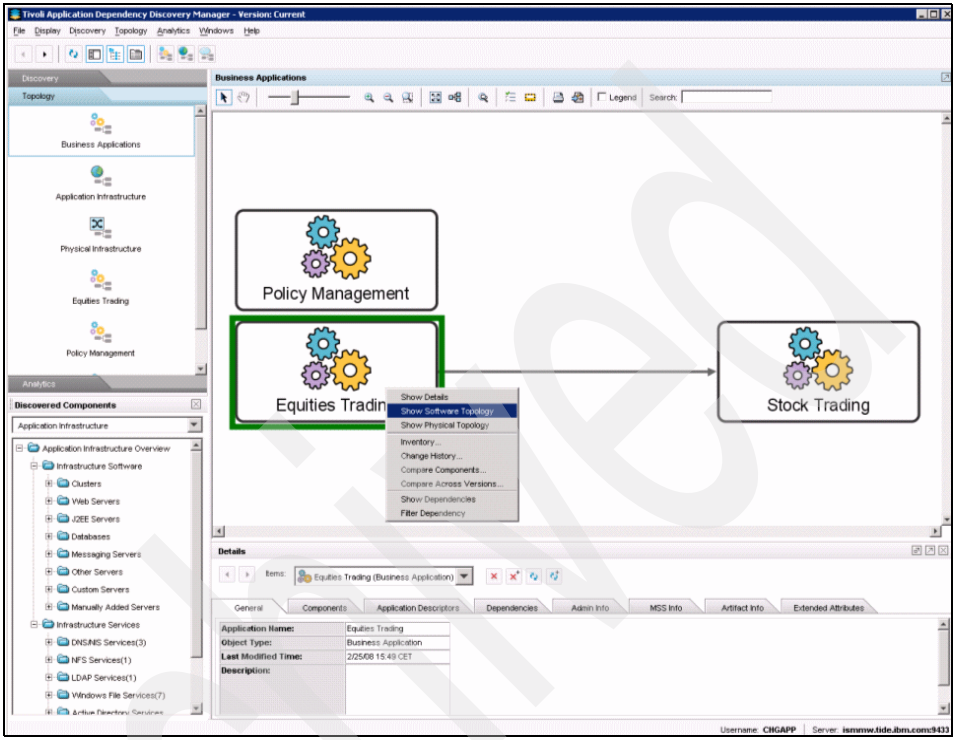


Figure 9-27 Business applications topology

5. Right-click the business application, in this scenario **Equities Trading**, and select **Show Software Topology** (Figure 9-28).

   The change analyst can now assess the business application and review all the relationships between each component that makes up the service. Figure 9-28 shows the software topology for the equities trading business application.

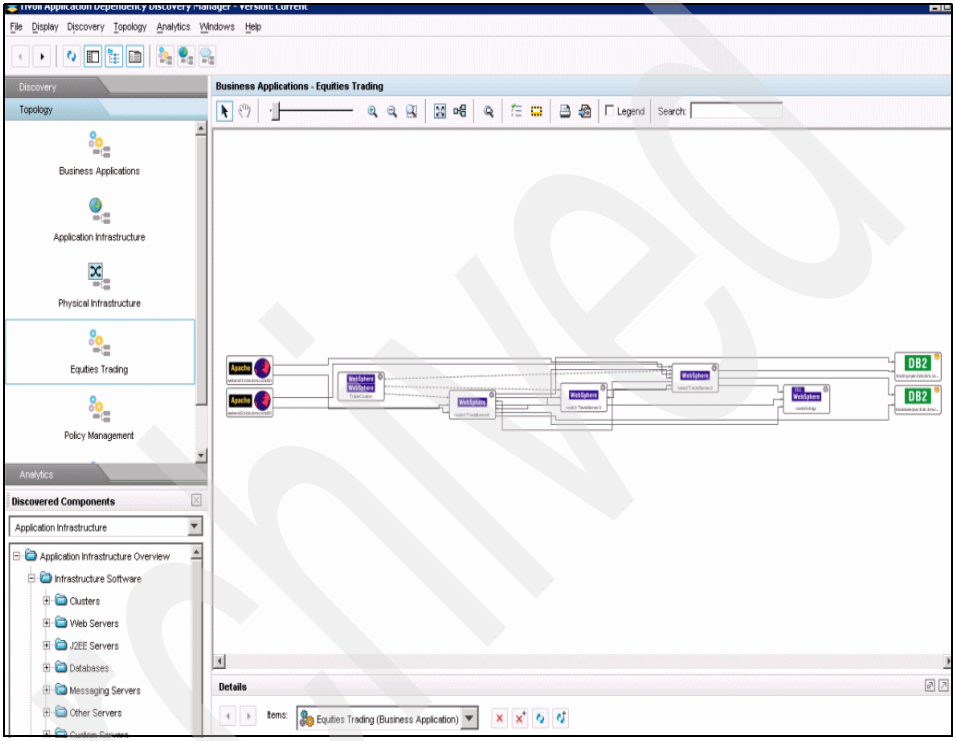


Figure 9-28 Equities trading software topology

The change analyst should also now review the physical topology that the business application supports.

6. Right-click the business application, as shown in Figure 9-27 on page 263, and select **Show Physical Topology**. You are presented with the physical components of the business application.

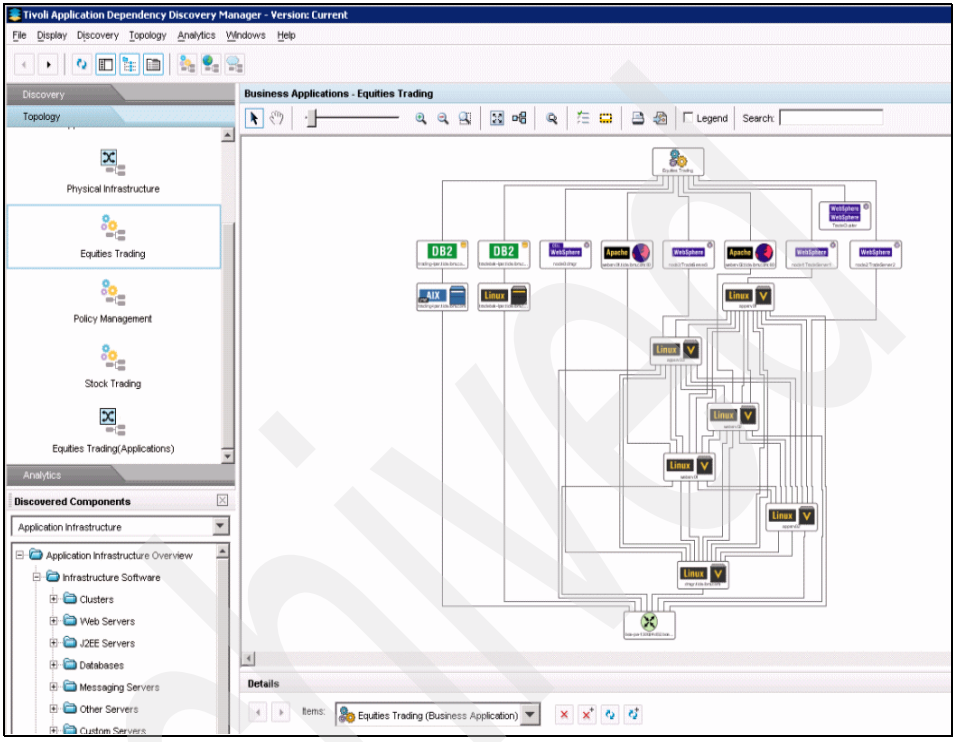


Figure 9-29 Equities trading hardware topology

9.5.3 Conclusion

After reviewing both the software and the physical aspects of the business application affected by a change request, the change analyst should have gained the necessary knowledge to make the correct decision about whether to approve the change request, to request more information, and so on.

9.5.4 Implementing the push button control to TADDM

If the Tivoli CCMDB has not been implemented with SRM, it is possible to create a simple push button control within a change request or incident to launch the TADDM console.

To create a push button control that launches TBSM, complete the following tasks:

1. Start the SRM application designer tool.
2. Edit the incident application.
3. Add a new section to the incident application.
4. Create a bush button control.
5. Configure the push button control.

Note: Before you begin, verify that the TADDM host name can be reached from a URL in your systems Web browser. For example, `http://<taddm host>:9430`.

Complete the following steps:

1. Log in to SRM with a valid user such as maxadmin. You are presented with the Start Center for this user.
2. Complete the following sequence from the Start Center:
   a. Select **Go To** → **System Configuration** → **Platform Configuration** → **Application Designer**.
   b. In the Application field, enter `INCIDENT`. From the list of applications returned, click **INCIDENT**, as shown in Figure 9-30.

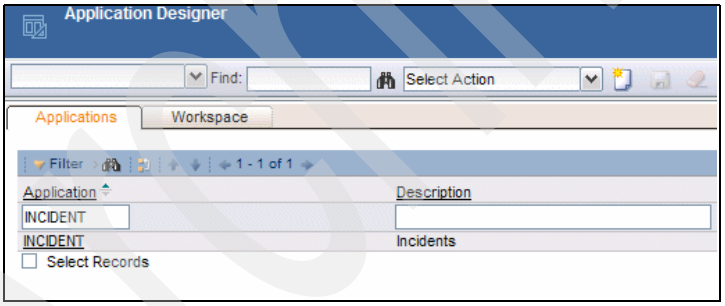


Figure 9-30 Application selection

3. Within the INCIDENT application, select the **Incident** tab, as demonstrated in Figure 9-31.

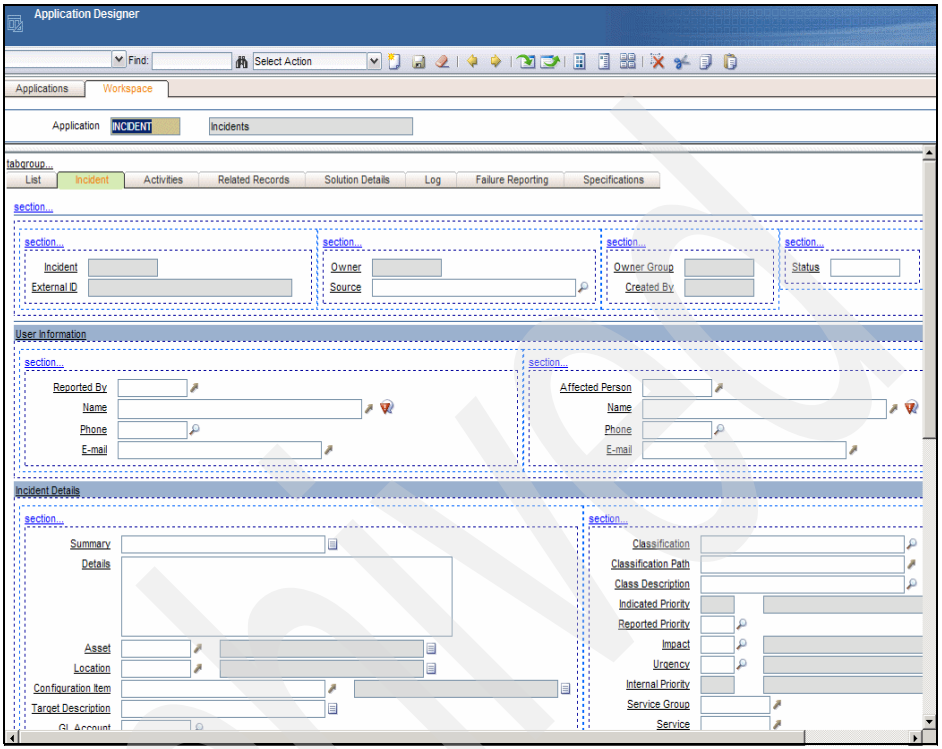


Figure 9-31 Incident tab

4. Scroll down to the bottom of the Incident tab. You should see the section buttongroup.
5. Repeat the steps explained in “Implementing the External Hyperlink Control to TBSM” on page 242 to create a new section.

6. From the Control Palette, drag the Pushbutton icon over to the new section, as shown in Figure 9-32.

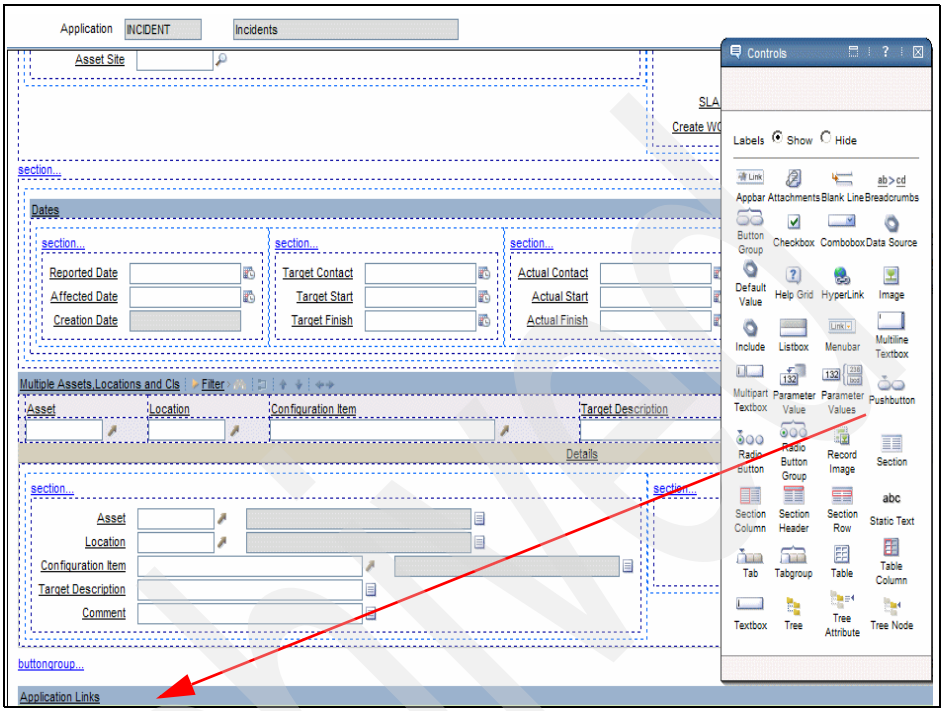


Figure 9-32 Drag the Pushbutton icon over to the new section

You should now be presented with a new push button like that shown in Figure 9-33.

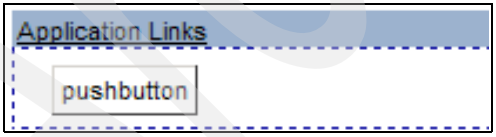


Figure 9-33 New pushbutton

7. Right-click the push button control and select **Properties**. The control properties window should now be displayed, as shown in Figure 8.

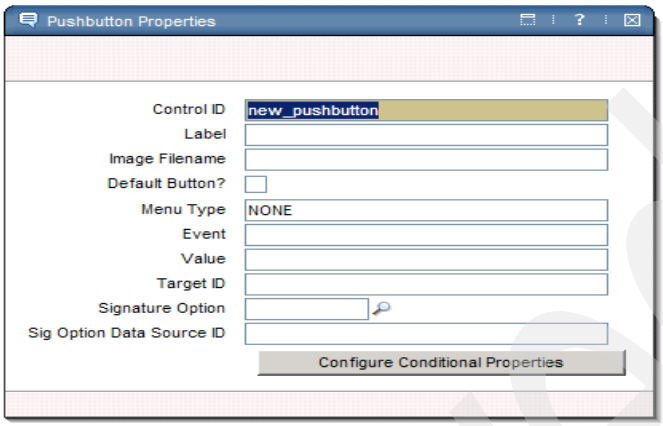


Figure 9-34 Pushbutton Properties

8. Configure the push button properties using Table 9-4 for reference.

Table 9-4 Pushbutton properties

Property	Value
Control ID	new_pushbutton
Label	Configuration Management
Image Filename	
Default Button?	
Menu Type	NONE
Event	launchexternal
Value	`iexplore http://tivsvr2:9430cdm/servlet/LICServlet?default.port=9433&graph=businessapplications`
Target ID	incident
Signature Option	
Sig Option Data Source ID	

Refer to 9.4.2, “TSBM 4.2 integration possibilities” on page 251, for an explanation of TADDM launch-in-context property values. Your push button control properties should look something like that shown in Figure 9-35.

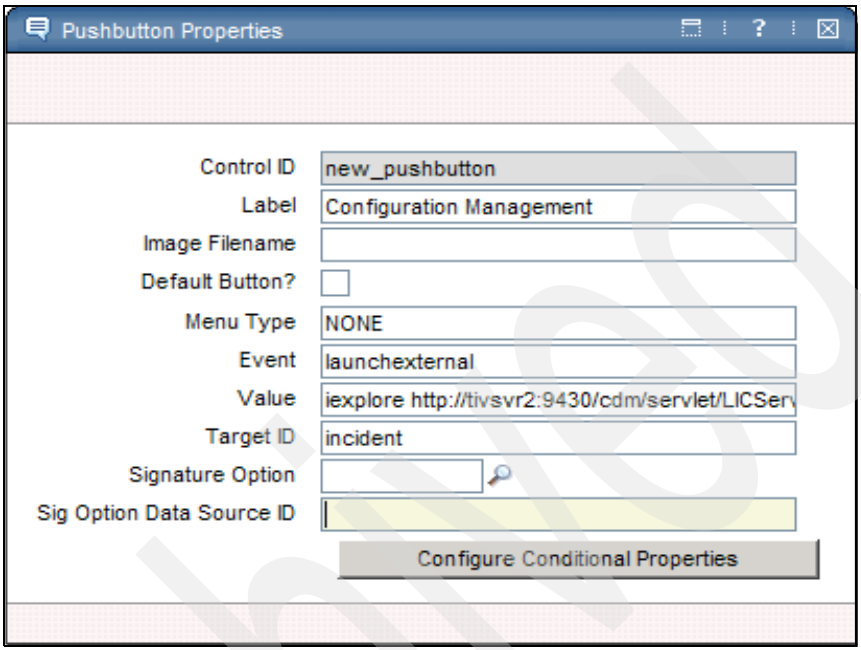


Figure 9-35 Pushbutton control properties

9. Click **Save** and return to your Start Center.

10. In the following example, shown in Figure 9-36, we have incident 1017 with the addition of a new push button control to Configuration Management. The user simply clicks the push button to launch TADDM.

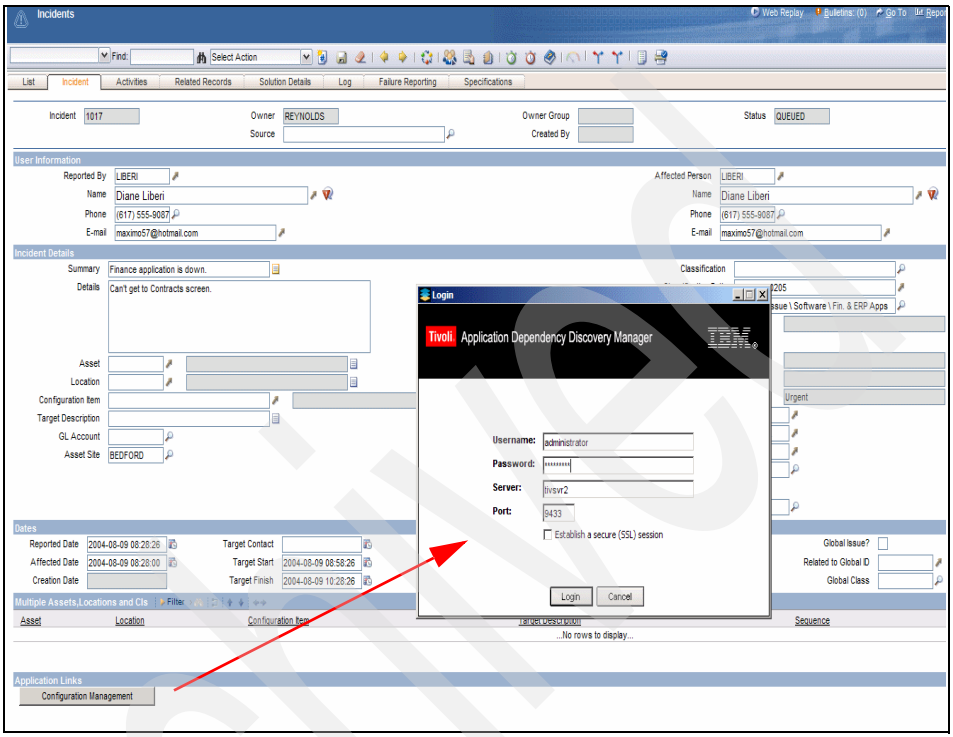


Figure 9-36 Incident with new TADDM push button control

9.6 Business impact analysis

From the Office of Government Commerce (OGC) Web site, the glossary of IT infrastructure library described at `http://www.best-management-practice.com`, *Business Impact Analysis (BIA)* is defined as:

> BIA is the Activity in Business Continuity Management that identifies Vital Business Functions and their dependencies. These dependencies may include Suppliers, people, other Business Processes, IT Services, etc.
>
> BIA defines the recovery requirements for IT services. These requirements include Recovery Time Objectives, Recovery Point Objectives and minimum Service Level Targets for each IT service.

Tivoli Business Continuity Process Manager (described in 2.2.5, “IBM Tivoli Business Continuity Process Manager (TBCPM) V7.1 overview” on page 47) can help you implement business impact analysis. Refer to this section for information about this tool and how it can help in this area of Service Management.

BIA is well documented in the following ITIL v3 set of books:

- Service Strategy ITIL V3
- Service Design, ITIL, Version 3
- Service Operation ITIL, Version 3
- Continual Service Improvement

9.6.1 Service Strategy

The Service Strategy ITIL V3 book provides in-depth guidance on how to design, develop, and implement Service Management. The aim is to strengthen an organization’s capability and also turn Service Management into a strategic asset.

BIA is well documented in the “Technology and Strategy” section of the book, and both the “Financial Management” and “Return on Investment” sections of the “Service Economics” chapter.

The book describes how to recognize an organization’s most critical services and analyze the severity of outages that can then be translated into financial value. As a result of understanding these services, together with associated operational risks, the information gathered can help improve the performance of business operations by enabling better BIA decisions about the following processes:

- Incident Management
- Problem Management
- Change Management
- Release Management

9.6.2 Service Design

As the title suggests, the *Service Design, ITIL, Version 3* book aims to help organizations design and structure IT services so that they are correctly aligned to underpinning business processes. BIA is an important step in this process and it is part of the service implementation design for business continuity management.

We strongly recommend that a BIA is conducted as part of the service design or when a change to a service is required. Doing so helps define:

- Which are the most critical IT services?
- What constitutes a major outage of these IT services?
- What is the impact to the business?

9.6.3 Service Operation

The *Service Operation ITIL, Version 3* book for ITIL v3 describes the IT Service Management phase that is accountable for delivering, managing, and monitoring day-to-day business activities and services.

This book explains that service operation personnel should have in place appropriate processes and tools to enable them to have an end-to-end view of the components that make up the service from a business perspective, such as:

- Threats to service (financial value)
- Service failures
- Quality of service
- Shared or interfacing processes and tools

9.6.4 Continual Service Improvement

The ITIL v3 book *Continual Service Improvement (CSI)* aims to help organizations not only measure and report on business services, but to continually improve the services being provided. This assumes that CSI is actually adopted with clearly defined goals, well documented procedures, and known roles and responsibilities.

BIA is detailed in the “Service Measurement and CSI Fundamentals” chapters of this book and its alignment with financial benefits that could be obtained through implementation of CSI.

10

BIRT reporting

This chapter provides information about Business Intelligence Reporting Tool (BIRT) Change and Configuration Management Database (CCMDB) Reporting and contains the following sections:

10.1 BIRT reporting

Change and Configuration Management Database V7.1.1 comes integrated with the Eclipse Foundation's Business Intelligence Reporting Tool. BIRT is an open source reporting system that integrates with Java/J2EE applications, such as CCMDB V7.1.1, to produce reports. BIRT utilizes XML report definitions to generate reports in PDF or HTML output. BIRT uses the data from CCMDB V7.1.1 and manages and displays it to users in a way in which they can immediately take action if necessary. That action may involve drilling down into reports for a specific problem issue or an analysis of the data for the cost for regulatory purposes.

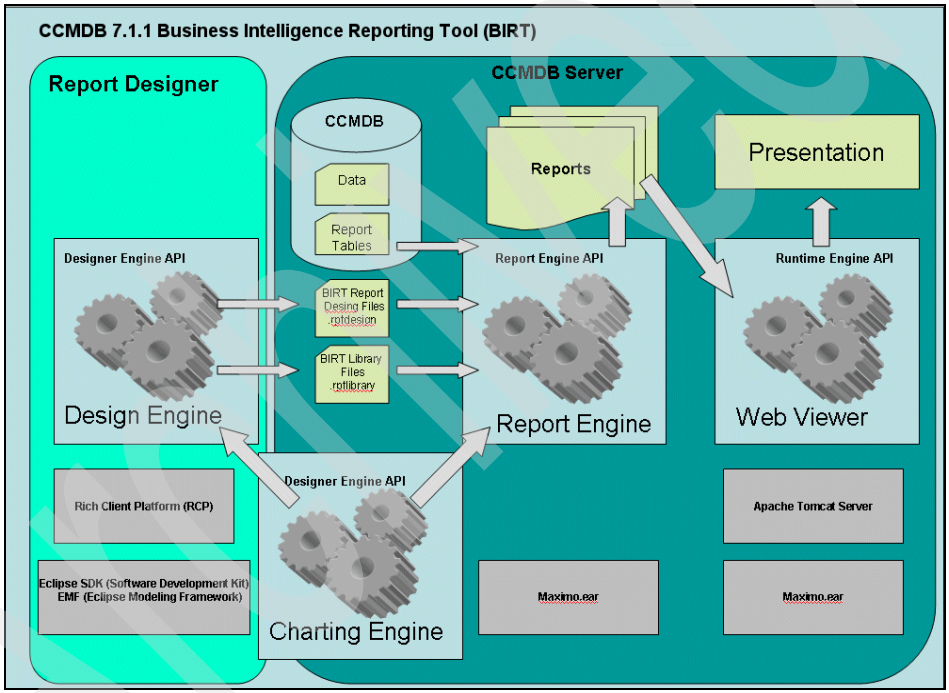


Figure 10-1 BIRT architecture

The components of BIRT are:

- BIRT Report Designer

  The BIRT Report Designer is a visual tool provided by Eclipse, as a Rich Client Platform (RCP) application. The RCP is available as a set of plug-ins that are installed on an existing Eclipse server or as an all-in-one package including Eclipse. This tool makes it easier for developers to design reports. It

must be downloaded and installed separately. It is not part of the CCMDB V7.1.1 installation.

- Design Engine

  The Design Engine is responsible for creating and modifying report designs. The created report design is stored in .rptdesign and .rptlibrary files. The Design Engine API (DEAPI) performs a wide range of low-level tasks:

  - Reads and writes design files
  - Maintains the command history for undo/redo
  - Provides a rich semantic representation of the report design
  - Provides metadata about the report object model
  - Performs property value validation
  - Notifies the application when the model changes

  BIRT Report Design files are XML files and have the extension .rptdesign. BIRT reports can contain single or multiple files. The files are categorized as either library files or resource files.

  BIRT library files are also XML files and have the extension .rptlibrary. BIRT library files can contain code that is used multiple times for items such as font type, size, page numbers, and time stamp.

  Resource files contain items such as images or external files. Resource files can be used by either report design files or library files.

  The XML of the BIRT report details which library files and resource files the report requires. In the XML file, a flag indicates whether the file is a library file. Without these files, the BIRT report does not execute.

  For further information refer to the *Change and Configuration Management Database: Report Developer Guide* at:

  `http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.ccmdb.doc_7.1/reference/ccmdb71_report_dev_guide.pdf`

- Charting Engine

  The Charting Engine is used to design and generate charts that are used by the Design Engine and Report Engine in order to deliver charts. It contains chart models and factory classes. The Charting Engine API (CEAPI) allows a developer to add charting capabilities to the application.

- Report Engine

  The Report Engine API (REAPI) generates and renders reports from a report design file. The engine supports the following operations:

  - Discover the set of parameters defined for a report.
  - Get the default values for parameters.
  - Run a report to produce HTML/paginated HTML or PDF output.
  - Fetch an image or chart for a report.
  - Export CSV.
  - Retrieve TOCs, bookmarks, and so on.

The BIRT Report Engine as part of CCMDB V7.1.1 stores its data in the CCMDB directory.

The main steps of the reporting process are shown in Figure 10-2. These are:

1. Report preparation in the BIRT Report Designer environment.
2. Report administration.
3. Report running in the IBM Service Management (ISM) environment.

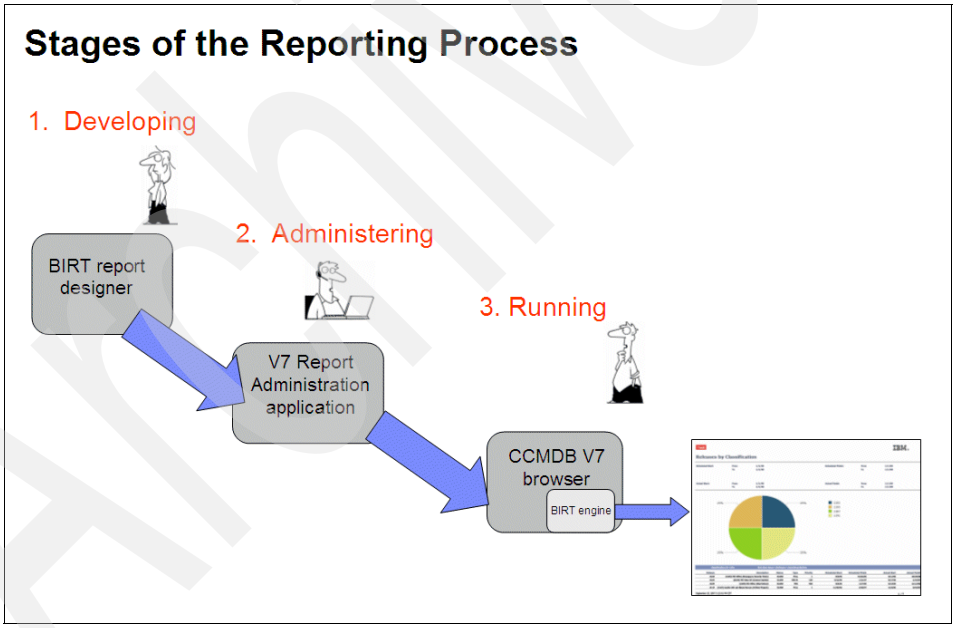


Figure 10-2 Reporting process

CCMDB reports can be accessed from menu by selecting **GoTo** → **Administration** → **Reporting** → **Report Administration**. See Figure 10-3.

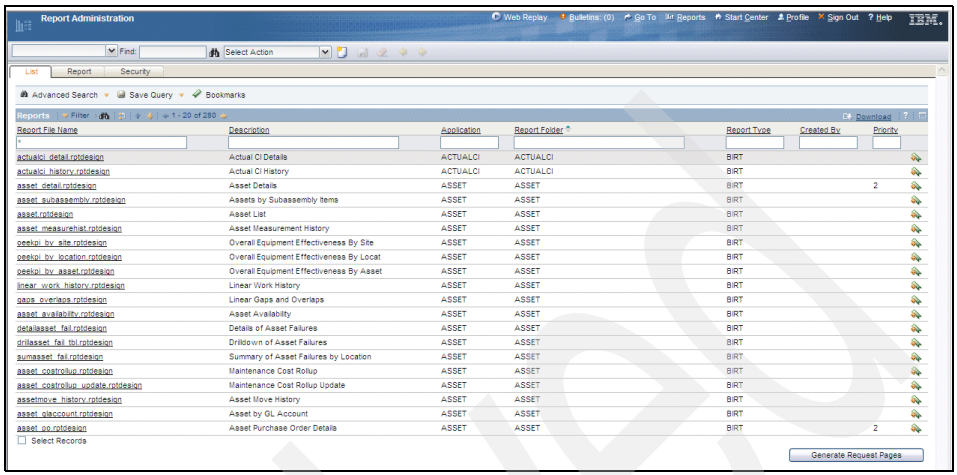


Figure 10-3 CCMDB 7.1 Report Administration panel

BIRT Report Administration

The BIRT Report Administration is integrated into CCMDB V7.1.1. As the report administrator, you can specify the following for users:

- The availability of reports and how they open, run, and print
- The appearance of report titles and headings
- Report security settings

The following figures provide a short introduction to the report administration application. The look and feel is like all other applications launched from the Start Center. There are two ways to open the Report Administration application. The first way is for the initial administration, while the second way can be used when the reports are already defined.

The first way is to start the Report Administration application from the Start Center by selecting **GoTo** → **Administration** → **Reporting** → **Report Administration**, as shown in Figure 10-4.

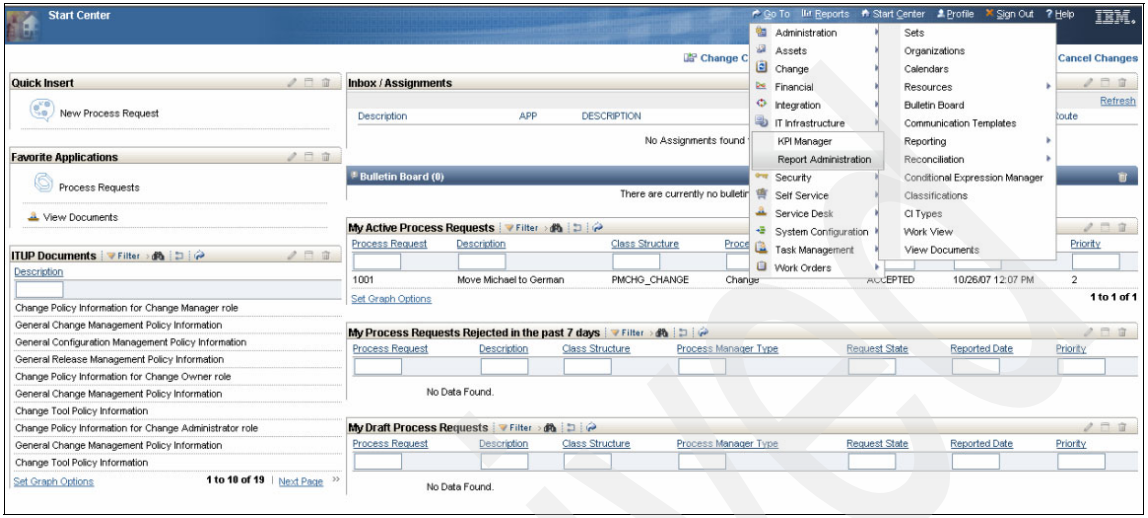


Figure 10-4 Accessing the Report Administration application through GoTo

The second way is to start the Report Administration application from the Start Center by selecting **Reports** → **Administration** → **Reporting** → **Report Administration**, as shown in Figure 10-5.

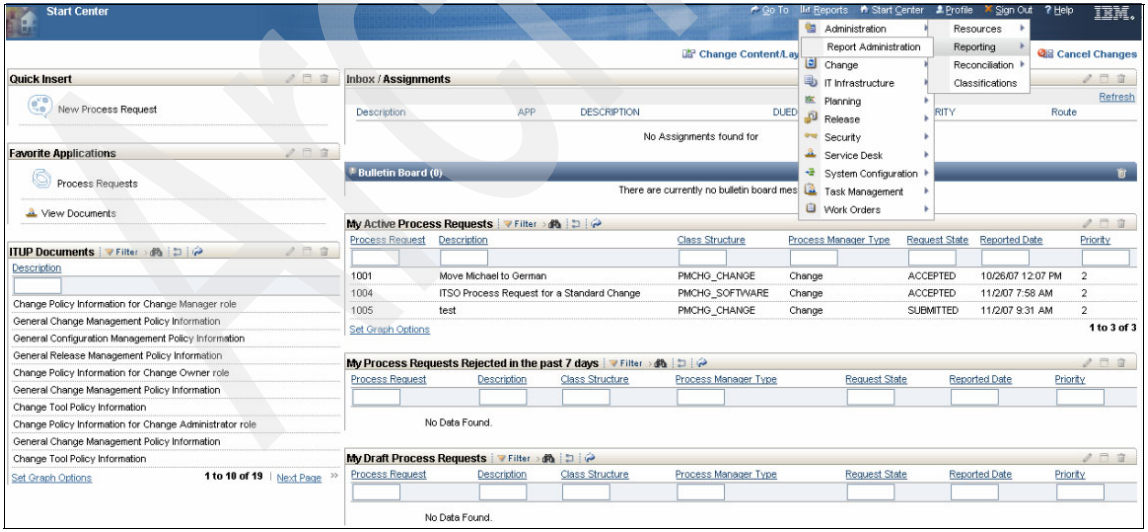


Figure 10-5 Accessing the Report Administration application through the Reports option

The Report Administration application has the following tabs:

- List: List all existing reports (Figure 10-6)
- Reports: Details of a selected report (Figure 10-7).
- Security: Set and view report and application security (Figure 10-8 on page 282).
- Labels: Change report labels and settings, like label key and label value (Figure 10-9 on page 282).

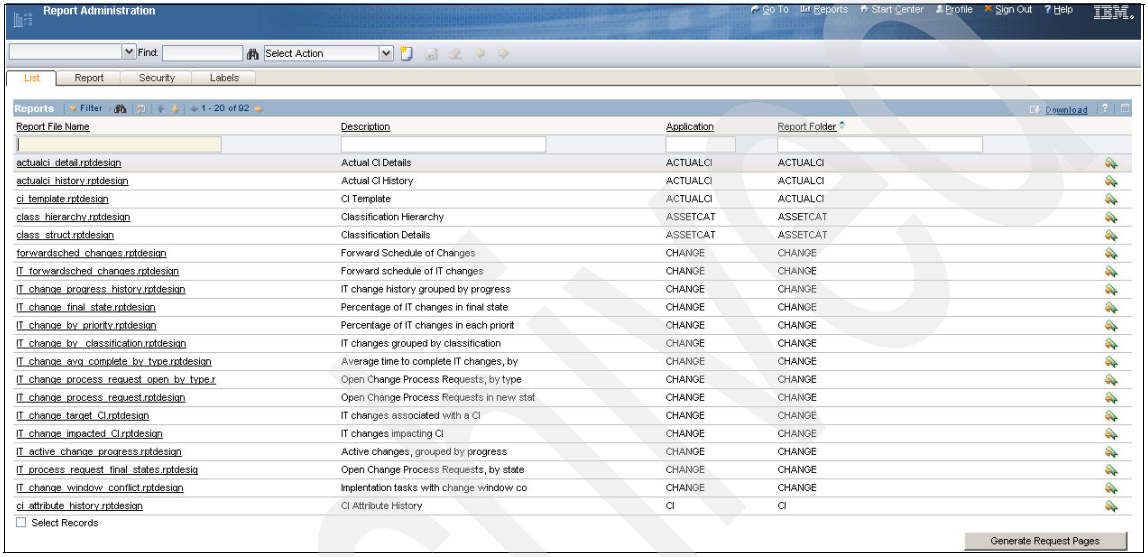


Figure 10-6 List of reports

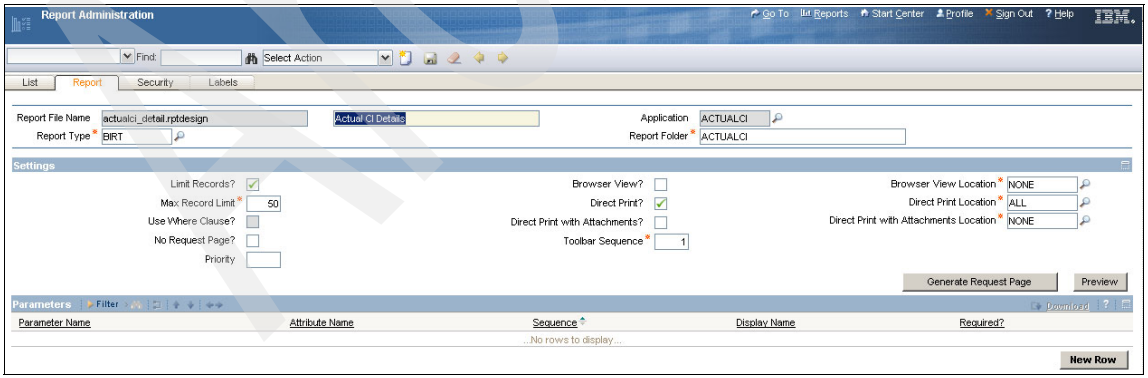


Figure 10-7 Report details

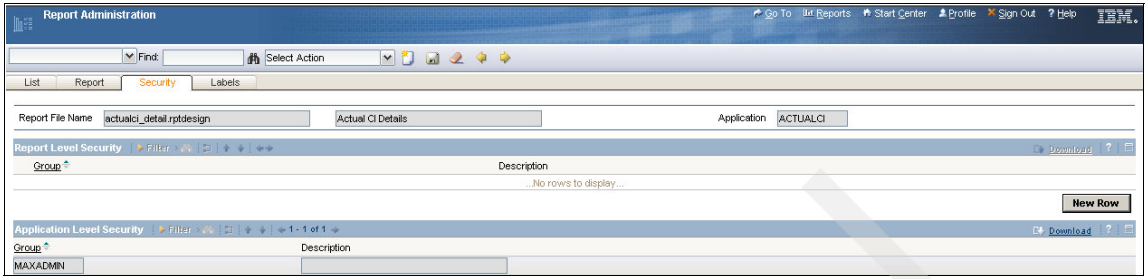


Figure 10-8 Report security

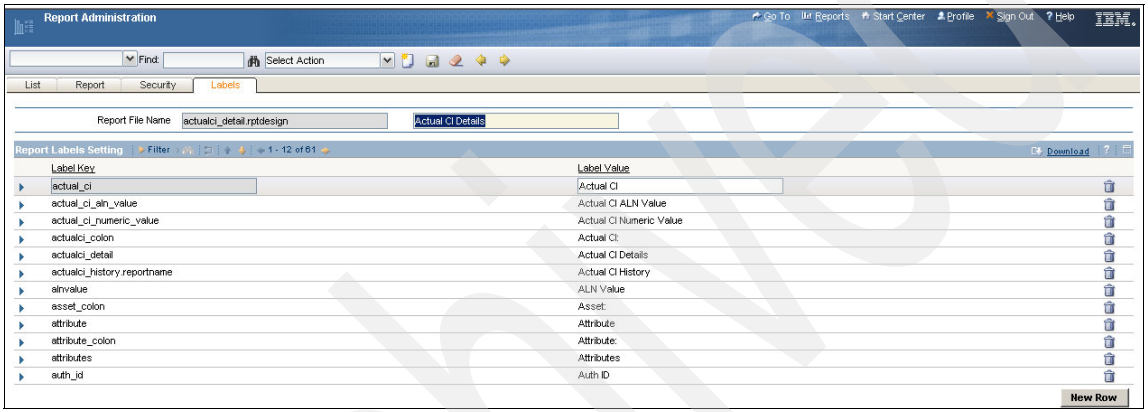


Figure 10-9 Report labels

For the List tab, the select action only offers general administration tasks like:

- View scheduled reports.
- Set application security.
- View group security.
- View .ibrarl files.
- Run reports.

Once a specific report is selected, additional select action items are available:

- Import report.
- Import library file.
- View report dependencies.
- Add to bookmark.
- Duplicate report.
- Delete report.

The following sections discuss the more complex select actions in more detail.

View scheduled reports

The View Scheduled Reports dialog box lets you manage scheduled report jobs. You can view the report, load, and delete scheduled report jobs as necessary, as shown in Figure 10-10.

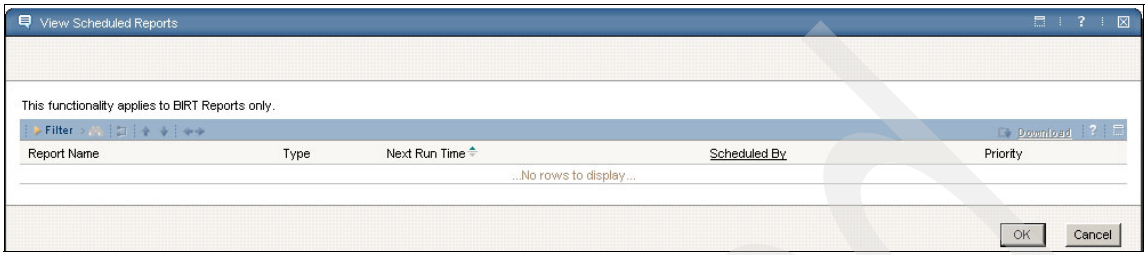


Figure 10-10 View Scheduled Reports

Set application security

The application security settings let you 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 for each individual application for new or customized reports, as shown in Figure 10-11.

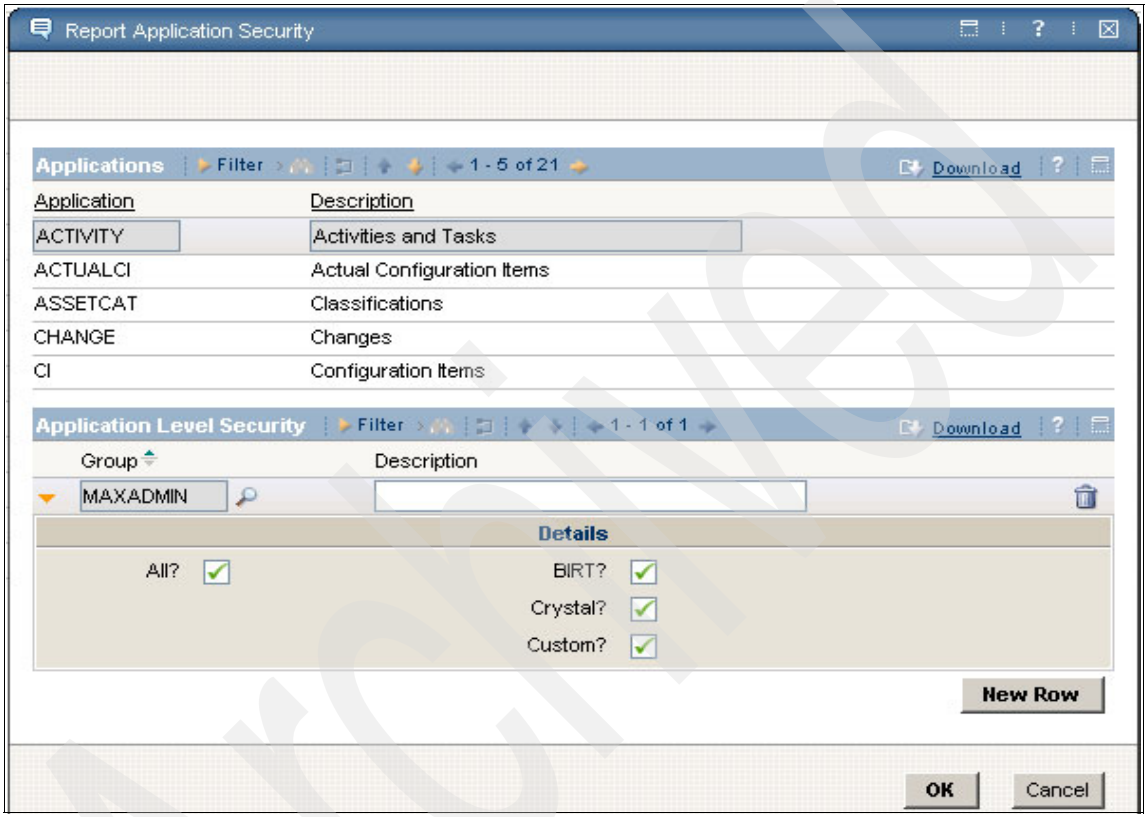


Figure 10-11 Report Application Security

The two different ways of setting security for all reports for an application and for individual reports are shown in Figure 10-12 and Figure 10-13.

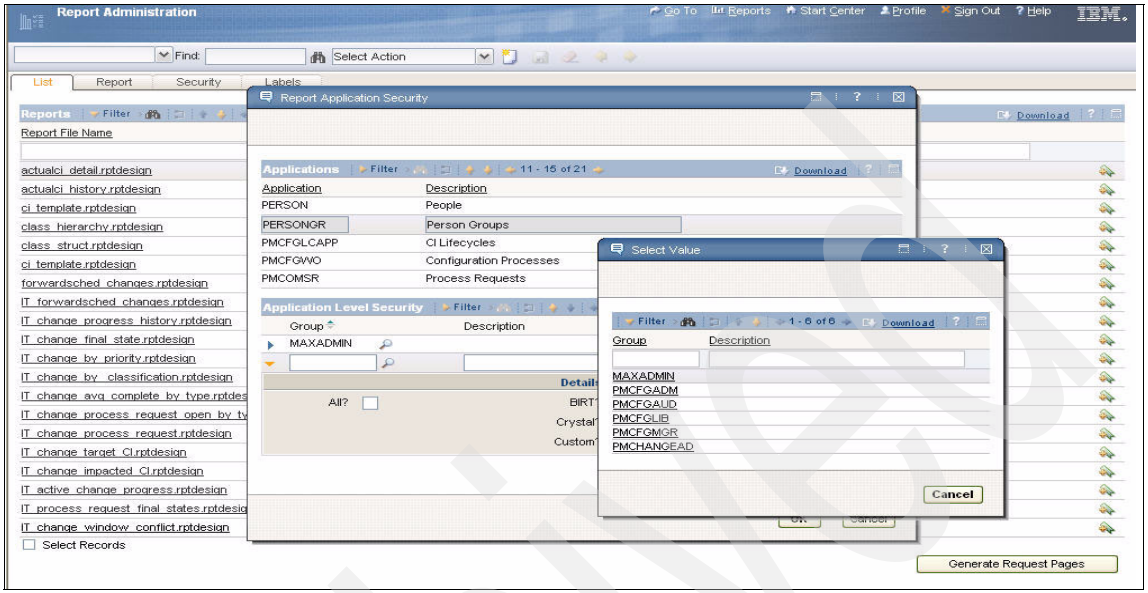


Figure 10-12 Application security settings

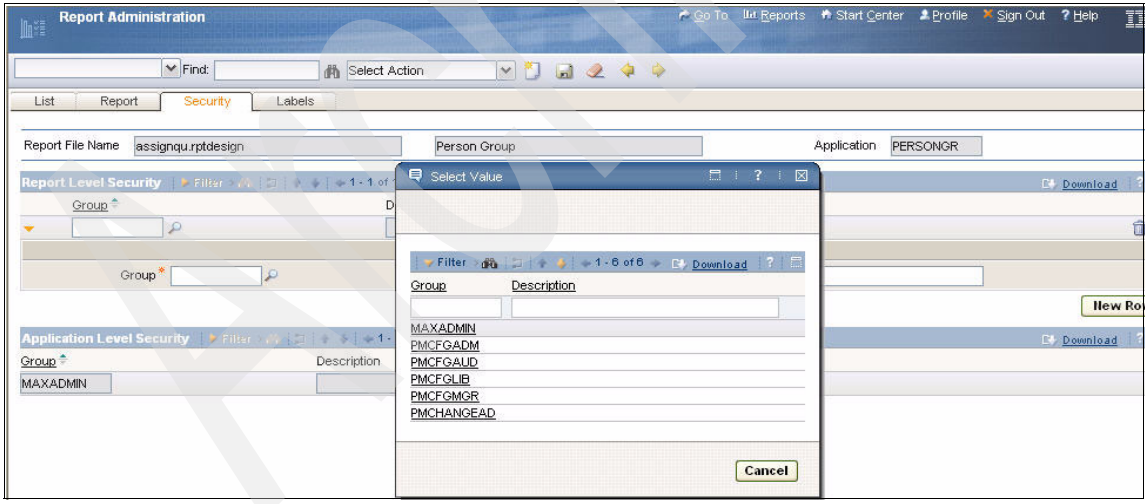


Figure 10-13 Individual security settings

View group security

It is possible to manage report security for a group through the Report Administration application. The MAXADMIN group has access to registered out-of-the-box reports. It is necessary to set up application access for other groups individually, as shown in Figure 10-14.

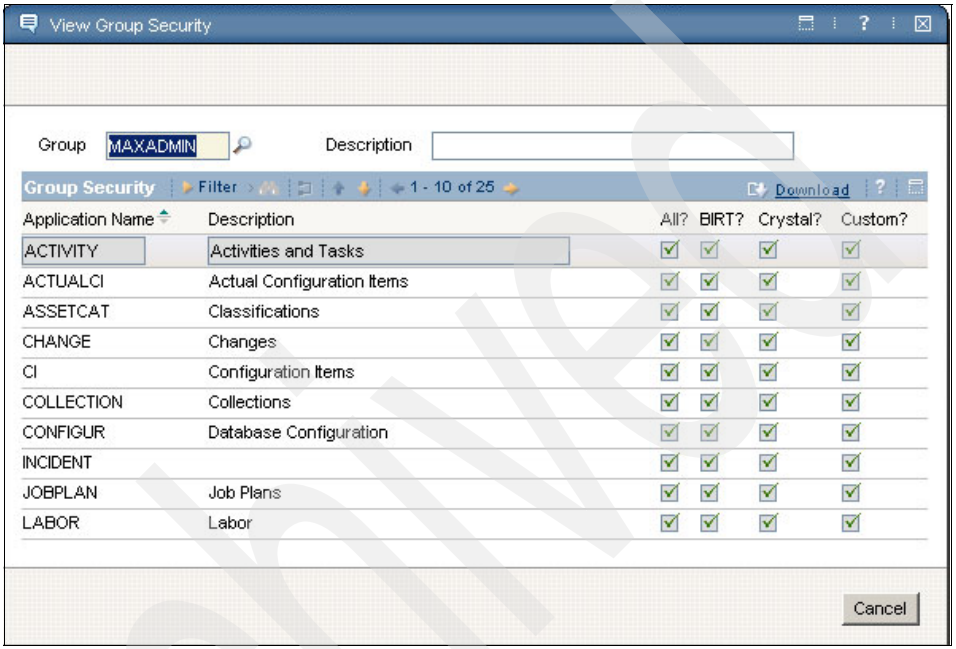


Figure 10-14 View Group Security

View library files

Use the view library file action to determine whether the libraries that you need for a report already exist in the database, as shown in Figure 10-15.

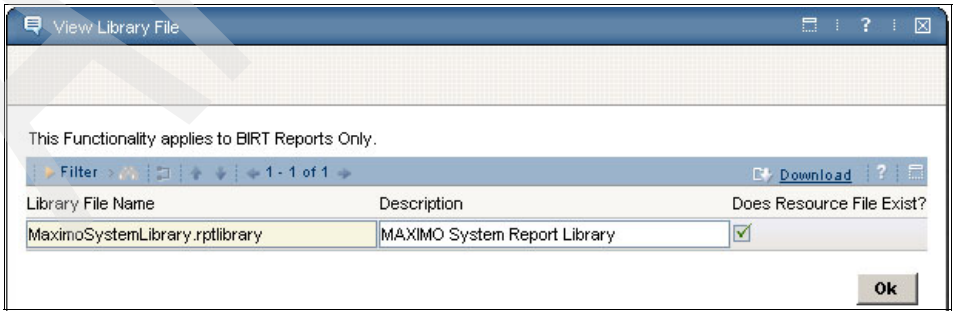


Figure 10-15 View Library Files

Import report

Perform this action to add a new report to your database or bring an updated version of an existing report into your database, as shown in Figure 10-16.

Before you import the report design file, you must import any associated library files.

This action is only available from the Report tab for the following reasons:

- If the report is new, you use the Report tab to add the report to the Report Administration application and then import the report to the database.
- If the report already exists, you import the report from the Report tab to be certain that you choose a correct combination of report design file and application name.

To add multiple design files, use the **`importreport.cmd`** command.

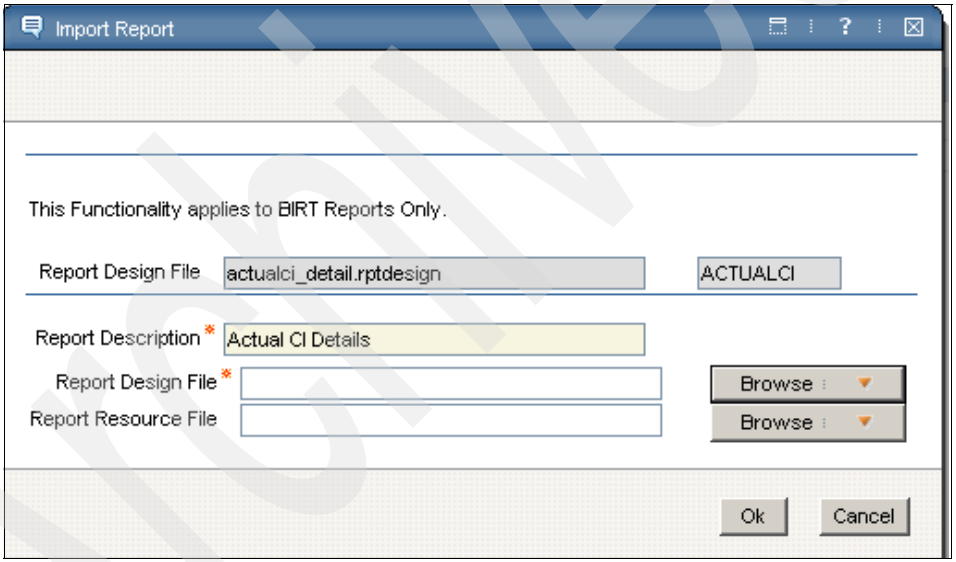


Figure 10-16 Import Report

Import library file

Library files contain components that you can use in one or more report designs to provide consistent behavior and performance. Library files are useful when many reports use the same component multiple times. Use the Report Administration application to import a report library file into the database, as shown in Figure 10-17. You import a library file before you import the corresponding reports. In the Report Resource File field, enter the location of any resource files. Resource files contain items such as images or external files. This field is optional.

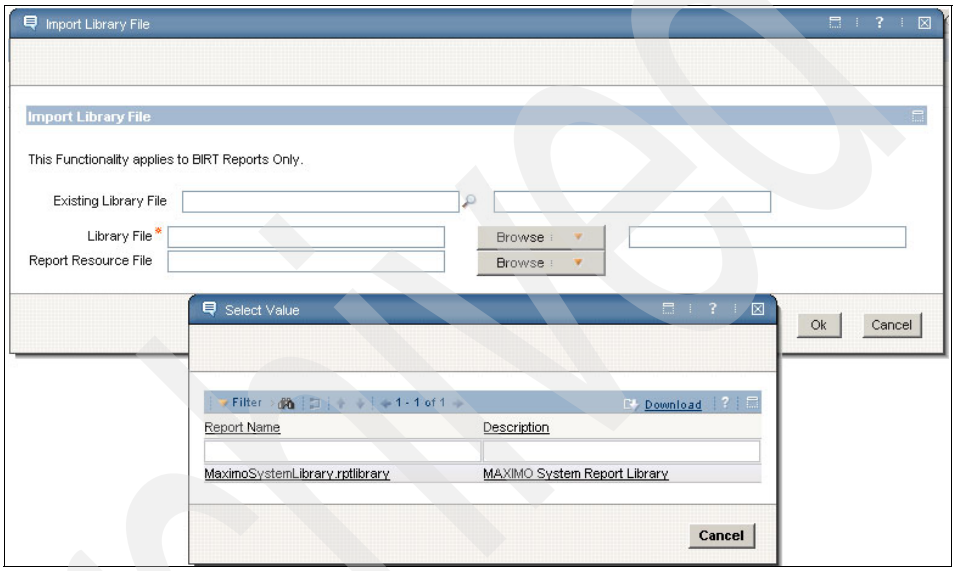


Figure 10-17 Import resources

View report dependencies

Use the view report dependencies action to view the libraries that a report design file requires. For each report library, you also can view dependent library files and check for any resource files. View report dependencies for BIRT reports, as shown in Figure 10-18.

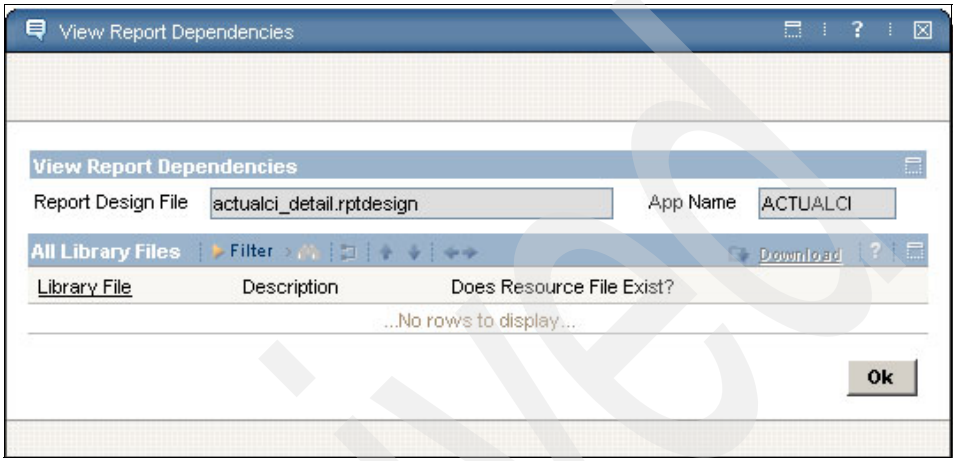


Figure 10-18 View Report Dependencies

Duplicate report

Among the reasons for duplicating a report are:

- You create a cloned application. You duplicate the report and save the duplicate to the cloned application.
- You want to register (add) a report to multiple related applications. You duplicate the report and save it to the related applications.

Delete report

When you delete a report, you remove the report and its associated files from the database. You also remove any scheduled activities for the report.

10.2 Report planning considerations

Starting with Report Design, some consideration must be taken. For example, some of the initial information required includes:

- The name of the requester
- Template type
- The need for any graphs, such as pie or bar graphs
- The file name of the existing report if this specification is a modification request
- The required title for the report

This ensures that report requesters have researched their needs. Gathering this information could make apparent whether an existing report satisfies the reporting needs.

Then we recommend designing a report to capture:

- The necessary parameter required at run time
- User input needed for the report to be executed

This information should also include the:

- Maximum number of records to be retrieved
- Mode for running the report
- User-supplied attributes at the time of running the report

During the process of report definition it is necessary to determine exactly what information is required for the report. The design specification supports this process by:

- Ensuring that the report is necessary
- Confirming the information that is to be retrieved from the system tables
- Determining the required layout of the results

10.2.1 SQL query

Report creation requires the definition of required queries. This process is best done using a database tool or SQL generation plug-in. BIRT does not provide any debugging or verification process for SQL queries. At the same time it is difficult to determine issues with queries if they are built using the BIRT Designer, and an external tool could provide a troubleshooting environment.

It is useful during the SQL query definition to determine the product's data type (maxtypes) of the fields that will be used in the SQL query.

It is possible to query the MAXATTRIBUTE object directly by selecting attributename, maxtype from maxattribute, where objectname = 'OBJECTNAME' order by attributname.

Or it is possible use the product's database configuration application to look up the maxtypes using the Type field on the Attribute tab. This information is required for the output columns and for the fetch method.

It is necessary to verify in the queries the list of columns used and the provided joins. If the joins are not provided the reference will be the information from the MAXRELATIONSHIP table.

For the database objects that have data (are not null) on MAXATTRIBUTE.PRIMARYKEYCOLSEQ, it is necessary to verify the MAXATTRIBUTE entries. They might be needed for the joins.

We recommend verifying that the database objects are at the SITE, ORG, or SYSTEM level in MAXOBJECT, because they might affect the SQL joins. We also recommend verifying that group by and order by exist and are correct.

The queries should be ANSI and JDBC™ and capable to run against all databases. Database-specific queries should be used only in specific conditions were generic SQL will not work as required.

10.2.2 Computed fields

It is possible to use a computed field to concatenate values that are in other fields, for example, creating a full_address_info field that includes the address_data and phone_data fields from a file.

It is possible do calculations from other fields, for example, calculating the extended cost of an order using the number of items ordered multiplied by the cost of each item.

It is possible to use JavaScript™ or built-in functionality of BIRT to define computed fields.

Defining the computed field in the data set has several advantages instead of defining it directly in the report layout. In the Data Set Editor it is possible to test the results of the calculation using the preview results. The computed field is available to lists, charts, and tables that uses the data set.

BIRT Report Designer calculates the computed value only once, without considering how many times the computed field is used in the report.

If the computed field is defined in the report layout, each instance is calculated every time that it occurs in the report.

10.2.3 Database objects attributes

Running a report about the attribute of the main database object for an application can be considered implicit to that application.

There are related database objects and their attributes that are defined through MAXRELATIONSHIPS or additional columns that are defined through joins of SQL database table. These are explicitly defined. Implicit parameters will be included in the where parameter and do not need to be explicitly included in the report SQL. Explicitly parameters will be passed to the report in a comma-delimited string and can contain operators. The values must be parsed before being included in the report SQL.

If there are explicit parameters that must be included in the SQL (and are not included in the Where clause), these parameters do not have to be included directly. Instead, pass then through the MXReportSQLFormat.createParamWhereClause method.

We recommend verifying each parameter against the specification. All parameters should be strings except for dates. Only date parameters can have the allow null values selected.

10.2.4 Steps to create a report

When a report in BIRT is created it is necessary to specify the query in the open method of the data set. When the query is working in the open method, update the fetch method to retrieve the data in a format that the BIRT report engine understands. This process requires mapping the table data to supported data types in the BIRT Designer. The fetch method does this mapping.

After table data is mapped correctly to supported data types, the fields become available in the Layout Designer to populate the report. Labels and other report elements are placed in the report layout to get the required design.

When the report is created it is possible to run it not only in the CCMDB environment but within the BIRT Report Designer, too. This functionality tests all phases of the report, from query to the populating report layout. If problems arise during the BIRT Report Designer phase its possible to fix them directly in the

designer phase before importing them in the CCMDB environment. To integrate the report into the product it is necessary to create/update the applicable report.xml files, create/update the applicable *.properties file, then import, register the report, and run it inside the CCMDB environment.

10.3 TADDM reporting scenario

A possible scenario considering an enterprise with high CI numbers is to be able to get reports not only from Authorized CI database but also the discovery database (discovered by the TADDM component of CCMDB). The reason for this is that during the promotion phase, only the required fields and relationship of CIs are promoted from the discovery data space to the actual, then finally to the authorized data space. In this way the Authorized CI database contains only the data required, for example, for the CCMDB change process, avoiding database size increases with information that is not required by CCMDB processes.

But considering that BIRT is an Eclipse-based open source reporting system based on Java and J2EE, we could use BIRT to run report directly against the TADDM database (or discovery data space).

Important: Currently, this scenario can be run from the BIRT Designer workstation, but cannot be run from the CCMDB server because it points to the Discovered CIs database, not the Authorized CIs database. In a future release of CCMDB, it is planned to add the capability to run the Report Engine that is installed with CCMDB to be able to run against an external database.

As previously explained, BIRT report has two main components:

- A Report Designer based on Eclipse
- A runtime component that can be add to the application server

Important: The reporting environment described for TADDM V7.1.2 is based on information about a product that at the time the book was written is not in general availability. This means that when the official version will be available there may be differences from what has been described in this chapter.

Next TADDM release V7.1.2 will integrate BIRT runtime V2.2.1 in the Domain Manager GUI. The integration will include some reports out of the box, but it will allow new reports defined with the BIRT Designer to run in the TADDM BIRT run time.

BIRT supports a variety of ways to get data. The relevant ones that will be available with TADDM are:

- JBDC: This type of data source can be used to extract data directly from TADDM DB views.
- XML: This type of data source can be used to extract data from XML files obtained using the TADDM API program.
- Scripted: You can use a scripted data source to use the TADDM Java API directly to obtain data.
- ModelBased JDBC: This is a custom data source that is facilitated by TADDM to enable you to design reports using the TADDM model schema, and then they are converted under the covers to the appropriate underlying JDBC data source.

The out-of-box report provided with the TADDM V7.1.2 run time can be used as examples for writing new reports.

To run these reports it is necessary to go on the Analytics tab under Domain Manager, then select the **BIRT Report** tab. You will see the list of the sample reports that are available. These reports can be downloaded from the server to the user's local machine for editing.

The location of a sample report on the TADDM V7.1.2 server is COLLATION_HOME/deploy-tomcat/birt-viewer/WEB-INF/report.

Note: The reports have the extension .rptdesign.

Once the logon to the domain manager is executed, going to the domain, it is available as list of predeployed reports and functional buttons ADD, DELETE, RUN, DOWNLOAD:

- ADD: Deploys a new report that has been designed. If you selecting add a pop-up appears asking for the location of the .rptdesign file, along with a name and description for the report. This adds the new report to the TADDM BIRT runtime engine.
- DELETE: Deletes a report design file that has been deployed onto the TADDM BIRT runtime engine.
- RUN: Runs the report that has been deployed to the TADDM BIRT runtime engine. Selecting and running the report will open a new browser window with the report selected to run.
- DOWNLOAD: Downloads a deployed report design file. When a report is selected, choosing download brings up a pop-up asking where to save the file in the client machine.

Reports can be created directly against the TADDM database using the Custom or Details panel database views.

Custom views are defined in the custom-views.xml file. This file has a format very similar to the screencontent.xml file, with the addition of attributes such as the name of the view and the names of the columns in the view. This file defines views that are shipped with TADDM V7.1.2.

The Detail panel views mirror the data displayed in the Detail panel tabs in the UI.

The view definitions use left outer joins where required to prevent the loss of data. They are programmatically generated using the same kind of metadata (for example, screencontent.xml, attributenames.xml) that the Details panels use. The model-object-meta.xml file is used for metadata about the model objects. The details panel and custom views use another set of views (known as the building block views). There is one building block view per model object class that can be persisted. Each view is limited to the rows and columns that relate to the relevant model object class (and its subclasses). The building block views serve as a layer of abstraction over the real database tables.

Since the views are programmatically generated, they can be kept in sync with the model when it changes from release to release. The views are documented in the DDL scripts in dist/etc/views. For example:

- create_building_block_views_db2.sql
- create_custom_views_db2.sql
- create_detail_panel_views_db2.sql

The file dist/etc/views/detail_panel_views.txt shows the mapping from the Detail panel tabs to view names. To follow the mapping:

1. Look in discovered components.
2. Choose **List/Search** from the first drop-down
3. Choose a component from the Components drop-down (for example, Linux Computer System).
4. Select a system, right-click, and choose **Show Details**. You see the first tab (usually named General) with its fields populated with data.

5. Search in the detail_panel_views.txt file (in etc/views) for Linux Computer System as a layout. The content should be as shown in Example 10-1.

Example 10-1 detail_panel_views.txt

```
########## Linux Computer System...<Layout> ##########
...General.....................................<Tab Level 1>
......General..................................... .........<TabData>
......LinuxUnitaryComputerSystem.General...... ..........<Content>
.........DP_LINUX_COMP_GENERAL_V.............. ..............<View>
```

This tells that the DP_LINUX_COMP_GENERAL_V view has all the fields that are defined in the General tab for the Linux Computer System.

Custom view XML

The value added to the custom-views.xml should follow the rules given in Example 10-2.

Example 10-2 custom-views.xml

```
Entity: view ..................... Contains fields
  Attribute: className ........... The model object class name
  Attribute: viewName ............ The name of the view. Should
                                   start with "CM_" and end with
                                   "_V".  Avoid names that are
                                    already in use. Maximum length
                                      is 30 characters.
  Attribute: includePrimaryKeys ... Include primary keys as columns.
                                   True or false. Set to true if
                                   this view is required to join
                                   with other views.
Entity: field .................... Contains plains and/or nesteds
Entity: nested ................... Contains nesteds and/or plains
  Attribute: className ........... The model object class name of
                                   the nested
  Attribute: fieldName ........... The field name of the nested
Entity: nested ................... Contains nesteds and/or plains
  Attribute: className ........... The model object class name of
                                   the nested
  Attribute: fieldName ........... The field name of the nested
Entity: plain .................... No contained entities
  Attribute: fieldName ........... The field name of the plain
  Attribute: nameInView .......... The name of the column in the
                                   view. Maximum length is
```

```
                                  30 characters. Avoid DB2/Oracle
                                  reserved words.
  Attribute: displayType ......... Optional. One of:
                                  speed .......... value in MHz
                                  memory ......... value in B/MB/GB
                                  mBytes ......... value in Mbytes
                                  date ........... Timestamp in
                                  YYYY-MM-DD HH24:MI:SS format -
                                  Used for fields that contain
                                  epoch time in milliseconds.
                                  networkSpeed ... value in
                                  Mbits/sec
                                  StorageGBytes .. value in GB
```

User-defined views

Views can be created outside the TADDM build by running the user_views.sh/bat scripts in support/bin. By default, the views are defined in support/bin/user-views.xml. This file uses the same format as custom-views.xml. A typical use case might be:

1. A customer runs a report, then sees a need for more attributes.
2. A customer TADDM expert determines which model attributes are needed.
3. The view definition for the report in etc/views/custom-views.xml is copied to support/bin/user-views.xml, then modified to include the extra attributes. The view is renamed so that the original view is not overwritten.
4. The user runs **`user-views.sh scripts`** to create the scripts.
5. The user runs **`user-views.sh recreate`** to create the new view.

10.3.1 Simple report example

This example explains how create a simple report in BIRT Designer using one of the Details panel views. This report will display the host name of any Linux system that has more than one CPU, listed by CPU number (highest to lowest).

This example assumes that BIRT Designer is installed on a Windows system, and TADDM is installed on a Linux system. Some files must be copied from the Linux to the Windows system.

1. The first task is to locate the Details panel with the required information. Open the TADDM UI, choose **List/Search** in the drop-down under Discovered Components, choose **Linux Computer System** in the Component drop-down, right-click one of the systems, and choose **Show Details**.
2. In the first tab, General, we see Number of Logical CPUs. We also see the host name, which is the other required attribute. What we need is to find the Detail panel view for this tab. Open BIRT Designer. Choose a new/existing workspace directory, for example, c:\birt\work.
3. Create a report project. From the menu, select **Window** → **Perspective** → **Report Design** → **File** → **New** → **Project** → **Business Intelligence and Reporting Tools** → **Report Project** → **Next**. For project name, enter `reportproject1` and select **Finish**.
4. Create a report. In the Navigator view, right-click **reportproject1** and select **New** → **Report**. Enter `linuxcpu.rptdesign` as the file name and click **Finish**.
5. Create a data source to point to the TADDM Database. In the Data Explorer view, right-click **Data Sources** and select **New Data Source**. Choose **JDBC Data Source** from the list of data source types, enter `DataSource1` as the data source name, and click **Next**.
6. In the Driver Class drop-down select **com.ibm.db2.jcc.DB2Driver (V3.3)** if it is available. If it is not, choose **Manage Drivers** → **Add**, locate the db2 jcc driver jar on your disk (you can copy this from the TADDM Server in dist/lib/jdbc), and click **OK**.
7. For URL, user name, password, enter the values used to connect to the TADDM database (these can be copied from the JDBC properties in dist/etc/collation.properties on the TADDM Server). Example 10-3 provides example values for DB2.

Example 10-3 Connection parameters for DB2

```
com.collation.db.user=<my-user>
com.collation.db.password=<my-password>
com.collation.db.url=jdbc:db2://<my-taddm-host>:60000/cmdb
```

8. Select **Test Connection**, and you should get a Connection Successful message. Select **Finish**.
9. Create a data set to point to the required view. In the Data Explorer view, right-click **Data Sets** and select **New Data Set**. Enter `Data Set1` for the data set name, select **Data Source1** as the data source, select **Sql Select Query** as the data set type, and click **Next**.

10. In the Query panel, select **View** in the Type drop-down, enter `DP_LINUX` as the filter name, and click **Apply Filter**. Note that all the detail panel view names start with DP_ and then the name of the relevant component.

11. Select the plus sign (+) next to the user (same as the user name above). It should be visible the view names given in Example 10-4.

Example 10-4 View names

```
DP_LINUX_COMP_GENERAL_V
DP_LINUX_COMP_OS_V
DP_LINUX_COMP_PACKAGES_V
```

In this case the first name looks like the correct one, since is relates to a tab called General on a component called Linux Computer System. We can verify this by previewing the data and checking that it matches the data in the Details panel tab.

12. Click **Next** in the right-hand panel, then double-click **DP_LINUX_COMP_GENERAL_V**. This should copy the view name into the pane. Click **Next** to and enter and asterisk (*). The text should now look like this (user name may differ):

```
select * from DB2ADMIN.DP_LINUX_COMP_GENERAL_V
```

13. Select **Finish**. In the Edit Data Set panel, click **Preview Results**. Locate the same Linux system that was displayed in the Details panel tab and verify that the same data is displayed here.

14. To clean up the column names in the report, select **Output Columns**. Select **FULLY_QUALIFIED_DOMAIN_NA_C1** and enter hostname in the Display Name field. Select **NUMBER_OF_LOGICAL_CPUS_C1** and enter number of CPUs in the Display Name field. Click **OK**.

15. Add content to the report design. In the Palette view, select **Label** and drag it to right-hand panel (Layout tab of linuxcpu.design tab). In the Layout tab, double-click **Label** and enter MultiProcessor Linux Systems.

16. In the Property Editor below, in Properties → General, choose **Arial** as the ont, **14 points** as the size, and **yellow** as the background color. In Properties → Padding enter 20 points for top, left, and bottom.

17. In the Palette view select **Table**, then drag it under the label that you just added. Enter `2` for number of columns, and `Data Set1` for Data Set. Click **OK**.

18. Select the **Data Explorer View**. Expand **Data Set1** and select **Hostname**, then drag it into the detail row in the first column in the table on the Layout tab. The title should be filled in automatically for the first column. Select Number of CPUs, then drag it into the detail row in the second column. Again, the title row should be filled in automatically.

19. Right-click **Data Set1** and choose **Edit**. Enter `DP_LINUX_COMP_GENERAL_V` as the filter and click **Apply Filter**. In the SQL panel, remove the asterisk (*) after the select. With the cursor one space after the word select, double-click the column that starts FULLY_QUAL, then double-click the column that starts NUMBER_OF_LOGICAL. Then clean up the SQL so that it looks like this:

```
select
FULLY_QUALIFIED_DOMAIN_NA_C1,
NUMBER_OF_LOGICAL_CPUS_C1
from
DP_LINUX_COMP_GENERAL_V
```

20. Since we need the data to be displayed only when the number of CPUs is more than 1, add this syntax:

```
WHERE NUMBER_OF_LOGICAL_CPUS_C1 > 1
```

21. Since the data should be ordered form highest to lowest based on the number of CPUs, add this syntax:

```
ORDER BY
NUMBER_OF_LOGICAL_CPUS_C1
DESC
```

Click **OK**.

22. Now we are ready to preview the report. In the Navigator view, right-click **linuxcpu.rptdesign** and select **Report** → **Run Report**. A new Report Viewer should open with the results from the report, as shown in Figure 10-19.

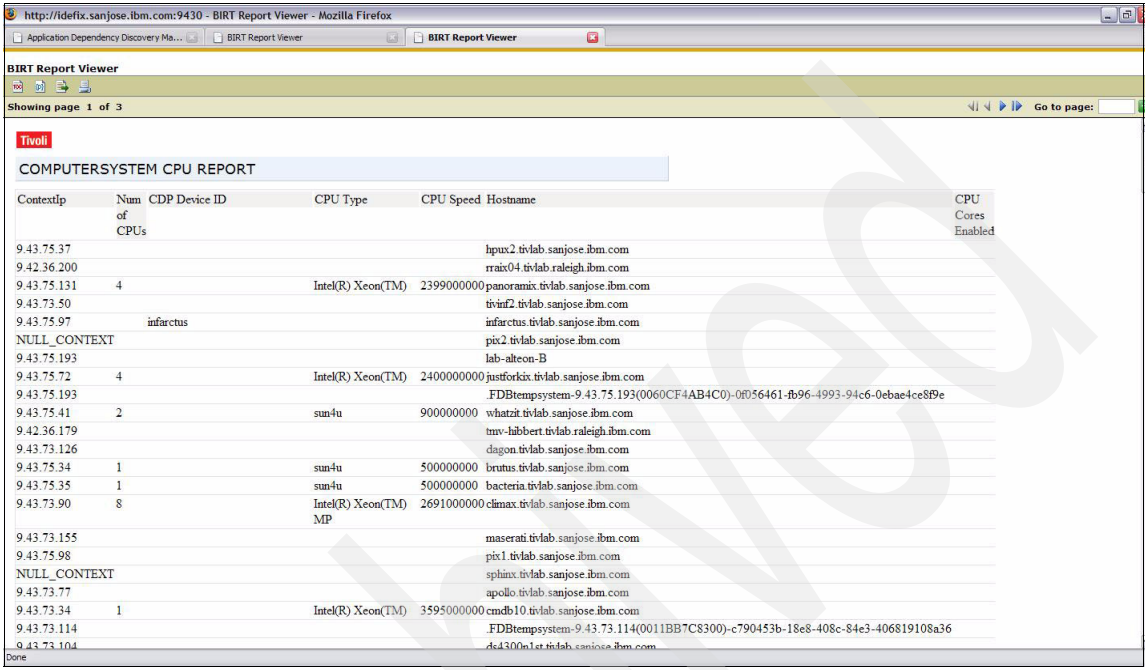


Figure 10-19 TADDM report example

10.3.2 Conclusions

We have seen that BIRT reporting is available in CCMDB v.7.1.1 for promoted CIs and in TADDM as a runtime component from the next release V7.1.2. These functionalities allow you to address the reports requirement starting from the discovery phase until the promoted CIs. The standalone runtime BIRT allows you to get reports from the external database and join these values with CI attributes (for example, connecting and reporting to the inventory database).

In a future release of CCMDB, it is planned to add the capability to run the Report Engine that is installed with CCMDB against not only the promoted CIs, but also against data on an external database.

If it is required to have a central point for reporting in the CCMDB V7.1.1, a customization can be done in CCMDB to link, for example, to PDF reporting files that are produced in runtime BIRT external to CCMDB. This allows users to see reports from CCMDB interfaces, but requires a customization to make reports

available and to update the link. This reason for this is that options cannot be changed dynamically for reports that are produced with a runtime BIRT interface that is external to the CCMDB.

A

How to design an IBM Service Management solution using Tivoli products

This appendix provides information about how to design an IBM Service Management (ISM) solution using Tivoli products.

ISM Assessment Tool

One of the biggest obstacles to implementing IT Service Management best practices is determining where to begin focusing your efforts. This assessment will help you identify and prioritize areas for improvement. IBM has created a Web site that describes the ISM Assessment Tool:

https://www14.software.ibm.com/webapp/iwm/web/reg/signup.do?lang=en_US&source=itsmat

An Information Technology Infrastructure Library (ITIL) Service Management self-assessment can be done at the following Web site:

http://www.itsmfi.org/content/self-assessment-itil-v2

This assessment will give you an idea of where you are now in terms of Service Management maturity (similar to the ITIL Service Management Maturity scale) and help you identify the gap between your current position and where you want to be in the future.

Starting points in implementing ISM: 5-entry points

How would you design an ISM solution using Tivoli software products? Here we discuss high-level design and 5-entry points method.

IBM Service Management delivers the visibility, control, and automation required to deliver quality service, manage risk and compliance, maximize return on investments, and accelerate business growth.

Depending on your specific needs and focused initiatives, IBM has identified a set of entry points for clients to begin their implementation/journey for Service Management. These entry points represent areas of pressing challenges that customers and industry practitioners (and analysts) have identified to yield values in their respective areas of focus. Briefly, the entry points are:

- Discover: Understand infrastructure and business dependencies.
- Monitor: Track infrastructure health and compliance.
- Protect: Ensure that the infrastructure is secure and resilient against threats and disasters.
- Industrialize: Streamline workflows and processes for repeatable, scalable, and consistent results.
- Integrate: Align and integrate operations and business for optimal impact.

Discover

Understand what resources are deployed, how resources are used and by whom, and how resources relate to and can impact successful business service delivery.

Infrastructure Discovery and Mapping products are Tivoli Application Dependency Discovery Manager and Change (TADDM) and Configuration Management Data Base (CCMDB).

Monitor

The following products are used to monitor all resources, events, performance, service levels, and users, providing total visibility into the business:

- Event and performance management products
  - Tivoli Netcool/OMNIbus
  - Tivoli Netcool/Impact
  - Tivoli Performance Analyzer
- SOA application performance products
  - IBM Tivoli Composite Application Monitoring for Response Time Tracking
  - Tivoli Federated Identity Manager
- Policy and regulatory control monitoring products
  - Tivoli Security Information and Event Manager
  - Tivoli zSecure
- User activity monitoring/log management products
  - Tivoli Compliance Insight Manager
  - Tivoli zSecure
- User access rights/user life cycle management products
  - Tivoli Identity Manager
  - Tivoli Access Manager
  - Tivoli Compliance Insight Manager

Protect

The following products keep applications, data, and services secure, protected from malicious or fraudulent use, and hardened against failure and catastrophe:

- Vulnerability and Threat Management products

  Tivoli Security Information and Event Manager

- Data backup, restore, and retention products
  - IBM Tivoli Storage Manager Family
  - TotalStorage Productivity Center/SSPC
  - Tivoli Continuous Data Protection for Files
  - IBM Systems Storage Archive Manager
- Business continuity/disaster recovery products
  - IBM Tivoli Storage Manager Family
  - IBM Tivoli System Automation Family
  - IBM TotalStorage Productivity Center/SSPC

Industrialize

The following products reduce costs, improve responsiveness, and reduce errors by creating automated, repeatable, consistent, and scalable task management:

- Virtualization Management products
  - IBM Tivoli Monitoring
  - Tivoli Provisioning Manager
  - OMNIbus, SAN Volume Controller
- Energy Management for green savings products
  - IBM Tivoli Monitoring
  - IBM Tivoli Usage and Accounting Management
  - IBM Maximo Spatial
  - Active Energy Manager
- Single Sign On products
  - IBM Tivoli Security Services
  - IBM Tivoli Access Manager for eSSO

- IT Asset Management products

  Tivoli Asset Mgmt for IT

- Usage Accounting products:
  - IBM Tivoli Usage and Accounting Management
  - Power Management™
  - Tivoli Asset Management for IT

Integrate

The following products align IT planning and execution to business users and stakeholders, streamline the planning process, and enable life-cycle governance:

- Business Service Management products
  - Tivoli Business Service Manager
  - Tivoli Netcool/OMNIbus
  - Tivoli Netcool/Impact
- Service Desk products

  Tivoli Service Request Manager

- Service Deployment and Lifecycle Management t products:
  - Tivoli Provisioning Manager
  - Tivoli Composite Application Monitoring

Whiteboard overview of ISM

Figure A-1 will help you understand where the IBM Service Management strategy fits into your business objectives.

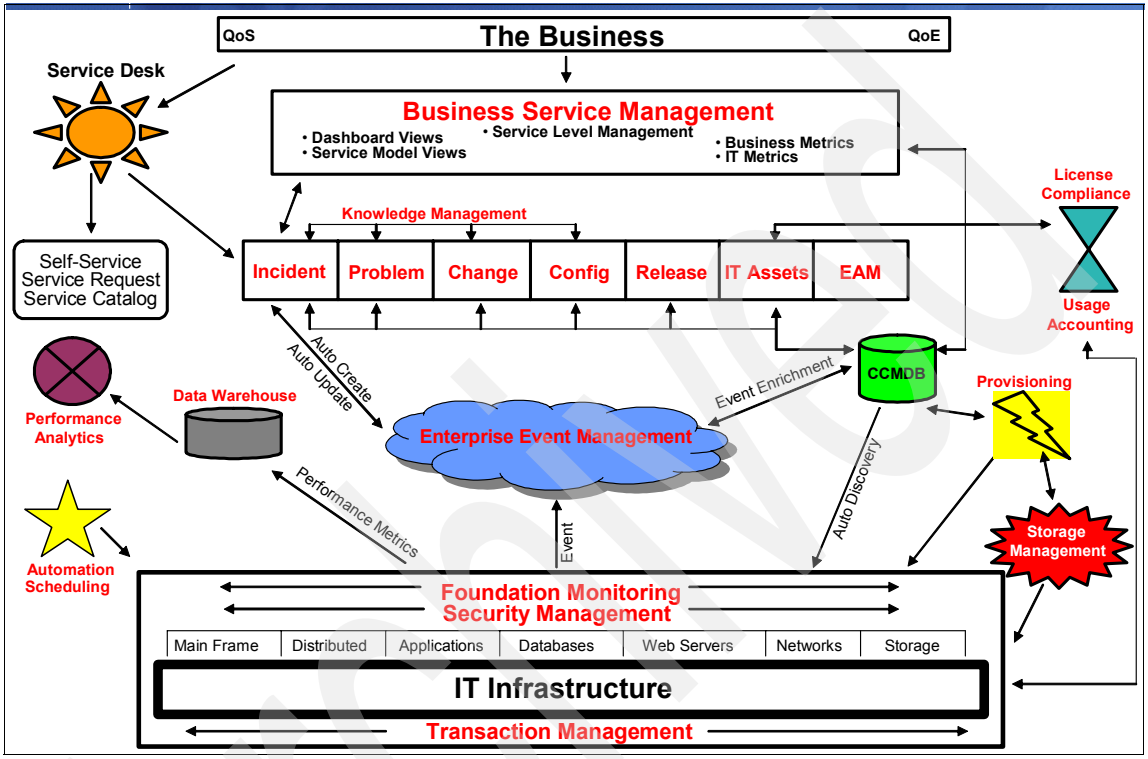


Figure A-1 Whiteboard overview of ISM

The top of the diagram shows your business objectives. Often the objectives between the lines of business and IT are very clear. They are to deliver a quality service at an agreed-upon cost and the service must be available when the LOB needs it. Depending on the service, this does not necessarily mean 24/7/365.

Step 1: The business

What do the business user, external consumer, and internal client care about? Users are not concerned with the details of the infrastructure, they just want to be able to complete specific transactions, whether it is an ATM withdrawal, placing an order, providing a client service, or delivering a product. This block illustrates the users' expectation: the quality of service being provided and the quality of the users' experience. The primary goal of IT is to ensure cost-effective quality of

service to the business. Understand what is good service to your client. Ask them how they ensure that they provide good service. Do they have SLAs in place between IT and the business? If so, how do they define SLAs? What measurements are used to determine what a good service is?

Step 2: IT infrastructure

Refer to the bottom of Figure A-1 on page 308 and the box labeled IT Infrastructure. Make sure that you understand the complexity of your own environment. Once you understand the environment you can start filling in the sections in the IT Infrastructure block. Figure A-1 on page 308 provides an example of a typical IT environment that includes networks, operating systems (MF/DS), applications, databases, storage, and so on.

Fully investigate your environment, understanding that most IT runs silos and is managed by different management teams and point solutions. Integration of silos is the most important strategy for aligning the IT organization with the business.

1. Do you monitor each of the components?
2. If so, do you use a single solution?
3. If you do not use a single solution, understand what is needed to integrate it or what is needed to fix the problem.

This is the foundation monitoring in the IT Infrastructure block. It is very importance to integrate foundation monitoring in a centralized approach to monitor the speed and performance in your environment.

Security Management in the IT infrastructure block

Do you have security compliance within the environment today?

Identify how you are managing user access in your environment today. Do you do this with an automated processes in place that allows users to manage their own accounts and perform tasks such as password resets and business application access based on policies? The value of integrated security management in your environment will reduce the risk of managing multiple products, and security can be alerted of impact to business and service to users. If you have had any recent security rated problems investigate how they happened and what you are doing to prevent any breaches in the future. If you are standardizing with a single security management solution you can put more emphasis on compliance controls around user access to company information and make it easier to address audit requirements. You must enforce the concept in regards to corporate security practices no matter how quickly you can determine the root cause of a problem. It is always better to prevent a problem from happening in

the first place. Grant users the correct access at the correct time to the correct information. Restrict everything else.

Transaction Management on the IT Infrastructure block

Understand the difference between monitoring your IT environment and monitoring business transactions. Are you able to track transactions from their origination point through the various IT components and back to the origination point? Do you use this data to troubleshoot and diagnose problems in your environment? IT organizations spend more time diagnosing problems than they do actually restoring the service back to the business. Without some level of visibility into transactions you have no way of measuring user experience.

Step 3: Enterprise Event Management

How do you manage IT events currently?

Enterprise Event Management centralizes all alerts from various event sources, filters those alerts based on priority, correlates the alerts with other alerts, de-duplicates/consolidates repeat alerts, and provides event enrichment by leveraging other disparate data sources as well as an integrated CCMDB (formerly CMDB, which stores only configuration data). Effective Enterprise Event Management reduces IT costs by minimizing the typical chaotic response to critical events that potentially impact the business. By centralizing Event Management the IT organization can gain better control over problems before they turn into outages.

Step 4: CCMDB

You should be familiar with the term CMDB, which stands for Configuration Management Database. The Tivoli products term is CCMDB, which stands for Change and Configuration Management Database. Do you currently have a CMDB in their environment? It is okay if you do. In fact, it has become an accepted approach to have many CMDBs. However, each must be able to share data by federation. The difference between a CCMDB and just a CMDB is that a CCMDB contains not only the configurations of IT assets (CIs), but also changes made to those IT assets. Why is this important? This data can be used many different ways. First, the CCMDB is a provider to many consumers. One consumer is the Enterprise Event Management component discussed earlier. In this you provided knowledge regarding event enrichment. To put this in context, when an event occurs the Enterprise Event Management process may be to leverage configuration and change data stored in a CCMDB to enrich the event. This is important because it allows the recipient of the event to make a better informed decision quicker. It automates the otherwise manual task of data

gathering. Also, another important point is that a CCMDB stores relationship information, which can also be leveraged by Enterprise Event Management to determine impact to other IT assets. You must also understand that for a CCMDB to be useful it must be populated with accurate and up-to-date information. To accomplish this it is important that a CCMDB provide auto discovery of your IT infrastructure. Not only a subset, but the entire IT infrastructure from A–Z applications, databases, and operating systems, including mainframes. The auto discovery must also be capable of detecting relationships between the IT assets.

Step 5: Service Management platform (Incident, Problem, Change, Configuration, Release, IT Assets, Enterprise Assets)

An IBM Tivoli high-level overview of the IBM Service Management platform is a single J2EE platform for which all ISM applications plug into and in some cases run directly on. By building a unified platform IBM has taken data integration to the next level. Sharing data between applications has never been easier. What are you using for Incident, Problem, Change, and Configuration and Release Management applications? These applications should be integrated at the data level and integration can be thorough third-party or service implementations. Integrating processes and data will tie into the CCMDB to share consistent data, also allowing direct integration with Enterprise Event Management in a bi-directional way. By it being bi-directional you will be able to have an automated approach to Event Management and Incident Management. In Figure A-1 on page 308 provisioning is connected to the CCMDB.

How are you currently performing system, application, and database deployments? Do you have an automated repeatable method for doing it? Is it integrated within your change, configuration, and release processes?

Automated provisioning is important because of the flexibility that it can offer within the environment. Due to the fact that technology is constantly changing, it is imperative that you understand the provisioning standard and be flexible enough to handle changes. Tivoli provisioning technologies allow you to accomplish enterprise-wide deployments of operating systems, databases, and applications and it would benefit you greatly to have a single solution that is both scalable and intuitive, not to mention cost effective.

Step 6: Service Desk, Self-service, Service Request, Service Catalog and Knowledge Management

Do you currently have a Service Desk in your environment? If yes, how is the Service Desk positioned within the organization? For example, does it have more

of an IT focus or a business focus? Does it handle requests outside or un-related to IT?

The Service Desk block with a connecting line below pointing to self-service, service request, and Service Catalog. The line in Figure A-1 on page 308 that connects incident, problem, change, and config is Knowledge Management within your environment. End-to-end Service Management using IBM Service Management Portfolio A Service Desk is defined by the goal of a Service Desk to act as the central point of contact between the user and IT. However, much has changed in recent years as the Service Desk has developed into a fully functioning business application that not only supports users, but also handles requests from the business itself. By enabling users with a self-service interface that enables them to become more productive, knowledgeable, and efficient, the Service Desk has created a more satisfied user. This addition has also allowed the people representing a Service Desk to be more productive and focus their time on working on issues that impact the business. Because every request is different they must be handled differently.

Not all requests are incidents and not all incidents stem from a request. This is where it is important to stress the automated integration with an Enterprise Event Management application that automates much of the manual operations between IT and the Service Desk as well as provides advance warnings of potential impacts to the business, which impacts call volume at the Service Desk. At a high-level Knowledge Management allows both users and Service Desk analysts to make better use of their time. If either experiences an issue that has already been addressed it can be referenced in the knowledge base and quick resolved by following the recommended steps.

The Service Catalog is also a new addition to the existing functionality of a Service Desk. The Service Catalog provides many capabilities, which begin with enabling IT to keep a record of the services provided to the business, the given attributes of the service, instruction for requesting the service, instruction for approving the service, as well as any agreements between the business and IT relating to the service, such as SLAs. A very important add-on to a Service Catalog is the ability to automate service request fulfillment with automated provisioning.

Step 7: License compliance, usage accounting and IT Asset Management

How are you tracking software licenses? Do you know how and when you are out of compliance? Do you have the ability to track IT assets including software (both enterprise and desktop)? Measuring software usage helps you better plan and budget for future software spending or charge back.

Tivoli License Compliance and Usage Accounting can track the IT Asset life cycles including software and hardware as it correlates to license compliance and usage. Software overspending is a part of many IT organizations. You buy too much not knowing what is really being used. When it comes time to validate how many licenses you have versus how many are in use, you often rely on a third party to provide this as a service and the expense then becomes even greater. Worse yet, if you have grown out of compliance you could face huge penalties. It is important for you to understand the value of knowing what you have and what you use. In many cases costs can be recovered and invested in other areas of IT. Enterprise Asset Management is another reason why a you would choose Tivoli as the Enterprise Asset Management application provider, in addition to the fact that we offer application-specific focus on industries and are the only provider of a unified EAM and IT Asset application. In addition, IBM is leading the way in programs like Green Datacenter and Spatial Asset Management offerings. These technologies are focused on streamlining Asset Management and reducing the overall cost associated with maintaining assets.

Step 8: Storage Management

How are you managing your storage environments, such as data retention, availability, compliance, growth, costs, and so on? Storage environments for many customers are growing exponentially. Often times IT organizations have no control over the rapid growth of storage environments.

This is largely due to the lack of visibility and control over how the storage is being used. Many times IT organizations choose to keep everything in fear of violating a regulatory requirement, which means many duplicates. There are hundreds of regulatory requirements imposed on businesses such as Basel II,

SEC Rule 17a-4, and HIPAA, and they all require some level of proof that they are being adhered to. Proving compliance can cost time and money, most of which comes from the IT organization's bottom line. By gaining visibility into a client's storage environment they are able to rid themselves of duplicate data, data that otherwise violates corporate policies, and data that does not fall under any regulatory requirement. In addition to gaining visibility to your IT storage environment, there are also benefits in automating the management of the environment, such as configuration changes. By building an automated process to handle otherwise manual tasks and making them repeatable you can save time and money. For example, in medium-to-large environments clients may have hundreds to thousands of storage devices. Many times when a storage administrator makes an *ad hoc* configuration change to one device it has to then be repeated on multiple devices. This takes a copious amount of time and increases the chance of end-to-end Service Management using IBM Service Management Portfolio making a mistake. By automating this process by

leveraging technology you not only save time, but also reduce the risk of human error.

Step 9: Automation and scheduling

Today's information technology operation’s support model is rapidly becoming untenable. According to a recent IDC study commissioned by IBM Tivoli, approximately 70% of today's IT budget is labor. Because IT is increasingly fundamental to revenue in today's technology-driven businesses and will likely continue to grow in size and complexity, companies are facing a growth-versus-cost dilemma. IT complexity must be simplified, or growth will begin to be hampered by the cost required to support it.

IBM Self-Managing Autonomic Technology enables fluid response to change. Autonomic computing systems have the ability to manage themselves and dynamically adapt to change in accordance with business policies and objectives, enabling computers to identify and correct problems often before they are noticed by IT personnel. IBM delivers Self-Managing Autonomic Technology solutions to help companies transform their IT infrastructures into more resilient, responsive, efficient, and secure systems that deliver significant value today.

As part of the IBM IT Service Management vision, autonomic computing can help you efficiently manage and proactively deliver much-needed IT services that drive business performance. Because self-managing systems can automatically detect improper operation of systems, transactions, and business processes, and then initiate corrective action without negatively impacting system uptime, you can deliver information and services whenever needed.

Step 10: Business Service Management

The correct information for the correct people at the correct time—that is what drives effective decision making for today’s organizations. However, organizations now need more information, more often, and from more sources than ever before to deal with a growing onslaught of business and technology challenges. Lines of business are under increasing pressure to grow revenue. Customers are demanding higher value and convenience. New government regulations mandate greater accountability through external audits and compliance reporting. Mergers and acquisitions require extensive IT integration and add greater complexity to both business and operations. Accordingly, operations are under increasing pressure to develop IT solutions that properly align with business objectives, provide accurate, timely, and comprehensive information to users, and support service level agreements across the business. To address these challenges and achieve business objectives, organizations require holistic visibility into critical services, processes, and transactions across

the enterprise and beyond. More specifically, lines of business and operations need contextual service visibility, linking infrastructure to corresponding services, processes, and customers. At the same time, business and operations need actionable intelligence that includes transactional, compliance, revenue, service level, and other success indicators required to effectively manage ongoing delivery against objectives.

Abbreviations and acronyms

API	Application-programming interface
BCP	Business Continuity Process Manager
BIA	Business Impact Analysis
BIRT	Business Intelligence Reporting Tool
BPEL	Business Process Execution Language
CCMDB	Change and Configuration Management Database
CDM	Common Data Model
CI	Configuration item
CLI	Command line interface
CMDB	Configuration Management Database
COBIT	Control Objectives for Information and related Technology
CSI	Continual Service Improvement
CTI	Computer Telephony Interface
DLA	Discovery Library Adapter
DSL	Definitive Software Library
ESS	Extended Security Services
FP	Fix pack
GA	Generally available
GB	Giga byte
GUI	Graphical user interface
GUID	Globally unique identifier
HACMP	High Availability Cluster Multi-Processing
HIPAA	Health Insurance Portability and Accountability Act
IBM	International Business Machines Corporation
IM	Integration Module
ISM	IBM Service Management
ITIC	IBM Tivoli Integration Composer
ITIL	Information Technology Infrastructure Library
ITM	IBM Tivoli Monitoring
ITSO	International Technical Support Organization
ITUP	IBM Tivoli Unified Process
ITUPC	IBM Tivoli Unified Process Composer
JDBC	Java Database Connectivity
JVM	Java Virtual Machine
KPI	Key Performance Indicator
LDAP	Lightweight Directory Access Protocol,
LDO	Logical Device Operation
LIC	Launch-in-context
LMO	Logical Management Operation
LOB	Line of business
LPAR	Logical partition
MB	Mega byte
MBO	Maximo Business Object
MEA	Maximo Enterprise Adapter
OMP	Operational Management product
OPAL	Open Process Automation Library
PMP	Process Manager product
RFC	Request for change

RPM	Release Process Manager
SLA	Service Level Agreement
SM	Service Management
SOA	Service-oriented architecture
SRM	Service Request Manager
TADDM	Tivoli Application Dependency Discovery Manager
TAMIT	Tivoli Asset Management for IT
TBSM	Tivoli Business Service Manager
TCM	Tivoli Configuration Manager
TDI	Tivoli Directory Integrator
TEC	Tivoli Enterprise Console
TEP	Tivoli Enterprise Portal
TIM	Tivoli Identity Manager
TPM	Tivoli Provisioning Manager
TRPM	Tivoli Release Process Manager
TSRM	Tivoli Service Request Manager
VMM	Virtual Member Manager
WSDL	Web Services Description Language
XML	Extensible Markup Language

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks publications

For information about ordering these publications, see “How to get Redbooks publications” on page 321. Note that some of the documents referenced here may be available in softcopy only.

- *Implementing IBM Tivoli Service Request Manager V7.1 Service Desk*, SG24-7579
- *Implementing IBM Tivoli Service Request Manager V7.1 Service Catalog*, SG24-7613
- *IBM Tivoli Asset Management for IT Portfolio Overview*, SG24-7376
- *Deployment Guide Series: IBM Tivoli CCMDB Overview and Deployment Planning*, SG24-7565
- *Integration Guide for IBM Tivoli Service Request Manager V7.1*, SG24-7580
- *Deployment Guide Series: IBM Tivoli CCMDB Overview and Deployment Planning*, SG24-7565
- *IBM Tivoli Application Dependency Discovery Manager Capabilities and Best Practices*, SG24-7519

Online resources

These Web sites are also relevant as further information sources:

- Tivoli Open Process Automation Library (OPAL)

  http://catalog.lotus.com/wps/portal/topal
- IBM Service Management information center

  http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp
- IBM Systems Journal - Fall 2007 issue

  http://ww.research.ibm.com/journal/sj46-3.html

- IBM United States Software Announcement 208-241

  http://www-01.ibm.com/common/ssi/index.wss

- ITIL Service Management Self-Assessment

  http://www.itsmfi.org/content/self-assessment-itil-v2

- IBM Service Management (ISM) Assessment Tool

  https://www14.software.ibm.com/webapp/iwm/web/reg/signup.do?lang=en_US&source=itsmat

- Tivoli Business Service Manager Installation Guide

  http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=/com.ibm.tivoli.itbsm.doc/installguide/bsmi_t_installing_eif_probe.html

- Netcool/Omnibus Gateway HP Openview ServiceCenter Reference Guide

  http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.netcool_OMNIbus.doc/hpscgw-pdf.pdftp://www-01.ibm.com/common/ssi/index.wss

- Glossary of IT infrastructure library

  http://www.best-management-practice.com

- IBM Tivoli Business Continuity Process Manager product Web page

  http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.bcpm.doc/welcome.htm

- IBM Tivoli Change and Configuration Management Database product

  http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.ccmdb.doc_7.1.1/ccmdb_welcome.htm

- IBM Tivoli Service Request Manager product

  http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.srm.doc_7.1/srm_welcome.htm

- IBM Tivoli Release Process Manager product

  http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?topic=/com.ibm.rpm.doc_7.1.1/rpm_welcome.htm

- IBM Tivoli Asset Management for IT product

  http://www-01.ibm.com/software/tivoli/products/asset-management-it

- IBM Tivoli Unified Process (ITUP)

  http://www-01.ibm.com/software/tivoli/governance/servicemanagement/itup/tool.html

How to get Redbooks publications

You can search for, view, or download Redbooks publications, Redpapers publications, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks publications, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

D

E

F

G

H

I

J

L

M

N

T

U

V

W

X

Redbooks

End-to-End Service Management Using IBM Service Management Portfolio

End-to-End Service Management Using IBM Service Management Portfolio

Learn how to use ISM Portfolio for comprehensive Service Management

Implement real-life scenarios

Learn best practices

IBM® Tivoli® Service Request Manager, IBM Tivoli Change and Configuration Management Database, IBM Tivoli Asset Management for IT, IBM Tivoli Release Process Manager, and IBM Tivoli Business Continuity Process Manager are key components of the IBM Service Management strategy. This IBM Redbooks® publication presents scenarios on the combined usage of these products for implementing a complete, end-to-end Service Management solution.

We start by introducing Information Technology Infrastructure Library (ITIL®)-based Service Management, followed by a discussion of how to design your ITIL-based Service Management solution using the Tivoli products.

We have included several scenarios that will help you understand how these products work together in a real customer environment.

This book is a major reference for IT specialists working on implementing ITIL-based Service Management solutions using Tivoli products. IT managers, IT architect, and pre-sales technical specialists will also benefit from information presented in this book.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7677-00 ISBN 0738432210