WebSphere Process Integration V6: Business Process Management Modeling through Monitoring

WebSphere Business Modeler

WebSphere Integration Developer
WebSphere Process Server

WebSphere Business Monitor

Ueli Wahli
Larissa Leybovich
Neil MacKinnon
Eric Prevost
Russell Scher
Andre Venancio
Sascha Wiederkom

**Final draft:
March 1, 2006**

Redbooks

International Technical Support Organization

WebSphere Process Integration V6: Business Process Management Modeling through Monitoring

March 2006

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

First Edition (March 2006)

This edition applies to WebSphere Business Modeler Version 6.0, WebSphere Integration
Developer Version 6.0.1, WebSphere Process Server Version 6.0.1, and WebSphere Business
Monitor Version 6.0.

This document created or updated on March 1, 2006.

Contents

Notices

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates 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. 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 IBM's application programming interfaces.

Trademarks

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

@server®	Cube Views™	IMS™
@server®	CICS®	Lotus®
Redbooks (logo) ™	Domino®	Rational®
AIX®	DB2 Universal Database™	Redbooks™
ClearCase®	DB2®	WebSphere®
Cloudscape™	IBM®	

The following terms are trademarks of other companies:

EJB, Java, Java Naming and Directory Interface, JavaScript, JavaServer, JavaServer Pages, JDBC, JSP, J2EE, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Visio, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Pentium, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

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

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

Preface

This redbook presents a business process management (BPM) "round trip" scenario, showing how a business can use a full business integration solution to complete the following tasks:

- ► Model and simulate a business process
- ► Develop and test an application to implement the business process
- ► Deploy and run the application on a server
- ► Monitor the application to observe pre-determined key performance indicators
- ► Import the observed data to make revisions to the original process model

The business scenario described in this document has been simplified to provide a full description of each stage of the BPM end-to-end process. To avoid an overly large and unwieldy document, the authors' focus is on specific tasks, elements, and details, and not on presenting all possible facets of a complex business process.

Each IBM® software product included in the full BPM cycle has a large number of features and capabilities. For details on all of the available functions of these products, see the help documentation included with each product.

The redbook is structured into five parts:

1. Business process management—This part provides a general introduction of the subject and an overview of the scenario that is used throughout the document.

2. Modeling the business—This part describes how to use **WebSphere® Business Modeler** to model the business process and its resources, simulate the process, and analyze the simulation results. Finally it introduces the business measures defined for monitoring the process.

3. Development and deployment—This part described how to implement the business process application, exported from the Modeler, using **WebSphere Integration Developer** and **WebSphere Process Server**.

4. Monitoring the application—This part describes how to use **WebSphere Business Monitor** to monitor the running application and measure the business process.

5. Continuous process improvement—This part provides information about the round trip scenario of improving the model based on the measurements, and regenerating and redeploying the modified application for further measurements.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working in Markham (Toronto), Canada, for the International Technical Support Organization, Raleigh Center.

Andre   Russ   Larissa   Ueli   Sascha   Eric

Ueli Wahli is a Consultant IT Specialist at the IBM International Technical Support Organization in San Jose, California. Before joining the ITSO 20 years ago, Ueli worked in technical support at IBM Switzerland. He writes extensively and teaches IBM classes worldwide about WebSphere Application Server, and WebSphere and Rational® application development products. In his ITSO career, Ueli has produced more than 30 IBM Redbooks™. Ueli holds a degree in Mathematics from the Swiss Federal Institute of Technology.

Larissa Leybovich is a Consulting IT Specialist in the Software Services Group of IBM with focus on business process modeling and B2B implementations. She became a highly experienced business process analyst, modeler, an e-business integration specialist by analyzing, modeling, designing, and implementing a variety of business applications, products, and B2B business solutions for the last 24 years. Larissa is specializing in providing leadership and direction on process modeling methodology, best practices, development of standards for business process modeling, and integration with WebSphere Process Server. Larissa holds a B.S. degree in Systems and Data Processing from Washington University, St. Louis, Missouri and a B.S. degree in Electrical Engineering from the Institute For Motion Picture Engineers, St. Petersburg, U.S.S.R.

Eric Prevost is an IT Architect at the IBM Business Consulting Services in Luxembourg. He has 6 years of experience in architecture and implementation of enterprise-wide, mission-critical systems. He holds a Master in Computer Engineering from the University of Marne-la-Vallée, France. He has worked at IBM for 8 years. His areas of expertise include workflow, integration, and collaboration solution. He teaches classes on groupware for post-graduate students at the EFREI school of Villejuif, France.

Russell Scher is a Software Engineer within IBM's Software Group, Customer Support L2 WebSphere Businee Integration Modeler and Monitor. He joined IBM with years of experience in industry verticals including telecom, semi-con, aerospace and retail. He specializes in Monitor, Modeler, workflow, and overall BPM technologies. He holds a Bachelor's of Science degree in Electrical Engineering from UCLA.

Andre Venancio is an Senior IT Specialist with the WebSphere Software Platform (Software Group) in IBM Brazil. He has six years of experience with WebSphere products developing solutions with portal, application connectivity (MOM) and process integration. He holds a Bachelor's Degree in Mathematics from the Fundação Santo André in Brazil and a post-graduate degree from Faculdade de Informática e Administração Paulista (FIAP), São Paulo, Brazil on enterprise solutions using distributed object technologies and Web services components engineering using J2EE™.

Sascha Wiederkom is an IT Specialist at the IBM Software Group in Austria. He has 5 years of experience in implementing enterprise wide Web projects for customers in Austria. He has worked for IBM for 6 years, mainly with WebSphere Application Server, application development, and business integration tools. He holds a Master Degree in Computer Science from the Technical University of Vienna, Austria.

Thank you

A special "Thank you" for Neil MacKinnon of the Toronto lab who developed the original business model and scenario documentation for the case study.

Neil MacKinnon is an Information Developer with the WebSphere Business Modeler team in Toronto. He joined IBM Canada in 2001 as a technical editor, and has been a member of the WebSphere Business Modeler team since 2004. Neil holds a Bachelor of Arts degree in History from the University of Guelph, and prior to joining IBM worked as writer and editor for several legal, medical, and technical publishers.

Contributors

This redbook would not have been possible without the help of numerous people across the IBM company.

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

► **Samantha Shurety**, IBM Toronto lab, for proposing this project, getting it approved, and supporting the ITSO team to run the residency in Toronto.

► **Jinchao Huang**, **Raymond Kong**, and **Randy Williams**, IBM Toronto lab, for developing together with with Neil MacKinnon the overall scenario and the initial business model for WebSphere Business Modeler.

► **Sherry Ocopnick**, IBM Toronto lab, for her help to define appropriate business measures for monitoring and dimensional analysis.

► **John McLean**, **Dominique Emond**, **Rakesh Brahmhatt**, **Wendy Butt**, and **Sunita Chacko**, IBM Toronto lab, for their help to implement the business process in WebSphere Integration Developer.

► **Andreas Schoen** and **Stefan Ruettinger**, IBM Boeblingen lab, for their help to implement human task processing through customized JSPs.

► **Eric Erpenbach**, AIM Customer Programs, for helping with the implementation of the business process using WebSphere Integration Developer.

► **Ben Parees** and **Jim Thorpe**, IBM Raleigh lab, and **Mohamed Abdel-Aziz**, **Passinte Heikal**, **Deena Hassan**, and **Hossam Badawi**, IBM Egypt, for helping with the setup and configuration of WebSphere Business Monitor.

► **Jason Fox**, IBM Burlingame, for providing the hardware environment to run WebSphere Business Monitor.

► **Scott Walden**, IBM Raleigh, for providing guidance and training material for the monitoring environment.

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll 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 Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

► Use the online **Contact us** review redbook form found at:

`ibm.com`/redbooks

► Send your comments in an email to:

redbook@us.ibm.com

► Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 662
P.O. Box 12195
Research Triangle Park, NC 27709-2195

Part 1

Introduction

In this part we introduce the concepts of business process management, the IBM process integration products that support business process management, and the case study, a company named *ClipsAndTacks*, that we use throughout the rest of the book.

Business process management

In this chapter we introduce the concepts and benefits of business process management (BPM) and service-oriented architecture (SOA) that enables business value through operational flexibility, responsiveness and reuse.

We also introduce the Service Component Architecture (SCA) and Services Data Objects (SDO) as building stones for the SOA architecture.

What is a business process?

If you search the Web for a definition of *business process*, you find many definitions. All these definitions contain certain common elements:

► A business process is triggered by a business event.

► A business process has an input and creates an output that is of value to the organization, its stake holders, or its customers.

► A business process is composed of related structural activities. Material and/or information flows between the process activities.

► A business process can be part of a larger process and can include or depend on other business processes.

► A business process can be viewed as the workflow for a use case.

► A business process usually depends upon several business functions for support, for example, IT, personnel, and accommodation.

What is business process management?

Business process management (BPM) leads to business innovation and optimization by implementing business strategy through modeling, developing, deploying and managing business processes throughout their entire lifecycle. BPM acts as an enabler for the businesses in defining and implementing strategic business goals and then measuring and managing company's financial and operational performance against these goals. The power of optimal results from the BPM lifecycle activities is derived from the integrated set of the robust technology infrastructure and tools.

Business process management provides a convergence of technology that removes business and IT constraints through integration and enhanced technology to help streamline business transformation. These capabilities provide tight integration of operational and analytical environments, business and IT environments, and strategy with daily operations.

Business process management combines business processes, information, and IT resources, aligning your organization's core assets—people, information, technology, and processes—to create a single integrated view, with real-time intelligence, of both its business measurements and IT system performance. This integration of resources allows your organization to obtain business information faster, respond more quickly to market trends and competitive threats, and improve operational efficiencies and business results—all attributes of an on demand enterprise.

Benefits of BPM

BPM allows for an enterprise to be flexible and responsive to the ever changing on-demand business through the optimization and automation of the business processes to:

► Identify and eliminate redundancies and bottlenecks

► Reduce risk by gaining an understanding of process impacts prior to operationalizing

► Decouple business integration logic from its underlying implementation code

► Increase portability and decrease mainteance cost by being based on industry standards

► Automate process implementation, eliminate manual deployment tasks

► Immediately execute new business rules and processes

► Visualize actual process performance against key performance indicators

► Pinpoint future process improvements

The robust business process management solution will provide powerful tools for the business and IT side of the business to meet their challenges:

► The **business executive** needs:

– Revenue growth with cost containment

– Responsiveness to business conditions and ability to pursue new market opportunities

– Improving internal skills, capabilities and leadership as first step toward growth

► The challenges for the **IT executive** are:

– Aligning IT and business goals to grow revenue and contain costs

– Building responsiveness and agility into the organization through IT

– Enabling people and teams to be more effective through IT

Information on business process management

You can find definitions and more information on business process management on the Internet, for example:

► Wikipedia:

http://en.wikipedia.org/wiki/Business_Process_Management

► Business Process Management Initiative:

http://www.bpmi.org/

IBM business process management solution

The IBM process integration portfolio provides capabilities required for the delivery of the comprehensive enterprise wide business process management strategies and solution. It offers a holistic approach to transform and manage a business by aligning strategic and operational objectives with business activities and supporting IT services.

The IBM BPM solution includes development tools, used to implement custom artifacts that leverage the infrastructure capabilities, and business performance management tools, used to monitor and manage the runtime implementations at both the IT and business process levels.

Business process management allows companies to implement the continuous end-to-end business process lifecycle in an open environment (Figure 1-1).

Websphere Integration Developer
- **Discover**
- **Design, construct, and test**
- **Compose**

Webphere Process Server
- **Integrate people**
- **Integrate processes**
- **Manage and integrate information**
- **Run and manage complex applications**

Websphere Business Modeler
- **Gather requirements**
- **Model, design & simulate**
- **Define business metrics**

Websphere Business Monitor
- **Monitor business metrics and runtime**
- **Dashboards, scorecards**

Figure 1-1   Business process management lifecycle

The IBM BPM solution component tools support the following major activities:

► **Model**—Capture, simulate, analyze and optimize business models to reduce risk and increase flexibility

► **Assemble**—Develop, assemble and test integrated solution

- ▶ **Deploy**—Direct deployment of models and policies to realize business intent
- ▶ **Manage**—Manage the deployed models:
  - **Monitor** and correlate metrics and alerts in real-time from internal and external sources to gain visibility into the business and IT performance
  - **Analyze performance** results to gaining insight into the business metrics and information for contextual based decision making
  - **Act** by responding at the right time to insights through collaboration, optimization, and automation to excel

One key feature of the IBM business process management is the linkage between the development platform and the business performance management services. It is addressed in the Governance & Processes section in Figure 1-1:

- ▶ Governance reflects all corporate guidelines
- ▶ Processes covers the business performance management area

This functionality enables the delivery of runtime data and statistics into the development environment to allow for the completion of the analyses that drives iterative process re-engineering through a continuous business process improvement cycle.

Buisiness performance management

Business performance management tools incorporate monitoring capabilities that aggregate operational and process metrics in order to efficiently manage systems and processes. Managing these systems requires a set of capabilities that span the needs of IT operations professionals and business analysts who manage the business operations of the enterprise.

These capabilities are delivered through a set of comprehensive services that collect and present both IT and process-level data, allowing business dashboards, administrative dashboards, and other IT- level displays to be used to manage system resources and business processes.

Through these displays and services, it is possible for line-of-business (LOB) and IT personnel to collaborate to determine, for example, what business process paths may not be performing at maximum efficiency, the impact of system problems on specific processes, or the relationship of system performance to business process performance.

This collaboration allows IT personnel and assets to be tied more directly to the business success of the enterprise than they traditionally have been.

For more information on business performance management, refer to:

► IBM business performance management community:

http://www.ibm.com/software/info/topic/perform/partnerpage.html

► Wikipedia:

http://en.wikipedia.org/wiki/Business_performance_management

IBM business process management products

The BPM solution consists of four separate products:

► WebSphere Business Modeler
► WebSphere Integration Developer
► WebSphere Process Server
► WebSphere Business Monitor

Together, these products enable businesses and other organizations to plan and implement a unified business process strategy based on realistic simulations and observed data.

At a high level, here is how you use these products together to get the right BPM solution for your business needs:

► With **Modeler** you to begin the cycle by designing the optimal business process for a particular case. Typically, Modeler is used by a business analyst.

► With **Integration Developer** you implement the model by creating the application code to automate the model and to access subsystems, such as databases and enterprise information systems. Integration Developer is used by a developer who integrates services to create an application, and an application programmer who focuses on the development of particular services.

► **Process Server** provides the production server to run and manage the application you create. Typically, a system administrator works with Process Server.

► **Monitor** provides real time performance monitoring of the application. Monitor is usually set up and administered by a system administrator.

Once built, improvements to the model and application are on-going.

For more information on the IBM business process management products, refer to:

http://www.ibm.com/developerworks/websphere/zones/businessintegration/

Service-oriented architecture (SOA)

Service-oriented architecture (SOA) is the IT model which enables business value through operational flexibility, responsiveness, and reuse. SOA is an application framework that takes everyday business applications, breaks them into individual business functions and processes, called services, and then inter-relates them through well-defined interfaces and contracts. The interfaces are defined in a neutral manner that is be independent of the hardware platform, the operating system, and the programming language in which the service is implemented. This allows services, built on a variety of such systems, to interact with each other in a uniform and universal manner.

SOA is an evolution as opposed to a revolution. The growing market momentum around SOA means that companies that compete with you are adopting SOA for strategic advantage. SOA would be impossible without industry standards. While there have always been standards proposed and declared in the IT world, finally there are the critical mass of very broad industry support to firmly establish that the current SOA standards are real, meaningful, and are here to stay.

Benefits of SOA

Establishing a service-oriented architecture can help prepare both IT and business processes for rapid change. Even in the early stages of adopting an SOA, your organization will benefit from:

▶ Increase revenue—Create new routes to market and create new value from existing systems

▶ Provide a flexible business model—React to market changes more quickly

▶ Drive down cost—Eliminate duplicate systems, build once and leverage, and improve time to market

▶ Reduce risk and exposure—Improve visibility into business operations

The SOA approach can bridge the gap between what you want your business to accomplish and the infrastructure tools you need to get there:

▶ Decrease development and deployment cycle times by using pre-built, reusable services building blocks

▶ Integrate across the enterprise—even historically separate systems—and facilitate mergers and acquisitions of enterprises

▶ Reduce cycle times and costs by moving from manual to automated transactions

▶ Make it easier to do business with business partners by increasing your flexibility

> ▶ Brings adaptable, scalable solutions to complex business problems by using best practices, such as layering and loosely-coupled components

IBM SOA foundation

The IBM SOA foundation is an integrated, open set of software, best practices and patterns that provides you with what you need to get you started with SOA. The SOA foundation provides full support for the SOA lifecycle through an integrated set of tools and runtime components that allow you to leverage skills and investments across the common runtime, tooling, and management infrastructure.

The components are modular allowing you to pick and choose the pieces you need to deliver an immediate impact while knowing that what you pick will work with pieces you add later on. In addition, the SOA foundation is scalable allowing you to start small and grow as fast as your business requires. The SOA foundation provides extensive support for business and IT standards; facilitating greater interoperability and portability between applications. It can also help you to leverage SOA to extend the value of the applications and business processes that are running your business today.

The SOA reference architecture (Figure 1-2) is a way of looking at the set of services that go into building an SOA. These capabilities can be implemented on a build as-you-go basis allowing capabilities and project level solutions to be easily added as new requirements are addressed over time.

The backbone of the reference architecture is the enterprise service bus (ESB) that facilitates communication between services. he reference architecture is a great tool for laying out roadmaps for pursuing SOA.

Every components included in this architecture is provided by separate IBM products. The highlighted parts show the components which address the business process management area.
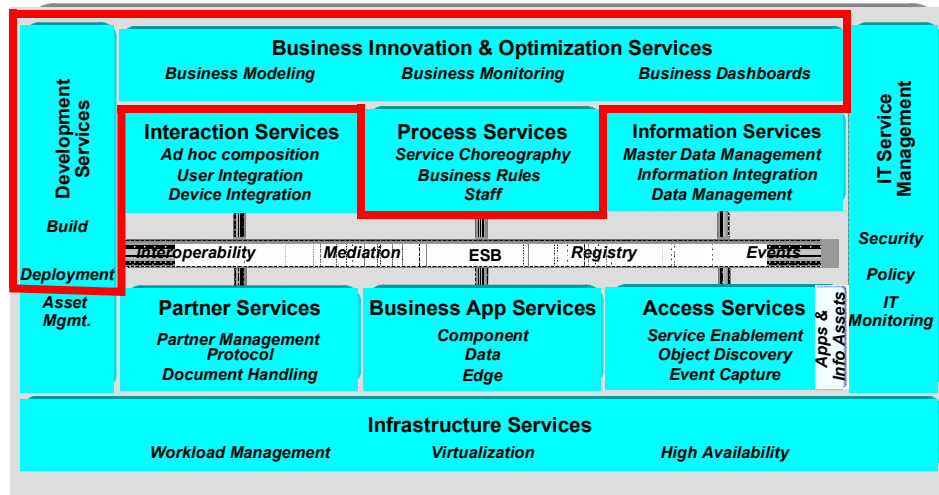
Figure 1-2   IBM SOA reference architecture

This service-oriented architecture provides a modular, scalable, portable and interopable environment to support the equivalent aspects in the business area.

The reference architecture shows the tight integration with the others IT critical aspects such as security, IT monitoring, virtualization, and workload management.

IBM BPM solution on the SOA foundation

The BPM solution available from IBM enables service-oriented, end-to-end process management for your organization based on SOA. Its SOA component applications enables businesses and organizations to plan, develop, implement, and improve their processes.

This solution lets organizations closely scrutinize costs, scheduling, resources, workflow, and other factors that affect the efficiency and viability of a given process, and determine the most effective methods for revising and improving that process.

Service Component Architecture (SCA)

Service Component Architecture takes advantage of the emerging computing trend called service-oriented architecture (SOA), which structures IT assets as a series of reusable services that perform a business function.

Service Component Architecture (SCA) is a new technology to simplify application development and implementation in a service-oriented architecture (SOA). With SCA, customers will be able to more easily create new and transform existing IT assets into reusable services that may be rapidly adapted to changing business requirements. Furthermore, these new technologies greatly reduce complexity associated with developing applications by providing a way to unify these services regardless of the programming languages they were written in and the platforms used to run them.

By structuring applications as a series of services, IT assets become more agile and organizations are better able to align their investments in dynamic business environments. In addition, adopting these new standards will provide organizations a higher degree of investment protection, as they will be able to reuse services with a variety of middleware technologies.

SCA provides an open, technology-neutral model for implementing IT services defined in terms of a business function, and do not unnecessarily expose the programmer to the complexity of traditional middleware programming. SCA also provides a model for the assembly of business solutions from collections of individual services, with control over aspects of the solution such as access methods and security.

SCA gives developers and architects the ability to represent business logic as reusable components that can be easily integrated into any SCA-compliant application or solution. The resulting application is known as a composite application.

Service Data Objects (SDO)

Service Data Objects is a technology that was originally developed as a joint collaboration between BEA and IBM and is now being developed by BEA, IBM, Oracle, SAP, Siebel, Sybase and XCalia. SDO is designed to simplify and unify the way in which applications handle data. Using SDO, application programmers can uniformly access and manipulate data from heterogeneous data sources, including relational databases, XML data sources, Web services and enterprise information systems.

SDOs specify a standard way to access data and can be used to modify business data regardless of how it is physically accessed. Developers and architects do not need to know the technical details of how to access a particular back-end data source in order to use SDO in their composite applications. Consequently, they can use static or dynamic programming styles and obtain connected as well as disconnected access.

SDO complements SCA by providing a common way to access many different kinds of data. The specification reduces the skill levels and time required to access and manipulate business data. Today, a multitude of APIs are used to manipulate data. These APIs tend to tightly couple the source and target of the data making their use error-prone and subject to breaking as business requirements evolve. SDO makes it easier to use and realize the value of these APIs without having to code directly to them.

Information on SCA and SDO

In response to requests from customers and independent software vendor partners, IBM is jointly delivering two specifications for building systems that use a service-oriented architecture (SOA), which aim to provide developers with simpler and more powerful ways of constructing applications based on SOA: Service Component Architecture (SCA) and Service Data Objects (SDO).

These specifications are available at:

http://www.ibm.com/developerworks/webservices/library/specification/ws-scas dosumm/

http://www.ibm.com/developerworks/webservices/library/specification/ws-sca/

http://www.ibm.com/developerworks/library/specification/ws-sdo/

More on business performance management

There is no standardization of terms in this area. For example, Meta Group and Aberdeen use business performance management (BPM), Gartner refers to corporate performance management (CPM), and companies such as PeopleSoft and Business Objects use the term enterprise performance management (EPM). IBM has adopted the business performance management term when referring to performance management practice and solutions.

As a first deliverable, the business performance management standards group has developed a common definition of business performance management that provides appropriate context for performance management, including the following principles:

► BPM is a set of integrated, closed-loop management and analytic processes, supported by technology, that address financial as well as operational activities.

► BPM is an enabler for businesses in defining strategic goals, and then measuring and managing performance against those goals.

► Core BPM processes include financial and operational planning, consolidation and reporting, modeling, analysis, and monitoring of key performance indicators (KPIs) linked to organizational strategy.

The business performance management addresses metrics, methods, processes and associated systems used for the purpose of monitoring and managing the business performance of an enterprise including :

► Encompasses balanced scorecards, operational dashboards, business activity monitoring, budgeting/planning/forecasting,

► Leverages business intelligence, systems dynamics, Sarbanes-Oxley (SOX), six sigma, value creation, process management, and organizational design,

► Requires a holistic vision of how to align business decisions and activities with strategic and operational objectives, though Implementation is typically along an incremental route-map.

Summary

In this chapter we introduced business process management and the IBM business process management solution. We also touched on the relationship between business process management and business preformance management.

2

Product overview

This chapter provides an overview of the core WebSphere process integration products. These products support complete business process management lifecycle with modular approach.

WebSphere process integration products enable you to:

► Model, simulate, and analyze complex business scenarios quickly and effectively before they are implemented.

► Transform existing processes to be dynamic and adaptive to deliver cost effective business agility.

► Allow users to monitor the business processes they have implemented so that they can continuously make improvements to them.

In this chapter we introduce these products:

► WebSphere Business Modeler Version 6

► WebSphere Integration Developer Version 6

► WebSphere Process Server Version 6

► WebSphere Business Monitor Version 6

WebSphere Business Modeler Version 6

IBM WebSphere Business Modeler (Figure 2-1) is a business process modeling tool that enables you to model, design, simulate, analyze and generate reports for your business processes, integrate your new and revised process, and define your organizations, resources and business items.

Accurate definition and modeling business processes is a critical factor in improving business performance. A business process is defined by interactions that occur between an organization's components and its environment as the organization pursues its business objectives. Business processes are often complex because of numerous incremental changes that are made in reaction to business circumstances. Without formal process documentation and process management systems, these process complexities can burden an organization with unnecessary hindrances and bottlenecks. A well-constructed business process model can help you locate and eliminate those hidden inefficiencies, costs and delays.
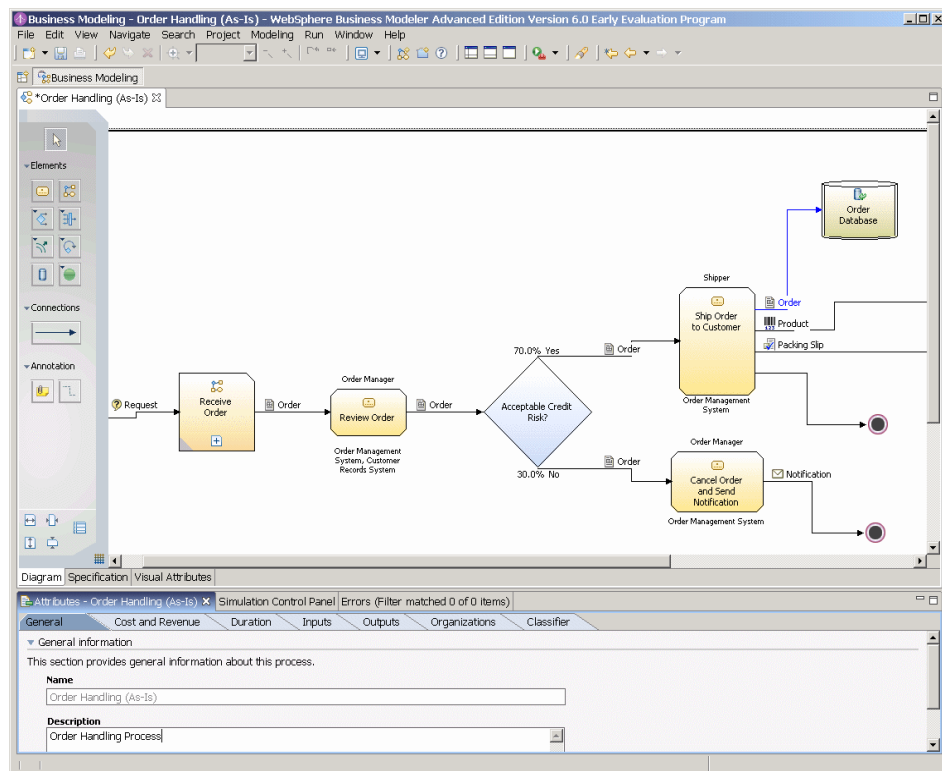
Figure 2-1   WebSphere Business Modeler

Complete and robust process modeling tool

A business process model is a visual representation of a process that contains supporting information. To create effective models, you must have a well-designed modeling structure that ensures consistent and complete representation of relevant information, including normal operations as well as alternatives and exceptions to standard procedures. You can use business process models to achieve many goals, including the following:

► Documenting existing procedures
► Determining requirements for staff, systems, and facilities
► Planning changes to existing processes and systems
► Testing and analyzing existing and proposed processes

WebSphere Business Modeler enables you to:

► Transform business process for automation
► Separate your business process model from its underlying implementation
► Create business processes that are based on industry standards
► Leverage skill sets of business and IT professionals
► Collaborate with team members
► Simulate and analyze business process

Leverage skill sets of business and IT professionals

Whether you are a business generalist or an IT specialist, WebSphere Business Modeler provides the versatility to fit your particular set of skills. Using WebSphere Business Modeler, professionals with different scopes of interest and expertise can build process models to meet a wide range of business objectives. From the business analyst who requires a high-level view of a process to drive strategic decisions, to the program developer who uses a process model as the framework for a new application, competitive businesses require a versatile modeling tool that has the flexibility to meet the needs of both business and technical professionals.

Collaborate with team members

WebSphere Business Modeler provides a built-in capability to connect with a separate product, WebSphere Business Modeler Publishing Server.

WebSphere Business Modeler Publishing Server enhances WebSphere Business Modeler by providing a way to publish business processes and related business information, such as organization diagrams, to a secure Web site. This capability supports the development, documentation, and dissemination of business process models on an enterprise and worldwide scale. By publishing business processes in a Web-based format, multiple people on multiple teams around the globe can view and contribute to the development of the business processes.

Simulate and analyze business process

WebSphere Business Modeler provides a simulation function that lets you simulate and analyze your processes under any set of circumstances. When you simulate runtime processing, WebSphere Business Modeler provides an animated view of the business process in action. You can specify a wide variety of different conditions for the simulation, including the rate and composition of process inputs and the number of personnel and system resources available to handle the process. Through simulation you can quickly determine how the performance of your business process is affected in various real or hypothetical conditions.

Transform business process for implementation

WebSphere Business Modeler enables you to transform business process models to IT-level models. You can export a Business Process Execution Language (BPEL) version of a business process model, then use WebSphere Integration Developer to import the BPEL model and use this as a basis to create an application.

Revolutionize business flexibility

Because business environments are constantly changing, requiring continual fine-tuning of processes, business improvement is a perpetual race. WebSphere Business Modeler facilitates communication between business organizations by allowing you to create a process model that has far broader uses than a static drawing. Furthermore, WebSphere Business Modeler delivers cost saving benefits by providing a single tool that effectively utilizes the same process information for many purposes, reducing the duplication of effort required by using multiple, incompatible tools.

More information

For more information on WebSphere Business Modeler, refer to:

http://www.ibm.com/software/integration/wbimodeler/

WebSphere Integration Developer Version 6

IBM WebSphere Integration Developer (Figure 2-2) is the integration tool you use for all your process integration requirements. It allows you to build flexible, composite applications by wiring service components with minimal skills based on service-oriented architecture (SOA).

IBM WebSphere Integration Developer, Version 6 software is Eclipse technology-based tooling designed to enable you to rapidly assemble business solutions based on a composite application development framework and using minimal programming skills.

With WebSphere Integration Developer, you can author SOA-based services and choreograph them into business processes that you can deploy on IBM WebSphere Process Server. WebSphere Integration Developer offers a role-based development experience that specifically targets the integration developer on a single and integrated Eclipse platform.
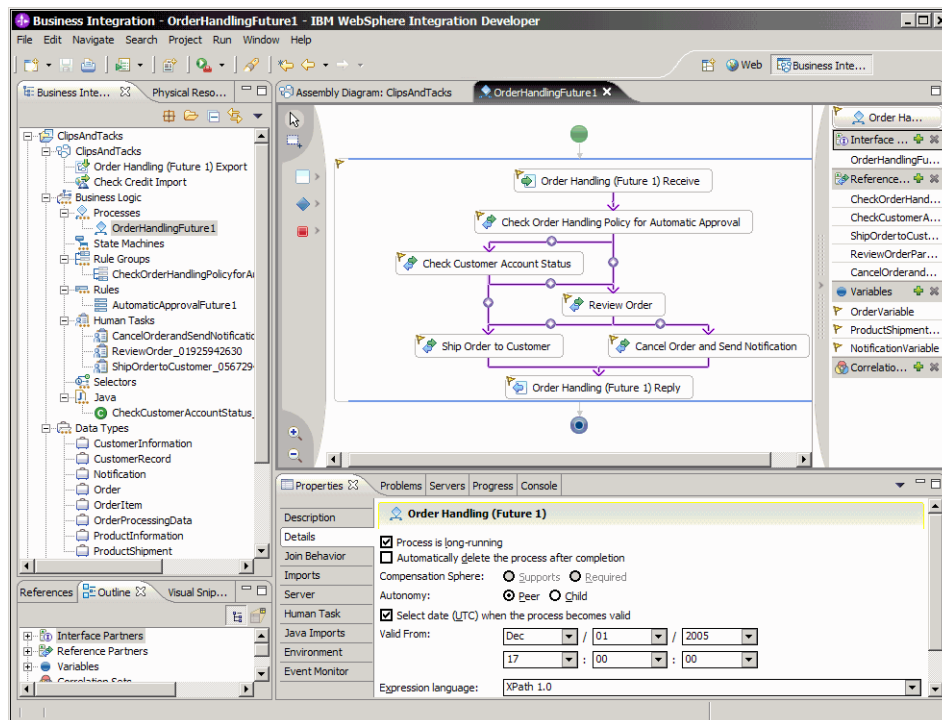
Figure 2-2   WebSphere Integration Developer

WebSphere Integration Developer is the tool for all your integration requirements and fully implements the SOA. SOA and the Service Component Architecture (SCA) enable you to convert and use your existing IT architecture as reusable service components. The framework is comprehensive and extensive, which helps you transform your enterprise to SOA by leveraging your existing IT architecture. Because WebSphere Integration Developer is based on standards-based technologies, such as Web Services Description Language (WSDL), XML Schema Definition (XSD) and Business Process Execution Language (BPEL), you can model, implement and deploy complex composite applications without extensive knowledge of the underlying implementation.

WebSphere Integration Developer is a fully integrated development environment based on the Eclipse 3 platform. Its graphically rich interface allows developers to create composite business applications by wiring service components with minimal skills and agnostic of underlying programming implementations. It has fully integrated testing, debug and deployment environment that allows you to easily deploy to WebSphere Process Server. Once deployed, you can dynamically change and adapt to changes with its rich features, such as business rules, selectors, and state machines.

WebSphere Integration Developer complements IBM WebSphere Business Modeler Version 6, and can be used in conjunction with IBM Rational Software Architect Version 6 and IBM Rational Application Developer Version 6. When combined into a single integrated development environment (IDE), these products provide a complete suite of tools to model, simulate, author, and deploy composite SOA applications.

WebSphere Integration Developer delivers a rich set of features to assemble, deploy, and manage business processes:

► Import business models from WebSphere Business Modeler and transforming to BPEL flows to automate the business processes

► Drag and drop features to create business processes

► Widgets to wire service components including:

  – Business processes
  – Human tasks
  – Business state machines
  – Business rules
  – Supporting services
  – Interface maps
  – Business object maps
  – Relationships
  – Selectors
  – Java™ objects

- ► Online library to browse and reuse service components
- ► Import services from ERP and EIS systems, such as PeopleSoft, SAP, CICS®, and IMS™

In addition, WebSphere Integration Developer enables:

- ► Composite application development:

  WebSphere Integration Developer allows you to render existing applications as standards based services. Services can be assembled together without the associated complexities of the underlying IT required for complex business applications. Now, you start building a solution from the services available to you and build only the services that don't exist, enabling true incremental solution.

- ► Flexibility for managing deployed processes:

  WebSphere Integration Developer allows you to test and debug business processes, and deploy them on WebSphere Process Server with a few mouse clicks.

  WebSphere Integration Developer's rich GUI intensive features allow for change and management of deployed processes with minimal skills and disruption.

More information

For more information on WebSphere Integration Developer, refer to:

http://www.ibm.com/software/integration/wid/

WebSphere Process Server Version 6

IBM WebSphere Process Server (Figure 2-3) is a business integration server. It is built to support solutions created based on service-oriented architecture (SOA). You can use it to build advanced business processes and traditional business integration such as enterprise application integration. WebSphere Process Server is based on WebSphere Application Server and best of the breed WebSphere Business Integration technologies.

By building on top of WebSphere Application Server Network Deployment, WebSphere Process Server can take advantage of all the mature capabilities it provides, such as clustering, high availability, embedded messaging and transaction management.

Foundation
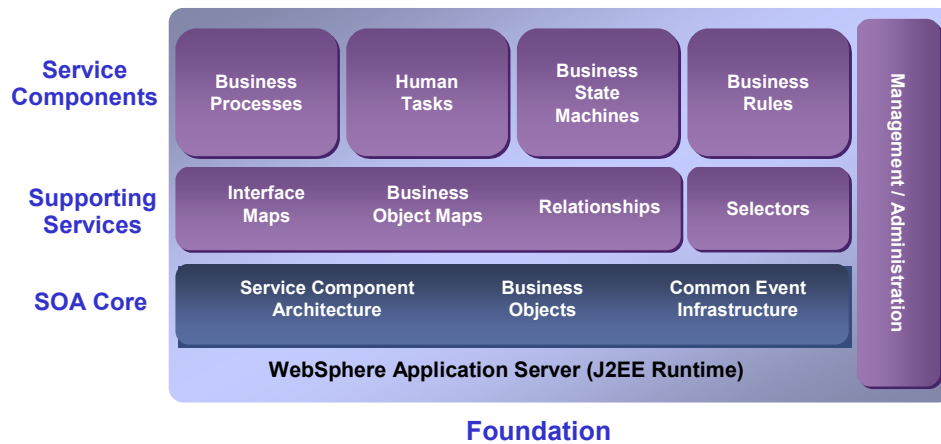
Figure 2-3   WebSphere Process Server architecture

WebSphere Process Server includes three layers:

► SOA core
► Supporting services
► Service components

SOA core

SOA core consists of Service Component Architecture (SCA), business objects, and the common event infrastructure (CEI).

Service Component Architecture (SCA)

SCA is a unified service oriented programming model. SCA defines how one can abstract and represent a service without close coupling with the details of the implementation. It defines a simple unified client programming model, as well as defining how services can be composed together to form a solution. Service Component Architecture is a universal model for business services that publish or operate on business data. Service Data Objects (SDO) provides the universal model for business data.

An SCA component is a component that runs on a SCA enabled run-time. An SCA component needs to specify its interface, implementation, as well as references (Figure 2-4).
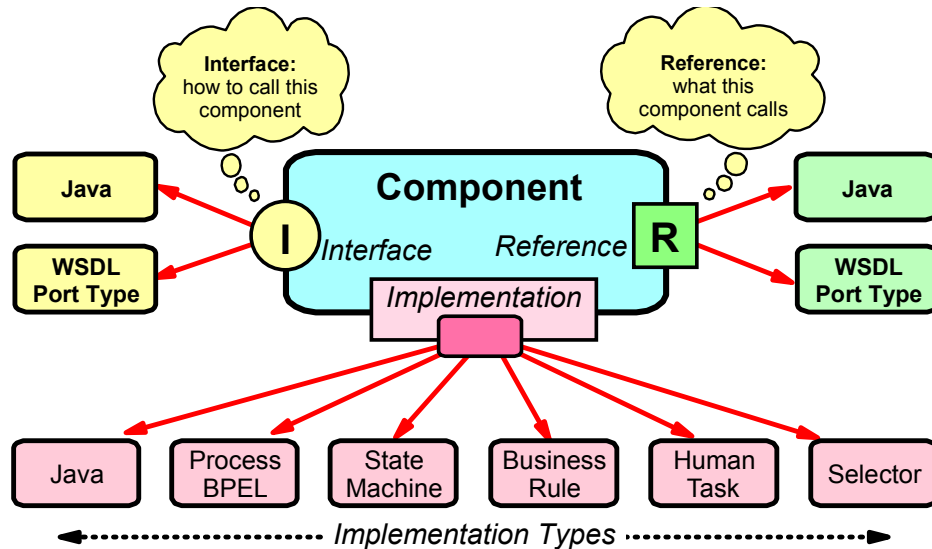
Figure 2-4   Service component

► **SCA interface**

   By definition, an interface is the place at which independent and often
   unrelated systems meet and communicate with each other. An interface can
   be defined by any programming/interface definition language.

   WebSphere Process Server currently supports a Java interface definition and
   an XML definition (WSDL port type). Arguments described in an XML
   Schema are exposed to programmers as SDO data objects.

   The WebSphere Integration Developer tooling primarily generates interfaces
   using the WSDL representation.

► **SCA implementation**

   The SCA implementation specifies the implementation type of the
   component's interface. Developers can implement business services in their
   language of choice (for example, Java, BPEL, or state machine).

   Current implementation types include business process, human task,
   interface map, selector, business rules, business state machine, and Java.

► **SCA references**

   An SCA reference specifies other SCA services that a component uses.
   These can be softlinks, which do not have to specify which specific
   component will be used.

Business objects

Business objects provide an abstraction for data access and is an important concept for integrating applications and is used to interact within Service Component Architecture. Business objects provide rich features to map, manage and transform data to underlying IT and are described through standards based XML Schema (XSD).

Common event infrastructure (CEI)

The common event infrastructure allows service component to emit events that can be captured by business monitors such as WebSphere Business Monitor for real-time monitoring of business processes.

Common business events (CBE) is a common business event model that captures the basic business event information, such as the date/time an event is created, correlation IDs (relevant business events), IBM has also proposed CBE for consideration as a new standard to the Organization for the Advancement of Structured Information Standards (OASIS).

Supporting services

On top of the runtime infrastructure and the SOA core, WebSphere Process Server offers a variety of service components. Supporting services are components that are needed in any integration solution including data transformation and synchronization services.

Implementing the main features of WebSphere Enterprise Service Bus (ESB), a mediation component can be used to convert XML-based data formats. It can also be used to mediate between various transports, for example JMS and Web services. A mediation component contains a "message flow" that operates on a message including transformation, logging and filtering and custom Java operations. Above and beyond the ESB mediation component, WebSphere Process Server contains higher-level transformation capabilities.

Interface maps

Interface maps let you invoke components by translating these calls. It is possible for interfaces of existing components to match semantically but not syntactically (for example, *updateCustomer* versus *updateCustomerInDB2*). This is especially true for components that already exist and services that need to be accessed. Additionally, you can use business objects to translate the actual business object parameters of a service invocation.

Business object map

A business object map lets you translate one type of business object into another type of business object. You can use these maps in a variety of ways, for example, in an interface map to convert one type of parameter data into another.

Relationship

A relationship can be used to establish relationship instances between objects in disparate backend systems. You may want to access the same data within business integration scenarios, for example, customer records, or in various backend systems, such as an ERP system and a CRM system. A common problem for keeping business objects in sync is that different backend systems use different keys to represent the same object. These relationships are typically accessed from a business object map when translating one business object format into another.

Selector

A selector can be used for dynamic selection and invocation of different services, which all share the same interface. For example a customer support process could use different human tasks implementations during holidays than during regular working days. WebSphere Process Server offers a Web-based interface to enable dynamic updates to the selection criteria and target services, which means that a module that has been deployed at a later time can still be called by this selector component enabling dynamic changes to the integration solution.

Service components

WebSphere Process Server provides business processes, human tasks, business states machine and business rules service components.

Business process

A business process component in WebSphere Process Server implements a Web services Business Process Execution Language (BPEL) compliant process engine. You can develop and deploy business processes that support long and short running business processes and a compensation model within a scalable infrastructure. You can create BPEL models in WebSphere Integration Developer or import from a business model that you can create in WebSphere Business Modeler.

Human tasks

Human tasks are stand-alone components in WebSphere Process Server that you can use to assign work to employees or to invoke any other service. Additionally, the human task manager supports the ad hoc creation and tracking of tasks. You can use existing LDAP directories (as well as operating system

repositories and the WebSphere user registry) to access staff information. WebSphere Process Server also supports multi-level escalation for human tasks including e-mail notification and priority aging. WebSphere Process Server includes an extensible Web client that you can use to work with tasks or processes. This Web client is based on a set of reusable JavaServer™ Faces (JSF) components that you can use to create custom clients or embed human task functionality into other Web applications.

Business state machines

Business state machines provide another way of modeling a business process. This lets you represent your company's business processes based on states and events, which sometimes are easier to model than a graph-oriented business process model. One example would be an ordering process where you can modify or cancel the order at any time during the order process until the order is actually fulfilled.

Business rules

Business rules are a means of implementing and enforcing business policy through externalization of business function. This enables dynamic changes of a business process for a more responsive businesses environment.

Business rule authoring is supported with Eclipse-based desktop tooling. Business analysts can use the Web-based runtime tooling included in WebSphere Process Server to update business rules as business needs dictate without affecting other services.

Imports and exports

An import is a way of representing an external service in the SOA environment. For example, retrieving data from an ERP system can be turned into a service using the enterprise service discovery wizard. An export is the reverse, a way of representing an SOA service to an external service, such as a Web client.

> **Note:** WebSphere Process Server also includes support for business-to-business (B2B) scenarios by including the IBM WebSphere Partner Gateway.

More information

For more information on WebSphere Process Server, refer to:

> http://www.ibm.com/software/integration/wps/

WebSphere Business Monitor Version 6

WebSphere Business Monitor is a Web application that is deployed and run on WebSphere Process Server Version 6. It displays dashboards, which are containers (portals) that enable you to monitor different aspects of business performance.

You can use the WebSphere Business Monitor to capture real-time, work-in-progress items and perform corrective actions by reassigning or suspending activities or processes. You can display real-time data from work items produced as the monitored process is running, and can also retrieve and view the historical data of the process.

Dashboards serve a wide audience, essentially all line-of-business and systems management users in addition to business executives, enabling them to perform these tasks:

► Monitor and manage business performance indicators

► Retrieve information quickly and efficiently

► Personalize the analysis and display of business performance reports, and compress information to focus on the business objectives and the key performance indicators (KPI)

► View business-critical information graphically, using visual cues such as color to improve the probability of timely problem determination and the speed of decision making

► Visualize performance data such as KPIs and metrics, which may be summarized in reports and graphs

► Analyze and investigate business situations by using drill-down capabilities to trace situations to individual events and inspect event details

► Set up actions and alerts that are part of the management phase of a business performance management solution

Each dashboard is composed of one or several data snapshots, referred to as views. The dashboards run within the WebSphere Portal Server environment, and for each dashboard, a portal page is created and a set of views (portlets) are laid out in the portal page. The dashboard administrator can set up the following types of dashboard views in WebSphere Business Monitor:

► **Active instances**—Display the values of all the business measures (KPIs, metrics, stopwatches, and counters) that you defined in the business measures model.

► **KPIs**—Display the values of KPIs relative to their acceptable limits (below limits, within limits, or above limits).

- ► **Gauges**—Display KPI values in the form of a gauge, like a speedometer or tachometer, relative to their acceptable limits or margins.

- ► **Scorecards**—Monitor KPIs grouped by the scorecard perspectives that you select. By default, a scorecards view displays four perspectives: Financial, Customer, Learning and Growth, and Internal Business Process. Within each perspective, you can choose to monitor one or more KPIs.

- ► **Alerts**—Display notifications that are sent when a business situation occurs.

- ► **Dimensions**—Provide a multidimensional view of business performance data. You can pivot on any defined business dimension to analyze different aspects of the historical performance.

- ► **Reports**—Display performance reports relative to a time axis. Such reports typically contain tables and graphs with textual descriptions summarizing the analysis.

- ► **Process diagrams**—Display a process model with visual cues showing the status of each run of the process.

- ► **Organizations**—Display the business organization units and their employees, in the form of a navigation tree. This information is stored on a user registry (such as LDAP) that WebSphere Portal is configured to use, and is not taken from WebSphere Business Modeler.

- ► **Export values**—Enables you to export the values resulting from the running processes to an XML file that can be imported by WebSphere Business Modeler.

More information

For more information on WebSphere Business Monitor, refer to:

http://www.ibm.com/software/integration/wbimonitor/

Summary

In this chapter we introduced the four products that are part of the IBM business process management solution. We will use the four products in our scenario to move a business process application from modeling to implementation to monitoring.

The online Information Center documentation of the four product can be found at:

http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp

3

Case study: ClipsAndTacks

This chapter describes a business process management (BPM) scenario, that shows how a business can use a full IBM WebSphere Business Integration solution to complete the following end-to-end business process lifecycle tasks:

► Model and simulate a business process

► Define key performance indicators

► Develop and test an application to implement the business process

► Deploy and run the application on a server.

► Monitor the application to observe pre-determined key performance indicators

► Import the observed data to make required revisions to the original process model in the Modeler.

The business scenario described in this document has been simplified in order to provide a full description of each stage of the BPM end-to-end process lifecycle. To avoid an overly large and unwieldy document, the authors' focus is on specific tasks, elements, and details, and not on presenting all possible facets of a complex business process. Each IBM software product included in the full BPM cycle has a large number of features and capabilities. For details on all of the available functions of these products, see the help documentation included with each product.

29

Case study: ClipsAndTacks Office Supplies Ltd.

This case study describes a fictional company that is seeking to improve one of its business processes. It shows how a business can quickly respond to the needs of its customers using an IBM business process management solution. The business described in this scenario, ClipsAndTacks Office Supplies Ltd. (abbreviated as *ClipsAndTacks* from now on) is experiencing a business problem that is negatively impacting its bottom line. The company needs to plan and implement a revised process that will address the business problem. To complete this business transformation, ClipsAndTacks will use the component products of IBM's business process management solution.

Background

ClipsAndTacks is a medium-sized office supply company operating in eastern Canada and the northeastern United States. The company has grown slowly and has achieved a significant customer base through its excellent customer service practices and reputation for quality products. Most ClipsAndTacks customers are businesses; ClipsAndTacks does not allow accounts for non-business customers.

Business problem

ClipsAndTacks has been losing customers to Office Market, its main competitor. Office Market is a national office supply chain which provides an online catalog and ordering process for its customers. From Office Market's Web portal, customers can view available products and submit an order 24 hours a day, 7 days a week.

Although it has been losing customers, and consequently revenue, ClipsAndTacks' costs have remained constant relative to the order handling process. They have maintained the same number of customer representatives to avoid introducing further delays into the ordering process due to longer call-waiting times. Each customer representative traditionally receives an increase in their hourly wage each year. The company also continues to maintain separate customer and product management systems, neither of which have been upgraded in several years. Both systems are prone to problems and outages, and require considerable maintenance.

Several customer surveys have indicated that ClipsAndTacks' customers are not satisfied with the ordering process. The telephone ordering procedure is time-consuming, and customers are frustrated at being placed on hold while waiting for the next available representative. Regular customers are frustrated at the amount of time it takes to receive their orders.

The delays are most often caused by the order review process, requiring each order over $500 to be reviewed by an order manager for any credit risk before it is sent on to be filled. Customers repeatedly cite Office Market's online order process as a quicker and more convenient method for ordering their office supplies, and their comparatively quicker delivery time as a major advantage over ClipsAndTacks.

Summary of problems:

► Call center hours of operation are not convenient

► Telephone order submission process is too long

► Order review process delays shipments

► Regular customers, in particular, resent delays due to order reviews

► Company losing customers and revenue

Business objectives

As a result of the customer surveys, ClipsAndTacks' management has decided that the order handling process has to be updated so that it can fill orders in a shorter amount of time. Company management wants to establish an automated process that shortens order turnaround time, especially for trusted repeat customers.

The planned improvements include a new Web-based ordering system, which is a customer's access point to an almost totally automated ordering application. Simply put, it is an order handling application available 24 hours a day over the Internet.

The high-level business objectives of ClipsAndTacks are to increase revenue and reduce costs.

Note: Costs are currently not available for monitoring in WebSphere Business Monitor. We do not monitor and analyze costs in this document.

Specifically, management wants to achieve the following objectives:

► Reduce the average time from when orders are received to the time they are shipped to 3 days

► Achieve an order approval rate of 90% or better

Current order handling process

ClipsAndTacks publishes a product catalog which is mailed to its existing account customers. Customers can only place orders by telephone, with calls routed through the company call center. The call center is staffed by customer representatives and takes calls between the hours of 8:00 a.m. and 5:00 p.m. Eastern Time, Monday to Friday.

For new customers, a customer service representative manually enters the contact and address information to the customer database and assigns a customer number. Existing customers have to provide their customer number so that the representative can retrieve the customer record containing the customer's information. When the customer information is retrieved, the customer provides the details of the order.

All orders are forwarded to an order manager. If an order total is $500 or more, the order manager must review it before sending it to the warehouse to be filled. The order manager assesses the credit risk of each order: if the order is deemed an acceptable risk, it is sent on to the warehouse to be filled; if it is not deemed an acceptable risk, the order manager cancels the order and sends an e-mail notification to the customer. All approved orders are sent to the warehouse to be filled.

Summary of business rules:

► Customers can only order by telephone

► Orders are accepted only from 8:00 a.m. to 5:00 p.m. Eastern Time, Monday to Friday

► Customer service representatives handle all inbound order requests

► All orders are forwarded to the order manager for review

► Orders over $500 must be approved by the order manager

Planned revisions to the order handling process

ClipsAndTacks' management wants to implement an order submission process that will allow ClipsAndTacks to compete with Office Market. In response to a common complaint in the customer surveys, the new process will eliminate the need for contact between customers and customer service representatives when an order is placed. Customers will be able to browse the ClipsAndTacks product catalog and enter their own order information using a Web application. New customers will be able to enter their company information and receive a customer account number immediately.

Customers who have a customer number will be able to enter it and prompt the Web application to retrieve their information and pre-fill the Web form with their address and preferred shipping information. The new Web application, including the product catalog and order form, will be available 24 hours a day, 7 days a week.

When the customer submits the order, a business rules engine will be checked to ensure appropriate action on the order. In response to comments from regular customers, the threshold for order review will be raised to $750. If an order is for an amount under $750, it is approved automatically, pending a check of the account status (that is, are there any outstanding charges against the account that have not been paid). If the account is in good standing, the order is sent for shipping. If the account is not in good standing, it is sent to an order manager for review. If an order is for an amount greater than $750, it is sent for to an order manager for review. Based on the review, the order manager decides whether to send the order for fulfilment or to cancel the order and notify the customer.

Summary of revisions:

► Customers can order online (shorten order process)

► Orders are accepted 24 hours a day, 7 days a week

► Implementation of rules/policy engine

► Threshold for order review has been raised to $750

► Orders over $750 must be approved by the order manager (shorten average order time and increase percentage of approved orders).

Note: We will have to reiterate that we are presenting a simplified process. The outright cancellation of an order is less likely than a request for prepayment, or some other arrangement between the companies. A simplified scenario is presented in order to show details of each phase of the full cycle.

Key performance indicators

Key performance indicators (KPI) are the detailed specifications required to track business objectives. Each KPI is associated with a specific process, and is quantifiable, measurable, and results-oriented.

The ClipsAndTacks management team wants to be able to measure the results of the revised process when it is implemented to ensure that it is helping to meet the company's business objectives. To measure the revised order handling process, management has identified two key performance indicators that will measure the success of the new process. Each of these key performance indicators will comprise at least one metric.

To gauge the performance of the revised process and determine if it helping to achieve the company's business goals, the ClipsAndTacks management team will set the following KPIs:

► **Average order fulfilment time is 3 days or less:**

 – Target: 3 days
 – Lower margin: 1 day
 – Upper margin: 3 days

Required business measures elements:

 – Order Fulfillment Timer stopwatch
 – New Order Trigger
 – Ship Order to Customer Trigger

► **Number of approved orders is greater than 90%**

 – Target: 90%
 – Lower target margin (%): 85.5%
 – Upper target margin (%): 90%

Required business measures elements:

 – Total Orders aggregate metric
 – Shipped Orders aggregate metric
 – Order Counter
 – Shipped Order Counter
 – New Order Trigger (reuse)
 – Ship Order to Customer Trigger (reuse)

Roles

The following key roles take part in the ClipsAndTacks BPM scenario:

► **Customer**

Typically, ClipsAndTacks customers are business professionals who purchases office supplies for a company. Customers are comfortable with Web processes, and expect an order process to be quick and simple. They also expect that orders will be delivered promptly.

► **Business analyst**

ClipsAndTacks employs a business analyst on staff. The business analyst is responsible for understanding the company's existing processes and designing ways to improve those processes. The analyst gathers and documents information about the company's processes by reviewing reports, interviewing employees, and observing performance. The analyst then models and simulates current and planned practices.

The business analyst also defines the business measures, such as the key performance indicators, that will be monitored. In our scenario the business analyst uses WebSphere Business Modeler to model, simulate, and analyze the business process.

► **Order manager (order approver)**

A ClipsAndTacks staff member who has the authority to accept or reject a purchase order. Orders that are over the limit of $750 are routed to the order manager, who either approves or declines the order. ClipsAndTacks management provides guidelines to the order manager that help making the decision.

► **System architect (integration developer)**

The system architect is responsible for designing, building, and testing the implementation of the revised process that has been modeled by the business analyst. In our scenario the architect uses WebSphere Integration Developer to implement the business process and WebSphere Process Server to test the implementation.

► **Application programmer**

The application programmer is responsible for implementing some of the specific services that the systems architect has specified at a higher level. For example, the programmer writes the Web application to be used by the customers, and the Java code to interface with external services.

► **System administrator**

The system administrator is responsible for deploying the finished application in a production environment using WebSphere Process Server and for monitoring the runtime process using WebSphere Business Monitor. The administrator ensures that the component programs operate as they are designed. The administrator records relevant data about the process and produces performance reports for company management.

Throughout this redbook, key roles will be described for each stage in the process.

Subsystems

Figure 3-1 shows the abstract logical subsystems that comprise the solution architecture. The essential subsystems for the ordering process are shown in green with a bold border.
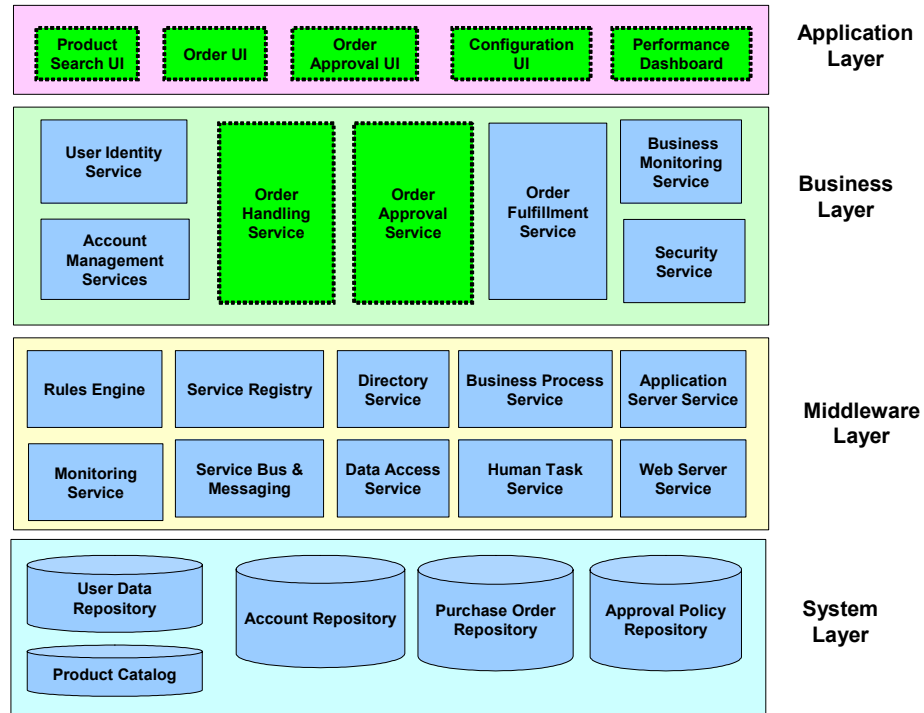
Figure 3-1   Solution architecture

Table 3-1 describes the essential subsystems.

Table 3-1   Abstract logical subsystems that comprise the solution architecture

Subsystem	Description
Product Search UI	Web-based user interface for customers to search and/or select products from an integrated product catalog.
Order UI	Web-based user interface for customers to enter order information, such as quantity and shipping method, or to view the details of existing orders.
Order Approval UI	User interface for the order approver to view and authorize orders.
Configuration UI	User interface for ClipsAndTacks' staff to configure the ordering process, specifically, the order approval process.

Subsystem	Description
Performance Dashboard	Web-based user interface for Business Performance Analyst to query and view business performance results
Order Handling Service	A business process that responds to the Order UI and creates new orders or displays existing order status. The process determines whether an order can be automatically processed according to the procurement policies.
Order Approval Service	A workflow process that is used by the order approval staff to validate and approve orders.

Hardware prerequisites

You must have installed and configured the full BPM suite of products before completing the set of tasks described in this redbook. See the individual product documentation for installation and configuration instructions. System hardware requirements are shown in Table 3-2.

Table 3-2   Hardware requirements

WebSphere Business Modeler	
Processor	Pentium® III 500 MHz (or equivalent) or faster
RAM	768 MB (1 GB recommended)
Disk Space	500 MB to install
Display	Minimum 1024 by 768 resolution
WebSphere Integration Developer	
Processor	Pentium IV 1 GHz (or equivalent) or faster (2.5 GHz recommended)
RAM	1 GB (1.5 to 2GB recommended)
Disk Space	4 GB to install
Display	Minimum 1024 by 768 resolution
WebSphere Process Server	
Processor	Pentium IV 1 GHz (or equivalent) or faster
RAM	1 GB
Disk Space	2 GB to install

Display	Minimum 1024 by 768 resolution
WebSphere Business Monitor	
Processor	Pentium IV 1 GHz (or equivalent) or faster
RAM	1 GB
Disk Space	2 GB to install
Display	Minimum 1024 by 768 resolution

More information

For more information on system requirements, refer to these Web sites:

► WebSphere Business Modeler:

   http://www.ibm.com/software/integration/wbimodeler/advanced/sysreq/

► WebSphere Integration Developer:

   http://www.ibm.com/software/integration/wid/sysreqs/

► WebSphere Process Server:

   http://www.ibm.com/software/integration/wps/sysreqs/

► WebSphere Business Monitor:

   http://www.ibm.com/software/integration/wbimonitor/requirements/

Summary

In this chapter we introduced the ClipsAndTacks company. We will use their order handling business process to demonstrate how the IBM business process management products can be used to model, implement, run, and monitor the ClipsAndTacks application.

Part 2

Modeling the business

In this part we describe the modeling activities of the ClipsAndTacks business analyst.

We start with the model of the current business, *Order Handling (Current)*, and simulate its behavior to find out the bottleneck. Then we describe how to improve the model to provide for a better customer response time and for more orders approved. This model is referred to as *Order Handling (Future 1)*. We simulate and analyze the new model.

Finally we decide what business measures to define and what key performance indicators (KPI) to measure when we implement the new model in WebSphere Process Server and monitor the execution using WebSphere Business Monitor.

4

Modeling the current business process

This chapter describes how the ClipsAndTacks order handling process, *Order Handling (Current)*, was modeled and imported into WebSphere Business Modeler.

The key to a successful re-engineering of a business process is to thoroughly understand the details of the existing process and to accurately forecast the results of the changes to that process. In order to analyze and predict the outcome of a revised process, a business first needs to create an accurate representation of the process with a model and then study how that process performs under different conditions.

A common error that businesses often make when they set out to revise their processes is failing to fully investigate and understand their current process and their objectives in re-engineering that process. Without devoting the necessary time to study, analyze, and plan, many businesses find that either they have not adequately addressed the original problem, or they have simply exchanged a recognized problem for another unanticipated one.

41

Introduction to modeling the current business process

Before implementing a revised order handling process, the company must fully document, model, and analyze the current process so that it can fully understand where the business problems lie and set realistic targets for the process improvement.

At ClipsAndTacks, the lead business analyst has been given the task of designing the revised order handling process. On its surface, this step seems fairly straightforward. To ensure that his model is as accurate as possible, The business analyst must first document and sketch the current process flow, and list all of the key resources (people, equipment, material), business items (documents, records, products), and business rules (decision logic) included in the process.

> **Note:** For detailed instructions on how to create each of the elements described in following sections, refer to the help documentation included with WebSphere Business Modeler.

Documenting the current process

For the remaining of this chapter, you are now the business analyst.

The first step is to investigate and fully document the current process. By reviewing available data reports and interviewing each employee involved in the process, you can gain an understanding of how the process currently works, and where the problem areas lie. For ClipsAndTacks, you must monitor each stage of the process, observing the employees' interaction with customers, and the progress of orders through the complete order handling process.

You have to focus on recording all key aspects of the process:

► What is the process flow?
► What are the resources used in the process?
► What are the business items acted on by the process?

As you study the process and gather your data, record the following:

► Activity inputs and outputs
► Task variations and when those variations occur
► Alternative tasks
► Complete task descriptions
► Roles associated with tasks

After you have gathered sufficient data, record the findings and map out a rough flow diagram of the current process. Make note of the output at each stage of the process, indicating stages where there is more than one possible outcome. List all of the activities, resources, roles, and business items involved in the process.

Process flow

When you have gathered all available data and observed the current process in action, you will be able to write out the process flow, including any spot where there is more than one possible outcome. At ClipsAndTacks, the current order handling process flow takes place as follows:

1. **The customer service representative receives the call from the customer.**

2. **First determine if the customer has an account.**

   2.1 If the customer has an existing account:

   – Enter the account number.
   – Enter the order information.
   – Approve order or send for review.

   2.2. If the customer does not have an account:

   – Enter customer information and assign an account number.
   – Enter the order information.
   – Approve order or send for review.

3. **If the order is approved:**

   – Send the order to the warehouse.
   – Record order in order records database.
   – Issue packing slip.
   – Ship the product.

4. **If the order is sent for review:**

   – Review the order manually.
   – Determine if the order is an acceptable credit risk.

   4.1 If the order is an acceptable credit risk:

   – Send the order to the warehouse.
   – Record order in order records database.
   – Issue packing slip.
   – Ship the product.

   4.2 If the order is not an acceptable credit risk:

   – Cancel the order.
   – Send an order cancellation notification to a customer.

At this stage, you have enough information to sketch out the process. You can use WebSphere Business Modeler to sketch out the rough process, or simply draw it using pencil and paper. The process flow for the Order Handling process might look as shown in Figure 4-1.
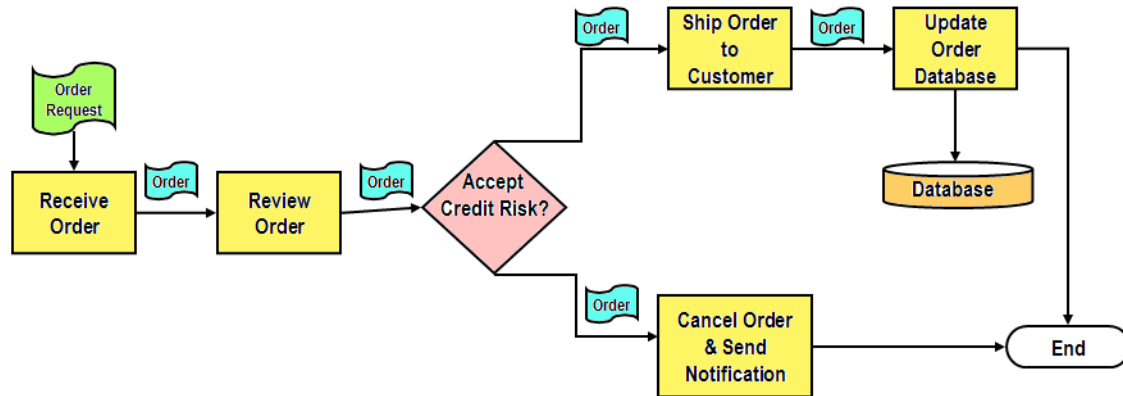

Figure 4-1   Manual process flow draft

Business items

Review the flow to determine the business items that are acted on during the process. Remember that business items are business documents, work products, or commodities that are transformed in business operations. You can model as a business item anything that is created, assembled, inspected, tested, modified, or worked upon. Business items undergo changes as they are passed from one process step to the next.

For the ClipsAndTacks order handling items, the following business items are acted on in the current order handling process:

► Request (a verbal request from the customer)

► Order (a list of products requested by the customer with the shipping and billing information)

► Customer record

► Packing slip with packaged product

► Cancellation notification to a customer

Resources

A key to documenting any process is determining the resources and roles required to complete each of the activities. Resources represent the people, equipment, or material used to perform a project or a task. Resources are not the same as business items. The objects that undergo changes and are passed from one process step to the next should be modeled as business items, whereas the things that are performing the work or are required prerequisites for this work, such as machines, fuel, vehicles, or skilled personnel, should be modeled as resources.

Resources are represented by the following people roles and subsystems in the ClipsAndTacks order handling process:

► Customer service representative
► Order manager
► Shipper
► Customer records system
► Product catalog system
► Order management system

Timetables

An accurate model must also take into account the schedules of the resources involved in the process. In WebSphere Business Modeler, you can define timetables that specify what times certain resources are available. The following timetables are required to indicate the work hours of the key roles in the current order handling process:

► Day shift
► Weekend

Creating the current process diagram using the Modeler

In this section we describe how to create the model of the current process flow.

Detailed steps to create the model

Business modeling is an iterative process, requiring the business analyst to continually revise the process as he or she gains a deeper understanding of the goals, requirements, and individual activities involved. The business analyst must continue to meet with subject-matter experts to gather information and validate draft models.

The current order handling process diagram reflects the current process flow as you have documented it.

Because this redbook is not focused on just the modeling of the process, but instead it has the goal of documenting the complete end-to-end business process management lifecycle, we will not go through each step of creating a current process model in this chapter.

The detailed documentation on how to use Modeler is available in the help documentation included with the product. In Chapter 6, "Modeling the Future 1 business process" on page 83, we will go through the detailed steps for modifying an existing process model. In this chapter we import the current order handling process model that was previously built by the business analyst.

Modeler quick guide

This redbook is not a complete guide to WebSphere Business Modeler V6. However, we provide here a few simple guidelines for working with the Modeler.

Start the Modeler using *Start → Programs → WebSphere Business Modeler → WebSphere Business Modeler*.

Perspectives

You most often work in the Business Modeling perspective. You may also use the Resource perspective to see the underlying physical files.

Modeling mode

The Modeler supports a number of modes that provide more details and capabilities:

▶ Basic—Focus on purely business tasks
▶ Intermediate—Adds input and output details and formal expressions
▶ Advanced—Models that form the basis of software applications
▶ WebSphere Business Integration Server Foundation
▶ WebSphere MQ Workflow
▶ WebSphere Process Server

The first three modes are for modeling the processes, business items, resources, organization, and so forth. The last three modes enable you to define details that will be used when the processes are exported for usage in one of the three products.

For our scenario we work using the *Advanced* mode for modeling, and the *WebSphere Process Server* mode before exporting to Integration Developer and process Server. The mode is set by selecting *Modeling → Mode → Advanced* (for example).

Help and tutorial

The Modeler provides extensive help. Select *Help → Help Contents* to open the help facility. If you are a beginner, expand *WebSphere Business Modeler Advanced* and select *Samples and tutorials → Tutorial: Quickstart*. Then go through the complete tutorial to become familiar with the Modeler.

Screen layout

Click the *Apply 4-pane layout* icon to see the Project Tree, and edit area, the Outline, and Attributes/Errors. This layout gives you access to all important tasks.

Importing the current process model using the Modeler

Listed below are the detailed steps for the importing of the existing process model that was previously built using the Modeler product.

You can import models or definitions from several different formats into the WebSphere Business Modeler Version 6.

Specifically, you can import files in the following formats:

► Business Integration Modeler project (.zip)
► WebSphere MQ Workflow (.fdl)
► WebSphere Business Integration Workbench 4.2.4 (.org)
► Delimited text files (.csv, .txt)
► XML Schema Definition (.xsd)
► Microsoft® Visio® (.vdx)
► WebSphere Business Integration Modeler XML (.xml)
► Monitoring result (.xml)
► CEI Event Catalog
► Web Service Interface (.wsdl)

We import the current order handling process model that was previously modeled and exported as a WebSphere Business Modeler project zip file.

The Business Integration Modeler project ZIP file can be found in the sample code available with this redbook (see Appendix C, "Additional material" on page 439):

```
SG247148\sampcode\model\ClipsAndTacks Chapter 4 Solution.zip
```

To import the ZIP file, complete the following steps:

▶ In the Project Tree, right-click and select *Import*. The Import wizard appears.

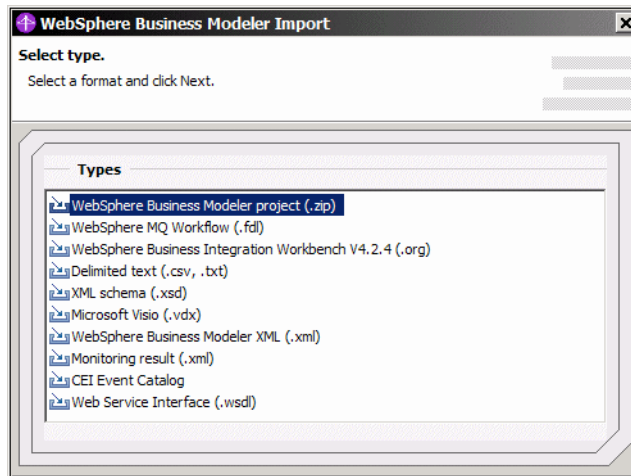▶ Select *WebSphere Business Modeler project* and click *Next* (Figure 4-2).

Figure 4-2   Select model import type

▶ Click *Browse* to select the source directory that contains the file you want to import (Figure 4-3).
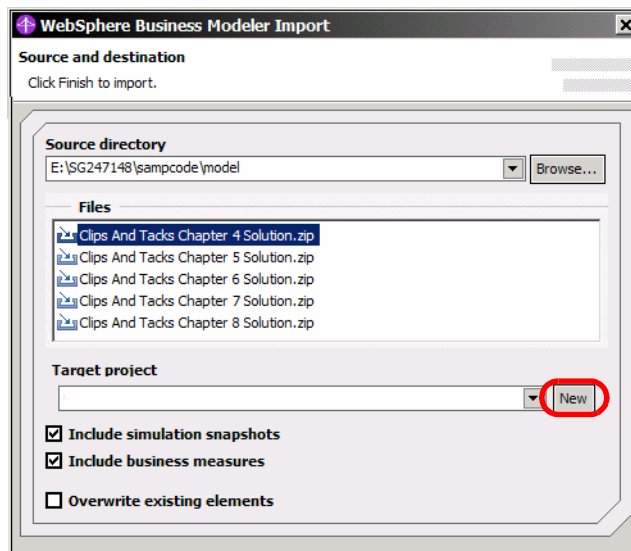

Figure 4-3   Importing a model

► In the Files list, select the ZIP file containing the project or element that you want to import. Select the `Clips And Tacks Chapter 4 Solution.zip` file.

► Select *Include simulation snapshots* and *Include business measures*.

► Optionally select *Overwrite existing elements* to replace any of the elements with the same name as an element being imported. Otherwise, you will be warned when an element is about to be overwritten.

> **Note:** If you choose to overwrite a catalog, be aware that the entire catalog and all of its contents will be deleted and the contents of the new catalog will be imported instead.

► In the Target project field, select an existing project from the drop-down list or click *New* to create a new project `Clips And Tacks Chapter 4 Solution` (Figure 4-4).
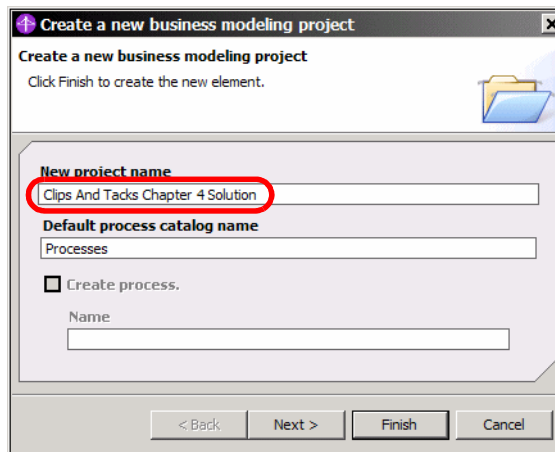

Figure 4-4   Creating a new project ClipsAndTacks

► Click *Next*. Select the default *Free-Form Layout*, then click *Finish*.

► The project is created and its named filled into the dialog. Click *Finish* to import the ZIP file.

► If there are any errors or warnings during the import process, click *Details* to read them.

The project files have been imported into the project you specified. The Project Tree view of the project is shown in Figure 4-5.
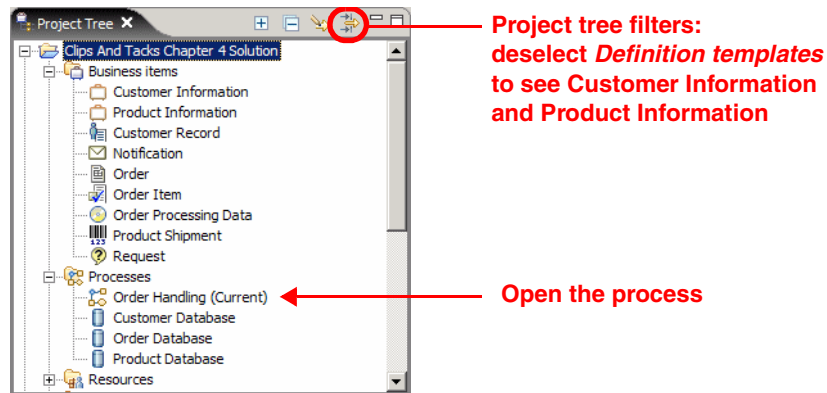
Figure 4-5   Project Tree of the order handling process

Order Handling (Current) process

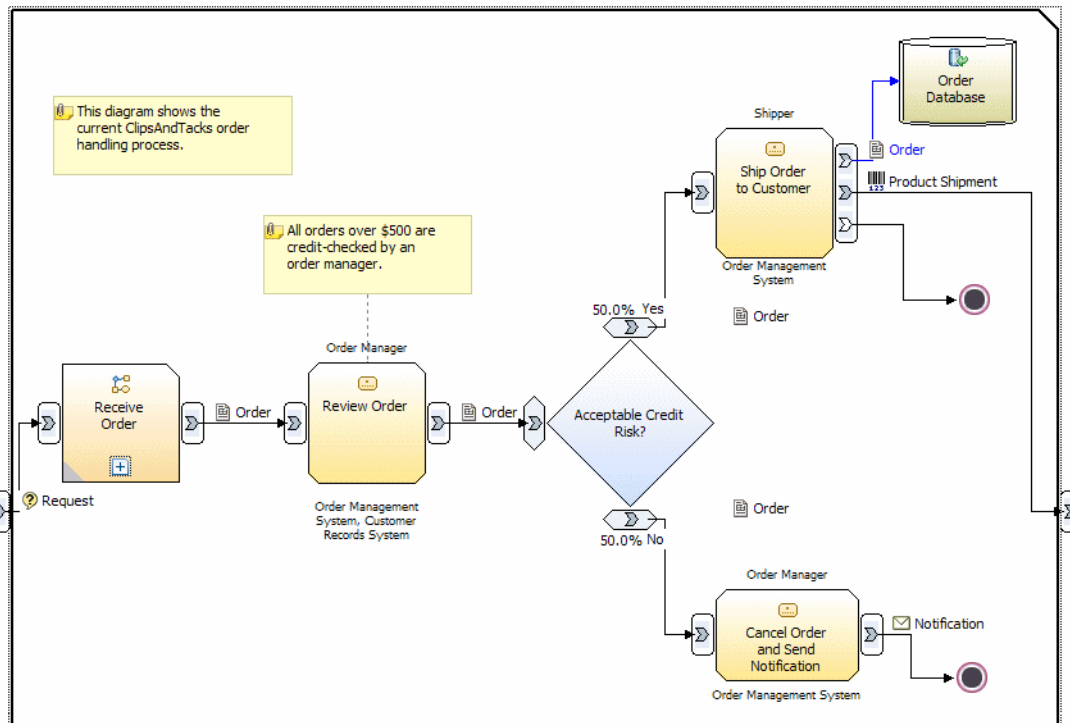The imported current order handling process model is shown in Figure 4-6.

Figure 4-6   Current order handling process model (compressed to fit)

The activities in the order handling process are:

▶ **Receive Order**—A local process that handles the interaction of the customer with an agent.

▶ **Review Order**—An order manager reviews all orders over $500 and decides if the order should be approved or declined.

▶ **Ship Order to Customer**—If the credit risk is acceptable the order is processed, recorded in an order database, and shipped.

▶ **Cancel Order and Send Notification**—If the order is declined a cancellation notification is sent to the customer.

The last three activities are so called *human tasks*, performed by a person. The input to the process is a customer request, and the output is the product shipment.

The default probability for each branch of Acceptable Credit Risk is 50%.

Embedded process

To view the embedded process, Receive Order, select the process and *Edit* (context menu). The sequence of the Receive Order process is shown in Figure 4-7.
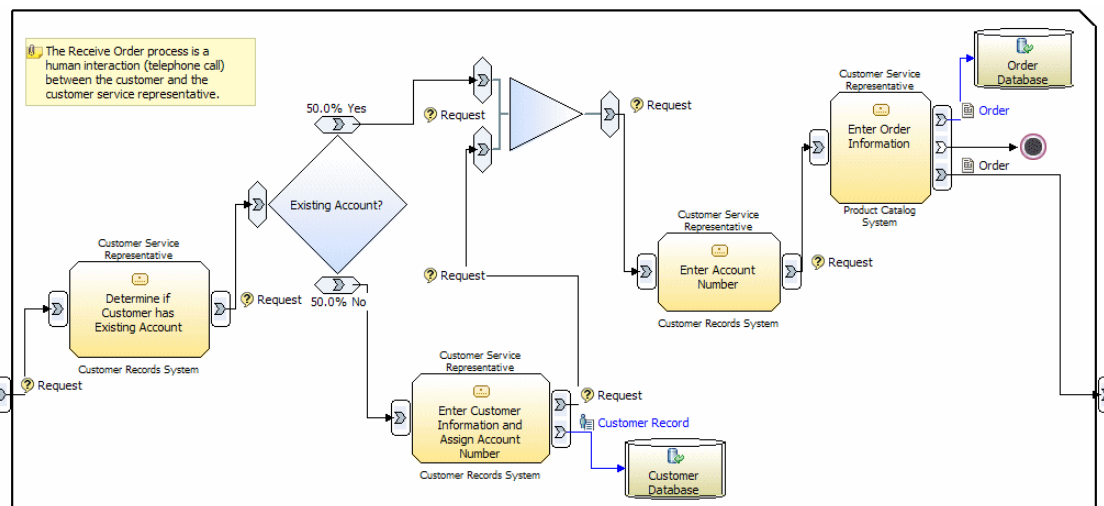

Figure 4-7   Receive Order subprocess (compressed)

To go back to the order handling process, right-click in an empty space in the diagram and select *Parent* (context menu).

Business items

The business items that used during the process are listed here:

▶ **Order**—The Order is the main business item that flows through the whole process (Figure 4-8). An order consists of an `OrderNumber`, an `OrderStatus`, the `TotalPrice` of the order, a `Customer` with all its information, the `OrderItems` (which products), and a processing field (`automaticApproval` coming from **Order Processing Data**).
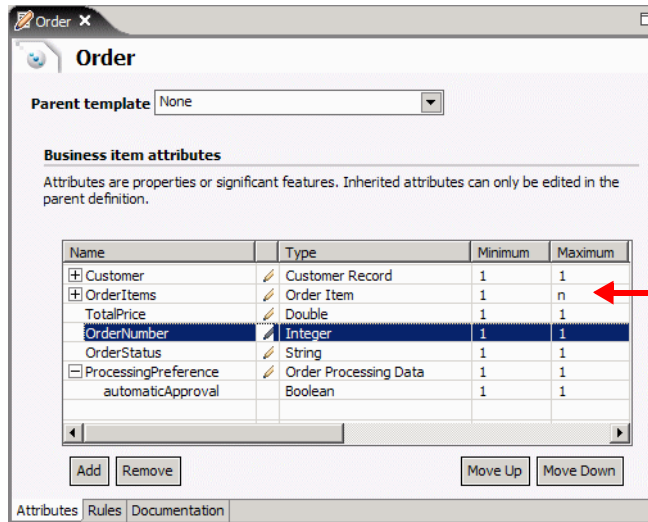

Figure 4-8   Business item: Order

▶ **Customer Record**—The data kept for a customer is composed of basic data coming from **Customer Information** and additional attributes, `Classification`, `Rating`, and `AvailableCredit` (Figure 4-9).

Figure 4-9   Business Item: Customer Record

▶ **Order Item**—An order item consists of a `Quantity` of a product ordered. The product attributes come from **Product Information** (Figure 4-10).

Name		Type	Minimum	Maximum
ProductName	ᵻ	String	1	1
ProductNumber	ᵻ	String	1	1
Price	ᵻ	Double	1	1
Description	ᵻ	String	1	1
Quantity	⁄	Integer	1	1

Product Information

Figure 4-10   Business item: Order Item and Product Information

▶ **Product Shipment**—Similar to the order, consist of a `Packing Slip Number`, `OrderNumber`, `TotalPrice`, `Customer`, and `OrderItems`.

▶ **Notification**—Consists of the `email` address and the `text` to be sent to a customer when an order is declined.

▶ **Request**—This is the request coming from a customer to order some products.

The business items Customer Record, Order, and Order Item will be used for the database design when we implement the application in a WebSphere server.

> **Note:** These Customer Record business item already includes some fields that will only be used later in our scenario (`Classification` and `Rating`).

Description of some business item attributes
Most of the attributes are self-explanatory, the others are explained here:

▶ `OrderStatus`—The status of an order during the process: `NEW`, `APPROVED`, `DECLINED`, `SHIPPED`.

▶ `automaticApproval`—An order can be automatically approved if the total price is below a certain value. This attribute is either `false` or `true`.

▶ `Classification`—later in our scenario customers are classified as `REGULAR`, `SILVER`, or `GOLD`. The total price for automatic approval will depend on this classification.

▶ `Rating`—The credit rating of a customer as retrieved from an external service. This value will be used by the order manager to decide if an order should be approved or declined.

▶ `AvailableCredit`—The available credit of a customer. This value will be used by the order manager to decide if an order should be approved or declined.

Resources

The resources defined for our scenario are:

► Customer service representative—Takes the phone call from a customer.

► Order manager—Decides if an order should be approved or declined.

► Shipper—Package the products and ship to the customer.

► Customer Records System, Order Management System, Product Catalog System—Computer applications used for customer information, orders, and the product catalog.

► Day Shift and Weekend—Time tables.

Organizations and classifiers

No organizations and classifiers have been defined for our scenario.

Summary

In this chapter we described how ClipsAndTacks modeled their current business to get a better understanding of the business process, the activities, and to prepare for simulation and analysis.

5

Simulating and analyzing the current process

This chapter describes how the ClipsAndTacks current process is simulated and analyzed in the Modeler.

The key to a successful simulation and analyzing of the current process is to describe correctly what information we should gather from the real world business process to simulate a process.

The first part of this chapter provides an example of a set of information required to implement a representative simulation and analysis.

The second part of this chapter shows a demonstration on how to gather and analyze information provided by the simulator of Websphere Business Modeler.

55

Overview of process simulation

These are the major steps to run a simulation of a process:

► Define resources and probabilities:

– Corporate strategies
– Process flow
– Human resources needs and costs matrix
– Duration matrix
– Human resource availability matrix
– Probabilities on decision matrix
– Probabilities on output matrix

► Define a simulation profile and attributes related to the simulation runs

► Enter all simulation attributes in the Modeler

► Run a simulation snapshot

► Analyze simulation results

ClipsAndTacks process assessment for the simulation

After modeling the business process, you can use WebSphere Business Modeler to simulate the running of the process. Simulating allows you to assess the performance of the process, generate statistics about its execution, and pinpoint potential areas of improvement. A process simulation is a simulated performance of a real world business process in a virtual environment.

Before simulating the current process, the lead business analyst in charge of this project must organize workshops with operational officers to gather the following information about the real wold business process:

► The corporate strategy, also called business objectives, is described in Chapter 3, "Case study: ClipsAndTacks" on page 29.

► The documentation and the design of the processes are described Chapter 4, "Modeling the current business process" on page 41.

As a result of these workshops a number of documents are produced.

Role resources matrix

The role resources matrix (Figure 5-1) shows the number of people for a specific role and resources assigned to activities. This matrix also shows the cost by roles. Usually the cost is defined by the salary divided by the unit of measure, an hour in our case.

Resources / Activity	Costs USD per hour	Determine if Customer has Existing Account	Enter Customer Information and Assign Account Number	Enter Account Number	Enter Order Information	Review Order	Cancel Order and Send Notification	Ship Order to Customer
Customer Service Representative	11.00	1	1	1	1			
Order Manager	20.00					1	1	
Shipper	10.00							1
Order Management System	0.00					1	1	1
Customer Records System	0.00	1	1	1		1		

Figure 5-1   Roles matrix: Cost and resource/role per activity

Duration matrix

The duration matrix (Figure 5-2) shows the duration of human tasks for a specific role and a specific activity. In this example, there is only one role per activity, but there could be multiple human roles for one activity.

Be careful, the total duration is not equal to the sum of the resources durations, because some resources are used in parallel.

	Determine if Customer has Existing Account	Enter Customer Information and Assign Account Number	Enter Account Number	Enter Order Information	Review Order	Cancel Order and Send Notification	Ship Order to Customer
Activity duration	20 sec	6 min	20 sec	12 min	20 min	2 min	16 min
Resources / Activity							
Customer Service Representative	20 sec	5 min 45 sec	10 sec	12 min			
Order Manager					20 min	2 min	
Shipper							15 min
Order Management System					1 min	2 min	1 min
Customer Records System	10 sec	15 sec	5 sec		1 min		

Figure 5-2   Duration matrix: Duration of activity by role and resources

Availability matrix

The human resources availability matrix (Figure 5-3) shows the timetables assigned to human roles. In our business case, only one timetable named *Day Shift* is used (described as a resource in the Modeler).

The day shift is defined as:

► 9 working hours a day
► Working days are Monday to Friday
► Working hours 8:00 AM to 5:00 PM

Timetable / Resources	Customer Service Representative	Order Manager	Shipper	Order Management System	Customer Records System
Day Shift	1	1	1	1	1

Figure 5-3   Availability matrix: Time table per role

Note: the Weekend timetable is used as an exemption for the day shift timetable.

Decision probabilities

One of the process model components is a decision. You can assign a probability on decision choices (for example: is a customer an existing customer?), which determines the method of selecting a path through the process.

Decision / Probability	Yes	No
Acceptable Risk Credit	70%	30%
Existing Account	50%	50%

Figure 5-4   Decision matrix: Probability (yes/no) per decision

> **Note:** Websphere Business Modeler allows to define probabilities on output criteria, which are the allowable combinations of outputs for an activity. In our case, you could find an example of multiple outputs with a probability defined in the activity "Ship Order to Customer".

Simulation profile information

The assessment of the current process provides information about the duration of the benchmark to reflect a representative simulation. For our scenario, we define:

► The number of tokens for the simulation: `270`

    – A token represents a unit of work that is received by a process and transferred between different activities in the process flow. By specifying token creation settings, you define the quantity and rate of inputs that the process handles in a simulation run. For our example, a token represents a request. Assuming that we get 30 requests per hour, there will be 270 requests per 9 hour day.

► The maximum duration of the benchmark: `365 days`

    – Specify the maximum duration that a simulation will run. The maximum duration is the real time during which the simulation occurs.

► The start date and time of the simulation: `Monday, October 24, 2005 08:00:00 AM`

    – Specify start and end dates and times to define the virtual time in which the simulation takes place.

► The time measurement unit for results: `Minutes`

    – Select a unit of time that will be used for defining time-related distributions and for recording the results of process simulations.

► The distribution model for requests: `uniform distribution by minutes`

    – Measurements using any variable, even the same variable on the same subject, result in different outcomes. The pattern of different outcomes is called the distribution, which can be described mathematically and graphically. The distribution describes the relative number of times each possible outcome will occur in a number of trials.

> **Note:** For more information about distribution models, refer to the product documentation: *Simulation processes → Setting Simulation → Attributes → Specifying token creation setting*s

► Steady delay for the process: `0 minute`

    – Specify a period which must elapse in the virtual time of a simulation run before statistics gathering begins.

- ► Method of selecting an output path: `Base on probabilities`

  - – Select a method that the process simulator will use to determine which processing path to follow when a process or an activity in a process has more than one set of outputs defined by output criteria.

- ► Recurring time interval for bundle creation: `2 Minutes` (270 in 9 hours)

Populate the simulation environment

Simulation attributes allow you to configure a process so that it behaves in a manner that resembles a real world business process.

A simulation environment is divided on four layers:

- ► The global simulation preferences (select *Windows → Preferences → Business modeling → Simulation*) hold the default values for the local preferences of any newly created simulation snapshot.

- ► The local simulation (process default element) preferences are applied as default values for the simulation attributes of any new simulation profiles that you create for the current snapshot.

- ► The top-level process simulation attributes (process snapshot element) where you define the behavior of a process as a whole during a simulation.

- ► The low-level activity simulation attributes (process element or process snapshot element) where you define the behavior of an activity in a simulation.

> **Note:** For more information on simulation level and informations, refer to the product documentation under *Simulating Processes*.

With all information acquired, the first task is to assign a timetable and cost elements to the human role resources.

> **Note:** You can import the model populated with simulation information from:
>
> `SG247148\sampcode\model\Clips And Tacks Chapter 5 Solution.zip`
>
> See "Importing the current process model using the Modeler" on page 47 for instructions on how to import a model.

Populate role resource information in the project

Roles add additional characteristics to resources. For example, an Employee resource could have the role of Customer Service Representative, Order manager, or Shipper.

You can specify the roles that are required to complete any task that you model. You can add costs and availability to roles. For example, a certain role may cost $20 an hour and be available only from Monday to Friday.

Customer service representative

In the Project Tree (Figure 5-5), navigate to the *Resources* and open the role that you want to modify (you have to insert data for the three resources).
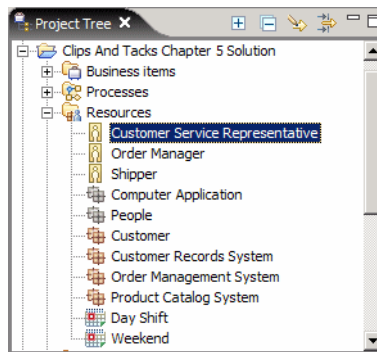
Figure 5-5   Open a resources role

In the Customer Service Representative editor area (Figure 5-6), insert the cost following this sequence of action:

► Select the *Costs* tab and click *Add.*

► Select *Cost per time unit* and click *OK*. The *Cost per unit time* appears in the list.

► Enter the cost value (11.00 USD for the Customer Service Representative).

► Click *Edit* to update the time unit, select *1 hour*, and click *OK*.

► Save the changes.
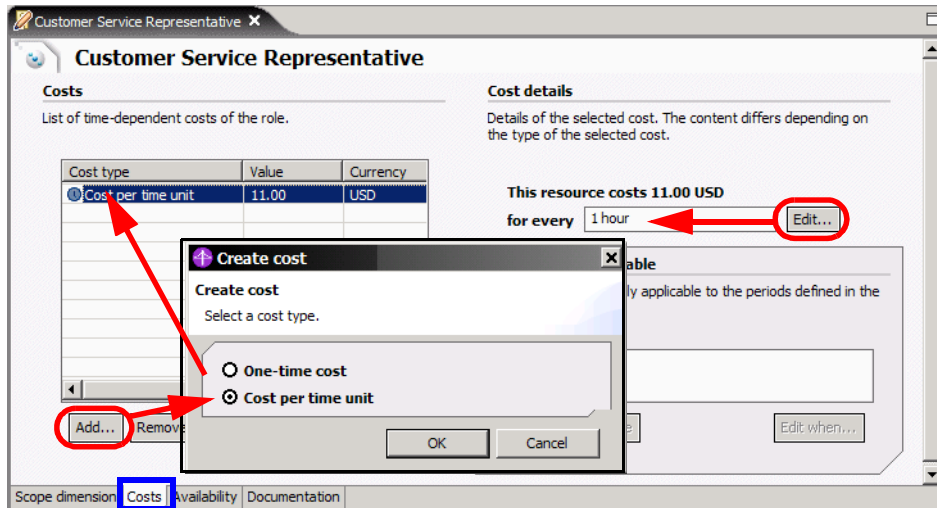
Figure 5-6   Role resource costs

In the Customer Service Representative editor insert the duration (Figure 5-7):

▶ Select the *Availability* tab, and click *Add.*
▶ Select *Day Shift*, and click *OK.*
▶ Save the changes.
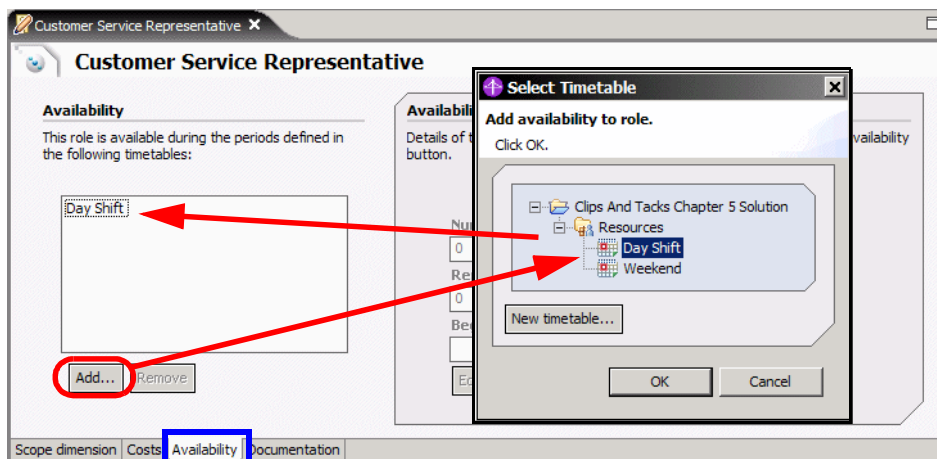
Figure 5-7   Role resource availability

Order Manager and Shipper

Perform the same operation for the Order Manager and Shipper roles and set the cost to $20 and $10, and the availability to Day Shift.

Populate duration information in the process

You can specify the processing time for an element such as a task or process. This is the time while the element is actively executing, rather than the elapsed time, which may include delays while waiting for a resource. For a task, you can also specify the maximum amount of time that the task should wait for a resource before failing.

In our case we will assign two types of duration information, one for the activity and one for the human tasks (role resource).

To be able to select individual activities, open the process diagram for the Order Handling (Current) process.

In the process flow select the activity, and in the Attributes view, select the *Duration* tab and enter the duration value (Figure 5-8).
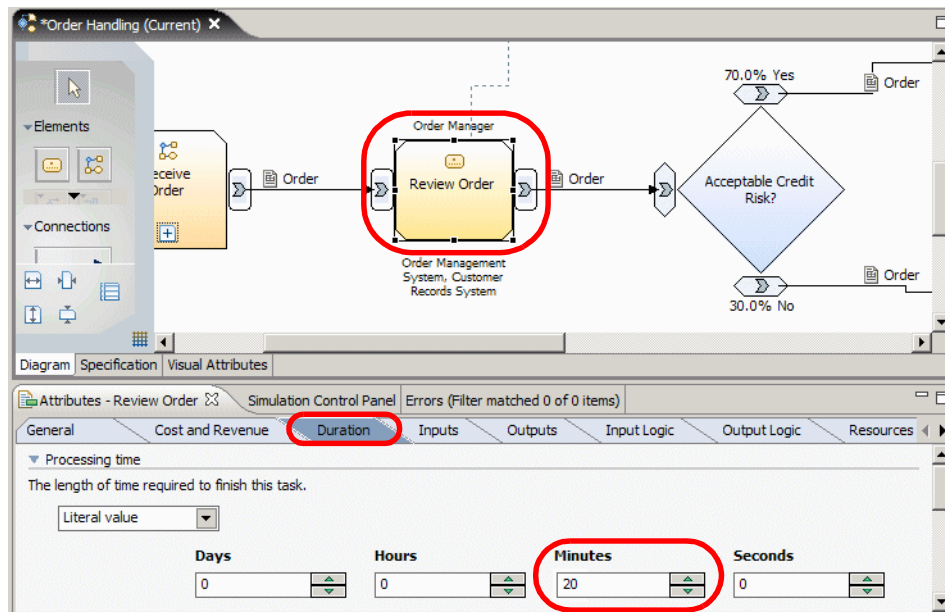
Figure 5-8   Activity duration

To populate the resource (human task) duration time (Figure 5-9):

- ▶ Select the activity in the model, for example, *Review Order*.
- ▶ Select the *Resources* tab in the Attributes view.
- ▶ Click Add to add a role.
- ▶ Click ... in the *Role* column to select the role, for example, *Order Manager*.

► Click ... in the *Time required* column and select the duration time, for example, 10 minutes.
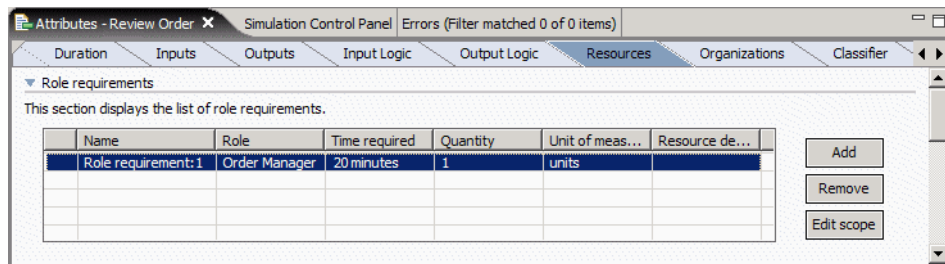► Save the changes.

Figure 5-9   Activity role duration

The activity duration can have an overhead time on top of the human role duration time (this is the case for the Enter Account Number activity).

> **Important:** in this example, you define duration times in the original flow, which is the master for the simulation snapshots. It is possible to change those values for each simulation without changing data in the original flow. This particularity enables you to run multiple simulations without alteration of the original values that reflect the real world business process.

Validate the simulation data

You can validate in WebSphere Business Modeler if you have populated all the required information. Select the project and *Static Analysis → General Analysis → Matrix Analysis* (context). This function displays the roles by activities (Figure 5-10):

► Select *Activity for rows*, *Role* for columns, and click *Next*.
► Select *Processes*, select the process you want to analyze, and click *Finish*.
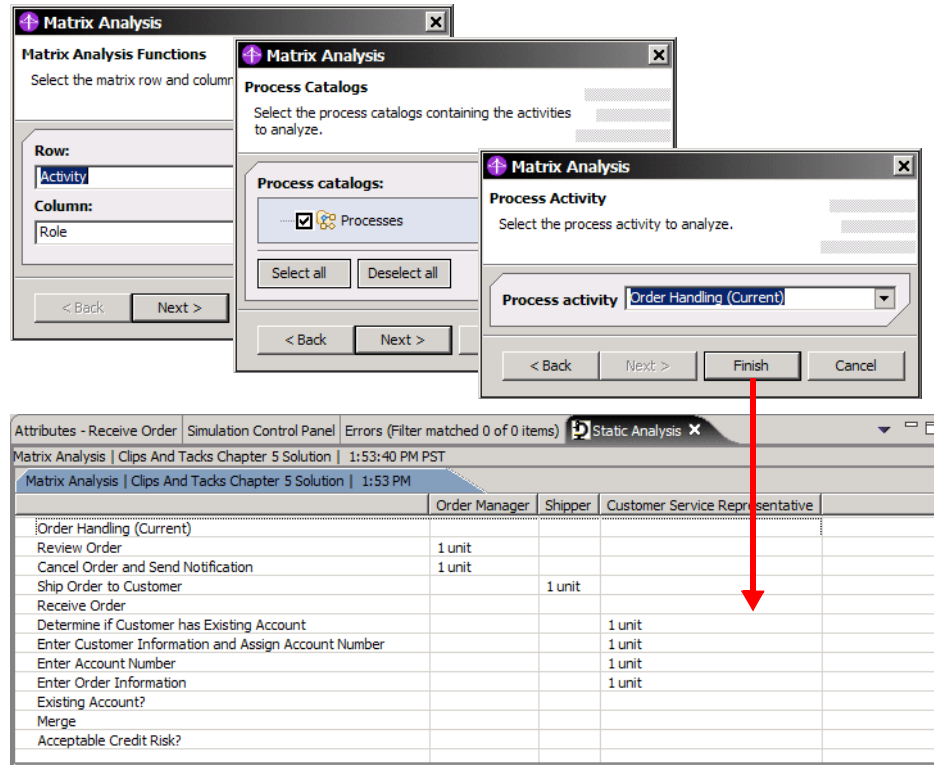
Figure 5-10   Static analysis: Activity versus role

Tip: After you create the simulation snapshot, you will be able to validate your process elements including durations, probabilities, and costs. See "Verify the profile specification" on page 71 for more information.

Populate probability information in the decision nodes

You can add a probability to each branch of a decision to indicate the probability of that branch.

The probability for each branch is set to 50% when the first two branches are created. You change original probabilities in the process to reflect the real world business process. In the process flow, select the Acceptable Credit Risk activity in the process flow. In the Attributes view, *Output branches* tab, set the probability values for the Yes and No branches (Figure 5-11). Note that you can also overtype the values in the process diagram.

Figure 5-11   Decision activity probability

Populate probability information in activity nodes

If you have more than one output criterion, you can optionally add probabilities to indicate how often each of the criteria occurs. For example, you could specify 60% for the more likely output and 40% for the less likely output. If you specify a probability for one output criterion, you must specify probabilities for all. If you do not specify probabilities, each output criterion is considered equally likely when you create a simulation snapshot.

For example, select the Ship Order to Customer activity in the process flow. In the Attributes view, Output Logic tab, you can see the three outputs as equal (Figure 5-12). To assign probabilities you would have to add rows and enter a probability for each output.

In our model we do not assign probabilities to the output of activity nodes.

Figure 5-12   Activity output logic probability

Once you have populated every real world business process value in the simulation environment you are ready to create a simulation snapshot. A snapshot reflects the real world simulation including your assumptions.

Simulating the current process

In this section we simulate the current process and analyze the results.

Process instance simulation

Before running the simulation on your process, it is necessary to build a simulation snapshot and add information to get an accurate simulation.

The simulation information includes:

- ► The number of tokens for the simulation: **270** (one day)
- ► The maximum duration of the benchmark: **365 days**
- ► The start time of the simulation: **Monday, October 24, 2005 08:00:00 AM**
- ► The time measurement unit for results: **Minutes**
- ► The distribution model: **uniform distribution by minutes**
- ► The random number seed: **1**
- ► The steady delay for the process: **0 minute**
- ► The method of selecting an output path: **Base on probabilities**
- ► The recurring time interval for token creation: **2 Minutes** (270 a day)

Creating a simulation snapshot

When you simulate a process, the tool adds a simulation snapshot as a child element of the process in the project tree. A simulation snapshot is a record of the complete process model at the moment when you simulated the process. This record contains a copy of all the elements of your project that the process may use, such as business items, resources, and global tasks. You may want to create multiple simulation snapshots for the same process after making changes to the project or to the process itself, so that you can compare the effect of these changes.

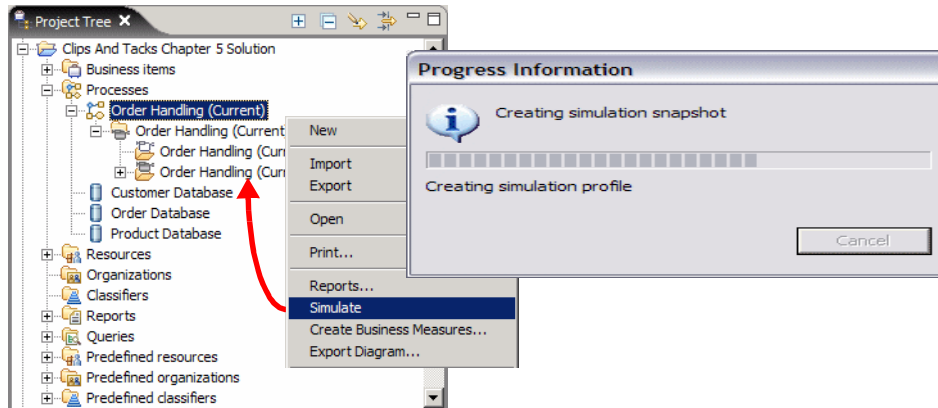To create a simulation snapshot, select the Order handling (Current) process and *Simulate* (Figure 5-13).

Figure 5-13   Creating the simulation snapshot for a process

Within the simulation snapshot, the tool also creates tho folders:

▶ **Defaults**—The defaults folder contains a set of local preferences for simulation attributes. When you create a new simulation profile for a simulation snapshot, the values specified in the local preferences are used for the simulation attributes of the process and activities within the process. The initial values of the local preferences are inherited from the global simulation preferences (*Windows → Preferences → Business Modeling → Simulation*).

▶ **Profile**—Each simulation snapshot contains an initial simulation profile. The simulation profile contains a copy of the process model at the time that you created the simulation snapshot. You can customize the process contained in this simulation profile, and you can create additional simulation profiles within the same simulation snapshot. Typically, you would create multiple simulation profiles for a simulation snapshot when you are experimenting with changes to the fields in the simulation profile, to determine the effect on process results.

After you create snapshot you have to populate the system with your simulation data. The simulation profile diagram opens.

Simulation snapshot defaults

Open the Order Handling (Current) Defaults and specify the values for your simulation as shown in Figure 5-14.

▶ In the *General* tab select:

  – Resource always available: *False*
  – Disable resource allocation: *False*
  – Use resource time: *True*

▶ In the *Token* tab select:

– Total number of tokens: **270**
– Random time trigger: *Uniform distribution* and *Minutes*
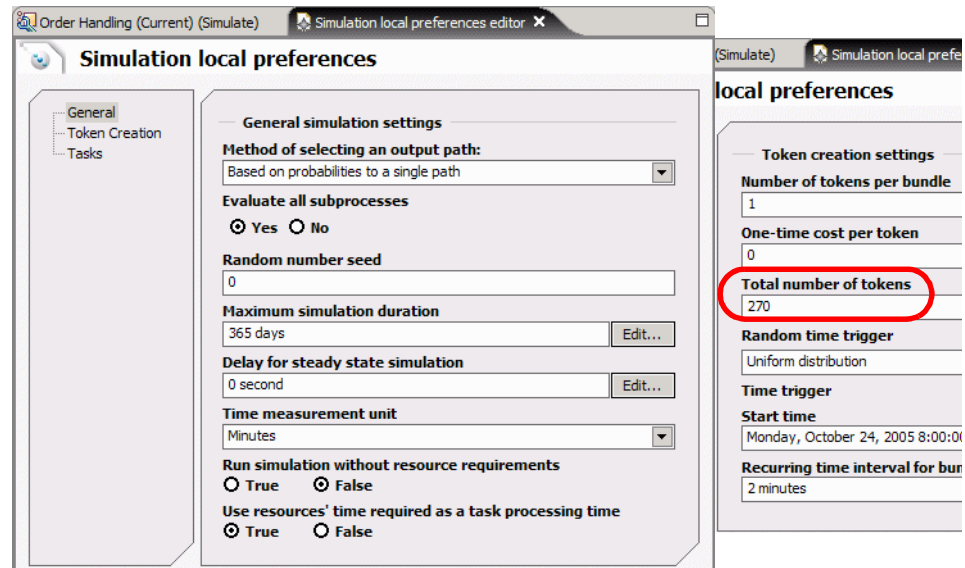– Recurring time interval for bundle creation: `2 minutes`

Figure 5-14   Process simulation default profile

Simulation snapshot process profile

The simulation profile should be open (after creation), but you can also open it from the Project Tree.

> **Important:** To run and display a simulation properly, you have to set your computer time zone before a simulation. Your time zone should be equal to time zone used in the project (except for specific reason). In the current project every element's time zone (timetable, simulation attributes) is defined as (GMT -5) Eastern Time. If you change your computer time zone you have to restart the Websphere Business Modeler.

▶ Populate the *General* tab (Figure 5-15):

– Starting date (GMT-5), and ending date (GMT-5)
– Evaluate all subprocesses: *Yes*
– Time measurement unit: *Minutes*
– Maximum simulation duration: `365 days`
– Random number seed: `1`
– Delay of steady state simulation: `0 second`

– Method of selecting an output path: *Based on probabilities*
– Resources' time required: *Yes*
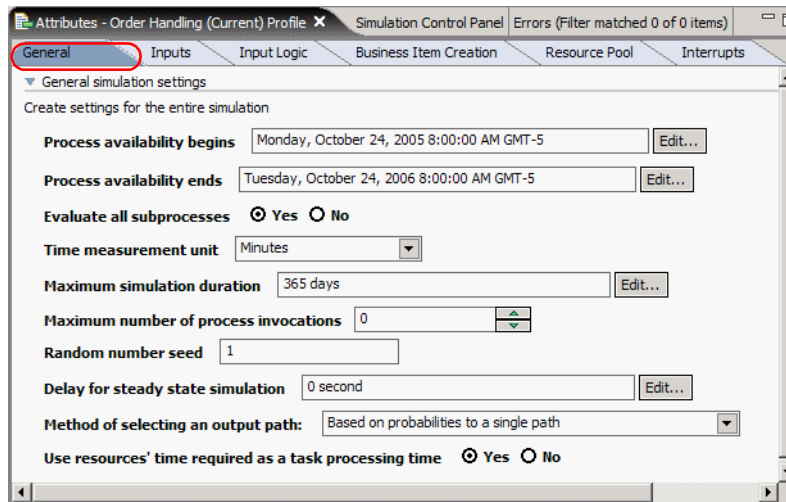
Figure 5-15　Simulation profile: General

► Populate the *Inputs* tab with the number of tokens, the start time, and interval (Figure 5-16):
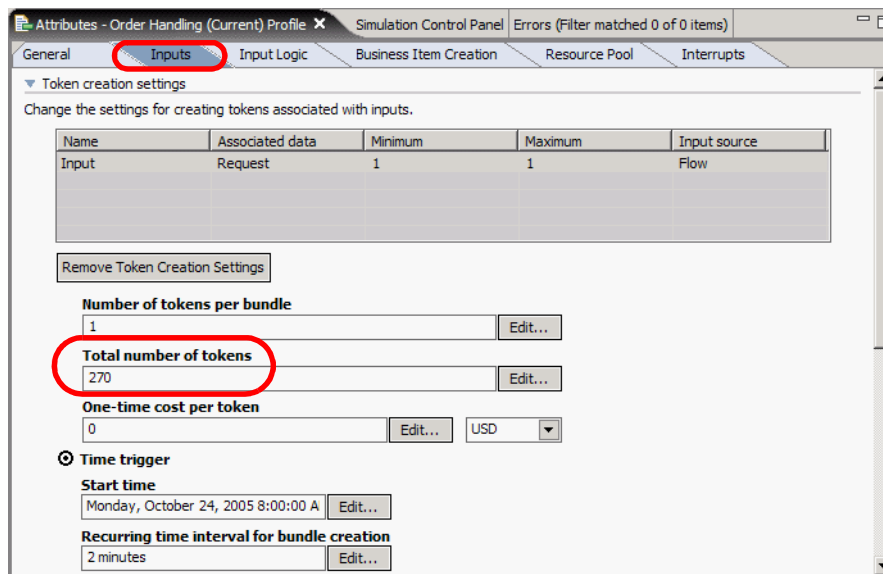
Figure 5-16　Simulation profile: Inputs

- Total number of tokens: **270**
- Select the *Time trigger*
- Start time: same as the process starting date and time zone
- Recurring time interval: **2 minutes**

► Finally, populate the number of human resource available in the *Resource pool* tab (Figure 5-17):

For each role resource (Customer Service Representative, Order Manager, Shipper), deselect *Unlimited* and enter 1 in the quantity box.
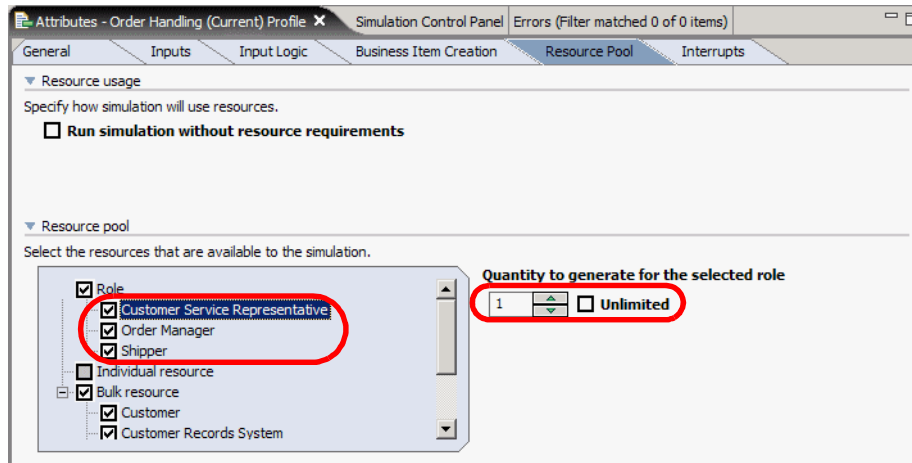
Figure 5-17   Simulation profile: Resource pool

Verify the profile specification

You can verify the profile specification by selecting the Order Handling (Current) Profile and *Profile Analysis → Profile Specification.* Select all processes when prompted.

The profile specification (Figure 5-18) lists all the processes with the resources that are used and the duration of each resource.

> **Note:** You can also see the simulation attributes in the *Simulation Attributes* tab of the process (behind the *Diagram* tab).

Activity Name	Activity Durat...	Requirement ...	Resource or Role Name	Requirement Dur...	Quantity	Inp...	Di...
⊟ Acceptable Credit Risk?							
						No	30%
						Yes	70%
⊟ Cancel Order and Send N...	1 second						
		Role	Order Manager	2 minutes	1.0		
		Bulk Resource	Order Management System	2 minutes	1.0		
Receive Order							
⊟ Receive Order/Determine...	20 seconds						
		Role	Customer Service Repres...	20 seconds	1.0		
		Bulk Resource	Customer Records System	10 seconds	1.0		
⊟ Receive Order/Enter Acc...	20 seconds						
		Role	Customer Service Repres...	10 seconds	1.0		
		Bulk Resource	Customer Records System	5 seconds	1.0		
⊟ Receive Order/Enter Cust...	6 minutes						
		Role	Customer Service Repres...	5 minutes 45 sec...	1.0		
		Bulk Resource	Customer Records System	15 seconds	1.0		
⊟ Receive Order/Enter Ord...	12 minutes						
		Role	Customer Service Repres...	12 minutes	1.0		
		Bulk Resource	Product Catalog System	8 minutes	1.0		
⊟ Receive Order/Existing A...							
						Yes	50%
						No	50%
⊟ Review Order	20 minutes						
		Role	Order Manager	20 minutes	1.0		
		Bulk Resource	Order Management System	1 minute	1.0		
		Bulk Resource	Customer Records System	1 minute	1.0		
⊟ Ship Order to Customer	16 minutes						
		Role	Shipper	15 minutes	1.0		
		Bulk Resource	Order Management System	1 minute	1.0		

Figure 5-18   Profile analysis: Processes with resource sand duration

Running the simulation

We are now ready to run the simulation. Open the *Simulation Control Panel* view (Figure 5-19) behind the Attributes area, and click the green arrow icon ▶ to start the simulation.
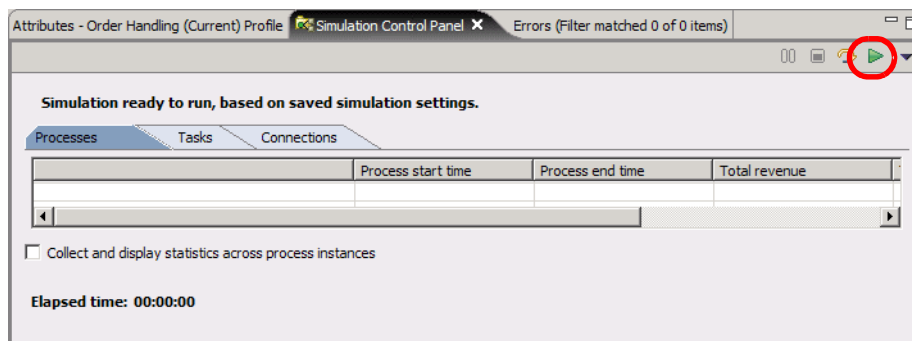

Figure 5-19   Simulation control panel

The system shows the token moving in the flow and the list of process instances (Figure 5-20). At this point, you might pause ⏸ , stop ⏹ , or step through ↻ the simulation.
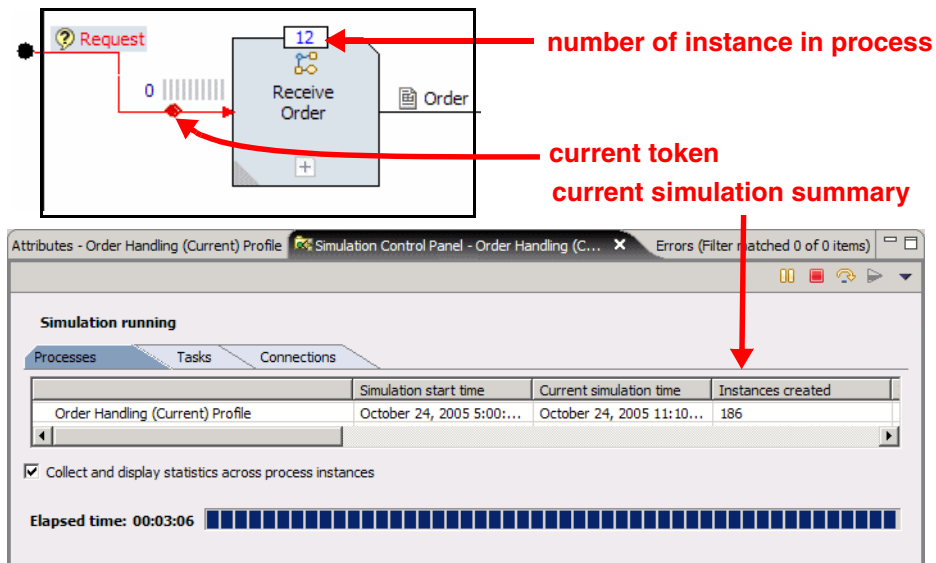

Figure 5-20   Current process simulation running

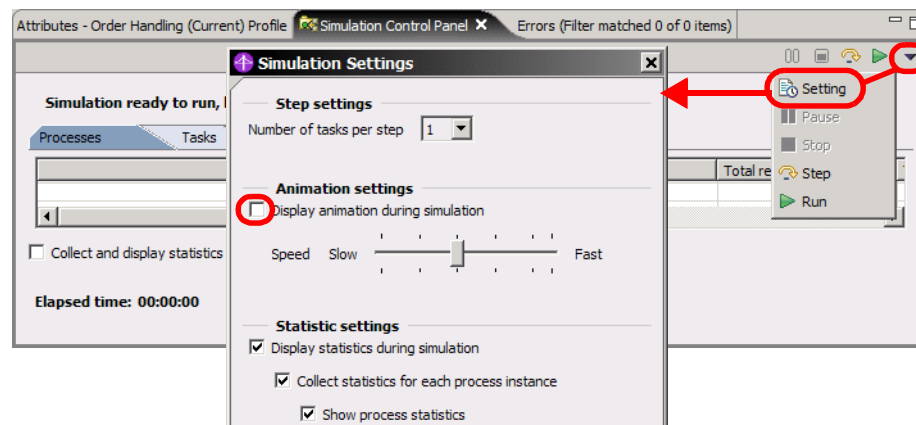**Tip:** You can run the simulation faster without animating the simulation (Figure 5-21).

Figure 5-21   Simulation settings

Analyzing the simulation results (Current)

Once the simulation is complete, you can now use the dynamic analysis function on the simulation result element (Figure 5-22).
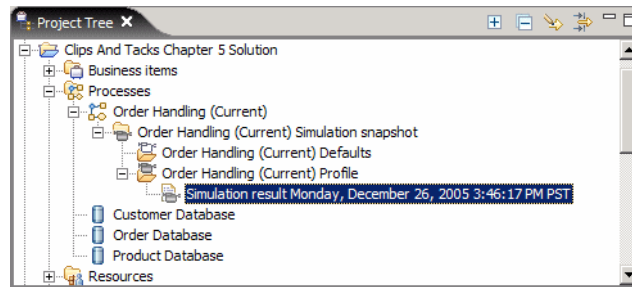
Figure 5-22   Simulation result element

For the current case, the business analyst needs four categories of the process:

► Process duration
► Process cases summary
► Resource usage
► Process cost

Process duration

To get the process duration information, select the simulation result element and *Dynamic Analysis → Process Cases Analysis → Process Duration*:

► When prompted, select *All process instances*.
► When prompted, click *Yes* for the Path Signatures.

This analysis shows process elapsed duration and throughput details for each process case in a simulation.

Process elapsed duration is the duration that a process case takes if started at a specific time and date. Process elapsed duration includes transfer times between activities and the elapsed durations of all activities on the critical path. The critical path is defined as the processing path that has the longest duration of all parallel paths in the process case. Calculations are performed per case by getting the simple average of the process instances duration records in a case.

You can use this analysis when you want to examine process level processing durations and throughputs for each generated process case. This analysis, like other process case analysis, may reveal unexpected results within specific process cases.

For example, you may determine that the average throughput in a particular process case is unacceptably low. As a result of reviewing the information that this analysis presents, you may decide that you need to modify the process model or reset resource levels, or you may determine that you want to investigate further with another type of process case analysis such as process resource analysis.

Alternatively, you can examine the duration results for specific process instances within a process case by running the process instance summary analysis and then the process instance time analysis.

In the summary the process instances analysis shows four cases that reflect four different ways of processing of customers requests (Figure 5-23).

Attributes - Review ...	Simulation Control P...	Errors (Filter match...	Dynamic Analysis ✕	Static Analysis	Technical Attributes ...		

Process Duration | Simulation result Thursday, December 29, 2005 3:15:06 PM PST | Order Handling (Current) Profile | 3:46:24 PM PST

Process Duration | Simulation result Thursday, December 29, 2005 3:15:06 PM PST | Order Handling (Current) Profile | 3:46 PM

Case Name	Distribution	Success Status	Average Elapsed Duration	Average Throughput
Case 1	34.07%	Succeeded	3 days 17 hours 26 minutes 4.945 seconds	0.011181 work items / hour
Case 2	18.15%	Succeeded	10 days 5 minutes 21.938 seconds	0.004165 work items / hour
Case 3	34.07%	Succeeded	8 days 1 hour 51.956 seconds	0.005181 work items / hour
Case 4	13.7%	Succeeded	6 days 5 hours 24 minutes 37.432 seconds	0.006693 work items / hour
Weighted Average			6 days 16 hours 17 minutes 18.5 seconds	0.006239 work items / hour

Figure 5-23   Process duration analysis

Select a case and a blue line in the process diagram shows the path:

► The first case reflects a **shipped** product for a new customer (with new customer entry in the company database).

► The second case reflects a **cancelled** order for a new customer.

► The third case reflects a **shipped** product for an existing customer.

► The fourth case reflects a **cancelled** order for an existing customer.

At this point, we can validate that of 270 requests only 68% (34.07 + 34.07) were shipped to customers.

One of the goals of the improvement will be to increase this percentage.

Another type of analysis can be done with the simulation tool. If you know how long your customers can wait for a product and you want to know how long your company can wait before taking corrective action. You can run the simulation for many durations (1 day, 2 days, 1 week, 2 weeks, 4 weeks, 8 weeks). As a result you can see the curve of order handling duration and the average delay to ship a product to a customer (Figure 5-24).

For instance, if your customer can wait 100 days for a product, we have to take an action (for example, add a new shipper) after 4 weeks of running. Or we can choose to improve your process to get a more linear result.
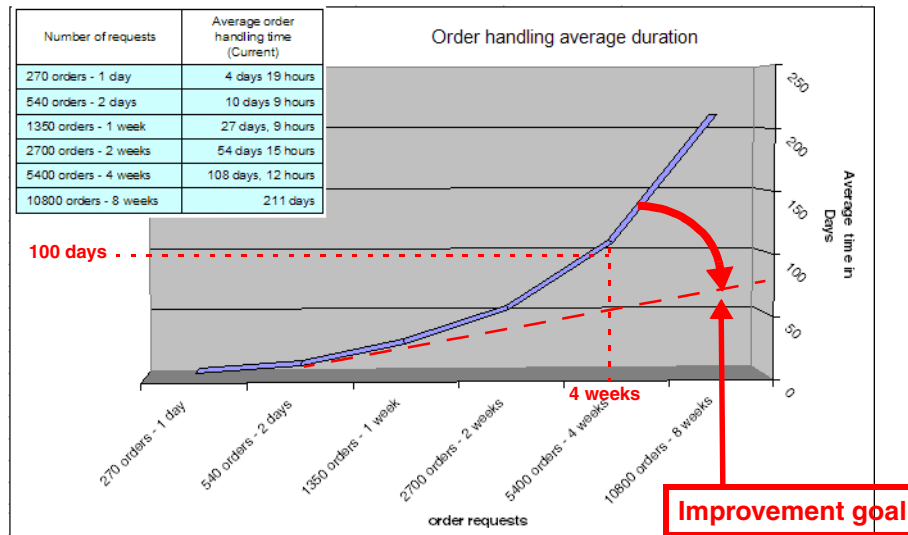

Figure 5-24   Order handling average duration curve

Conclusion of the business analyst

The business analyst detects an abnormal behavior and a bottleneck in these results. He should analyze the worst case to figure out the resource problem. Now we know that we cannot sustain the current process in the long run.

Process cases summary

To see the process cases summary, select simulation result element and *Dynamic Analysis → Process Cases Analysis → Process Cases Summary*.

The process cases summary analysis (Figure 5-25) shows summary details for all the process cases produced during the simulation of a process. A process case is defined as a set of process instances that have the same processing path.

You can use this analysis when you want an overview of the process cases generated in a simulation. This analysis provides high level summary information for each process case, including duration and cost information and an indicator whether the process case was successful or not. This analysis lists the activities completed in each case and quantifies their average total cost and average total elapsed duration.

This analysis—like other process case analysis—may reveal unexpected results within specific process cases. For example, you may determine that the average process elapsed duration in a particular process case is unacceptably high. As a result of reviewing the information that this analysis presents, you may decide that you need to modify a process model or reset resource levels, or you may determine that you want to investigate further with another type of process case analysis or process instance analysis. Alternatively, you may investigate the reasons that cause a particular process case to fail.

Figure 5-25   Process cases summary

If we analyze details of the two worst cases (case 1 and case 4), we can see what the time consuming activities are (Figure 5-26).

Figure 5-26   Process cases 1 and 4 analysis in detail

We find that the high consuming activities are Enter Account Number and Enter Order Information. The Receive Order subprocess that contains the two consuming activities, is the first bottleneck of the current process.

Conclusion of the business analyst

Several customer surveys have indicated that ClipsAndTacks' customers are not satisfied with the ordering process. The telephone ordering procedure is time-consuming, and customers are frustrated at being placed on hold while waiting for the next available representative. Regular customers are frustrated at the amount of time it takes to receive their orders. These delays are caused by the enter order information process, requiring lot of time to the customer service representative. The process cases summary enables us to assign the delays to the enter order information and account number activities. Now we have to look inside those activities to figure out which resource is an issue.

Resource usage

To see the resource usage, select the simulation result element and *Dynamic Analysis → Aggregated Analysis → Resource Usage*.

This analysis shows information on usage of each resource that is allocated in a process simulation.

This analysis helps in resource planning as it enables you to see how each resource is allocated to different activities across the process. In addition to showing how a resource uses its time to accomplish one or more activities, this analysis shows where shortages of resources cause delays in completing activities. You can use this information to determine where additional resources are required.

We take a look on the enter order information activity (Figure 5-27) to identify which resources to analyze. In this activity two resources are used. One resource is a bulk resource, so this resource cannot be the bottleneck. The other resource is the customer service representative. We have to look in the resource usage analysis if this resource has high shortage duration.
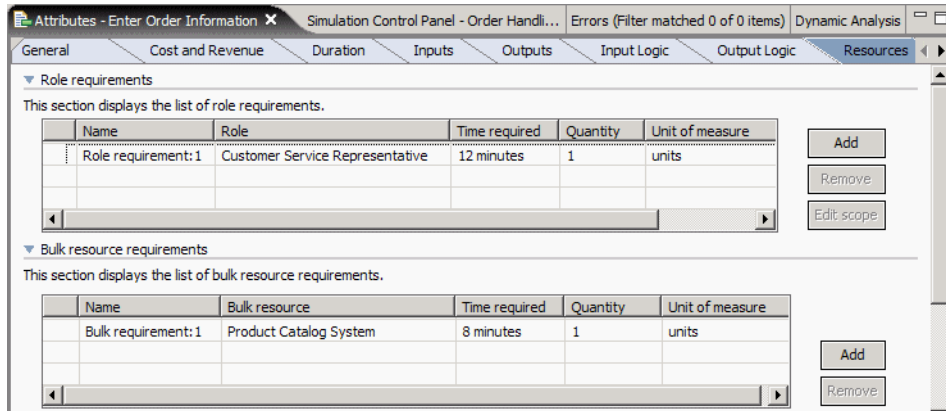
Figure 5-27   Handling Order (Current): Review Order: Enter order information

The resource usage sheet (Figure 5-28) shows the exponential growth of the shortage duration for the customer service representative. This demonstrates that the customer service representative is really the bottleneck of this process.
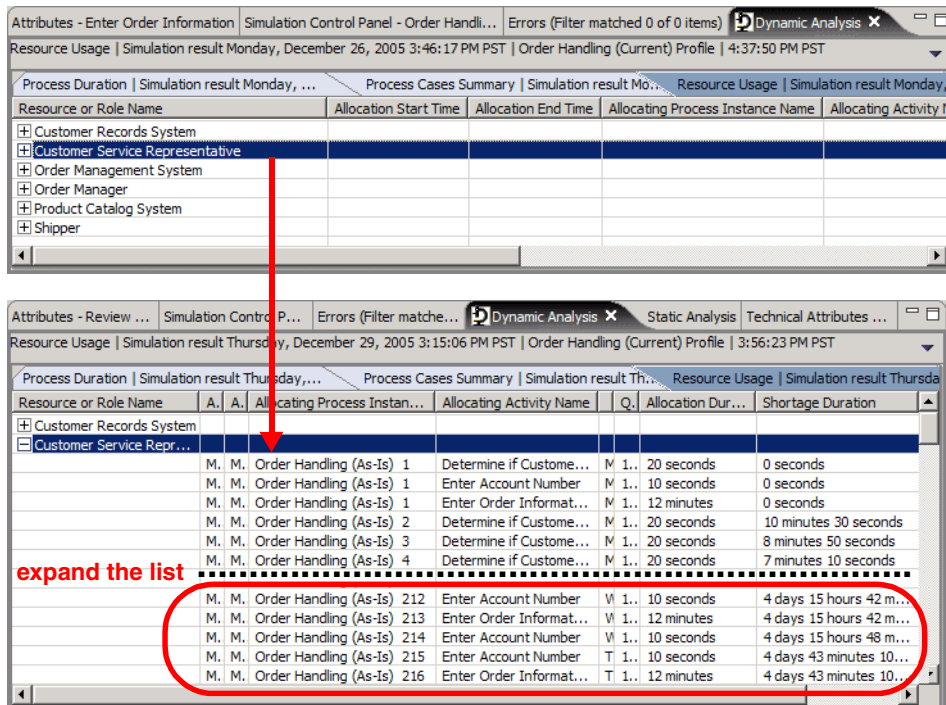

Figure 5-28   Customer service representative shortage duration

Conclusion of the business analyst

This analysis shows the overbooking of the customer service representative and the result in term of shortage durations. To fix this problem, two solutions can be proposed:

► Add a new customer service representative

► Transfer activities from the customer service representative to the customer by introducing a Web application to enter orders.

In response to this bottleneck, the new process will eliminate the need for contact between customers and customer service representatives when an order is placed. Customers will be able to browse the ClipsAndTacks product catalog and enter their own order information using a Web application. New customers will be able to enter their company information and receive a customer account number immediately.

Customers who have an account number will be able to enter it and prompt the Web application to retrieve their information and pre-fill the Web form with their address and preferred shipping information. The new Web application, including the product catalog and order form, will be available 24 hours a day, 7 days a week.

Process cost

To see the process cost, select the simulation result element and *Dynamic Analysis → Process Cases Analysis → Process Cost*.

The process cost analysis shows the average cost and revenue for all process instances in each case in the current simulation result, and the weighted average costs and revenues for all process cases (Figure 5-29).

You can use this analysis when you want to examine process level costs and revenues for each generated process case. This analysis, like other process case analyses, may reveal unexpected results within specific process cases.

For example, you may determine that the average profit in a particular process case is unacceptably low. As a result of reviewing the information that this analysis presents, you may decide that you need to modify a process model. You can also examine the cost results for specific process instances within a process case by running the process instance summary analysis and then the process instance cost analysis.

Figure 5-29   Process cost analysis

During this simulation the average costs were:

▶ $16.26 to ship a product for a new customer
▶ $15.48 to cancel an order for an existing customer
▶ $17.31 to ship a product for an existing customer
▶ $14.43 to cancel an order for a new customer

▶   The average cost for the process is $16.22

Note: Suppose you want to know the number of order cancelled by customers. You assume how long the customers can wait for a product. Then populate the duration limit in the Ship Order to Customer activity, *Duration* tab, *Resource wait time* field. With this parameter, every order not shipped due to a very long processing duration will be considered a failed instance.

Closing reports

To close a report use the arrow pull-down ▼ in the Dynamic Analysis view and select *Close*.

Other reports

There are many more dynamic analysis reports that you can explore. Experiment with the Dynamic Analysis menus to view more reports.

More information

For more information on Analysis, refer to the product documentation under *Analyzing models and simulations*.

Summary

The simulation and analysis of the current process model shows significant bottlenecks that must be reduced to keep the current customers happy and to handle future increase in customer demand.

6

Modeling the Future 1 business process

This chapter describes how the ClipsAndTacks current process is modified to become the Future 1 process.

The key to a successful transformation of the current process is to define step by step all the information acquired by the business analyst.

The first part of this chapter describe the result of the current process analysis.

The second part of this chapter shows how the business analyst modifies the process step-by-step to build the new process using WebSphere Business Modeler. No simulation information is populated at this phase.

The third part of this chapter shows how the IT architect populates the technical information in accordance to the BPEL technical constraints. Typically the busines analyst would consult with the systems architect to discuss implementations issues.

                                                                              83

Documenting the Future 1 process

The Future 1 model enables an organization to capture the potential results of any changes it makes to its process. The Future 1 model does not only provide simulation data and analysis; it will serve as a blueprint for the solution architect and programmers whose responsibility it is to create and implement the new runtime process.

Business revision

The management of ClipsAndTacks wants to improve the company's revenue by improving its order handling process. The assumption is that a shorter wait time for orders and fewer rejected orders will improve customer satisfaction and result in increased new and repeat business for the company.

In the ClipsAndTacks scenario, you will use the current process model as the starting point for the planned revisions to the process model. The key changes in the revised process:

► A Web application enables customers to enter their own account and order information.

► Orders are sent for review or shipping depending on a new business rule, where the threshold value for automatically approved orders is raised to $750.

The business analyst and the management team have determined that the current order handling rule—that is, orders over a certain amount must be approved by an order manager—should remain in place in the new process, but that the rule has to be enhanced. The threshold will be raised from $500 to $750 to reduce the number of orders requiring approval. The business analyst believes that the raised threshold will reduce demand on the order approver, thus speeding up the approvals process for those orders that require it, and reducing the average order completion time.

Summary of revisions:

► In the Receive Order process, customer service representatives are replaced by a single Web Application that provides customer and product information (shorten order process).

► Orders are accepted 24 hours a day, 7 days a week.

► Customer completes the Web order form and submits the order; no customer service representatives are required in this process (reduce labour costs).

► Orders are checked automatically against a business rules engine.

► Regular customer orders over $750 must be approved by the order manager (shorten average order time and increase percentage of approved orders).

Flow revision

The new order handling process contains many of the same activities, but the actors in the process have changed. The process proceeds as shown in Figure 6-1.
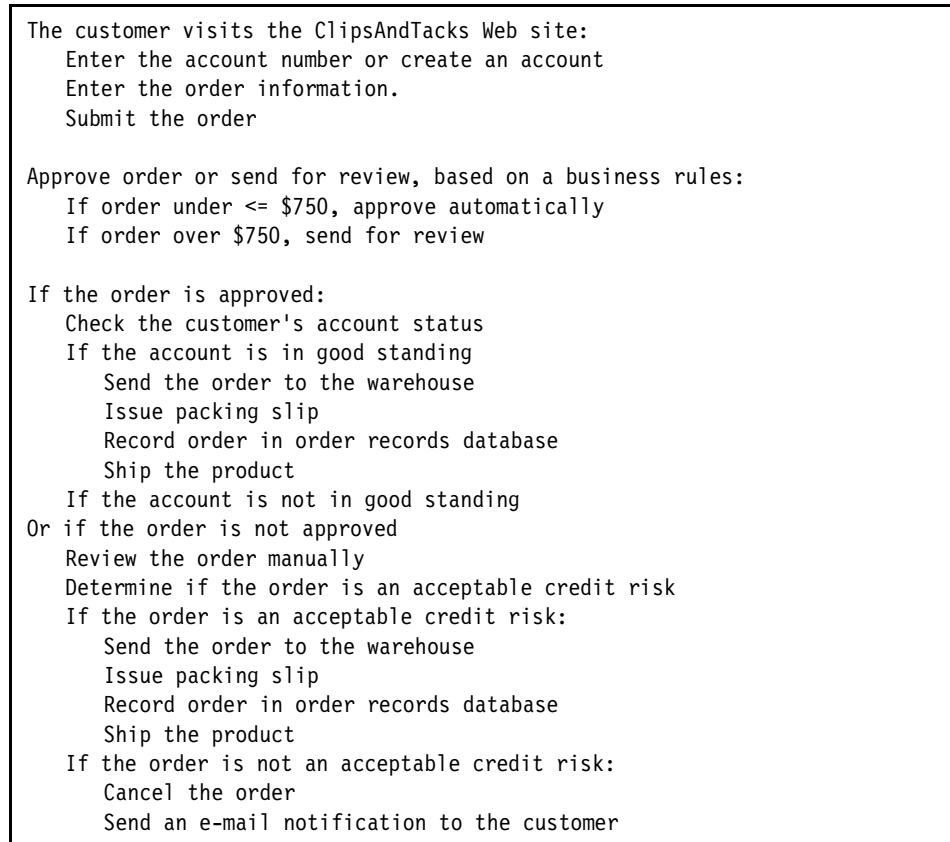
```
The customer visits the ClipsAndTacks Web site:
    Enter the account number or create an account
    Enter the order information.
    Submit the order

Approve order or send for review, based on a business rules:
    If order under <= $750, approve automatically
    If order over $750, send for review

If the order is approved:
    Check the customer's account status
    If the account is in good standing
        Send the order to the warehouse
        Issue packing slip
        Record order in order records database
        Ship the product
    If the account is not in good standing
Or if the order is not approved
    Review the order manually
    Determine if the order is an acceptable credit risk
    If the order is an acceptable credit risk:
        Send the order to the warehouse
        Issue packing slip
        Record order in order records database
        Ship the product
    If the order is not an acceptable credit risk:
        Cancel the order
        Send an e-mail notification to the customer
```

Figure 6-1   Order handling process description

Resource items revision

To model the revised process, the business analyst requires to create the following model elements:

▶ Web application resource—Replaces customer service representatives, customer records system, and product catalog system in the Receive Order process.

▶ Online application timetable—The 24/7 timetable on which the new Web application operates.

Technical outline of the implementation in the Modeler

We make a copy of the current process model and rename it to **Order Handling (Future 1)**. We make the Receive Order process a global process and we update each task that previously required the customer service representative to access the product catalog system, the customer records system, and the order management system. We replace the representatives and various systems with a single Web application that will handle the product catalog, customer records, and the customer order. We also update the model to show that the customer is now a resource used in the process.

Building the Future 1 process

In this section, we explain how to build the new process step-by-step:

► Create new timetable resource
► Create new non-consumable bulk resource
► Create the new process Order Handling (Future 1)
► Create new activities
► Create new decisions
► Create merge nodes
► Create connections
► Populate resources in activities
► Populate expressions in decisions
► Populate business comments
► Validate the process
► Organize the diagram

> **Note:** You can import the revised model from:
>
>   SG247148\sampcode\model\Clips And Tacks Chapter 6 Solution.zip
>
> See "Importing the current process model using the Modeler" on page 47 for instructions on how to import a model.

Creating timetables

A timetable is a schedule of times. In business process modeling, timetables are usually associated with resources or costs. A timetable for a resource indicates availability (such as Monday to Friday). A timetable for the cost of a resource indicates when the cost applies.

You can set up timetables containing recurring time intervals that are relevant to the business. For example, you may want to model costs of resources that vary

depending on the time of day, such as electricity, or costs of resources that vary depending on the time of year, such as seasonal workers.

In this section, we create a timetable to define the online time of the future Web application and a new timetable to define the frequency of customer requests.

Creation of the Online Application timetable

In the Project Tree select *Resources* and *New → Timetable.*

In the dialog box (Figure 6-2), enter the name `Online Application` and click *Finish*.
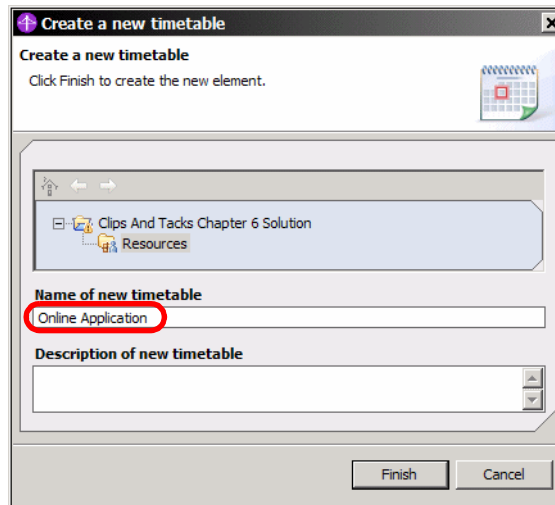
Figure 6-2   Create timetable dialog

Once you created the new timetable, you have to populate all parameters (Figure 6-3):

► Leave *Forever* selected and leave the repetition period as 1 day.

► Click *Select Time.*

► In the DateTime dialog box set the time zone (`GMT-5`), year (`2005`), month (`July`), day (`18`), hour (`12`), minutes (`0`), seconds (`0`) and `A.M`. The date and time specified here should be less or equal to the starting the date of your future simulation. Click *OK*.

> **Note:** The time zone must be equal to the time zone defined in the rest of the project and your computer time zone. The value 12:00:00 A.M stands for midnight.

► Click *Select duration*, enter 24 hours, and click *OK*.

► In the Selected interval details area set the same time as above (July 18, 2005, 12am).

► Save and close the dialog.

Figure 6-3   Online Application timetable

Creation of the Online Request timetable

Create another timetable in the same way and name it Online Request.

Populate the Online Request timetable with the same values as the Online Application timetable (Figure 6-3).

Add an exemption period

Select the *Exemption Periods* tab (Figure 6-4) and click *Add*. Select *Weekend* and click *OK*. The data of the Weekend is filled in the right-hand side.
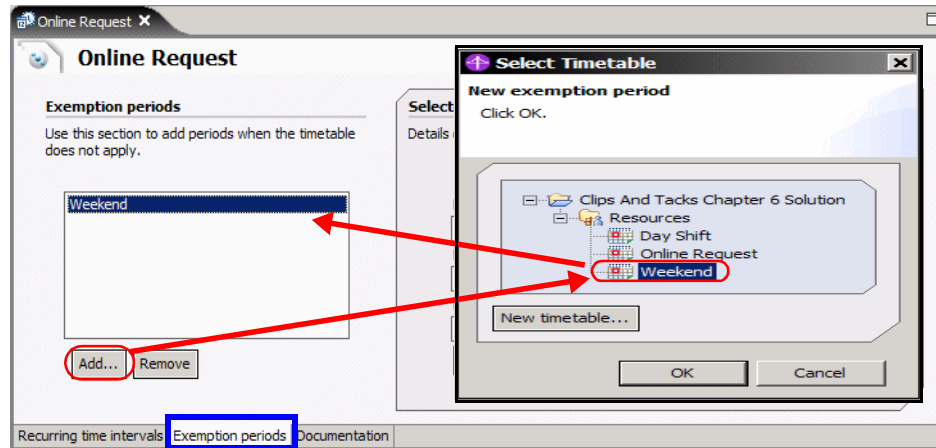
Figure 6-4   Online request timetable, exemption tab

Create a new non-consumable bulk resource

Resources represent the people, equipment, or material used to perform a project or a task. Examples of resources are Computer and Employee.

You can model two types of resources: individual resources and bulk resources:

► **Individual resources** are resources where a specific instance is required, whereas bulk resources are resources where any instance of a type of resource from a pool can be used. Individual resources include people and computers, and bulk resources include power and water.

► **Bulk resources** can represent the material used to perform a project or a task. They can be non-consumable (such as employees, vehicles, or equipment) or consumable (such as fuel or printer paper). Consumable resources are diminished, or perhaps even used up, during the process.

You can define bulk resources as resources that are not uniquely identified, but whether resources need to be identifiable may depend on how they are being used. In a car rental agency, the cars can be viewed as either bulk or individual resources. From an executive's perspective, individual cars are not of interest, but bulk information is. At the rental desk, the individual resource information of each specific car is important. The modeling of a resource therefore depends on the process being modeled and its purpose.

Resources may have specified periods when they are available. To specify availability, you can use an existing timetable or create a new timetable that indicates the periods of availability. If you do not specify the availability of a resource, it is assumed that the resource is always available. You can also add

costs and qualifications to resources. The qualifications are specific roles that this resource must fulfill.

In this section we create a new bulk resource that describe the operational working time and the number of user session for the Web application:

► Select *Resources* and select *New → Resource*.

► In the dialog (Figure 6-5), enter the name `Web Application`, select *Computer Application* for Associated resources definition, select *bulk* as Resource type, and click *Finish*.
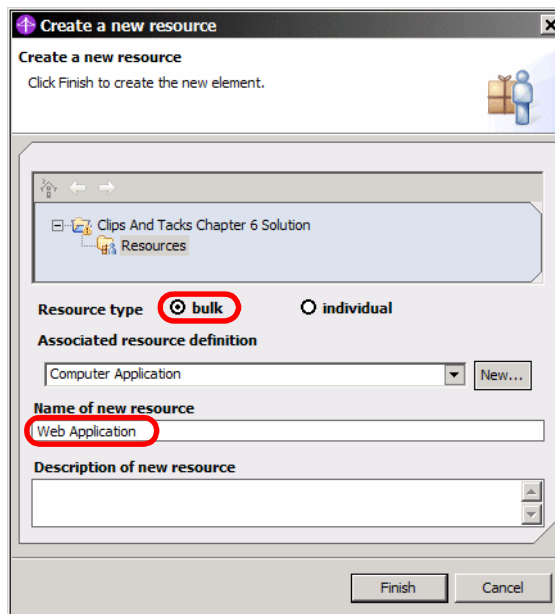
Figure 6-5   Create resource dialog

► Populate the Web Application resource with the cost (Figure 6-6):
  – Select the *Costs* tab
  – Click *Add*.
  – Select *Cost per quantity and time unit* and click *OK*.
  – Enter the cost as `1 USD`.

► Populate the Web Application resource with the availability:
  – Select the *Availability* tab
  – Click *Add*.
  – Select *Online Application* and click *OK*.
  – Set the number of authorized session to `1000`.
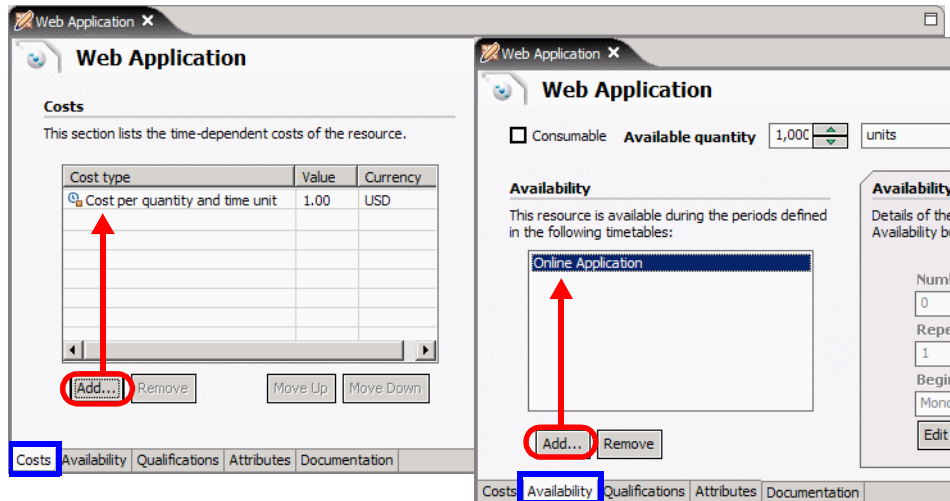
Figure 6-6   Web Application bulk resource cost and availability

Create the new process: Order Handling (Future 1)

In this section we copy the current process to create a new process renamed to Order Handling (Future 1):

► Expand Processes, select the *Order Handling (Current)* process and *Copy*.

► Select *Processes* and *Paste*. Now you have a new process named `Copy of Order Handling (Current)`.

► Rename the process by selecting the copied process and *Rename*. Enter `Order Handling (Future 1)` as new name (Figure 6-7).
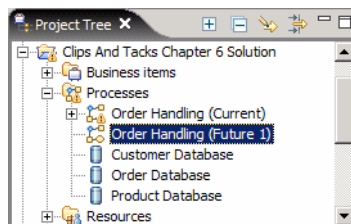
Figure 6-7   Renaming the Order Handling (Future 1) process

Overview of the new process

The finished process is shown in Figure 6-8 (see also Figure 6-22 on page 106). We will implement the process in stages.

Figure 6-8   Order Handling (Future 1) process diagram

Delete an activity

Open the Order Handling (Future 1) process.

We transfer the receive order process activities to the customer using a Web application. Select the Receive Order process and *Delete*.

Create new activities

Tasks are the basic building blocks representing activities in a process model. Each task performs some function. Visually, a task represents the lowest level of work you can portray in a process.

Tasks are atomic actions, meaning they can not be broken down any further, in contrast to processes, which may be decomposed into another flow.

In this section we create two new tasks representing activities in accordance with business analyst requirements:

► **Check Order Handling Policy for Automatic Approval**—In this activity the ClipsAndTacks company asks to check orders automatically against a business rules engine with the rule shown in Figure 6-9.

```
Order Handling Policy
========================================================================
Default: Orders are reviewed by the system for automatic approval
========================================================================
If the total price of an order is less than $750,
then the order can be automatically approved without review.
========================================================================
```

Figure 6-9   Check Order Handling Policy for Automatic Approval

► **Check Customer Account Status**—This activity is a back office system activity where the customer credit rating is verified to set an available credit limit. This activity has an insignificant duration.

Creating an activity

To create an activity follow the steps outlined below.

Create the activity for automatic approval

► Select the *Create a local task* icon and move the mouse to the diagram area, then click to insert the activity.

► Select the new activity, and select the *General* tab in the Attributes view.

- ► Change the name field to `Check Order Handling Policy for Automatic Approval`.

- ► Define the output path business item (Figure 6-10):

  - – Select the *Outputs* tab and click *Add*.
  - – Select the column *Associated Data* and click the ⌨ icon.
  - – Select *Order* and click *OK*.
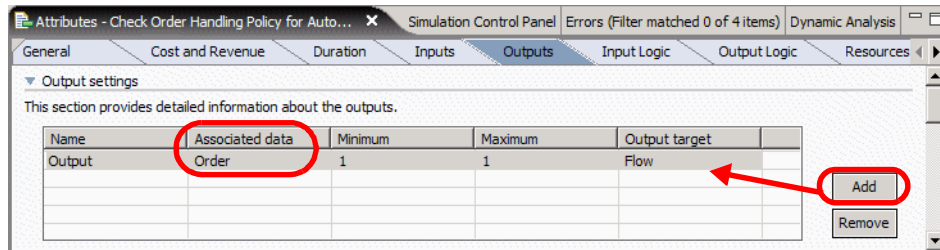
Figure 6-10   Define the output path business item

Create the activity to check the customer account status

Repeat the steps above to create an activity named `Check Customer Account Status` and define the Order business item as output.

Note that you also create new tasks (and other objects) by right-clicking in the diagram area and selecting *New → Local Task*.

Create new decisions

A decision routes inputs to one of several alternative outgoing paths. You can think of a decision as a question that determines the exact set of activities to perform during the execution of a process.

Questions might include: "What type of order?", "How will the order be shipped?", and "How will the customer pay?"

Decisions are flow control constructs rather than activities like tasks or processes. They have no costs or duration, and are used to show alternate paths from a preceding activity.

There are two types of decisions: simple decisions and multiple-choice decisions. In this section we use simple decisions.

A simple decision has one incoming branch with one input, and two outgoing branches with one output each. When the process is running, the process flow takes one outgoing branch if a certain condition is true, and the other branch if

the same condition is false. The decision selects the outcome based on the incoming data.

Creating the decisions

We create two decision nodes:

► Approve Without Review?
► Account in Good Standing?

Create the decision to approve without review

► Select the *Create a simple decision* icon ◇ and move the mouse to the diagram area on the right side of Check Order Handling Policy for Automatic Approval, then click to insert the decision node (Figure 6-11).

► Select the decision node, and select the *General* tab in the Attributes view.

► Change the name field to `Approve Without Review?`
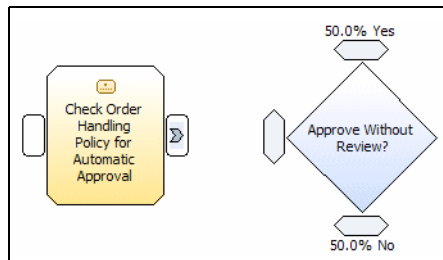
Figure 6-11   Decision Approve Without Review?

Create the decision to verify the account is in good standing

Repeat the steps above and create a decision named `Account in Good Standing?` on the right side of Check Customer Account Status

Create merge nodes

Merges combine multiple processing paths, recombining alternative flows back into a single flow. Merges have multiple incoming branches and one outgoing branch and are used to recombine separate paths in a process flow. A merge is normally used after an exclusive decision. It runs whenever one of its incoming branches is satisfied. As soon as an input is received at a merge, it is immediately sent out as output.

In this section we create two merge nodes necessary to recombine the process flow before the Review Order activity, and to recombine the process flow before the Ship Order to Customer activity.

To create a merge node, select *New → Merge* from the diagram's context menu. Place one merge node in front of the Review Order task, and one merge node in front of Ship Order to Customer. Accept the default names (`Merge` and `Merge:2`).

Create connections

A connection is a link between two elements. Connections can be used to specify the chronological sequence of activities in a process. Each task, subprocess, decision, or other element passes control to the next task or element along a connection. You can also associate business items with connections to pass data from element to element. Each connection can only have one associated business item, but you can use multiple connections if you need to pass multiple business items between two elements.

> **Tip:** If you place the mouse pointer over a connection in the process diagram, the source and the target of the connection are displayed.

In this section we connect the new activities, decisions, and merge nodes.

Connect the process input to the automatic approval

Create the connection from the activity process input node to Check Order Handling for Automatic Approval (Figure 6-12).

▶ Select the *Create connection* icon ⟶ .

▶ Select the process input node.

▶ Move you mouse to the input node of Check Order Handling for Automatic Approval and click.
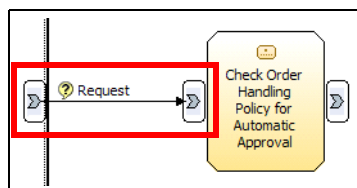
Figure 6-12   Create a connection from the input to Check Order Handling

Create connections between activities, decisions, and merges

> **Tip:** Follow the steps in the exact sequence shown in Figure 6-13. This sequence enables you propagate the output data type (Order).

Follow the steps of the diagram using the connection (arrow) element (Figure 6-13):

1. Select the output node of Check Order Handling for Automatic Approval. Move the mouse to the input node of Approve Without Review? The Yes and No outputs must be grey (that means the data type was set).

2. Select the *No* output of Approve Without Review? Move the mouse to the bottom input of merge node 1.

3. Select the *Yes* output of Approve Without Review? Move the mouse to the input node of Check Customer Account Status.

4. Select the output node of the merge. Move the mouse to the input node of Review Order.
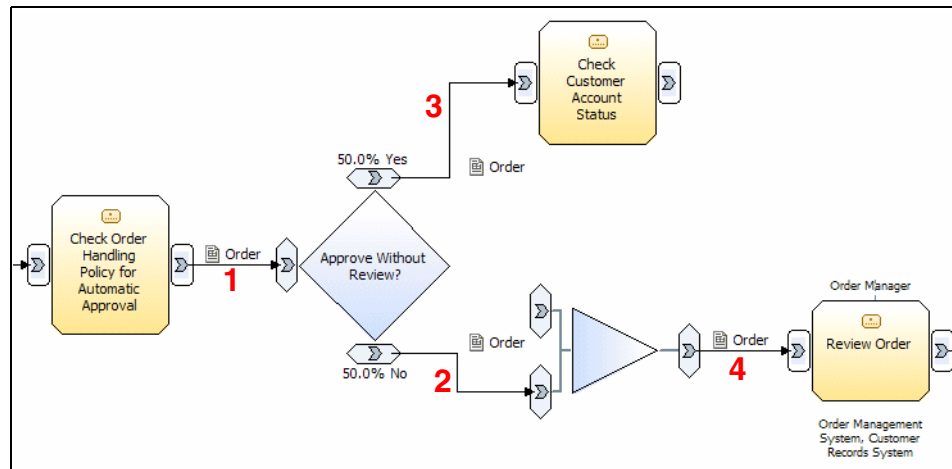
Figure 6-13   Connect and propagate the Order item through the process (step1)

Follow the steps of the diagram (Figure 6-14):

1. Select the output node of Check Customer Account Status. Move the mouse to the input node of Account in Good Standing? The Yes and No outputs must be grey (that means the data type was set)

2. Select the *No* output of Account in Good Standing? Move the mouse to the top input of merge node 1.

3. Select the *Yes* output of Account in Good Standing? Move the mouse to the top input of merge node 2.
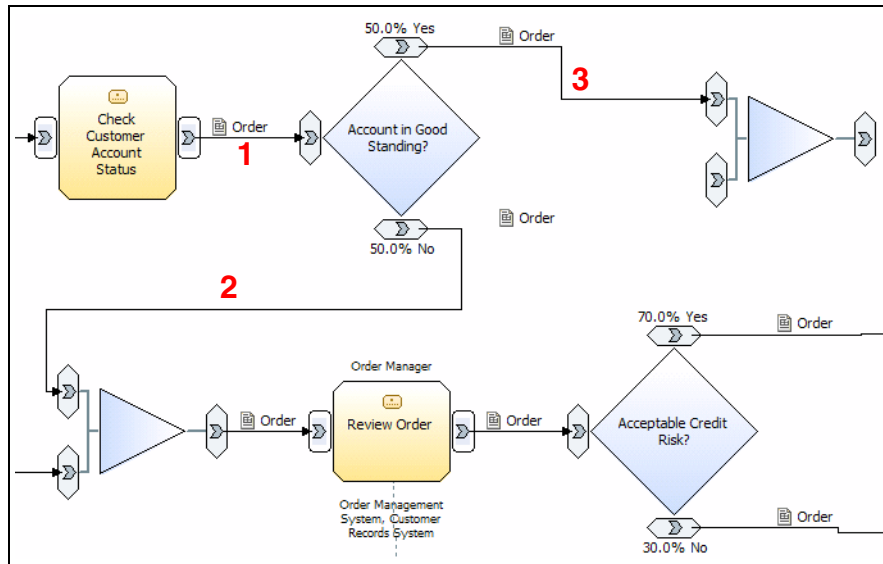
Figure 6-14   Connect and propagate the Order item through the process (step2)

Rewire an existing connection from the decision Acceptable Credit Risk? output to the merge node 2 (Figure 6-15):

1. Select the end of the connection in the input of the activity Ship Order to Customer and move the arrow to the bottom input of the merge node 2.

2. Connect the output of the merge node to Ship Order to Customer.
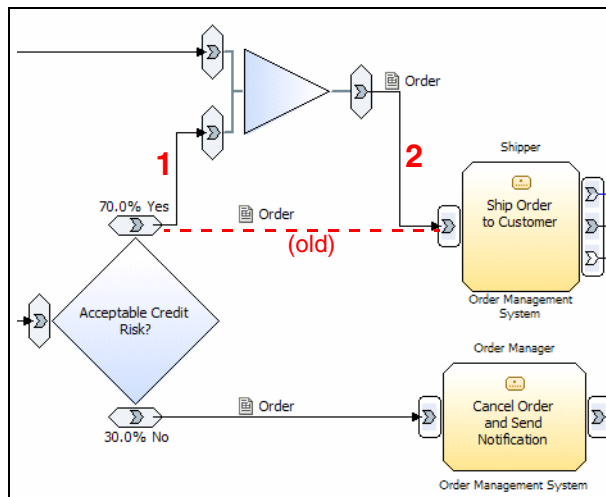
Figure 6-15   Move the connection from Acceptable Credit Risk to the merge node

Change the business item of the process input

Because we deleted the Receive Order subprocess, the business item defined in the input node of the process is wrong. The request item generated by the system is now an order generated by the customer using the Web front-end.

To change the business item of the input process node (Figure 6-16):

1. Select the process input node.
2. In the Attributes view *General* tab, click *Browse*.
3. Select the business item Order and click *OK*.
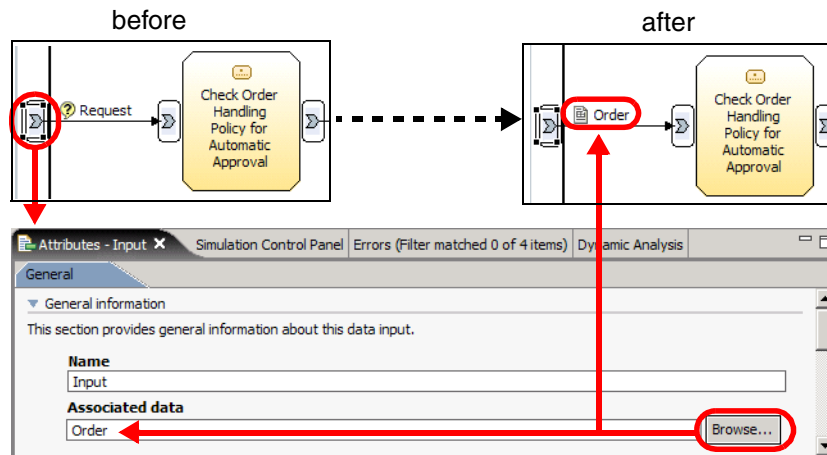
Figure 6-16   Change the business item type of the process input node

Populate resources for the activities

When you model a task, you can specify the individual or bulk resources that are required to complete the task. Resources represent values or instances of resource definitions.

To populate the bulk resources in the Check Order Handling Policy for Automatic Approval activity (Figure 6-17):

1. Select the activity Check Order Handling Policy for Automatic Approval.

2. Select the *Resource* tab, navigate to the Bulk resource requirements area, and click *Add*.

3. Click on the [icon] icon and select *Order Management System*.

4. Click *OK*.
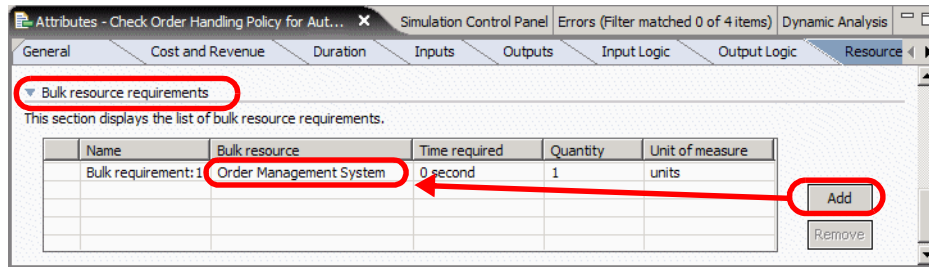
Figure 6-17   Populate resources in Check Order Handling Policy...

Follow the same steps to populate the bulk resources in the Check Customer Account Status activity, but select *Customer Records System*.

Populate expression in decisions

We have to define expression for every decision node. This element describes the behavior in a runtime mode or a simulation mode instead of probabilities. In our case, this expression is only used in the runtime mode (not in the simulation mode).

Populate the expression for Approve Without Review?

The expression is based on the business analyst description, approve order or send for review, based on the business rule:

▶ If the order is under or equal $750, approve automatically.
▶ If the order is over $750, send for review.

In our case the business rules will be implemented in the Check Order Handling Policy for Automatic Approval activity. So in the decision following the activity you have just to check the result of the business rules.

▶ Select the decision Approve Without Review?
▶ Select the *Output Branches* tab and select the *Yes* condition (Figure 6-18).
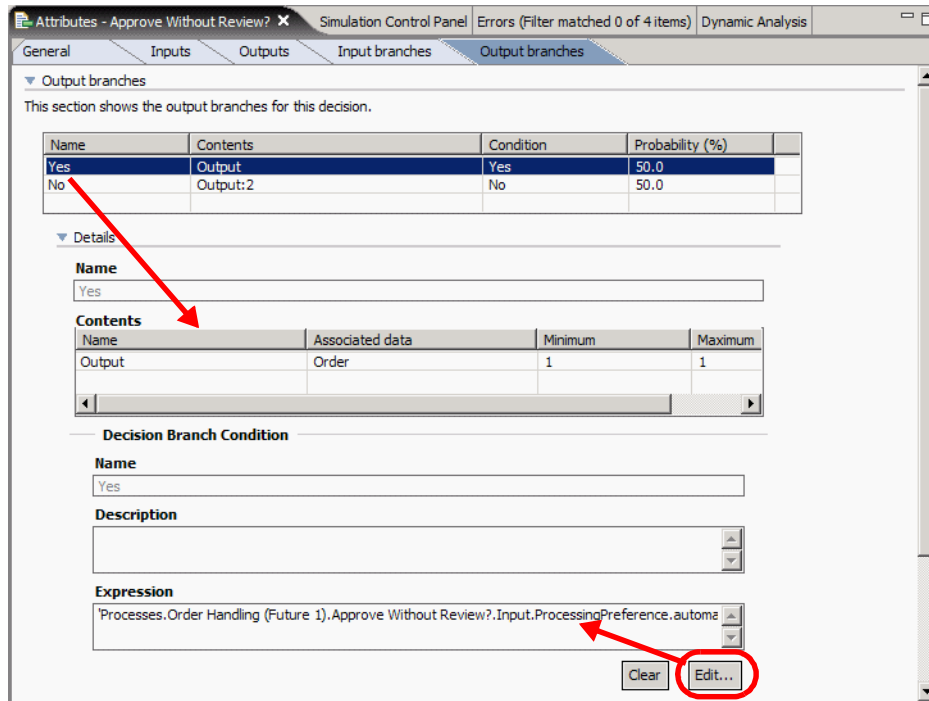
Figure 6-18   Approve without Review decision (Output Branches tab)

► Navigate to the Decision Branch Condition area and click *Edit*.

Define the expression in the Expression Builder by following these steps (Figure 6-19):

► Select the first term type as *Modeling artifact*.

► Navigate to the element *Processes → Order Handling (Future 1) → Approve Without Review? → Input → Processing Preference → automaticApproval*.

► Select the operator *is equal to*.

► Select the second term type as *boolean*.

► Select the term *true*.

► Click *Apply*.

► Click *OK*.

The final expression is:

```
'Processes.Order Handling (Future 1).Approve Without Review?
    .Input.ProcessingPreference.automaticApproval' is equal to true
```
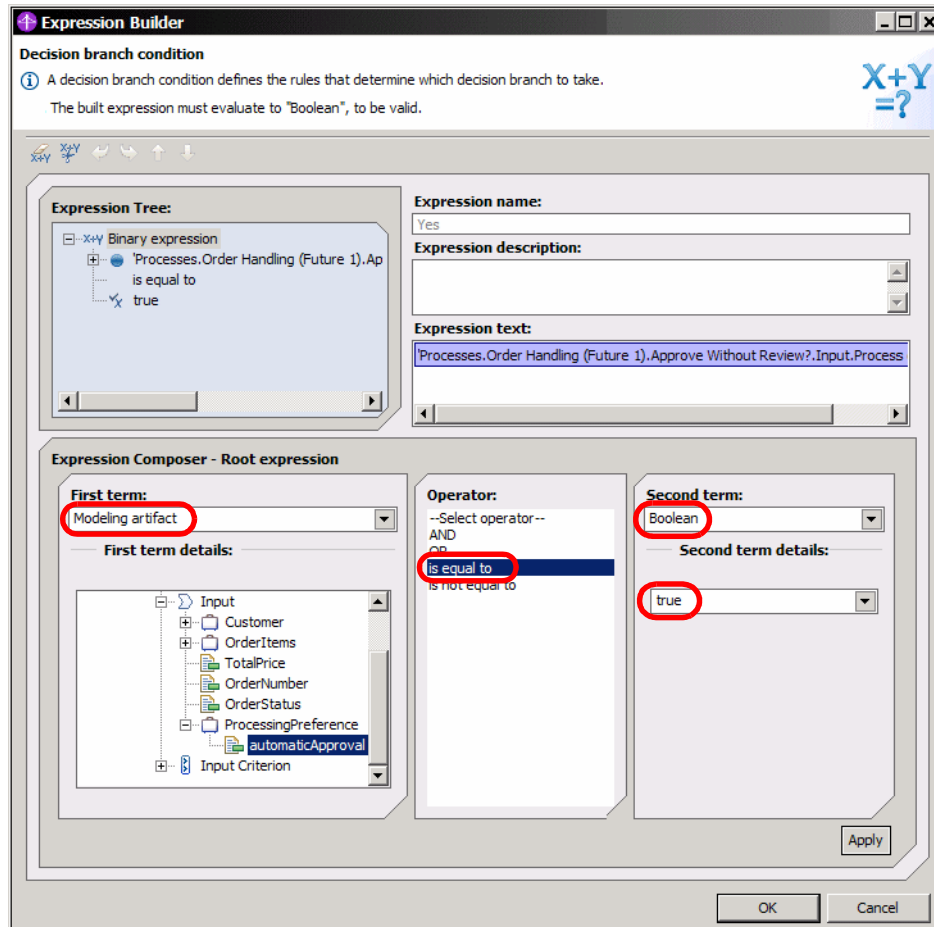
Figure 6-19   Expression Builder: Approve Without Review?

Note: You specified an expression for one of the branches in a simple decision, and WebSphere Business Modeler generates the expression for the other branch for you.

Populate the expression for Account in Good Standing?

Follow the same steps as above, but select these terms and an operator:

► Select the Account in Good Standing decision, open the Output branches tab, select the *Yes* output branch and drop down to the *Edit* button.

► Select *Modeling artifact* and navigate to the element *Processes → Order Handling (Future 1) → Account in Good Standing? → Input → TotalPrice*.

> ► Operator: *is less than or equal t*o

> ► Select *Modeling artifact* and navigate to the element *Processes → Order Handling (Future 1) → Account in Good Standing? → Input → Customer → AvailableCredit.*

The final expression is:

```
'Processes.Order Handling (Future 1).Account in Good Standing?
     .Input.TotalPrice' is less than or equal to '
   Processes.Order Handling (Future 1).Account in Good Standing?
     .Input.Customer.AvailableCredit'
```

Populate the expression for Acceptable Credit Risk?

Follow the same steps as above, but select these terms and an operator:

> ► Select the Acceptable Credit Risk decision, open the Output branches tab, select the *Yes* output branch and drop down to the *Edit* button.

> ► Navigate to the element *Processes → Order Handling (Future 1) → Acceptable Credit Risk? → Input → OrderStatus*.

> ► Operator: *is equal t*o

> ► Select *text* for the second term and enter APPROVED as value.

The final expression is:

```
'Processes.Order Handling (Future 1).Acceptable Credit Risk?
   .Input:2.OrderStatus' is equal to "APPROVED"
```

Populate business comments in the process

In the process we defined activities with resources, decisions with expressions, connections with business items. For now the activity behavior is not defined. To communicate the business requirements to the IT community, we have to comment activities where you require a specific behavior.

We provide a comment for the Order Handling Policy for Automatic Approval activity to communicate the business rule (Figure 6-20):

> ► Select the *Create annotation* icon ![icon] and move the mouse to the diagram area, and click to insert an annotation.

> ► Click on the *Create annotation connection* icon ![icon] and move the mouse to the annotation, click on the annotation, and move the mouse to the Order Handling Policy for Automatic Approval activity and click.

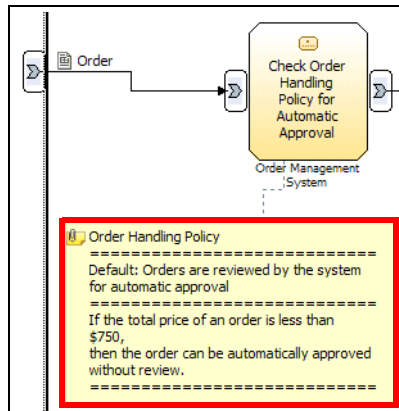> ► Enter the annotation text in the diagram or in the Attributes view.

Figure 6-20   Annotation for Check Order Handling Policy for Automatic Approval

▶ Also add annotations to the activities Ship Order to Customer

        Update order in Order Database to SHIPPED

and Cancel Order and Send Notification

        Update order in Order Database to DECLINED

Delete the Order Database that is connected to the Ship Order to Customer activity. We are using the annotation to convey the message to the implementer of the application.

Validate the process

Now the process is complete. To validate the process flow, right-click in the diagram area and select *Static Analysis → Paths Unable To Be Followed*. This summary returns a list of the paths within the process that cannot be followed because of an invalid input criterion on an activity in the path.

Creating a valid process is important for simulation and for accurate communication. If there are paths within a model that will never be followed, you have to be aware of this.

If this analysis reveals that the input or output criterion of an activity is modeled in such a way that its path is unable to be followed, you can make changes to the model to ensure that the deficiency is corrected.

The input criterion can be invalid for one of the following reasons:

▶ No inputs are specified

► One of the inputs of the input criterion does not have an incoming connector (excluding inputs whose input source is a repository or constant value)

If you get an empty summary, your process is fine. If you get an error result as shown in Figure 6-21, navigate in the tree to find an empty element incoming or outgoing, click on the root element to find it in the diagram.
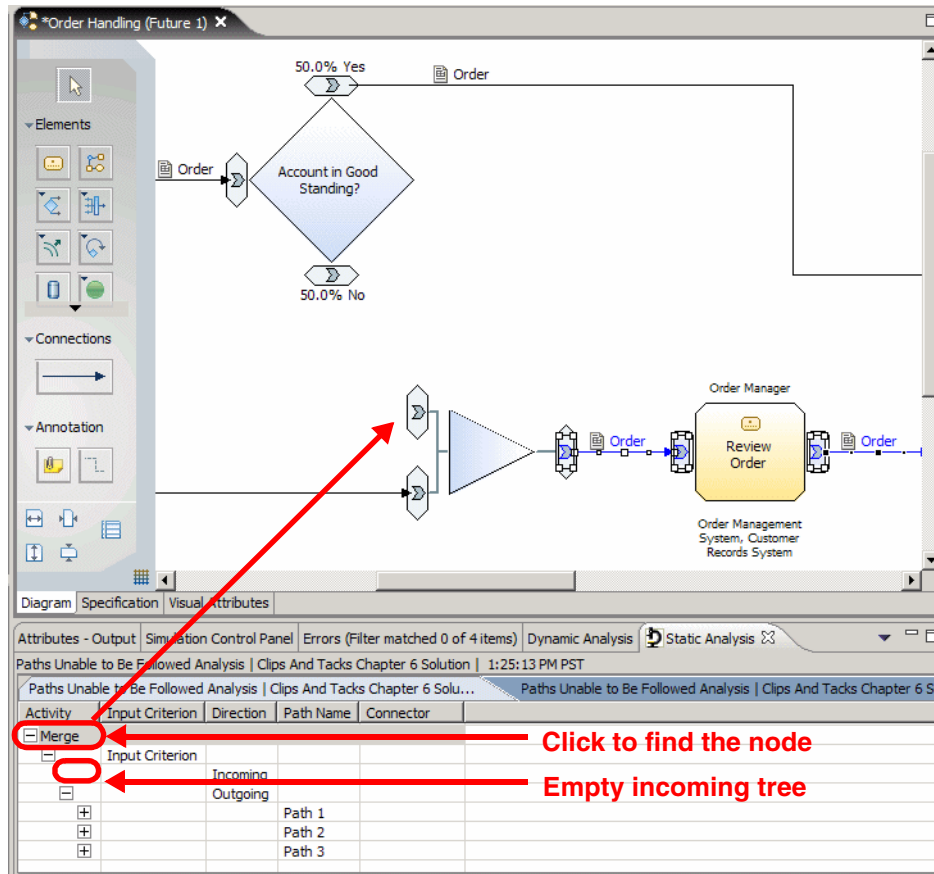
Figure 6-21   Static analysis: Paths unable to be followed result

Organize the diagram

To organize the diagram in a sequential flow, right-click in the diagram area and select *Auto-Layout Left to Right*. This arranges the layout of the diagram so that the direction of flow goes from left to right, and cleans up any overlapping nodes or connections. The result should be similar to Figure 6-8 on page 92 and Figure 6-22.

This diagram shows the future 1
ClipsAndTacks order handling
process.

Order Handling Policy
============================
Default: Orders are reviewed by the system
for automatic approval
============================
If the total price of an order is less than
$750,
then the order can be automatically approved
without review.
============================

Update order in
Order Database
to SHIPPED

Update order in
Order Database to
DECLINED
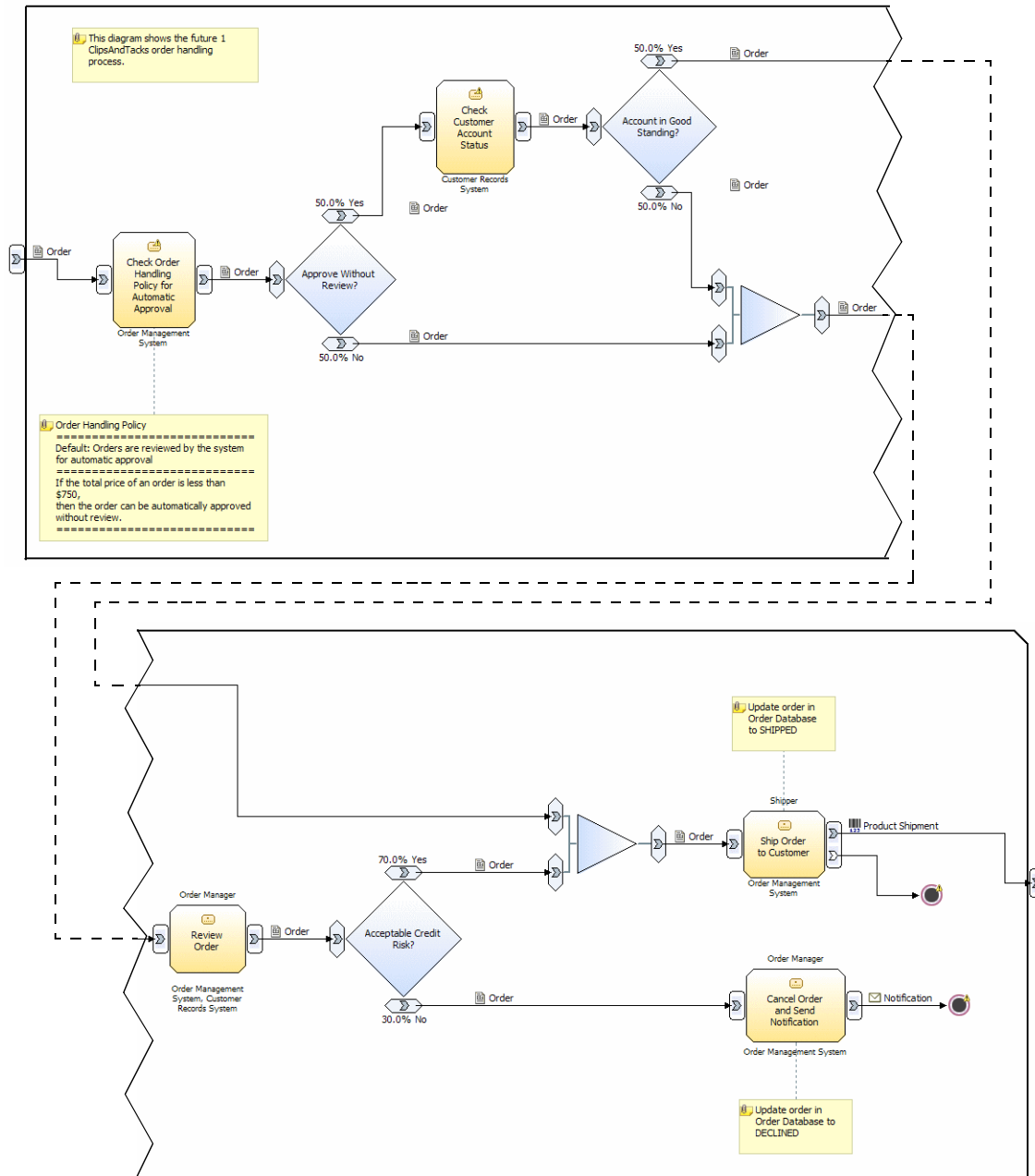
Figure 6-22   Order Handling (Future 1) process diagram

Building an overall process for simulation

To model the customer Web application that provides the input for the Order Handling process, we create a parent process named Overall Order (Future 1) that invokes the Order Handling process.

Create the main process Overall Order (Future 1)

First we create the new process (Figure 6-23):

► In the Project Tree, select *Processes* and *New* (context).

► Enter the name of the process: `Overall Order (Future 1)`.

► Click *Finish* and the new process opens in the diagram editor.
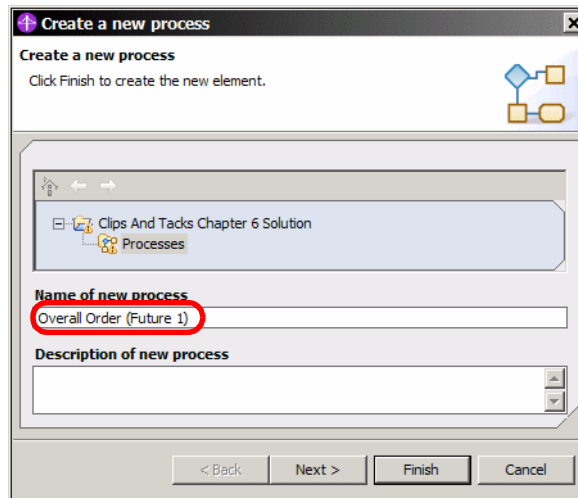
Figure 6-23   Create the new process Overall Order (Future 1)

Create the Receive Order activity

To create the Receive Order activity:

► Select the *Create a local task* icon and move the mouse to the diagram area, then click to insert the activity.

► Select the activity, and in the *General* tab of the Attributes area change the name field to Receive Order.

► Define the output path in the *Outputs* tab by setting the associated data to the Order item.

► Define the input path in the *Inputs* tab by setting the associated data to the Request item.

Create the subordinate process Order Handling (Future 1)

To create the subprocess call to Order Handling (Future 1):

► Right-click on the diagram area and select *New → Global Process.*

► Select the Order Handling (Future 1) process.

Create a Stop node in the process

Click on the ⬤ icon and select the *Stop* icon ◉ and drop it into the diagram.

Delete the Start node and the existing Stop node (very far at bottom right).

Create the connections

Connect the activities in sequential way:

► Input node to the input of the Receive Order activity

► Output of the Receive Order activity to the input node of the Order Handling (Future 1) process

► Output of the Order Handling (Future 1) process to the Stop node

► The finished process is shown in Figure 6-24.
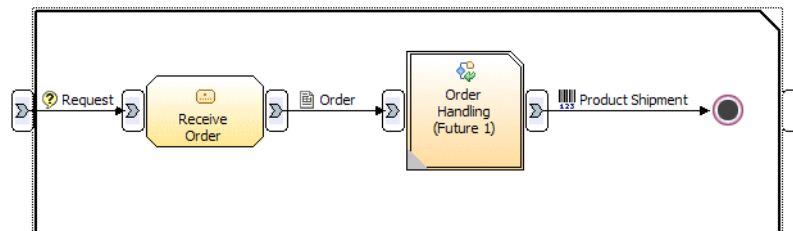
Figure 6-24   Overall Order (Future 1) process

Populate the bulk resource in the Receive Order activity

We define Customer and Web Application as resources (Figure 6-25):

► Select the activity Receive Order.

► In the *Resource* tab, navigate to the Bulk resource requirements area, and click *Add.*

► Click on ⬚, select *Customer*, and click *OK*.

► Click *Add*, click on ⬚, select *Web Application*, and click *OK*.
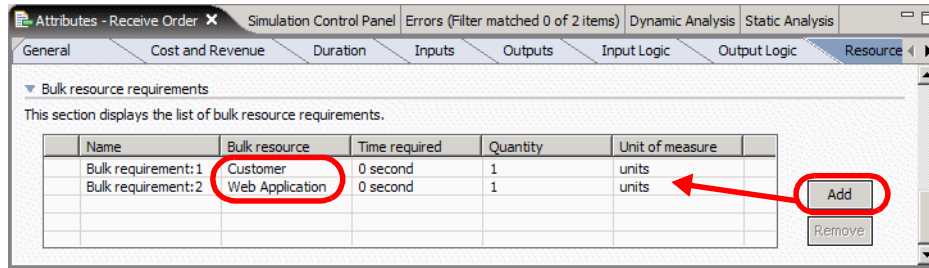
Figure 6-25   Populate resources in receive order activity

Business Process Execution Language

We can export files that can be imported into WebSphere Integration Developer to create an implementation for WebSphere Process Server. We can select an entire project, a data catalog, process catalog, resource catalog, or organization catalog with its contents, or a single process, global task, service, or business item. If you select a process, the process and any business items it references are also exported.

You can export a project as a set of Business Process Execution Language (BPEL) files for exporting into WebSphere Integration Developer. Integration Developer is the tool used to implement business processes for deployment on Process Server. In Integration Developer you create the application code for the model by accessing subsystems, such as databases and EIS, where customer accounts and credit information are typically stored. Finally, you can test the application and deploy the application to Process Server.

Prepare the process for export

Before exporting a project destined for WebSphere Process Server, ensure you are working in the WebSphere Process Server mode and that your model has no errors.

To set the Modeler into WebSphere Process Server mode, select *Modeling* → *Mode* → *WebSphere Process Server*.

The exported process does not necessarily guarantee a complete process definition. For example, if the decision branches in the model have no formal expressions associated with them, no BPEL transition conditions will be generated. In addition, there are certain BPEL export restrictions that prevent all models from being fully transformed.

Errors

As soon as you switch to WebSphere Process Server mode, errors show up in the Project Tree and Errors view (Figure 6-26):
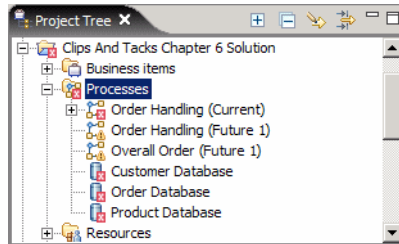
Figure 6-26   Project Tree with errors in Process Server mode

▶ Global repositories (`Customer Database`) are not supported.

▶ The Order Handling (Current) process shows multiple errors:
   – Decision output condition not defined
   – Repository data output must not reference a global repository

Before testing the export mode, you have to change some elements to be compliant with the WebSphere Process Server mode. Open the Order Handling (Future 1) process flow.

Populate technical attributes

To transfer information between the business analyst and the integration designer using WebSphere Integration Developer, the analysts should add some notes in the process diagram that will be transferred through the BPEL export.

Define a comment for the Ship Order to Customer activity (`Update order in Order Database to SHIPPED`) and Cancel Order and Send Notification activity (`Update order in Order Database to DECLINED`). Note that we already did this in "Populate business comments in the process" on page 103.

Business process integration and automation (BPIA) project

The Zurich Research Lab is developing tools (Eclipse plugins) to bridge the gap between business-level and IT-level models:

`http://www.zurich.ibm.com/csc/bit/bpia.html`

The BPIA project develops new tooling for WebSphere Business Modeler to transform, validate, view, synchronize, and search business process models. The project is also working on tooling support for the developer of model transformations that will make it easier to implement, reuse, and maintain model transformations.

Define the type of implementation for all activities

The BPEL export generates an SCA component for each process, global task, local task, and service element that is not being generated as an inline task. The generated component contains component references to create the component wiring information displayed in the Assembly Editor in Integration Developer.

The technical attributes define a set of component properties including the implementation type of the component. The Modeler provides five types of implementation:

► Rule group
► Human task
► Java
► Process
► State machine

If the technical attributes specify an implementation type, the Modeler generates the default implementation based on the implementation type. The default implementation provides a starting point to implement the component in Integration Developer.

For every activity of the process, you should define how this activity will be implemented,

Rule group

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data. An example of a simple rule group is shown in Figure 6-27.
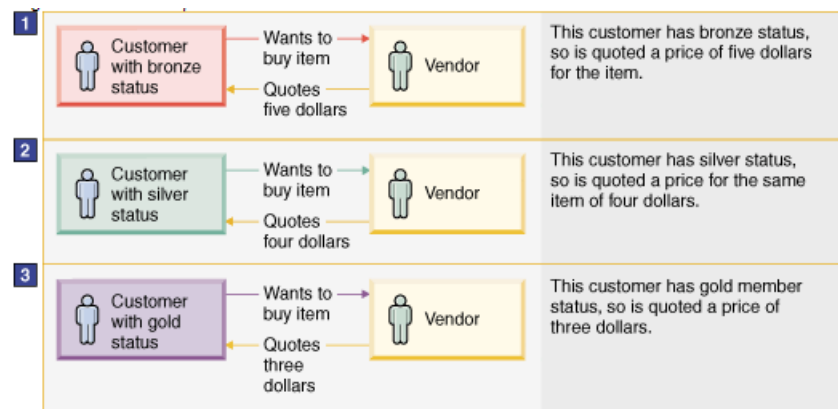
Figure 6-27   Example of a simple business rule group
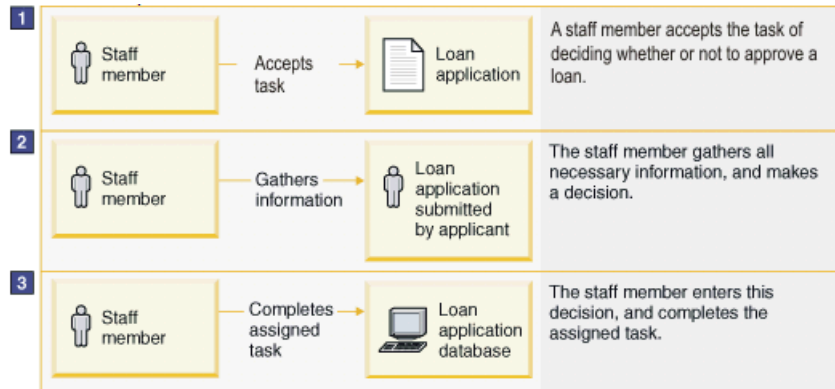
The Integration Developer tools have been designed so that users can easily compose integrative business solutions without programming skills. To this end, you can easily create and develop business rules in an intuitive graphical programming environment (Figure 6-28).
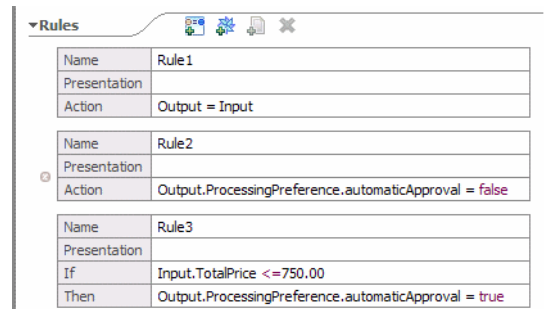
Figure 6-28   Activity rule group implementation

We will implement the task Check Order Handling Policy for Automatic Approval as a rule group:

► Select the Check Order Handling Policy for Automatic Approval activity in the process flow.

► In the Technical Attributes view select the *Implementation* tab (Figure 6-29).

► Select *Rule Group* in the Implementation type.
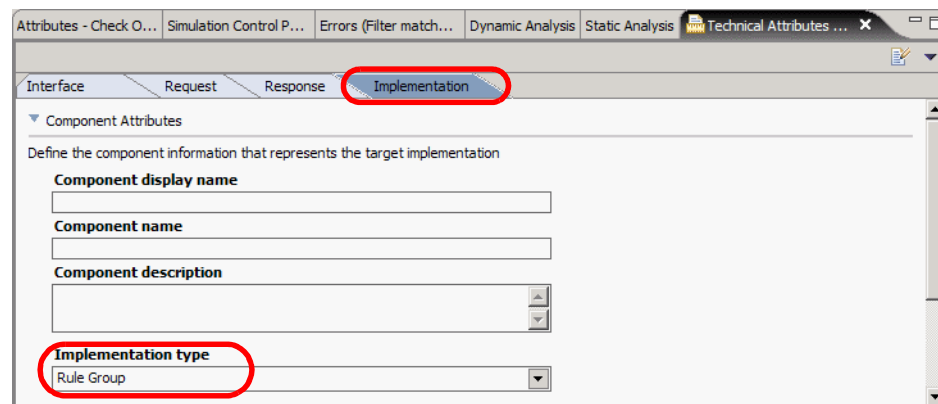
Figure 6-29   Check Order Handling Policy: Implementation type

Human tasks

A human task is, quite simply, a unit of work done by a human. Quite often, this task involves the interaction with other services, and thus becomes a task within a larger business goal.

Examples of human tasks are shown in Figure 6-30.

Figure 6-30   Examples of human tasks

The Integration Developer tools have been designed so that users can easily compose integrative business solutions without programming skills. To this end, you can easily create and develop a human task in an intuitive graphical programming environment called the human task editor (Figure 6-31).
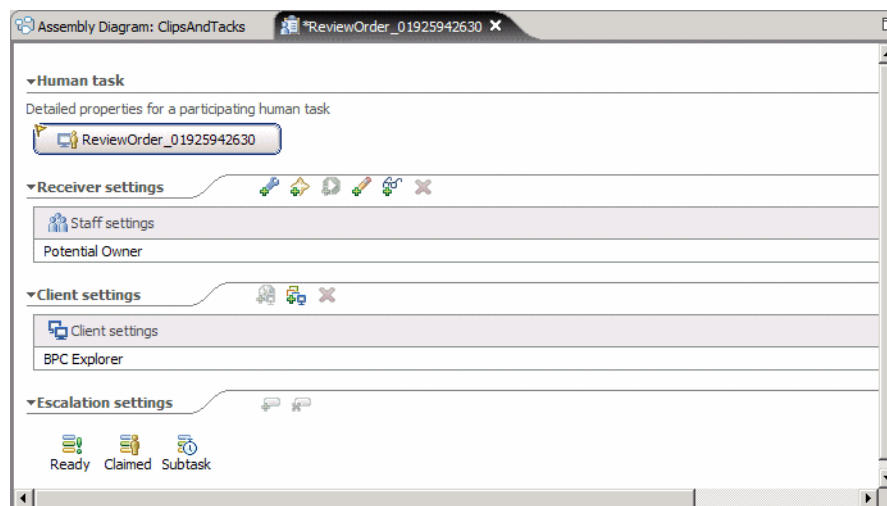
Figure 6-31   Activity human task implementation

The Modeler export generates two types of human tasks: inline task, which is similar to the Staff activity generation of WebSphere Business Integration Server Foundation V5, and a human task component, which is an SCA component that has the implementation type of human task:

► To export as an inline task, the task must have a role requirement or an individual resource requirement. For a role requirement of a task, the type must be specified as the predefined Person or Staff, or a resource definition based on the Person or Staff template.

► To export an activity as a human task component, the implementation type must be set to human task in the technical attributes.

We implement the activities Review Order, Ship Order to Customer, and Cancel Order and Send Notification as human tasks.

Repeat the next steps for each activity:

► Select the activity, for example, Review Order.

► In the Technical Attributes View select the *Implementation* tab.

► Select *Human Task* for the Implementation type (similar to Figure 6-29 on page 112).

Java

The Java implementation type allow you to create the activity using the Java language.

In Integration Developer, we will write Java code to implement the activity:

```
public DataObject InputCriterion(DataObject Input)  {
    System.out.println("Check Customer Account Status Invoked");

    // write Java code
    // minimally copy the input data object as output data object
    DataObject Output = Input;
    return Output;
}
```

We will implement the activity Check Customer Account Status as a Java task:

► Select the activity Check Customer Account Status.

► In the Technical Attributes View select the *Implementation* tab.

► Select *Java* for the Implementation type (similar to Figure 6-29 on page 112).

Process

This implementation type is the default mode of export. It lets you define the activity completely in Integration Developer.

The Integration Developer tools have been designed so that users can easily compose integrative business solutions without programming skills. To this end, you can easily create and develop a process task in an intuitive graphical programming environment called the process editor (Figure 6-32).
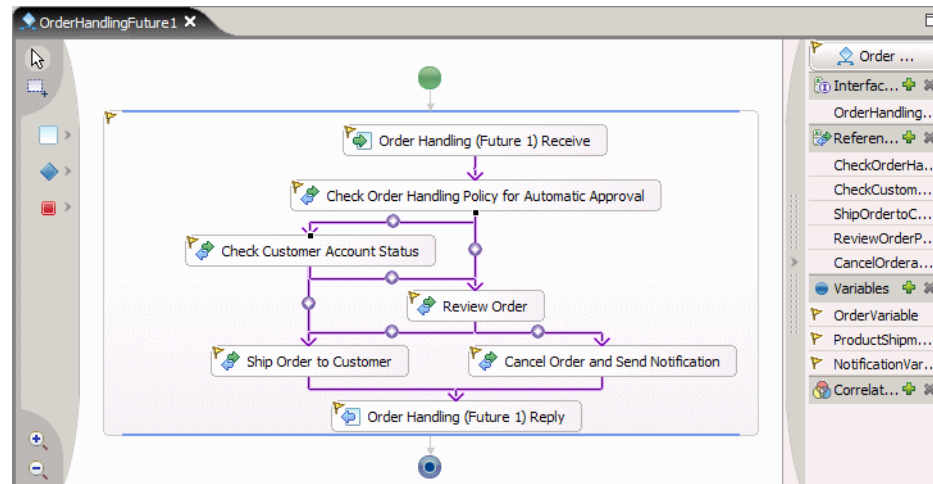

Figure 6-32   Activity or process with a process implementation

This implementation type is the default type for all processes, but we do not use it for any activity in our model.

Note: The business process itself is exported as BPEL even though we have no single activity that is implemented as a process.

State machine

A state machine is an event driven business transaction in which external operations trigger changes that guide the transaction from one discrete mode to another. Each mode is an individual state, and this mode determines what activities and operations can occur.

The Integration Developer tools have been designed so that users can easily compose integrative business solutions without programming skills. To this end, you can easily create and develop business state machines in an intuitive graphical programming environment called the business state machine editor (Figure 6-33).
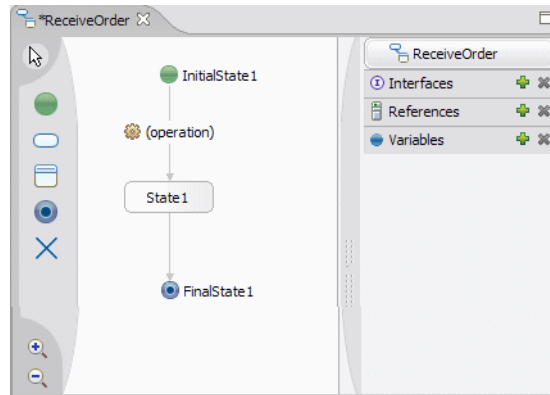
Figure 6-33   Activity state machine implementation

Figure 6-34 shows an example of a state machine.
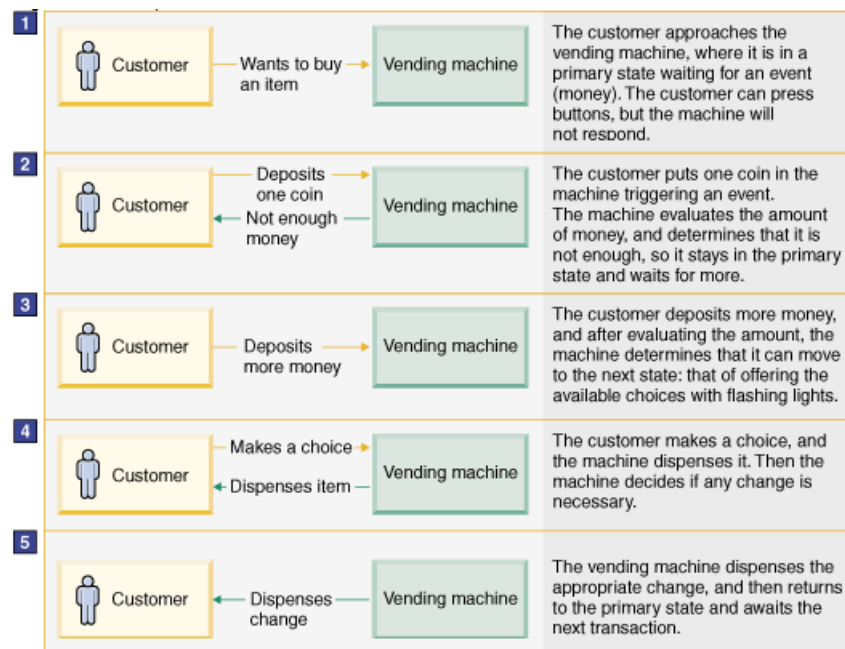
Figure 6-34   Example of a simple state machine

In our example, we will not use an implementation using a state machine.

Define the operation type for each activity

By default the Modeler assigns an operation type of Request/Response. This is suitable for all the activities in our model.

If any activity is a one way operation, you would set the operation type to *One way operation* in the Technical Attributes view (Figure 6-35).
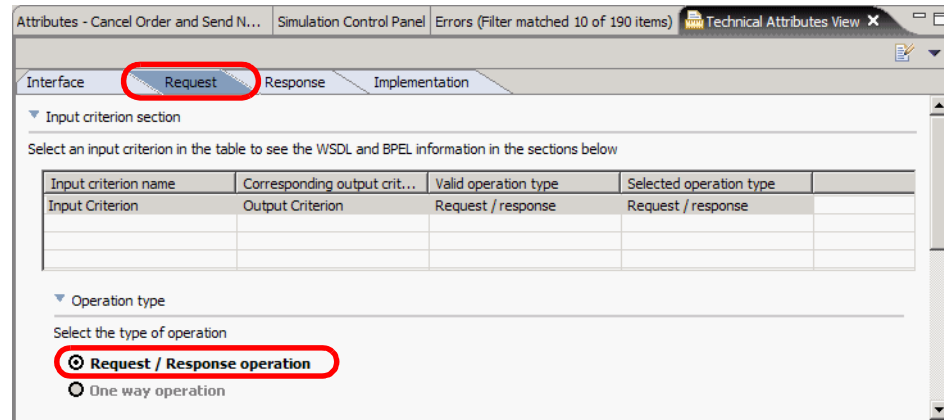
Figure 6-35   Setting the operation type

Export the BPEL for WebSphere Integration Developer

After all export parameters are set, we can export the process into BPEL format, to be imported into Integration Developer.

> **Important:** In Chapter 8, "Creating the business measures model" on page 145 we enhance the model with business measures to be used by WebSphere Business Monitor to measure and validate the requirements of the business process.
>
> If you want to work **without Business Monitor** you can export the current state of the model as BPEL and implement the application using Integration Developer as described in Chapter 9, "Developing the application using WebSphere Integration Developer" on page 187.
>
> If you want to work **with Business Monitor** you should first go through Chapter 8, "Creating the business measures model" on page 145 before exporting the BPEL.

Exporting the model for Integration Developer

To export the model as BPEL perform these steps:

► Select the ClipsAndTacks project and select *Export* (context).

► Then, select the type of export as *WebSphere Process Server* export (Figure 6-36).
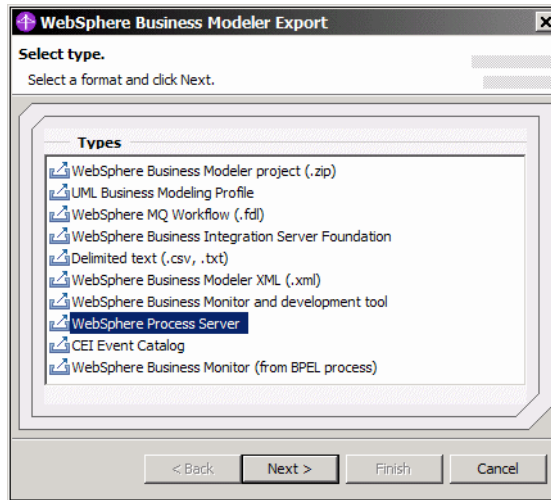
Figure 6-36   WebSphere Process Server export

Now, we have to define how we want to export:

► If you specified a module or library project, the export process creates a folder that contains all of the project's file.

► If you further specify a project interchange name, the export packages the folder into a ZIP file. The result will be a ZIP file in interchange format that can be imported into Integration Developer as is.

► If you do not specify a module or library, the export process creates a set of folders with BPEL and WSDL files for the exported processes and XSD files for the business items used within the processes. The files are placed into sub folders based on the catalog hierarchy of the process and business items.

Restriction: If you have processes with errors (in WebSphere Process Server modeling mode) you cannot export them to Integration Developer. In this case you have to export selected elements (processes) that are free of errors.

The export of the ClipsAndTacks example is shown in Figure 6-37.
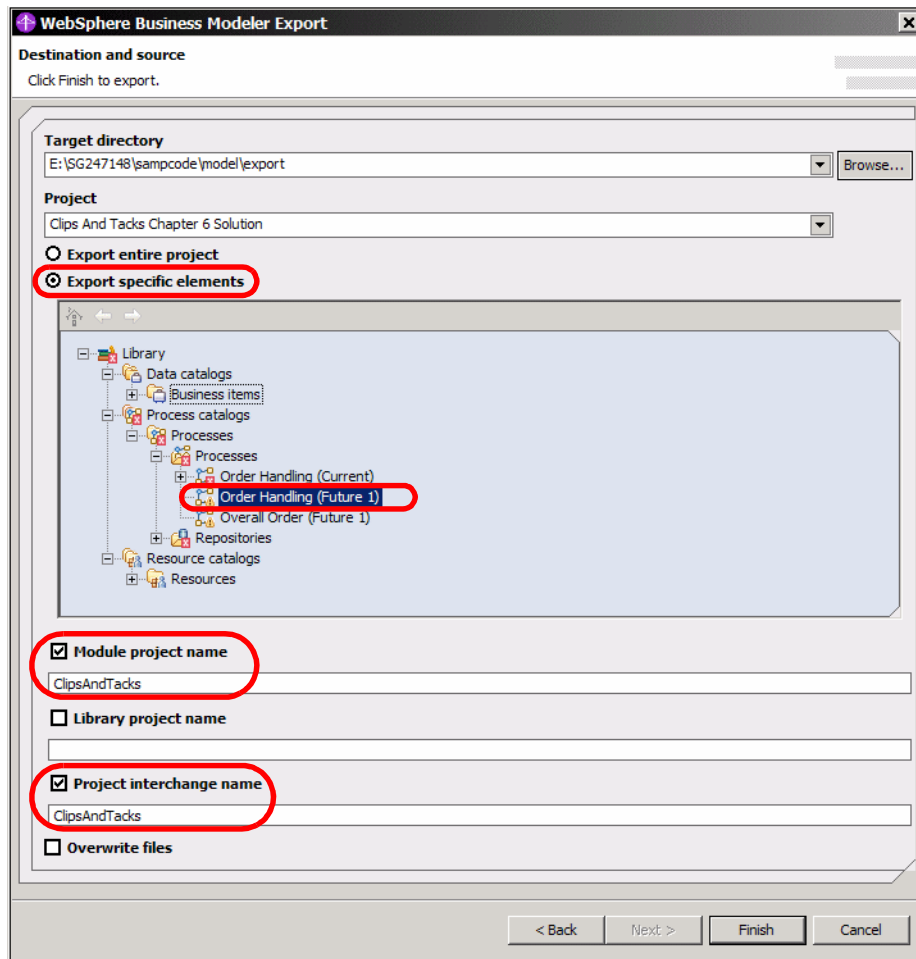
Figure 6-37   WebSphere Process Server export dialog

► Enter the target directory, for example:

    SG247148\sampcode\model\export

► The project is preselected.

► Select *Module project name* and enter the names as `ClipsAndTacks`.

This name, with a suffix of `App`, will become the name of the enterprise application in Integration Developer and Process Server.

► Select *Project interchange name* and enter the name as `ClipsAndTacks`. This name does not have to match the project name.

► Select *Export specific elements* and navigate to the process Order Handling (Future 1). Select the process.

You do not have to select the business items; they are exported automatically with the selected process.

► Click *OK*.

Exported file and content

The exported project interchange file (ClipsAndTack.zip) contains all the elements required for Integration Developer. Inside the zip file, you find BPEL and WSDL files organized by process folders (Figure 6-38).

```
ClipsAndTacks
    .project
    sca.module
    businessitems
        Businessitems.xsd
    processes
        orderhandlingfuture1
            CancelOrderandSendNotification1177120723Interface.wsdl
            CancelOrderandSendNotification_01780893802.component
            CancelOrderandSendNotification_01780893802.tel
            CancelOrderandSendNotification_01780893802_tel.mon
            CheckCustomerAccountStatus0319499984Interface.wsdl
            CheckCustomerAccountStatus_1292162843.component
            CheckCustomerAccountStatus_1292162843Impl.java
            CheckOrderHandlingPolicyforAutomaticApproval.......Interface.wsdl
            CheckOrderHandlingPolicyforAutomaticApproval_517466816.brg
            CheckOrderHandlingPolicyforAutomaticApproval_517466816.brgt
            CheckOrderHandlingPolicyforAutomaticApproval_517466816.component
            OrderHandlingFuture1.bpel
            OrderHandlingFuture1Artifacts.wsdl
            OrderHandlingFuture1Interface.wsdl
            OrderHandlingFuture1_1581013419.component
            OrderHandlingFuture1_1581013419.export
            OrderHandlingFuture1_bpel.mon
            ReviewOrder0129443853Interface.wsdl
            ReviewOrder_01925942630.component
            ReviewOrder_01925942630.tel
            ReviewOrder_01925942630_tel.mon
            ShipOrdertoCustomer01971481858Interface.wsdl
            ShipOrdertoCustomer_0567294367.component
            ShipOrdertoCustomer_0567294367.tel
            ShipOrdertoCustomer_0567294367_tel.mon
```

Figure 6-38   Structure of exported project files

► There is one `.bpel` file for the business process.

► For each activity there is a `.component` and a `.wsdl` file. The `.component` file defines the service component and the `.wsdl` file defines its interface.

► For the rule group activity there is a `.brg` and a `.brgt` file. These files define that the component must be implemented using a business rule.

► For the Java activity there is a `.java` file. This file defines a Java class with the required methods. One of the methods must be completed in the development tool.

► For human tasks there is a `.tel` file. This file defines the component as a human task.

► The `.mon` files for the business process and the human tasks define that events are generated for the Monitor.

Notes:

► All spaces and special characters are removed from the generated names.

► The numbered suffixes may be different in your case. The numbers change when you copy or rename a process.

► The `Businessitems.xsd` file contains all the business items used in the process.

We will use the interchange file in Chapter 9, "Developing the application using WebSphere Integration Developer" on page 187.

Summary

In this chapter we described how to implement changes in a business process based on simulation and analysis.

We then described how to setup implementation details for WebSphere Process Server and how to export the model into a file suitable for WebSphere Integration Developer.

7

Simulating and analyzing the Future 1 process

This chapter describes how the ClipsAndTacks Order Handling (Future 1) process was simulated and analyzed once it was revised in the Modeler.

First, we recompile all the Future 1 process model information related to the simulation, then we enter this information into the simulation attributes of the model and generate simulation snapshots.

Once simulation results and statistics are available we analyze the new simulation performance statistics from the Modeler analysis reports and make conclusions if the new revised process meets the objectives set by the company management.

Overview of simulating the Future 1 process

For the simulation and analysis of the Future 1 process we are going to perform the same steps of the process simulation as we described in Chapter 5, "Simulating and analyzing the current process" on page 55:

► Define resource requirements and decision probabilities:

– Review corporate strategy and objectives

– Review the Future 1 process model

– Define these matrixes:

• Roles and costs
• Activity durations
• Resource availability
• Probabilities on process decisions

► Define the simulation profile and attributes related to the simulation run

► Enter the simulation attributes in the Modeler

► Create and run a simulation snapshot

► Analyze simulation results

► Conclusion

ClipsAndTacks process assessment for the simulation

Following the process simulation methodology outlined above, the lead business analyst has completed the following steps:

► **Review corporate strategy and objectives**

The lead business analyst has confirmed that company objectives for the revised process are as follows:

– The high-level business objectives level are to increase revenue and reduce costs.

– Specifically, management wants to achieve these objectives:

• Reduce the average time from when orders are received to the time they are shipped to 3 days.

• Achieve an order shipping rate of 90% or better.

▶ **Review the Future 1 process model**

The revised order handling process model is shown in Figure 7-1, which is a compressed version of Figure 6-22 on page 106. The business analyst has reviewed all the process model components to prepare for the next step of defining the simulation related matrixes.
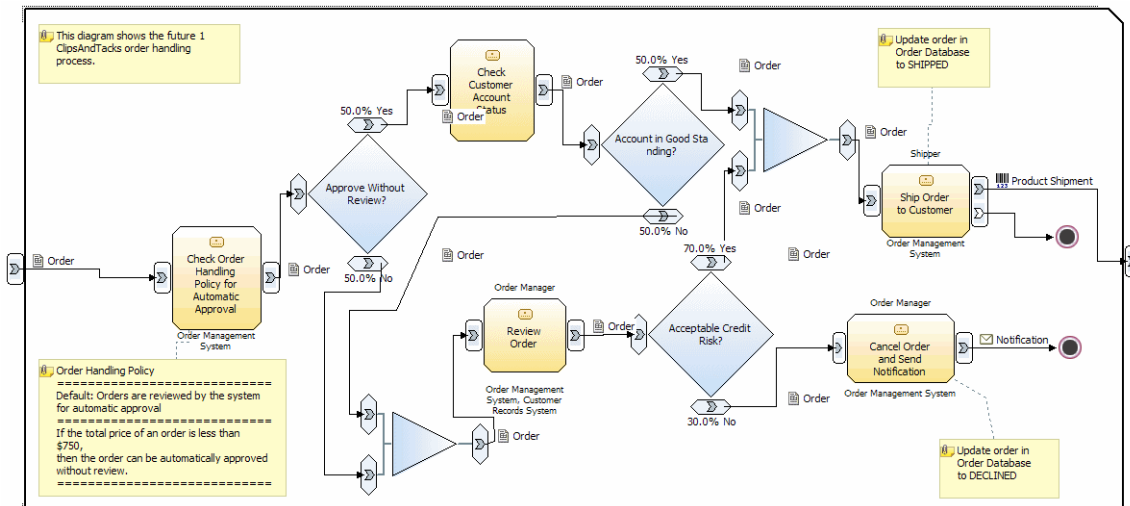
Figure 7-1   Order Handling (Future 1) process model

As a result of the revision of the future process the following matrixes are compiled:

▶ Roles and costs
▶ Activity durations
▶ Resource availability
▶ Probabilities on process decisions

Roles and costs matrix

The roles and costs matrix (Figure 7-2) shows the number of humans in each role with the assignment of the cost to activities. The cost is defined by the salary divided by the unit of measure, an hour in our case. The differences from the current process model are the following:

▶ Customer Services Representative role is removed
▶ Customer and Order Management System roles are added

Roles/ Activities	Costs $ USD per hour	Receive Order	Check Order Handling Policy	Check Customer Account Status	Review Order	Cancel Order and Send Notification	Ship Order to Customer
Customer	0	1					
Order Management System	1	1	1	1			
Order Manager	20.00				1	1	
Shipper	10.00						1

Figure 7-2   Roles matrix: Role and cost per activity

Duration matrix

The duration matrix (Figure 7-3) shows the durations of human or system tasks for a specific role and activity. In this example, there is only one role per activity, however, there could be multiple human roles for one activity. The differences from the current process model are the following:

► Customer Services Representative role is removed, Customer, Customer Records System, and Order Management System roles and corresponding activity durations are added.

► Order Manager time reduced to 15 minutes (better training).

Resources/ Activities	Receive Order	Check Order Handling Policy	Check Account Status	Review Order	Cancel Order Send Notification	Ship Order To Customer
Activity Duration	7 min 20 sec	1 sec	1 sec	15 min	2 min	16 min
Customer	7 min 20 sec					
Web application	30 sec					
Customer Records System			1 sec	1 min		
Order Mgmt System		1 sec		1 min	2 min	1 min
Order Manager				15 min	2 min	
Shipper						15 min

Figure 7-3   Duration matrix: Duration of activity per resource

Availability matrix

The resources availability matrix (Figure 7-4) shows the timetables assigned to the resources. In our business case there are two timetables used for the Future 1 process model. The timetables are defined in the modeler as follows:

► Day Shift

  – 9 working hours a day,
  – Working days are Monday to Friday
  – Working hours 8:00 AM to 5:00 PM.

► Online Application

  – 24 hours x 7 days a week

Resources/ Timetable	Day Shift	Online Application
Web Application		1
Order Manager	1	
Shipper	1	

Figure 7-4   Availability matrix: Timetable per resource

Probabilities on process decisions

The decision probabilities matrix was updated for the Future 1 process as shown in Figure 7-5.

Decision / Probability	Yes	No
Acceptable Credit Risk	70%	30%
Approve without Review	65%	35%
Account in Good Standing	85%	15%

Figure 7-5   Decision matrix: yes-no probability per decision

Populate the simulation environment

We have to enter all the required information for process simulation in the Modeler.

> **Note:** You can import the model populated with simulation information from:
>
> `SG247148\sampcode\model\Clips And Tacks Chapter 7 Solution.zip`
>
> See "Importing the current process model using the Modeler" on page 47 for instructions on how to import a model.

Enter simulation attributes in the Modeler

All the required simulation information has been defined and we are ready to enter these simulation attributes in the Modeler. The step-by-step details for entering simulation related attributes into the Modeler are described in "Populate the simulation environment" on page 60.

In this chapter we show the results of the model population with the simulation attributes.

The following tasks have to be completed:

► Populate role resources with costs and availability
► Populate activity duration for all activities
► Populate probability on the output branches for all decisions

Populate role resources with costs and availability

The main difference in process resources for the Future1 process versus the current process is that there are only two human resources participating in the Order Handling process: Order Manager and Shipper. The Customer Service Representative is not involved in this process any longer, the Web application is performing this function now.

After costs and availability for the Order Manager and Shipper are entered into the Modeler these two resource attributes look as shown in Figure 7-6.
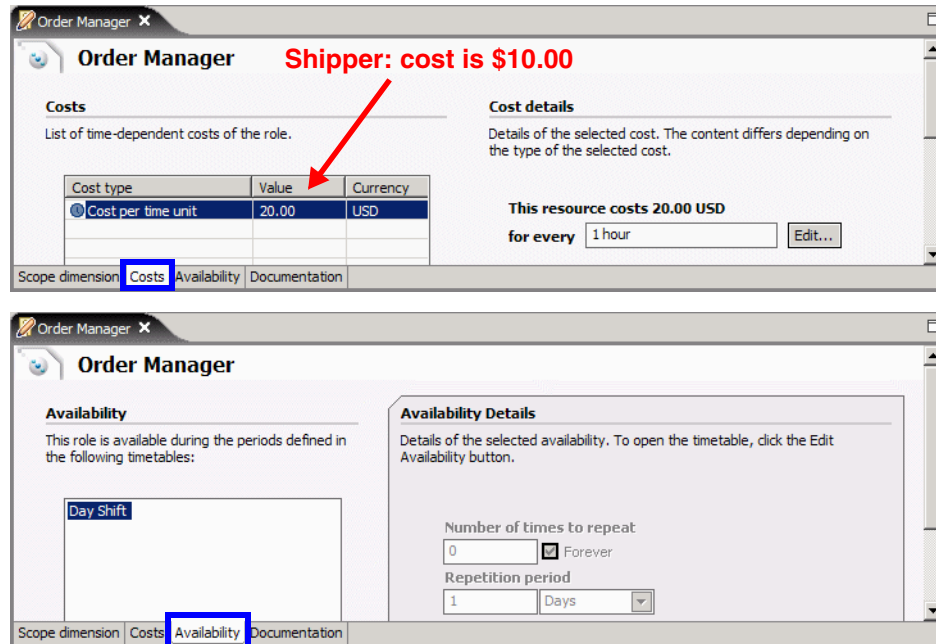
Figure 7-6   Order Manager cost and availability

Populate activity duration

The next step is to enter the processing time for the process activities. The activity duration represents the time while the activity is actively executing, rather than the elapsed time, which may include delays while waiting for a resource. It is also possible to specify the maximum amount of time that the task should wait for a resource before failing.

For our business case we will assign two types of duration information, one for the activity and one for the human tasks (role resource):

▶ To set the whole activity processing time, open the Order Handling Future 1 process.

▶ In the process flow select the activity, select the *Duration* tab and enter the duration value for each of the Order Handling Future 1 process activities as shown in Figure 7-7 using the data from Figure 7-3 on page 126.
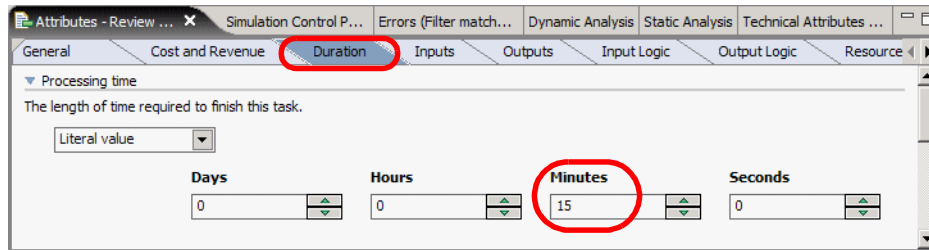
Figure 7-7   Activity duration for the Review Order activity

The activity durations populated are:

► Check Order Handling Policy: 1 second
► Check Customer Account Status: 1 second
► Review Order: 15 minutes
► Ship Order to Customer: 16 minutes
► Cancel Order and Send Notification: 2 minutes
► Receive Order (in Overall Order process): 7 minutes 20 seconds

The resource durations are entered on the Resources tab under Role requirements (for people) and Bulk requirements (for systems) as shown in Figure 7-8 for the Ship Order to Customer activity.
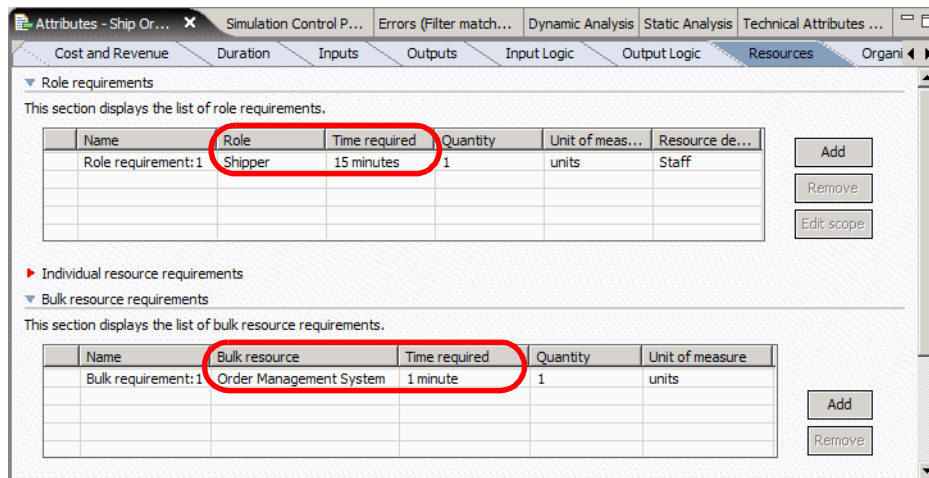
Figure 7-8   Resource durations for Ship Order to Customer

Populate probabilities on decision output branches

Enter probability to each output branch on all decisions to indicate the probability of that branch executing at any given time. Open the process flow, select every decision you plan to change, select the *Output branches* tab and enter the probability values assembled in Figure 7-5 on page 127.
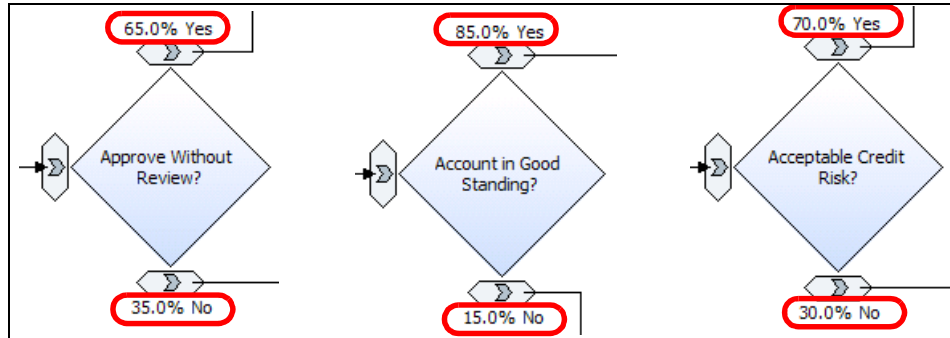
Figure 7-9   Probabilities on output branches

Once, you populated all business process values in the process model you are ready to create a simulation snapshot and then run a simulation for the Order Handling Future 1 process.

Simulating the Future 1 process

In this section we simulate the Overall Order (Future 1) process, which calls the Order Handling (Future 1) process.

Process instance simulation

Before running the simulation on your process, you have to create a simulation snapshot for this process and add a simulation profile with the appropriate attributes.

Simulation profile information

Process simulation attributes define conditions and behavior for a process for a duration of a simulation run. We will simulate 540 tokens arriving every 2 minutes.

> **Note:** For more information about simulation attributes and distribution models, refer to the product documentation: *Simulation processes → Setting Simulation Attributes → Specifying token creation settings.*

Creating the simulation snapshot

To create a simulation profile select the Overall Order (Future 1) process in the Project Tree and *Simulate* in the context menu.

For detailed instructions refer to "Creating a simulation snapshot" on page 67. Under the snapshots you will find the Overall Order (Future 1) -date- Defaults and the Overall Order (Future 1) -date- profile.

Populate the simulation snapshot

Open the Defaults and set the simulating attributes (Figure 7-10).
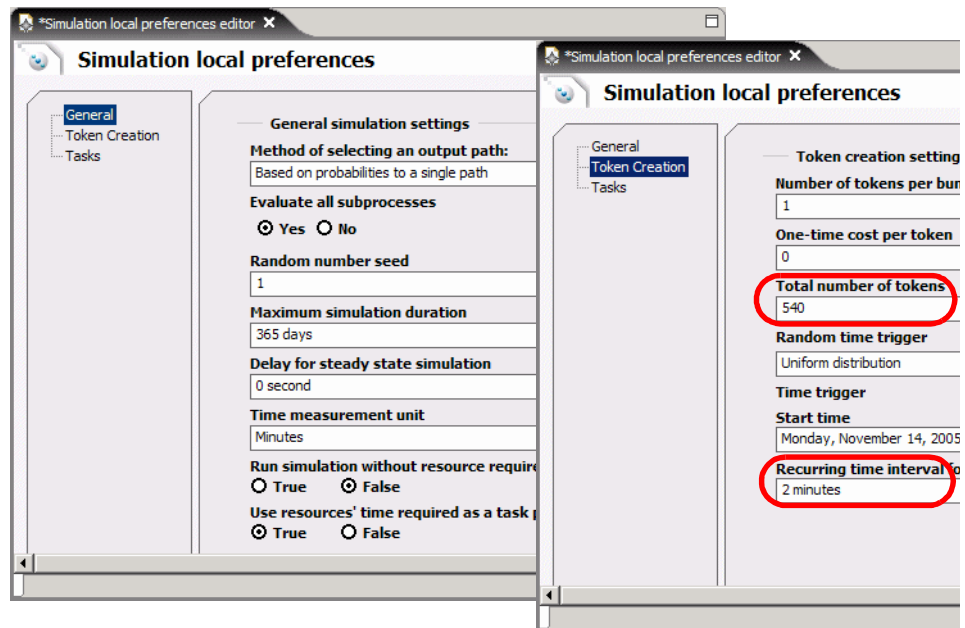
Figure 7-10   Simulation local preferences

Populate the simulation process profile

Open the profile and in the Attributes view, General tab, set these values (Figure 7-11):

► Starting date (*GMT-5*), ending date (*GMT-5*)
► Evaluate all subprocesses check: *Yes*
► Time measurement unit: *Minutes*
► Maximum simulation duration: *365 days*
► Delay of steady state simulation: 0
► Method of selecting an output path: *Based on probabilities*
► Resources' time required: *Yes*
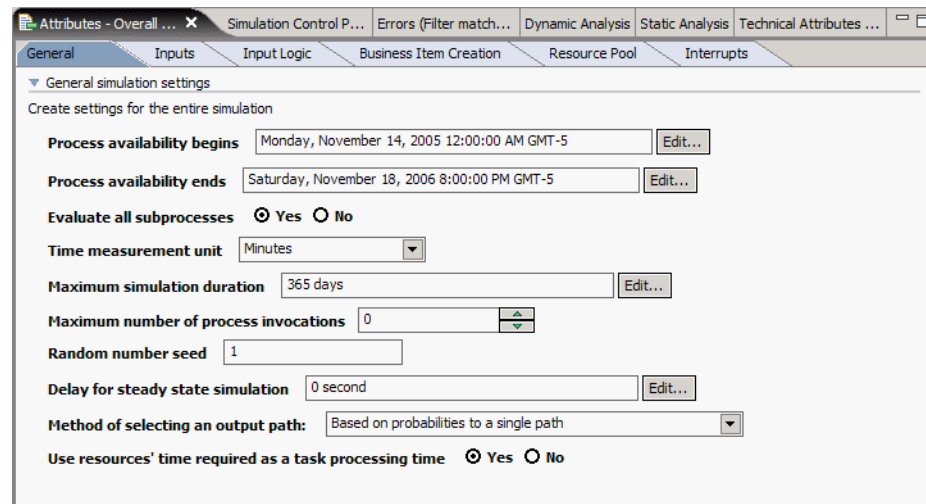
Figure 7-11   Simulation profile: General

Next populate the *Inputs* tab as shown in Figure 7-12:

► Total number of tokens: **540** (2 days of 270 requests per day)

► Select *Timetable trigger*.

► For the timetable click *Browse* and select the *Online Request* timetable.

► For the recurring time interval set the frequency to **5 minutes 20 seconds** (2880 minutes for 2 days divided by 540 tokens).

► Set the maximum tokens per interval to 270.
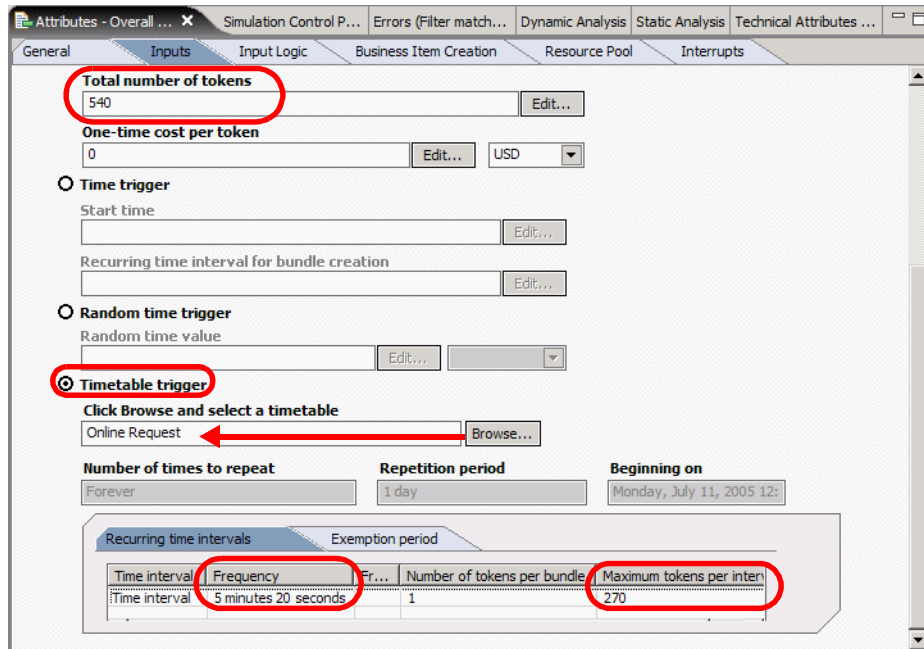
Figure 7-12   Simulation profile: Inputs

On the *Resource Pool* tab, for the three roles, deselect *Unlimited* and set the quantity to one (1), except that we employ two shippers (Figure 7-13).
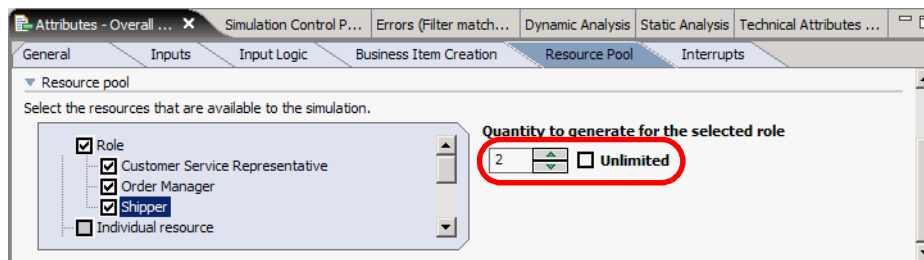
Figure 7-13   Simulation profile: Resource pool

Validate the simulation attributes

To validate the simulation attributes select the Overall Order (Future 1) simulation profile and *Profile Analysis → Profile Specification → Select All*.

This function displays the resources with costs and durations by activity as shown on Figure 7-14.

Activity Name	Activity Durat...	Requireme...	Resource or Role N...	Requirement...	Q...	Input...	Distrib
Order Handling (Future 1)							
☐ Order Handling (Future 1)/Acceptable C...							
						Yes	70%
						No	30%
☐ Order Handling (Future 1)/Account in G...							
						Yes	85%
						No	15%
☐ Order Handling (Future 1)/Approve Wit...							
						Yes	65%
						No	35%
☐ Order Handling (Future 1)/Cancel Order...	2 minutes						
		Role	Order Manager	2 minutes	1.0		
		Bulk Resource	Order Management...	2 minutes	1.0		
☐ Order Handling (Future 1)/Check Custo...	1 second						
		Bulk Resource	Customer Records ...	1 second	1.0		
☐ Order Handling (Future 1)/Check Order ...	1 second						
		Bulk Resource	Order Management...	1 second	1.0		
☐ Order Handling (Future 1)/Review Order	15 minutes						
		Role	Order Manager	15 minutes	1.0		
		Bulk Resource	Order Management...	1 minute	1.0		
		Bulk Resource	Customer Records ...	1 minute	1.0		
☐ Order Handling (Future 1)/Ship Order t...	16 minutes						
		Role	Shipper	15 minutes	1.0		
		Bulk Resource	Order Management...	1 minute	1.0		
☐ Receive Order	7 minutes 20 ...						
		Bulk Resource	Customer	7 minutes 20 ...	1.0		
		Bulk Resource	Web Application	30 seconds	1.0		

Figure 7-14   Profile analysis

Running the simulation

To run the simulation use the Simulation Control Panel view (behind the Attributes view), and click the green arrow ▶ to start the simulation (Figure 7-15).

See "Running the simulation" on page 72 for instructions on how to run with animation or without animation (use the down arrow and select *Settings*).
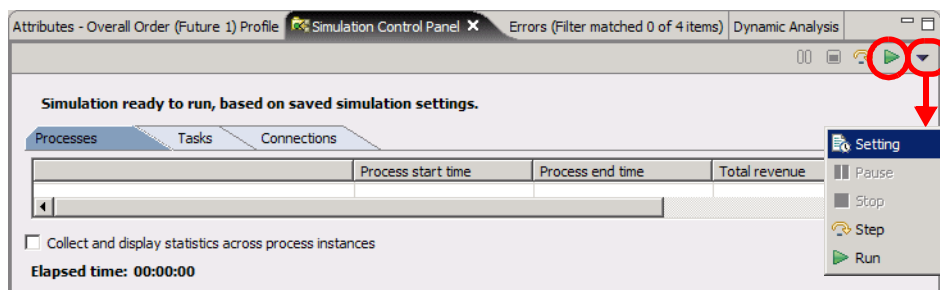
Figure 7-15   Simulation Control Panel

Analyzing the Future 1 simulation results

Once the simulation of the process is complete the business analyst generates the same reports as for the current process. He performs an analysis based on the simulation data presented in the Modeler's dynamic analysis and reporting capabilities. He compares the revised process results with those of the current process to be sure that the new process will help meet the objectives set by management.

The business analyst runs the following specific reports by selecting the results and *Dynamic Analysis*:

▶ **Process duration**—Shows average duration of each process case

▶ **Process cases summary**—Shows the percentage of each possible case in the process, for example, the percentage of orders shipped without requiring approval

▶ **Resource usage**—Shows resource usage for each process instance, including the duration of resource shortage when this resource is not available due to the work on another task

▶ **Process cost**—Shows average cost for each process case

The business analyst also runs process comparison reports and compares the results of the revised process with those of the current process to ensure that the revisions he has made are achieving the desired objectives.

When the business analyst and the management team are satisfied with the data that the simulations have produced, the business analyst is ready to create a business measures model that will enable the company to monitor, analyze, and report on actual runtime data.

Process duration

Process duration analysis shows the process instance elapsed duration and throughput details for each process case in a simulation.

To get the process duration information, select the simulation result element and *Dynamic Analysis → Process Cases Analysis → Process Duration* (Figure 7-16).

Process elapsed duration includes transfer times between activities and the elapsed durations of all activities on a path (called a *case*). A process case is defined as a set of process instances that have the same processing path. Calculations are performed per case by getting the simple average of all the process instance duration records in a case.

Figure 7-16   Analysis: Process duration

► The first case is for a **cancelled** customer order that was approved without the Order Manager review initially, but was then cancelled due to an unacceptable credit risk by the Order Manager.

► The second case reflect a **cancelled** order for a customer that was not automatically approved by the system and the Order Manager cancelled the order due to an unacceptable credit risk.

► The third case is for an order that was automatically approved without the review and **shipped** to a customer.

► The fourth case reflects a **shipped** product to a customer for an order that was automatically approved without the review, but the review by the Order Manager was required because the customer account was not in good standing.

► The fifth case is for a **shipped** product to a customer with the required review and approval by the Order Manager.

Conclusion of the business analyst

At this point, we can validate that on 540 orders, about 86% were shipped to customers and the average duration of the revised process is 4 days and 3 hours.

Process cases summary

The process cases summary analysis shows summary details for all the process cases produced during the simulation of a process.

To get this information, select the simulation result element and *Dynamic Analysis → Process Cases Analysis → Process Cases Summary* (Figure 7-17).

When you expand a case you can see which activities are on the path of a case and where the most time is spent.

	Attributes - Overall ...	Simulation Control P...	Errors (Filter match...	Dynamic Analysis ✕	Static Analysis	Technical Attributes ...

Process Cases Summary | Simulation result Friday, December 30, 2005 8:01:24 AM PST | Overall Order (Future 1) Friday, December 30, 2005 7:4 ▾

Process Duration | Simulation result Friday, December 30, 2005 8:01:2... ‖ Process Cases Summary | Simulation result Friday, December 30, 20

Case Name	A.	A.	A.	Number of Process Instances	Average Process Total Cost	Average Process Elapsed Duration	Distribution	Success St
⊞ Case 1				15	$5.67	9 days 9 minutes 39.732 seconds	2.78%	Succeeded
⊞ Case 2				58	$5.68	8 days 13 hours 11 minutes 42....	10.74%	Succeeded
⊞ Case 3				289	$2.51	2 days 5 hours 47.113 seconds	53.52%	Succeeded
⊞ Case 4				48	$7.51	5 days 13 hours 33 minutes 57....	8.89%	Succeeded
⊞ Case 5				130	$7.51	5 days 10 hours 26 minutes 31....	24.07%	Succeeded

Process Cases Summary | Simulation result Friday, December 30, 2005 8:01:24 AM PST | Overall Order (Future 1) Friday, December 30, 2005 7:4 ▾

Process Duration | Simulation result Friday, December 30, 2005 8:01:2... ‖ Process Cases Summary | Simulation result Friday, December 30, 20

Case Name	Activity Name	Average Total Cost	Average Elapsed Duration	Number of Process Inst
⊟ Case 1				15
	Acceptable Credit Risk?	$0.00	0 seconds	
	Account in Good Standing?	$0.00	0 seconds	
	Approve Without Review?	$0.00	0 seconds	
	Cancel Order and Send Notification	$0.67	4 days 20 hours 31 minutes 28 seconds	
	Check Customer Account Status	$0.00	1 second	
	Check Order Handling Policy for Auto...	$0.00	1 second	
	Merge	$0.00	0 seconds	
	Review Order	$5.00	4 days 3 hours 33 minutes 49.732 seconds	
	Order Handling (Future 1)	$0.00	9 days 5 minutes 19.732 seconds	
	Receive Order	$0.01	4 minutes 20 seconds	
⊞ Case 2				58
⊞ Case 3				289
⊟ Case 4				48
	Acceptable Credit Risk?	$0.00	0 seconds	
	Account in Good Standing?	$0.00	0 seconds	
	Approve Without Review?	$0.00	0 seconds	
	Check Customer Account Status	$0.00	1 second	
	Check Order Handling Policy for Auto...	$0.00	1 second	
	Merge	$0.00	0 seconds	
	Merge:2	$0.00	0 seconds	
	Review Order	$5.00	4 days 3 hours 53 minutes 22.249 seconds	
	Ship Order to Customer	$2.50	1 day 9 hours 36 minutes 13.75 seconds	
	Order Handling (Future 1)	$0.00	5 days 13 hours 29 minutes 37.999 seconds	
	Receive Order	$0.01	4 minutes 20 seconds	
⊞ Case 5				130

Figure 7-17   Analysis: process cases summary

Conclusion of the business analyst

All process cases run significantly faster and the most time consuming activity of Review Order has to be analyzed and possibly staffed with more than one person. For example, two Order Managers can handle the outstanding backlog faster then one. This will result in the improved customer satisfaction with the faster order turn around,

Resource usage

This analysis shows information on usage of each resource that is allocated in a process simulation.

To get this information, select the simulation result element and *Dynamic Analysis → Aggregated Analysis → Resource Usage* (Figure 7-18).

Figure 7-18   Analysis: Resources

Conclusion of the business analyst

The human tasks performed by the Order Manager is still the bottleneck in the system. ClipsAndTacks may decide to monitor this process further to see if these shortage situations are affecting customer satisfaction. The new Web application, which replaced the Customer Service Representative and is available 24 by 7, does resolve the resource shortage duration issues of the current Receive Order process.

Process cost

The process cost analysis shows the average and weighted average costs and revenue for all process instances in each case of the simulation result.

To get this information, select the simulation result element and *Dynamic Analysis → Process Cases Analysis → Process Cost* (Figure 7-19).

Figure 7-19   Analysis: Process cost

Conclusion of the business analyst

We see that the Order Handling Future 1 process costs has come down to $4.58 from the $16.22 for the current process. The average costs for the Future 1 five case processes are ranging from $2.51 to $7.51.

Process comparison analysis

You can perform several kinds of analysis to compare the weighted average analysis results for two simulated processes that use the same input parameters.

To perform a comparison analysis, select one of the simulation results that you want to analyze and *Dynamic Analysis → Processes Comparison Analysis* and then one of the following choices:

▶ Processes Duration Comparison
▶ Processes Activities Total Time Comparison
▶ Processes Cost Comparison
▶ Processes NPV / IRR Comparison
▶ Processes Break Even Comparison
▶ Processes Resources Time Comparison
▶ Processes Resources Cost Comparison
▶ Processes Classifier Weighted Average Duration Comparison
▶ Processes Classifier Weighted Average Cost Comparison

A dialog opens where you select the second simulation results that you want to compare with the first results. Click *OK*.

Select the type of process instances you want to include in the analysis:

▶ Succeeded process instances only
▶ Failed process instances only
▶ All process instances

Note: You must have simulation results for the current and the future process to be able to compare results.

Processes duration comparison

This analysis compares the weighted average duration results for two process simulations that use the same input parameters.

For each process considered in the comparison, this analysis displays the information shown in Figure 7-20.

Figure 7-20   Comparison: Process duration

Calculated values are based on weighted average values calculated according to the process duration analysis. The following information is displayed in the process duration comparison:

► Simulation Result Name—The name and the time stamp of the simulation result to which the process belongs.

► Process Name—The name of the process.

► Weighted Average Elapsed Duration—Weighted average elapsed duration of the process SUM (case Average Elapsed Duration * case Distribution / cases total Distributions) for all cases.

► Weighted Average Throughput—Weighted average throughput of the process SUM (case Average Throughput * case Distribution / cases total Distributions) for all cases.

For each column that displays a numerical result, the following comparative information is also displayed:

► Difference—Calculated as: first process value - second process value
► Percentage change—Calculated as: (difference / first process value) * 100

This reports shows a large improvement in the process duration metrics for the Order Handling Current versus Future 1 processes.

Processes cost comparison analysis

This analysis compares the weighted average cost and revenue results for two process simulations that use the same input parameters.

For each process considered in the comparison, this analysis displays the information shown in Figure 7-21. Calculated values are based on weighted average values calculated according to process cost analysis.

For each column that displays a numerical result, the following comparative information is also displayed:

► Difference—Calculated as: first process value - second process value
► Percentage change—Calculated as: (Difference / first process value) * 100

Figure 7-21   Comparison: Process cost

This reports shows a 200% improvement in the process cost metrics for the Order Handling Current versus Future 1 processes.

More information

For more information on comparison Analysis, refer to the product documentation under *Analyzing models and simulations*.

Additional considerations

There has been a substantial drop in price, but a seasoned business analyst would look at the costs outside of those generated by the Modeler, or add additional costs to the model based on discussions with the systems architect, who will build the application.

Specifically, the business analyst would consider the costs for buying, licensing, developing and maintaining the proposed application. Also, the business analyst would check the assumptions about how long the customer might spend entering the order information using the Web application with a usability expert.

Some questions that the systems architect would ask the business analyst:

► How many customers will be using the application right away? How many in the near future?

► How many customers might use the application at the same time?

► Is it critical to guarantee completion of an order (in which case a messaging system would have to be added to the expense) or is an occasional failure in the system with a returned error message acceptable?

► Should the application have a service department that a customer could reach at anytime?

The answers to these questions would drive the total cost of the application.

Perspective

A wise business analyst would put the analysis figures in perspective;

► For example, even with the dramatic improvement promised by the online application, if ClipsAndTacks were a small corporation and the costs for creating the application were substantial, then the business analyst would still not proceed.

► However, if the ClipsAndTacks were a medium to large size corporation with in-house developers and most of the software already purchased, then the online application would make more sense—not just from an efficiency point of view, but also from the opportunity to increase sales by offering the service nationally (or internationally) on the Web.

Summary

In this chapter we simulated the Order handling (Future 1) process and analyzed the results. We can see a major improvement by having the customer handle the data entry for orders. The implementation of the automatic approval and the customer credit check also helps for the overall achievement of 90% shipped orders.

8

Creating the business measures model

This chapter describes the steps performed with a new feature of the WebSphere Business Modeler Version 6, the Business Measures editor. We will define all the relevant information about the process that will be monitored by WebSphere Business Monitor Version 6.

In Chapter 7, "Simulating and analyzing the Future 1 process" on page 123 we worked with the simulation of the process. Now we have the opportunity to measure the same process in action, real-time, enabling you to measure the performance of the process, retrieving the real information that flow through the Order Handling (Future 1) process.

The topics outline of this chapter is as follows:

► Business measures

► ClipsAndTacks key performance indicators (KPI)

► Dimensional analysis and situation events

► Export the business measure model

Business measures

In this section the we discuss the following topics:

► Introduction to business measures
► Model, assemble, deploy and manage the complete lifecycle
► Elements of the Business Measures editor
► WebSphere Business Monitor views

Introduction to business measures

A business measures model describes business metrics, their dependencies on incoming events, conditions warranting business action (business situations), and outbound events that represent notifications of such conditions and may trigger business actions.

Specifically, the business measures model describes how to perform the following:

► Gather information, from real-time events

► Aggregate information to calculate higher-level business metrics and key performance indicators (KPIs)

► Represent the calculated values on a number of dashboard views and reports, based on the business needs

► Emit events in reaction of business conditions that could be used to trigger actions.

The key to having a successful set of business measures is deciding upon the vital measures that are linked to your business success. You study the process and the business goals to determine which business measures will be needed from the executing process. Once the measures have been established, you can monitor and evaluate them.

Model, assemble, deploy, and manage the complete lifecycle

WebSphere Business Modeler enables you to transform business process models to IT-level models. You can export Business Process Execution Language (BPEL) and business measures, then use WebSphere Integration Developer to import the BPEL model and assemble the services and components. Once you finish the assemble you can deploy to WebSphere Process Server to execute your application. You can import the business measures into WebSphere Business Monitor to monitor and evaluate the business process at runtime (Figure 8-1).
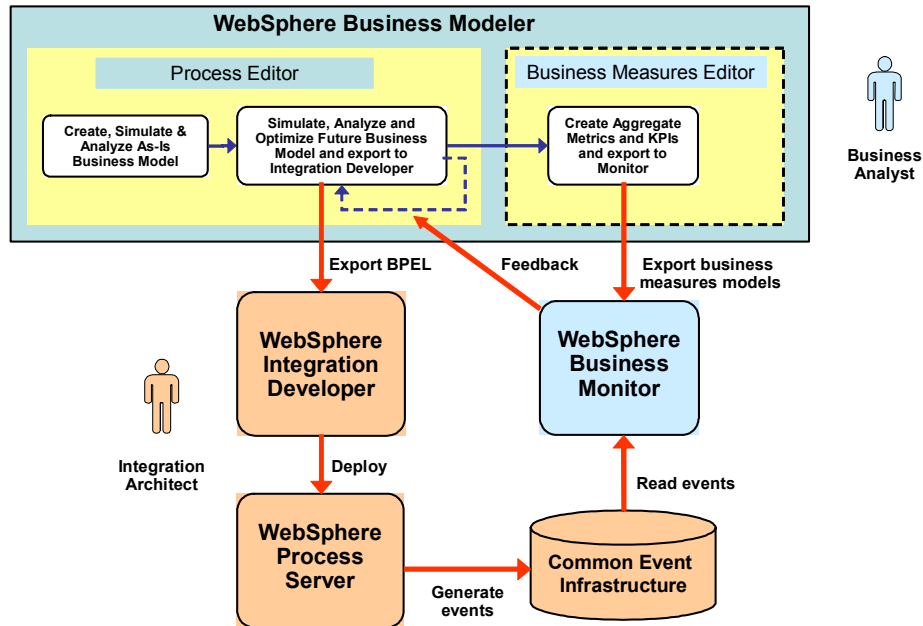
Figure 8-1   BPM complete lifecycle

WebSphere Business Monitor displays dashboards (Figure 8-2), which are containers (portals) that enable you to monitor different aspects of business performance. You can use WebSphere Business Monitor to capture real-time, work-in-process items and perform corrective actions by reassigning, re-prioritizing, or suspending them. You can also display real-time data from work items produced by the runtime, and retrieve and view the historical data of the monitored process.
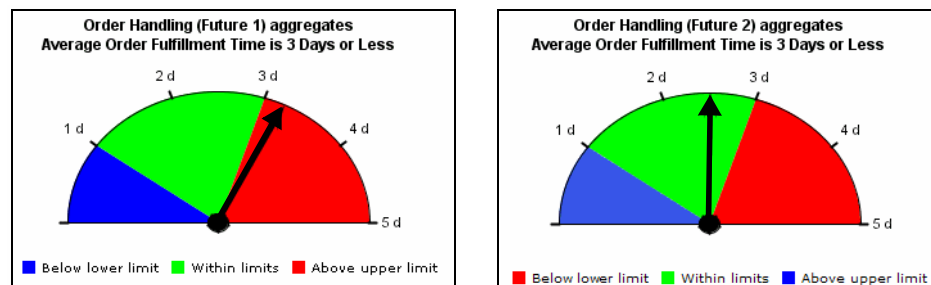
Figure 8-2   Dashboard example

Elements of the Business Measures editor

You can combine WebSphere Business Modeler with WebSphere Business Monitor for business performance management, the process of continuously defining, analyzing, and improving a business process. Starting with a process model in the Modeler, you create a business measures model that enables you to specify the performance management aspects, including business metrics, key performance indicators (KPIs), and situation events.

Figure 8-3 shows the business measures elements relationship.
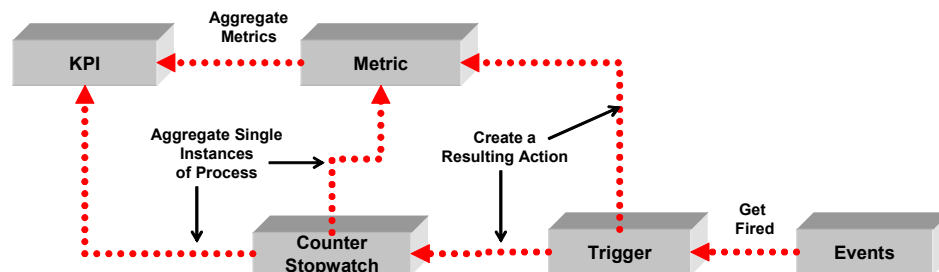
Figure 8-3   Business measures relationship

We describe the following business measures elements:

- ► KPIs (key performance indicators)
- ► Metrics
- ► Stopwatches
- ► Counters
- ► Triggers
- ► Properties
- ► Events
- ► Expression builder
- ► Technical diagram

Key performance indicators

Key performance indicators (KPI) are the detailed specifications required to track business objectives. Each KPI is associated with a specific process, and is quantifiable, measurable, and results-oriented.They are defined within the context of the Business Measures editor and evaluated by WebSphere Business Monitor.

The following attributes can be configured for KPIs:

- ► **Name**—Specifies the name of the key performance indicator that will be show in the WebSphere Business Monitor interface.

► **Type**—Specifies the data types that will be handle by the measure. For example, the types could be: Integer, Long, Duration, DateTime and Boolean.

► **Aggregation function**—Specifies the function that will be execute against the aggregation source. The available functions are: total, average, minimum, maximum and user defined.

► **Aggregation source**—Specifies the source business measure element that will be aggregate.You can perform calculation using more than one source element and conversions through the expression builder.

► **Use target and limit**—Using the target attributes you have the option to specifies a target and allowable margins. Another option is have just lower and upper limits without the a target value, to form a range of performance for the process to achieve.

► **Lower target margin** (%) and **upper target margin** (%)—Specifies the allowable range of performance of a target.

► **Lower limit** and **upper limit**—Specifies the allowable range of performance without the use of a target.

Depending of how you configure the KPI attributes, they map to views in WebSphere Business Monitor (Figure 8-4).

	Activity Instance View	KPI View	Scorecard View	Gauges View
Target with lower and upper margin defined (%)	✔		✔	✔
Target without lower and upper margin defined (%)	✔			
Upper and lower limits without a target	✔	✔		✔

Figure 8-4   WebSphere Business Monitor views

KPIs can optionally generate situation events that can cause business actions, for example, an ATM running out of cash. In this case an administrator will use the action manager in WebSphere Business Monitor to specify what happens when the situation event is received

Note: The following elements can be used by KPIs as source for aggregation or user defined calculations:

► Metrics
► Stopwatches
► Counters

Metrics ▭

A metric is a measurement of a process or a process element that is used to assess business performance. A metric can be used alone or in combination with other metrics to define the calculation for a KPI, which measures performance against a business objective. The values that metrics return are captured and evaluated using WebSphere Business Monitor.

There are three ways to define metrics:

► **Business item metric**—Uses the data carried by a business item through one run of the process.

► **Instance metric**—Return the result from one run of the process.

► **Aggregate metric**—Provide calculations across multiple runs of the process, for finding the average, maximum, minimum, or total number of occurrences.

Examples of business metrics are a supplier's average response time and the cost of the risk assessment step in an insurance process.

> **Note:** The following elements can be used by a metric as source for aggregation, user defined calculations, or resulting actions:
> ► Stopwatches
> ► Counters
> ► Triggers

Stopwatches ▭

Stopwatches are specialized business measure elements that keep track of durations of an activity, group of activities, or process. You can start, stop and reset stopwatches according with the duration measure that you want to retrieve. You can use stopwatches as a source for business metrics and KPI that often depend on elapsed time.

Counters ▭

Counters are specialized business measures elements that keep track of the number of occurrences of some situation or event. Counters can be increased by one, decreased by one, or set to zero by specific triggers.

You can use a counter to track, for example, the number of times a task was started within a process where the task is contained in a loop. Counters can be used to measure numbers within one run of the process, but you can reuse them within a KPI or aggregate metric to determine the average, maximum, minimum, or total across multiple runs of the process.

Triggers �

A trigger is a mechanism that detects an occurrence, and initiates an action in response. For example, you could set a trigger to update a metric each time a activity ends.

Triggers can detect such things as the arrival of an input at a specific task, a change in the value of a metric, or a recurring period such as every three hours. In response, triggers can do things such as start stopwatches, increment counters, and change the values of metrics.

Triggers can detect any of the following occurrences:

- ► A change of state
- ► A change in a metric
- ► A change in a counter
- ► A specific time interval
- ► The arrival of an input (to a task or process)
- ► The production of an output (of a task or process)

A trigger, in turn, can cause any of the following actions:

- ► Evaluation of a metric
- ► A change in a stopwatch (stop, start, or reset)
- ► A change in a counter (increment, decrement, or set to zero)
- ► The production of a situation event

Properties

The Business Measures editor provides predefined measures for each process, task, subprocess, and loop. In the intermediate and advanced modeling modes, you can view these predefined measures in the Properties tab of the Attributes view.

The predefined measures are grouped together into templates. When the process runs, you can view the values of these measures in the Active Instances view of WebSphere Business Monitor. You can also view the historical values of these measures in the Dimensions and Reports views.

We can use predefined measures to build higher level measures by using them within the calculations for instance metrics. Within the expression builder, select *Modeling Artifacts* to see the available properties for each process and task.

We have two types of predefined measures, for process and for activities, shown here with some examples of properties:

- ► Process properties
  - – Process State

- Process Start Time
- Process Elapsed Duration Timer
- Process isDelayed

▶ Activity Properties

- Activity State Metric
- Activity Start Time Metric
- Activity Elapsed Duration Timer
- Activity Assigned User Metric
- Activity Counter

Events

In a real-time monitor, such as WebSphere Business Monitor, events are both the sources of up-to-date state information from which the values of metrics are derived, and the means of notification, which can trigger business actions. The first kind are called situation events and the second kind are called inbound events.

Situation events

Situation events are generated when a business situation arises, such as a printer running out of paper or an ATM running out of cash. Other business situations could be an unacceptable customer response time or a declining sales count that has dropped below a predefined threshold.

Inbound events

The Business Measures editor provides predefined inbound event definitions for each process, task, subprocess, and loop. In the Intermediate Business Modeling and Advanced Business Modeling modes, you can view these predefined inbound event definitions in the Event Type tab of the Attributes view. We have two types of inbound event, state change event and predefined event. Here are some examples of events:

▶ State change event—Each time a process or a task in the process changes its state (for example, going from ready to running, or from running to completed), it emits a predefined event called a state change event. You can use these events as the source for state change triggers that are used to update instance metrics, stopwatches, and counters.

▶ Predefined event—Besides state change event definitions, other predefined event definitions are also provided. Here are some examples of additional predefined events that are published in the runtime environment:

- Activity work item created
- Activity work item deleted
- Activity work item transferred

Expression builder

The expression builder is a tool that enables you to identify, describe, and compose expressions for your models. For example, in the KPIs settings when you have a user defined aggregation function, the expression builder could be used to perform conversions and any kind of calculations using modeling artifacts, function, sub-expression, and numbers.

Technical diagram

In WebSphere Process Server mode, a *Technical Diagram* tab is available in addition to the regular diagram in the Business Measures editor. This tab enables you to view the runtime artifacts of your business measures models directly from within the Business Measures editor.

WebSphere Business Monitor dashboard views

The key performance indicators (KPIs) and other business measures that you define in the Business Measures editor are displayed in the WebSphere Business Monitor dashboard views. A dashboard view is a type of display supported by one or more dashboard categories. The display is targeted to address specific functions or responsibilities of a particular role. Each view type supports a number of display properties to customize the display. The dashboard view types are:

► **Active Instances**—Displays the instances of a specific process and the runtime values of selected business measures.

► **KPIs**—Displays the details of individual KPIs.

► **Scorecards**—Primarily supports executive users. A scorecard groups the KPIs of particular interest to executives.

► **Gauges**—Visually represents the values of KPIs relative to KPI limits or relative to the KPI target, in the form of a gauge that looks like an automobile speedometer or tachometer.

► **Dimensions**—Dimensional analysis provides business insight by summarizing business metrics. It organizes data into levels of detail that you can drill down to extract significant information.

► **Reports**—Displays performance reports showing the values of metrics aggregated over a period of time in tables and graphs.

More information

The detailed documentation of business metrics and how to create business measures with WebSphere Modeler Version 6 is available in the help documentation included with the product or refer to:

http://www.ibm.com/software/integration/wbimodeler/

ClipsAndTacks key performance indicators

In this section the we discuss the following topics:

► Definition of ClipsAndTacks KPIs

► How to create the business measures model

► Detailed steps to create two KPIs:

  – Average order fulfillment is three days or less
  – Number of approved orders is greater than 90%

Definition of ClipsAndTacks KPIs

The ClipsAndTacks management team wants to be able to measure the results of the revised process when it is implemented to ensure that it is helping to meet the company's business objectives. To measure the revised order handling process, management has identified two key performance indicators that will measure the success of the new process.

► Average order fulfillment is 3 days or less
► Number of approved orders is greater than 90%

Creating the business measures model

We work with the Order Handling (Future 1) process to define the business measures. Set the modeling mode to advanced, by selecting *Modeling → Mode → Advanced*.

> **Note:** You can import the model populated with business measures from:
>
> `SG247148\sampcode\model\Clips And Tacks Chapter 8 Solution.zip`
>
> See "Importing the current process model using the Modeler" on page 47 for instructions on how to import a model.

Preparation

There is currently a restriction that when exporting a model to both Integration Developer and Business Monitor **you can only export the complete project**, that is you cannot select only the Order Handling (Future 1) project (see Figure 6-37 on page 119).

Therefore, we have deleted the Order Handling (Current) process and the Overall Order (Future 1) process from the project. We also deleted the three databases so that no errors are reported in the project even in Process Server modeling mode.

To create the KPIs the first step is create the business measures model.It enables you to create the measure elements:

► If you have not already done so, make business measures visible in the Project Tree, click the *Project tree filters* icon, and deselect both *Business measures* (Figure 8-5).
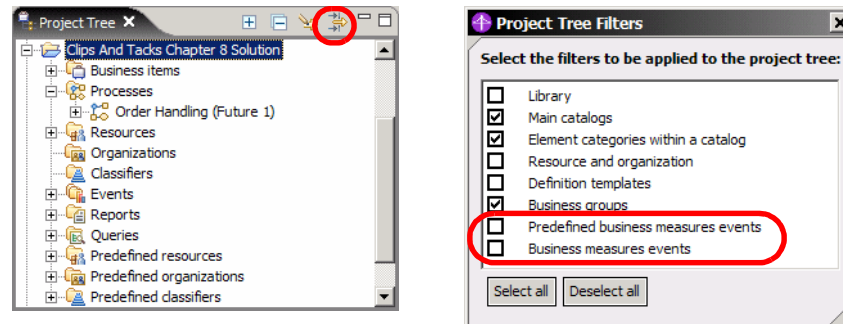
Figure 8-5   Project tree filter

► Select the Order Handling (Future 1) process in the Project Tree and select *Create Business Measures*.

  This option is only available if you have not previously created business measures for this process. Otherwise, you must select *Synchronize* to synchronize the process with the business measures.

► A dialog opens (Figure 8-6) and displays the elements of the process that are available for monitoring. Elements that cannot have associated business measures, such as repositories, are not listed in the dialog box and simply copied to the diagram in the Business Measures editor.

► Leave the defaults and click *OK* to create the business measures model.
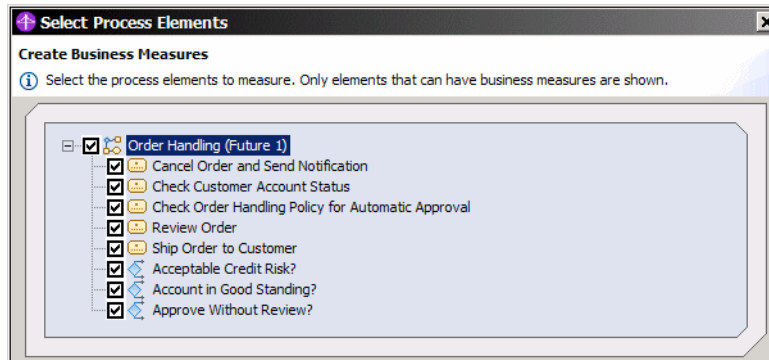
Figure 8-6   Element selection to create business measures model

► A new business measures model node is created in the Project Tree under the process node (Figure 8-7).
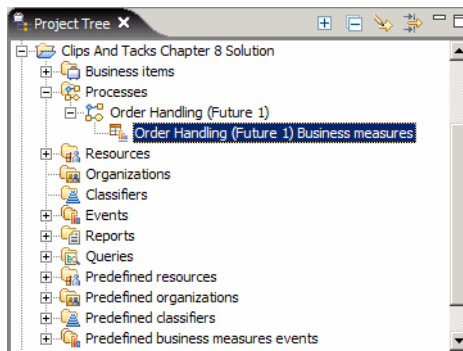
Figure 8-7   Business measures: Model

► The Business Measures editor opens on the KPIs and Aggregate Metrics tab (Figure 8-8).
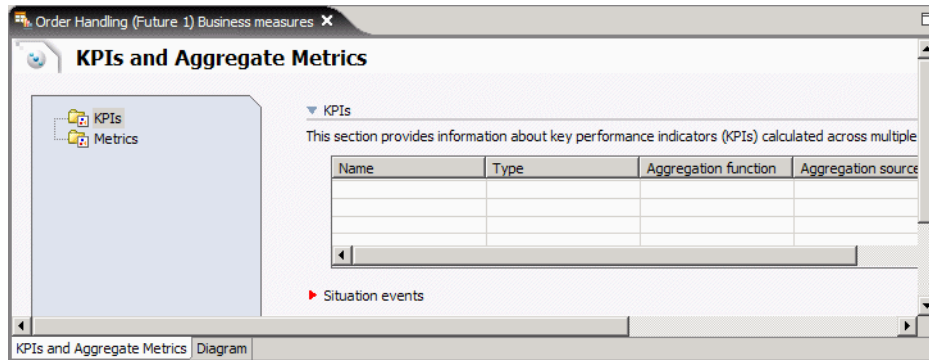
Figure 8-8   Business measures: KPIs and Aggregate Metrics

▶ Select the *Diagram* tab (Figure 8-9) to see a graphical representation of the business measures model. The attributes of the business measures model are displayed in the Attributes view. Set the duration of the process to 7 days.
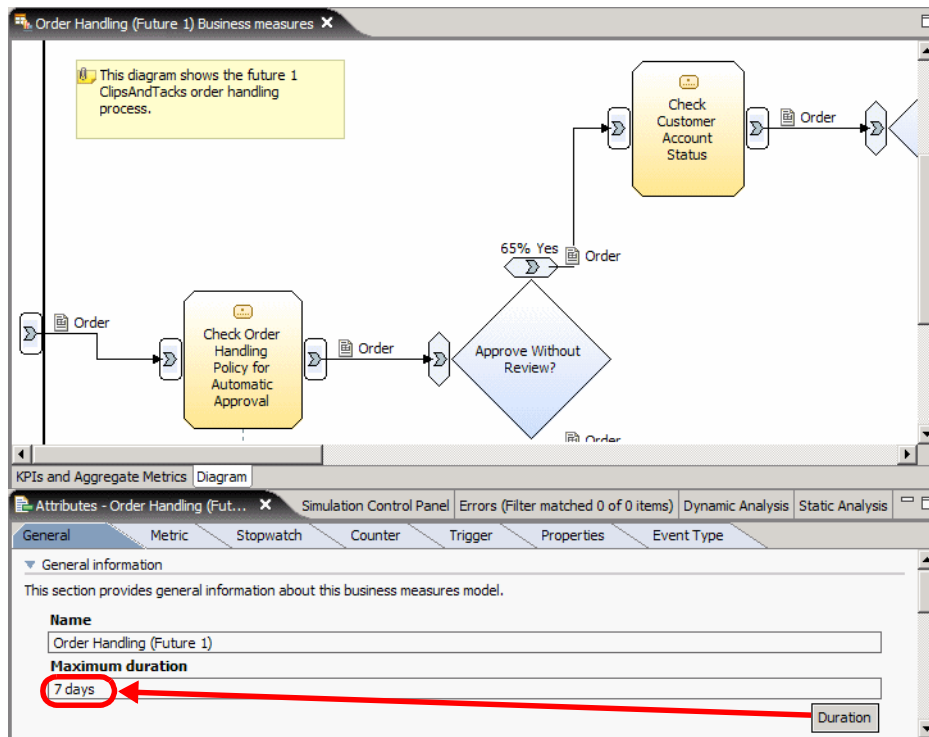
Figure 8-9   Business measures: Diagram and attributes

Implementing the KPIs

Now we implement the ClipsAndTacks KPIs.

Average order fulfillment is 3 days or less 📊

In this section we provide the following:

▶ Overview of the *Average Order Fulfillment is 3 Days or Less* KPI
▶ Detailed steps to create the business measures elements

Overview of Average Order Fulfillment is 3 Days or Less KPI

The Average Order Fulfillment is 3 Days or Less KPI will allow the management team to monitor (through the WebSphere Monitor) the average amount of time to ship orders to customers. To analyze this information will help ClipsAndTacks to reduce the average time from when orders are received to the time they are shipped to 3 days. The simulation of the Order Handling (Future 1) process in Chapter 7, "Simulating and analyzing the Future 1 process" on page 123 indicates a average time of over 4 days to complete the execution of the process.

We will create the following elements:

▶ Triggers

– New Order Trigger
– Ship Order to Customer Trigger

▶ Stopwatch

– Order Fulfillment Timer

▶ KPI

– Average Order Fulfillment is 3 Days or Less

• Lower limit: 1 day — 24 hours
• Upper limit: 3 days — 72 hours

The sequence of this KPI in a single process instance is described in Figure 8-10:

1. The Check Order Handling Policy activity changes status to *Running*.

2. The New Order Trigger is fired.

3. The New Order Trigger starts the Order Fulfilment stopwatch.

4. The Order Fulfilment stopwatch measures the elapsed duration.

5. The Ship Order activity changes the status to *Completed*.

6. The Ship Order to Customer Trigger is fired.

7. The Ship Order to Customer Trigger stops the Order Fulfilment stopwatch.

8. The Average Order Fulfilment KPI aggregates the Order Fulfilment stopwatch value.

9. The Average Order Fulfilment KPI feeds the values in the WebSphere Business Monitor dashboards and dimensions.
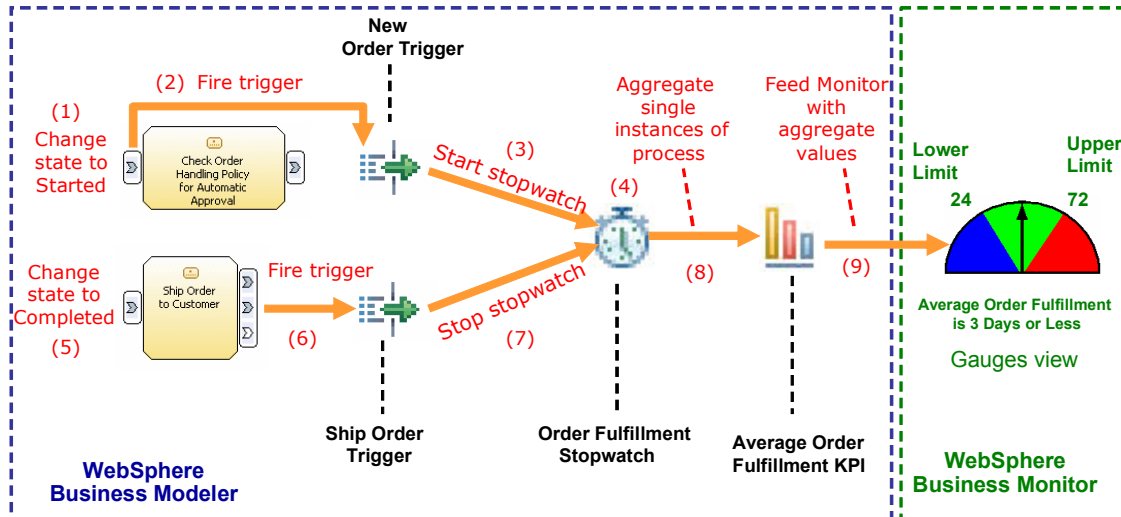
Figure 8-10   Calculating the Average Order Fulfillment Time is 3 Days or Less KPI

Detailed steps to create the business measures elements

The following elements have to be created.

- ► New Order Trigger
- ► Ship Order to Customer Trigger
- ► Order Fulfillment Timer
- ► Average Order Fulfillment is 3 days or less (KPI)

Create the New Order Trigger

Now we have to define a trigger that identifies the start time for shipment. We looked at the process and concluded that this trigger have to fire at the start of the Check Order Handling Policy for Automatic Approval activity.

- ► Open the Business Measures editor for the Order Handling (Future 1) business measures process from the Project Tree.

- ► By default the Business Measures editor opens on the *KPIs and Aggregate Metrics* tab. Select the *Diagram* tab (Figure 8-9 on page 157).

► Select the Check Order Handling Policy for Automatic Approval activity to open up the Attributes view of this activity.

► Select the *Trigger* tab (Figure 8-11).

► Click *Add* to create a trigger with the name `Trigger1`.

► Overtype the name with `New Order Trigger`.

► In the Source category column select *State change* (preselected).

► In the Source column, click the `...` button and select *Activity State Started* for the transition event.

► Notice the trigger symbol that appears in the diagram above the activity.
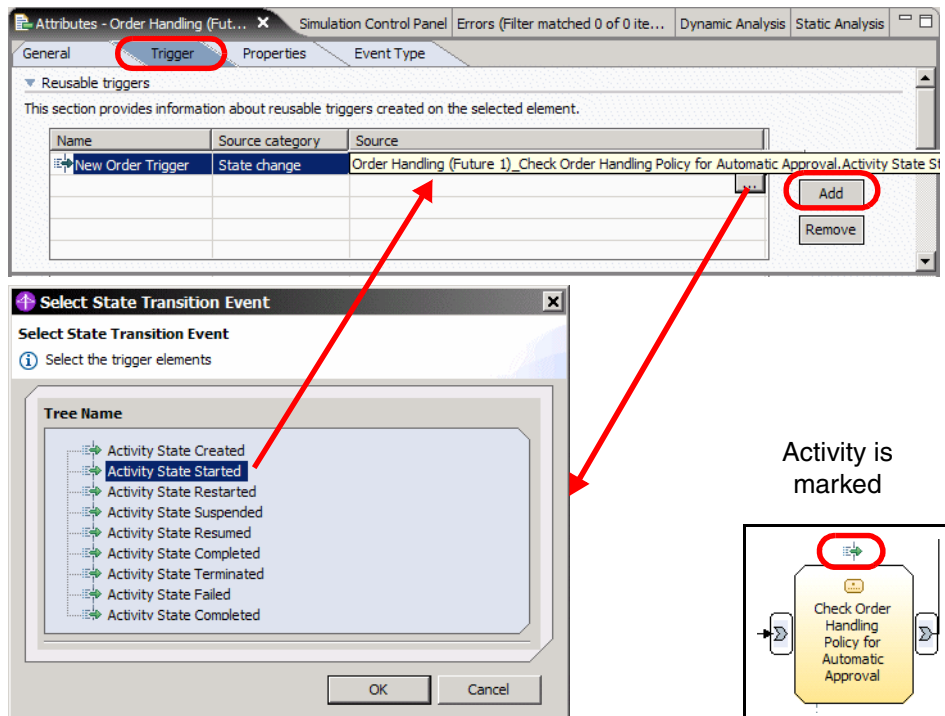
Figure 8-11   Creating a trigger (New Order Trigger)

Create the Ship Order to Customer Trigger

Now we have to define a trigger that identifies the end time for shipment. This trigger has to fire when the Ship Order to Customer activity is completed.

Follow the same steps as for the New Order Trigger, using a name of `Ship Order to Customer Trigger`, and select *Activity State Completed* in the transition event dialog (Figure 8-12).
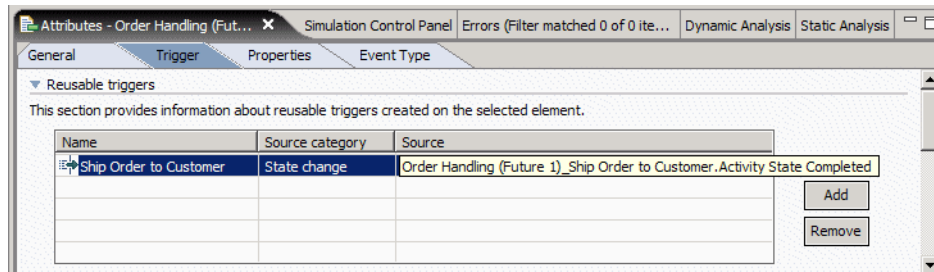
Figure 8-12   Creating a trigger (Ship Order to Customer Trigger)

We have finished the creation of the two triggers; the next step is to create the stopwatch that will be fired by these triggers.

Create Order Fulfillment Timer stopwatch

We will define the Order Fulfillment Timer stopwatch that gets started on the start of the Check Order Handling Policy for Automatic Approval activity and stopped after the Ship Order to Customer activity has completed.

This stopwatch captures the order fulfilment duration that will be used to build the Average Order Fulfillment is 3 Days or Less KPI. Stopwatches work only within a single runtime process instance bases.

► Click on the white space in the process diagram to set definitions for the process, instead of for an activity.

► Select the *Stopwatch* tab (Figure 8-13).

► Click *Add* to create an entry with the name `Stopwatch1` which we change to `Order Fulfillment Timer`.

► In the Triggered actions section click *Add* twice to create two triggers.

► Change the resulting action to *Stop* for the second trigger.

► In the Trigger column click the ... button in the first row. In the Select Trigger dialog, select *Select existing trigger*, expand the Check Order Handling Policy activity and select the *New Order Trigger*.

► For the second trigger select the *Ship Order to Customer Trigger*.

► Notice the stopwatch symbol that appears in the diagram above the process.
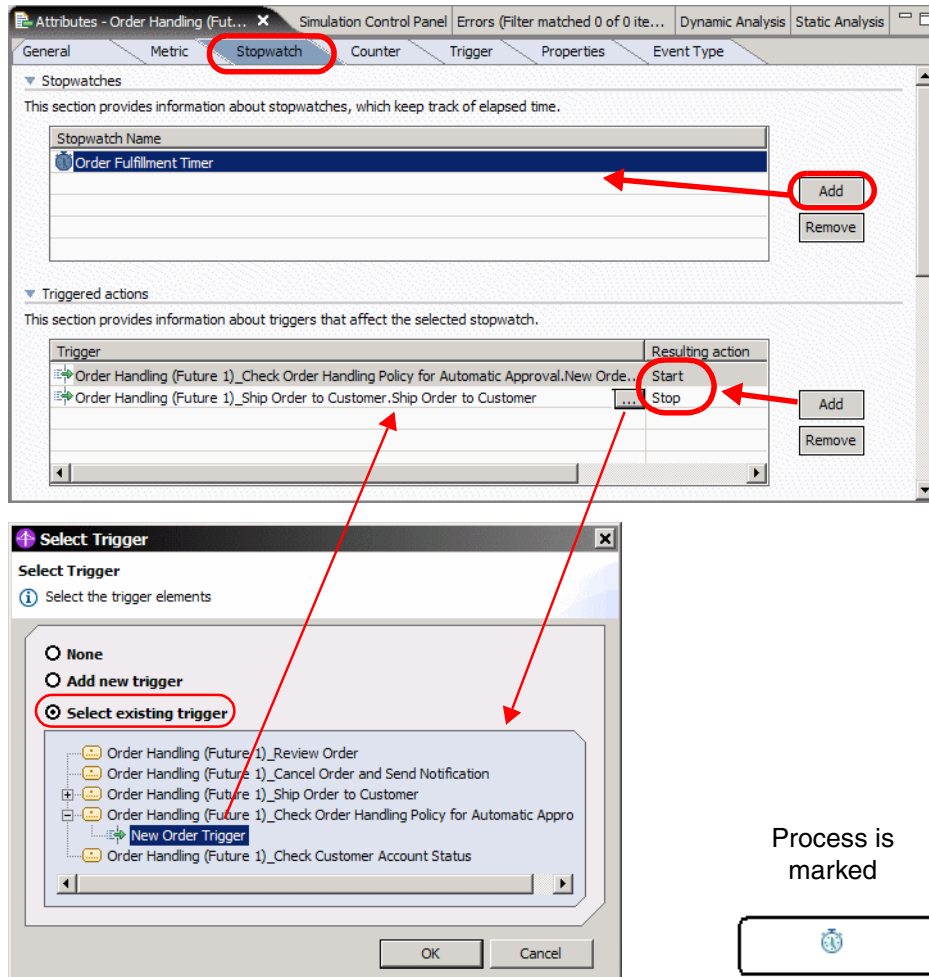
Figure 8-13   Creating a stopwatch

We have finished the creation of the Order Fulfillment Timer stopwatch, the next step is create the KPI.

Create the Average Order Fulfillment is 3 Days or Less KPI

At this point we have defined the triggers and the stopwatch that are the two components of this KPI. The configuration of this KPI will enable the aggregation of the single instances durations of the Order Fulfillment Timer stopwatch and generate an average of that value.

► Select the *KPIs and Aggregate Metrics* tab.

► Select *KPIs* in the left section.

► In the KPIs section, click *Add* under the table and a row is added in the table (Figure 8-14).
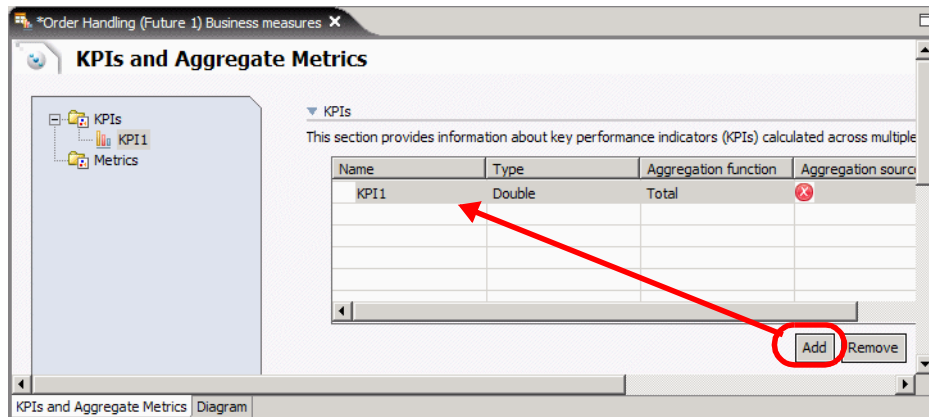
Figure 8-14   Creating a KPI

► Use the following attributes (Figure 8-15):

   – Name: `Average Order Fulfillment is 3 Days or Less`
   – Type: *Duration*
   – Aggregation function: *Average*
   – Aggregation source: *Order Fulfillment Timer*
   – Use Target: *deselect*
   – Lower limit: *24 hours*
   – Upper limit: *72 hours*
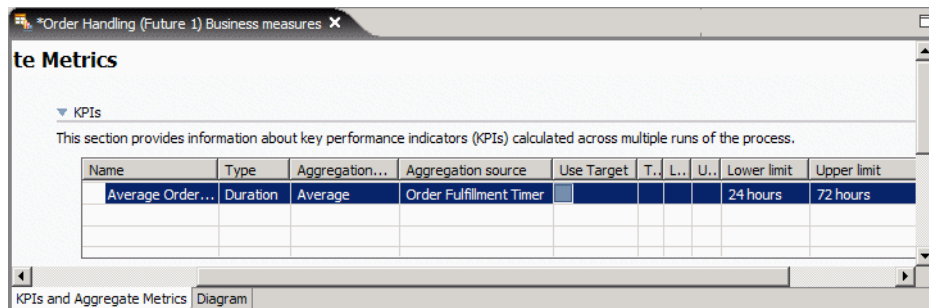
Figure 8-15   KPI Average Order Fulfillment is 3 Days or Less

Note: Lower limits and upper limits specify the allowable range of performance with a graphical representation in the Business Monitor.

Number of approved orders is greater than 90% 📊

In this section we provide the following:

► Overview of Number of approved orders is greater than 90% KPI

► Detail steps to create the business measures elements

Overview of 90 Percent of Orders Are Approved KPI

The 90 Percent of Orders Are Approved KPI will allow the management team to monitor the percentage of approved orders, the analyzes of this information will help ClipsAndTacks to achieve an order approval rate of 90% or better. The simulation of the Order Handling (Future 1) process indicates a percentage of approved orders of about 90%.

In this step we will create or reuse the following business measures elements (Figure 8-16):

► Triggers

  – New Order Trigger (reuse)
  – Ship Order to Customer Trigger (reuse)

► Counters

  – Order Counter
  – Shipped Order Counter
  – Declined Order Counter (not used for KPI)

► Aggregate metrics

  – Total Orders
  – Shipped Orders
  – Declined Orders (not used for KPI)

► KPI

  – 90 Percent of Orders Are Approved

    • Target: 90%
    • Lower target margin: 85.5% (5% underachievement)
    • Upper target margin: 90% (0% overachievement)

The sequence of this KPI in a single process instance is described in Figure 8-16:

1. The Check Order Handling Policy activity changes status to *Running*.

2. The New Order Trigger is fired.

3. The New Order Trigger starts the Order Counter.

4. The Order Counter is increased by one.

5. The Total Orders metric aggregates the Order Counter value.

6. The Ship Order activity changes the status to *Completed*.

7. The Ship Order to Customer Trigger is fired.

8. The Ship Order to Customer Trigger starts the Shipped Order Counter.

9. The Shipped Order Counter is increased by one.

10. The Shipped Orders metric aggregates the Shipped Order Counter value.

11. The 90 Percent of Orders Are Approved KPI calculates the percentage from the Shipped Orders metric and Total Orders metric.

12. The 90 Percent of Orders Are Approved KPI feeds the values into the WebSphere Business Monitor dashboards and dimensions.
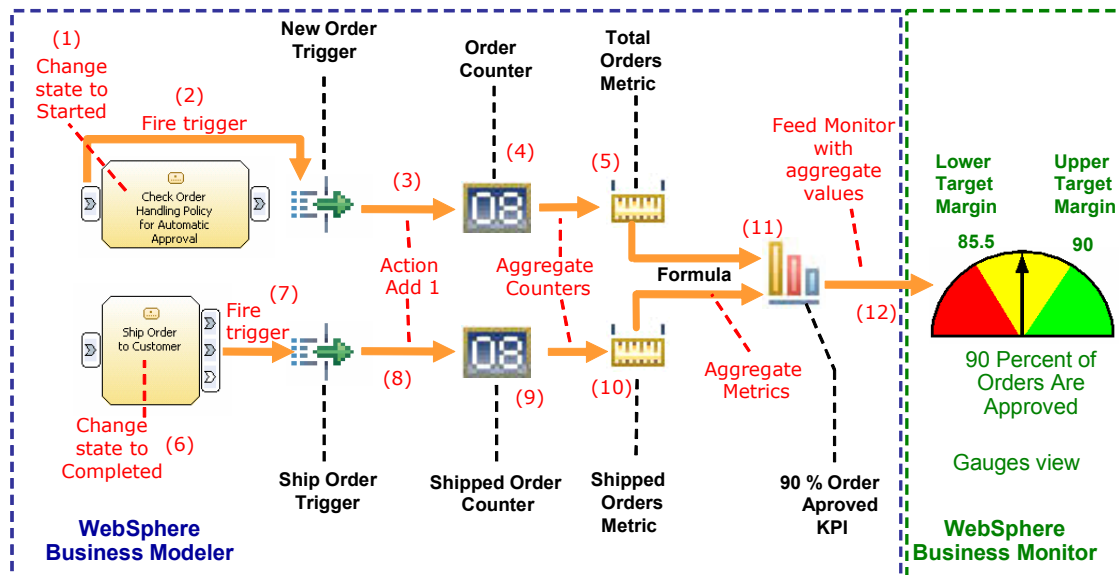
Figure 8-16   Calculating the 90% of Orders Are Approved KPI

Detail steps to create the business measures elements

The following elements have to be created.

► Order Counter
► Shipped Order Counter
► Total Orders aggregate metric
► Shipped Orders aggregate metric
► 90 Percent of Orders Are Approved KPI

Create the Order Counter 08

We define an Order Counter as a counter that gets increased by one on the start of the Check Order Handling Policy for Automatic Approval activity.

The information provided for this counter will be used to build the Total Orders aggregate metric. Counters work only within a single runtime process instance bases.

► Select the *Diagram* tab to see a graphical representation of the business measures model.

► Click on the white space in the process model to set definitions for the process, instead of for a activity.

► Select the *Counter* tab (Figure 8-17).

► Click *Add* to create an entry with the name `Counter1`, which we change to `Order Counter`.

► Click *Add* in the Triggered actions section.

► Select *Add one* for resulting action.

► In the Trigger column click the [...] button and select the *New Order Trigger* (under existing triggers expand the Check Order Handling Policy activity).
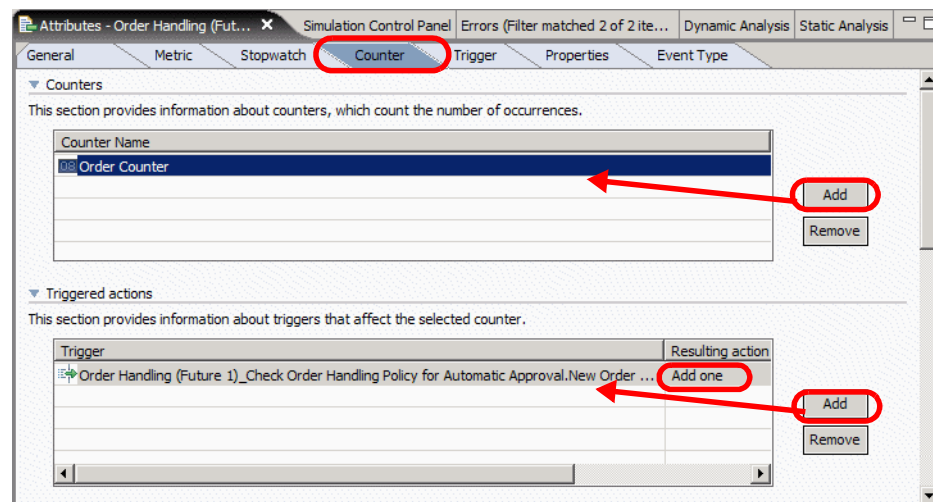
Figure 8-17   Create a counter (Order Counter)

We have finished the creation of the Order Counter.

Create the Shipped Order Counter 08

We define the Ship Order Counter as a counter that gets increased by one when the Ship Order to Customer activity was completed. Follow the same steps as for the Order Counter, but select the *Ship Order to Customer Trigger*.

The information provided for this counter will be used to build the Shipped Orders metric.

Create the Declined Order Counter

This counter is not used for the KPI, but may be nice to have in the Monitor. When building this counter, use a dynamic new trigger instead of a reusable trigger (Figure 8-18):

▸ Select State change as source category.
▸ Select *Activity State Completed* in the Cancel Order activity as source.
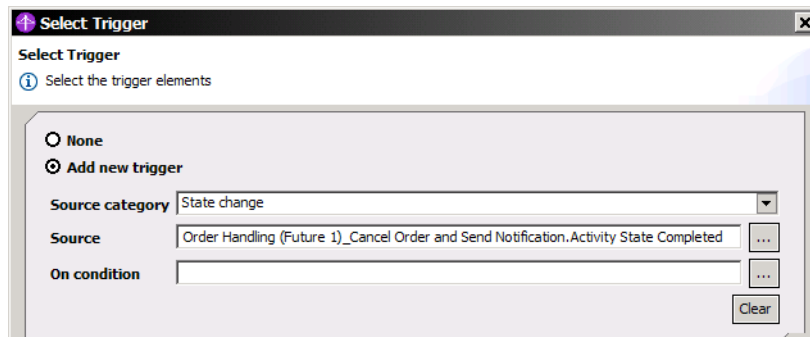
Figure 8-18   Creating a new trigger for a counter

We have finished the creation of the three counters.

Create the Total Orders metric

The Total Orders metric enables the aggregation of the single instances execution of the Order Counter and generation of the a total number of that value.

▸ Select the *KPIs and Aggregate Metrics* tab.

▸ Select *Metrics* in the left section. Click *Add* under the Metrics table.

▸ A new row is added in the table. Use the following attributes (Figure 8-19):

 – Name: `Total Orders`
 – Type: *Integer*
 – Aggregation function: *Total*
 – Aggregation source: *Order Counter*
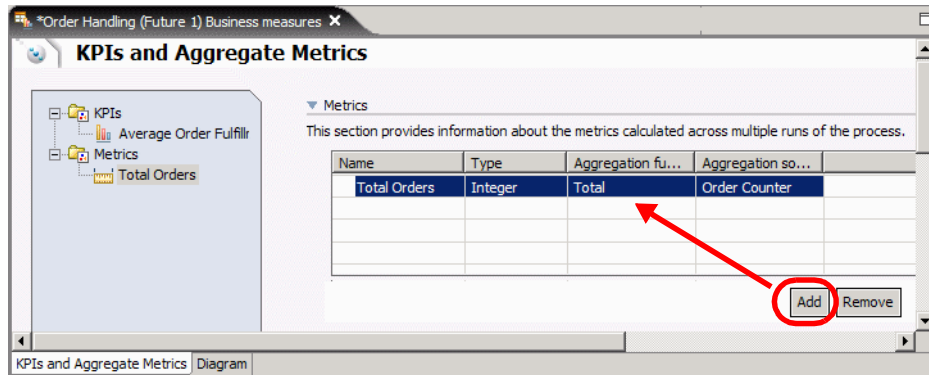
Figure 8-19   Aggregate Metric: Total Orders

Create the Shipped Orders metric

The Shipped Orders metric will enable the aggregation of the single instances execution of the Shipped Order Counter and generation of the a total number of that value.

Create the Shipped Orders metric in the same way using the Shipped Order Counter.

Create the Declined Orders metric

Create the Declined Orders metric in the same way using the Declined Order Counter.
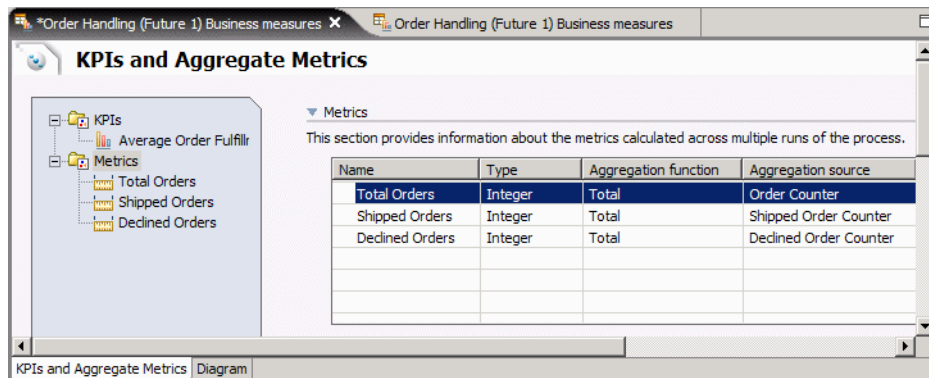
Figure 8-20   Aggregate Metric Ship Order Total

We have finished the creation of the metrics.

Create the 90 Percent of Orders Are Approved KPI

At this point we have defined the triggers, counters, and metrics that are the three components of this KPI. For this KPI we create a *User Defined calculation* through the expression builder, using two metrics to calculate the percentage of approved orders.

► Select the *KPIs and Aggregate Metrics* tab (Figure 8-21).

► Select *KPIs* in the tree. Click *Add* under the KPIs table and a row is added.

► We will create this KPI in two steps, first use these attributes):

  – Name: 90 Percent of Orders Are Approved
  – Type: *Double*
  – Aggregation function: *User Defined*
  – Use Target: leave selected
  – Target: 90
  – Lower target margin (%): 5
  – Upper target margin (%): 0
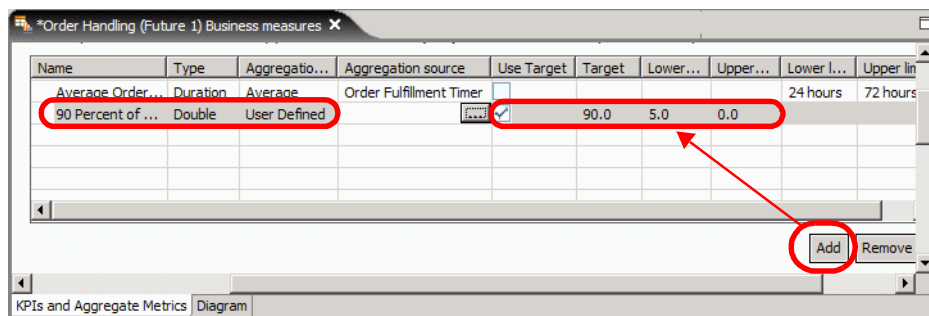
Figure 8-21   KPI 90 Percent of Orders Are Approved

► Second, click in the Aggregation source column and click the ⬚ button.

► The expression builder opens (Figure 8-22):

  – Select *Sub-expression* for the first term.
  – Select *x* as operator.
  – Select *Number* for the second term list and enter 100 as value.
  – Click *Apply*.
  – Click *Edit* in the left window to specify the first term sub-expression.
  – Select *Modeling artifact* for the first term and select *Shipped Orders.*
  – Select / as operator.
  – Select *Modeling artifact* for the second term and select *Total Orders*.
  – Click *Apply* and click *OK*.
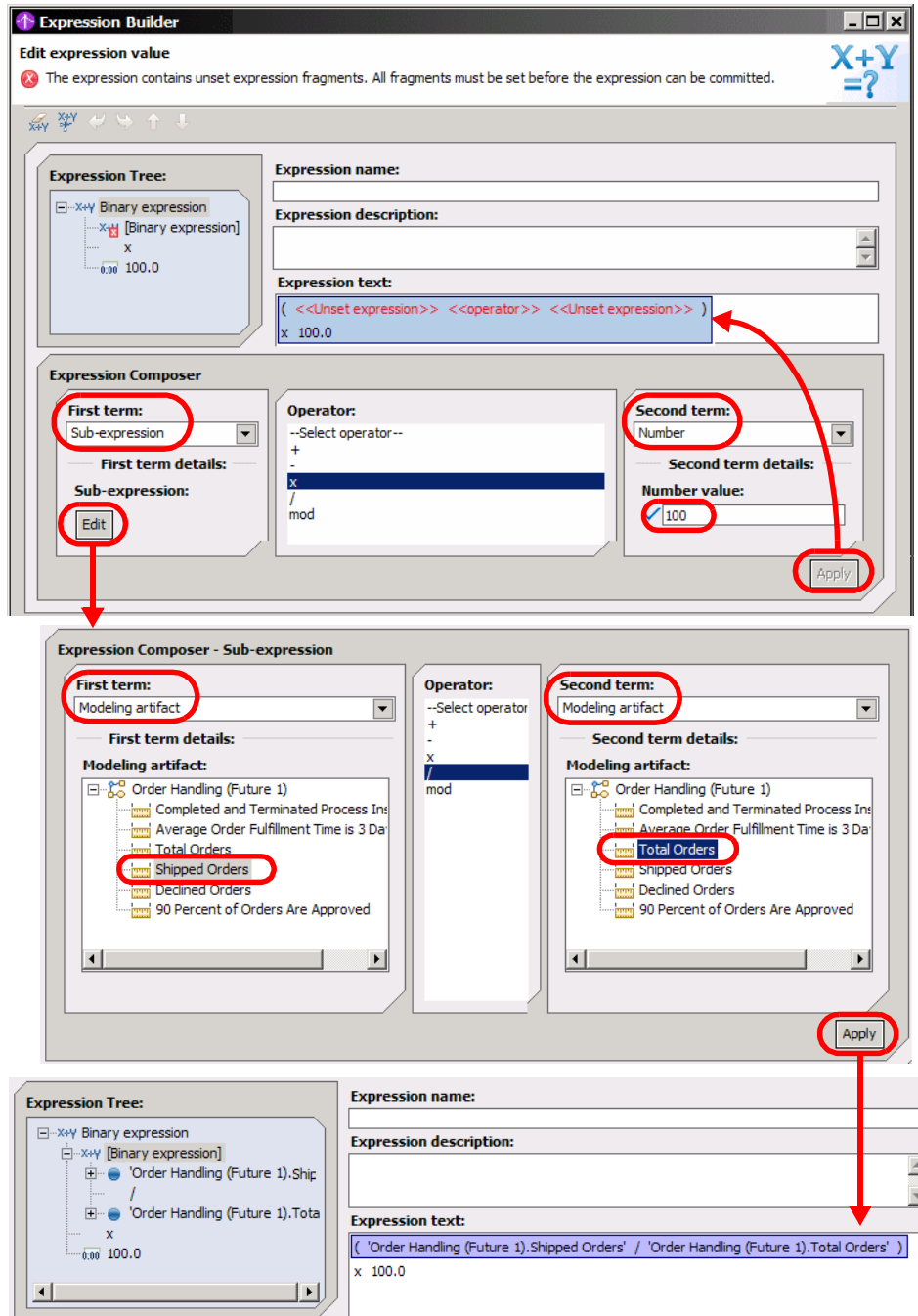
Figure 8-22   Expression builder

We have finished the creation of the 90 Percent of Orders Are Approved KPI (Figure 8-23).

Figure 8-23   90 Percent of Orders Are Approved KPI

Dimensional analysis

Dimensional analysis organizes historical data into levels of detail that you can drill down to extract significant information. To be able to perform dimensional analysis in WebSphere Business Monitor, you must specify some information in the business measures model for each instance metric, stopwatch, and counter.

A process can be described in terms of quantitative data, which takes on many values and participates in calculations, and in terms of dimensions, which are entry points for manipulating and analyzing the data in meaningful ways. Generally, any measure with non-numeric values is a level of a dimension, and you analyze other measures against dimensions.

For example, using dimensional analysis we can answer question like:

► Where are orders coming from? (Canada or USA)
► Are more orders declined from the USA than from Canada?
► Are we processing orders faster from cities in the USA?
► Are the orders from the USA for a higher total order price?

Basically we can analyze numerical data (counters, stopwatches) against other data that we capture (country, city, customer classification).

Adding a dimension

We can only perform dimensional analysis in the Monitor if we have at least one dimension specified in the business measures.

A dimension can be multi-level, such as a *location*, which breaks down into country and city.

To add the location as a dimension, perform these steps:

► Select the *Diagram* tab in the business measures model and click into an empty area to see the Attributes of the process.

► Click *Add* in the Dimensions section and overtype the default name with `Location Dimension` (Figure 8-24).
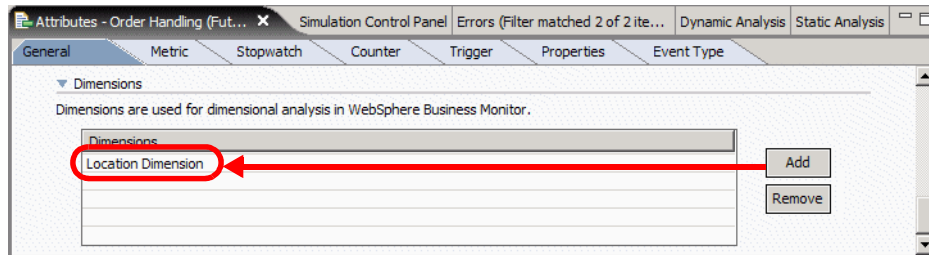
Figure 8-24   Creating a dimension

Adding metrics for dimensional analysis

The structure of the dimension is defined by using some of the business items of the process. These metrics are defined in the *Metrics* tab:

► Click *Add* twice to define two metrics and overtype the names with `Country` and `City`. Set the type to `String` and enter a description (Figure 8-25).

► Add another metric named `TotalPrice` of type `Double`. We will be able to analyze the order price against the location.
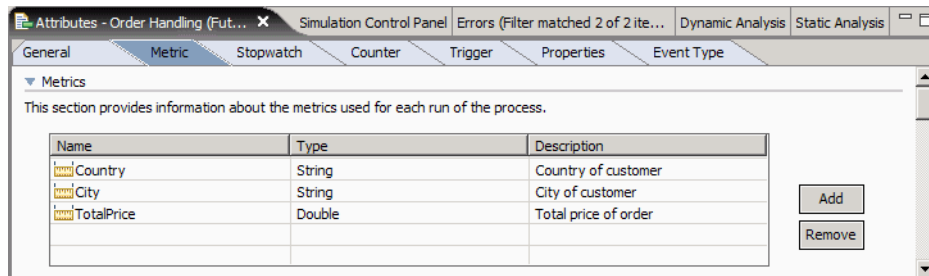
Figure 8-25   Creating metrics for dimensional analysis

Setting the value from a business item

Select each metric in turn and scroll down to the Value section:

► Click *Add* in the Value section to add a trigger.

► Click in the Trigger field and create a new trigger (Figure 8-26):

– Select *Add a new trigger*.
– Select *Inputs and outputs* as source category.
– Select the `Order Handling (Future 1)_Input Criterion` as source.
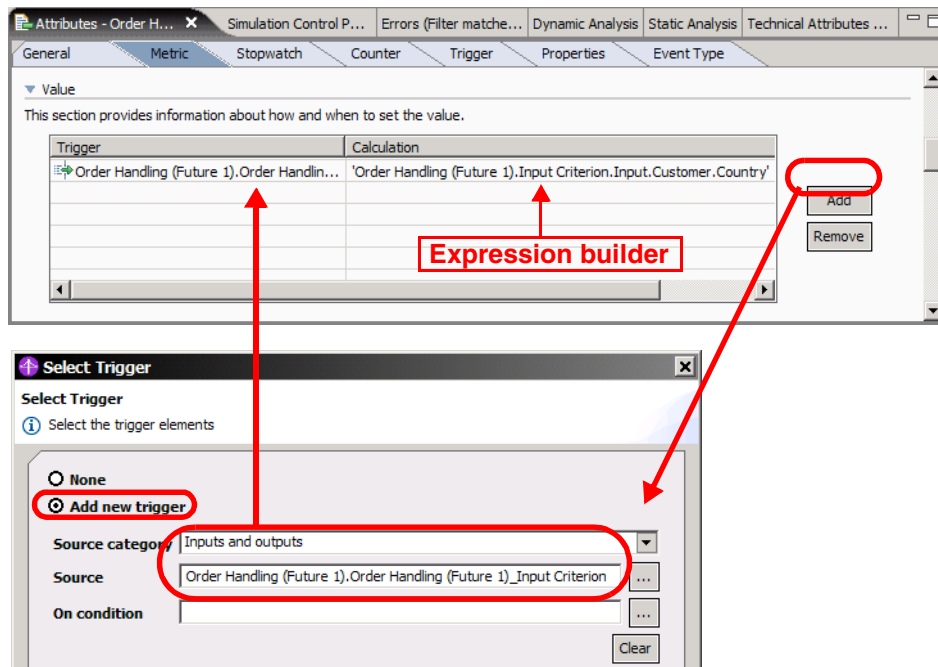
Figure 8-26   Using a trigger to set the value of a metric

► Click in the Calculation field (`Empty Calculation`). In the expression builder select *Modeling artifact* for the first term. Expand the process and select *Input Criterion* → *Input* → *Customer* → *Country*. Click *Apply* and *OK*.

► Set the values of the three metrics to:

```
Order Handling (Future 1).Input Criterion.Input.Customer.Country
Order Handling (Future 1).Input Criterion.Input.Customer.City
Order Handling (Future 1).Input Criterion.Input.TotalPrice
```

Setting the dimensional analysis properties

Select each metric in turn and scroll down to the Dimensional analysis and database schema settings (Figure 8-27):

► Select the `Country` metric and select *Aggregation group in dimensional analysis*, set the maximum length to `20`, select *Set as part of the dimension key*, select the *Location Dimension*, and set the aggregation group level to `0`.
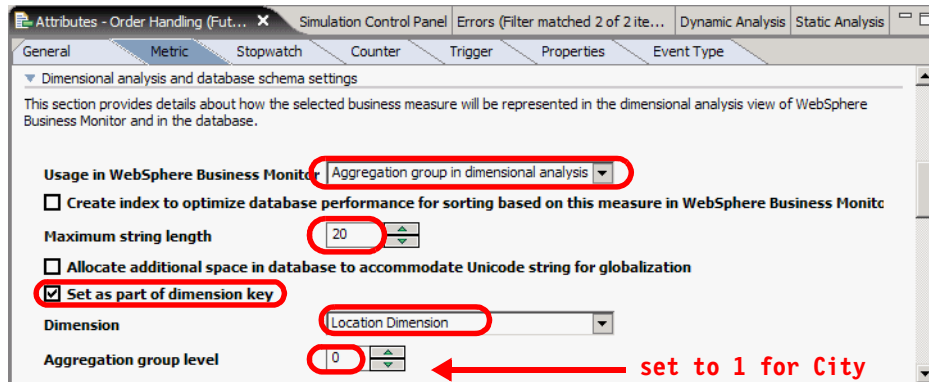
Figure 8-27   Set the country and city as dimension keys

► Select the `City` metric and specify the same values as for `Country`, but set the aggregation group level to `1`.

► Select the `TotalPrice` metric. Select *Quantitative data in dimensional analysis*. Click *Add* twice for aggregation measures, set the names to `Average of total price` and `Sum of total price`, and select *Average* and *Sum* as aggregation functions (Figure 8-28).

► Optionally select *Create index to optimize database performance*.
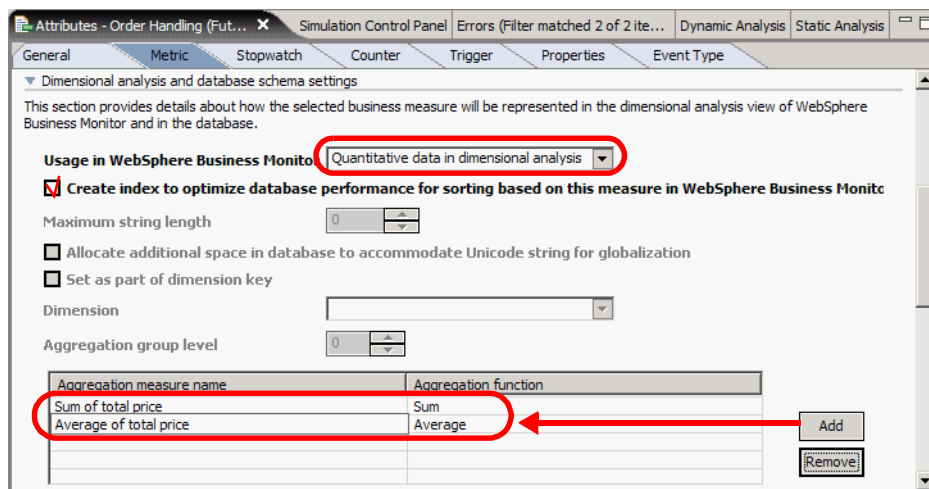
Figure 8-28   Preparing total price for dimensional analysis

Preparing the stopwatch and counters for dimensional analysis

To use the stopwatch and the counters that we defined for the KPIs also for dimensional analysis we have to change some of their parameters:

► Select the *Stopwatch* tab, then select the *Order Fulfillment Timer*. By default the usage in Monitor is set to *Active data about running process*.

► To use a stopwatch or counter for dimensional analysis we have to set the usage to *Quantitative data in dimensional analysis* and add an aggregation measure.

► Click *Add* for a measure and set the name as `Average Order Fulfillment Time` (Figure 8-29).
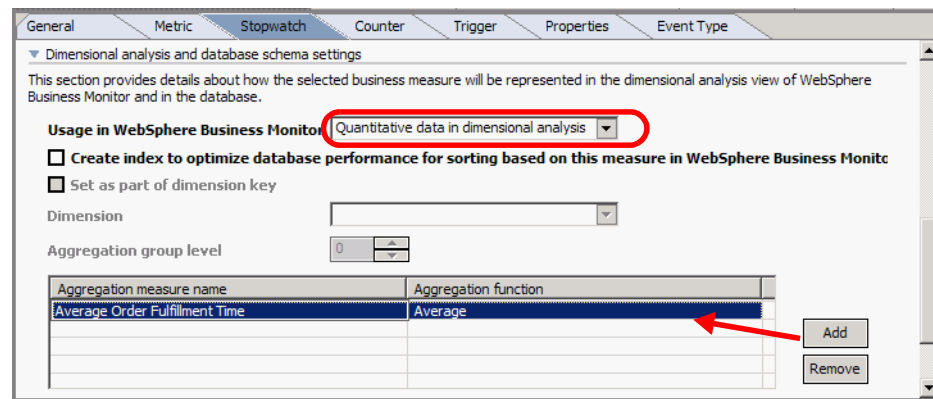
Figure 8-29   Preparing the stopwatch for dimensional analysis

► Select the *Counter* tab, then select each counter. Select *Quantitative data in dimensional analysis* and add an aggregation measure (`Total Orders`, `Total Shipped Orders`, `Total Declined Orders`) and select the *Sum* function (Figure 8-30).
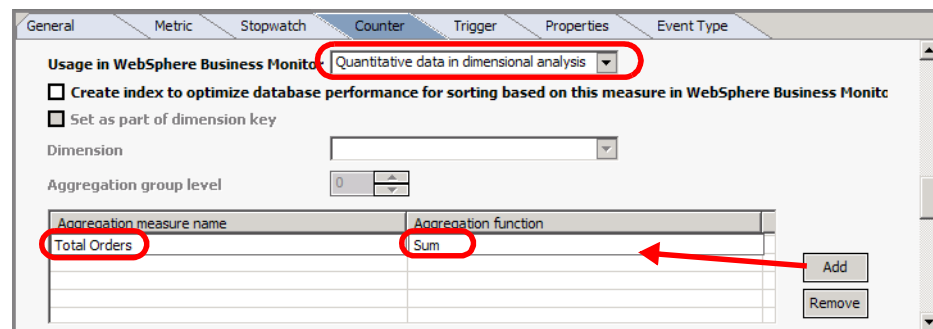
Figure 8-30   Preparing counters for dimensional analysis

Situation events

To make use of the Monitor event facility we define two situation events that may occur during execution of the business process:

▶ We want to be notified when the average order processing time is very high.

▶ We want to be notified when five orders are declined (this may not be very useful but illustrates the event trigger facility).

Defining event processing is a two-step process; first we define the events, then we define when the event is triggered.

Creating events

Event are Modeler objects defined in the Project Tree:

▶ Select *Events* and *New Event Definition*. In the dialog (Figure 8-31) enter a name, for example, `Late Average Order Shipped Event`.
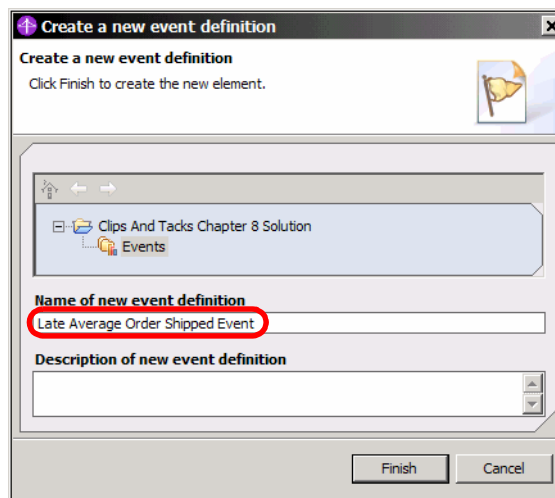
Figure 8-31   Creating an event

▶ The new event inherits four attributes from the `Business Situation` parent. Click *Add* to add an extra attribute named `Average Order Processing Time` with a type of `Duration` (Figure 8-32).

▶ You can add text on the *Documentation* tab:

`Event is triggered when the average order time exceeds a certain limit`
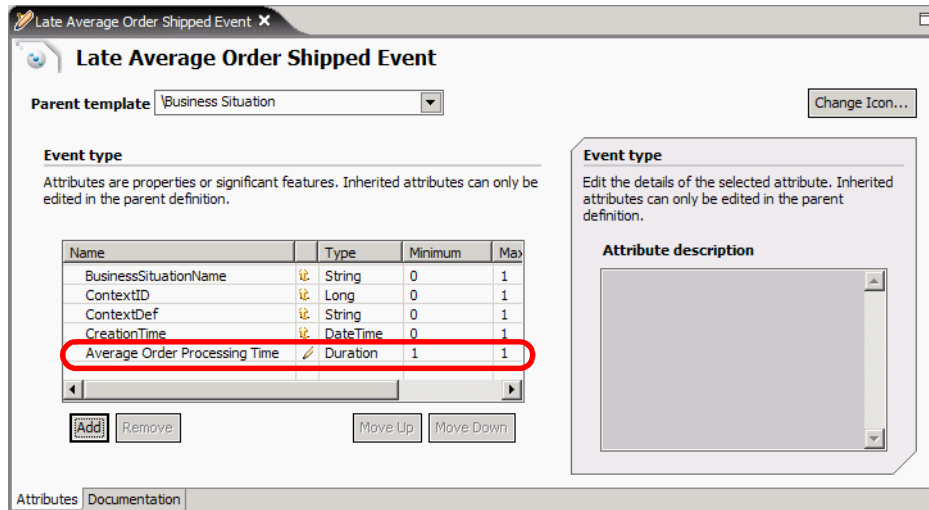
Figure 8-32   Adding an event attribute

▶ Create a second event named `Orders Declined Event` with no extra attribute.
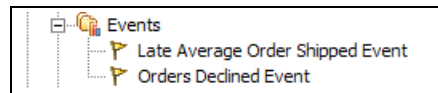
Triggering an event by a KPI

We have to specify when the events are triggered during the process. The Late Average Order Shipped Event is based on the first KPI:

▶ Open the Business Measures editor and select the *Average Order Fulfillment KPI* (Figure 8-33).

▶ Click *Add* in the Situation events section and select the *Late Average Order Shipped Event* when prompted.

▶ Select the event in the list and select *Send every time*.

▶ To set the condition click *Edit* and use the expression builder to build the expression:

```
'Average Order Fulfillment is 3 Days or Less' is greater than or equal to
                          Duration[3 days 1 hour]
```

– Select *Modeling artifact* as the first term and select *Average Order Fulfillment Time*.

– Select *is greater than or equal to*.

– Select *Duration* for the second term and select 3 days and 1 hour.

– Do not forget to click *Apply* before clicking *OK*.

Figure 8-33 Triggering an event by a KPI

► To set the values of the event attributes, double-click in the field, then click the ⋯ button and use the expression builder to calculate the value (Figure 8-34).
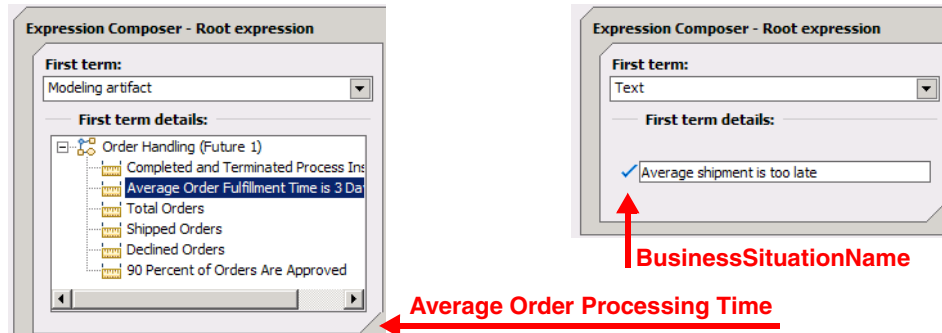
Figure 8-34   Calculating event attribute values

Triggering an event by a metric

We trigger the Orders Declined Event by the Declined Orders metric:

► Select the *Declined Orders* metric in the Business Measures editor (Figure 8-35).
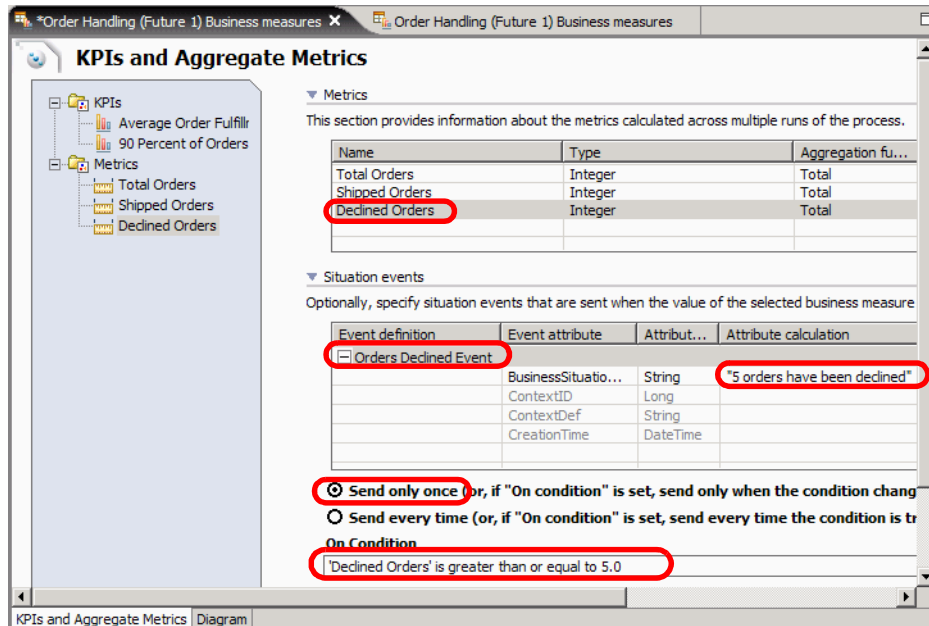
Figure 8-35   Triggering an event by a metric

► Click *Add* for Situation events and select the *Orders Declined Event*.

► Select *Send only once.* Click *Edit* and use the expression builder to build the expression:

`'Declined Orders' is greater than or equal to 5.0`

► Use the expression builder to set the `BusinessSituationName` attribute to the value `5 orders have been declined.`

Situation events for metrics, counters, and triggers

You can also trigger events from metrics, counters, and triggers defined in the Attributes view fro the process or for individual activities:

► Select the process (click in the diagram), then select the *Metric*, *Counter*, or *Trigger* tab in the Attributes view. There is a Situation events section in each tab where you can trigger a predefined event for a selected metric, counter, or trigger.

► Select an activity in the diagram and you can add a situation event for a trigger defined for that activity,

We do not set any additional triggered events for our scenario.

Export the process with the business measures model

You can export a project with the business measures model from WebSphere Business Modeler to import into WebSphere Integration Developer to implement the application and into WebSphere Business Monitor to monitor the business measures.

Before exporting, ensure that you are working in the WebSphere Process Server mode (select *Modeling → Mode → WebSphere Process Server*) and that your model has no errors. You will generally have to return to the process model to fix any errors and then select *Synchronize* to refresh the business measures model.

Set the valid from date

An important decision is when the process with the business measure should go into effect:

► Open the Business measures editor and select the *Diagram* tab:

► In the Attributes view *General* tab set the valid from date (Figure 8-36).
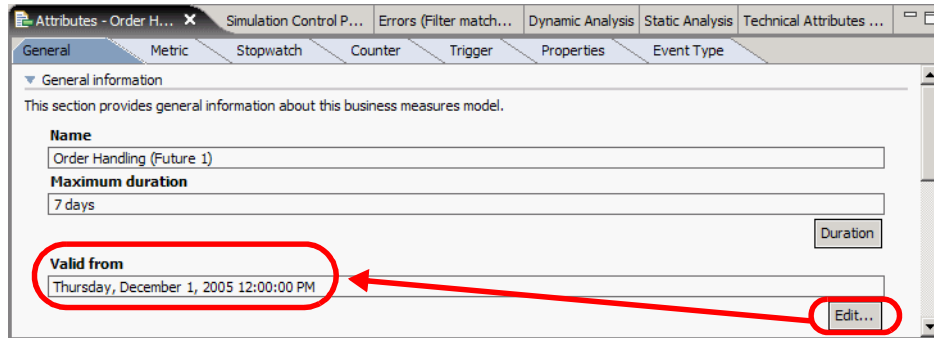
Figure 8-36   Setting the valid from date

The valid from date goes forward to Integration Developer and the Monitor and the two dates must not be changed, otherwise the Monitor disregards events.

Exporting the process with business measures

Compare this export with "Exporting the model for Integration Developer" on page 118.

To export the process for Integration Edition and Monitor, select either the project or the business measures and *Export* (context):

► Select the type of export you want. Select *WebSphere Business Monitor and development tool* (Figure 8-37). Click *Next*.
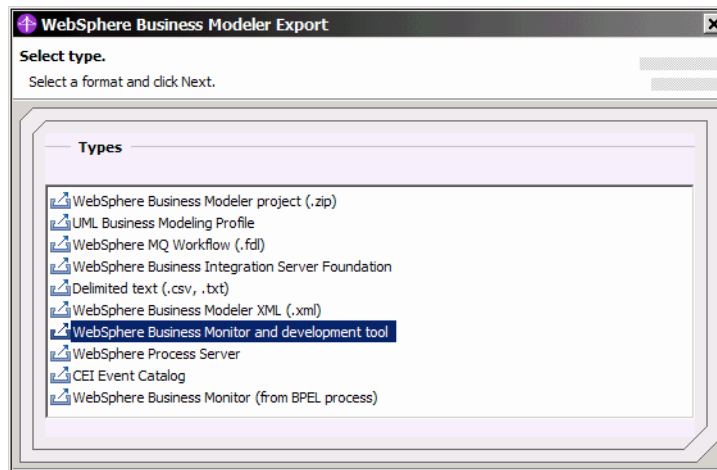
Figure 8-37   Export project with business measures

► Define the destination and source information export (Figure 8-38):

  – Target directory: `C:\SG247148\sampcode\model\export`
  – Monitor project version: `1`
  – Select *Export user defined event types to event catalog*
  – Click *Next*.
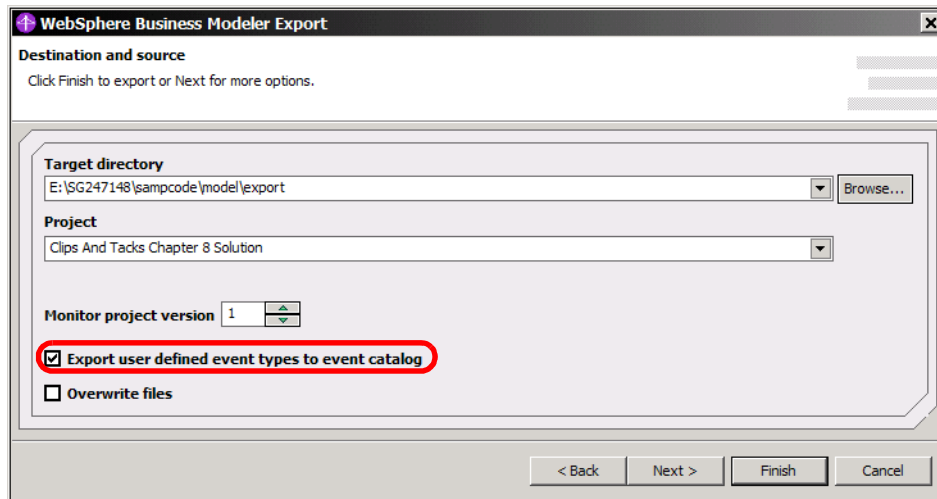
Figure 8-38   Export: Destination and source

► Define the destination and source information (Figure 8-39):

  – Select *Module project name* and enter: `ClipsAndTacks`
  – Select *Project Interchange name* and enter: `ClipsAndTacks`
  – WID Workspace location: select the same target directory
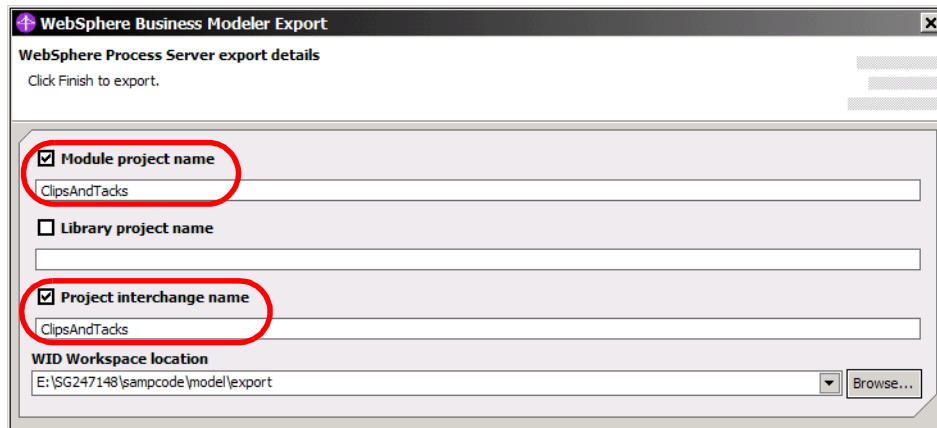
Figure 8-39   Export: WebSphere Process Server export details

▶ Click *Finish*.

> **Note:** If you only specified a module or library project, the export process creates a folder that contains all of the project files (BPEL, WSDL, ...). If you further specify a project interchange name, the export process packages the folder into a ZIP file.

Exported files

The target directory specified in Figure 8-38 contains two files:

▶ The **Monitor.zip** file, which contains the `model.xmi` file required for WebSphere Business Monitor. The `model.xmi` file contains the business measures that will be calculated in the Monitor.

▶ The **EventCatalog** folder with an `EventCatalog.xml` file that holds the definition of the situation events. This file is currently not used by the Monitor.

The WID Workspace location specified in Figure 8-39 contains the project interchange file, **ClipsAndTacks.zip**, that can be imported into WebSphere Integration Edition. The content of this file is described in "Exported file and content" on page 120.

We use the exported project interchange file in Chapter 9, "Developing the application using WebSphere Integration Developer" on page 187.

Summary

In this chapter we added the business measures to the Order Handling (Future 1) process so that we can monitor the business process using WebSphere Business Monitor.

We defined triggers, counters, a stopwatch, metrics, and the key performance indicators (KPI).

We also defined a location dimension, consisting of the customer's country and city, so that we can perform dimensional analysis of our numeric data (order counters and total price) against the customer location using the Monitor.

Part 3

Development and deployment

In this part we describe how the process model exported from the Modeler is implemented using WebSphere Integration Developer.

In Modeler we indicated how certain tasks should be implemented; now we actually perform the implementation using business rules, Java code, a Web service call, and the human task manager.

After testing the application in Integration Developer we deploy the application to Process Server.

Finally we experiment with advanced facilities, such as the Business Rules Manager and security for human tasks.

9

Developing the application using WebSphere Integration Developer

This chapter describes in detail how to use WebSphere Integration Developer to import the business process application exported from the Modeler and to complete the implementation.

WebSphere Business Modeler cannot be used to generate code for business rules, Java tasks, and to configure human tasks. These activities must be completed using a development tool.

We also have to implement the Web front-end that customers will use to submit their orders, and a database to store customer and order information.

We show how to configure and use the embeded WebSphere Process Server test environment to test the application.

Human tasks must be processed by the order manager and the shipper of ClipsAndTacks. We want to implement a nice user interface for these roles.

Note: the activities in this chapter are performed by the systems architect and possibly an application programmer.

Overview of the application implementation

To complete the application in WebSphere Integration Developer we proceed through the following steps:

► Setup a database to keep the customer orders
► Configure the WebSphere Process Server test environment
► Import the model
► Implement a business rule
► Implement a Java activity
► Configure the human tasks
► Test the process
► Add a stand-alone reference to invoke the process
► Implement the Web front-end to invoke the process
► Implement customized JSPs for the BPC Explorer
► Run the application in the test environment
► Implement an external Web service
► Implementing a human task application

Setup a database to keep the customer orders

The application requires a database to keep customer information, the products that can be ordered, and the orders of the customers.

The layout of the `CLIPTACK` database tables is shown in Figure 9-1.
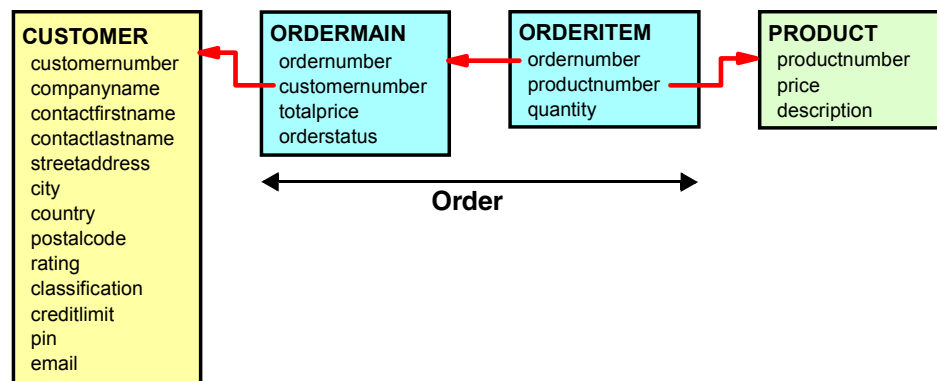
Figure 9-1   CLIPTACK database tables

Description of the database tables

The `CLIPTACK` database keeps track of the orders. Customers can submit orders through a Web front-end. During processing of the order the status of the order is updated.

CUSTOMER table

A customer is a company that orders products from ClipsAndTacks. Most of the information about a customer is self-explanatory, so we will only document some of the columns:

► **Rating**—The rating column hold a numeric value that represents the credit rating of that customer. We assume that there is an external service that we can query to get a rating for a company.

► **Classification**—The classification of a customer is either `REGULAR`, `SILVER`, or `GOLD`. This column is not used in the first implementation of the application. Later, in Chapter 15, "Implementing the Future 2 process using WebSphere Integration Developer" on page 375, we will implement business rules based on the classification.

► **Credit limit**—The credit limit column is used in the application to decide if a customer order is approved. We will use the rating obtained from the external service to adjust the credit limit.

► **Pin**—the password used by a customer to login to the Web front-end application. For our sample the pin is identical to the customer number.

The `CUSTOMER` table is initialized with a few customers:

```
Number Company               City, Country      Rating  CreditLimit
12345  ABC Finance Ltd.       Buffalo, USA          777  2000.00
11111  Auto Insurance Company Etobicoke, Canada     666  1300.00
22222  ABC University         Markham, Canada       555   500.00
33333  US Auto Financing      Chicago, USA          765  1999.00
44444  Insurance For You      Toronto, Canada       632  1444.00
```

PRODUCT table

The `PRODUCT` table is simple and contains five products:

```
Number  Name                               Price
RB-0001  All-In-One Printer                150.00
RB-0002  Manager Chair                      79.00
RB-0003  5 MP Digital Camera               499.00
RB-0004  Cordless Phone with Answering Machine  89.00
RB-0005  3-Drawer File Cabinet             214.00
```

ORDERMAIN table

The ORDERMAIN table records an order of a customer. An order number is generated, and the order status is tracked:

► **Status**—The order status starts with the value NEW, then it is either APPROVED or DECLINED. Approved orders then are changed to SHIPPED when processing is complete.

► **Total price**—The total price of an order is calculated from the items (products) that are ordered.

One sample order is initialized in the table:

```
Number   Customer   TotalPrice   Status
3001     12345         308.00    SHIPPED
```

ORDERITEM table

The ORDERITEM table contains the number of products that are ordered. The sample order contains two items:

```
Number   Product   Quantity
3001     RB-0001          1
3001     RB-0002          2
```

Populating the database

The Web front-end application (see "Implementing a Web front-end" on page 217) contains a servlet to defined the database tables and load the sample data. The data description language (DDL) statements and the sample data are retrieved from a properties file (clipstacks.properties).

Physical database

For our sample we use a Cloudscape™ database named CLIPTACK. We could have used DB2® as well. The access of the database is through a data source and there is no dependency in our application on the type of database.

For the location of the database we use an existing directory inside the Process Server:

```
C:\<WPS-HOME>\profiles\ProcSrv01\databases
```

In the Integration Developer test environment this location is:

```
C:\<WID-HOME>\pf\wps\databases
```

WPS-HOME and WID-HOME are the installation directories of the WebSphere Process Server and Integration Developer.

Configure the Process Server test environment

To access the `CLIPTACK` database from the server we define a data source. We can use either the administrative console or a JACL script.

In this section we describe how to use the administrative console. When we deploy the application to a real server we describe how to use a JACL script (see "Using a JACL script to define the data source" on page 253).

Start Integration Developer

Start the Integration Developer using *Start → All Programs → IBM WebSphere → Integration Developer V6.0.1 → WebSphere Integration Developer V6.0.1*.

We suggest to use a new workspace for all the work on the ClipsAndTacks application (Figure 9-2), for example:

```
C:\Documents and Settings\Administrator\IBM\wid6.0\workspace <== default
C:\Workspaces\WID601sg247148                                 <== our choice
```
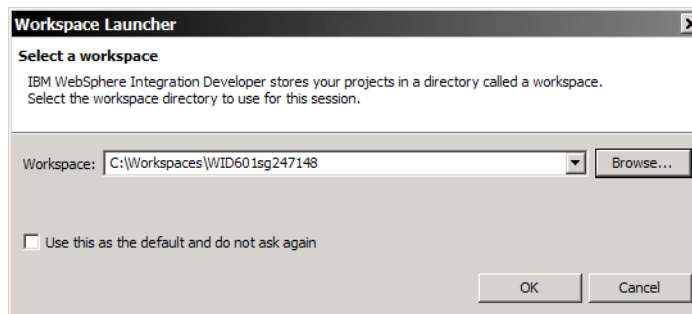

Figure 9-2   Workspace selection

Integration Developer opens with a Welcome page that you can close. The default perspective is the Business Integration perspective. For parts of our work we require the Web perspective.

Therefore, open the Web perspective (click the [icon] icon, select *Other*, then select *Show all*, then select *Web*, and click *OK*), and when prompted, enable the Web development capability (Figure 9-3).
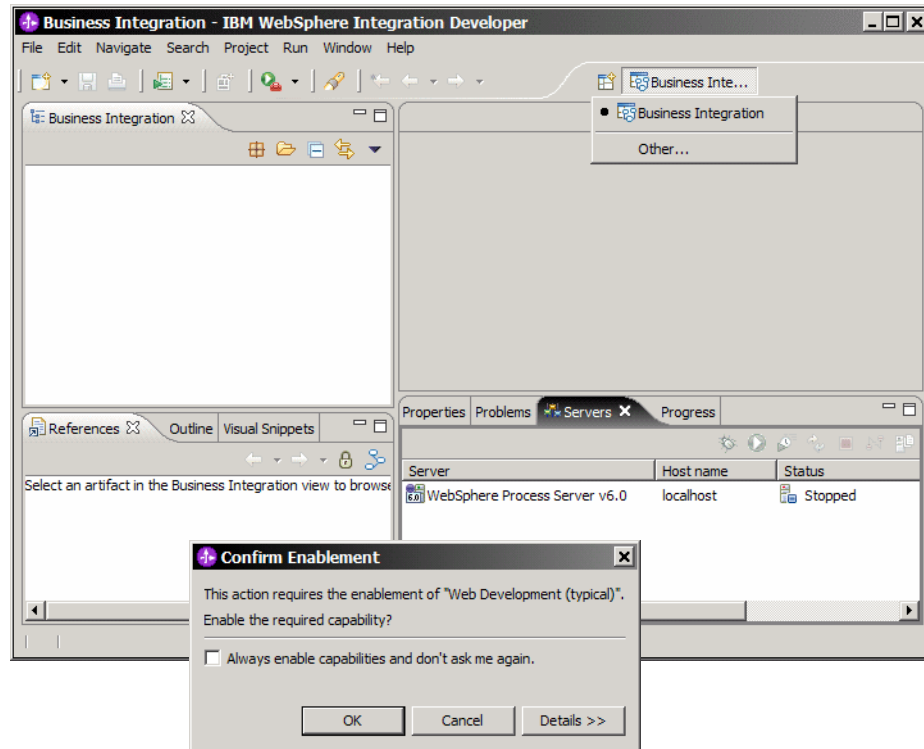
Figure 9-3   Opening the Web perspective

Start the process server and administrative console

To define the data source we start the process server and administrative console:

► In the Web perspective, Servers view, click ▶ to start the server (Figure 9-4). Then wait until the server is stared and the Console view shows the server console output.
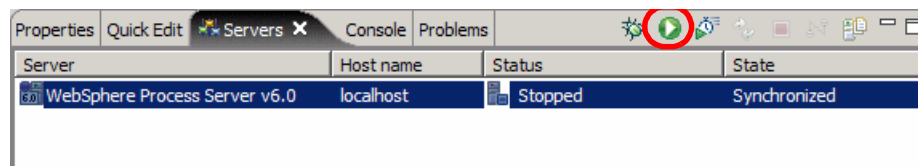
Figure 9-4   Start WebSphere Process Server v6.0

Be patient, starting the server takes about 5 minutes.

► Integration Developer communicates with the server through either RMI or SOAP. You can select the protocol by opening the server configuration—double-click *WebSphere Process Server v6.0* (Figure 9-5).
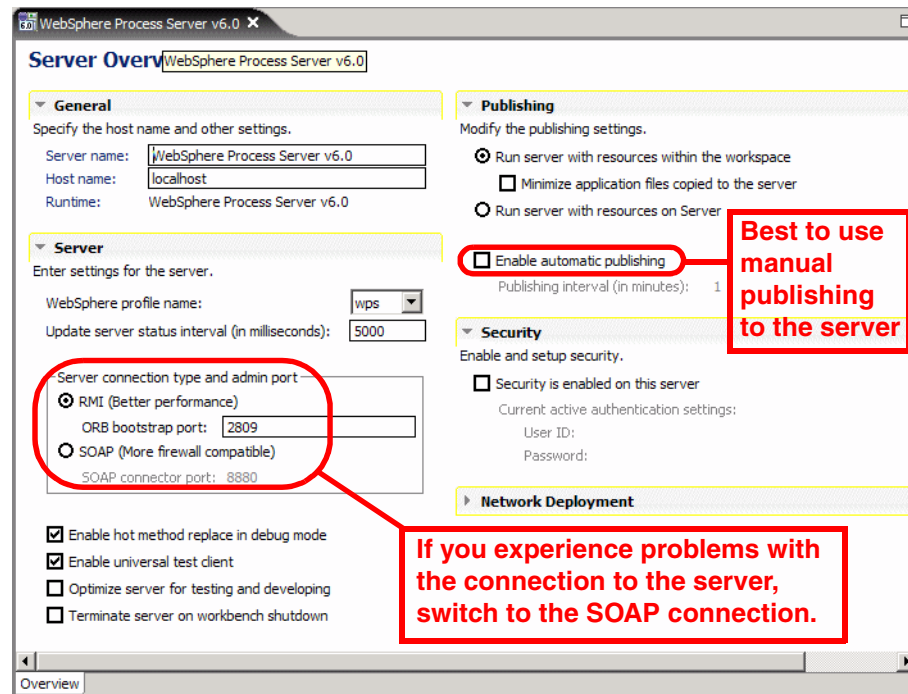
Figure 9-5   Process server configuration

► Start the administrative console by selecting the server and *Run administrative console* (context menu, Figure 9-6).
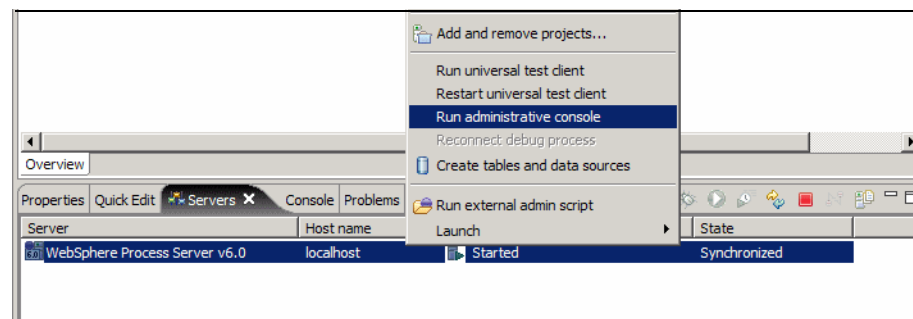
Figure 9-6   Start administrative console

► Enter any user ID or leave the field empty. A user ID is only required when security is enabled in the server. Click *Log in*.

Create a data source for the database

Data sources are attached to a JDBC™ provider. We will use the Cloudscape JDBC provider for our database:

► In the administrative console expand and select *Resources → JDBC Providers.*

► Select *Server* as scope, and click *Apply*. We define the data source for the `CLIPTACK` database at the server level (Figure 9-7).
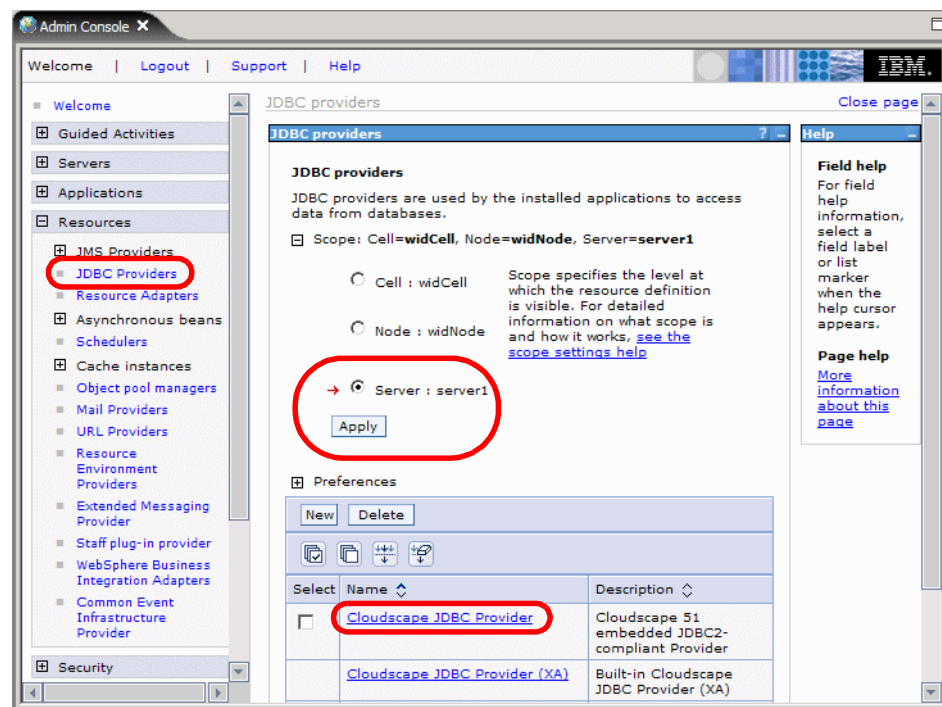
Figure 9-7   Change scope of JDBC Provider

► Click *Cloudscape JDBC Provider*.

► Click *Data sources* to define the data source (Figure 9-8).
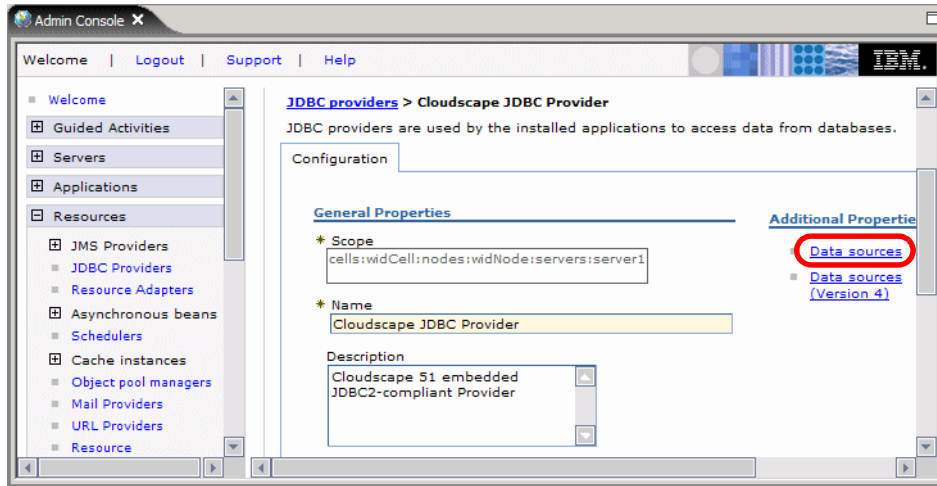
Figure 9-8   Cloudscape JDBC Provider

A number of Cloudscape data sources already exist (Figure 9-9). We create a
new data source for the `CLIPTACK` database by clicking *New*.
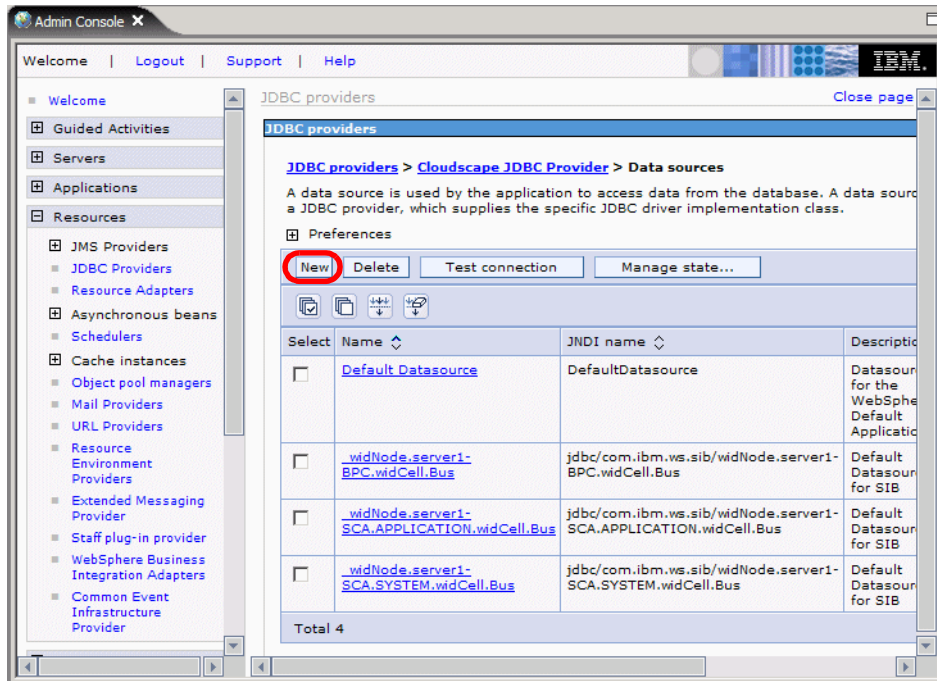
Figure 9-9   Cloudscape JDBC provider data sources

► Enter these values (Figure 9-10):

– Name: `ClipsAndTacks`

– JNDI name: `jdbc/cliptack`

– Deselect: *Use this Data Source for container managed persistence*

– Database name:

`${USER_INSTALL_ROOT}/databases/CLIPTACK`

The `${USER_INSTALL_ROOT}` variable points to the server profile where we want to define the database. In the test environment this location is:

`C:\<WID-HOME>\pf\wps`

► Click *Apply*.
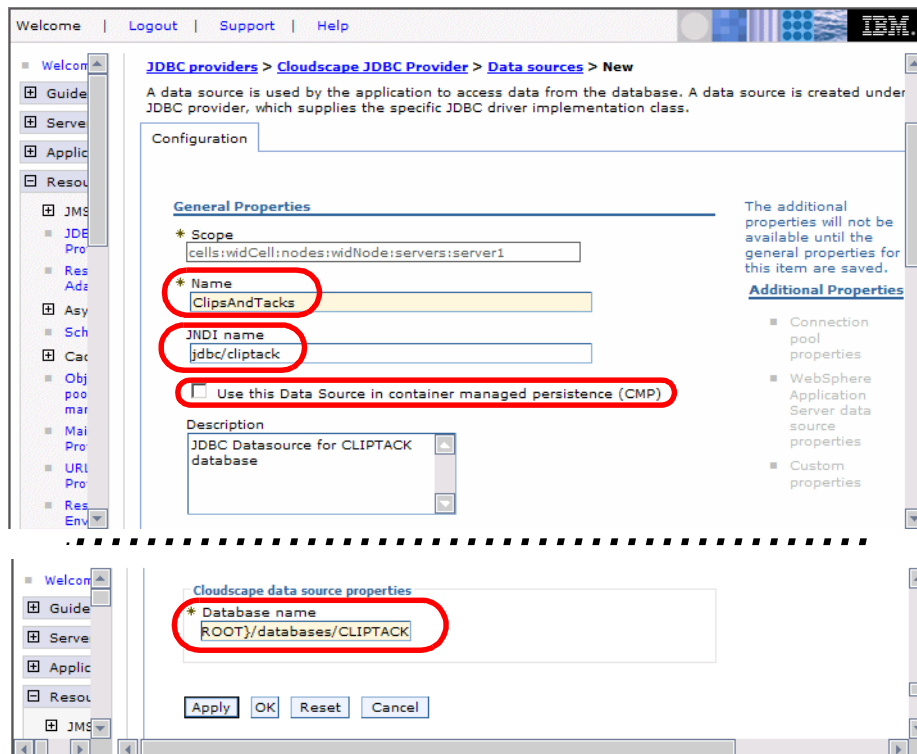
Figure 9-10   Create the data source for the CLIPTACK database

► Select *Custom properties* under Additional Properties.

► Select the *createDatabase* property and enter a value of `create` (Figure 9-11). This specification will create the database on the first access.
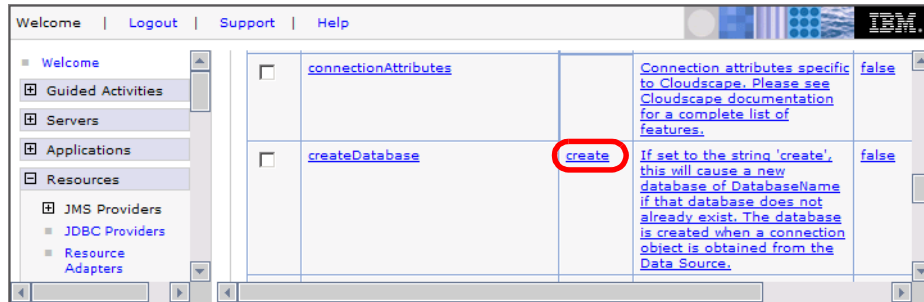
Figure 9-11   Automatic database creation

▶   Save the configuration (click *Save*).

▶   In the data source list (of the Cloudscape JDBC provider), select the *ClipsAndTacks* data source and click *Test connection* (Figure 9-12).
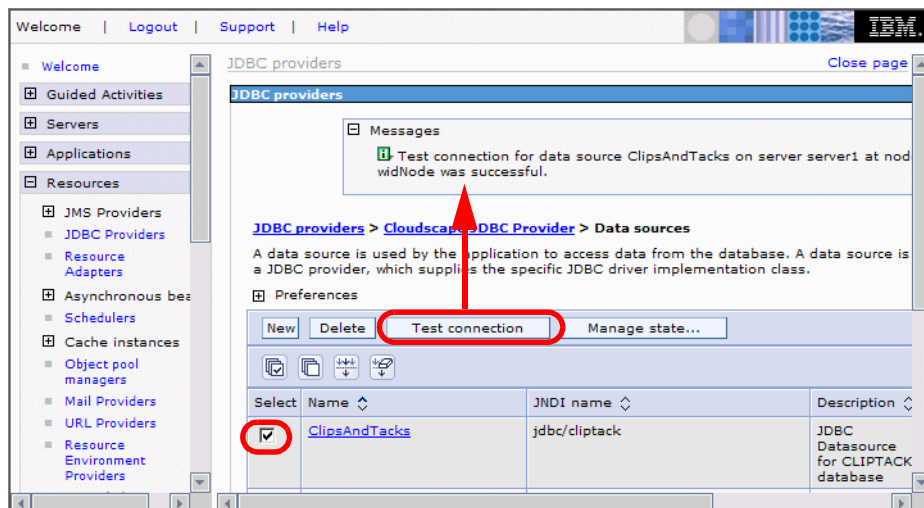
Figure 9-12   Testing the data source

▶   You should receive a message indicating success. You can verify that the database has been created by exploring the directory:

    C:\<WID-HOME>\pf\wps\databases\CLIPTACK

▶   Click *Logout* to close the administrative console.

At this time the database is created but does not contain any tables or sample data. We will initialize the database later using the Web front-end application (see "Initialize the database" on page 223).

Import the model

In this section we import the model that was created in WebSphere Business Modeler and exported as a project interchange file.

We have two models that we can use:

▶ The exported model from Chapter 6, "Modeling the Future 1 business process" on page 83 that does not contain any business measures. Choose this model if you do not have access to WebSphere Business Monitor.

We provide the project interchange file in:

        SG247148\sampcode\model\export\noMonitor\ClipsAndTacks.zip

▶ The exported model from Chapter 8, "Creating the business measures model" on page 145 that contains the business measures. Choose this model if you have WebSphere Business Monitor installed and want to measure the application.

We provide the project interchange file in:

        SG247148\sampcode\model\export\withMonitor\ClipsAndTacks.zip

Import the project interchange file

To import the model and generate the processes in Integration Developer follow these steps:

▶ Select *File → Import*.

▶ Select *Project Interchange* and click *Next*.

▶ Click *Browse* and locate the ClipsAndTacks.zip file. Select the ClipsAndTacks application and click *Finish* (Figure 9-13).
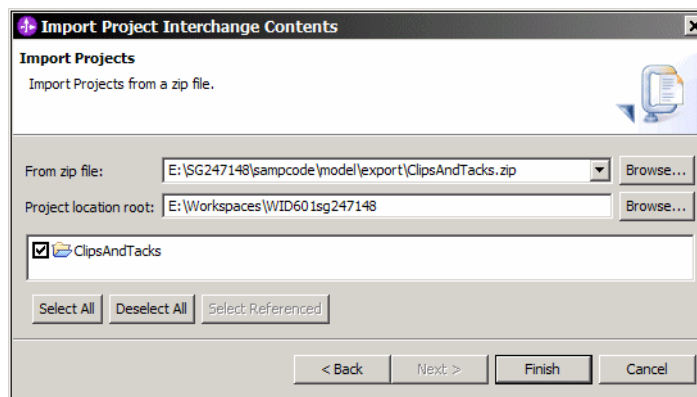
Figure 9-13   Import project interchange file

Be patient, from the model a number of projects are generated and compiled. Watch the progress indicator at the bottom right:
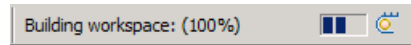
Generated projects from the model

The list of projects that are generated from the model is shown in Figure 9-14.
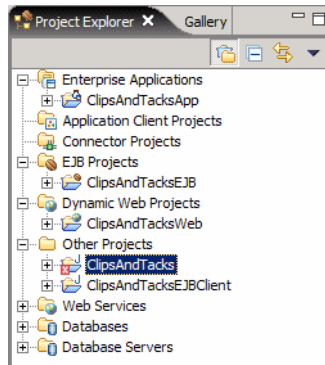
Figure 9-14   projects generated from model import

► `ClipsAndTacks`—the business process, a Java project.

► `ClipsAndTacksApp`—the enterprise application

► `ClipsAndTacksEJB`—an EJB™ project with a number of session beans and one message-driven bean

► `ClipsAndTacksWeb`—a Web project (empty)

► `ClipsAndTacksEJBClient`—the client projects for the EJB project.

Our work will be concentrated on the `ClipsAndTacks` project. Note that it shows an error, which you can also see in the Problems view (Figure 9-15):

```
The definition of operation InputCriterion must have at least one enabled
selection criterion -- either a criteria-based destination or a default
destination.
```
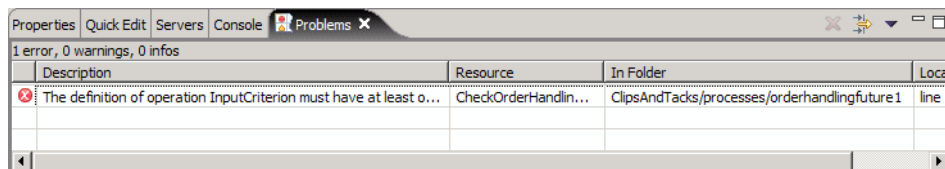

Figure 9-15   Error after import

When you expand the `ClipsAndTacks` project in the Business Integration perspective, you can see that the error comes from the rule group `CheckOrderHandlingPolicyforAutomaticApproval`.

The business rule for automatic approval was an annotation in the Modeler; now we have to implement a real business rule.

Business Integration view

In the Business Integration view we can see the details of the business process (Figure 9-16).
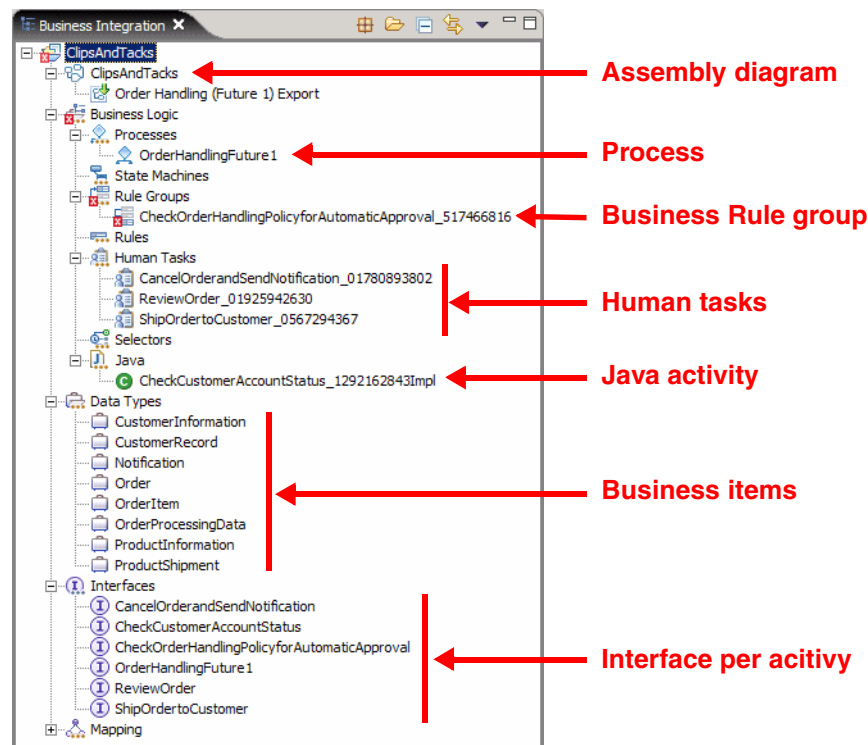
Figure 9-16   Business Integration view

Notice that all the activites have a suffix appended to the name and that blanks and parenthesis have been eliminated. The activity type that we seelcted in the Modeler has been used to generate different artifacts in the development tool.

Assembly diagram

The assembly diagram is shown in Figure 9-17. The diagram does not imply a sequence of operation of the five activities on the right; it only shows what activities are invoked by the process. Notice the different icons for the activities: We have one business rule, one Java implementation, and three human tasks. When you select an activity, some details are visible in the Properties view.
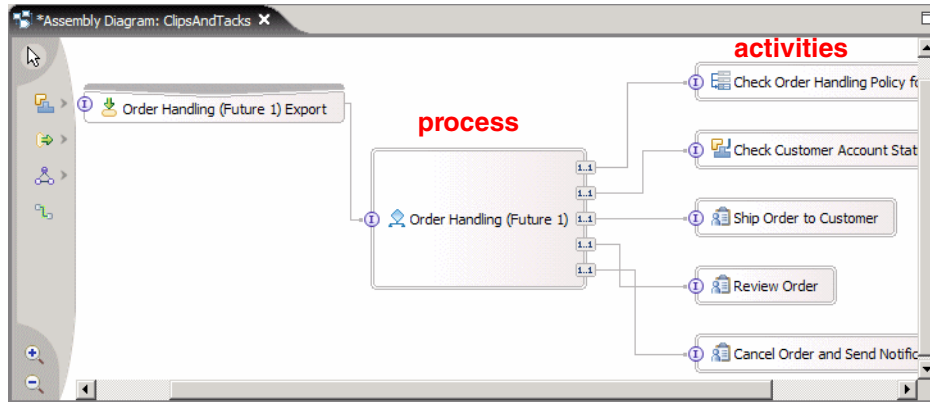
Figure 9-17   Assembly diagram

Process diagram

The process diagram for `OrderHandlingFuture1` is shown in Figure 9-18.
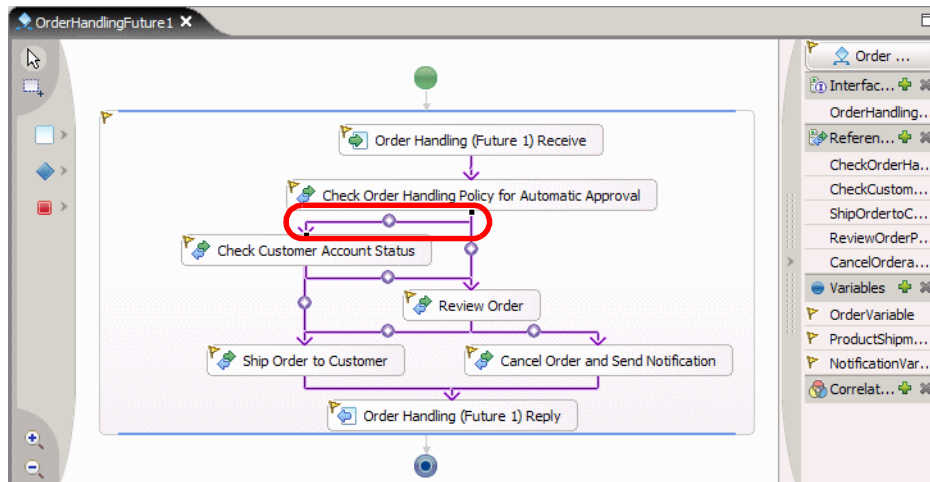
Figure 9-18   Process diagram

When you select one of the links (indicated by the square dots in Figure 9-18) you can see the corresponding code in the Properties view, Details tab:

```
return ((OrderVariable.getDataObject("ProcessingPreference")
                  .getBoolean ("automaticApproval")) == (true));
```

Keeping processes once they are finished

The Properties view shows a number of details about the process. For example, in the Details tab (Figure 9-19) you can see this information:

▶ Process is long running.

▶ Automatically delete the process after completion is selected by default.

   You may deselect this flag so that finished processes are still visible, but only use this in the test environment and not for production. Save the process afterwards.

▶ The valid from date is December 1, 2005, 20:00 (8pm). This is the GMT equivalent of the date December 1, 2005, 12 pm (Pacific time zone) that we set in the Modeler.
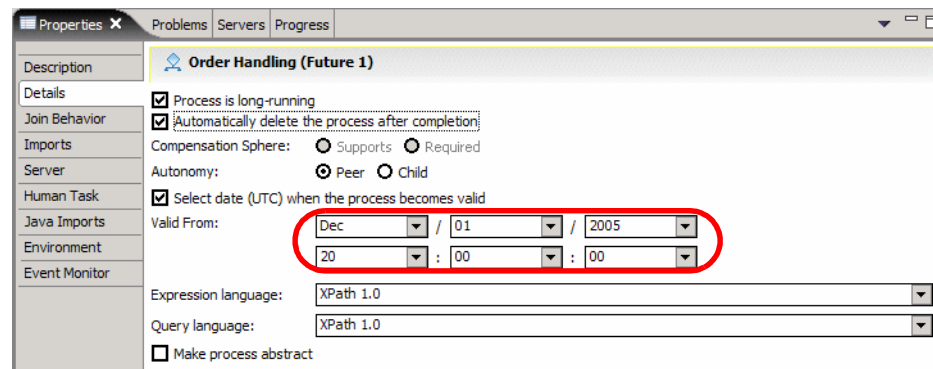
Figure 9-19   Process properties

Important: Do not change the valid from date; it must match what is exported to the Business Monitor, otherwise no events will be sent to the Monitor.

Business objects (data types)

The data types are imported as business objects from the Modeler as a `Businessitems.xsd` file under `ClipsAndTacks/businessitems`.

You can open a data type to see its definition in a diagram (Figure 9-20).
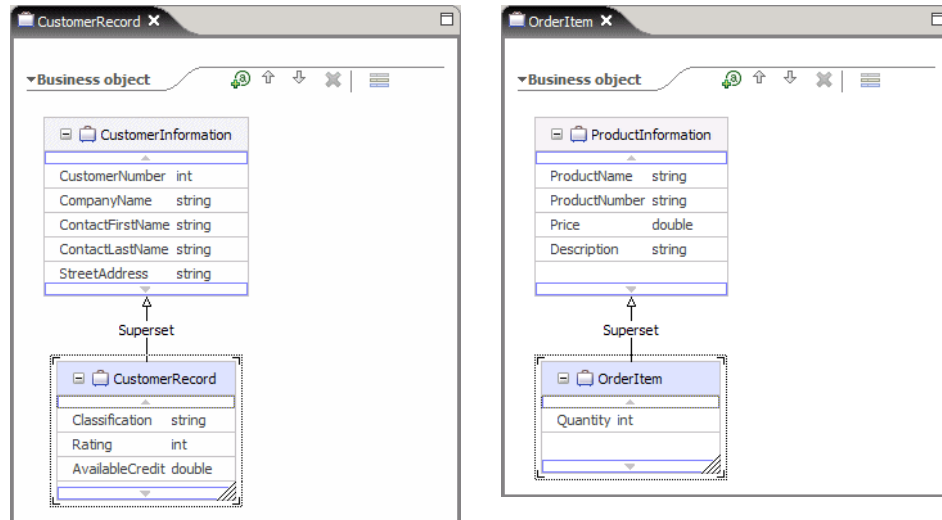
Figure 9-20   Data types

Physical resources

To see all the files that make up the business process, open the Physical
Resources view (*Window → Show View → Physical Resources*):

► The `businessitems` folder with the `Businessitems.xsd` file.

► The `gen/src/processes/orderhandlingfuture1/brg` folder with the generated
Java code `CheckOrderHandlingforAutomaticApproval`.

► The `processes/orderhandlingfuture1` folder with:

– A business rule group
`brg/CheckOrderHandlingPolicyforAutomaticApproval`.

– A `wsdl` file for each activity, which corresponds to each interface in the
Business Integration view.

– A `component` file with references to the interface and the implementation.

– A `tel` file and a _tel.mon file for each human task.

– The `OrderHandlingFuture1.bpel` file. This is the business process itself;
opening this file opens the process diagram.

► The `sca.module` with `ClipsAndTacks`, which is the assembly diagram.

The next tasks are to finish the implementation of the process.

Implementing a business rule

The business rule for automatic approval is an annotation in the Modeler:

```
Order Handling Policy
==============================================
Default: Orfers are reviewed by the system
for automatic approval
==============================================
If the toal price of an order is less the $750
then the order can be automatically approved
without review.
==============================================
```

For the implementation we have to implement the business rule:

▶ Open the business rule group `Check Order Handling Policy for Automatic Approval`, either from the assembly diagram, or by expanding the project in the Business Integration view (expand *ClipsAndTacks → Business Logic → Rule Groups → CheckOrderHandlingPolicyforAutomaticApproval*).

▶ Select *InputCriterion* (Figure 9-21).
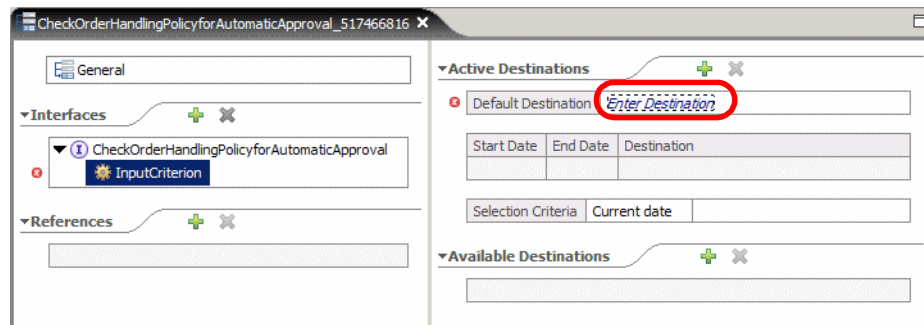
Figure 9-21   Rule group destination

▶ Click *Enter Destination* and select *New RuleSet* from the pull-down menu. In the New Rule Set dialog:
  – The folder is set to `processes/orderhandlingfuture1`.
  – Change the name to `AutomaticApprovalFuture1`.
  – Click *Finish*.

The rule set for `InputCriterion` opens (Figure 9-22):

▶ Add two action rules by clicking the *Add Action Rule* icon 🌼 .

▶ Add one if-then rule by clicking the *Add If-Then Rule* icon 📋 .

▶ In the first rule, click on *Action*, then select *Output*, *=*, *Input*.

► In the second rule, click on *Action*, then select *Output →
ProcessingPreference → automaticApproval, =, false*.

► In the third rule, click *Action*, then select *Output → ProcessingPreference →
automaticApproval, =, true*. Then click *Condition*, select *Input → TotalPrice,
<=, Number → 750.00*.
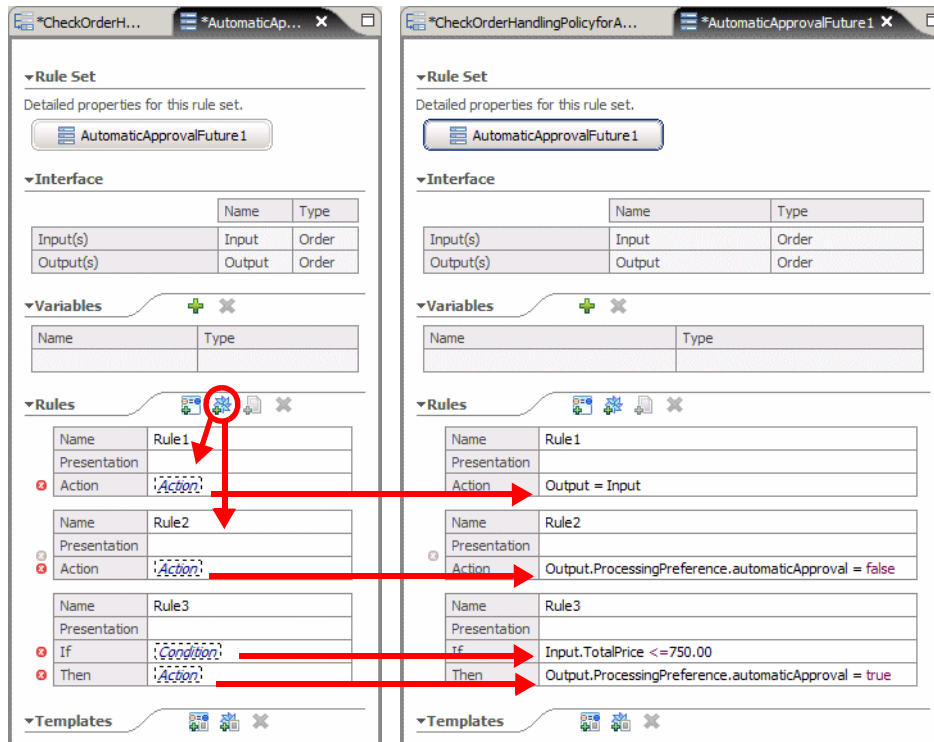
Figure 9-22   Defining business rules

► Save the new rule and the modified rule group. The error disappears from the
Problems view, but seems to still show up in the Business Integration view.

Implementing a Java activity

The activity `Check Customer Account Status` has a Java implementation. Open
the activity from the assembly diagram or from the Business Integration view
(expand *ClipsAndTacks → Business Logic → Java*).

The Java code `CheckCustomerAccountStatus_1292162843Impl` opens. There is
one method that requires code:

```
public DataObject InputCriterion(DataObject Input)  {
    //TODO Needs to be implemented.
    return null;
}
```

The purpose of this activity is to verify the available credit of the customer so that a decision can be made for approval of the order. The available credit is based on the *rating* of the customer by some credit rating service.

For our scenario we use a simple approach:

► For the first implementation we use a JavaBean to randomly calculate a new rating, and then adjust the available credit.

► Later, in "Implementing an external Web service" on page 234 we use an external Web service to provide a new rating, and then adjust the available credit.

The logic for this activity is provided in a JavaBean with the name `CreditRating`. Complete the `InputCriterion` method with this code:

```
public DataObject InputCriterion(DataObject Input)  {
    System.out.println("Check Customer Account Status Invoked");

    // create CreditRating bean
    com.clipstacks.credit.CreditRating creditRating =
                            new com.clipstacks.credit.CreditRating();

    // call CreditRating bean to update the BO
    DataObject orderOut = creditRating.calculateCreditRating(Input);

    return orderOut;
}
```

Save the code. Ignore the error for now; we have to implement the `CreditRating` JavaBean.

Implementing the credit check

To implement the credit check using the `CreditRating` JavaBean follow these steps:

► Create a Java package named `com.clipstacks.credit` in the `ClipsAndTacks` project (select *File → New → Other → Java → Package*).

► Import the `CreditRating` class from the sample code into the package:

```
SG247148\sampcode\wid\creditRating\CreditRating.java
```

Open the `CreditRating` code (use the Physical Resources view or the Web perspective) and study the code:

► A boolean switch, `useWebservice`, determines if an external Web service is invoked. For now, this value is `false`.

► A `creditAdjustmentFactor` (0.1) is used for the calculation of the new available credit of the customer.

► The `calculateCreditRating` method receives the input business object as parameter and returns the output business object.

► The customer information is retrieved from the input business object.

► For now let us not investigate how the external Web service is called.

► A new rating is calculated as a random number between 500 and 800:

```
newRating = 650 + (new java.util.Random()).nextInt()%150;
```

► The available credit is calculated as:

```
availCredit = availCredit +(newRating-oldRating)*creditAdjustmentFactor;
```

This calculation adjusts the available credit up or down depending on the change in the rating.

► The output business object is created using the factory and copied from the input business object using the copy service.

► The customer rating and available credit are updated with the new values.

► The output business object is returned.

We will replace the random rating with an external Web service later.

Configuring the human tasks

We have three human tasks that have to be configured for processing. As a first implementation we define that everybody is allowed to process the human tasks and we use the BPC Explorer that comes with the process server to work on the human tasks.

To configure the human tasks perform these steps for each of the three human tasks (`Review Order`, `Ship Order to Customer`, and `Cancel Order and Send Notification`):

► Open the human task from the assembly diagram or from the Business Integration view (expand *ClipsAndTacks* → *Business Logic* → *Human Tasks*).

► Select *Potential Owner* under Receiver Settings. In the Properties view, select *Everybody* as verb (Figure 9-23).
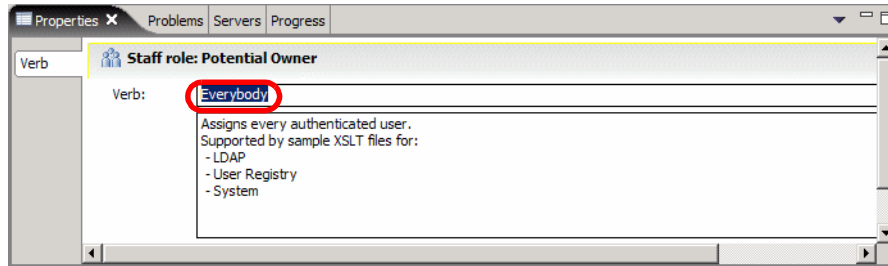
Figure 9-23   Human tasks: Change Role Members to Everybody

► For Client settings click the *BPC Explorer* icon 🖧 to setup BPC Explorer for human tasks processing (Figure 9-24).

  Notice the Client Settings in the Properties view. We will defined tailored JSPs for human tasks later in "Implementing customized JSPs for the BPC Explorer" on page 225.
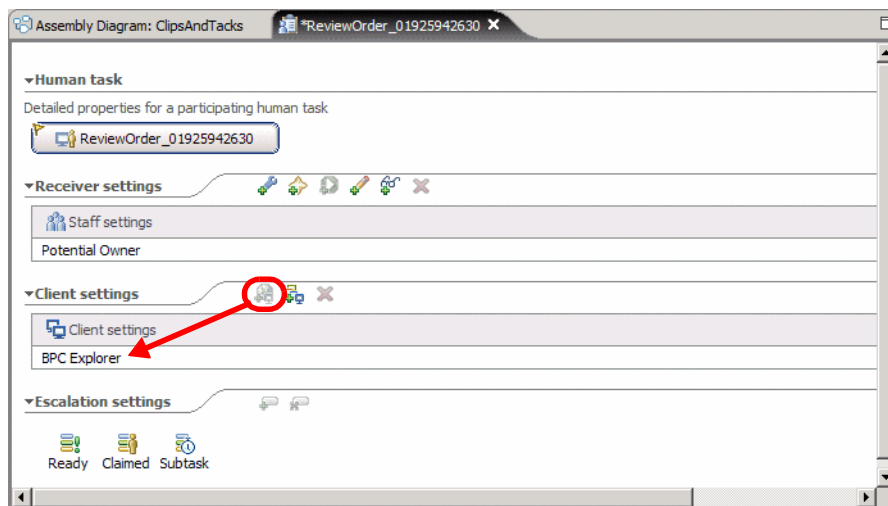

Figure 9-24   Human tasks configuration

Do not forget to configure all three human tasks.

Testing the application

To test the application we add it to the server and use the built-in process test facility.

Deployment of the application for testing

Before we can test the process we have to deploy the application to the test server:

► In the Servers view, select the WebSphere Process Server v6.0 and *Add and remove projects* (context menu).

► Select the `ClipsAndTacksApp` project and click *Add>* (Figure 9-25). Click *Finish* and the application is deployed. Watch the progress indicator at the bottom right.
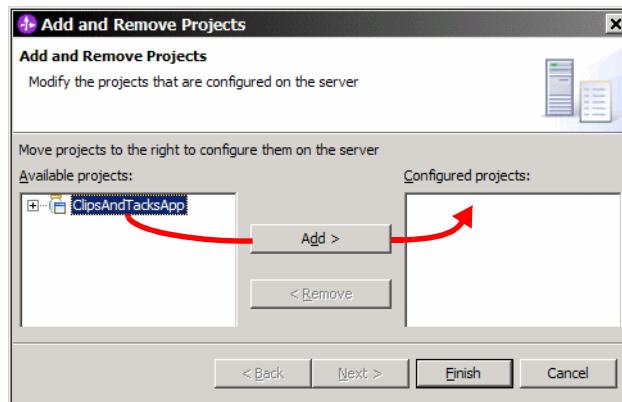
Figure 9-25   Add project to server for testing

Tip: If you receive an error when adding the project to the server, see "Deployment failure: duplicate valid from date" on page 247 for a possible source of this problem.

Using the built-in process test facility

To test the ClipsAndTacks module follow these steps:

► Select the `ClipsAndTacks` project and *Test → Test Module*. The `ClipsAndTacks_Test` panel opens (Figure 9-26).

► Under Detailed Properties, change the Component to `Order Handling` (`Future 1`). By default it is set to the first activity in alphabetical order.

► Enter values for the `Customer` data.

► To enter order items, select `OrderItems` and *Add Element* (context menu).
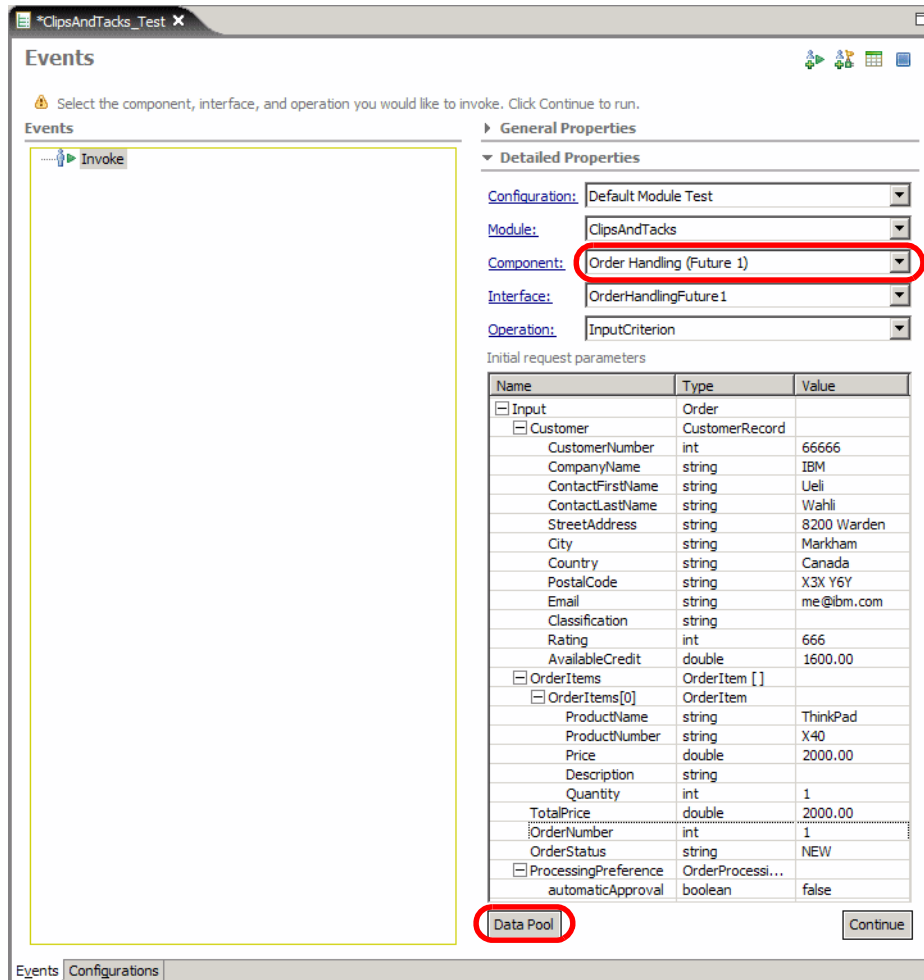
Figure 9-26   Testing: Input data

This is quite some work. To reduce the effort for future tests, we can save the values in a data pool:

► Select *Input* (the first line) and *Add Value to Pool* (context menu). In the Value Name dialog, enter a suitable name, for example, `CustomerIBM` and click *OK*.

► Click *Continue*.

► Select the deployment location as *WebSphere Process Server v6.0* and click *Finish* (Figure 9-27).
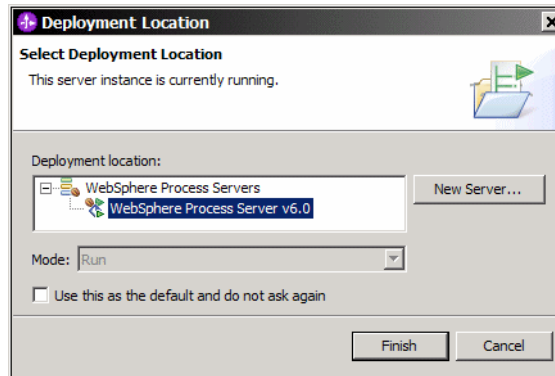
Figure 9-27   Testing: Deployment location

The process starts and events are displayed in the events list (Figure 9-28).
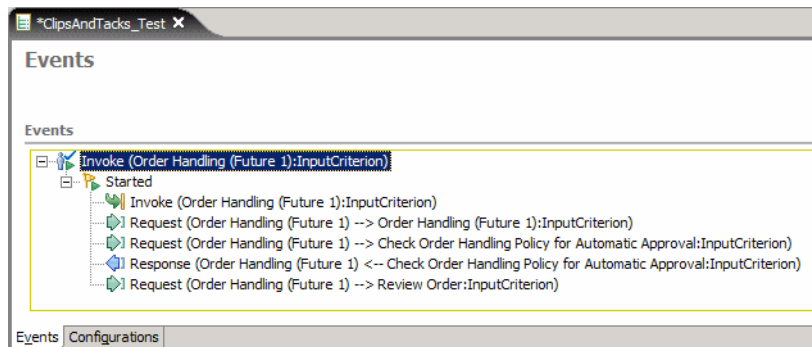
Figure 9-28   Testing: Process events

We can see that the `Order Handling (Future1)` process was invoked and has proceeded through the `Check Order Handling for Automatic Approval` activity (business rule) and is now waiting in the `Review Order` activity (human task).

Processing the human task using the BPC Explorer

We have to play the order manager now and decide if this order is approved or declined. For this purpose we start the BPC Explorer:

► Select the server in the Servers view and *Launch → BPC Explorer* (context menu) or use an external browser with the URL:

```
http://localhost:9080/bpc
```

► The BPC Explorer opens with the My Tasks view and the `ReviewOrder` activity is visible (Figure 9-29).
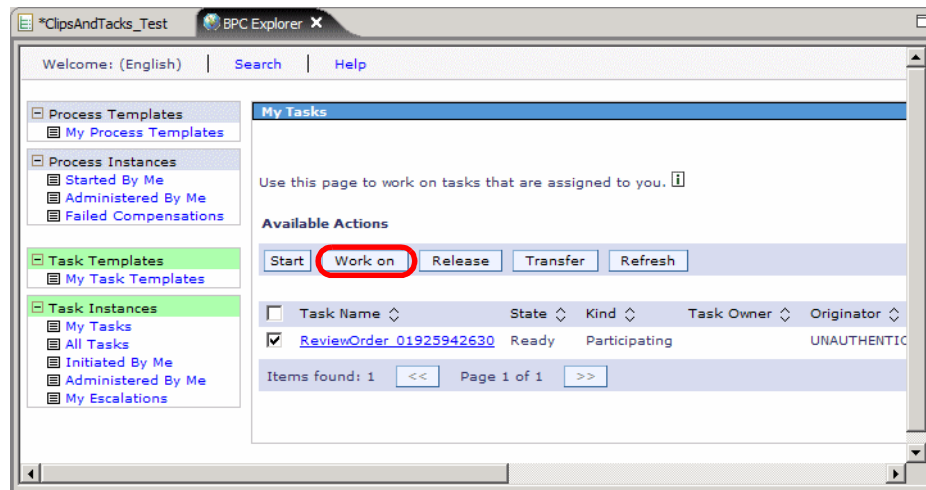
Figure 9-29   BPC Explorer: Tasks list

► You can click on the task itself and look at the details and input and output messages. Note that the output message is empty. The human task must be worked on and the output message completed.

► To work on the task, select the task (check box) and click *Work on*.

► The task open with the input message filled in, and an empty output message (Figure 9-30):

– The output message must be filled with data for further activities.

– There is no simple way to do this, each field must be copied from the input or entered.

– The most important field is the order status; it must be set to either APPROVED or DECLINED.

– When the output message is ready, click *Complete*.

– To delay the decision you can click *Release* and the activity is available for processing once more.

Figure 9-30   BPC Explorer: Work on

► Suppose we approve the order and click *Complete*.

► Refresh the tasks list in the BPC Explorer and you can see that process is now waiting in the `ShipOrdertoCustomer` activity.

► Select the activity and click *Work on*.

► Enter the values for the output message (a packing slip), and click *Complete*.

► The process finishes, which can be seen in the test events list (Figure 9-31).
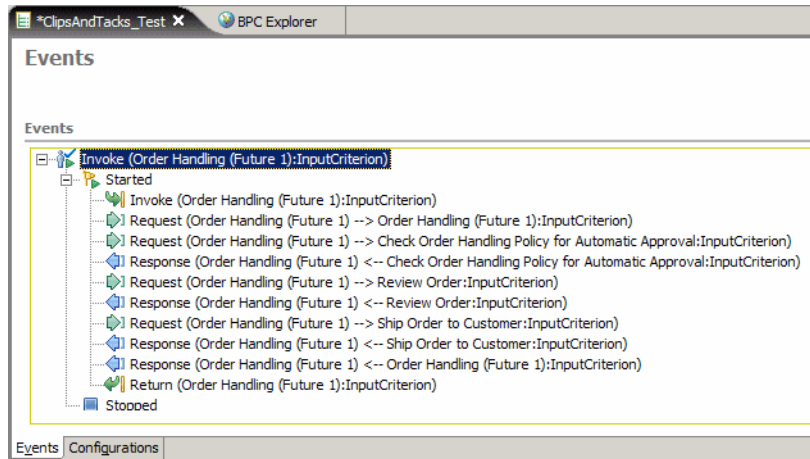
Figure 9-31   Testing: Process complete

We can see the process in the BPC Explorer by clicking *Started By Me* under Process Instances (Figure 9-32).

Note that finished processes are only visible if you deselect *Automatically delete the process after completion* in Figure 9-19 on page 202.
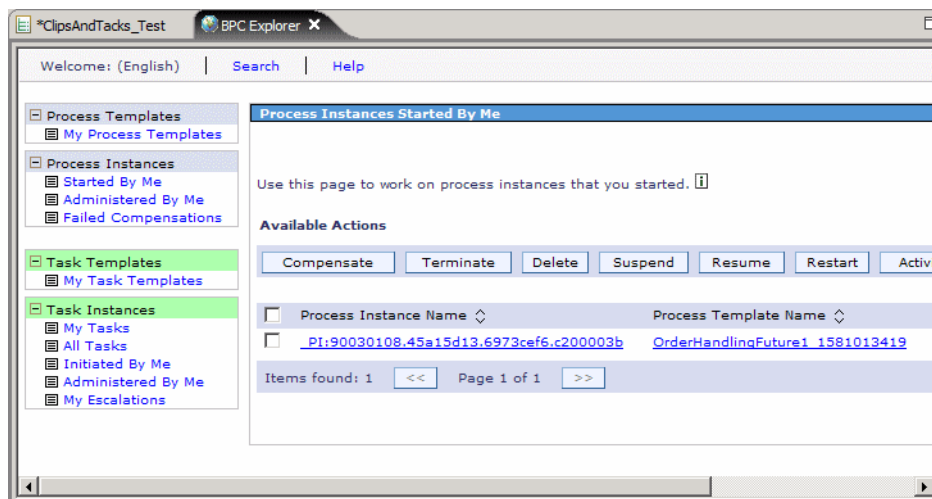
Figure 9-32   BPC Explorer: Complete processes

Click on the process to see its input and output messages, as well as the list of activities, with an indication if finished or skipped (Figure 9-33).
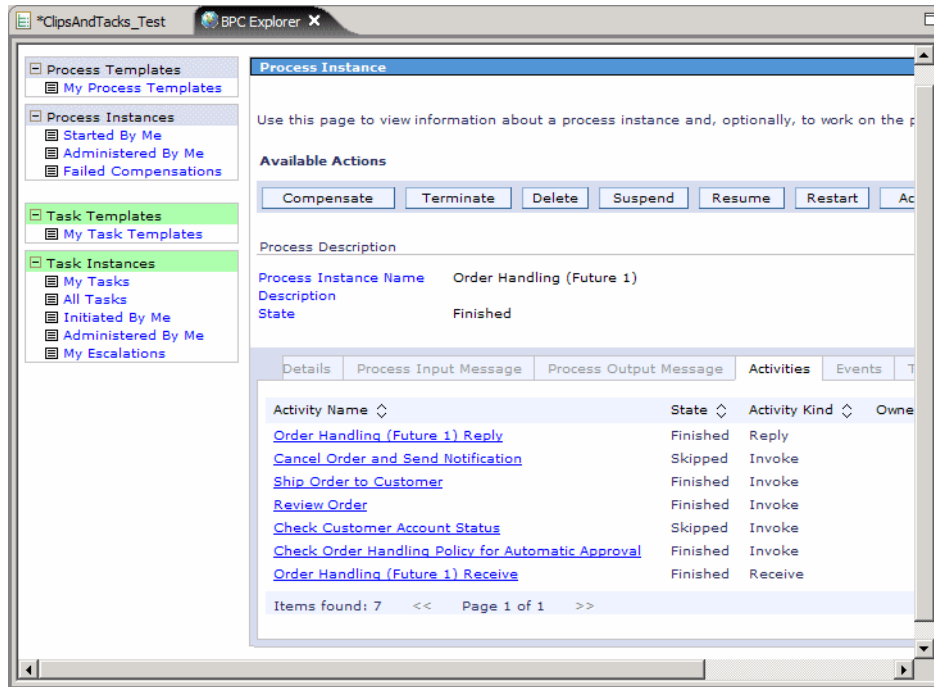
Figure 9-33   BPC Explorer: Completed activities

Rerunning tests and reusing data

To rerun a test, select the top event and *Rerun* (context menu). The same input data is used. For example, you can decline the order at this time.

For another test with different data, select the project and *Test → Test Module*. Do not forget to select the `Order Handling (Future 1)` component. To reuse the data from the data pool, select *Input* and click *Data Pool*. Select an existing entry, then modify the input data before clicking *Continue*.

For example, enter data with a total price below 750.00 and the order is automatically approved. In the BPC Explorer the `ShipOrdertoCustomer` activity is waiting and can be completed.

When closing the test, you are prompted to optionally save the execution trace. You can use any project location to save the execution trace.

Add a stand-alone reference to invoke the process

To be able to invoke the process from the Web front-end we define a stand-alone reference in the assembly diagram:

► Open the assembly diagram.

► Click the arrow next to the *Import* icon ![icon], then select the *Stand-Alone References* icon ![icon], then click into the diagram (Figure 9-34).
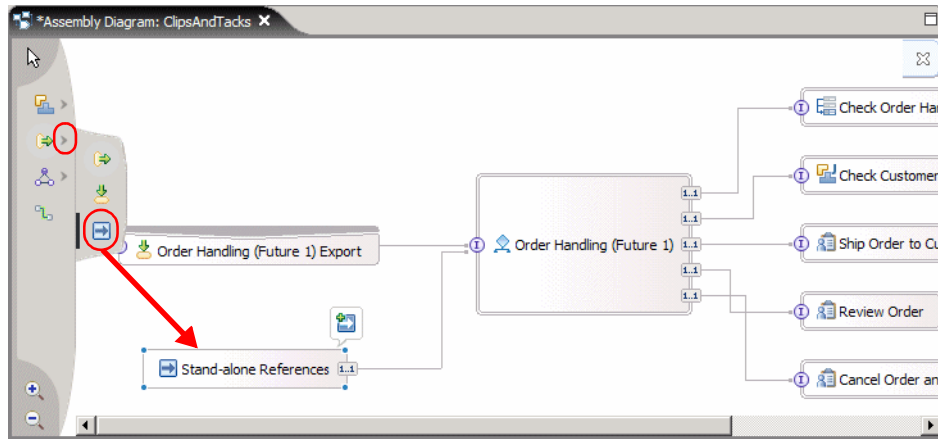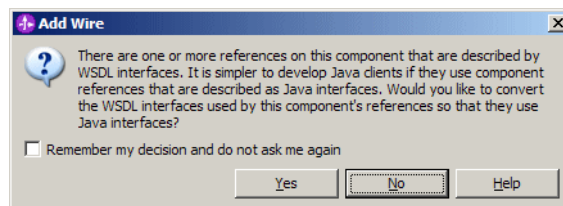

Figure 9-34   Adding a stand-alone reference

► Link the stand-alone reference to the Order Handling (Future 1) process. Select the wire icon ![icon], then click on the reference and on the process:

– A prompt appears to create a matching reference, click *OK*.

– A prompt appears to create a Java interface instead of the WSDL interface. **Click *No***. We will use the WSDL interface.

The stand-alone reference is named `OrderHandlingFuture1Partner` (look in the Properties view). We will use this name in the Web front-end,

► Rearrange the diagram (Figure 9-35).
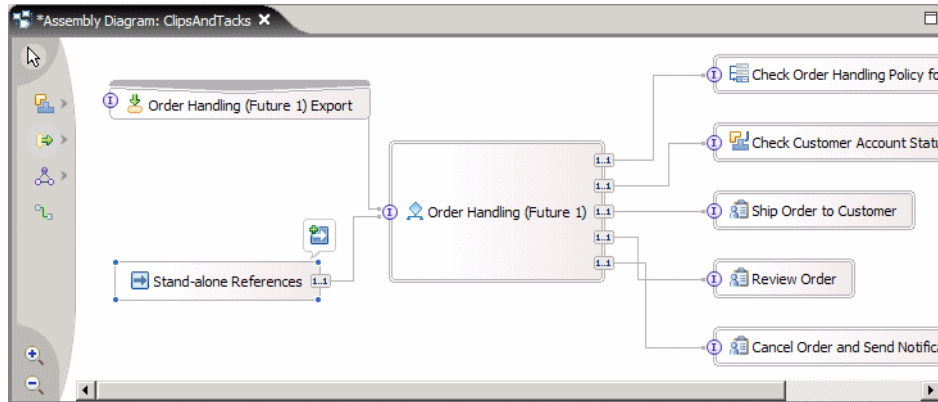
► Save and close the diagram.

Figure 9-35   Process diagram with stand-alone reference

Implementing a Web front-end

ClipsAndTacks uses a Web front-end application to submit customer orders to the business process. Through the self-service front-end, customers can select the products to be ordered and enter the order into the `CLIPTACK` database.

We do not describe here how to build such a Web application. We only describe the architecture and functionality, and how to imbed the Web application into the business process.

Architecture

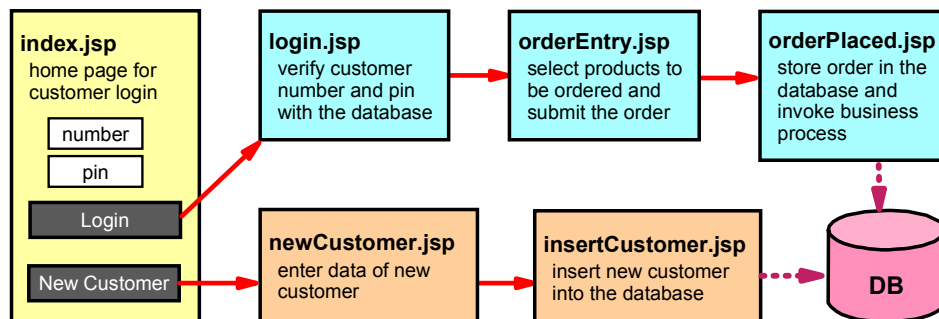The architecture of the Web application is shown in Figure 9-36.

Figure 9-36   Web front-end architecture

► The home page is used for login of the customer. New customer can register themselves in the database.

► The customer then selects the products from a (small) catalog and submits the order.

► The order is stored in the database and the business process is invoked.

Database access

To access the database the Web application uses a data access object (DAO) and data transfer objects (DTO). The DAO contains the logic to access the `CLIPTACK` database using JDBC. The DTOs hold the customer, product, and order information that is transferred between the Web application and the DAO.

► **CustomerDTO**—Holds all the customer information, matching the `CUSTOMER` table in the database.

► **ProductDTO**—Holds all the product information, matching the `PRODUCT` table.

► **OrderItemDTO**—Hold the information for one item ordered (quantity, product number and name, price).

► **OrderDTO**—Holds the complete order information with order number, total price, order status, automatic approval (set during the business process), one `CustomerDTO`, and an array of `OrderItemDTO`.

► **ClipsTacksDAO**—Used for all database access. The database is accessed through the data source defined in the server. The DAO provides these methods:

  – `getCustomer`—Retrieve a customer and verify the pin.

  – `insertCustomer`—Insert a new customer into the database.

  – `deleteCustomer`—Delete a customer and all its orders from the database.

  – `updateCustomer`—Update the rating and credit limit of a customer (used during the business process).

  – `getProducts`—Retrieve all the products for order entry.

  – `insertOrder`—Insert a new order and its order items into the database.

  – `updateOrder`—Update the order status (used during the business process).

  – `getConnection`—Get a database connection using a data source reference. This method is called by all the access methods.

  – `commit`—Commit changes to the database. This method is called by the methods that update the database.

Servlets

The Web front-end provides three servlets to initialize and manage the database:

▶ **CreateDatabaseServlet**—Run the data description language (DDL) statements to define the tables and foreign key relationships, and to load the sample data (five customers, five products, one order). This servlet must be run before using the front-Web end.

   The servlet uses a properties file (`WEB-INF/clipstacks.properties`) that holds the SQL statements that are executed.

▶ **ListServlet**—List the content of the database.

▶ **DeleteServlet**—Delete one customer from the database.

Properties file

The `clipstacks.properties` file contains the SQL statements to define the database tables and load sample data, enabling us to modify the data without reinstalling the application. Figure 9-37 shows an extract of the file.

```
database=CLIPTACK
createCustomer=CREATE TABLE CT.CUSTOMER ( customernumber INTEGER NOT NULL,
    companyname VARCHAR(32) NOT NULL, ..., PRIMARY KEY (customernumber) )
createProduct=CREATE TABLE CT.PRODUCT ( productnumber CHAR(10) NOT NULL,
    productname VARCHAR(40) NOT NULL, ..., PRIMARY KEY (productnumber) )
createOrder=CREATE TABLE CT.ORDERMAIN ( ordernumber INTEGER NOT NULL,
    customernumber INTEGER NOT NULL, totalprice DECIMAL(8,2) NOT NULL,
    orderstatus CHAR(8) NOT NULL, PRIMARY KEY (ordernumber) )
createItem=CREATE TABLE CT.ORDERITEM ( ordernumber INTEGER NOT NULL,
    productnumber CHAR(10) NOT NULL, quantity INTEGER NOT NULL, PRIMARY KEY
    (ordernumber, productnumber) )
orderCustomer=ALTER TABLE CT.ORDERMAIN ADD CONSTRAINT OrderCustomer FOREIGN
    KEY (customernumber) REFERENCES CT.CUSTOMER ON DELETE CASCADE
itemOrder=ALTER TABLE CT.ORDERITEM ADD CONSTRAINT OrderItemOrder FOREIGN KEY
    (ordernumber) REFERENCES CT.ORDERMAIN ON DELETE CASCADE
itemProduct=ALTER TABLE CT.ORDERITEM ADD CONSTRAINT OrderItemProduct FOREIGN
    KEY (productnumber) REFERENCES CT.PRODUCT ON DELETE CASCADE
insertCustomer=INSERT INTO CT.CUSTOMER VALUES (12345, 'ABC Finance Ltd.',
    'Marc', 'Shankaran', '1 Main Street', 'Buffalo', 'USA', '82840', 777,
    'GOLD', 2000.00, 12345, 'marc@abcfinance.com'),   (11111, ...), ...
insertProduct=INSERT INTO CT.PRODUCT VALUES ('RB-0001', 'All-In-One
    Printer', 150.00, 'Marvellous printer, fax, copier'), ...
insertOrder=INSERT INTO CT.ORDERMAIN VALUES (3001, 12345, 308.00, 'SHIPPED')
insertItem=INSERT INTO CT.ORDERITEM VALUES (3001, 'RB-0001', 1), (3001,
    'RB-0002', 2)
```

Figure 9-37   Properties file

Process invocation

The Web front-end contains a class, **InvokeOrderHandling**, that invokes the business process when an order is placed in the `orderPlaced.jsp`.

The business process is found using the name of the stand-alone reference (`OrderHandlingFuture1Partner`), then the operation (`InputCriterion`) is invoked using an input business object as parameter:

```
standAloneReferenceName = "OrderHandlingFuture1Partner";
operationName           = "InputCriterion";
serviceManager = new com.ibm.websphere.sca.ServiceManager();
service = (Service)serviceManager.locateService(standAloneReferenceName);

DataObject input = // create the business object from the orderDTO
Ticket output = service.invokeAsync(operationName, input);
```

To create the input business object, a factory (`BOFactory`) is used:

```
boFactory = (BOFactory)serviceManager
                      .locateService("com/ibm/websphere/bo/BOFactory");
namespace = "http://Businessitems"; // package name of bus.items
DataObject order = boFactory.create(namespace, "Order");
order.setInt("OrderNumber", orderDTO.getOrderNumber());
...
DataObject customer = boFactory.create(namespace, "CustomerRecord");
customer.setString("CompanyName", customerDTO.getCompanyName());
...
order.setDataObject("Customer", customer);
...
operationType = service.getReference().getOperationType(operationName);
DataObject input = boFactory.createByType(operationType.getInputType());
input.setDataObject("Input", order);
```

The full code on how to invoke the business process is in the `execute` method of the `InvokeOrderHandling` class.

Note: This logic uses the Service Component Architecture (SCA) API as provided by the IBM products. The business objects used in the business process are Service Data Objects (SDO). The specification of SCA and SDO are available at:

http://www.ibm.com/developerworks/webservices/library/specification/ws-sca/
http://www.ibm.com/developerworks/library/specification/ws-sdo/

For more information on SDO consult the redbook *WebSphere Studio 5.1.2 JavaServer Faces and Service Data Objects*, SG24-6361.

Importing the Web application

We provide the `ClipsAndTacksFrontWeb` application as a WAR file that can be imported into Integration Developer:

► In the Web perspective, select *Dynamic Web Projects* and *Import → WAR file* (context menu).

► Click *Browse* to locate the WAR file:

`SG247148\sampcode\wid\webfront\ClipsAndTacksFrontWeb.war`

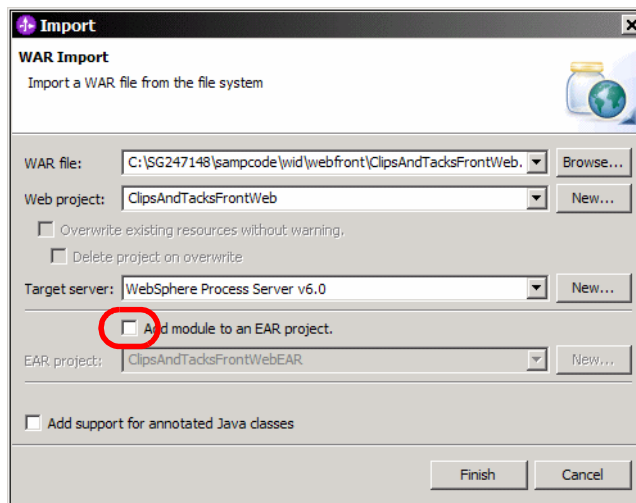► Deselect *Add module to an EAR project*.

► Click *Finish* (Figure 9-38).

Figure 9-38   Import Web front-end WAR file

Data source reference

The data source reference is visible in the Web deployment descriptor as a resource reference with the name `ClipsAndTacks`. This reference has a WebSphere binding of **jdbc/cliptack**, which matches the data source defined in the server.

Attaching the Web front-end to the process application

The Web application invokes the process through the stand-alone reference. This is only possible if the Web application is part of the same enterprise application as the process.

To attach the Web application to a business process application follow these steps:

► In the Business Integration perspective select the `ClipsAndTacks` project and *Open Dependency Editor* (context menu).

► In the Dependencies editor, expand the J2EE section and click *Add*.

► Select the `ClipsAndTacksFrontWeb` project and click *OK*.

► Select the `ClipsAndTacksFrontWeb` project in the list and deselect *On Build Path*, but leave *Deploy with Module* selected.

► Save and close the Dependencies editor (Figure 9-39).
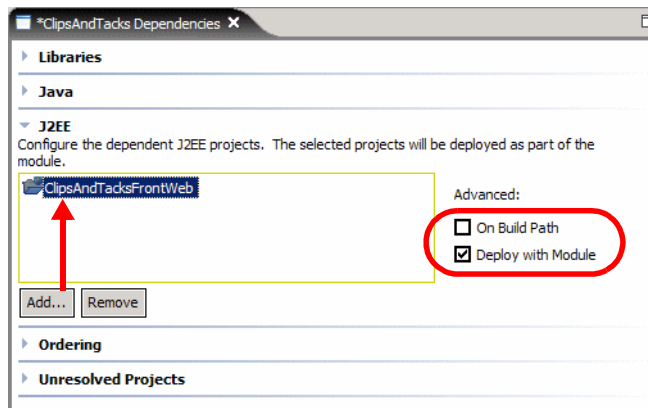
Figure 9-39   Dependencies editor to add Web project

Tip: Adding a project using the Dependencies editor adds the project to the enterprise application. You can verify this by opening the deployment descriptor of the `ClipsAndTacksApp` EAR. IF you use the EAR deployment descriptor to add a Web project, the project may disappear when you deploy or redeploy the application. Always use the Dependencies editor!

Redeploy the application

To activate the Web front-end in the server we have to redeploy the application:

► Select the server in the Servers view and *Publish* (context menu).

► Wait until the operation is finished. If the operation does not want to finish, cancel it.

Initialize the database

The `CLIPTACK` database does exist at this point (see "Create a data source for the database" on page 194), but the tables have not been defined.

To initialize the database we run the `CreateDatabaseServlet`:

► Expand the `ClipsAndTacksFrontWeb` project deployment descriptor.

► Select the `CreateDatabaseServlet` and *Run → Run on Serve*r (context).

► The process server is preselected. Select *Set server as project default* and click *Finish*.

► The servlet runs and displays its results:

```
Initializing database CLIPTACK
Connecting to database
Dropping existing tables...
drop table ct.orderitem
...
Creating tables...
Creating table CUSTOMER...
...
Creating foreign keys...
Inserting data into tables...
Insert table CUSTOMER...
...
End of database create: CLIPTACK
```

► You can also use an external browser with the URL:

```
http://localhost:9080/ClipsAndTacksFrontWeb/CreateDatabaseServlet
```

Tip: You can initialize the database at any time by rerunning the `CreateDatabaseServlet.`

List the database content

You can list the database at any time by running the `ListServlet`:

```
http://localhost:9080/ClipsAndTacksFrontWeb/ListServlet
```

Figure 9-40 shows the initial database content.

Customer data

Number	Company	Contact	Address	Rating	Class	Credit	Pin	e-mail
12345	ABC Finance Ltd.	Marc Shankaran	1 Main Street, Buffalo, USA, 82840	777	GOLD	$2000.00	12345	marc@abcfinance.com
11111	Auto Insurance Company	Richard Doe	27 New Street, Etobicoke, Canada, M8Z 2S6	666	SILVER	$1300.00	11111	richard@autoinsurance.com
22222	ABC University	James James	8200 New Street, Markham, Canada, M9W 9M9	555	REGULAR	$500.00	22222	james@abcuniversity.com
33333	US Auto Financing	Jack Jackson	55 Long Ave, Chicago, USA, 60606	765	GOLD	$1999.00	33333	jack@usautofinance.com
44444	Insurance For You	Dana Danison	44 Dawes Rd, Toronto, Canada, M4C 5C2	632	SILVER	$1444.00	44444	dana@insurance4you.com

Customer Number [   ]   [Delete Customer]

Product data

Number	Name	Price	Description
RB-0001	All-In-One Printer	$150.00	Marvellous printer, fax, copier
RB-0002	Manager Chair	$79.00	Leather chair that turns and massages
RB-0003	5 MP Digital Camera	$499.00	This digital camera takes MPEG4 movies
RB-0004	Cordless Phone with Answering Machine	$89.00	This phone features a digital answering machine
RB-0005	3-Drawer File Cabinet	$214.00	This cabinet has enough space for your lunch

Order data

Number	Customer	Total Price	Status	Quantity/Product	
3001	12345	$308.00	SHIPPED	1	RB-0001
				2	RB-0002

End of database list

Figure 9-40   Database content

Implementing customized JSPs for the BPC Explorer

At this point we can submit orders using the Web front-end, but before we do that we want to improve the handling of the human tasks so that we do not have to enter all the data for the output message.

A customized human task JSP™ is invoked by the BPC Explorer instead of the standard JavaServer Faces pages. The customized JSPs can create the output message based on the input message and also provide actions. For example, we are able to approve or decline an order through a pull-down menu.

Importing the human task JSPs

We provide the human task JSPs in a Web application with the name `ClipsAndTacksHumanTasks`. We have to import the WAR file and attach the Web project to the business process:

► Import the `ClipsAndTacksHumanTasks.war` file from:

    `SG247148\sampcode\wid\humantaskBPC\ClipsAndTacksHumanTasks.war`

Be sure to deselect *Add module to an EAR project*.

► Use the Dependencies editor to add the `ClipsAndTacksHumanTasks` project to the `ClipsAndTacks` module (J2EE section).

► Redeploy the module.

Human task JSPs

There are five JSPs in the project:

► `cancelOrderInput.jsp`—Handles the input message for order cancellations

► `reviewOrderInput.jsp`—Handles the input message for order reviews

► `reviewOrderOutput.jsp`—Handles the output message for order reviews

► `shipOrderInput.jsp`—Handles the input message for order shipments

► `shipOrderOutput.jsp`—Handles the output message for order shipments

Note that the order cancellation activity has no output message.

Data source reference

The `ClipsAndTacksHumanTasks` Web project also contains the `ClipsAndTacks` resource reference for the data source. The human task JSPs have to maintain the database with the latest customer rating and available credit and the order status.

Configuring the BPC Explorer for customized JSPs

The next step is to configure the BPC Explorer to invoke the customized JSPs:

► Open the **Review Order** activity from the assembly diagram or from the Business Integration view (under *Business Logic → Human Tasks*).

► Select the *BPC Explorer* under Client settings, and in the Properties view click *Add* to define a customized JSP.

► In the JSP Definition dialog select *Input message JSP* for type and *Potential owner* for apply. The click *Browse* and locate the `reviewOrderInput.jsp` in the `ClipsAndTacksHumanTasks` project (Figure 9-41). Click *OK*.
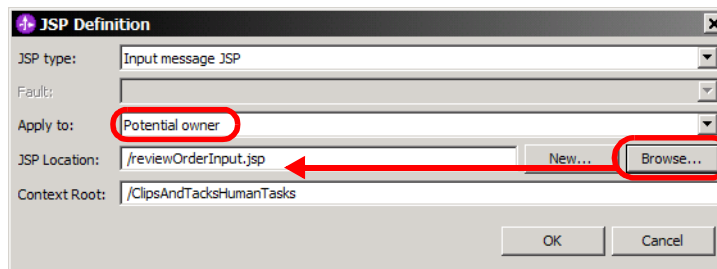
Figure 9-41   JSP selection for a human task

► The selected JSP appears in the list.

► Click *Add* once more to define the *Output message JSP* as `reviewOrderOutput.jsp`.

► The list of JSPs for the Review Order activity is shown in Figure 9-42.
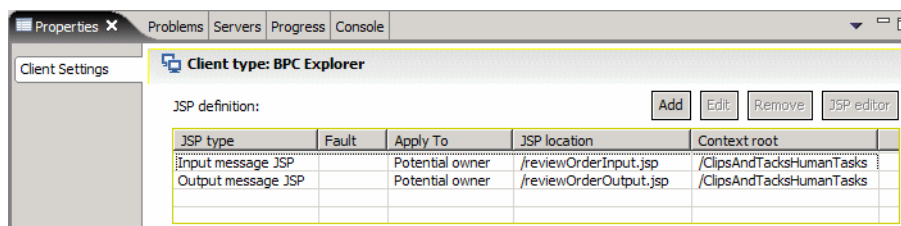
Figure 9-42   Human task JSPs for reviewing the order

Repeat this process for the **Ship Order to Customer** activity and define the `shipOrderInput.jsp` and `shipOrderOutput.jsp`.

Repeat this process for the **Cancel Order and Send Notification** activity and define the `cancelOrderInput.jsp` (there is no output message).

Human task JSP processing logic

To understand how to implement a human task JSP let us open the JSPs for the **Review Order** activity (`reviewOrderInput.jsp` and `reviewOrderOutput.jsp`).

The `reviewOrderInput.jsp` receives these objects in the request block:

► **message**—The data object of the input message

► **messageMap**—A Java map holding XPath expression attribute names and their values of the input message. The XPath expression names mirror the layout of the data object.

The `reviewOrderOutput.jsp` receives the objects of the same names, but for the output message. **The output JSP has no access to the input message**.

In our case the output message is basically the same as the input message, with a few fields changed (for example, the order status changes to `APPROVED` or `DECLINED`). To provide the output JSP with the input message, we pass the input message from the input JSP to the output JSP.

Input message JSP

In this section we look at some code fragments of the `reviewOrderInput.jsp`:

► Retrieve the input message:

```
com.ibm.ws.bo.impl.BusObjImpl inputOrder =
        (com.ibm.ws.bo.impl.BusObjImpl)request.getAttribute("message");
```

► Pass the input message to the output JSP:

```
request.setAttribute("inputMessage", inputOrder);
```

► Retrieve the input data objects:

```
DataObject order     = inputOrder.getDataObject("Input");
DataObject customer  = order.getDataObject("Customer");
DataObject procpref  = order.getDataObject("ProcessingPreference");
java.util.List items = order.getList("OrderItems");
```

► Retrieve data items for display in the JSP:

```
int customerNumber = customer.getInt    ("CustomerNumber");
String company     = customer.getString("CompanyName");
double totalPrice  = order.getDouble    ("TotalPrice");
```

► Update the database with new customer rating and available credit (this happens if an order was automatically approved, but over the credit):

```
boolean automatic   = procpref.getBoolean("automaticApproval");
if ( automatic ) {
   clipstacksDAO.updateCustomer( customer.getInt("CustomerNumber"),
      customer.getInt("Rating"),
```

```
                new java.math.BigDecimal(customer.getDouble("AvailableCredit")) );
            orderStatus = "Auto-APPROVED, over available credit";
    }
```

► The rest of the JSP displays the input message in an HTML table.

Output message JSP

In this section we look at some code fragments of the `reviewOrderOutput.jsp`:

► Retrieve the output message:

```
com.ibm.ws.bo.impl.BusObjImpl outputOrder =
        (com.ibm.ws.bo.impl.BusObjImpl)request.getAttribute("message");
```

► Retrieve the input message (passed from the input JSP):

```
com.ibm.ws.bo.impl.BusObjImpl inputOrder  =
    com.ibm.ws.bo.impl.BusObjImpl)request.getAttribute("inputMessage");
```

► When an output JSP is called from the **Work On** action in BPC Explorer, a
  prefix is set to be used in JSP fields for the output message:

```
String prefix = (String)request.getAttribute("prefix");
```

► Prepare the output message:

```
DataObject customerIn  = orderIn.getDataObject("Customer");
java.util.List itemsIn = orderIn.getList("OrderItems");
order.setDataObject("Customer", customerIn);
order.setList      ("OrderItems", itemsIn);
order.setInt       ("OrderNumber", orderIn.getInt("OrderNumber") );
order.setDouble    ("TotalPrice",  orderIn.getDouble("TotalPrice") );
order.setString    ("OrderStatus", orderIn.getString("OrderStatus") );
```

► Display a pull-down menu to approve or decline the order:

```
<h3>Approve or decline the order and click Complete</h3>
<SELECT id="status" name="${prefix}/Output/OrderStatus">
    <OPTION value="APPROVED" selected>Approve</OPTION>
    <OPTION value="DECLINED">Decline</OPTION>
</SELECT>
```

Note the prefix and the XPath expression for the name of the field:

```
${prefix}/Output/OrderStatus
```

This notation must be used to store values from JSP fields into the output
message.

The processing for the **Ship Order to Customer** activity is very similar:

► The `shipOrderInput.jsp` also updates the database with the latest customer
  information and passes the input message to the output JSP.

► The `shipOrderOutput.jsp` creates the output message (shipment) and
  updates the database with the order status `SHIPPED`.

The processing for the **Cancel Order and Send Notification** activity is different:

► The `cancelOrderInput.jsp` displays the input message and updates the database with the latest customer information and the order status (`DECLINED`).

► The standard output JSP is displayed by the BPC Explorer for the user to fill-in the e-mail notification.

Run the application in the test environment

The application is now ready to be run with the Web front-end and the customized human task JSPs.

Redeploy the application by removing it from the server and then adding it again (Select *Add an remove projects* in the Servers view).

Using the Web front-end

To start the Web front-end select the `ClipsAndTacksFrontWeb` project and *Run →
Run on Server* (context). Alternatively use an external browser with the URL:

> `http://localhost:9080/ClipsAndTacksFrontWeb/`

► The Welcome (login) page is displayed (Figure 9-43).
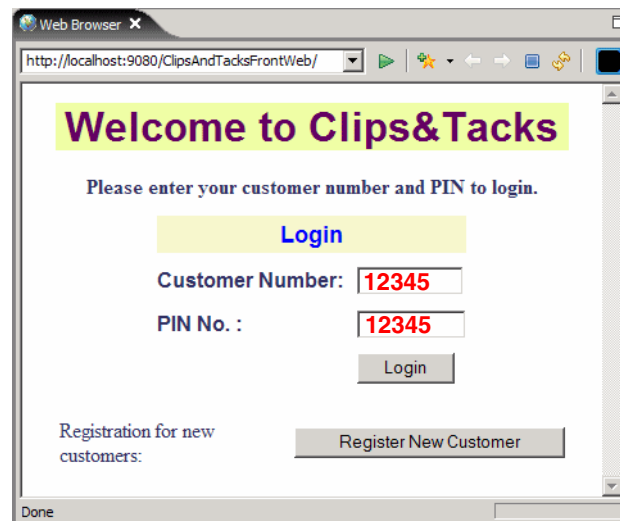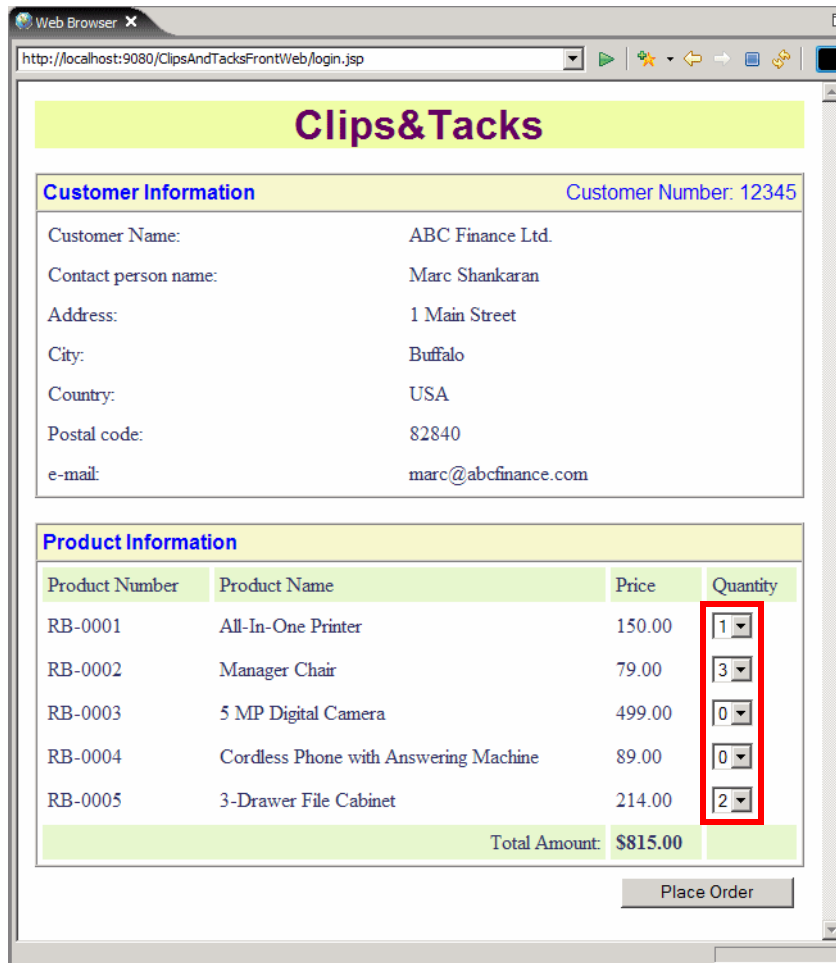
Our passwords are the same as the customer number

Figure 9-43   ClipsAndTacks: Welcome page

► Enter a customer number and pin and click *Login*.

► Place an order by selecting some products (Figure 9-44). Notice how the total amount is updated when you select products. This is done through a JavaScript™ routine.

Note that the total price is over $750. Therefore, the order will go to the order manager for approval through a human task.

Figure 9-44   ClipsAndTacks: Placing an order

► Click *Place Order* and the order is placed and a confirmation is issued (Figure 9-45).
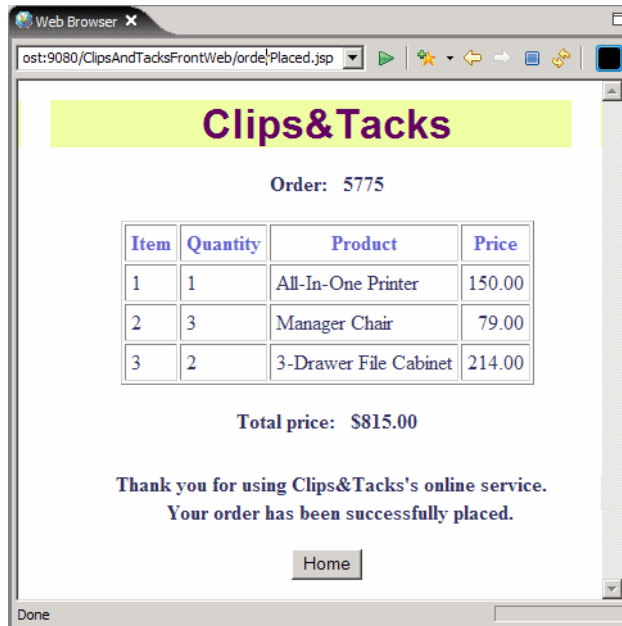
Figure 9-45   ClipsAndTacks: Order confirmation

► The business process is invoked through the stand-alone reference.

Using the human task JSPs with the BPC Explorer

The business process has been invoked and is waiting for human task interaction:

► Start the BPC Explorer (*Launch → BPC Explorer* in the Servers view) or enter the URL in a browser window:

```
http://localhost:9080/bpc
```

► The BPC Explorer opens with the My Tasks view and the `ReviewOrder` activity is visible (Figure 9-46).

You can click on the task to see its messages, or you can select the task and click *Work on*.
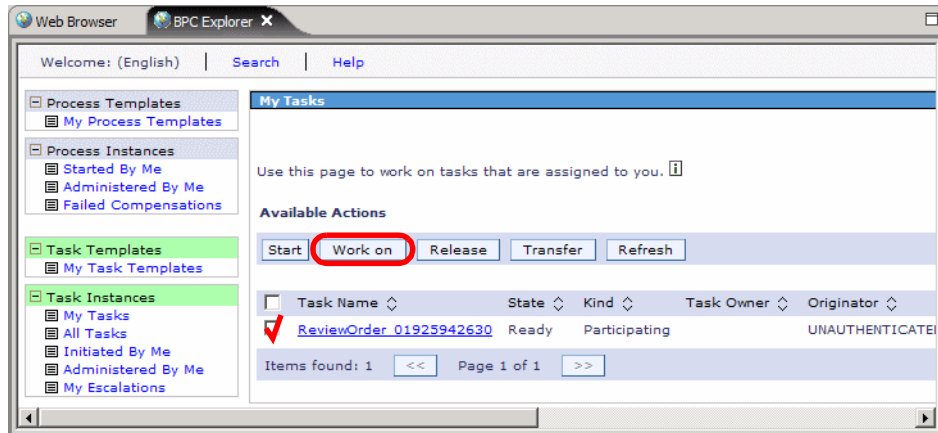
Figure 9-46   BPC Explorer with custom JSPs

► After clicking *Work on*, the custom JSP shows the input message and prompts to approve or decline the order (Figure 9-47).
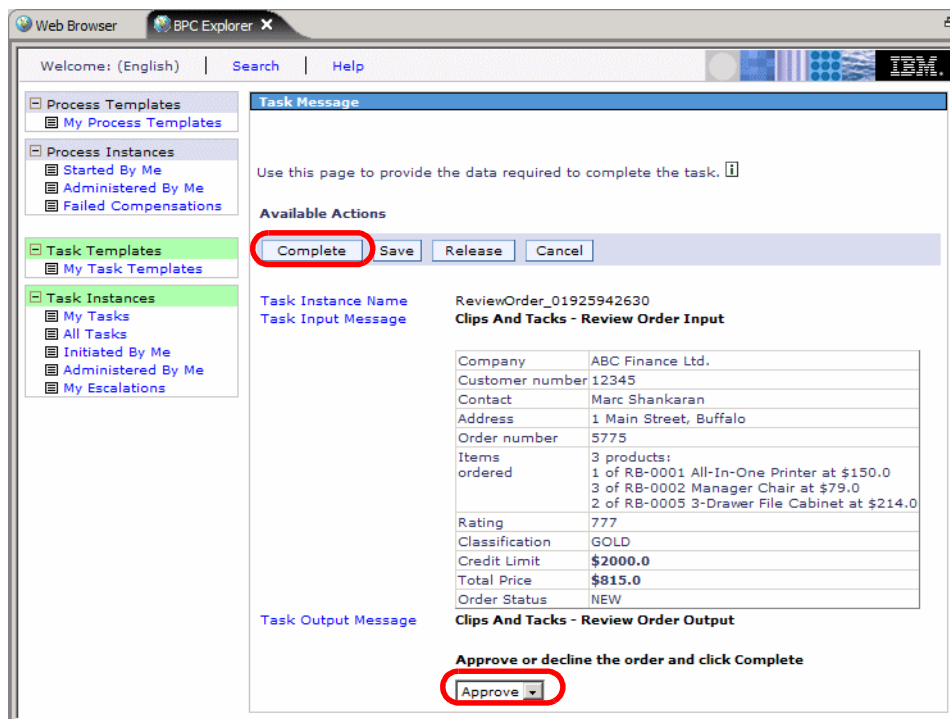
Figure 9-47   BPC Explorer: Custom JSPs for order review

► Select *Approve* and click *Complete*. After refreshing the task list the process is now waiting for shipping. Select the `ShipOrdertoCustomer` task and click *Work on*. The custom JSPs for shipping are displayed (Figure 9-48).
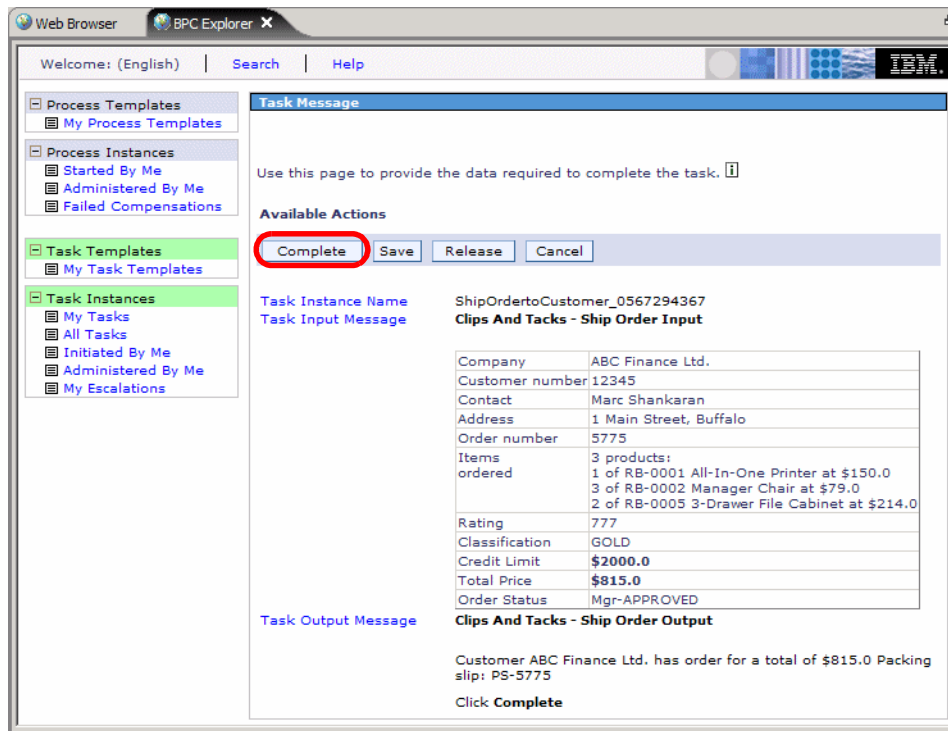
Figure 9-48   BPC Explorer: Custom JSPs for order shipment

► If you decline an order you can work with the custom JSP for order cancellation. The top part of the display is the same as above, the bottom part shows the standard JSP (Figure 9-49).
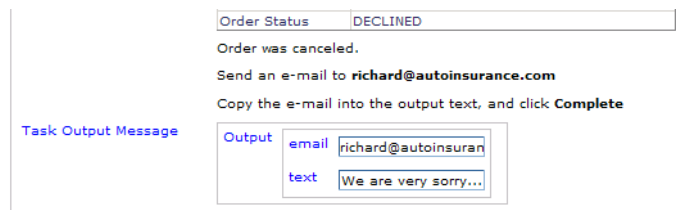
Figure 9-49   BPC Explorer: Custom JSP for order cancellation

Chapter 9. Developing the application using WebSphere Integration Developer   **233**

Implementing an external Web service

To demonstrate additional functionality of the process server and business process assembly, we implement a Web service that is called by the Check Customer Account Status activity.

We already imported the `CreditRating` JavaBean that calls the Web service in "Implementing a Java activity" on page 205.

Importing the Web service

We do not describe how to create the Web service. If you want to learn about creating Web services, refer to the redbook *WebSphere Version 6 Web Services Handbook Development and Deployment*, SG24-6461.

To import the Web service application, follow these steps:

► Select *File → Import → EAR File*.

► Click *Browse* and locate the file:

    SG247148\sampcode\wid\webservice\ClipsAndTacksServiceEAR.ear

► Click *Finish*.

The Web service application consists of a Web project, `ClipsAndTacksService`, which contains a JavaBean, `CreditService`. This JavaBean has one method, `checkCredit`, that has been turned into a Web service:

```
public int checkCredit(int customerNumber, String companyName) {
    int rating = 0;

    // calculate new rating between 500 and 800
    rating = 650 + randomGenerator.nextInt()%150;

    System.out.println("Web Service: Create rating for " + customerNumber +
                       " " + companyName + " ==> rating: " + rating);

    return rating;
}
```

As you can see we do not use a real external service; we only simulate such a response.

To invoke the Web service, the business process application requires the WSDL file from the service project:

    ClipsAndTacksService/WebContent/WEB-INF/wsdl/CreditService.wsdl

Adding the Web service to the assembly diagram

To attach the Web service to the business process follow these steps:

▶ Copy the WSDL file into the business process project:

```
From: ClipsAndTacksService/WebContent/WEB-INF/wsdl/CreditService.wsdl
To:   ClipsAndTacks
```

▶ The Web service shows up in the Business Integration view under Web Service Ports (Figure 9-50).
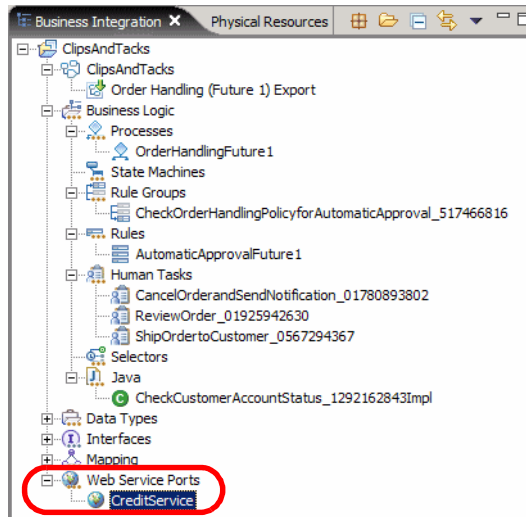
Figure 9-50   Business Integration view with Web service

▶ Open the assembly diagram.

▶ Select the `CreditService` and drag it into the diagram, next to the Check Customer Account Status activity.

▶ In the Component Creation dialog select *Import with Web Service Binding* (Figure 9-51). Click *OK*.
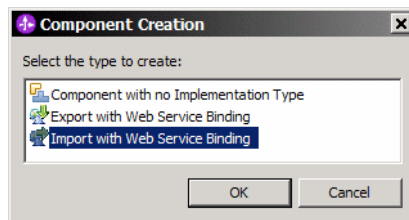
Figure 9-51   Component creation for a Web service

▶ Select the new component (`Import1`) and in the Properties view change the name to **Check Credit Import**.

▶ Select the wire icon and draw a connection from Check Customer Account Status to Check Credit Import. When prompted:

– Click *OK* to create a matching reference in the source node.

– Click *No* to use the WSDL interface (instead of converting to a Java interface).

The complete diagram is shown in Figure 9-52.
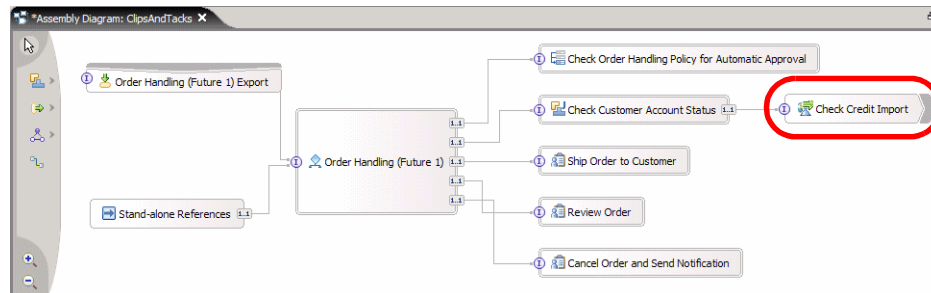
Figure 9-52   Assembly diagram with Web service invocation

Invoking the Web service from an activity

We already imported the `CreditRating` bean (see "Implementing the credit check" on page 206). Now we activate the call of the Web service:

▶ Open the `CreditRating` bean (in the `ClipsAndTacks` project).

▶ Set the flag to invoke the Web service to true:

```
static boolean useWebservice = true;
```

▶ The code to invoke the web service is shown here:

```
Service creditService =
        (Service)serviceManager.locateService("CreditServicePartner");
// create data object for Web Service call
DataObject wsInput = boFactory
      .createByElement("http://bean.creditservice.com","checkCredit");
wsInput.setInt("customerNumber", customerNumber);
wsInput.setString("companyName", companyName);
// invoke Web Service
DataObject wsResult = (DataObject)creditService
                                .invoke("checkCredit", wsInput);
// retrieve result from Web Service result
newRating = wsResult.getInt("checkCreditReturn");
```

> **Note:** The code in the `CreditRating` bean resets the `useWebservice` flag to `false` in an error occurs in the Web service. This allows for continued testing without having to fix the application immediately.

Deploying the Web service

The Web service must be installed in a server. In our case, we just add it to the test server by selecting *Add and remove projects* in the Servers view.

Testing the application with the Web service

Now we can test the Web service invocation by submitting an order with a total price of less than $750.00. The Check Customer Account Status activity is only invoked if the order is automatically approved.

The Console shows the trace of the Web service execution:

```
Check Customer Account Status Invoked
Rating/Credit before: 22222: 555/500.0
...invoking checkCredit Web Service ...
...
Web Service: Create rating for 22222 ABC University ==> rating: 759
Rating/Credit after : 22222: 759/520.4 (0.1)
```

Further messages in the Console show that the new rating and available credit are updated in the database, and the order is shipped:

```
Customer updated: 22222 (759,520.3999999999999)
Order update: 17034 SHIPPED
Shipment PS-17034 for customer ABC University status is: SHIPPED
```

Implementing a human task application

The BPC Explorer provides the functionality to work with human tasks, and with the addition of custom JSPs it is reasonably functional.

Human tasks can also be processed using an API that allows to search for human tasks, claim a task for processing, and complete or release a task. Using this API we can build a human task application that is independent of the BPC Explorer.

Importing the human task application

We provide this application in an interchange file:

```
SG247148\sampcode\wid\zInterchange\ClipsAndTacksHumanCustom-Interchange.zip
```

> **Note:** This interchange file includes the security code described in
> "Implementing security for human tasks" on page 267. The application with
> security also runs in a server where security is not enabled.

To import the application, follow these steps:

► Select *File → Import → Project Interchange*.

► Click *Browse* to locate the `ClipsAndTacksHumanCustom-Interchange.zip` file.

► Select both projects, `ClipsAndTacksHumanCustomEAR` and
  `ClipsAndTacksHumanCustomWeb`.

► Click *Finish*.

Add the `ClipsAndTacksHumanCustomEAR` to the server using *Add and remove
project.*

Architecture

The human task application consists of several JavaServer Faces JSPs and one
JavaBean, **TaskQuery**, which performs all the interaction with the human task
manager using the API.

In addition, the DTOs and the DAO of the `ClipsAndTacksFrontWeb` application are
used for order processing and database access. One extra DTO, `TaskOrderDTO`,
is a copy of the `OrderDTO` with two properties, `taskid` and `taskState`, added. The
`taskid` is the task ID that is used to claim and complete a task. The `taskState` is
either ready or claimed (there are other states that we do not handle).

TaskQuery JavaBean

The `TaskQuery` bean performs all the processing through these methods:

► **getTaskMgr**—Retrieve the task manager session EJB at initialization time
  using a local EJB reference:

```
LocalHumanTaskManagerHome taskHome =
              (LocalHumanTaskManagerHome)initialContext.lookup
              ("java:comp/env/ejb/LocalHumanTaskManagerHome");
// Access the local interface of the local session bean
taskMgr = taskHome.create();
```

The local EJB reference (`ejb/LocalHumanTaskManagerHome`) is defined in the deployment descriptor. It points to the JNDI name:

```
com/ibm/task/api/HumanTaskManagerHome
```

► **execute**—Search for human tasks that are ready for processing. This method accepts a parameter (task name) that enables us to search for tasks of a certain name, for example, `ReviewOrder`.

The search is performed using SQL:

```
selectCLause = "DISTINCT TASK.TKIID, TASK.NAME";
whereClause = "TASK.STATE = (TASK.STATE = TASK.STATE.STATE_READY OR
    TASK.STATE = TASK.STATE.STATE_CLAIMED) AND (TASK.KIND =
    TASK.KIND.KIND_PARTICIPATING OR TASK.KIND = TASK.KIND.KIND_HUMAN)";
orderClause = null;
threshold = null;
QueryResultSet result = taskMgr.query(selectClause,
        whereClause + " AND TASK.NAME LIKE '" + selectedTaskname + "%'",
        orderClause, threshold, null);
```

The task ID and the input data object can be retrieved from the result set:

```
result.next();
TKIID tkiid = (TKIID)result.getOID(1);
String taskname = result.getString(2);
ClientObjectWrapper input = taskMgr.getInputMessage(tkiid);
inputDO = (DataObject)input.getObject();
DataObject order = inputDO.getDataObject("Input");
......
```

From the input data objects an array of `TaskOrderDTO` is built and is available for retrieval.

► **getOrders**—A JSP can use the `getOrders` method to retrieve the list of orders that require an action.

► **claimReviewComplete**—This method claims a review order task, performs the approve or decline action (parameter), updates the database, constructs the output message, and completes the task using the API:

```
// claim the task for processing
ClientObjectWrapper input = taskMgr.claim(tkiid);
// retrieve input data object
DataObject inputDO  = (DataObject)input.getObject();
DataObject orderIn  = inputDO.getDataObject("Input");
.....
// create output data object
ClientObjectWrapper output = taskMgr.createOutputMessage(tkiid);
DataObject outputDO = (DataObject)output.getObject();
DataObject orderOut = outputDO.getDataObject("Output");
if (orderOut == null) {
    ServiceManager serviceManager = new ServiceManager();
```

```
          BOFactory boFactory = (BOFactory)serviceManager
                         .locateService("com/ibm/websphere/bo/BOFactory");
          orderOut = boFactory.create("http://Businessitems", "Order");
          outputDO.setDataObject("Output", orderOut);
      }
      ......
      // complete the task
      taskMgr.complete(tkiid, output);
```

► `claimShipComplete`—This method is similar to the `claimReviewComplete` method to perform the ship order operation.

► `claimCancelComplete`—This method is similar to the `claimReviewComplete` method to perform the cancel order operation.

► `claimOrder`—This method claims a task but does not complete it. We added this method so that the order manager can claim the task and complete it later. This should allow for a correct timing of the task in the Monitor.

► `getOrder`—This method retrieves the last order that was processed (and saved) by any of the `claim` methods.

JavaServer Faces pages

The JSF pages are:

► `index.jsp`—Home page to select the task to be performed (review, ship, cancel).

► `myReviews.jsp`—Retrieve the orders to be reviewed using the `TaskQuery` bean. Display the orders in a table, with a pull-down menu for the action (approve or decline) and a push button (Process) to complete the task.

► `myShipments.jsp`—Similar to `myReviews.jsp` for shipped orders. A shipped order is sent to the `packingSlip.jsp` to print the packing slip.

► `packingSlip.jsp`—Displays the packing slip for a shipped order.

► `myCancellations.jsp`—Similar to `myReviews.jsp` for cancelled orders. A cancelled order is sent to the `emailNotification.jsp` to send the e-mail.

► `emailNotification.jsp`—Displays the e-mail to be sent.

Additional pages are used for the home page and for security:

► `index.html`—A simple home page that forwards to the `index.jsp`.

► `login.jsp` and `loginError.html`—These pages are used for login of a user (for example, order manager) when running with security (see "Implementing security for human tasks" on page 267).

Her are some extract of the JSP code for review order:

► Access the `TaskQuery` bean on the HTTP session and call the `execute` method:

```
com.clipstacks.human.TaskQuery taskQuery =
         (TaskQuery)session.getAttribute("clipsTacksQuery");
if (taskQuery == null) {
    taskQuery = new com.clipstacks.human.TaskQuery();
    session.setAttribute("clipsTacksQuery", taskQuery);
}
taskQuery.execute("ReviewOrder");
```

► Access the number of orders and return if no orders are found:

```
if (taskQuery.getOrders().length == 0) { // no orders found
    FacesContext facesContext = FacesContext.getCurrentInstance();
    facesContext.getExternalContext().redirect
        ("faces/index.jsp?message=No orders for review found");
}
```

► The orders are displayed in a JSF data table.

► A hidden field is used to pass the order number when submitting an order for processing:

```
<h:inputHidden id="reviewNumber"
               value="#{varOrders.orderNumber}"></h:inputHidden>
```

► A pull-down menu is used for the action (approve, decline, or claim):

```
<h:selectOneMenu styleClass="selectOneMenu"
                             id="orderAction">
    <f:selectItem itemValue="NONE" itemLabel="Select an action" />
    <f:selectItem itemValue="APPROVED" itemLabel="Approve" />
    <f:selectItem itemValue="DECLINED" itemLabel="Decline" />
    <f:selectItem itemValue="CLAIMED" itemLabel="Claim" />
</h:selectOneMenu>
```

► The process button is a JSF command button:

```
<hx:commandExButton type="submit" value="Process"
        styleClass="commandExButton" id="completeButton"
        action="#{pc_MyReviews.doCompleteButtonAction}">
</hx:commandExButton>
```

► The process button invokes the action logic (`doCompleteButtonAction`) in the JSF Java class (`MyReviews.java`):

```
// get the action (approve or decline)
String action = (String)getOrderAction().getValue();
// get the order number from the hidden field
int reviewNumber = ((Integer)getReviewNumber().getValue()).intValue();
System.out.println("Action="+action + " order "+reviewNumber);
String message = null;
```

```
// verify that an action was selected
if (!action.equals("APPROVED") && !action.equals("DECLINED") &&
                              !action.equals("CLAIMED")) {
   message = "No action (Approve/Decline) was selected";
} else {
   // get the TaskQuery bean and process the order
   TaskQuery taskQuery = (TaskQuery)getSessionScope()
                              .get("clipsTacksQuery");
   if (taskQuery == null) {
      message = "ReviewOrder: Session data not available";
   } else {
      if (action.equals("CLAIMED"))
         message = taskQuery.claimOrder(reviewNumber);
      else  // APPROVED or DECLINED
         message = taskQuery.claimReviewComplete(action, reviewNumber);
   }
}
// prepare the response message
getRequestScope().put("message", message);
return null;
```

► For shipped and cancelled orders the order is forwarded to the `packingSlip` or `emailNotification` JSP:

```
getRequestScope().put("order", taskQuery.getOrder());
```

Using the human task application

To see how the human task application works, submit a few orders using the Web front-end. Then start the human task application by selecting the Web project (`ClipsAndTacksHumanCustomWeb`) and *Run → Run on Server*:

► The human task application starts with the home page (Figure 9-53).
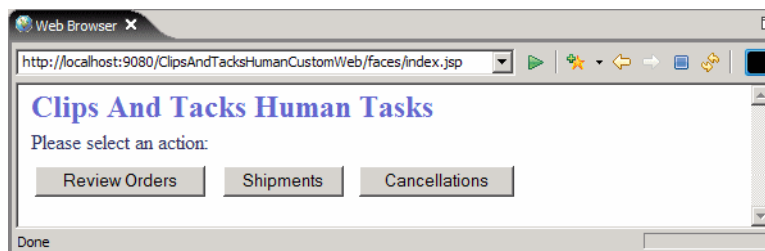
Figure 9-53   Human task application: Home

► Click *Review Orders* and the orders to be reviewed are displayed (Figure 9-54).

Figure 9-54   Human task application: Orders for review (1)

► Pick an order, select the action, and click *Process*.

– **Approve**—The order is claimed and approved and proceeds to shipping.

– **Decline**—The order is claimed and declined and proceeds to cancellations.

– **Claim**—The order is claimed and can be approved or declined later. The order stays in the list with the text `Task is CLAIMED`.

> **Note:** We implemented the claim action so that we can simulated the order manager activity of *thinking* about whether an order should be approved or declined.

► The page is redisplayed with the message and the order disappears for approve and decline actions (Figure 9-55).
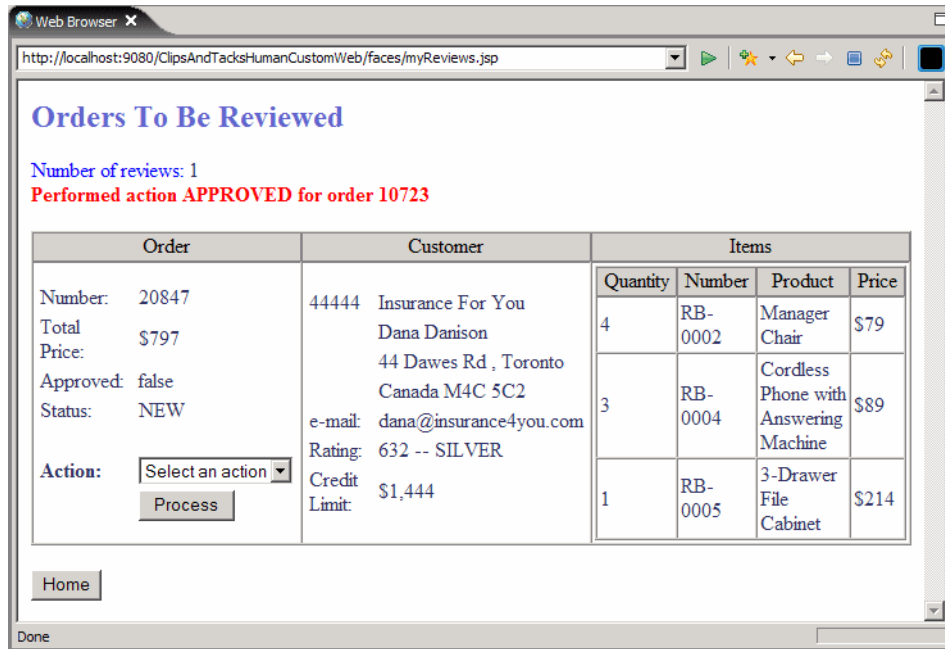
Figure 9-55   Human task application: Orders for review (2)

► Decline the other order and click *Process*. All orders to be reviewed are processed. Click *Home*, then click *Shipments*. There should be a few orders to be shipped (Figure 9-56).
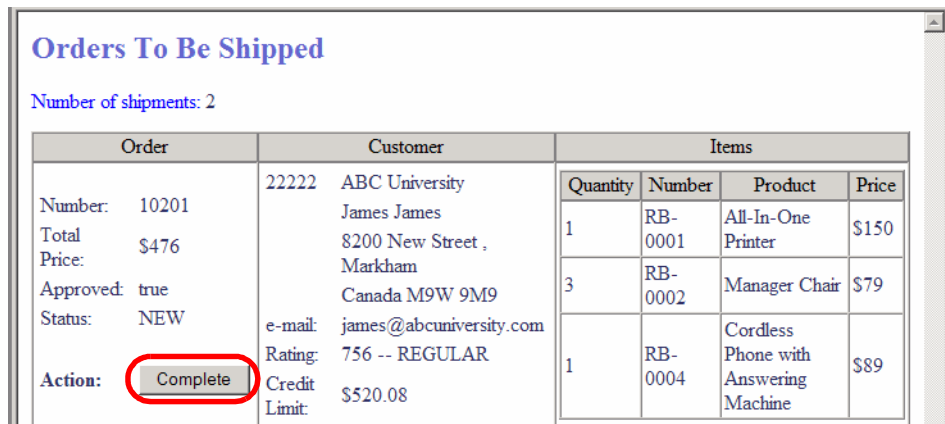
Figure 9-56   Human task application: Orders for shipment

► Select an order and click *Complete*. The shipment is confirmed and the packing slip is displayed for printing (Figure 9-57).
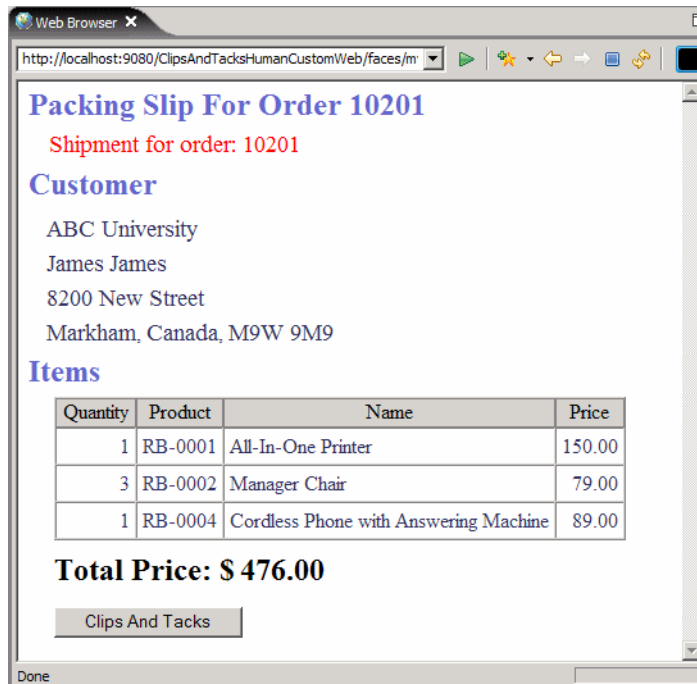
Figure 9-57   Human task application: Packing slip for shipped order

► Click *Home*, then click *Cancellations*. The cancelled order is displayed (Figure 9-58).

Figure 9-58   Human task application: Orders for cancellation

► Click *Complete* to process the order and send an e-mail notification (Figure 9-59).
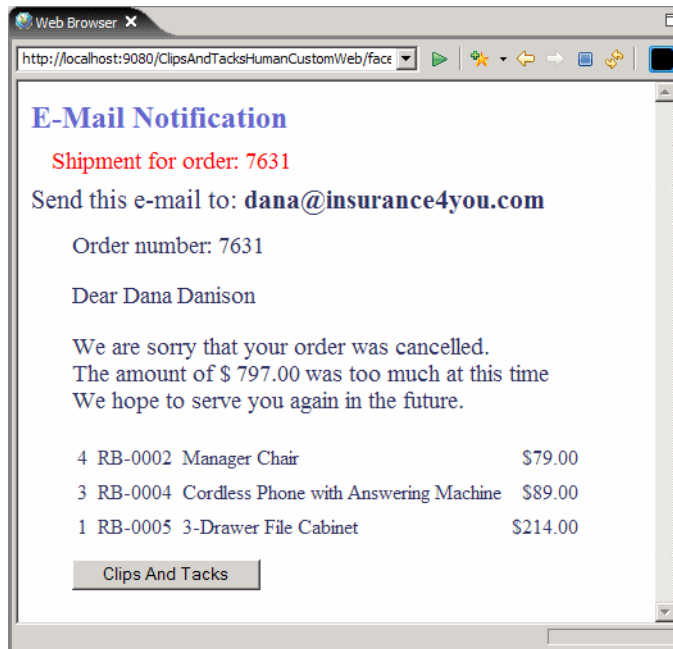
Figure 9-59   Human task application: e-mail notification

Summary

In this chapter we described in detail how to implement the business process application after importing the BPEL and associated files generated by WebSphere Business Modeler.

We showed how to complete the activities for business rules, Java, and human tasks. Then we implemented a Web front-end to invoke the business process. We showed how to manage human tasks using the BPC Explorer without or with customized JSPs. Finally we showed how to invoke an external Web service from an activity, and how to implement a stand-alone application for human task management.

> **Important:** In a real development environment we would use team development with a repository, such as Rational ClearCase® or Concurrent Versions System (CVS).

Problems encountered

In this section we describe a number of problems that we encountered. This problems may be fixed in updates to the Integration Developer.

Deployment failure: duplicate valid from date

After making modifications to the installed business process application, deployment to the process server failed with this message:

```
[12/20/05 15:29:37:238 PST] 0000006a SystemErr     R
com.ibm.etools.j2ee.commonarchivecore.exception.DeploymentDescriptorLoadExc
eption: IWAE0022E Exception occurred loading deployment descriptor for
module "ClipsAndTacksEJB.jar" in EAR file
"e:\Workspaces\WID601sg247148\ClipsAndTacksApp"
!Stack_trace_of_nested_exce!
com.ibm.etools.j2ee.commonarchivecore.exception.DeploymentDescriptorLoadExc
eption: META-INF/ejb-jar.xml
!Stack_trace_of_nested_exce!
com.ibm.etools.j2ee.commonarchivecore.exception.ResourceLoadException:
IWAE0007E Could not load resource "META-INF/ejb-jar.xml" in archive
"ClipsAndTacksEJB.jar"
!Stack_trace_of_nested_exce!
Wrapped exception
org.xml.sax.SAXParseException: Duplicate unique value [ValidFrom] declared
for identity constraint of element "session".
```

When examining the `ejb-jar.xml` file in the `ClipsAndTacksEJB` project, we indeed found several duplicate entries of the form:

```
<env-entry>
    <env-entry-name>validFrom</env-entry-name>
    <env-entry-type>java.lang.String</env-entry-type>
    <env-entry-value>2005-12-01T17:00:00</env-entry-value>
</env-entry>
```

After deleting the duplicate entries the application can be deployed to the server without error.

Failure to remove old process instances and activities

It is possible that after removing a project from the server using *Add and remove projects* that old process instances and their activities are not deleted from the business process database (not the `CLIPTACK` database).

To delete old instance permanently you can use the administrative console to uninstall an enterprise application:

▶ Add the project to the server.

▶ Use the BPC Explorer that no process instances are found.

▶ If there are instance, start the administrative console.

▶ Expand *Applications* and select *Enterprise Applications*. The list of installed applications is displayed (Figure 9-60).
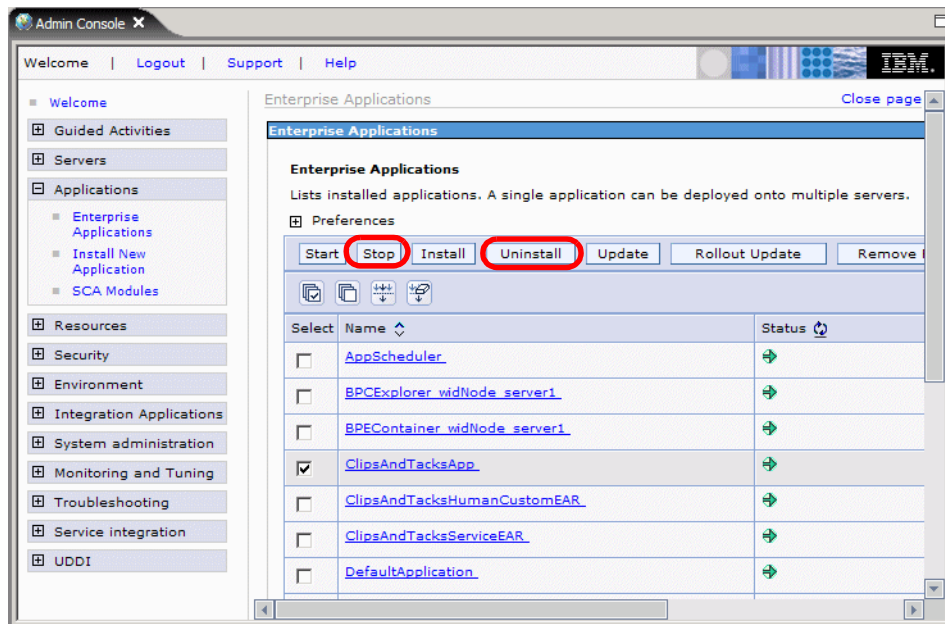

Figure 9-60   Administrative console: Installed applications

▶ Select the `ClipsAndTacksApplication` and click *Stop*.

▶ When the application is stopped, select it again and click *Uninstall*.

▶ When uninstallation is finished, save the changes (click *Save*).

▶ Close the administrative console.

▶ Add the project to the server again. The old process instances should be deleted.

Human task JSP cannot be selected

When defining the custom human task JSPs to the BPC Explorer it happened that after selection (see Figure 9-41 on page 226) the *OK* button was disabled.

▶ To fix this you have to open the `.websettings` file in the `ClipsAndTacksHumanTasks` project. Remove the line `<lib-modules/>`.

▶ Save and close the file.

▶ You have to change the filter to see files that start with a period (Figure 9-61).
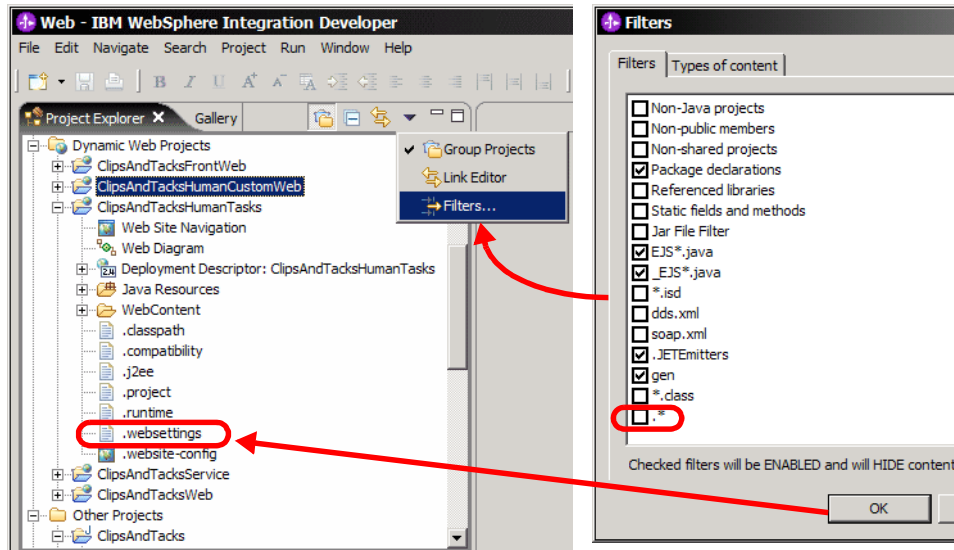
Figure 9-61   Changing the filter to see special files

WSDL file deleted from the EJB project

When we updated the human tasks for security (see "Using the groups for human tasks" on page 274) we noticed that the `wsdl` folder was deleted from the EJB project (`ClipsAndTacksEJB/META-INF`). This caused the application to fail when the Web service was invoked from the Java task.

You can have the `wsdl` folder regenerated by following these steps:

▶ Complete all updates to the human tasks.

▶ Verify that the `wsdl` folder has been deleted from the EJB project.

▶ Open the assembly diagram and delete the Cedit Service Import and the reference to the import (the little box `[1..1]` in Check Customer Account Status). Save the diagram.

▶ Delete the `CreditService` under Web Service Ports in the Business Integration view.

▶ Recreate the Web service invocation by following the instructions in "Adding the Web service to the assembly diagram" on page 235.

10

Deploying and running the application in Process Server

In this chapter we describe how to export the finished application from WebSphere Integration Developer and deploy it into a real WebSphere Process Server.

We describe how to configure the server with a data source and how to install the enterprise application.

Then we describe how to run the application using the Web front-end, the BPC Explorer, and the human task application.

Note: the activities in this chapter are performed by the system administrator.

Export application from Integration Developer

The complete ClipsAndTacks application consists of three enterprise applications:

- `ClipsAndTacksApp`—Business process
- `ClipsAndTacksHumanCustomEAR`—Human task application
- `ClipsAndTacksServiceEAR`—Web service

We have to install all three applications in the WebSphere Process Server.

Preparation

Open the `OrderHandlingFuture1` process and select *Automatically delete the process after completion* (important for production environment, see Figure 9-19 on page 202).

Web service deployment

In the test environment the Web service is invoked with the URL:

```
http://localhost:9080/ClipsAndTacksService/services/CreditService
```

This binding is visible in the Properties view (Binding tab) when you open the Check Credit Import from the assembly diagram.

In a real process server this URL may not work and has to be changed to a real Web address, for example:

```
http://www.clipsandtacks.com/ClipsAndTacksService/services/CreditService
```

For our scenario you can leave the `localhost:9080` address, unless the server uses a different set of ports, for example, `localhost:9081`.

> **Tip:** For testing and measuring in the process server you can also reset the switch in the `CreditRating` bean:
>
> ```
> static boolean useWebservice = false;
> ```
>
> See "Implementing the credit check" on page 206 and "Invoking the Web service from an activity" on page 236. With the switch set to `false` you do not have to deploy the Web service application to the process server.

Exporting the EAR files

To create the EAR files for the process server select each enterprise application and *Export → EAR File* (context):

► Click *Browse* to enter a destination (Figure 10-1), for example,

        C:\SG247148\solution\wps\ClipsAndTacksApp.ear
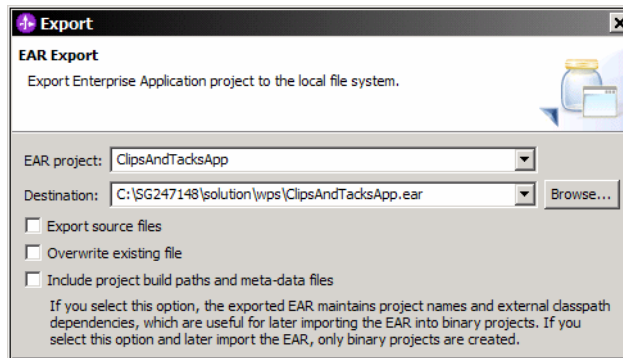
Figure 10-1   Exporting an enterprise application

► Click *Finish*.

► Repeat this for every enterprise application.

Copy the EAR file to the machine where WebSphere Process Server is installed, for example into the `installableApps` directory:

        C:\<WPS-HOME>\profiles\ProcSrv01\installableApps

Configuring Process Server

The process server must be configured with the `CLIPTACK` database that is used to store the orders. We have to define the same data source (`jdbc/cliptack`) that we defined in the test environment (see "Configure the Process Server test environment" on page 191).

We can use the administrative console as described in "Create a data source for the database" on page 194, or we can use a JACL script.

Using a JACL script to define the data source

We can automate configuration definitions by using JACL scripts. to define the data source follow these steps:

► Start the process server it it is not running.

► Open a command window in the process server `BIN` directory:

        cd <WPS-HOME>\profiles\ProcSrv01\bin

► Copy the JACL script into the same directory from:

```
SG247148\sampcode\cloudscape\datasource\DatasourceCLIPTACK.jacl
```

► Edit the `DatasourceCLIPTACK.jacl` file and change the variable `dblocation` to point to the correct directory:

```
set dblocation   "D:/WPS601/profiles/ProcSrv01"
```

► Run the JACL script using the command:

```
wsadmin.bat -f DatasourceCLIPTACK.jacl
```

Verify the data source and create the database

To verify the data source definition in the server, open the administrative console:

► Select *Resources → JDBC Providers*.

► Select *Server* and click *Apply*. You should see a new JDBC Provider named `Cloudscape JDBC Provider ClipsAndTacks`.

► Click *Cloudscape JDBC Provider ClipsAndTacks*, then click *Data sources*. You should see the `ClipsAndTacks` data source.

► Click *ClipsAndTacks* and verify the settings of the data source.

► Click *Custom properties* (right side) and verify that `createDatabase` is set to `create`.

► In the Data sources list, select the *ClipsAndTacks* data source (check box) and click *Test connection*. This action creates the database and you should receive a successful message.

We are ready now to install the application.

Install the Business Rules Manager

In Chapter 11, "Advanced facilities: Business Rules Manager and human task security" on page 261 and in Chapter 15, "Implementing the Future 2 process using WebSphere Integration Developer" on page 375 we use template-based business rules that can be modified at runtime.

To be able to modify a business rule at runtime we have to install the Business Rules Manager application:

► Open a command window in the process server `BIN` directory:

```
wsadmin.bat -f installBRManager.jacl
```

► The Business Rules Manager is installed and started. If you open the administrative console you can find the `BusinessRulesManager` application in the list of enterprise applications.

Install application in Process Server

Enterprise application are usually installed using the administrative console. We use the same steps for every EAR file to install the application in the server:

▶ Expand *Applications* and select *Install New Application*. The application installation dialog starts.

▶ Click *Browse* to locate the EAR file (Figure 10-2):

    C:\WPS601\profiles\ProcSrv01\installableApps\ClipsAndTacksApp.ear
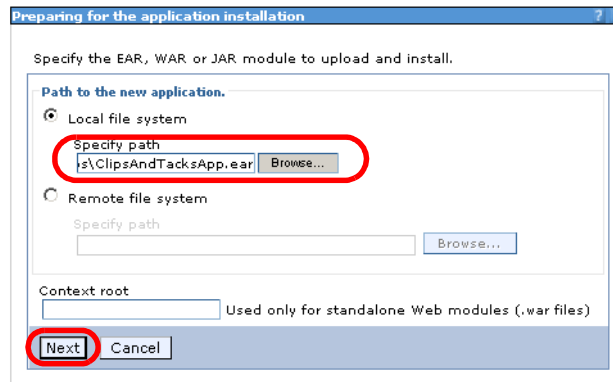
Figure 10-2   EAR installation: Location

▶ In the subsequent panels we basically accept the defaults:

  – Preparing for the application installation

  – Step 1: Select installation options (here we recommend to select *Pre-compile JSP*)

  – Step 2: Map modules to servers

  – Step 3: Provide listener bindings for message-driven beans

  – At this point you get an application resource warning for the JNDI name `sca/ClipsAndTacks/ActivationSpec`. Click *Continue*.

  – Step 4: Provide JNDI names for Beans

  – Step 5: Map resource references to resources

  – Step 6: Map virtual hosts for Web modules

  – Step 7: Ensure all unprotected 2.x methods have the correct level of protection

  – Step 8: Summary

► Click *Finish* and the application is installed. Wait for the message:

```
Application ClipsAndTacksApp successfully installed.
```

► Click *Save to Master Configuration*.

Repeat the application installation for the other enterprise applications, **ClipsAndTacksHumanCustomEAR** and **ClipsAndTacksServiceEAR**. Note that the number of steps is smaller for these applications.

Start the applications

After installation the applications can be started:

► Select *Applications → Enterprise Applications*.

► Select the three new applications and click *Start*.

Run the application (without monitor)

We are ready to run the application. However, first we have to create the CLIPTACK database for the front-end Web application.

We did not install an HTTP server so we have to submit all requests to **localhost:9080** (or the port of the server profile).

Create the database

To create the CLIPTACK database, open a browser and submit the URL:

```
http://localhost:9080/ClipsAndTacksFrontWeb/CreateDatabaseServlet
```

Then run the list servlet to verify the database content:

```
http://localhost:9080/ClipsAndTacksFrontWeb/ListServlet
```

Run the Web front-end to submit an order

Start the Web front-end with the URL:

```
http://localhost:9080/ClipsAndTacksFrontWeb
```

Login as a customer and submit an order over $750. Refer to Figure 9-43 on page 229 and subsequent figures for instructions.

Use the BPC Explorer

Start the BPC Explorer using the URL:

```
http://localhost:9080/bpc
```

Refer to "Using the human task JSPs with the BPC Explorer" on page 231.

► You should see the `ReviewOrder` task waiting for human processing (refer to Figure 9-46 on page 232 and subsequent figures).

► Select the task and click *Work on*.

► Select *Approve* and click *Complete*.

► Refresh the task list and the order is ready for shipment.

► Select the `ShipOrdertoCustomer` task and click *Work on*.

► Click *Complete*.

► Submit more orders.

Using the human task application

Refer to "Using the human task application" on page 242 for instructions.

Submit another order (over $750) using the Web front-end, then:

► Start the human task application using the URL:

```
http://localhost:9080/ClipsAndTacksHumanCustomWeb
```

► Click *Review Orders*, then approve the order (refer to Figure 9-53 on page 242 and subsequent figures).

► Click *Shipments* and ship the order.

► Submit an order under $750 to verify that automatic approval works. If you installed the Web service, it will be invoked.

Using the Business Rules Manager

To start the Business Rules Manager on the Process Server open a browser and enter this URL:

```
http://localhost:9080/br/webclient
```

Accessing the server log

The log of the process server can be found in:

```
<WPS-HOME>\profiles\ProcSrv01\logs\SystemOut.log
```

Changing the application

If you make changes to the application in Integration Developer, you have to reinstall the new EAR file after uninstalling the old application.

Uninstalling an application

Before an old application can be uninstalled, we must make sure that there are no active instances. First we make sure that no new instances can be created:

► Open the administrative console.

► Select *Applications → Enterprise Applications*, then select the enterprise application (`ClipsAndTacksApp`).

► Under Related items, click *EJB Modules*, then select the EJB JAR file.

► Under Additional Properties, select *Business Processes*.

► Select all processes and click *Stop*.

► Under Additional Properties, select *Human Tasks*.

► Select all task templates and click *Stop*.

Next we use the BPC Explorer to terminate active task instances:

► Under task instances select *Administered By Me*.
► Select all the tasks and click *Terminate*.

Finally we use the BPC Explorer to delete process instances:

► Under process instances select *Administered By Me*.
► Select all the processes and click *Delete*.

> **Note:** This action is only necessary when you set the flag to keep processes when they are finished (Figure 9-19 on page 202).
>
> Deleting the instances is not necessary in the Integration Developer test environment where active instances are removed automatically.

Now the application can be uninstalled:

► In the administrative console, select *Applications → Enterprise Applications*.
► Select the application (`ClipsAndTacksApp`) and click *Stop*.
► Select the application (`ClipsAndTacksApp`) and click *Uninstall*.
► Save the configuration changes.

Finally you can install the new EAR file using the installation process.

Summary

In this chapter we described how to deploy an applicaiton from Integration Developer to a real Process Server.

After configuring the server with the necessary data source we can install the business process application together with the supporting code.

11

Advanced facilities: Business Rules Manager and human task security

In this chapter we work with advanced facilities of the WebSphere Process Server.

First we work with the Business Rules Manager, which enables us to modify a business rule at runtime without having to reinstall an application.

Then we implement a simple security setup to control which users can invoke the human tasks to handle the orders.

    261

Business Rules Manager

The Business Rules Manager is a Web-based tool that assists the business analyst and other users in modifying business rule values. This optional tool of the process server is installed after the initial installation of the server.

The Business Rules Manager application can either be installed using the WebSphere administrative console or through a JACL script. You can use this Web-based client to browse and edit business rules.

Installing the Business Rules Manager

For testing the changing of a business rule we install the Business Rules Manager in Integration Developer:

► Open a command window in the `<WID-HOME>\runtimes\bi_v6\bin` directory.

► Run the JACL script to install the Business Rules Manager:

```
wsadmin.bat -f installBRManager.jacl
```

► The Business Rules Manager is installed and started. If you open the administrative console you can find the `BusinessRulesManager` application in the list of enterprise applications.

Defining a business rule based on a template

To make use of the Business Rules Manager we must have at last one business rule that is based on a template. The three rules that we defined in "Implementing a business rule" on page 204 in the rule set `AutomaticApprovalFuture1` are all fixed rules (Figure 11-1).
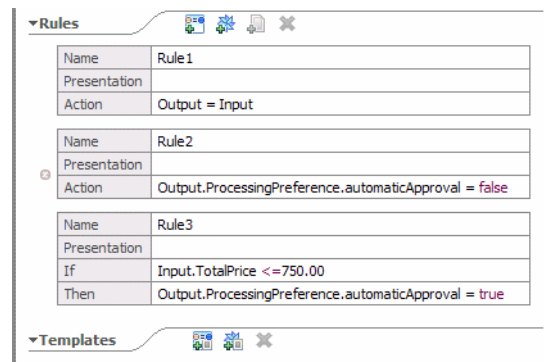
Figure 11-1   Current business rule

We change the third rule to a template rule that can be changed at runtime using the Business Rules Manager.

► Select the third rule (Rule 3) and *Delete Rule* (context).

► In the Templates section click the *Add If then Template* icon 🖼. A template is added. Overtype the name with `Total Price Template` (Figure 11-2).

Name	Total Price Template			
Presentation				
Parameters	Name	Type	Constraint	➕
If	*Condition*			
Then	*Action*			

Figure 11-2   Business rule template

► Click the ➕ icon to add a parameter (Figure 11-3):
  – Change the parameter name to `totalPrice`.
  – Click *type* and select *double*.
  – Click *None* (under Constraint) and select *Range*. An expression is added.
  – Click *Enter Expression* and select *Exclusive Range*.
  – Click on each number and enter `700.00` and `800.00`.

Name	Total Price Template			
Presentation				
Parameters	Name	Type	Constraint	➕
	totalPrice	double	Range	> 700.00 and < 800.00
If	*Condition*			
Then	*Action*			

Figure 11-3   Business rule template: Parameter

► Click *Condition* (If) and select *Input.Totalprice <= totalPrice*.

► Click *Action* (Then) and select *Output.ProcessingPreference.automaticApproval = true*.

► For the Presentation field enter:

`The maximum order value for automatic approval is`

Click after the text and an arrow shows up. Click the arrow and select *totalPrice*.

The final template rule is shown in Figure 11-4.

Name	Total Price Template			
Presentation	The maximum order value for automatic approval is totalPrice			
Parameters	Name	Type	Constraint	✚
	totalPrice	double	Range	> 700.00 and < 800.00
If	Input.TotalPrice <= totalPrice			
Then	Output.ProcessingPreference.automaticApproval = true			
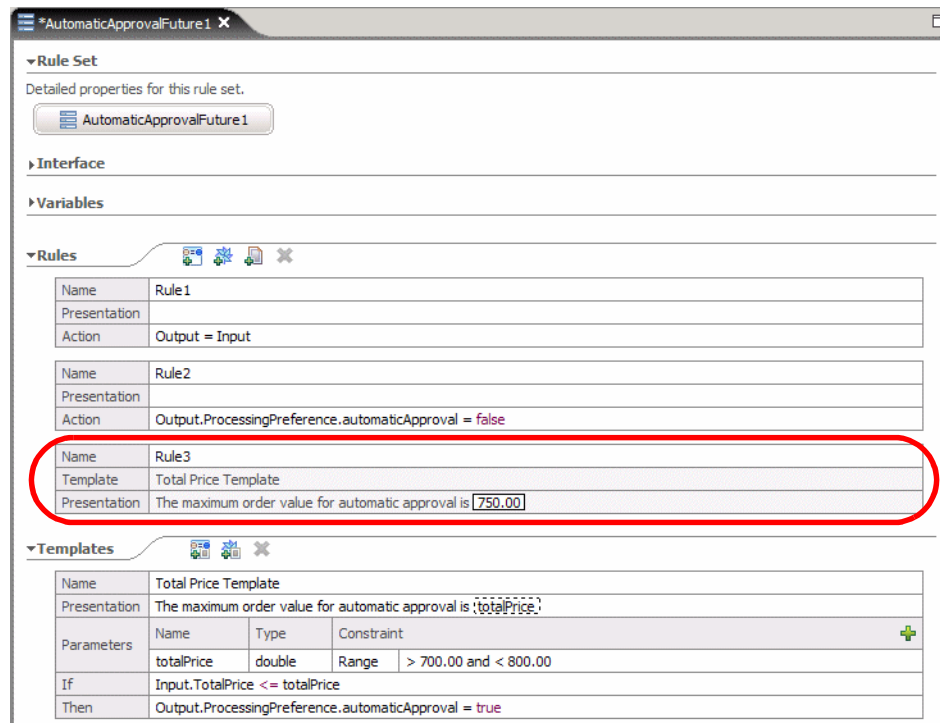
Figure 11-4　Business rule template: Complete

▶ In the Rules section click the A*dd Template Rule* icon . Select *Total Price Template*. A third rule is added (Figure 11-5).

Name	Rule3
Template	Total Price Template
Presentation	The maximum order value for automatic approval is *Enter Value*

Figure 11-5　New business rule using a template

▶ Overtype `Enter Value` with `750.00`. We have recreated the original business rule using a template (Figure 11-6).

Figure 11-6　Final business rule using a template

At this point we can test if the template rule works:

► Submit an order with a value below 750 and it should go directly to shipping.

► Submit an order with a value between 750 and 800 and it should go to the review activity.

► Use the human task application to process the orders.

Using the Business Rules Manager

Select the server in the Servers view and *Launch → Business Rules Manager*:

► The Business Rules Manager opens. Expand the sections (Figure 11-7).
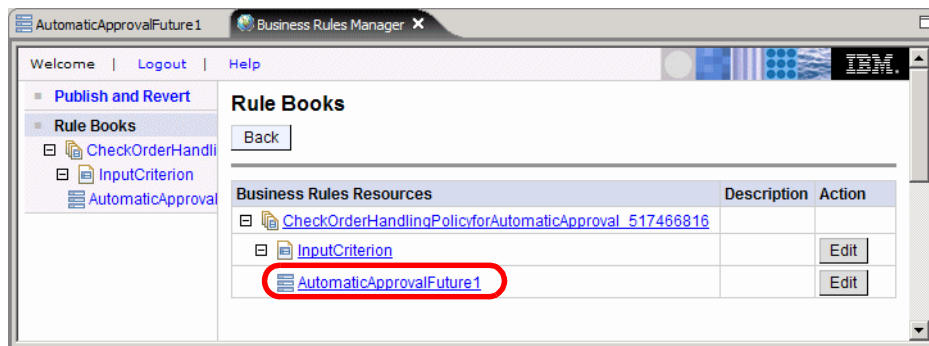
Figure 11-7   Business Rules Manager: Rule books

► Select *AutomaticApprovalFuture1* under Rule Books (Figure 11-8).
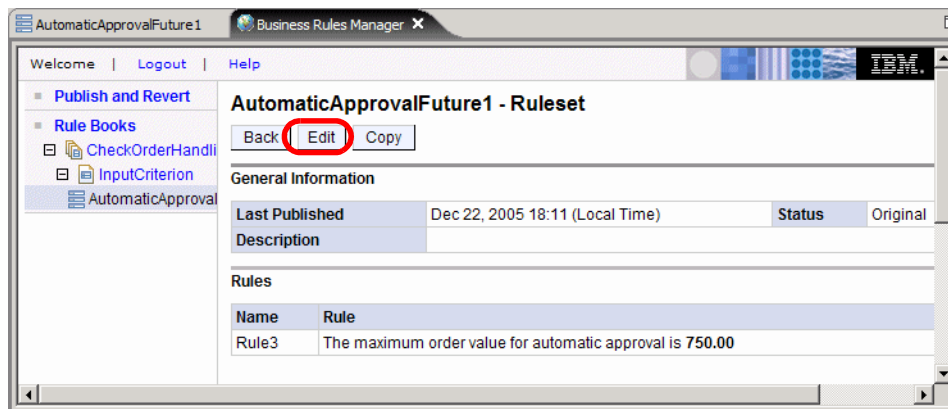
Figure 11-8   Business Rules Manager: Ruleset

► Click *Edit* (Figure 11-9).
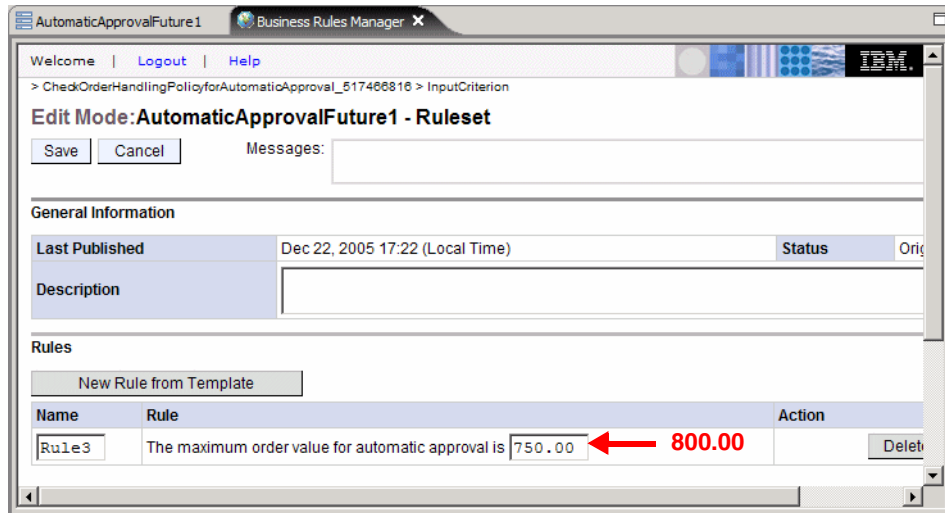
Figure 11-9   Changing a business rule at runtime

▶ Change the value to **800**, then click *Save*. Notice the message:

```
AutomaticApprovalFuture1" has been temporarily saved.
You may publish the changes from the "Publish and Revert" page.
```

▶ Select *Publish and Revert* (left side), then click *Publish* (Figure 11-10).
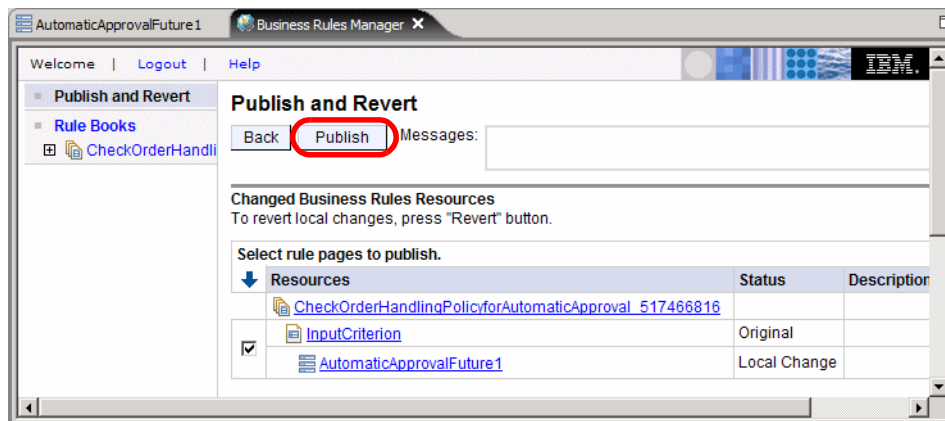
Figure 11-10   Publishing a business rule

The updated business rule is now active. Submit an order with a total price between 750 and 800. This order goes directly to shipping! This proves that the updated business rule is active.

Implementing security for human tasks

In this section we describe how to setup a simple security environment to control who can use the human task functions of reviewing orders, shipping orders, and canceling orders.

Authorization and authentication is a mandatory element in any Web application. WebSphere Process Server provides three different approaches to secure your server and application:

► **Local OS** (Operating System)—Local OS user registries are the user registries that are part of the underlying operating system that application servers are running on. Local OS user registries are always available, and are typically the simplest way to configure users because you have already set them up to log into your machine.

► **LDAP** (Lightweight Directory Access Protocol)—When working with multiple servers, such as clustering and high availability, and with multiple machines, you will probably run into limitations using a single machine's local operating system user registry, and have to move up to a directory server using LDAP for storing user and group information.

► **Custom user registry**—A third option for user registries is a user registry that is not one of the operating system user registries supported by WebSphere and is not an LDAP-compliant user registry. Connecting to these types of user registries is an advanced but supported option, and requires a software development effort to complete.

We will explore the custom user registry option to implement security for the human task application. In "Using an LDAP server" on page 280, we will configure an LDAP server to implement the same security constraints.

Define groups and users

To implement security we need groups and users. For our application we require an order manager, a shipper, and an administrator. Therefore we define the groups and users shown in Table 11-1. The user ID `wid` will be used to start and stop the server.

Table 11-1   Groups and users for human task security

Group	Users
administrator	wid, ueli
ordermanager	andre, ueli
shipper	russ, larissa,ueli

A custom registry requires a Java program that implements `UserRegistry`, which defines methods such as `initialize`, `checkPassword`, `getUsers`, `getGroups`, `getUsersForGroup`, `getUniqueGroupId`, and so forth.

Implementation

Our implementation uses two property files and a Java program provided in:

```
SG247148\sampcode\wid\humantaskSecurity
```

► **groups.props**—Property file with group definitions:

```
# Format:
# name:gid:users:display name
# where name   = groupId of the group
#       gid    = uniqueId of the group
#       users  = list of all the userIds that the group contains
#       display name = (optional) display name for the group
administrator:001:wid,ueli:Administrative group
ordermanager:002:andre,ueli:Order Manager group
shipper:003:russ,larissa,ueli:Shipper group
```

► **users.props**—Property file with user definitions

```
# Format:
# name:passwd:uid:gids:display name
# where name   = userId/userName of the user
#       passwd = password of the user
#       uid    = uniqueId of the user
#       gid    = groupIds of the groups that the user belongs to
#       display name = (optional) display name for the user
wid:wid:101:001:WID Admin
ueli:ueli1:102:001,002,003:Ueli Wahli
andre:andre1:103:002:Andre Venancio
russ:russ1:104:003:Russell Scher
larissa:larissa1:105:003:Larissa Leybovich
```

► **FileRegistrySample**—Java program that contains the code to read the property files and implement all the required methods.

Placement of the user registry files

We put the property files into the directory pointed to by the WebSphere Application server variable `${USER_INSTALL_ROOT}`. In Integration Developer this points to <WID-HOME>/pf/wps:

```
<WID-HOME>/pf/wps/UserCustomRegistry/groups.props
```

The `FileRegistrySample` program and subclass `RegExpSample` must be in the class path of the server. We put the program into:

```
<WID-HOME>/runtimes/bi_v6/lib/ext
```

Specify security in the Process Server

To enable security in the Process Server is a sequence of steps:

► Define security using a user registry
► Enable global security
► Install the file transfer utility for security (to see the console)
► Specify security in Integration Developer
► Restart the server

Define security using a user registry

With the server started, open the administrative console (select the server and *Run administrative console*):

► Login with any user ID.

► Expand *Security → Global Security* (Figure 11-11).
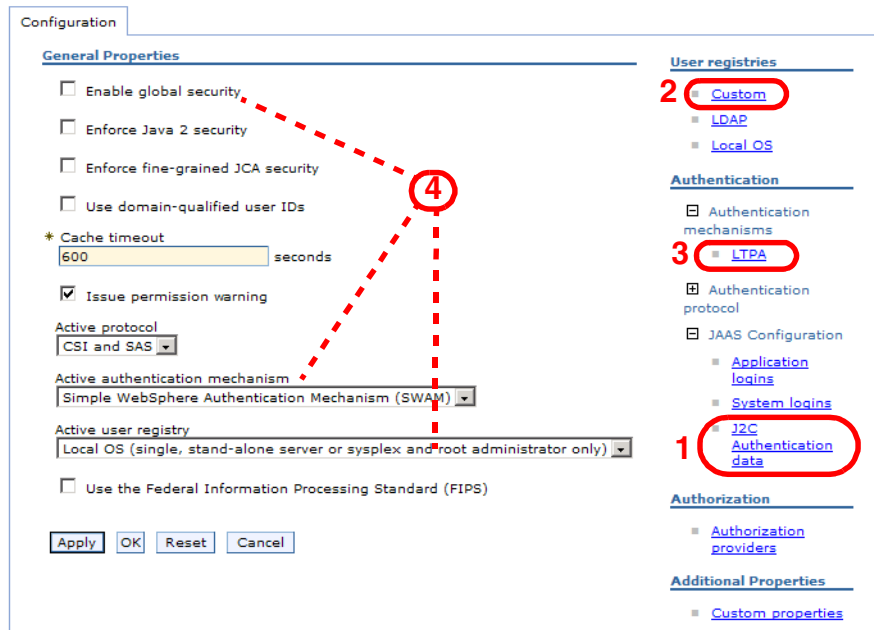
Figure 11-11   Global security: initial

► Under *Authentication* (right side) expand *JAAS Configuration*, and select *J2C Authentication data*. You should see at least 3 alias entries with a user ID of `wid` (Figure 11-12).

Each of these entries uses the user ID `wid` and the password `wid` (encrypted).

Global security > J2EE Connector Architecture (J2C) authentication data entries
Specifies a list of user IDs and passwords for Java 2 connector security to use.

⊞ Preferences

Select	Alias ↕	User ID ↕	Description ↕
☐	SCA_Auth_Alias	wid	This is the alias used by SCA to login to a secured SIBus
☐	widCell/BPEAuthDataAliasJMS_widNode_server1	wid	Authorization Alias for Process Choreographer
☐	widCell/widNode/server1/EventAuthDataAliasCloudScape	none	CloudScape authentication alias for the Common Event Infrastructure
☐	widNode/CommonEventInfrastructureJMSAuthAlias	wid	Authentication alias for the Common Event Infrastructure JMS Topics and Queues

Total 4

Figure 11-12   J2C authentication data entries

Note that we defined the user ID `wid` in `users.props` with the password `wid`. If you use a different password you must update the three aliases with your password.

▶ Back in Global Security select *Custom* under *User registries* (right side):

– Enter `wid` as user ID and password (Figure 11-13).
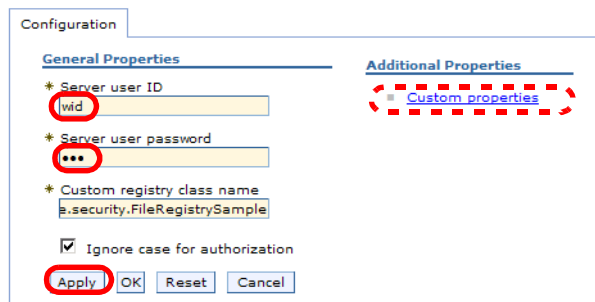
Figure 11-13   Custom registry definition

– The custom registry class name points by default to our program name `com.ibm.websphere.security.FileRegistrySample`.

– Select *Ignore case for authorization* (optional).

– Click *Apply*.

– Click *Custom properties*. Add two custom properties named `groupsFile` and `usersFile`, matching the code in the `FileRegistrySample` program. The two properties point to our files (Figure 11-14).
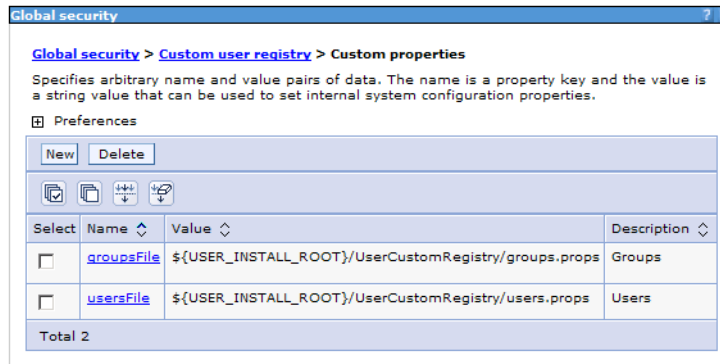
Figure 11-14   Custom registry properties

► Back in Global Security select *Authentication mechanism → LTPA* under *Authentication* (right side):

– Enter the password `wid` twice and click *Apply* (Figure 11-15).
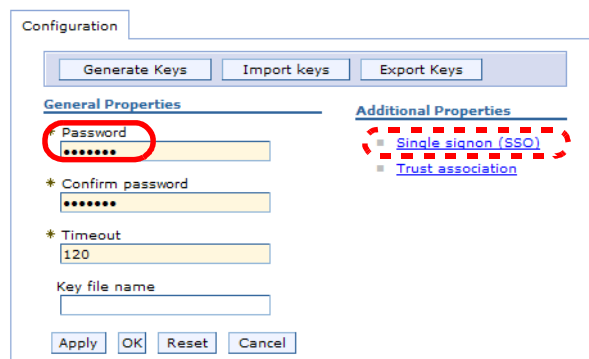
Figure 11-15   Authentication using LTPA

– Select *Single signon (SSO)*, then select *Enabled* (it should be selected by default) and click *OK*.

Enable global security

Next we run the server with security enabled:

► Select *Security → Global security* (Figure 11-16).

Chapter 11. Advanced facilities: Business Rules Manager and human task security   **271**

– Select *Enable global security*, and deselect *Enforce Java 2 security*.

– For active authentication mechanism select *Lightweight Third Party Authentication (LTPA)*.

– For active user registry select *Custom user registry*.

– Click *OK*.

Figure 11-16   Enable global security

Save the configuration

Click *Save* to save the configuration changes. then click *Logout* in the administrative console.

Install the file transfer utility for security

When security is enabled a special file transfer utility that uses security must be used by Integration Developer to retrieve the console. To install this utility we run a JACL script provided by the server:

► Open a command window in the directory:

```
<WID-HOME>/runtimes/bi_v6/bin
```

► Run the `wsadmin` command and watch for a successful installation:

```
wsadmin -profile redeployFileTransfer.jacl -lang jacl
        -c "fileTransferAuthenticationOn widCell widNode server1"
WASX7209I: Connected to process "server1" on node widNode using SOAP ...;
 The type of process is: UnManagedProcess
Uninstall filetransfer -cell widCell -node widNode -server server1
```

```
ADMA5017I: Uninstallation of filetransfer started.
ADMA5011I: The cleanup ... for application filetransfer is complete.
ADMA5106I: Application filetransfer uninstalled successfully.
Install C:\WID601\runtimes\bi_v6/systemApps/filetransferSecured.ear -cell
widCell -node widNode -server server1 -appname filetransfer
-usedefaultbindings -nocreateMBeansForResources
ADMA5016I: Installation of filetransfer started.
ADMA5011I: The cleanup ... for application filetransfer is complete.
ADMA5013I: Application filetransfer installed successfully.
```

Specify security in Integration Developer

Because the server is started and stopped from Integration Developer, we have to specify user ID and password in the server configuration:

▶ Select the server in the Servers view and *Open* (Figure 11-17).
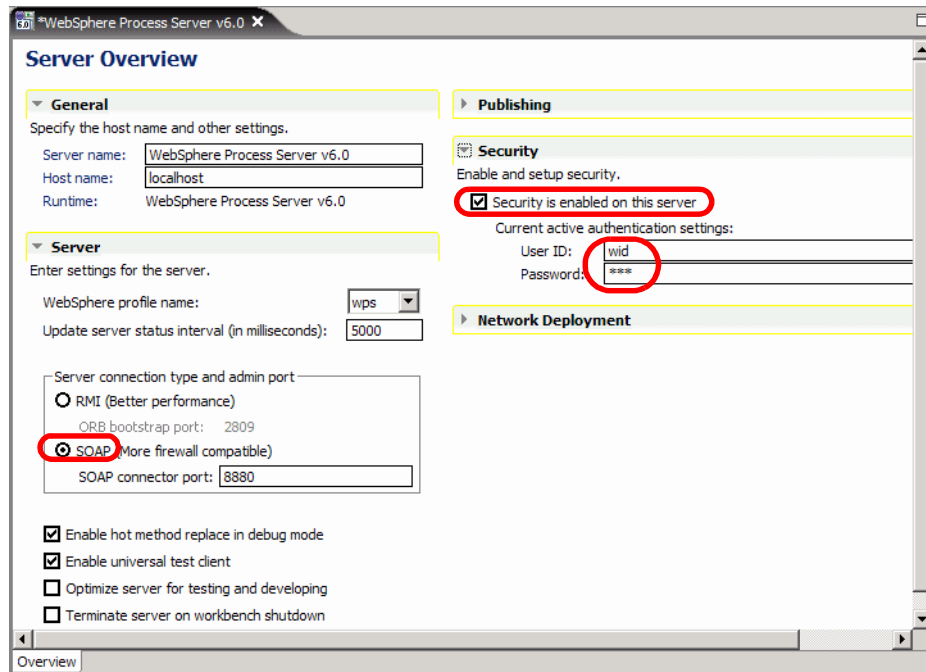
Figure 11-17   Server configuration in Integration Developer

– Select *SOAP* as the connection type, and enter port `8880`.

– Expand Security and select *Security is enabled on this server*. Enter `wid` as user ID and password. This must match what you specified in the server custom registry.

– Save and close the configuration.

Restart the server

Stop and start the server. You should see the console when the server is ready.

Start the administrative console. You have to accept a certificate and then you have to login with the specified user ID (`wid`) and password.

Using the groups for human tasks

To enable security for the human tasks we have to configure each human task with the correct group:

▶ In the Business Integration view expand *ClipsAndTacks → Business Logic → Human Tasks*.

▶ Open the `ReviewOrder` task (Figure 11-18):

  – For receiver settings, click the *Potential Instance Creator* icon 🔧 to add an entry to the list.

  – Select *Potential Instance Creator* and verify in the Properties view that the verb is set to `Everybody`.
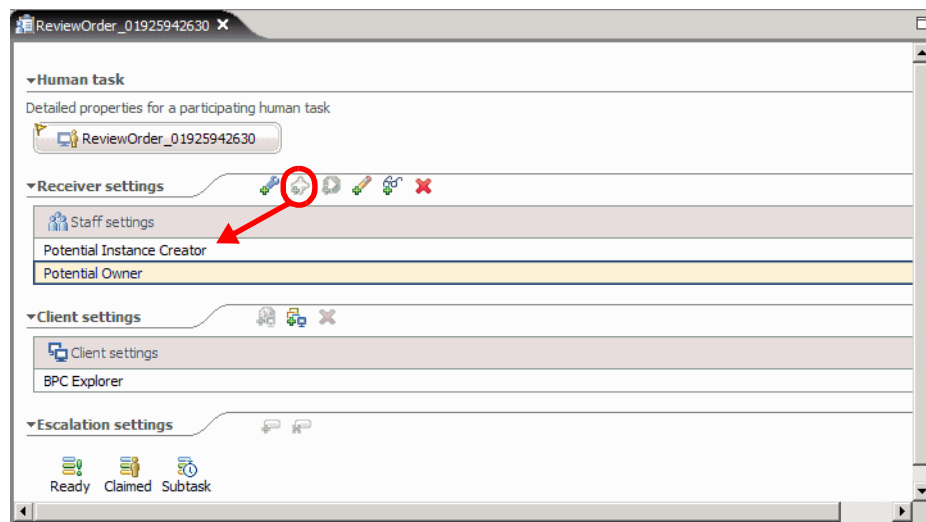
Figure 11-18   Human task configuration

  – Select *Potential Owner* and in the Properties view select *Group Members* as the verb (Figure 11-19).

    For parameters, enter **ordermanager** as group name, select *false* for subgroups, and enter **administrator** for alternative group name 1.
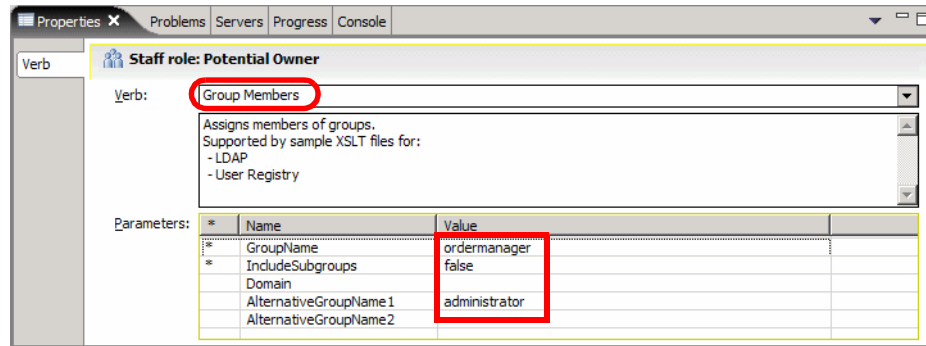
Figure 11-19   Human task properties

Only an order manager or the administrator will be able to claim (work on) a review order task.

▶ Open the `CancelOrderandSendNotification` task and perform the same configuration changes (*Group Members*, `ordermanager`, *false*, and `administrator`). Do not forget to add the Potential Instance Creator.

▶ Open the `ShipOrdertoCustomer` task and perform the same configuration changes:

– Add the Potential Instance Creator with `Everybody` as verb, and set the Potential Owner to *Group Members*.

– However, enter **shipper** as group name, select *false*, and enter `administrator` as alternative group name.

Only a user from the `shipper` group, or an administrator, can claim a shipping task.

Web service invocation

The invocation of the Web service from the `CreditRating` bean may fail after editing the human tasks.

> **Important:** This is an error in the product. The `wsdl` directory for the Web service is deleted from the EJB project `ClipsAndTacksEJB/META-INF/wsdl`.
>
> See "WSDL file deleted from the EJB project" on page 249 for instructions on how to get the EJB project generated correctly.

You could also open the `CreditRating` bean (in `ClipsAndTacks`) and change the flag so that the Web service is not invoked:

```
static boolean useWebservice = false;
```

Restart the application

Select the server in the Servers view and *Restart Project → ClipsAndTacksApp*.

Testing human task security with the BPC Explorer

Redeploy the application to the server. Then submit orders using the Web front-end.

Once you have orders waiting for human interaction, start the BPC Explorer by selecting the server and *Launch → BPC Explorer*:

► Accept the security certificate.

► You are prompted to login (Figure 11-20).
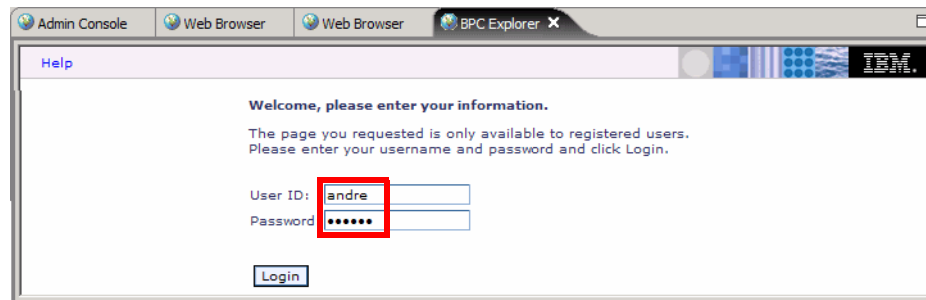
Figure 11-20   BPC Explorer: login

► Login as order manager (`andre`). You will only see review (`ReviewOrder`) and cancel (`CancelOrderandSendNotification`) tasks. You can claim and complete the tasks by clicking *Work on*.

► Click *Logout*, then login as a shipper (`russ`). You will only see shipping (`ShipOrdertoCustomer`) tasks, which you can claim and complete.

► Click *Logout*, then login as a administrator (`ueli`). You can see all human tasks waiting for interaction.

Using the customized human task Web application

The customized human task application does show the human tasks after configuring security. However, when you try to claim a task for processing you get this error:

```
Error: Exception CWTKA0068E: User 'UNAUTHENTICATED' is not allowed to
perform the requested action 'claim()' on task ...........
```

We have to implement security in the Web application. See "Implementing a human task application" on page 237 for the description of the application.

Implementing security in the human task Web application

First we have to provide a login page so that users can authenticate using the user IDs provided:

▶ `login.jsp`—Prompts the user for user ID and password. This is a standard facility of J2EE and uses the `j_security_check` servlet provided by the application server.

▶ `loginError.html`—This error page is displayed when the login fails.

Next we have to configure security in the Web application deployment descriptor:

▶ Open the deployment descriptor of the `ClipsAndTacksHumanCustomWeb` project and select the *Security* tab. Figure 11-21 shows the completed definitions.
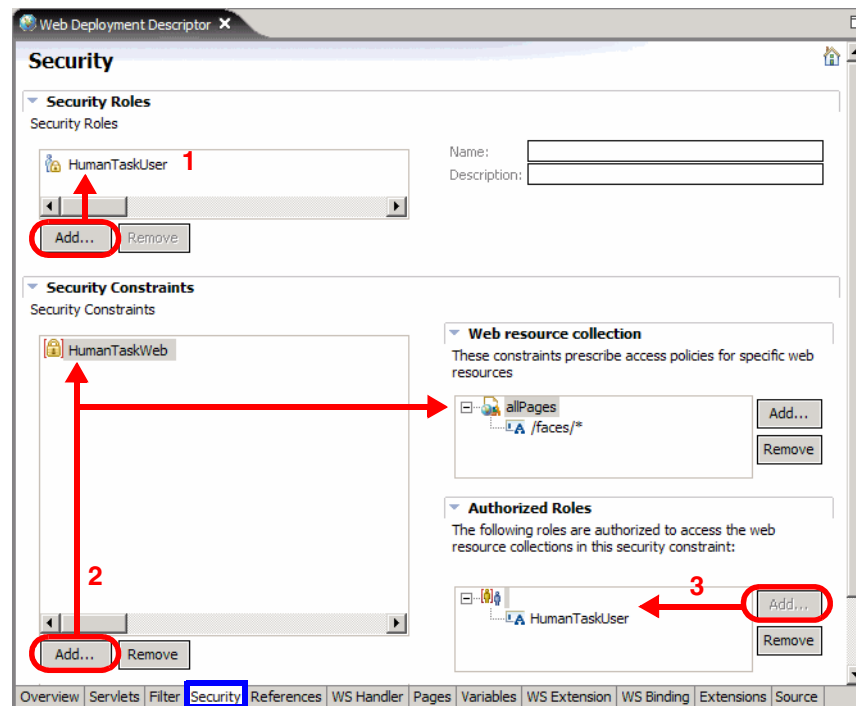
Figure 11-21   Web deployment descriptor with security

1. Click *Add* to define the `HumanTaskUser` security role.

2. Click *Add* to define the `HumanTaskWeb` security constraint. In the dialog enter `allPages` as resource name and `/faces/*` as ULR pattern. This protects all the pages that deal with the human tasks.

3. Select the `allPages` resource and click *Add* for authorized roles and select the `HumanTaskUser` role.

► Select the *Pages* tab. For login authentication method select *FORM*. Then select the `login.jsp` and the `loginError.html` page for the login and error pages (Figure 11-22).

You can also delete all the welcome pages except for `index.html`.
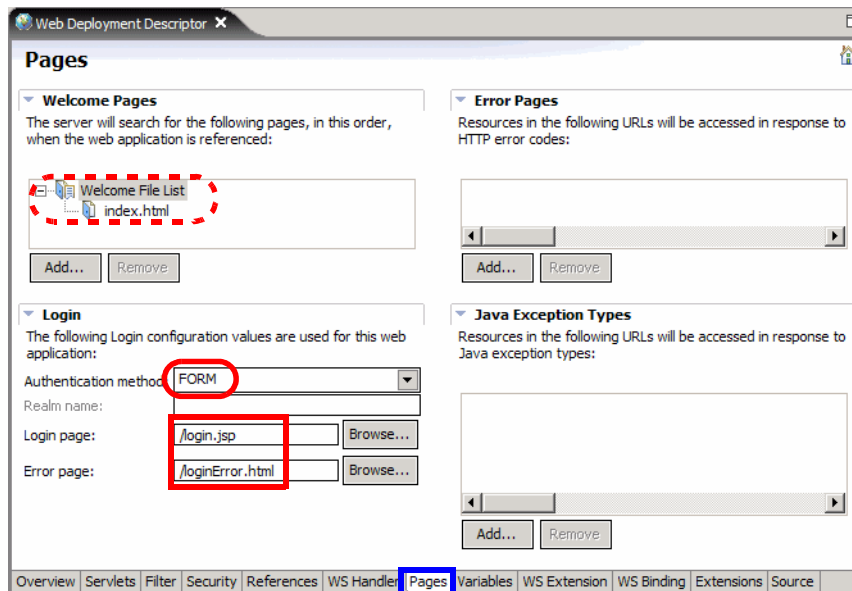
Figure 11-22   Web deployment descriptor login pages

► Save and close the deployment descriptor.

Now we have to update the enterprise application deployment descriptor:

► Open the deployment descriptor of the `ClipsAndTacksHumanCustomEAR` application and select the *Security* tab.

► Click *Gather*, then select the `HumanTaskUser` and select *All authenticated users* (Figure 11-23). Save and close the deployment descriptor.
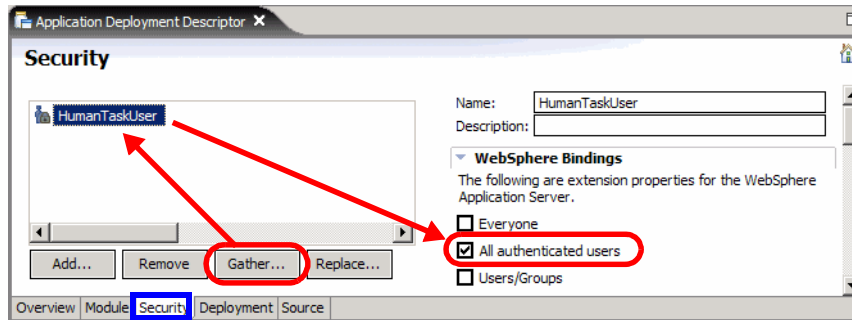
Figure 11-23   Enterprise application deployment descriptor security

Running the human task application with security

Restart the process server. The human task application may not work without restarting the server.

After submitting some orders, open the human task application:

```
https://localhost:9443/ClipsAndTacksHumanCustomWeb/
```

► Accept the security certificate.

► You are prompted to login. Login as order manager `andre` (Figure 11-24).
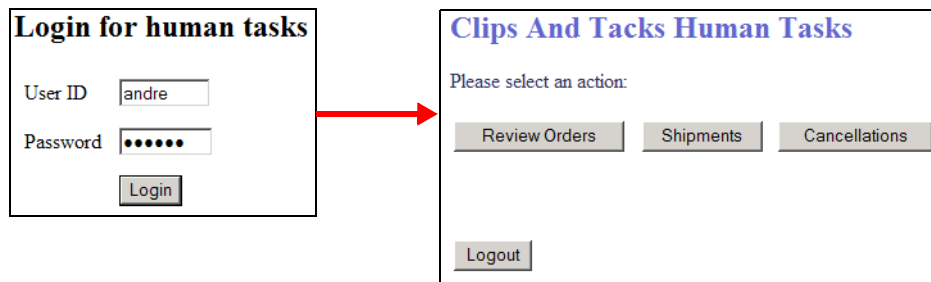
Figure 11-24   Human task application login

► Click *Review Orders* and you can process orders waiting for review. If you click *Shipments* you always get the message that no tasks are waiting. You can also process cancellations.

► Click *Logout* and login as a shipper (`russ`) and you can process shipments.

► Login as administrator (`ueli`) and you can process all tasks.

We have now implemented security for the human tasks and only authorized users can process the orders waiting for interaction.

Deployment of the application with the custom user registry

To run the application with security in the real Process Server, you have to configure the server in the same way as shown in "Specify security in the Process Server" on page 269.

You also have to place the properties files into the `${USER_INSTALL_ROOT}` directory and the Java program into the `${WAS_INSTALL_ROOT}/lib/ext` directory.

Export the business process application as a new EAR file and install it into the server after stopping and uninstalling the running application. Also reinstall the custom human task application.

Using an LDAP server

Instead of using a custom user registry we can also use an LDAP server. We had a Lotus® Domino® LDAP server available at:

```
ldap://domholsfx.burlingame.ibm.com
```

Groups and users
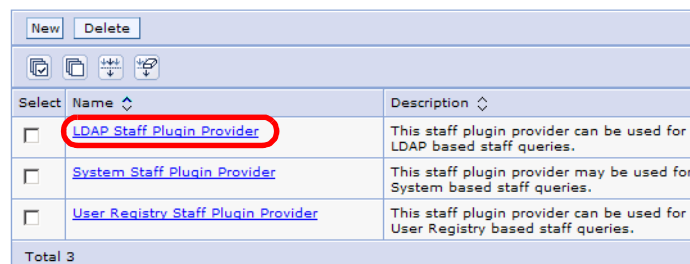
We defined the same groups and users in the LDAP server as described in "Define groups and users" on page 267.

Configuring the server for LDAP

To run the server with LDAP we have to configure global security and the human task plugin for LDAP. Open the administrative console and perform these steps:

Staff plug-in provider

►   Select *Resources → Staff plug-in provider*.

►   Click *LDAP Staff Plugin Provider* (Figure 11-25).

Figure 11-25   LDAP staff plugin provider

►   Under Additional Properties, click *Custom properties*.

▶ Select the property *ProviderURL* and change the value of the hostname from `localhost` to your LDAP server (Figure 11-26):

```
From: ldap://localhost:389
To:   ldap://domholsfx.burlingame.ibm.com:389
```
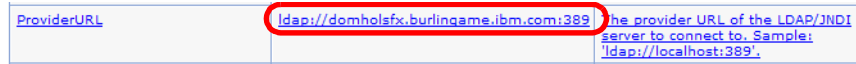

Figure 11-26   LDAP provider URL

▶ Click *OK*.

Enable global security

▶ Select *Security → Global security*.

▶ Under User registries select *LDAP*.

▶ Enter the values as shown in Figure 11-27.
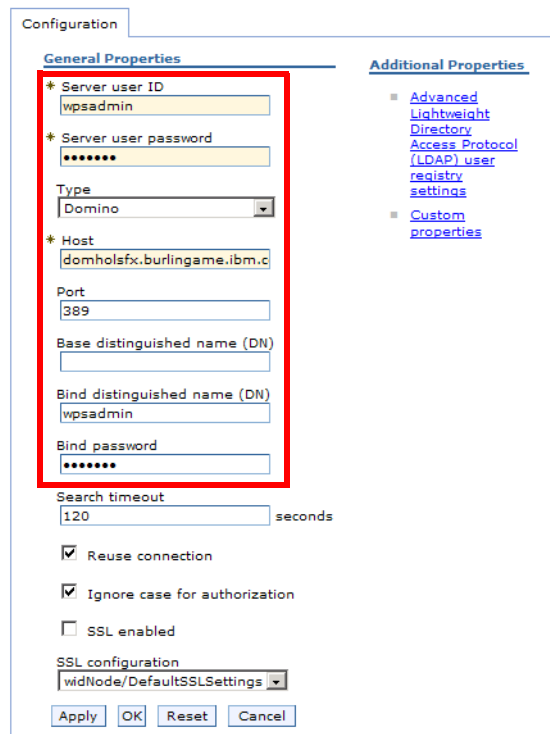
Figure 11-27   LDAP security configuration

▶ Enter the LDAP server user ID and password and the same values also for bind.

► Select the LDAP type (we used a Domino server).

► Enter the hostname.

► Click *OK*.

► Back in Global security select *Lightweight Directory Access Protocol (LDAP) user registry* (Figure 11-28).

► Click *OK*.

> **Tip:** If you get an error message about invalid user ID, deselect *Enable global security*, click *OK*, then select *Enable global security* (deselect *Enforce Java 2 security*) and click *OK*.
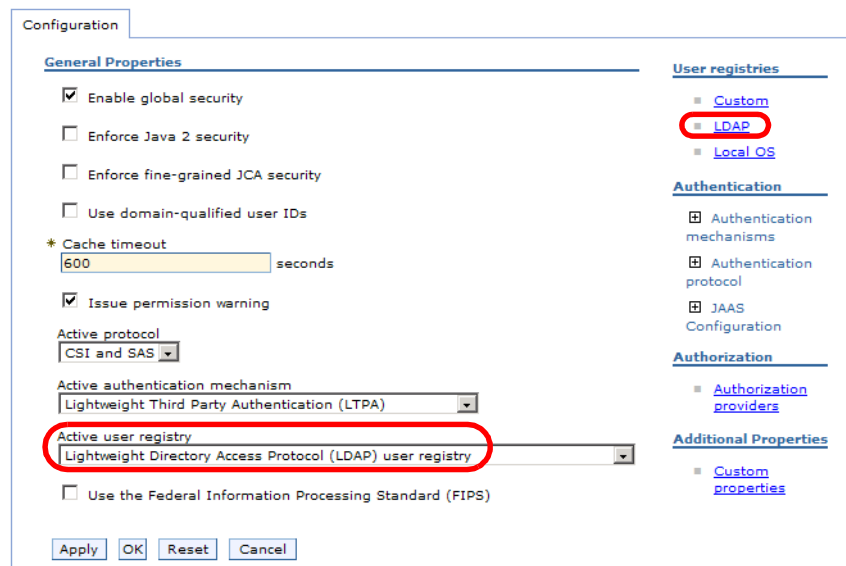
Figure 11-28   Global security using LDAP

► Save the configuration.

Stop and start the server

Before you start the server with LDAP security you also have to set the new server user ID and password in the server configuration (see "Specify security in Integration Developer" on page 273):

► Stop the server.

► Open the server configuration (see Figure 11-17 on page 273) and set the user ID and password to the same values as in Figure 11-27 on page 281.

▶   Start the server.

> **Tip:** This sequence is important. Always stop the server with the old security credentials, and start the server with the new user ID, otherwise you may not see the Console output.

Testing LDAP security

You can now submit orders, use the BPC Explorer (see "Testing human task security with the BPC Explorer" on page 276), and use the custom human task application (see "Using the customized human task Web application" on page 276).

Deployment of the application with LDAP

To run the application with LDAP security in the real Process Server, you have to configure the server in the same way as shown in "Configuring the server for LDAP" on page 280.

Export the business process application as a new EAR file and install it into the server after stopping and uninstalling the running application. Also reinstall the custom human task application with an exported EAR file with security configured.

Removing security

Once human task security is defined in the human tasks of the process, you cannot run the server without security. If you try to submit an order to the server running without security, you get error messages that work items cannot be created. You have to reconfigure the human tasks again and select *Everybody* as potential owner.

Summary

In this chapter we described how to work with dynamic business rules that can be modified at runtime using the Business Rules Manager.

We also described how to implement security in the Process Server so that we can control which users can process orders waiting for human interaction.

Part 4

Monitoring the application

In this part we describe how to use the WebSphere Business Monitor to monitor the execution of the business process application.

After introducing the Monitor, we describe in detail how to measure the ClipsAndTacks application and view the results in the Monitor dashboard.

Based on the monitoring results we go back to the Modeler to make changes in the model. We implement the changed model in Integration Developer and deploy it to the Process Server, where we monitor the new application once more.

12

Introducing WebSphere Business Monitor Version 6

This chapter introduces WebSphere Business Monitor, Version 6, into the end-to-end scenario for the ClipsAndTacks scenario.

We describe the architecture of the Monitor product and then configure the Monitor Server and Monitor Dashboard for the ClipsAndTacks application.

287

Overview

The WebSphere Business Monitor Version 6.0 is a Web-based client/server application that measures business performance, monitors processes and workflow, and reports on business operations. The information captured can help you identify problems, correct faults, and change processes to achieve a more efficient business.

WebSphere Business Monitor 6.0 monitors business processes at runtime by monitoring event-emitting runtime engines. *Only applications that are running on WebSphere Process Server Version 6.0.1 are currently supported.*

WebSphere Business Monitor calculates key performance indicators (KPIs) and metrics using collected events, based on a given model. The calculated KPIs and metrics values are represented on a number of views based on business needs. WebSphere Business Monitor notifies users of incidents requiring their attention and can also perform corrective actions to avoid failures. It supports different notification methods (alert, e-mail, cell phone, pager, and service invocation) against situations and actions associated with defined conditions.

WebSphere Business Monitor depends on business measures models for its monitoring procedure. These models are created in the Business Measures editor where you can specify the measuring points and event filters, define the measurements, their correlations, and sources of the business data. When the business measures model is complete, you can export it to WebSphere Business Monitor. It then recognizes the model to be monitored and the measurements to be captured from the incoming events.

You use the Business Measures editor to open the process models created in WebSphere Business Modeler and to create business measures models. For each business measures model, you can define the metrics and KPIs, event emission points, event filters, event composition rules, and situations that will trigger specific actions at runtime.

Key features of WebSphere Business Monitor 6:

► Captures a large amount of data through events from operation activities and transforms it into metric and KPI values

► Extracts the measurements variables from business data

► Displays the measurements values in useful views

► Provides analysis and reports

► Performs corrective actions

► Notifies users to take action to prevent failures

This chapter includes the following sections:

- ► Overview of the Modeler to Monitor closed loop cycle
- ► WebSphere Business Monitor architecture overview
- ► Business measures model deployment to Monitor Server
- ► Configuring alerts for the adaptive action manager
- ► Configuration of the Monitor dashboards
- ► Overview of using the Monitor dashboard client

> **Note:** Software installation (and pre-requisites) and deployment of Monitor 6 required prior to configuration and model deployment is discussed in detail in Appendix B, "Installation of WebSphere Business Monitor" on page 417.

Overview of the Modeler to Monitor closed loop cycle

In this section we describe the overview of taking the business process model from WebSphere Business Modeler V6 to WebSphere Business Monitor V6. In the following section we demonstrate this process with the Clips And Tacks scenario model and deploy it into the WebSphere Business Monitor Server and configure the WebSphere Business Monitor dashboards for viewing and analyzing the runtime data.

Figure 12-1 shows the overall closed loop process from Modeler V6 to Monitor V6 as well as the supporting tooling and runtime engine for the BPEL process:

- ► In Steps 1 and 2, where the process model is built using Modeler V6, as described in the previous chapters of this redbook. The model was developed, simulations were performed based on resource allocations and estimated task durations.
- ► Step 3a shows the export of the process model into Integration Developer, where the application is completed and deployed to Process Server.
- ► In Step 3b the business measures model is exported from Modeler as a Monitor artifact to be deployed to the Monitor Server.
- ► Step 4 shows the use of the Business Monitor Dashboards to view, analyze and perform reporting and analytics on the runtime process data of the business process as well as the specific business measure data and KPI's from the Business Monitor.
- ► In Step 5 the closed loop functionality is realized by feeding the actual runtime data back from Monitor to Modeler. The Monitor administrative console provides a dialog for exporting an XML file of the actual values. This includes extracting the data from the historical data store.
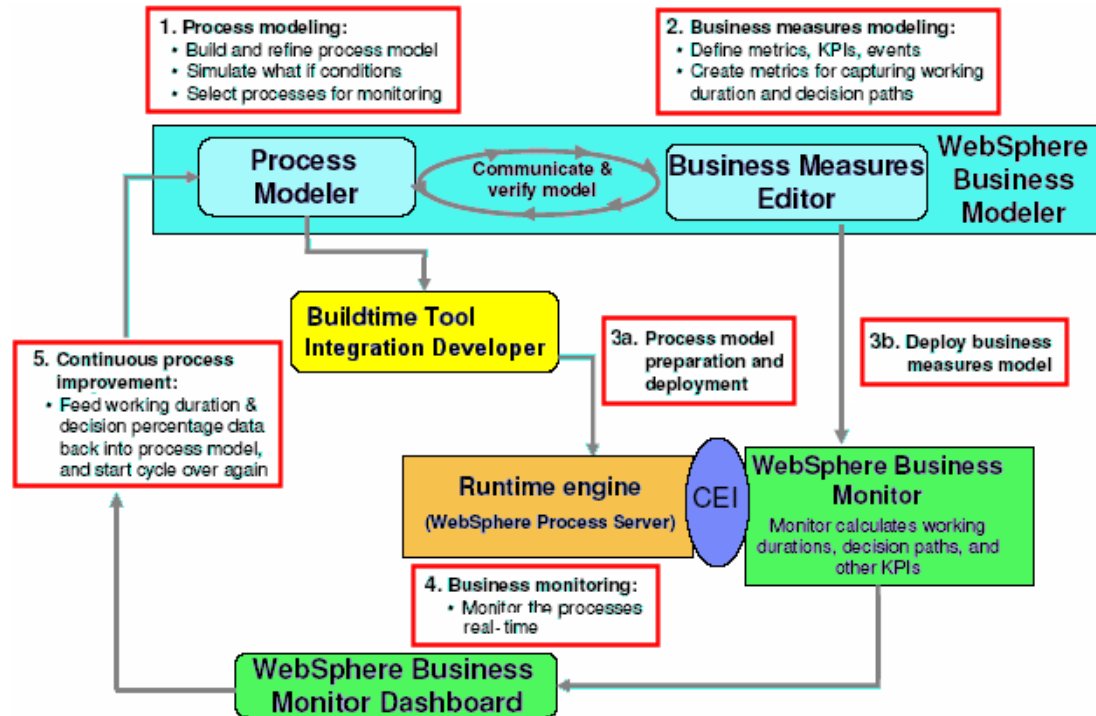
Figure 12-1   Closed loop process: Modeler V6 to Monitor V6

The last step is referred to as the *continuous process improvement cycle*. This methodology allows businesses to reconcile their "as-is" business process with the "to-be" business process with actual collected runtime data.

This data can be compared to the simulations that were performed in the Modeler and hence, optimize the "as-is" business process model toward a more efficient "to-be" model.

In Part 5, "Continuous process improvement" on page 355, we will explore the continuous process improvement lifecycle by utilizing IBM WebSphere Business Modeler and Monitor with our ClipsAndTacks project scenario. We will demonstrate how this methodology works and how it will yield higher efficiency in the overall practice of business performance management (BPM).

WebSphere Business Monitor architecture overview

The architecture of WebSphere Business Monitor 6.0 comprises a set of internal components and a group of external components.

The diagram shown in Figure 12-2 represents the overall logical reference architecture and the components of Monitor V6.
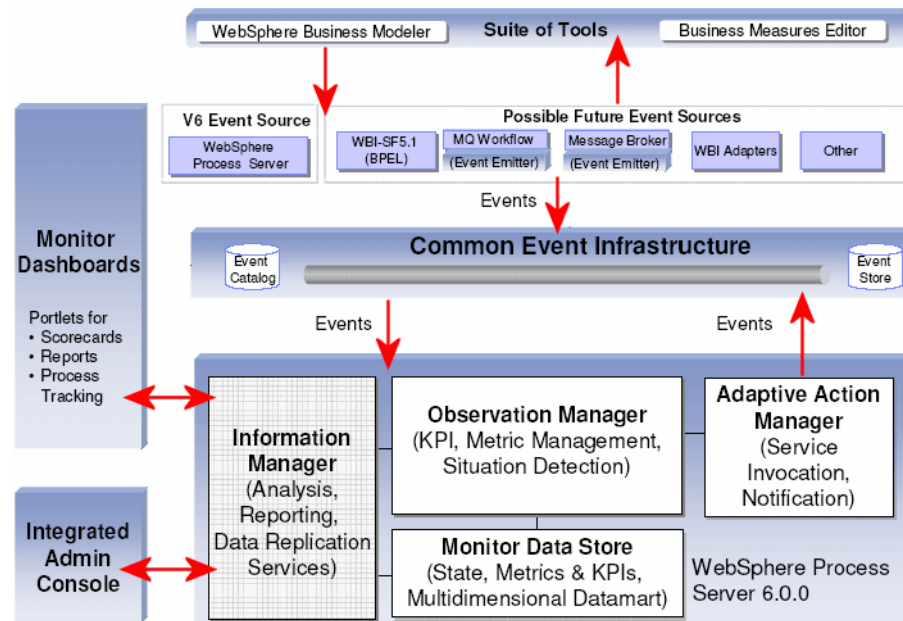
Figure 12-2   WebSphere Business Monitor V6: Reference architecture diagram

Overview of WebSphere Business Monitor 6.0 internal components

Here is a list of the internal components:

► **Monitor server**—Receives events, handles monitoring-context instances, and stores and persists runtime and historical metrics and KPI values of those instances.

► **Dashboards**—Display the monitored data. They provide a predefined set of views that can be customized to support different representations of data and offer enhanced data analysis.

▶ **Schema generator**—Generates database scripts to be used for creating databases tables in state, runtime, and historical databases. These databases contain the business measures models data. The schema generator also generates the DB2 Cube Views™ metadata description of the historical database and generates the metadata mappings for the replication manager.

▶ **Databases**—Provide the Monitor server with information for event processing. They also provide the dashboard client with information for populating views. Information is transferred across the databases through another monitor component, the replication manager.

▶ **Adaptive action manager**— Provides different types of business responses resulting from situations expressed within the incoming events.

Overview of WebSphere Business Monitor 6.0 external components

Here is the list of the external components:

▶ **Business measures editor** (BME)—It is used to create the business measures model that defines what should be monitored, for example, monitoring contexts, key performance indicators, metrics, and business situations.

▶ **Common event infrastructure** (CEI)—Provides event management by receiving events from event sources and transferring them to the event consumers that have expressed interest in those events.

▶ **DB2 Alphablox and DB2 Cube Views**—Provide enhanced data analysis for dashboards.

Figure 12-3 summarizes the overview of components discussed.
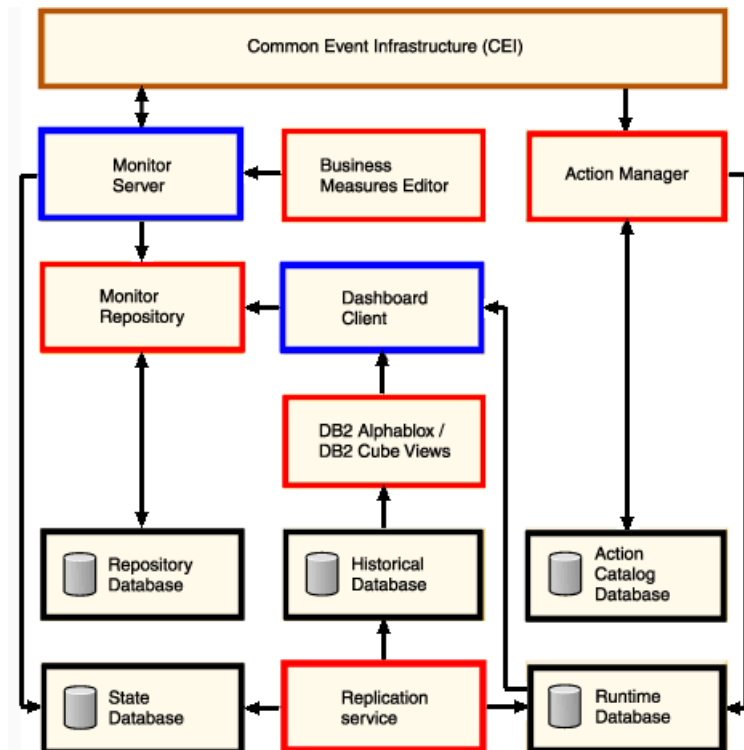
Figure 12-3   WebSphere Business Monitor V6: Component overview

Detailed description of Monitor V6.0 components

WebSphere Business Monitor 6.0 enables you to monitor the run time behavior of business processes through a Web application that will be deployed on WebSphere Process Server v6.0. The data that will be monitored is issued from a runtime engine.

Business measures model

Monitoring of data is based on a business measures model which includes artifacts that permit correlation of the runtime events with a specific instance, in addition to entries that specify situations. The business measures model is obtained from the original business model by editing entries that are essential for monitoring purposes: correlation of events, metric calculations, detecting situations. Editing of the model is done by the Business Measures Editor.

Event handling

Data is encapsulated in common base events (CBE) by means of event emitters and are transmitted on a common event bus, the common event infrastructure (CEI). For Monitor V6, only events emitted from WebSphere Process Server are supported.

Monitor V6 runs on WebSphere Process Server V6, which sits on top of WebSphere Application Server V6. The runtime engine (WebSphere Process Server) for the real application runs on another machine and CBEs are extracted from the engine and submitted on the event bus, configured as a service integration bus (SIB) between the two machines. The Monitor Server receives the CBE events from the SIB and matches them to monitoring context instances and calculates the appropriate metrics and KPIs to ultimately be displayed for the user on the dashboard.

The common event infrastructure (CEI) is a shared component that can operate either inside or outside WebSphere Application Server. The CEI provides facilities for the run-time environment to persistently store and retrieve events from many different programming environments. Events are represented using the common base event model, a standard, XML-based format that defines the structure of an event. The events are passed through JMS across the service integration bus (SIB).

Adaptive action manager

The adaptive action manager (or action manager) is another key component in the Monitor 6 product architecture, which provides the *sense and respond* functionality. The adaptive action manager is the WebSphere Business Monitor component that receives situation events emitted by the observation manager. It selects appropriate actions based on predefined bindings between the situations and actions that are set by the user, and invokes one or more action services. The CEI sends these situation events to the adaptive action manager, which parses them, selects appropriate actions based on predefined rules created by the user, and invokes a selected action or set of actions.

The adaptive action manager performs two types of actions: notification actions and service invocation actions:

► Notification actions take the form of e-mail, SMS, pager message, or a dashboard alert.

► Service invocation actions invoke a Web service, or a BPEL process through a Web service invocation.

The adaptive action manager parses the received situation event and selects an appropriate action by looking up the action in the action catalog database, where action-related information and binding information are stored. If the appropriate

action is a dashboard alert, the action manager extracts the data needed for the creation of the alert-notification record from the received situation event and inserts this record in the WebSphere Business Monitor Runtime database. The record is collected by the Alerts view in a dashboard. The adaptive action manager uses LDAP as the user-registry for user notifications.

Databases

The data architecture of Monitor V6 has been optimized for both transaction processing datastores as well as datamarts for reporting, analytics and business intelligence.

Simply stated, the Monitor V6 is responsible for its own datastore to handle data required for the monitoring operation: instances of running monitoring contexts and metric values. The performance is optimized by dividing the datastore into different databases, each being optimized for specific types of DB access operations.

DB2 replication services are responsible for moving state data to the historical datastore at configurable replication intervals. This fundamentally separates the transaction processing datastore from the historical datastore for high-performance event processing.

Data analysis can then be performed on the historical data, made available by introducing DB2 Cube Views and accessing cubes from DB2 Alphablox interface, which is the visualization module.

The database topology for the Monitor Server and Dashboard Server in a given server environment may vary, for example:

► The Monitor Server runs on its own machine with the State and Repository databases.

► The Monitor Dashboard runs on a separate machine using WebSphere Application Server and Portal Server with the Runtime and History databases.

This setup is done for performance reasons. However, you may want the Repository database on the Dashboard Server because the Monitor Server only uses the Repository database at the time that you import a model. However the Dashboard Server uses it frequently when configuring and displaying the dashboards.

Alternatively you may put all the databases on a dedicated database machine.

Figure 12-4 shows the typical topology.
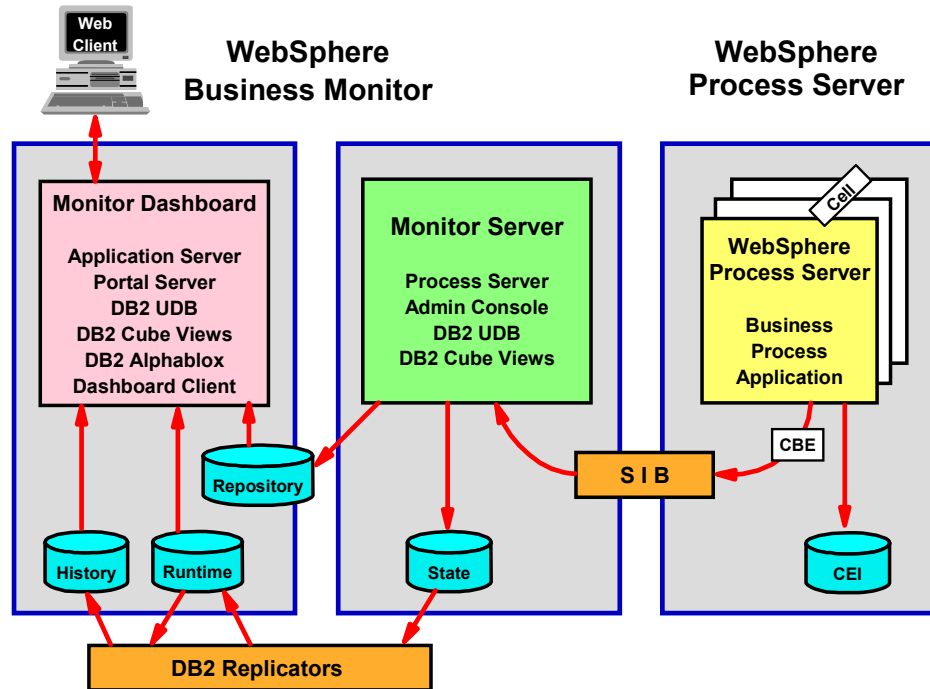
Figure 12-4   Typical Monitor databases and CEI topology

Monitor databases

The Monitor uses a number of databases to store event information. Here is a short description of the databases.

Repository database

The Repository database contains the metadata describing the currently deployed business measures models as well as information about the other Monitor databases. The Repository database contains the history of the deployed models. There is only one Repository database per Monitor installation.

The Repository database is used by the Launchpad, which populates it with the database attributes for the State, Runtime, and Historical databases. These attributes are the database name, database schema, and host names of the database server. They are used by the other Monitor components to access the State, Runtime, and Historical databases at runtime. The Repository database is also populated during the import of the business measures model.

State database

The State database stores information about running instances. This information includes metrics, business measures, and key performance indicators (KPIs) values. It is optimized for heavy transaction workloads. There is only one State database per Monitor installation.

Each process instance requires two tables in the State database to store metrics, business measures, and KPIs. The structure of these tables is as dynamic as the structure of the process instance. Each business measure is represented by a separate column in one of the two tables. Depending on the options selected during the building of the business measures models, much or all of the information in the State database is replicated to the Runtime database.

The State database is used by Monitor server. At runtime, the Monitor server inserts, retrieves, and updates the information of processes instances that reside in the State database, according to the processed events.

The State database stores the following information:

► Information about business measures groups, which is a part of the data in the imported business measures models.

► The running process instances that are created while the Monitor is running.

► The event entries of the running processes. The event entry is the event data that is received for updating a specific business measures group.

Runtime database

The Runtime database is similar in structure to the State database. It receives replicated information from the State database about the current status of all running processes as well as the final status of recently completed or failed processes. This information is used by Monitor dashboards. The Runtime database is also used by the Adaptive Action Manager to store alert notifications. There is only one Runtime database per Monitor installation.

The information in the Runtime database is replicated from the State database. The Runtime database stores:

► Alert notifications sent by the Adaptive Action Manager to the dashboards
► Process instance data
► Metrics values

The Runtime database is used by the Monitor dashboards. The dashboards retrieve the running or recently completed instances data required to populate the views from the Runtime database. The dashboard views use the Runtime database for analytical purposes, so it is optimized for query processing and aggregate query processing.

All completed instances remain in the Runtime database for 24 hours and are deleted afterwards. 24 hours is the default retention policy which can be modified as part of the data movement service configuration.

Historydatabase

The History database stores all completed and running process instances. It is used by the dashboards for enhanced data analysis using DB2 Alphablox. There is only one History database per Monitor installation. The data in the History database is never deleted.

The History database should only contain two years worth of historical data. This is one of Monitor product requirements. As mentioned before, the historical data is never deleted automatically, so the DBA is responsible for deleting the data that is greater than two years old.

The History database stores the information regarding long-running instances as well as completed instances. This information is stored as star schemas rather than in the flat transactional forms used in the State and Runtime databases. The History database is optimized for aggregated and long running queries. It is used by DB2 Alphablox in dashboards views to provide advanced multidimensional reports.

The information in the History database is replicated from the Runtime database.

The History database contains dynamic tables that are created according to the deployed business measures model. The schema generator generates the History database schema, which is used to create dynamic tables, and Cube Views definitions.

The History database is used by the Monitor dashboards. The dashboards retrieve the data required to populate some views from the History database. For example, the Reports view focuses on analyzing data extracted from the History database.

ClipsAndTacks Monitor topology and configuration

As we progress in this chapter through the ClipsAndTacks project scenario, we use a more simplified topology where all of the databases reside on the Monitor Server, and the Dashboard Server uses the Monitor Server databases as remote databases.

Figure 12-5 shows our software installation configuration and database topology for the ClipsAndTacks scenario.
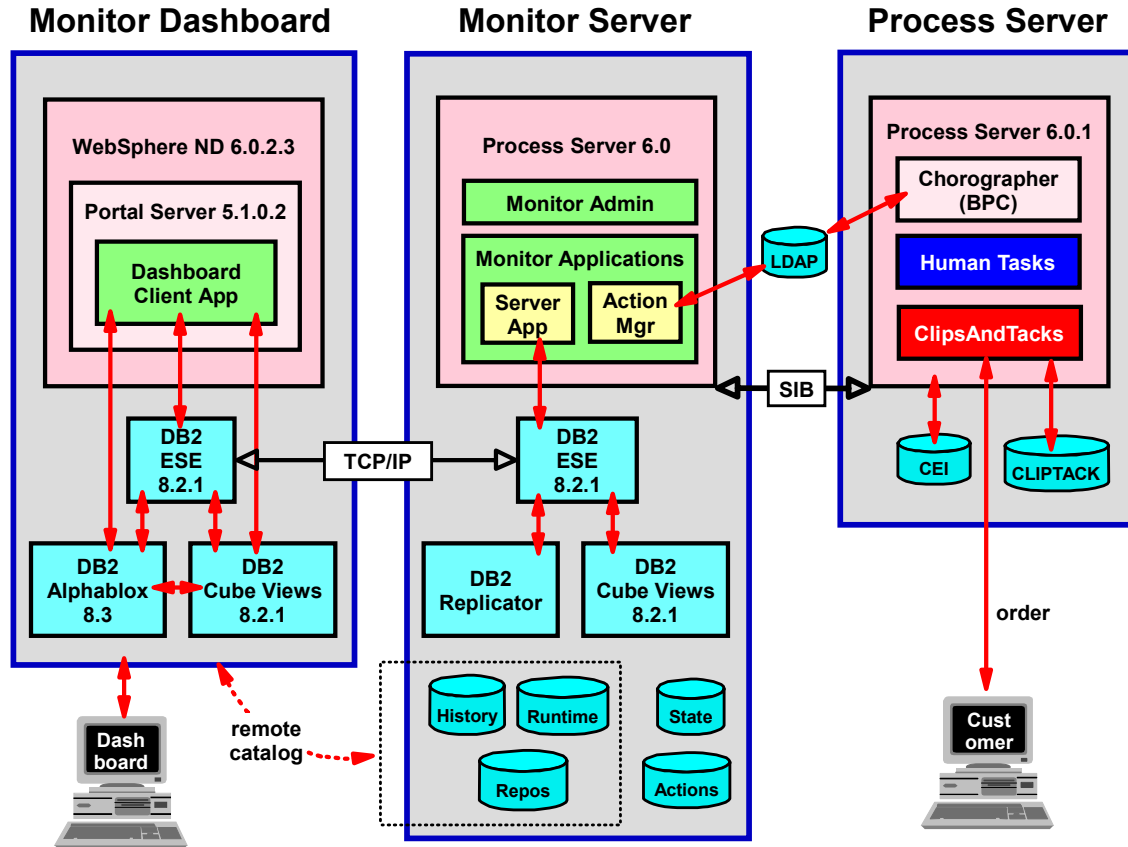
Figure 12-5   ClipsAndTacks project software configuration and topology

Here is a short description of our setup:

▶ We run one Process Server for the ClipsAndTacks business process. We use either the BPC Explorer or our own human task Web application to work on reviewing and shipping orders.

▶ The ClipsAndTacks Web application is used to submit orders.

▶ When working with security for human tasks we use an LDAP registry where groups and users have been defined.

▶ We use the same LDAP registry for action alerts.

▶ The databases used by the Monitor Dashboard are residing on the Monitor Server and are accessed as remote databases through TCP/IP.

Business measures model deployment to Monitor

In this section we demonstrate the process (Step 3b in Figure 12-1) of deploying our ClipsAndTacks scenario business measures model into the Monitor Server.

This process is accomplished by generating the necessary schema artifacts (DDL files and scripts for the DB2 replication manager to utilize) as well as the DB2 Cube Views metadata description files. Then finally importing the XMI file (zipped) through the Monitor administrative console into the Monitor Server, which creates a new observation manager instance for this model and automatically restarts the Monitor Server.

Referring back to the logical reference architecture diagram (Figure 12-2 on page 291), the observation manager and information manager of the Monitor Server are the components that make use of this information to process the incoming event data.

In "Configuring alerts for the adaptive action manager" on page 309 we demonstrate the configuration and usage of *situation events* in the adaptive action manager of the WebSphere Business Monitor V6, for what is often referred to as "sense and respond" functionality. We will introduce this functionality during the subsequent continuous process improvement lifecycle of the ClipsAndTacks scenario.

The pre-requisite for this section is a complete installation of the WebSphere Business Monitor V6 Server (as described in Appendix B, "Installation of WebSphere Business Monitor" on page 417).

> **Important:** When deploying a business measure model into Monitor Server, you lose all previous definitions of the business measures model if you do not rename the model. This is due to the fact that there currently is no capability to version the business measures model and data within Monitor and DB2 Cube Views. Thus, you should be cautious when importing a model that has the same name as one that is previously deployed into Monitor.

Business measures deployment overview

Figure 12-6 shows the sequence of steps involved when deploying the business measures to the Monitor.
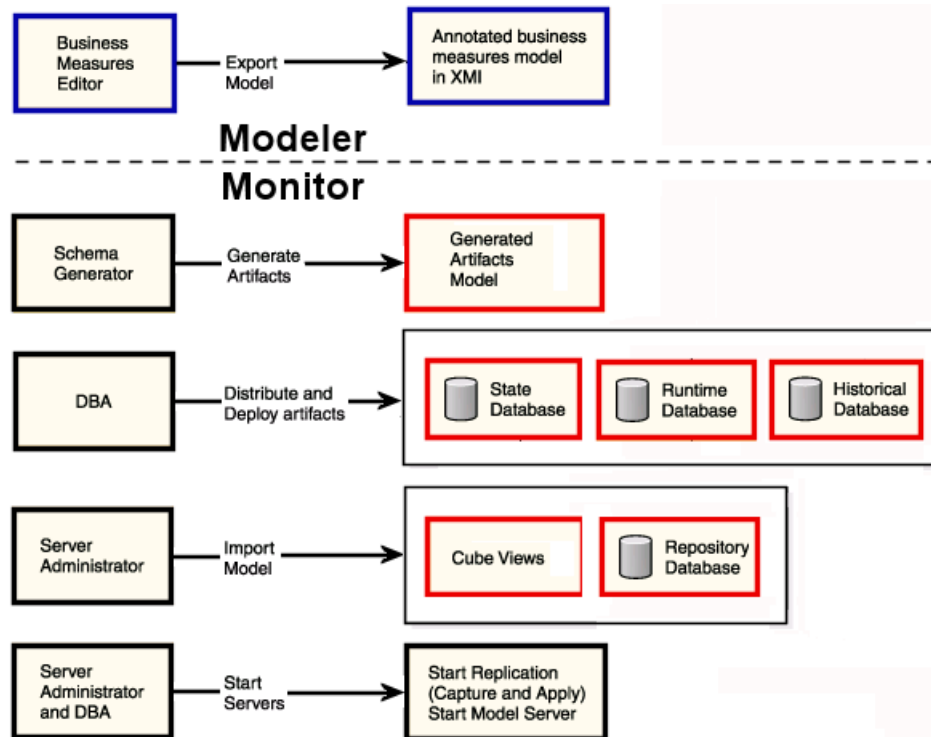
Figure 12-6   Business measures deployment sequence

We already exported the business measures from the Modeler into a
`Monitor.zip` file. Now we will deploy the model to our Monitor installation.

Import the business measures

First we will configure and import the business measures using the WebSphere
Business Monitor administrative console:

► Navigate to *WebSphere Business Monitor → Schema Generator →
  Configuration* (Figure 12-7). Select the *General Configuration* tab.

► In the Business Measures Model field specify the full path of where the
  business measures model zip file is located (as exported from the Modeler).

► In the Output Directory field specify the full path where the generated schema
  artifacts will be saved, for example:

    SG247148\zSchemagen\Future1

► Click *Apply*.
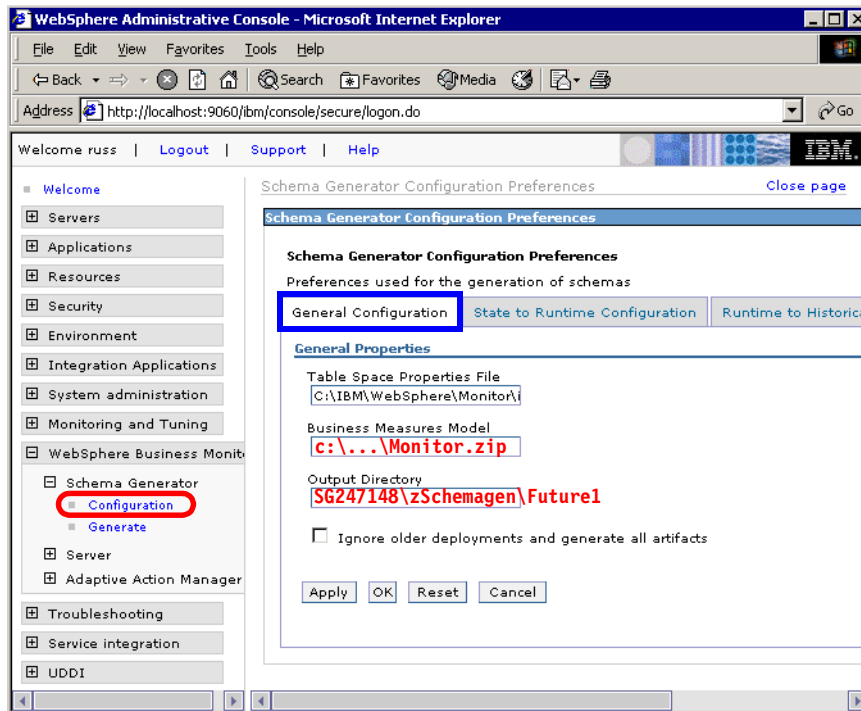
Figure 12-7   Business measures import configuration: General

Specify the schema generator configuration

Now select the *State to Runtime Configuration* tab, enter 5 (minutes) for the Runtime Database Population Interval, and click *Apply* (Figure 12-8).
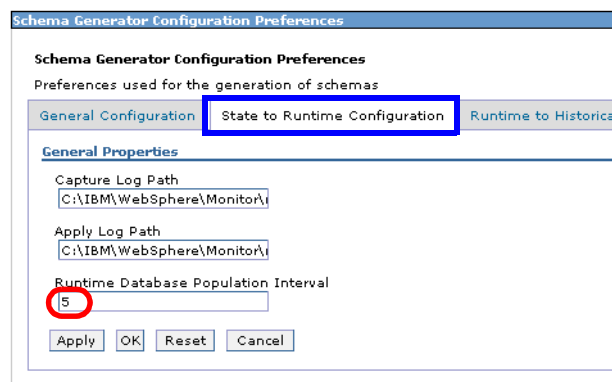
Figure 12-8   Business measures import configuration: State to Runtime

► Repeat this for the *Runtime to Historical Configuration*.

► Click *Save* to save the configuration.

The database population interval specified in the two places determines how often the DB2 replication services move data from the active state and runtime to the historical database, after a process has completed.

> **Note:** These values would normally be different within an enterprise-level implementation, depending upon the business needs by evaluating the trade-offs between performance and dashboard visibility and data refreshes.

Generate the schema

Select *Schema Generator → Generate* and click *Generate*. The schema generation takes some time. When completed you will get a confirmation (Figure 12-9).
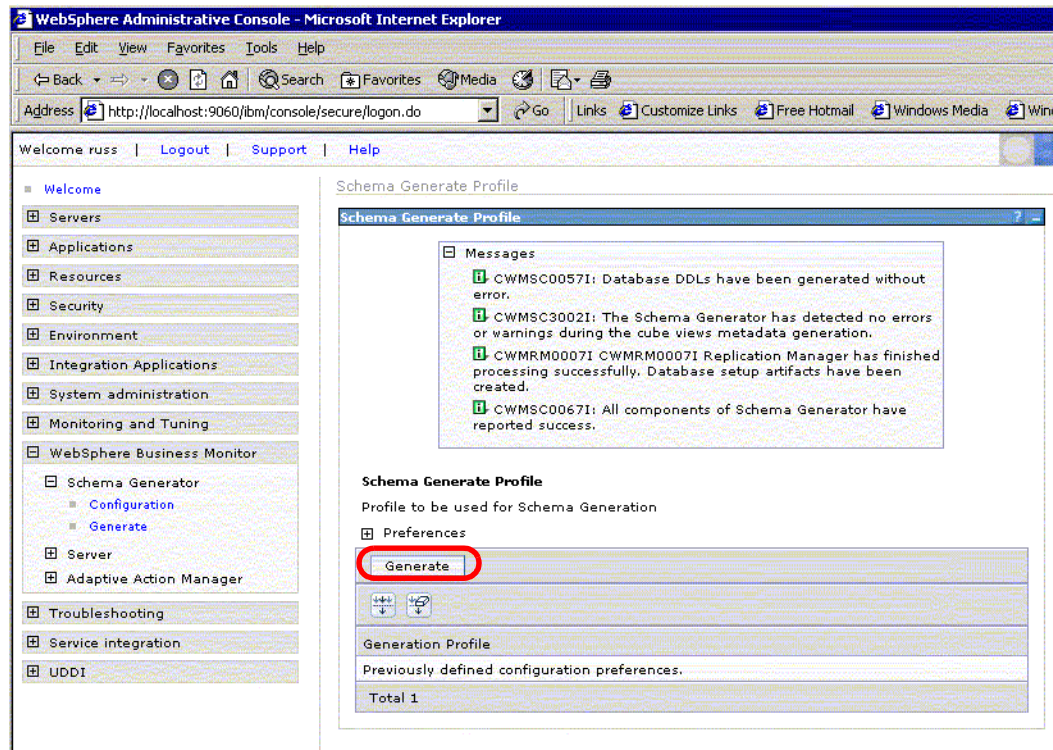
Figure 12-9   Schema generation

The output is generated into a `schemagen` subdirectory:

```
SG247148\zSchemagen\Future1\schemagen
```

Create the database tables

In Windows explorer, copy the folder (that you specified as the output directory to another folder on your system. We will unzip the artifacts and execute the scripts from the copy:

```
SG247148\zSchemagen\Future1\schemaExecute
abbreviated as <schemaExecute> from now on
```

► Unzip the three zip files in this directory:

```
DS_State_setup.zip
DS_Runtime_setup.zip
DS_Datamart_setup.zip
```

► From a DB2 Command Window, change directory to where the three DDL files for the three databases (`STATE`, `RUNTIME`, `HISTORY`) are located:

```
state.ddl
runtime.ddl
datamart.ddl
```

► Execute the DB2 commands to run the three DDL file. You may want to make a batch file (`execDDL.bat`) for these commands to deploy models easier as we progress (Figure 12-10).

```
db2 connect to STATE
db2 -tvf state.ddl > state.log
db2 disconnect STATE
echo STATE is finished

db2 connect to RUNTIME
db2 -tvf runtime.ddl > runtime.log
db2 disconnect RUNTIME
echo RUNTIME is finished

db2 connect to HISTORY
db2 -tvf datamart.ddl > datamartHistory.log
db2 disconnect HISTORY
echo HISTORY is finished
```

Figure 12-10   Sample batch file to execute the database DDL

Setup additional tables for replication

Now we execute the four batch files that generate the tables. These batch files run Java code so the PATH must be setup to include Java, for example:

```
set PATH=F:\IBM\WebSphere\ProcServer\java\bin;%PATH%

<schemaExecute>\State_to_Runtime_setup_source.bat
              \State_to_Runtime_setup_target.bat
              \Runtime_to_Historical_setup_source.bat
              \Runtime_to_Historical_setup_target.bat
```

When prompted a user ID just hit enter. The schema generator takes some time and displays a lot of text in each command window.

Again you may want to make a batch file (`execSchema.bat`) for the above commands to deploy models easier (Figure 12-11).

```
@echo off
set path=F:\IBM\WebSphere\ProcServer\java\bin;%PATH%
echo Executing the batch files now...
call State_to_Runtime_setup_source.bat
echo State_to_Runtime_setup_source.bat is done.
call State_to_Runtime_setup_target.bat
echo State_to_Runtime_setup_target.bat is done.
call Runtime_to_Historical_setup_source.bat
echo Runtime_to_Historical_setup_source.bat is done.
call Runtime_to_Historical_setup_target.bat
echo Runtime_to_Historical_setup_target.bat is done.
echo All batch files are finished
```

Figure 12-11   Sample batch file to execute the schema

> **Note:** Do not interrupt execution of these commands even when it appears to be stopped, as some of the database creation operations from these batches can take some time.

When all four batch files (or the `execSchema.bat`) have completed, locate four newly created batch files that begin with the words: `StartCapture` and `StartApply`. You should find files similar to these, although the exact filenames will be different each time the schema generator runs:

```
<schemaExecute>\State_to_Runtime\source\StartCapture_5.bat
              \State_to_Runtime\target\StartApply_12.bat
              \Runtime_to_Historical\source\StartCapture_5.bat
              \Runtime_to_Historical\target\StartApply_12.ba
```

Starting the database replicators

Execute each of the four batch files. These files launch the DB2 replicators for the business process model. You may create a batch file (startReplicators.bat) as shown in Figure 12-12, to be run from the base directory.

```
echo Starting Replicators
call State_to_Runtime\source\StartCapture_5.bat
call State_to_Runtime\target\StartApply_12.bat
call Runtime_to_Historical\source\StartCapture_5.bat
call Runtime_to_Historical\target\StartApply_12.bat
```

Figure 12-12   Sample batch file for DB2 replicators

The four filenames in this batch file are only valid for the last model that was deployed. You may use this to start the replicators for this model version.

When you deploy another version of the model, you must ensure that the StartCapture and StartApply filenames that are regenerated are replaced in this batch file.

Note: Do not close any of the four background DB2 replicator command windows that are launched! This is normal operation for the Monitor Server.

Figure 12-13 shows an example of executing the startReplicators.bat file and the four DB2 replicator command windows that it launches.
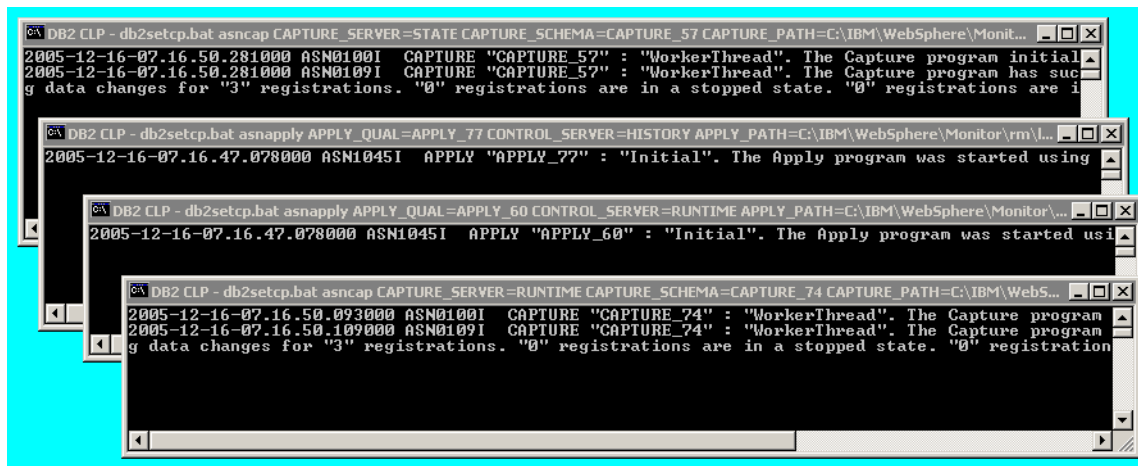
Figure 12-13   DB2 Replicators started with batch file, startReplicators.bat

Deploying the business measures

Now we can actually deploy the business measures model into Monitor Server:

► Return to WebSphere administrative console and select *WebSphere Business Monitor → Server → Business Measures Model → Model Import*.

► Click *Browse* to locate the `Monitor.zip` file, which is located in the same `<schemaExecute>` folder. Note that this file is not the same `Monitor.zip` file exported from the Modeler.

► Click *Import*. This will take some time, depending upon the complexity of your model. Wait until you see the confirmation screen, stating that the Import was successful (Figure 12-14).
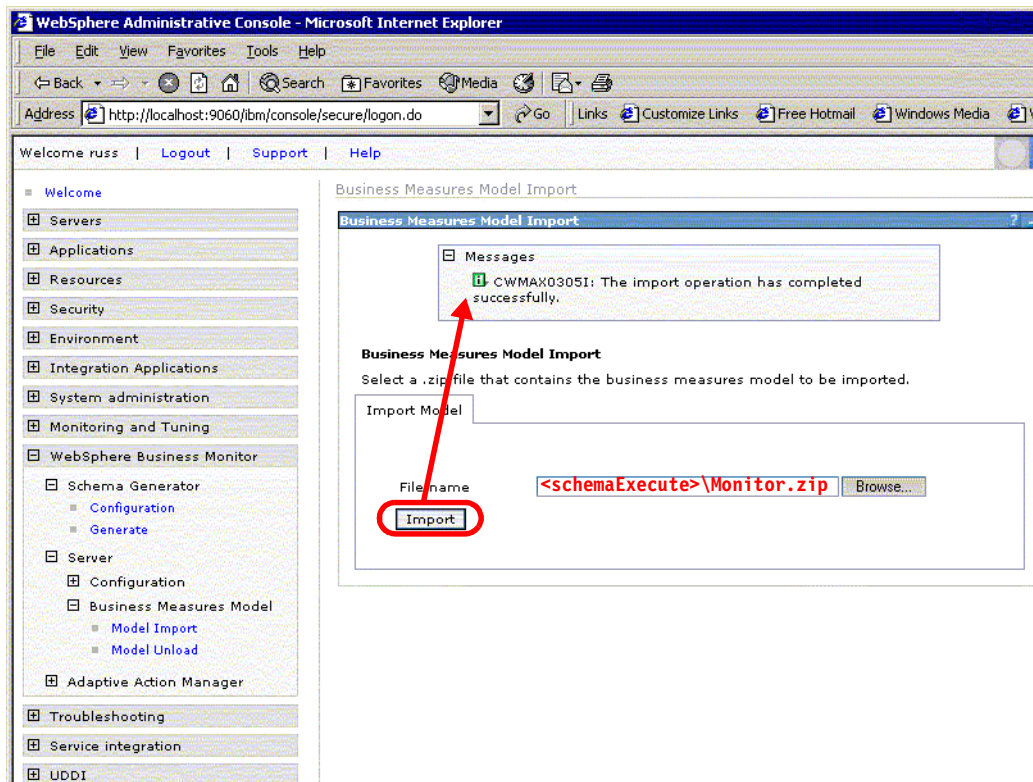
Figure 12-14   Business measure import successfully completed

Importing the DB2 Cube Views

We have now fully deployed the model into the Monitor Server, however prior to taking measurements and collecting event data, we must import the DB2 Cube Views generated model using the DB2 Cubes OLAP Center:

► Start OLAP Center by selecting *Start → Programs → DB2 → Business Intelligence Tools → OLAP Center*.

► Enter the name of your history database, `HISTORY`, the db2admin user ID and password.

► You receive a message dialog the first time using the OLAP Center. Click *Yes*, then from menu select: *OLAP Center → Import*.

► Browse to the directory containing the generated model and select the file: `<schemaExecute>\model_cv.xml`.

► Click *Next*. When the cube models have been loaded into the OLAP center, you receive a confirmation (Figure 12-15).
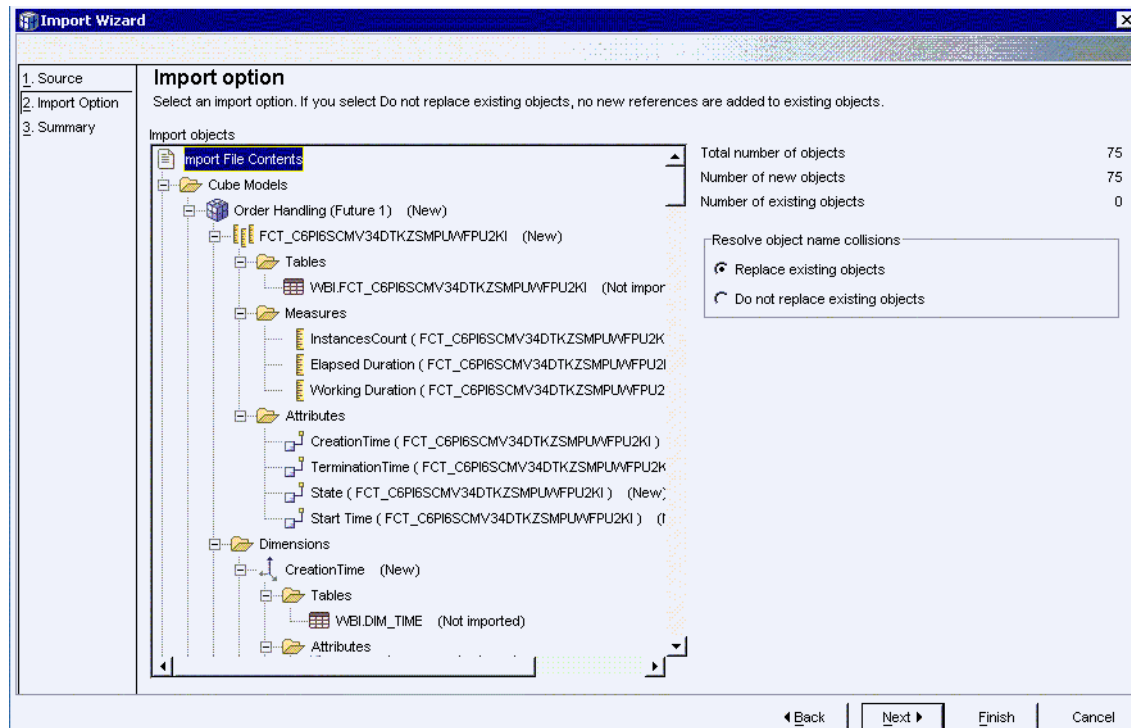
Figure 12-15   ClipsAndTacks: Order handling cube model view

► Click *Finish*.

> **Important:** For the Clips and Tacks database topology, this last step of loading the cube views model definitions into DB2 OLAP was performed on the Monitor Server machine because the `HISTORY` database resides on this machine. Normally, for performance reasons, the `HISTORY` database would reside on the Monitor Dashboard machine, and you would import the cube views model definitions into DB2 OLAP on the Monitor Dashboard machine.

You have now completed the business measures model deployment procedure into the WebSphere Business Monitor Server.

Configuring alerts for the adaptive action manager

This section we complete our configuration of the Monitor Server environment for the ClipsAndTacks scenario by configuring *situation events* within the Monitor Server administrative console:

► Select *WebSphere Business Monitor → Adaptive Action Manager → Configuration*.

► Select the *LDAP* tab. Enter your LDAP URL, LDAP administrator user ID and password (Figure 12-16).
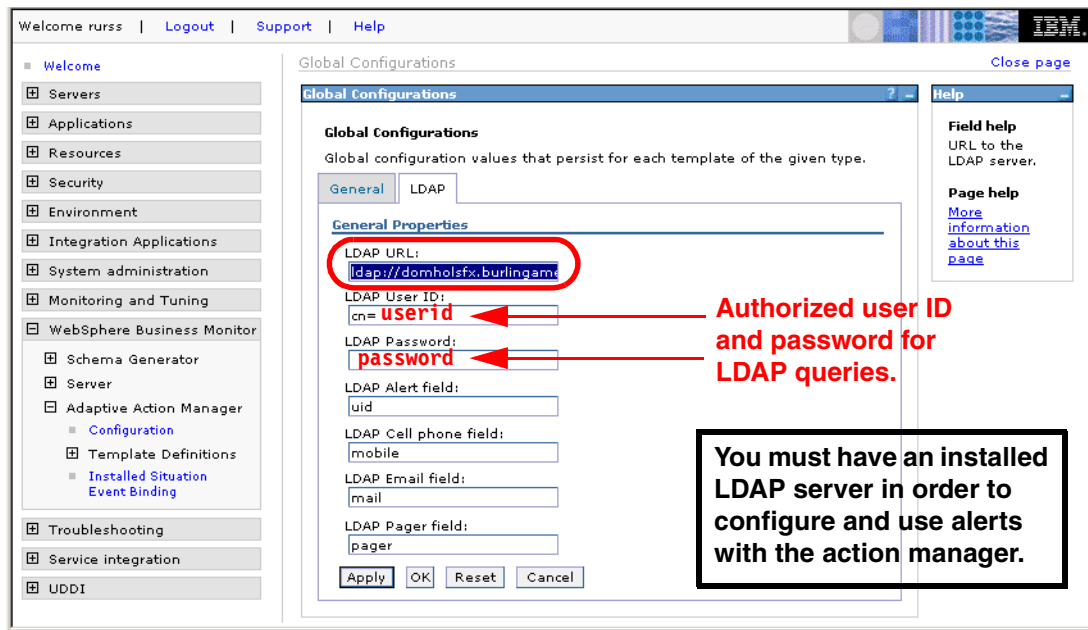
Figure 12-16   Adaptive action manager LDAP configuration

The LDAP URL may look like this when fully qualified with port#:

```
ldap://domholsfx.burlingame.ibm.com:389
```

In our case we use a Domino LDAP server on port 389.

► Select *Applications → Enterprise Applications* and restart the action manager enterprise application `IBM_WB_ACTIONMANAGER` for these changes to take effect.

Notifications

For each situation event that we defined in "Situation events" on page 176 we define a notification template for the action to be taken when the event occurs:

► In the administrative console and select *WebSphere Business Monitor → Adaptive Action Manager → Template Definitions → Notifications*.

► Click *New* and enter a name for the template, for example, `Late Order Shipped Alert` and a matching description of `Late Average Order Shipped Event` (Figure 12-17).

Figure 12-17   Action alert notification template configuration

– For type select *Alert*.

– Provide an LDAP query for the user(s) to be notified for this alert:

```
(&(objectClass=inetOrgPerson)(uid=andre))
```

The query will be dependent upon your LDAP server, in our case we select the order manager `andre`.

- Enter the subject (`Order Handling Alert`) and the body:

  `This is an automated alert that the average order handling time exceeds 3 days and 1 hour. Average time is: %Average Order Processing Time%`

  We defined the attribute `Average Order Processing Time` in the event.

- Click *OK*.

▶ Define a second template for the Orders Declined Event:

- Name—`Orders Declined Alert`
- Description—`Orders Declined Event`
- Body—`This is an automated alert that 5 or more orders have been declined (%BusinessSituationName%)`
- Alert, LDAP Query, Subject—Same as for the first event.

▶ The list of notification templates is shown in Figure 12-18
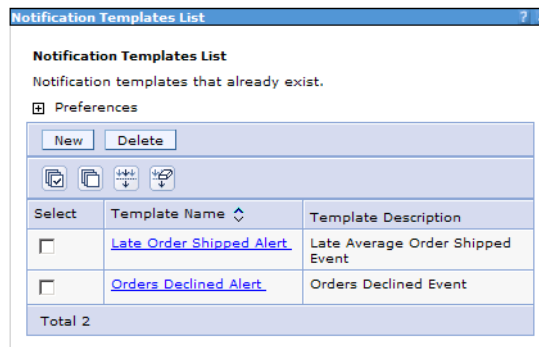
Figure 12-18   Alert notifications

Situation event binding

We have to define when the notifications are sent based on the events. The `BusinessSituationName` attribute of an event is used to trigger an action.

In "Triggering an event by a KPI" on page 177 we set the `BusinessSituationName` attribute for the two events to:

```
Average shipment is too late
5 orders have been declined
```

▶ Select *WebSphere Business Monitor → Adaptive Action Manager → Installed Situation Event Binding*.

► Click *New* and enter the situation event name, and a description. Click *Apply*. Click *Add* to select the template to bind for this situation event. In our case we select the `Late Order Shipped Alert`. Click *OK* (Figure 12-19).
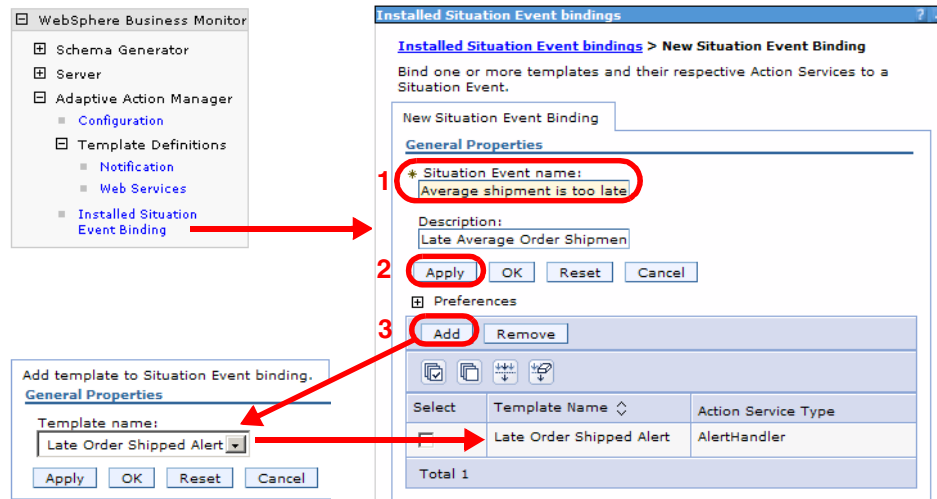
Figure 12-19   Situation event bindings configuration

► Define the second alert binding in the same way. The two bindings are shown in Figure 12-20.
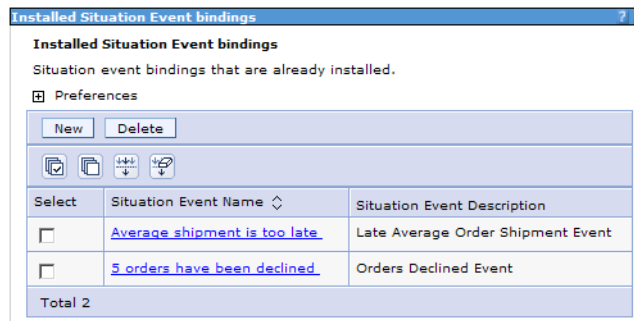
Figure 12-20   Situation event binding list

Note: The situation events as configured may not change for a new model deployment. This is based on your model and how the alerts were modeled. However, if you are unsure, you should revisit the adaptive action alerts configuration discussed in this section each time you deploy a new business measures model into Monitor.

Configuration of the Monitor Dashboard

In this section we continue with the configuration of the ClipsAndTacks scenario in the Monitor Dashboard Client environment:

▶ First we configure the DB2 Alphablox application through the DB2 Alphablox Administration Web-based client to define the cubes.

▶ Then we configure and personalize the Monitor Dashboard Client within WebSphere Portal Server to provide a certain "look and feel" for the subsequent sections, where we will use our customized dashboards for observing and analyzing data, performing analytics, and generating reports.

> **Note:** The pre-requisite for this section is a complete installation of the WebSphere Business Monitor V6 Server and Dashboard Client (as described in detail in Appendix B, "Installation of WebSphere Business Monitor" on page 417) in addition to completing the previous section of deploying the ClipsAndTacks model to the Monitor Server.

DB2 Alphablox setup

Startup the Portal Server on the Monitor Dashboard Client machine, if not already running.

On the Monitor Dashboard Client machine, start the DB2 Alphablox administration client by selecting *Start → Programs → IBM DB2 Alphablox 8.3 → AlphabloxAnalytics → DB2 Alphablox Administrative Pages*:

▶ Enter the userid `admin` and the correct password.

▶ Select *Administration → Data Sources* and click *Create*. Enter the data as follows:

    – Data Source Name: `HISTORY`
    – Adapter Type: *IBM DB2 JDBCType 4 Driver*
    – Database Name: Enter your historical database name, `HISTORY`.
    – Database Username/Password: credentials for the `db2admin` user

▶ Accept defaults for the rest and click *Save*.

▶ Create another data source, with name `HISTORY_CUBE` and select Adapter Type *Alphablox Cube Server Adapter*. Click *Save*.

Test both data sources you just created. You will see a dialog window indicating if the connection was successful.

▶ Select *Administration → Cubes* (Figure 12-21).

    – Select *Enable* to the right of the DB2 Alphablox Cube Name.

– Select *HISTORY* for Relational Data Source.

– Click *Enable DB2 Cube Views Settings*, select the Cube Model `WBI.Order Handling (Future 1)`, and click *Import Cube Definition*.

– Enter `Order Handling (Future 1)` as name.
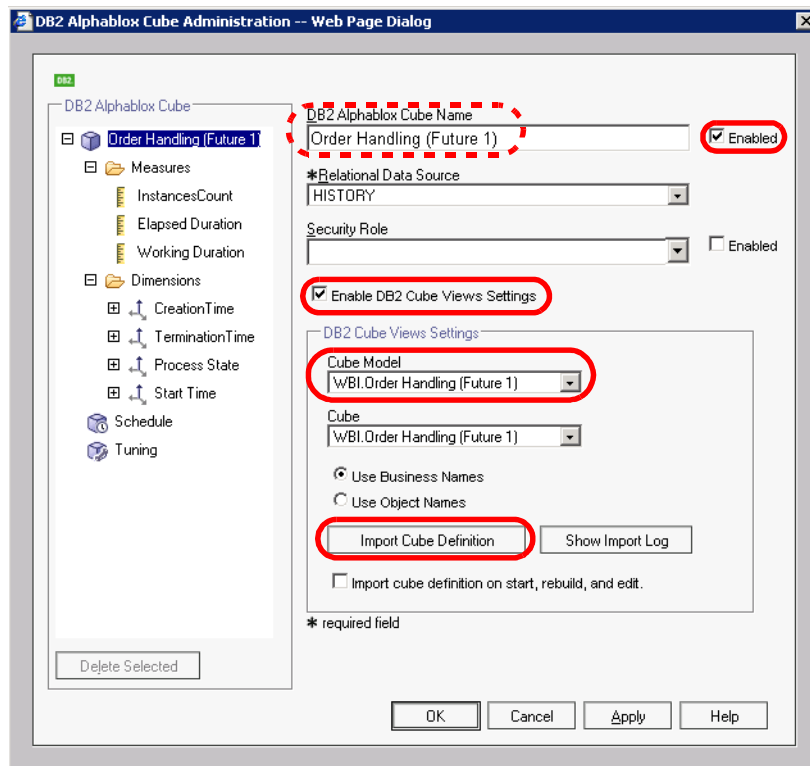
Figure 12-21   DB2 Alphablox Cube setup

► Click *OK* to save.

Important: You must enter the DB2 Alphablox Cube Name exactly as it is spelled in the DB2 Cube Views Settings or your installation will be invalid.

Repeat the same procedure for `Order Handling (Future 1)-aggregates` (Figure 12-22) and `Order Handling (Future 1)_Tasks` (Figure 12-23).
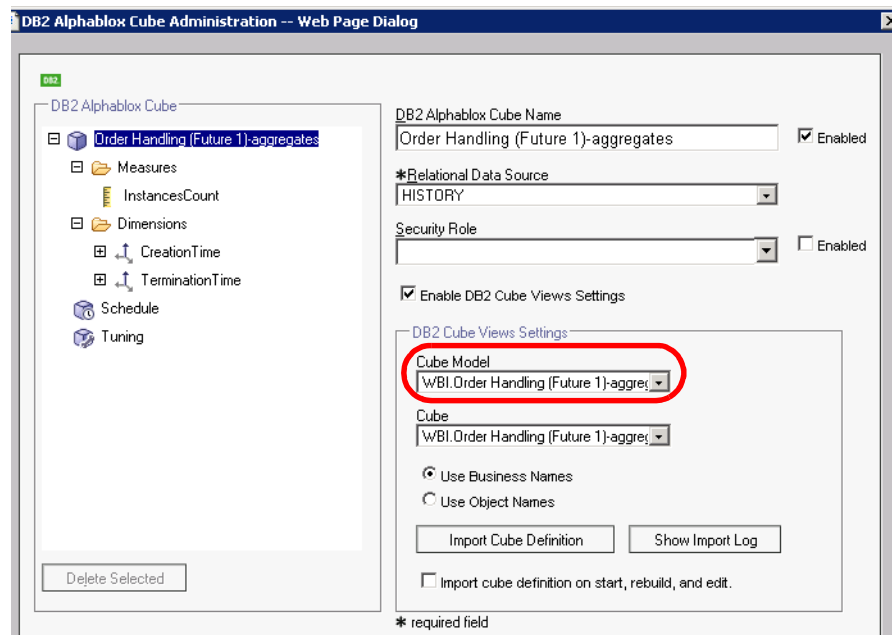
Figure 12-22   DB2 Alphablox Cube setup: Aggregates
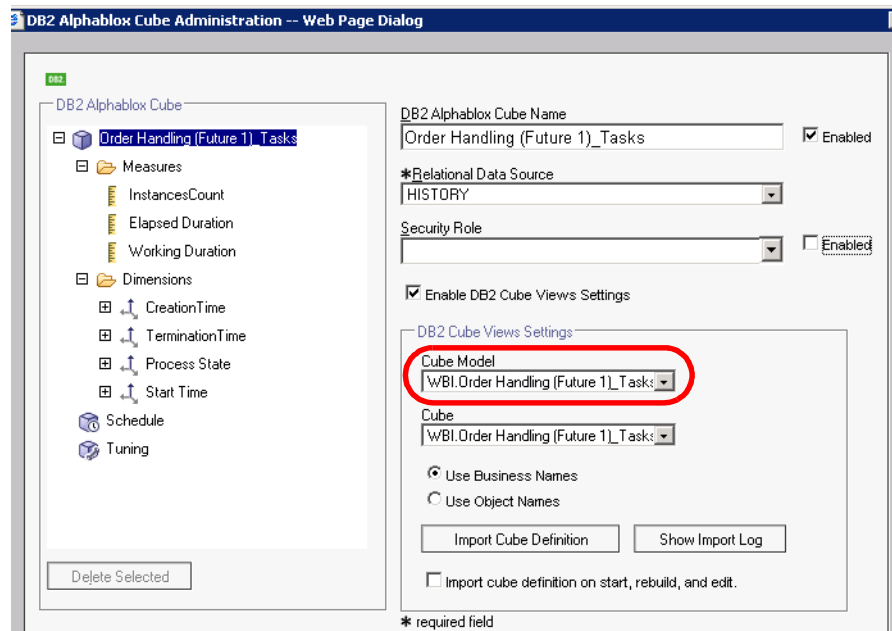
Figure 12-23   DB2 Alphablox Cube setup: Tasks

After you have created and saved all three Alphablox Cubes, select *Administration → General → Runtime Management → DB2 Alphablox Cubes*:

► For each of the cubes that you created, click *Start*. You get a confirmation as shown in Figure 12-24.
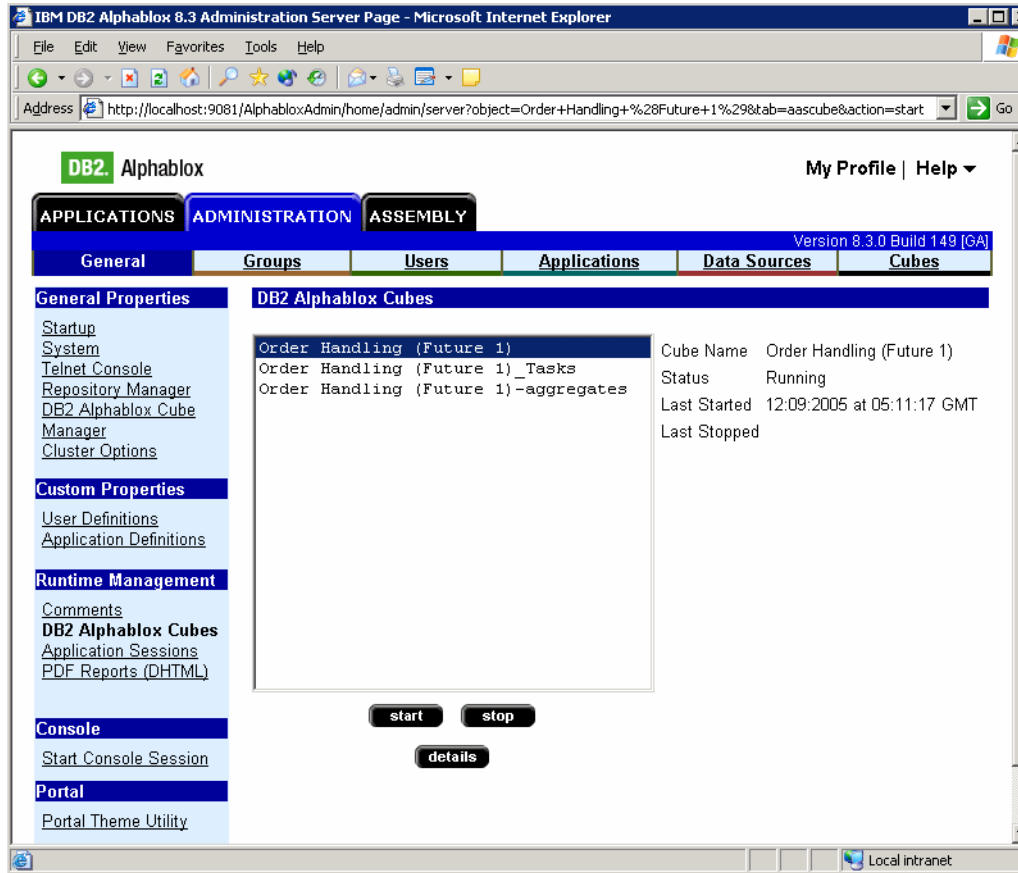
Figure 12-24   DB2 Alphablox Cubes: Runtime status screen

► Repeat this for the remaining two cubes and ensure all three DB2 Alphablox Cubes show a status of *running*.

You have now completed the DB2 Alphablox Cubes configuration.

Monitor Dashboard Client portal customization

This section pertains to configuring the Monitor Dashboard Client portlets within WebSphere Portal Server 5.1. For general help on WebSphere Portal Server 5.1, refer to the Administration and Users Guide. We will show some sample screens of how the customizations pertain to usage of the WebSphere Business Monitor Dashboard.

Open up the home URL for Monitor Dashboard machine:

```
http://localhost:9081/wps/portal
```

▶ Click *Log in* at the top right Figure 12-25().
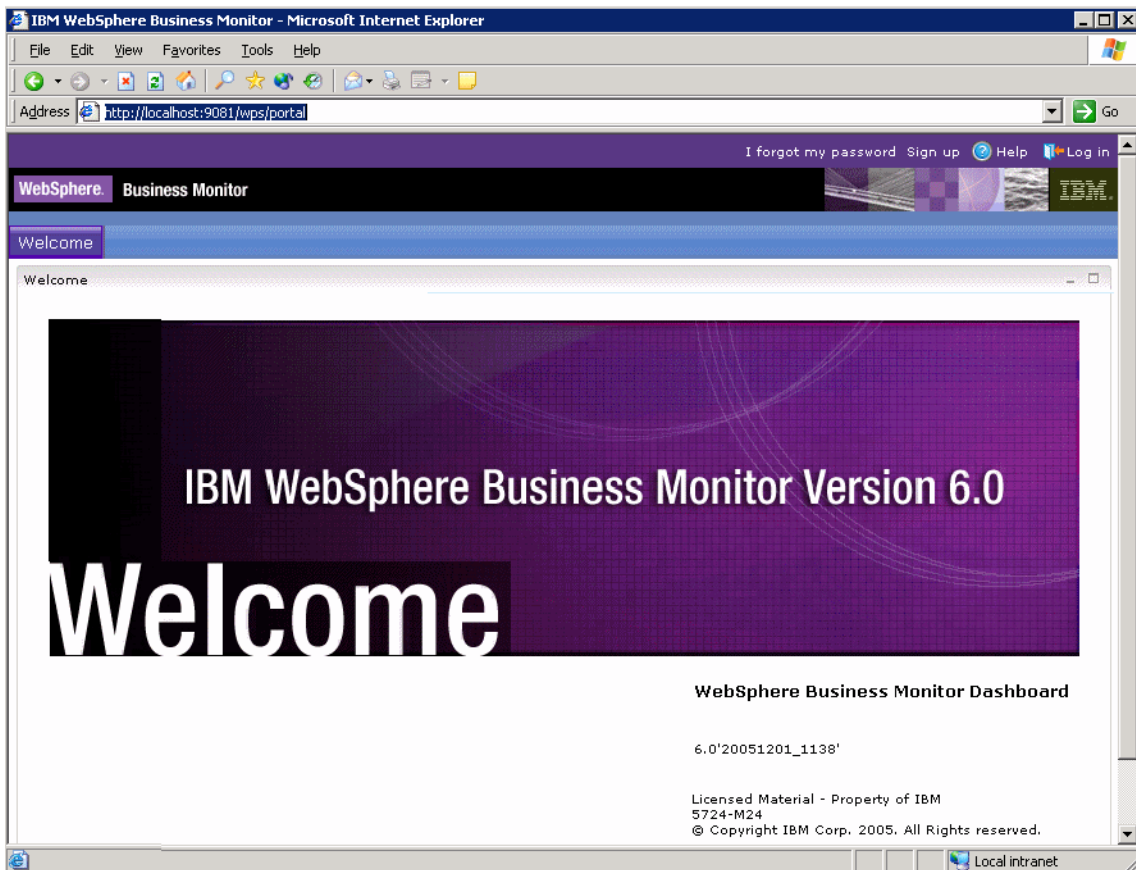
Figure 12-25   Monitor Dashboard Login

▶ Login with the user ID `wpsadmin` and the correct password.

► Select *Administration, Portlet Management → Portlets*, then go to page 11 of the portlets admin page to see the Monitor Dashboard portlets (Figure 12-26).
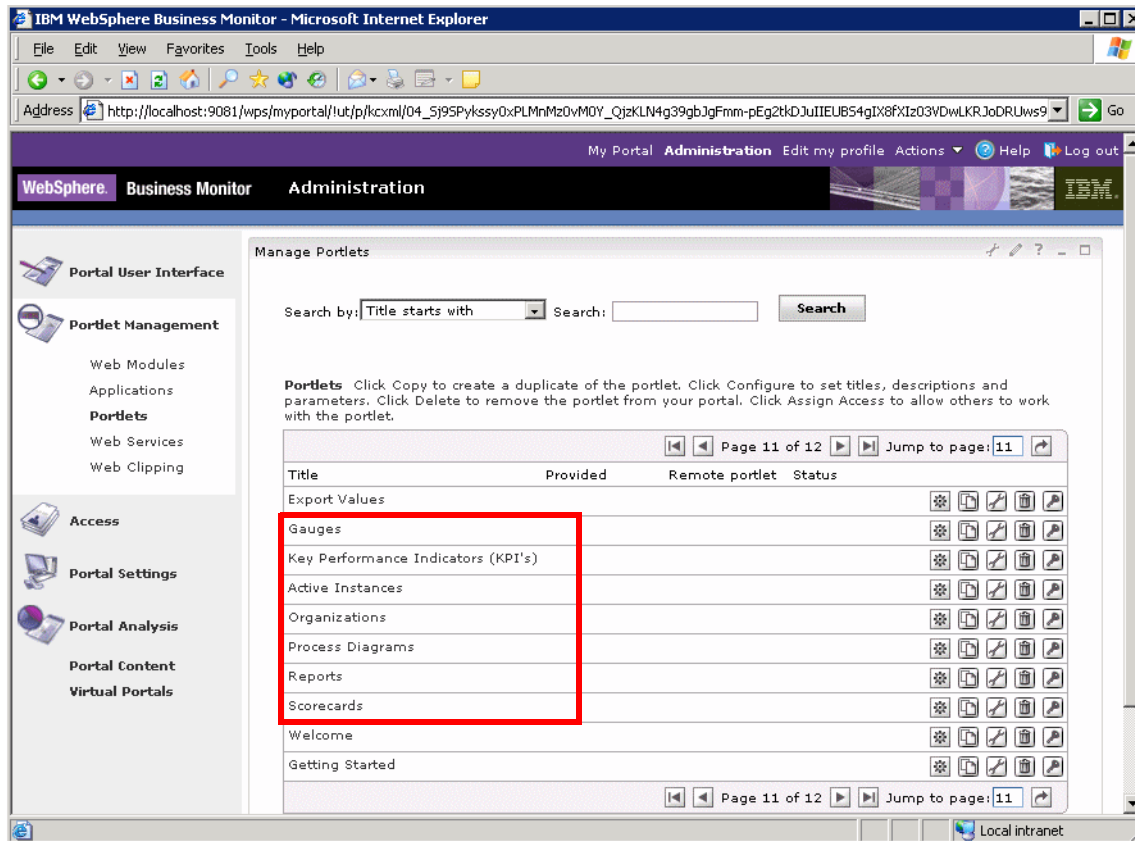
Figure 12-26   Portal Administration: Monitor Dashboard portlets

Here you will see the available Monitor Dashboard portlets. We suggest that you copy any of these portlets if you want to customize them for your pages, because this would modify the base portlet. Click the *Copy Portlet* icon to make a copy.

► Select *Portal User Interface → Manage Pages*.

► Click *New Page* and create a page. We name the page `Clips and Tacks Portal`.

► Select the new portal and create sub-pages for your customization. Our customization is shown in Figure 12-27.
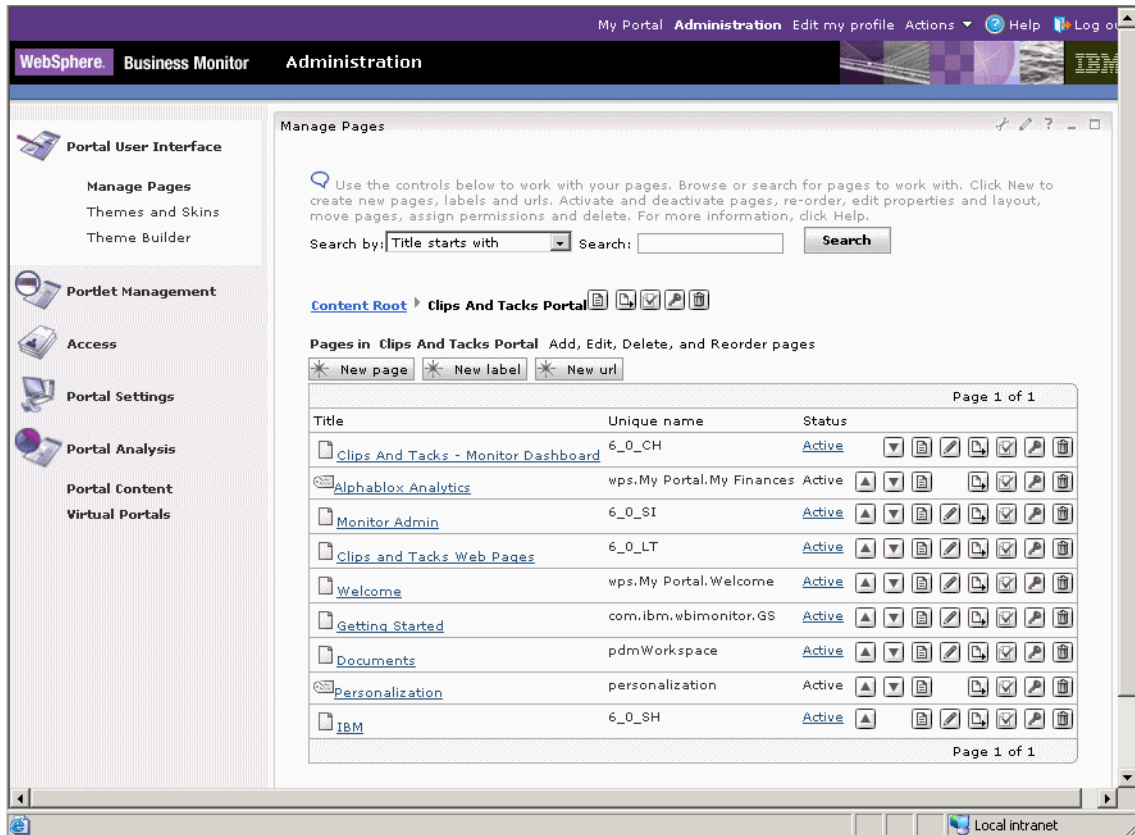
Figure 12-27   Sample ClipsAndTacks portal page administration

► For each page that you add, click on the configure page icon to add the portlets you want to add to a given page (Figure 12-28).
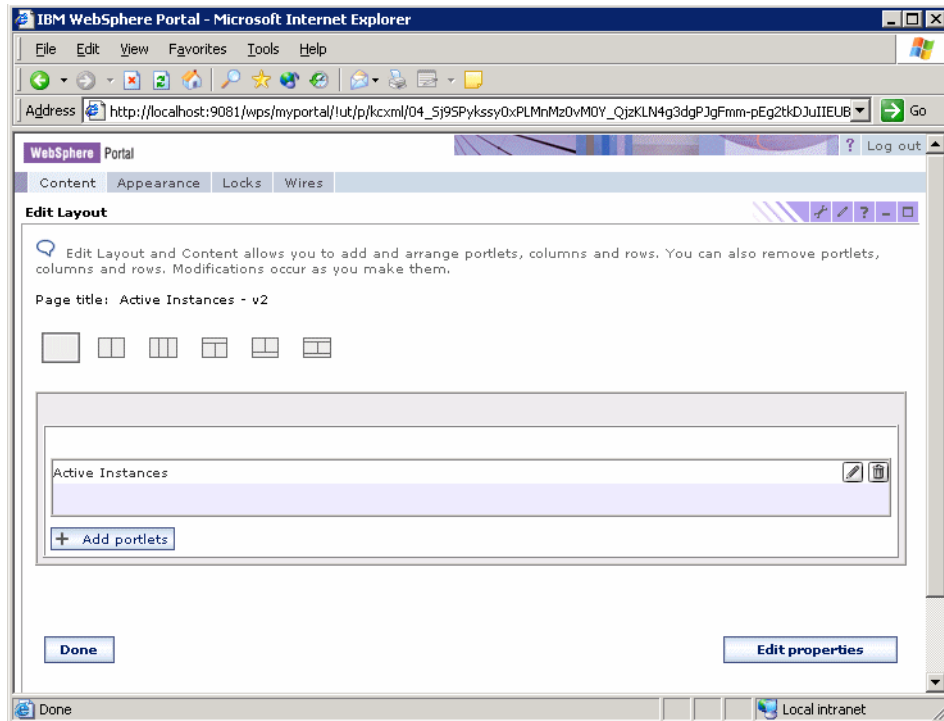
Figure 12-28   Active instances: Add portlets

► Click *Done*.

► To use that page, click *My Portal* at the top, then navigate to the page you just created, for example, *Active Instances*, under Clips and Tacks - Monitor Dashboard page.

   The first time you access this view, it will tell you it is not configured yet. To configure this view for this page click the on the configure icon 🔧 at the top right of portlet.

   You will be able to select the business measure model and measure group that you want to view for this page (Figure 12-29).
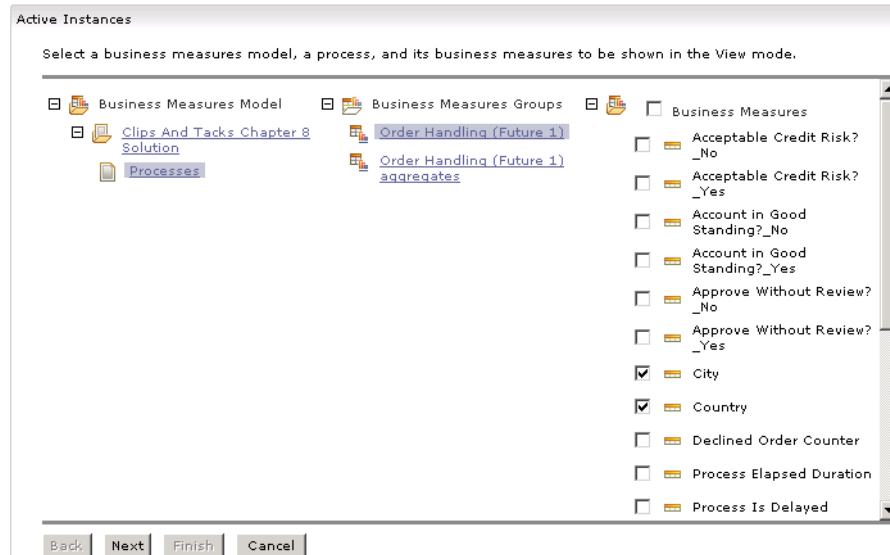
Figure 12-29   Configure the active instances page

► Click *Next* and select your sorting view options.

► Click *Next* and select filters (if any). We will use all defaults here.

► Click *Next* to get a preview of how this view will appear.

► Click *Finish* to accept this configuration and your page view will look as shown
   in Figure 12-30.
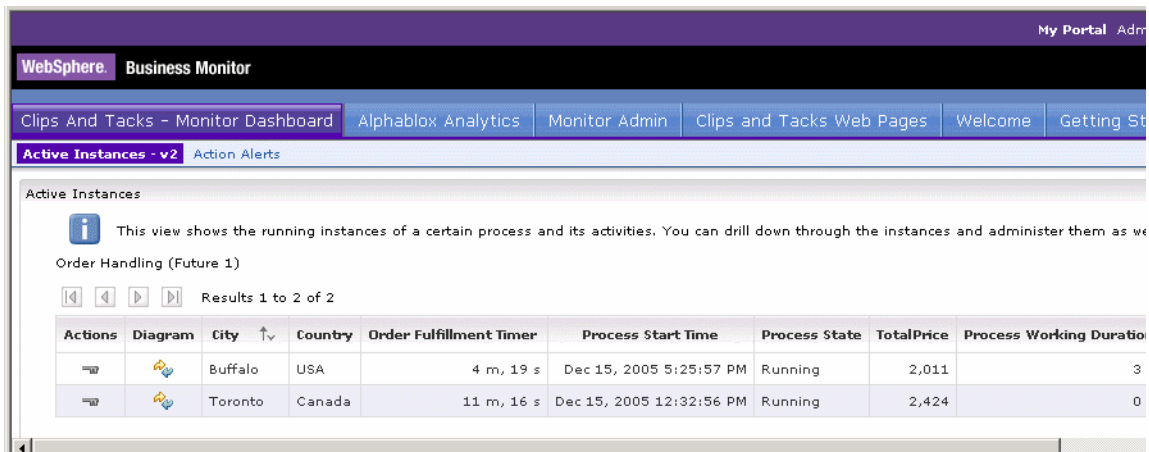
Figure 12-30   Active instances: Portlet view

You will repeat this for all of the views that you wish to configure on any new pages. There will be many layout options and styles. Please consult the WebSphere Portal Server Administration and Users Guide for additional details.

The portlets that we use for ClipsAndTacks are:

► Key Performance Indicators (KPIs)
► Scorecards
► Gauges
► Active Instances
► Dimensions

Overview of using the Monitor Dashboard Client

We end this chapter with a few sample views of what you can expect from the Monitor Dashboard. The real examples for our measurements are shown in "Measuring the process using the Monitor Dashboard" on page 333.

Active Instances view

Figure 12-31 shows the active process instances with an activity drill-down.

\

Figure 12-31   Active Instances: Activity drill-down

Process Diagram view

Figure 12-32 shows the graphical process diagram, similar to how the process diagram was built in the Modeler.
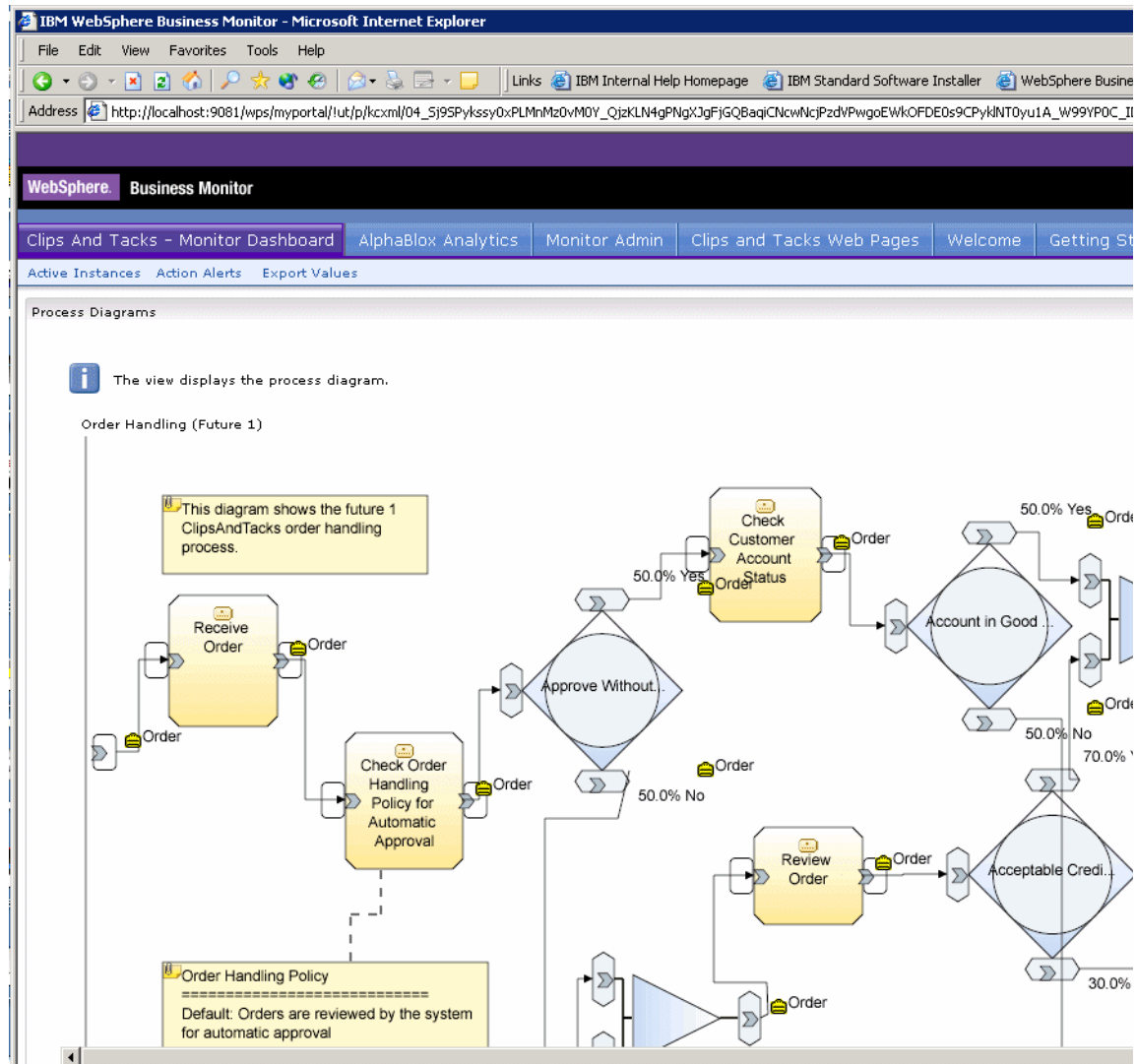
Figure 12-32   Process Diagram view

Other views

The other dasboard views (KPIs, Scorecards, and Gauges) are described in Chapter 13, "Measuring the Future 1 process" on page 329 when we measure the ClipsAndTacks application.

Dimensional analysis

Figure 12-33 shows a DB2 Alphablox Cube view with dimensional analysis data,
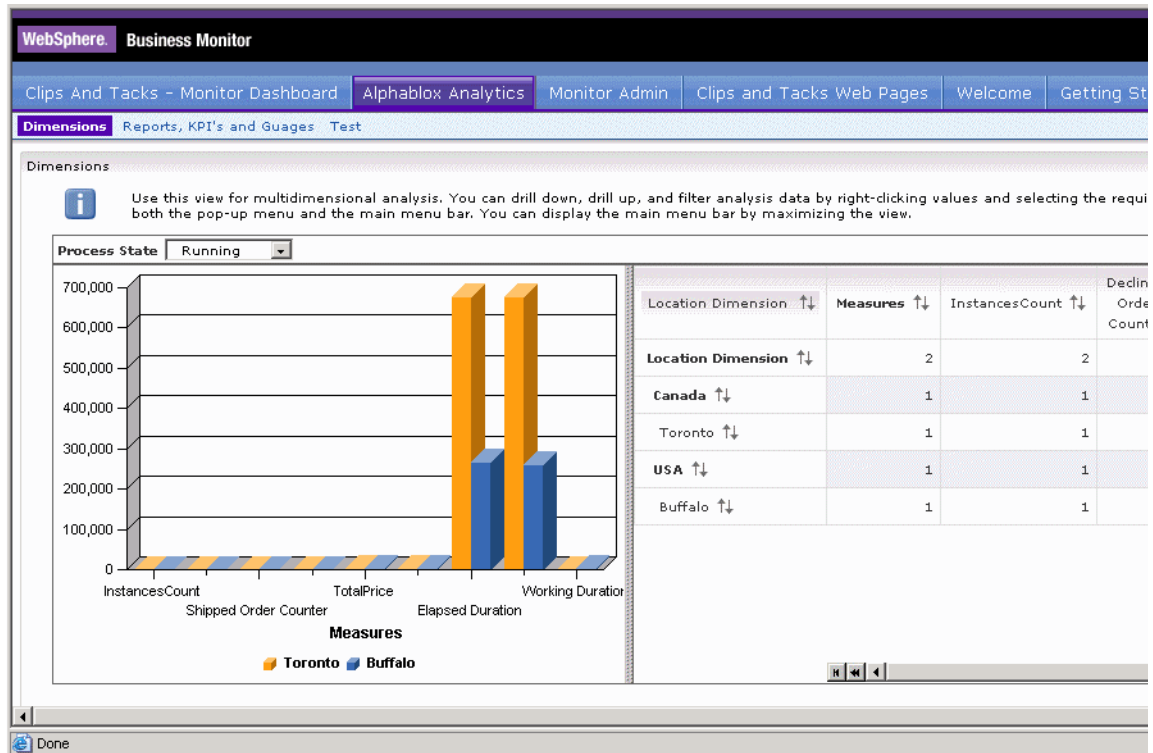
Figure 12-33   Monitor Alphablox Cube Views portlet: Sample view

Monitor Dashboard export

In Chapter 14, "Comparing the measurements and modeling the Future 2 process" on page 357 we will use the dashboard export facility to export runtime Monitor XML data back into Modeler for next cycle in the continuous process improvement lifecycle (Figure 12-34).
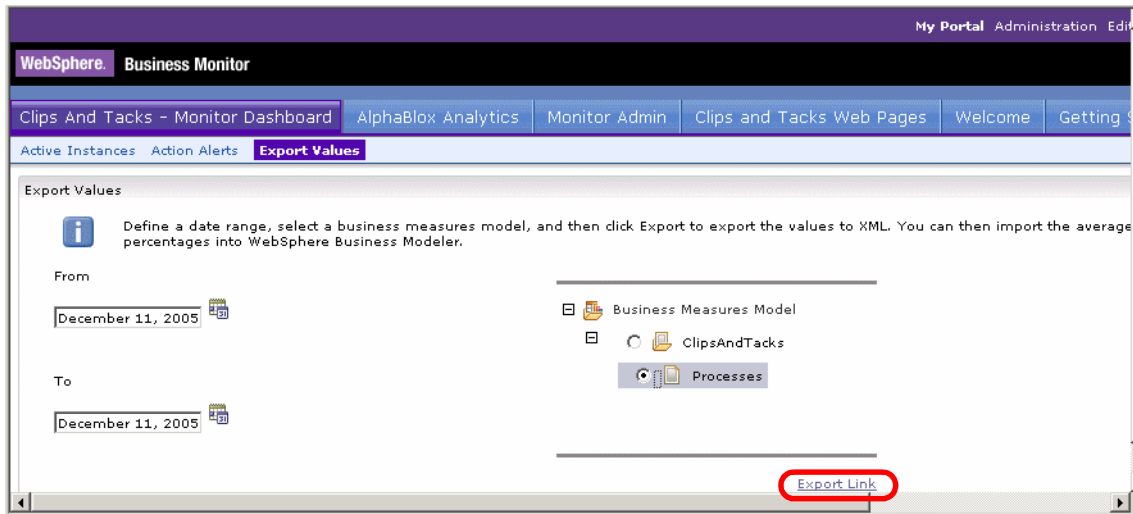
Figure 12-34   Export Monitor data

Running the ClipsAndTacks business process

The section takes us to the point where we collect the runtime data by creating new process instances of the ClipsAndTacks process model within the WebSphere Process Server.

We create new process instances with the ClipsAndTacks Web application and also use the human task management (BPC Explorer or custom human task application) to create different scenario paths through the process. These actions provide the event data that the Monitor Server will receive and process for us to view and analyze the runtime data in the Monitor Dashboard Client using some of the portlets that we customized.

The measurement of the ClipsAndTacks application is described in Chapter 13, "Measuring the Future 1 process" on page 329.

Making changes to the business measures

There is a good chance that your initial KPIs and metrics do not deliver the results that you want to get in the Monitor Dashboard. Without changing the running application you may want to change some of the business measures.

Here are short guidelines for such a change of business measures:

► Use the same Modeler project and do not rename the process. Renaming the process changes the generated activity names.

► Change the business measures using the Business Measures editor.

The following changes are allowed without problems:

– Add a new metric that is not part of a dimensions
– Delete a metric
– Change formula of a metric calculation

Other changes may result in possible problems:

– Change the datatype of a metric—Must be deleted/added in the editor, which results in two metrics in the dashboard

– Change the string length of a metric (only increase is allowed)—May run into database limits

– Changes in indices—Existing indices will be dropped

Changes that require a new observation model:

– A metric becomes a dimension
– Changing the keys of a dimension
– Add and delete of metrics to a dimension

> **Note:** To create a new observation model requires changing the process name and create an application that runs in parallel to the old application until it can replace the old application. A new observation model uses a new set of DB2 tables in the Monitor databases. This is what we will do in Part 5, "Continuous process improvement" on page 355.

► Export the process. Increase the Monitor version number for each update (see Figure 8-38 on page 182). Only keep the `Monitor.zip` file. You do not reimport the interchange file into Integration Developer.

► Backup the Monitor databases (Repository, State, Runtime, History). If anything goes wrong when deploying the updated business measures model you can go back to what you had deployed. Restoring the databases only works if no process application is running (creating events) during deployment.

► Open the administrative console in the Monitor Server. Navigate to *WebSphere Business Monitor → Server → Business Measures Model → Model Unload* (Figure 12-35):

– Select the model to unload and click *Delete* or *Delete and keep for reporting*. The second option keeps the data in the History database.
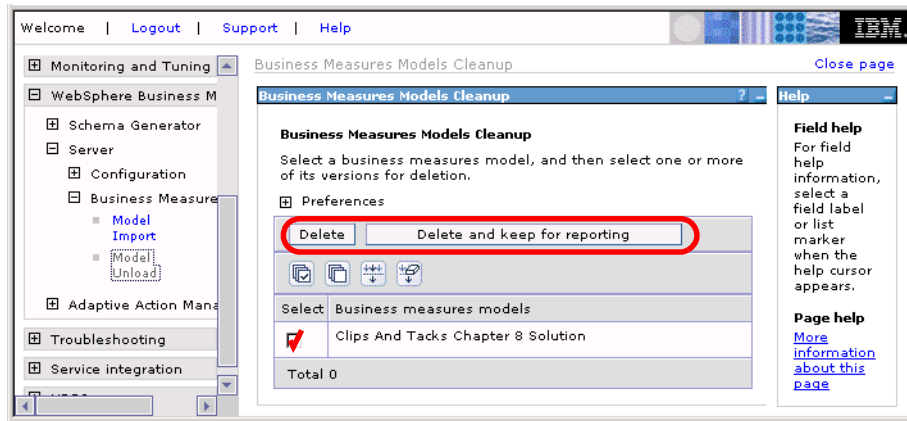
Figure 12-35　Unloading a model

► When the unload is finished, continue with deploying the business measures model as described in "Business measures model deployment to Monitor" on page 300. This includes schema generation, running the database scripts, and importing the model.

► If you encounter errors when running the database scripts you have to backout the changes manually. This is a cumbersome process. Only run one script at a time and verify its outcome before proceeding.

Summary

In this chapter we described how to setup the WebSphere Business Monitor for the ClipsAndTacks business process.

Setting up the Monitor includes deployment of the business measures model and configuring the Monitor Dashboard.

13

Measuring the Future 1 process

In this chapter we describe how we measure the ClipsAndTacks Future 1 process application by submitting orders, approving or declining the orders, and finally shipping the orders.

We describe where to find the information about the process instanced in the DB2 tables used by the Monitor.

After processing about 40 orders we use the Monitor Dashboard to view the KPIs and the metrics we setup for the process. We also use dimensional analysis to compare the metrics based on the location (country and city).

Measurement script

Our test environment is not a real ClipsAndTacks application where numerous customers submit their orders. To gather some meaningful results in the Monitor we developed a script to submit orders and process them through the ClipsAndTacks system using the human tasks for processing of the orders.

Figure 13-1 shows an extract of the script to submit orders:

► A small amount is under $750 and is automatically approved by the business rule.

► A big amount is over $750 and requires the order manager for approval.

► An amount between about $500 and $750 for customer 22222 is automatically approved, but does not pass the credit check and therefore also goes to the order manager.

► We submitted about 40 orders, shipped 33 orders, and declined 7 orders.

```
Order   Customer   Amount    AutoApprove   MgrApprove   Ship-When   Notif-When

1       12345      small     yes                        2 days
2       11111      big       no            yes          2
3       22222      small     yes                        3
4       33333      small     yes                        3
5       44444      small     yes                        3

6       12345      big       no            NO                       2
7       11111      small     yes                        3
8       22222      740       yes           yes          4
9       33333      big       no            yes          4
10      44444      small     yes                        4
```

Figure 13-1   Extract of script to submit orders

Human tasks

We processed the human tasks by claiming the process instances, then waiting about 15 minutes before processing. This resulted in a measured processing time in the Monitor.

We waited with shipping the order between two and four days to get an average of about three days.

The script was designed to give measurements that do not meet the expectations of ClipsAndTacks management so that the process must be improved.

Tracking the orders in the Monitor databases

As described in "Monitor databases" on page 296 the Monitor uses three databases for reporting: State, Runtime, and History. We looked at some of the tables to track the progress of monitoring the process instances.

The names of the tables and columns are generated by the schema generator, but through comments added to the tables we could identify the important tables:

- ▶ One table has a comment of `Order Handling (Future 1)`—This table stores process instance data.

- ▶ One table has a comment of `Order Handling (Future 1) aggregates monitoring context`—This table stores aggregates values such as the KPIs and metrics.

- ▶ In each table we could find the relevant columns through their comments.

Tables

Table 13-1 shows the names of the two tables in each database. Only the prefix of the name is different. The column names are the same in matching tables.

Table 13-1   Monitor tables for Order Handling Future 1: instances and aggregates

STATE	RUNTIME	HISTORY
WBI.**CTX**_AU66A1MBLTK GP5BCDTN5GEXGXY	WBI.**CTR**_AU66A1MBLTK GP5BCDTN5GEXGXY	WBI.**FCT**_AU66A1MBLTK GP5BCDTN5GEXGXY
WBI.**CTX**_YVJEKADLON ZXRIV2LNTSYTFCAA	WBI.**CTR**_YVJEKADLON ZXRIV2LNTSYTFCAA	WBI.**FCT**_YVJEKADLON ZXRIV2LNTSYTFCAA

State instance table

The State table only contains the process instances that are running. We developed a DB2 command to list the table:

```
db2 "select M_4TP2J26MSO21ULEIKTJLICMYKI as orders,
       M_BR2COUJUNTFCHWLKDOPTYGBULQ as shipped,
       M_3L5EJ66JHJNPZP2PTUJ3UKINQM as declined,
       M_WIYQXRWVWXR5BAGZP36SEG33ZI as price,
       M_BY5BZ5HTFRLL5NXNJLESQIELUA as appYes,
       M_ILGDHWLBJIEUQ6YOIIBVGMYA1M as appNo,
       M_BWTSFR2HG2G4ERBGDMQH4ZH4AA as appAuto,
       M_VMQL2EKF5EVK646JHNP3AIICQ3 as accGood,
       SUBSTR(M_BCGKIBDDUHC5TB52LNLQ465HVU,1,8) as state
    from WBI.CTX_AU66A1MBLTKGP5BCDTN5GEXGXY"
```

```
ORDERS SHIPPED DECLINED PRICE     APPYES APPNO APPAUTO ACCGOOD STATE
------ ------- -------- --------- ------ ----- ------- ------- -------
     1       0        0 +2.762E+3      1     0       0       0 Running
```

- – APPYES and APPNO are the actions of the order manager for approval
- – APPAUTO is automatic approval
- – ACCGOOD is account in good standing

The shipped and declined counters are only updated when the instance is complete, and then it disappears quickly from the State database after being replicated to the Runtime database.

Runtime instance table

The Runtime instance table contains the process instances of the last 24 hours.

```
ORDERS SHIPPED DECLINED PRICE     APPYES APPNO APPAUTO ACCGOOD STATE
------ ------- -------- --------- ------ ----- ------- ------- --------
     1       1        0 +3.002E+3      1     0       0       0 Complete
     1       1        0 +1.500E+2      0     0       1       1 Complete
     1       0        1 +6.890E+2      0     1       1       0 Complete
......
     1       0        0 +2.762E+3      1     0       0       0 Running
```

Note that the shipped or declined counter is now updated.

History instance table

The History instance table contains all the process instances. The process state is a numeric value indicating complete (1) or running (3).

```
ORDERS SHIPPED DECLINED PRICE     APPYES APPNO APPAUTO ACCGOOD STATE
------ ------- -------- --------- ------ ----- ------- ------- -----
     1       1        0 +3.002E+3      1     0       0       0 1
     1       1        0 +1.500E+2      0     0       1       1 1
     1       0        1 +6.890E+2      0     1       1       0 1
......
     1       0        0 +2.762E+3      1     0       0       0 3
```

Aggregate tables

The State and Runtime aggregate tables have the same data (after replicating). We developed a DB2 command to list the relevant values:

```
db2 "select M_GJVBXM64SZRZL6VFRO2JPFWOKU as orders,
       M_WNV1IMBIC1WCOKHNMRSUMNPDXI as shipped,
       M_XCRBHPQDRKN5RK5WSS4USORG33 as declined,
       DECIMAL(M_SQ4UXZLTJYW6HBZFX3AWUFA3DQ,8,2) as percentage,
       DECIMAL(M_ZET4V1C2PZF1RKFZEC3WA2OA3Y / 86400000.000,7,3) as time
     from WBI.CTR_YVJEKADLONZXRIV2LNTSYTFCAA"
```

```
ORDERS SHIPPED DECLINED PERCENTAGE TIME
------ ------- -------- ---------- --------
    41      34        7      82.92    3.122
```

► The `PERCENTAGE` column has the comment `90 Percent of Orders Are Approved`, therefore it is our KPI of the percentage of approved orders.

► The `TIME` column has the comment `Average Order Fulfillment Time is 3 Days or Less`, therefore it is our KPI of the order processing time. Note that this time column value is in milliseconds and by dividing by `86400000` we get days.

The History aggregate table does not contains the KPI and metrics.

Measuring the process using the Monitor Dashboard

Measuring the process using the Dashboard Client component of WebSphere Business Monitor Version 6.0 enables users to monitor business performance through a set of views.

The Dashboard Client operates within the IBM WebSphere Portal Version 5.1 environment. Each of the dashboard views is implemented as a portlet. A dashboard is created by assembling portlets in a portal page. To construct the WebSphere Business Monitor Dashboard Client, you create a portal page and add a set of views portlets to it.

Users can display the monitoring data through a set of dashboard views defined according to display properties and content. These views provide different representations of both runtime and historical data. DB2 Alphablox is used with the views for a more refined data analysis.

We describe and navigate through the following dashboard views:

► Runtime views

– Active Instances view
– Process Diagram view

► Historical views

– Scorecard view
– Key Performance Indicators (KPI) view
– Gauges view
– Dimensions view

In additional we provide the steps to create a new view using the Dimensions view as a base.

Navigating through the Dashboard Client with runtime views

The Dashboard Client allows users to view the executing instances, view their associated business measures and perform administrative actions, such as suspend, resume, and terminate a selected instance. Viewing the details of this runtime analysis can help you to closely monitor your business processes.

Active Instances view

The Active Instances view shows the details of a process at run time. It displays information about running instances as they happen.

Using the Active Instances view, you can monitor values of KPIs and metrics that belong to an aggregate business measures group. The available process instances for a selected business measure group are displayed. You can view and monitor the values of their associated KPIs, metrics, stopwatches, and counters. You can drill down on child instances of a parent instance to view the underlying activities, for example, items in process instances, whether they are realized by activities, local subprocesses, or global subprocesses.

The Active Instances portlet is shown in (Figure 13-2). It consists of a Active Instances table that displays the business measures of each currently running process instance.

Figure 13-2   Active Instances view portlet

Details of an active instance

Additional measures can be viewed, for example when you click on the small arrow at the Activities column ⌐ you will *drill down* to the detail view of each activity of the process instance (Figure 13-3).

Figure 13-3   Active Instances view: Details

In the first column you can see the Activity State for the select process instance. We have two activities already completed and one ready to work on. Notice that this activity has not been started yet, indicated by the date of `Dec 31, 1969` in the Activity Start Time column. Once a user works on this activity, the Activity State is set to `Running` and the Activity Start Time is a current value (Figure 13-4).
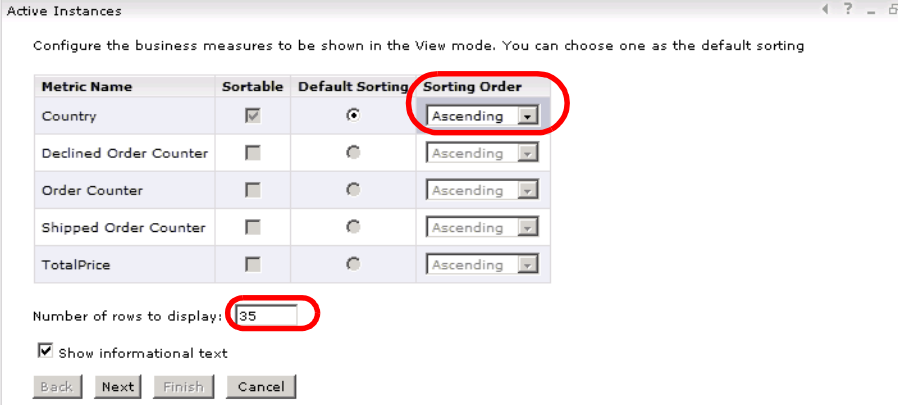
Figure 13-4   Detail Active Instances view per activities

Sorting active instances by business measures

In the Dashboard Client, a user can customize their view of the Active Instances portlet by sorting different measures and setting the number of rows to display (Figure 13-5). To sort the activities on the Active Instances portlet click the *Edit* icon 🖉 .

Figure 13-5   Sorting active instances by business measures

Filtering active instances by business measures

You can control the instances displayed in the Active Instances view by setting a filter expression in configure mode or edit mode. When you limit the number or type of instances in a view, you can focus on a particular area of interest.

In the filter dialog (Figure 13-6), you can display instances that satisfy certain criteria that you define during the configuration or the personalization of the Active Instances view. By setting filtering criteria based on one or more business measures, you can determine which process instances will be displayed in view mode in the Active Instances view.

You can build a formula of business measures and specific values that represents the filter expression. The filter expression can be simple or complex:

► A simple expression consists of a filter criterion based on only one business measure.

► A complex expression includes filter criteria based on multiple business measures, combined together by boolean operators, such as AND, OR, and NOT.

Each criterion in the filter expression is defined by specifying a value—or a range of values—for the measure you select. The values for measures are based on the nature of each business measure.

For example, a numeric business measure requires a numeric value and a logical operator. A duration business measure requires an operator (for example, <,<=...) and a duration (number of days, hours, minutes).
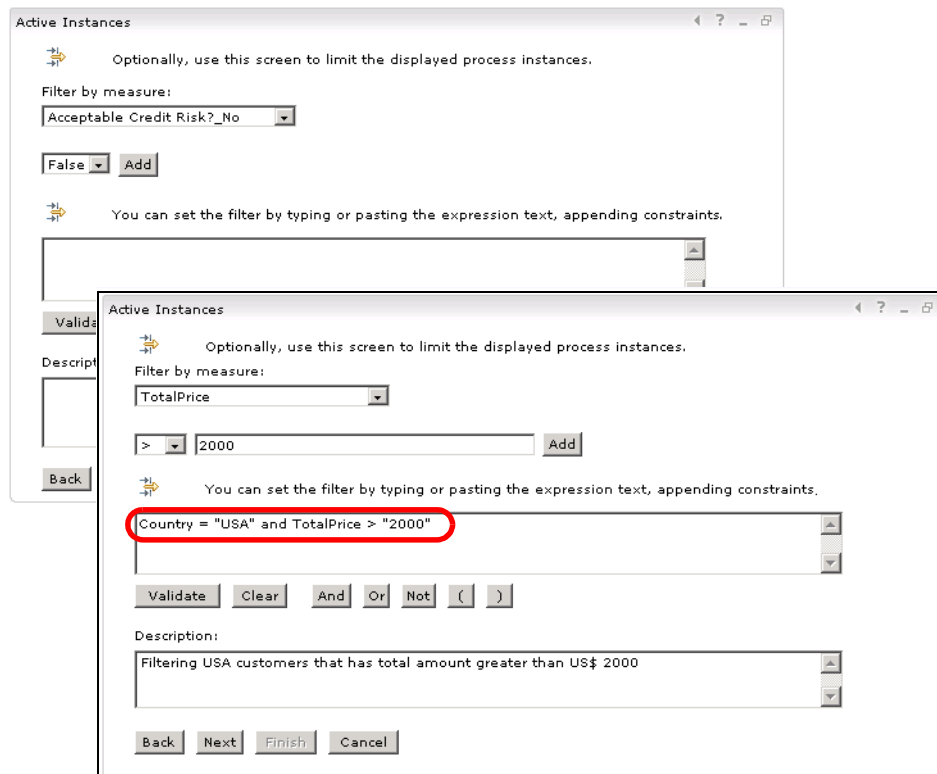
Figure 13-6   Filtering active instances

To create the sample filter of Figure 13-6 follow these steps:

► Click the *Edit* icon ✏ and navigate to the filter configuration page.

► Create a filter by selecting *Country* for Filter by measure, *Matching* in the drop-down list, and for the value type USA. Click *Add.*

► Add an *And* condition to the expression by selecting *Total Price* for Filter by measure, *> (greater than)* in the drop-down list. and for the value type 2000. Click *Add.*

► Click *Validate* and enter a description.

Figure 13-7 shows the result of filtering the active instances.
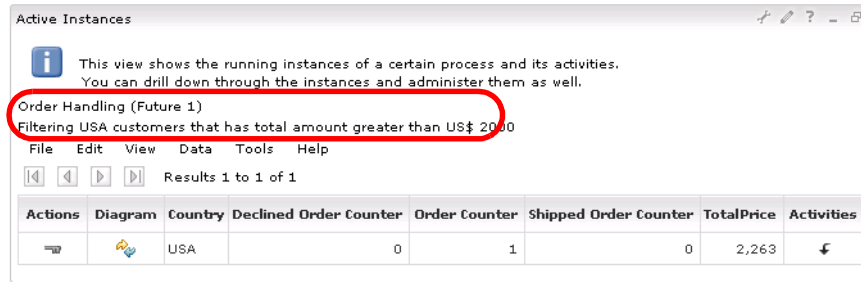
Figure 13-7   Active Instances view: Filtered

Process Diagram view

The Process Diagrams view displays process diagrams and process instance diagrams that visually illustrate the work flow of a business process.

This view requires a business measures model that contains one or more modeled processes. A compatible scalable vector graphics (SVG) viewer to view diagrams in the Process Diagrams view must be installed.

SVG is a royalty-free vendor-neutral open standard developed under the W3C (World Wide Web Consortium) process. For SVG implementations, go to:

http://www.w3.org/Graphics/SVG/SVG-Implementations

A process model is a representation of the business processes occurring at runtime. If the process model contains subprocesses, you can drill down to view the underlying items.

The Process Diagrams view displays diagrams of top-level processes. Objects within a process diagram include tasks, processes, connections, business items, and decisions. The numbers of ready and running instances of the processes are annotated on the displayed diagram.

Wiring views

You can also have the activities of one process instance highlighted. For this you have to establish a wiring connection between the Process Diagrams view and another view that displays process instances. When an instance is selected from a source view, the Process Diagrams view identifies the process instance activities whose states are ready, running, or completed, and marks them with a red frame. Valid wires are from the Alerts view to the Process Diagrams or Active Instances view, and from the Active Instances view to the Process Diagrams view.

When you click the *Diagram* icon 🐾 in the Active Instances view you can see the highlighted activities in the Process Diagram view (Figure 13-8).
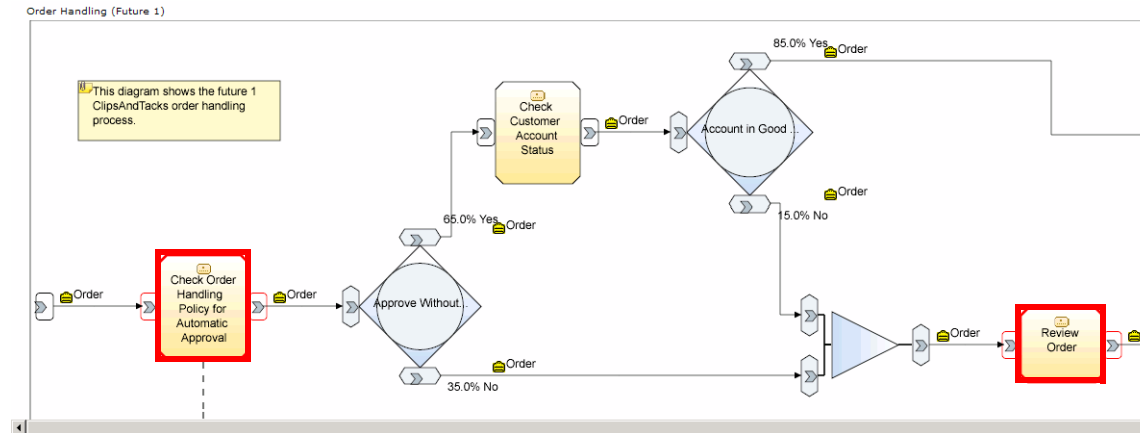
Figure 13-8   Process Diagram view with highlighted activities

Navigating through the Dashboard Client with historical views

The Monitor collects information from active instances and calculates the KPIs and metrics that were defined in the business measures of the process. In addition the instances can be aggregated against the predefined dimensions.

In Chapter 8, "Creating the business measures model" on page 145 we defined these KPIs, metrics, and dimensions:

▶ KPI—Average Order Fulfillment is 3 Days or Less
▶ KPI—90 Ppercent of Orders Are Approved
▶ Metric—Total Orders
▶ Metric—Shipped Orders
▶ Metric—Declined Orders
▶ Dimension—Location (country and city)
▶ Aggregate—Total Price

In the Monitor Dashboard we can view the KPIs, metrics, and perform dimensional analysis of the metrics and aggregates against the location dimension.

Note: Depending of how you configure the KPI attributes of targets and limits, the KPIs and Scorecards views represent the values using different icons. See Figure 8-4 on page 149 for more details.

Key Performance Indicators (KPI) view

The KPIs view (Figure 13-9) displays all KPI information so that business users can easily monitor them and take action if necessary.

KPIs are quantifiable measurements of the improvement or deterioration in the performance of an activity critical to the success of a business. These measurements break down key areas of your business so you can see how they contribute to business results.

When selecting KPIs to monitor, you should choose them to reflect the goals of your business, be critical to its success, and allow for corrective action by early detection of problems. The KPIs view displays KPIs that represent aggregate business measures. You must have a business measures model that contains the KPIs that you want to monitor, with defined upper and lower limits.

Figure 13-9   Key Performance Indicators view

As you can see, the percentage of approved orders is 81.48% and the average order fulfillment time is 3 days and 4 hours. The red arrow in the Status column indicates that the result is above limit specified by the management team. The analysis of this information will help ClipsAndTacks to better understand that situation and act to achieve an order fulfillment time of 3 days or less.

Note that the KPIs view only indicates the status against a limit, but not against a target.

Scorecards view

The Scorecards view (Figure 13-10) allows users to monitor groups of KPIs, based on some configurable business perspectives, where for each KPI information is available for the KPI relative to its target value.

A scorecard is a set of performance measures that are linked to objectives and goals of a specific business area. Business users select the KPIs pertaining to their various areas of responsibility and place them in perspectives (categories) on their scorecards. On the Scorecards view, users can easily watch the runtime values of vital numeric KPIs, monitoring them against their defined targets.
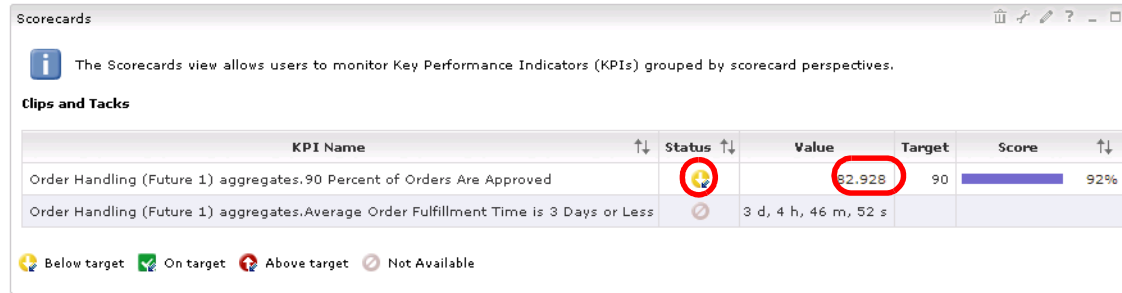
Figure 13-10   Scorecards view

As you can see the number of approved orders is 81.48%, well bellow the target of 90% specified by the management team to monitor the percentage of approved orders. The analysis of this information will help ClipsAndTacks to better understand that situation and act to achieve an order approval rate of 90% or better.

Note that the Scorecards view only indicates the status against a target, but not against a limit.

Gauges view

The Gauges view (Figure 13-11) displays individual KPIs using a graphical representation of gauges or dials.

Gauges help you to visualize information by representing KPI values on a gauge, such as a speedometer. A dial is used to represent the position of the KPI value in two different ways. It can be monitored relative to the lower and upper limits of the KPI, or relative to the target of the KPI. The Gauges view focuses on representing numeric KPIs that belong to aggregated business measures in a business measures model. Each gauge represents the value of a single KPI.

Both gauges for our two KPIs indicate that the target or limit was not reached.

Figure 13-11   Gauges view

> **Note:** The Average Order Fulfillment Time is 3 Days or Less KPI is defined as a type of *Duration*, so when defining the lower and upper limits you have to use milliseconds, for example, 5 days equals 432,000,000 milliseconds.

Dimensions view

Using the Dimensions view users can generate multidimensional reports that analyze different business aspects of historical business-performance data. Charts and grids present data for analysis against different dimensions.

A dimension is a conceptual axis over which a business is analyzed. For example, a retail business performance might be analyzed by time, products, and stores, that is, time, products, and stores are dimensions for such a business. Each of the dimensions has one or more levels that together define the overall hierarchy of the dimension. The time dimension might have year, quarter, and month levels.

There are some predefined dimensions, such as the process state, with values of `Completed` and `Running`.

Figure 13-12 shows the Dimensions view with the metrics of total, shipped, and declined orders against the process state.
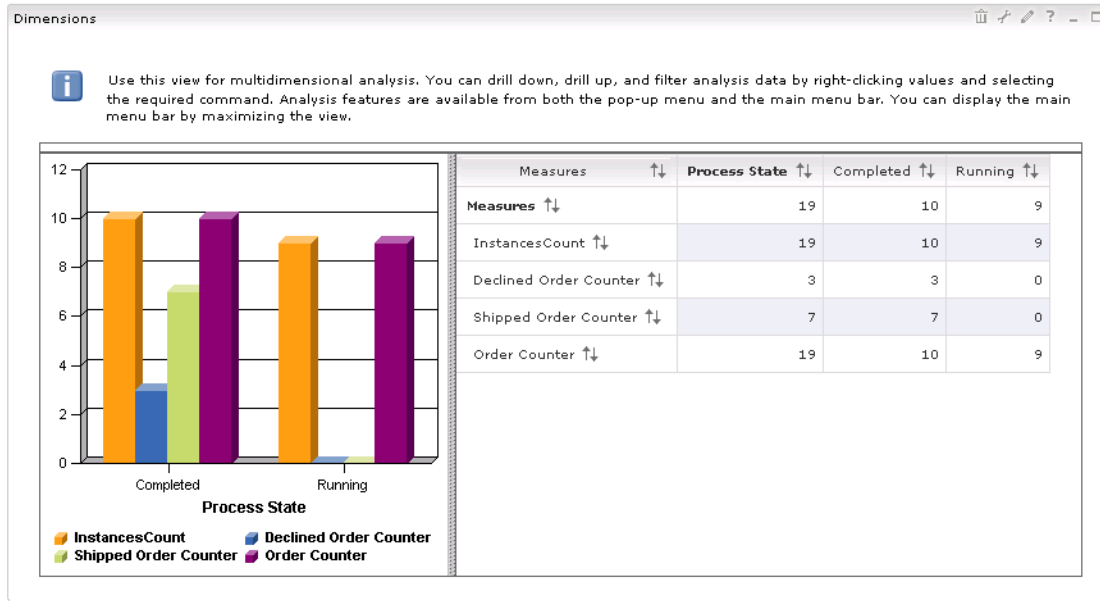
Figure 13-12   Dimensions view

You can choose from a wide variety of charts to display data (Figure 13-13).
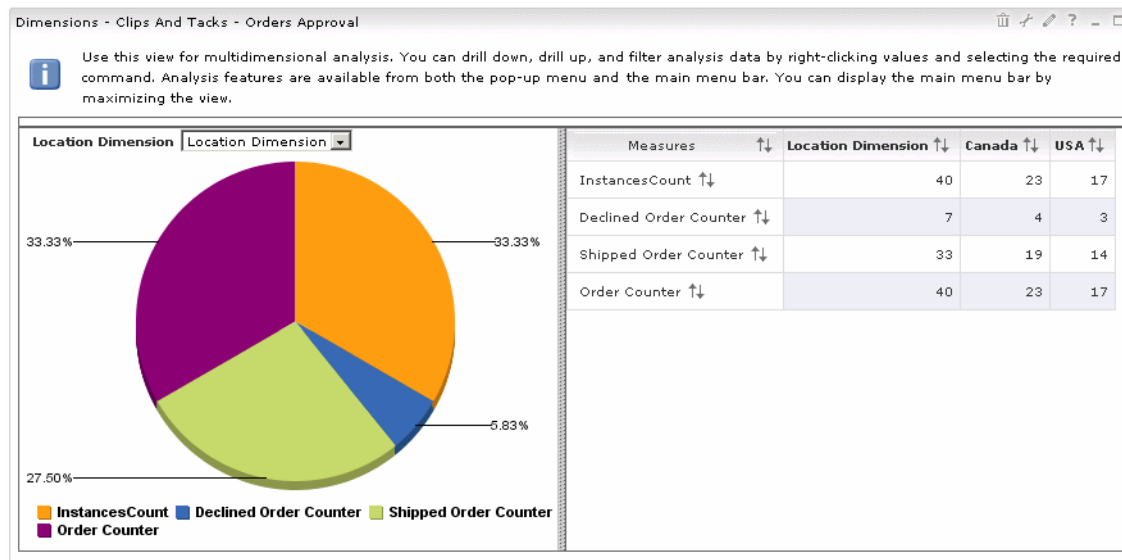
Figure 13-13   Dimension view using a pie chart

For example, using dimensional analysis we can answer questions such as Where are orders coming from? (Canada or USA). You can use information related to the business items of the process by assigning them to measures elements(Figure 13-14)
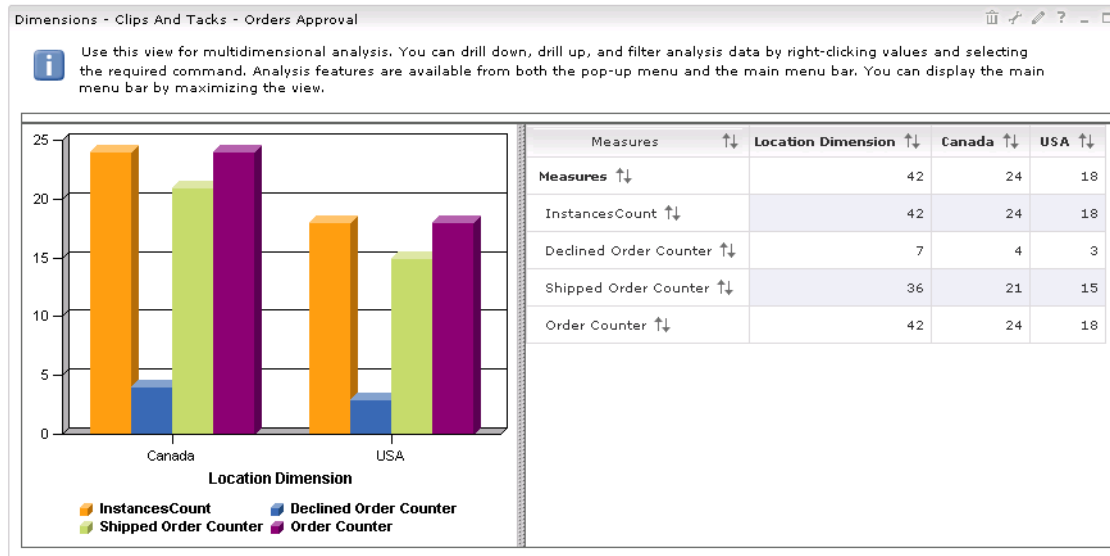
Figure 13-14   Dimension view using the location

Creating a new Dimensions view

To provide a instant view of the overall approval and shipping of the orders for our example, we will create a new view using the Dimensions view portlet as base, and include the new view into a portal page.

Open the home URL of the Monitor Dashboard machine:

```
http://<your-dashboard-machine>:9080/wps/portal
```

Creating a portlet

After login, select *Administration, Portlet Management → Portlets* and search for the Dimensions portlet. You can use the search portlet functionality or just navigate through the Manage Portlets interface to see the Monitor Dashboard portlets (Figure 13-15).
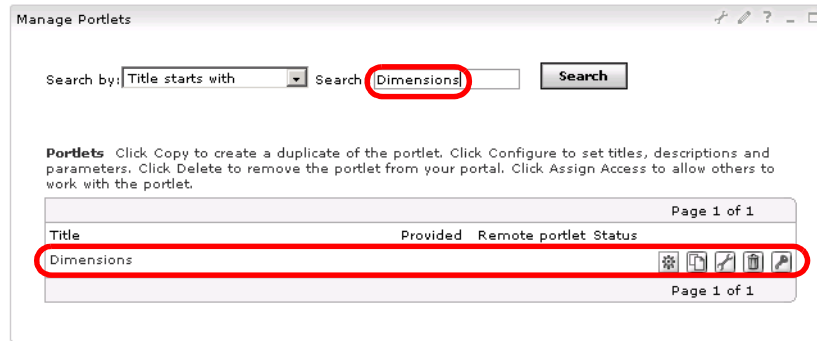
Figure 13-15   Manage portlets

► Click the *Copy Portlet* icon to make a copy of the Dimensions portlet, then provide a new name for the portlet:

– Portlet application: `WebSphere Business Monitor v6.0 - Dashboard Client Concrete Application - Clips And Tacks`

– Portlet: `Dimensions - Clips And Tacks - Orders Approval`

► Click *OK* and wait for the confirmation message (Figure 13-16).
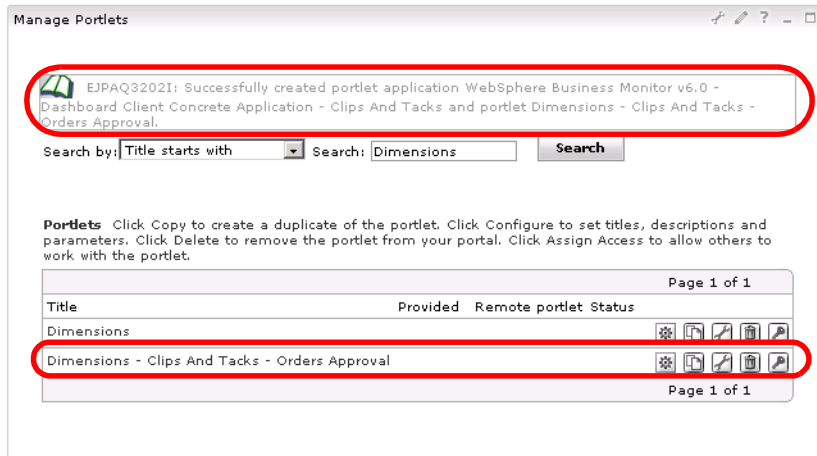
Figure 13-16   Copying a portlet

► Select *Portal User Interface* → *Manage Pages* to add the new portlet to a page so that users can see it.

In our scenario we add the new portlet in a predefined page called *Dimensions*. You can either create a new page or use a page that you already defined.

▶ Click the *Edit Page Layout* icon 🖉 to add the new portlet (Figure 13-17)
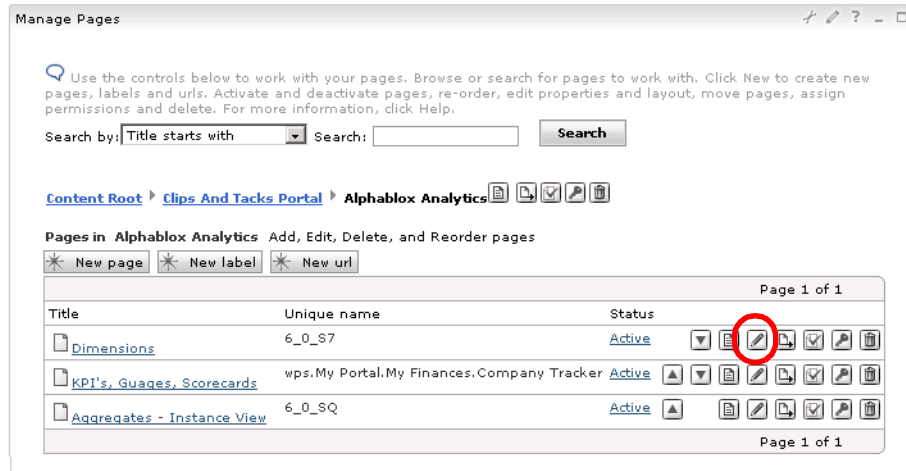
Figure 13-17   Edit a portal page

▶ In the Edit Layout page click the *Add Portlet* icon [➕ Add portlets] , then search for the new portlet.

▶ Select the *Dimensions - Clips And Tacks - Orders Approval* portlet and click *OK* to add it to the page, then click *Done*.

Configuring the portlet

Next we have to configure the portlet to present information about our business measures, such as total orders and shipped orders. Navigate to the page that you have added the portlet and click on the on the configure icon 🔧 at the top right of portlet.

▶ Select the dimension you want to view for this page, configure using the following information (Figure 13-18):

   – Subject area: *Order Handling (Future 1)*
   – Row dimension: *Measures*
   – Column dimension: *Location Dimension*
   – Page dimension: empty

▶ Click *Next*.
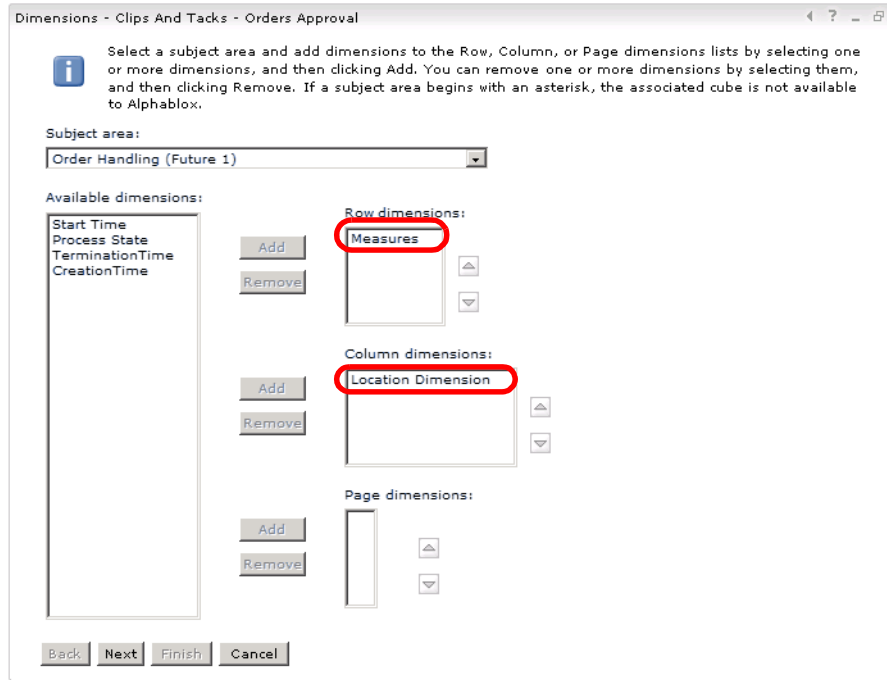
Figure 13-18   Configuring dimensions: Rows and columns

► We drill down the measures by double-clicking *Measures* on the (left or right side) or by selecting *Measures → Drill-Down* (Figure 13-19).
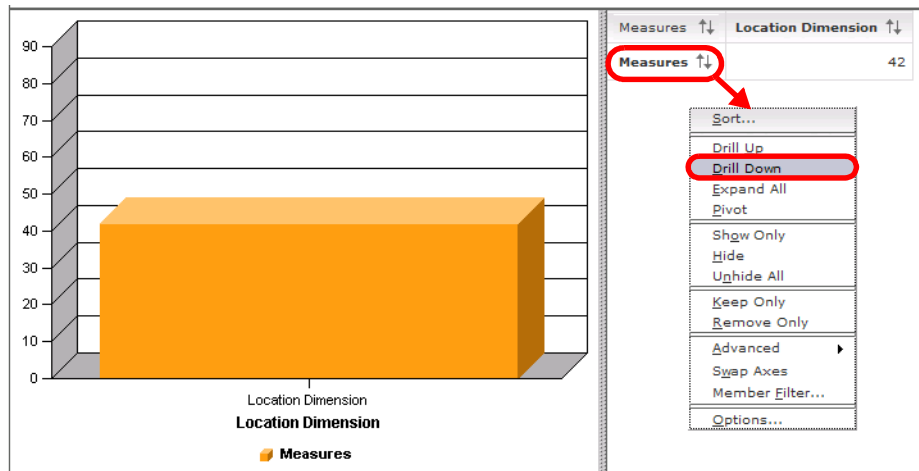
Figure 13-19   Configuring dimensions: Drill-down of measures

► The measures are displayed graphically and with values (Figure 13-20). Because there is only one scale, most values are too small to be shown in the graphic.
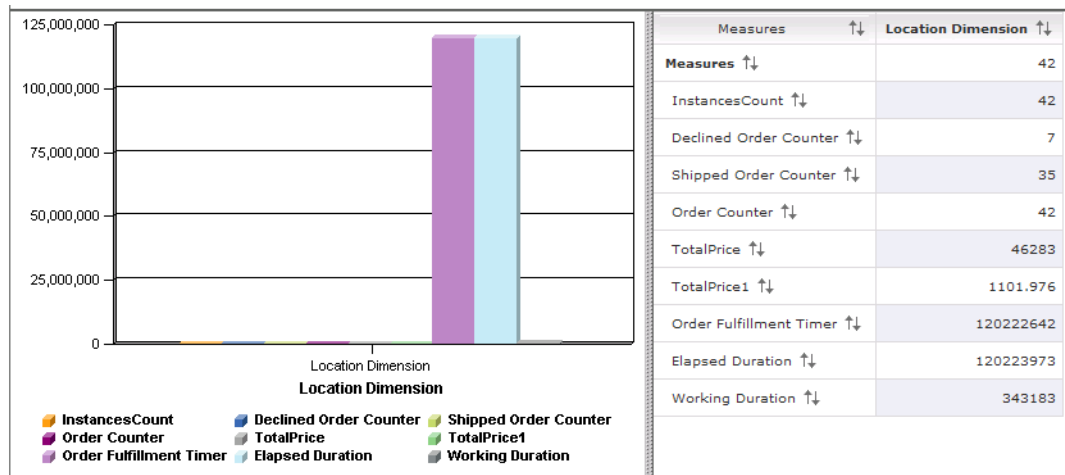
Figure 13-20   Configuring dimensions: Measures

> **Note:** There are two values labeled `TotalPrice` and `TotalPrice1`. In Figure 8-28 on page 174 we configured two aggregates for total price: *Sum of total price* and *Average of total price*. The `TotalPrice` measure is the sum, and the `TotalPrice1` measure is the average.
>
> This is a known defect in the Monitor that our assigned names are not displayed and will be fixed in a future update.

► Select *Instances Count → Remove Only* (context menu), then repeat this for *Order Fulfillment Timer*, *Elapsed Duration*, and *Working Duration* to remove them from the graphic.

► Select *Location Dimension* and *Drill-Down* (right side) to expand the location into countries, Canada and USA (Figure 13-21).

► We could drill-down the countries into cities, but we can do that later when we display the results.
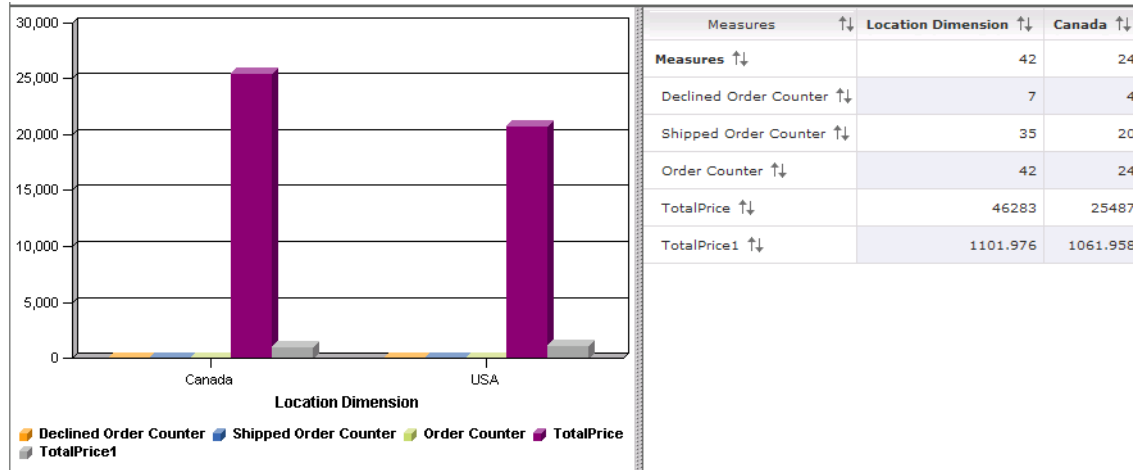
Figure 13-21   Configuring dimensions: Select measures and expand location

▶ Click *Finish* and the portlet is configured.

Working with the Dimensions view

To display nice graphical representations of our measures we have to select either an individual measure or multiple measures that fit into the same scale.

Average price per city

Select the `TotalPrice1` measure and *Show Only*. Select `Canada` and `USA` and *Drill-Down*. The graphic displays the average price per order for each city (Figure 13-22).
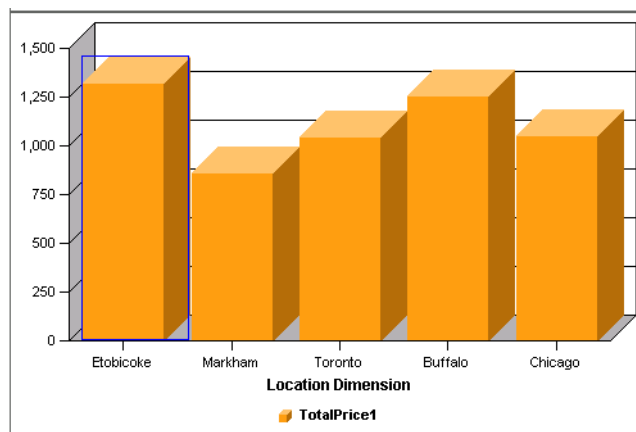
Figure 13-22   Average price per city

Orders total, shipped, and declined per city

Select `TotalPrice1` and *Unhide All* to get all the measures back. Then select `TotalPrice` and `TotalPrice1` and *Hide*. This leaves all the counters (Figure 13-23).
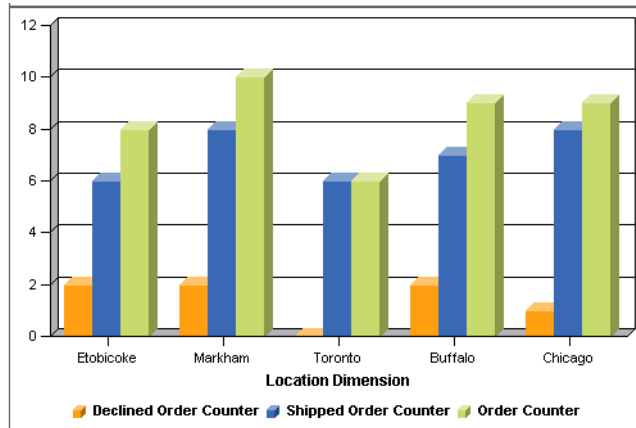
Figure 13-23   Orders per city

We can see all the measure data on the right side after unhiding all measures (Figure 13-24).

Measures ⇅	Location Dimension ⇅	Canada ⇅	Etobicoke ⇅	Markham ⇅	Toronto ⇅	USA ⇅	Buffalo ⇅	Chicago ⇅
Measures ⇅	42	24	8	10	6	18	9	9
Declined Order Counter ⇅	7	4	2	2	0	3	2	1
Shipped Order Counter ⇅	35	20	6	8	6	15	7	8
Order Counter ⇅	42	24	8	10	6	18	9	9
TotalPrice ⇅ Sum	46283	25487	10604	8600	6283	20796	11304	9492
TotalPrice1 ⇅ Average	1101.976	1061.958	1325.5	860	1047.167	1155.333	1256	1054.667

Figure 13-24   All measures by location

You can use the double-arrow to sort the table by one of the measures.

Using the time as a dimension

Figure 13-25 shows a graphic where the start time of the order handling process is used as a dimension. We expanded the year into months and displayed the shipped orders as a function of the starting time.
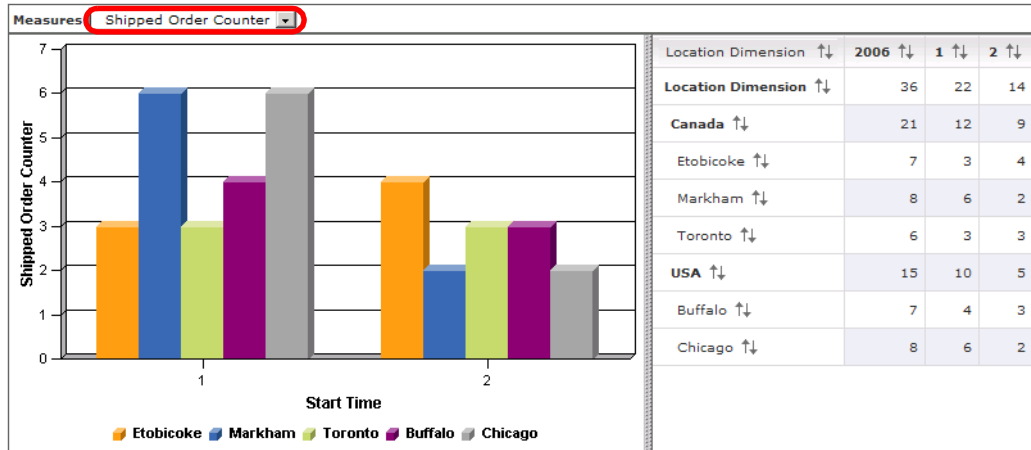
Figure 13-25   Shipped orders per month and city

We configured this graphic using the location as rows, start time as columns, and measures as the page dimension (Figure 13-26). Then we drilled-down location and start time, and removed all years except 2006, expanded again, and removed all the months except January and February.
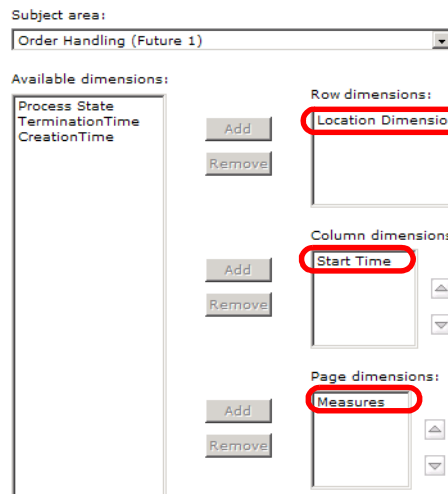
Figure 13-26   Customize a graph using start time

Viewing an action alert

In "Configuring alerts for the adaptive action manager" on page 309 we defined two alerts to be triggered based on the situation events we defined in "Situation events" on page 176.

One of the alerts is triggered when the average order processing time is above `3 days 1 hour`. The alert is sent to the order manager `andre`.

When `andre` logs into the dashboard, he will see the alerts in the Alerts view (Figure 13-27).
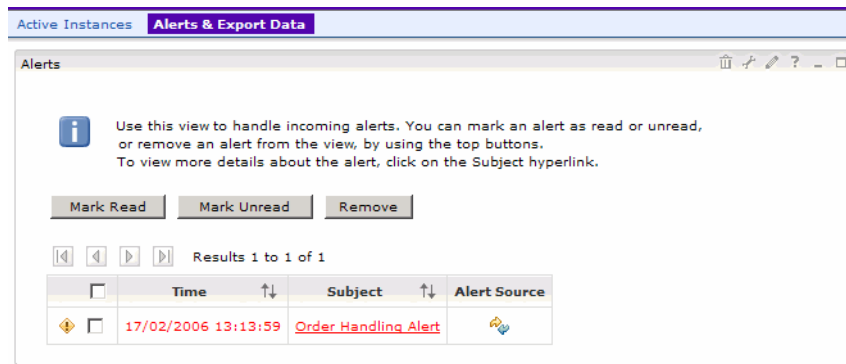
Figure 13-27   Alerts view: Action alert

Click on the alert to see the details (Figure 13-28).
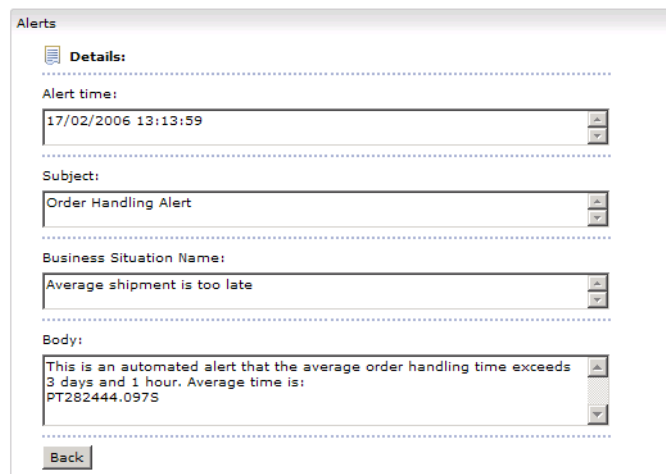
Figure 13-28   Action alert details

Notice the text that we configured in the notification template (Figure 12-17 on page 310). The variable `%Average Order Processing Time%` has been replaced with the current value of `PT282444.097` seconds. This is approximately 3 days and 6 hours. We should have used a formula to calculate the time in days when we defined the business measure.

After viewing the alert, the red color and the warning sign disappear. By clicking *Mark Unread*, the color and warning sign are displayed again.

Wiring the Alerts view

Notice the 🔧 icon in the Alert Source column. The Alerts view can be wired to the Active Instances or Process Diagram view to display the process that triggered the alert, when the icon is clicked. The target view must be configured into the same portal page.

LDAP configuration error

If your LDAP configuration in incorrect (bad user ID or password, or bad query) you will not receive the alerts and error message are written to the Monitor server log file, for example:

```
[2/17/06 11:25:23:750 PST] 0000001c AlertActionHa I
com.ibm.wbimonitor.actionmgr.handling.AlertActionHandler
sendAlert(CommonBaseEvent, Properties) CWMAA0004I: No user ids for Alerts
found for LDAP query: select users.user_id from users where
(&(objectClass=top)(uid=andre)) on root: "" from field: uid.
   Handler: AlertHandler
   Template Name: Late Order Shipped Alert
   Template Id: yjvLltU82/nv19dKtkD5qA==
   Situation Event: Average shipment is too late
```

The query does not need a SELECT clause!

Database replication problem

By default database replication only goes back for seven days. If you do not have the replicators running for seven days, then you get an error at the start of the capture replicator:

```
ASN0121E CAPTURE "CAPTURE_141" : "WorkerThread". The Capture program warm
start failed because existing data is too old. The Capture program will
terminate.
```

You can overwrite the default time limit in the capture command batch files. See "Starting the database replicators" on page 306 for the location of the batch files.

You have to edit the two data capture files (`StartCapture_x.bat`) and change the command:

```
db2cmd asncap CAPTURE_SERVER=STATE CAPTURE_SCHEMA=CAPTURE_xxx
CAPTURE_PATH="C:\IBM\WebSphere\Monitor\rm\logs\s2r\capture"
```

If you know when you last run the command, calculate the time in minutes as `days*1440` (for example `20160` for `14` days), then change the command to:

```
db2cmd asncap CAPTURE_SERVER=STATE CAPTURE_SCHEMA=CAPTURE_xxx
CAPTURE_PATH="C:\IBM\WebSphere\Monitor\rm\logs\s2r\capture" LAG_LIMIT=20160
```

The actual value when the command ran the last time can be found in the table `CAPTURE_xxx.IBMSNAP_CAPMON`, column `SYNCHTIME`, in the State and Runtime databases. To calculate the minimum number of minutes for the `LAG_LIMIT` parameter, you can issue this SQL statement against the table:

```
SELECT TIMESTAMPDIFF(4, CHAR(CURRENT TIMESTAMP - MAX(SYNCHTIME))) as
Minimum_Minutes FROM CAPTURE_xxx.IBMSNAP_CAPMON;
```

Alternatively you can issue a cold start of the capture commands, however, this will replicate all the data in the tables:

```
db2cmd asncap CAPTURE_SERVER=STATE CAPTURE_SCHEMA=CAPTURE_xxx
CAPTURE_PATH="C:\IBM\WebSphere\Monitor\rm\logs\s2r\capture"
STARTMODE=warmsa
```

You can find information about the database replication configuration in the Monitor Information Center at:

http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp

Summary

In this chapter we described the measurement of the Order Handling (Future 1) process.

Because our application is not a real application with many customers we used a script to submit orders to get some meaningful results.

We used the Monitor Dashboard to view and analyze the results. These results will enable ClipsAndTacks management to take action to improve the key performance indicators.

Part 5

Continuous process improvement

In this part we describe how to improve the business process based on the measurements performed using WebSphere Business Monitor.

We compare the results of the Monitor with our simulation results in the Modeler. Then we change the process to improve the order turn-around time. We export the model once more and implement the new model in WebSphere Integration Developer. Finally we deploy the model to WebSphere Process Server and monitor the new application.

14

Comparing the measurements and modeling the Future 2 process

This chapter describes how we use the measurements from the WebSphere Business Monitor to improve the business process.

We export measurement results from the Monitor and import them into the Modeler. After comparing the Monitor results with the Modeler simulation we decide what changes should be made to the business process.

We implement the proposed changes and create the *Order Handling (Future 2)* process. We simulate and analyze the new process to predict the improvements.

One of the major changes is to add a customer classification of regular, silver, and gold, and use this classification to approve more orders automatically. We also update the business measures and use the classification for dimensional analysis.

Export data from the Monitor

After completing the measurements of the Future 1 process we export the data from the Monitor into an XML file (Figure 14-1).
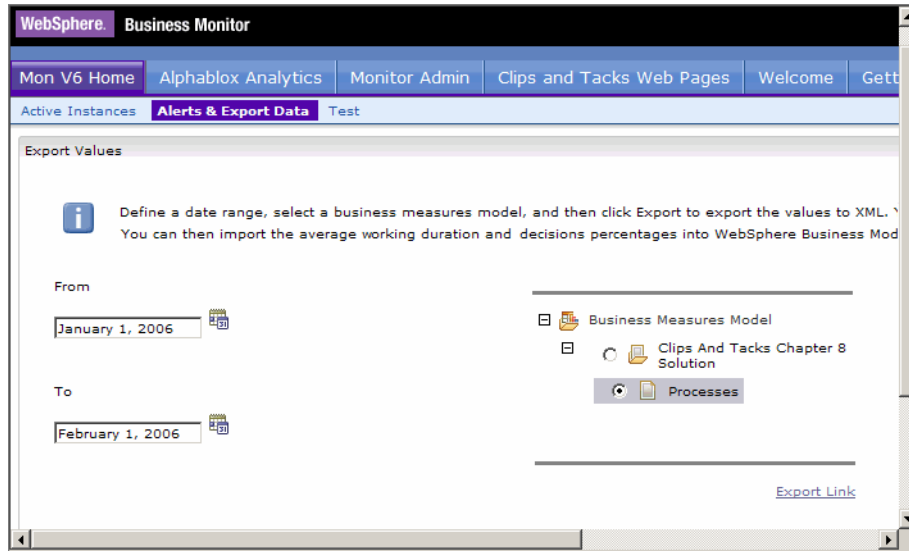
Figure 14-1   Exporting Monitor measurement data

You are prompted for a location and name of the output file, for example:

    SG247148\sampcode\monitor\Future1Export.xml

The output file contains:

► Process elapsed duration (193177114) and working duration (1194)

► Durations for all tasks (all durations are in milliseconds)

► Percentages for all decision blocks

The XML file procuded by the Monitor is shown in Figure 14-2. The values are not easily matched to the tasks in the Modeler process diagram. The internal uId (unique ID) are not visible in the Modeler diagram or Attributes view, they are stored in the underlying XMI files.

```
<?xml version="1.0" encoding="UTF-8"?><!--Licensed Material -
   Property of IBM 5724-M24 ..........>
<runtimeData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   fileVersion="6" name="Clips And Tacks Chapter 8 Solution"
   xsi:noNamespaceSchemaLocation="ActualValuesSchema.xsd">
 <process elapsedDuration="193177114" qualifiedName="NA" type="Process"
      uId="BLM-6647c2d7b35fab7b5f82e2272db98ea0" version="1"
      workingDuration="1194">
   <activity elapsedDuration="0" qualifiedName="NA" type="Task"
      uId="BLM-17c7ecc253684edfd8d9d0b0e97c602c" workingDuration="165321"/>
   <activity elapsedDuration="839" qualifiedName="NA" type="Task"
      uId="BLM-664f46d66a10c31fcd839711e818e16c" workingDuration="839"/>
   <activity elapsedDuration="1131" qualifiedName="NA" type="Task"
      uId="BLM-370ed2d5f3e2b2c3b183d99d8c7dbf6e" workingDuration="1131"/>
   <activity elapsedDuration="0" qualifiedName="NA" type="Task"
      uId="BLM-d8ea62bb5541154703ac82f66504025f" workingDuration="852123"/>
   <activity elapsedDuration="0" qualifiedName="NA" type="Task"
      uId="BLM-788ae979e10fc50e8929950676ea2dd3" workingDuration="1020123"/>
   <activity type="Decision" uId="BLM-e3964f67f2fb2decce3048a333a20b57">
      <path outputPinSetName="NA"
         outputPinSetUId="BLM-e3daf6c08b4973bb43cb3208c4777852"
         percent="42.5"/>
      <path outputPinSetName="NA" outputPinSetUId="..."
         percent="57.5"/></activity>
   <activity type="Decision" uId="BLM-2340071d778b45c931fed2346a3fd7c4">
      <path outputPinSetName="NA" outputPinSetUId="..."
         percent="21.73913"/>
      <path outputPinSetName="NA" outputPinSetUId="..."
         percent="78.26087"/></activity>
   <activity type="Decision" uId="BLM-76b37c32fdba08b97d30c33936083e66">
      <path outputPinSetName="NA" outputPinSetUId="..."
         percent="31.81818"/>
      <path outputPinSetName="NA" outputPinSetUId="...b"
         percent="68.18182"/></activity>
 </process>
</runtimeData>
```

Figure 14-2   Monitor export XML file

Important: The XML file was manually updated with valid percentage values. The Monitor currently calculates the percentages so that the total of all six values is 100%, instead of making the total of each decision equal to 100% as required by the Modeler.

Import data into Modeler

We can import the Monitor output into a Modeler project and update the model with the measured values:

▶ In the Modeler select the project Clips And Tacks Chapter 8 Solution and *Import* (context).

▶ Select *Monitor result (.xml)* and click *Next* (Figure 14-3).
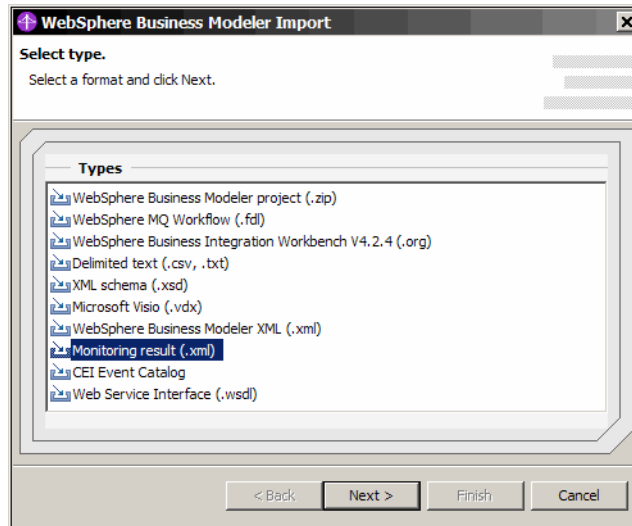
Figure 14-3   Modeler: Importing Monitor results (1)

▶ Select the exported file and click *Next* (Figure 14-4).
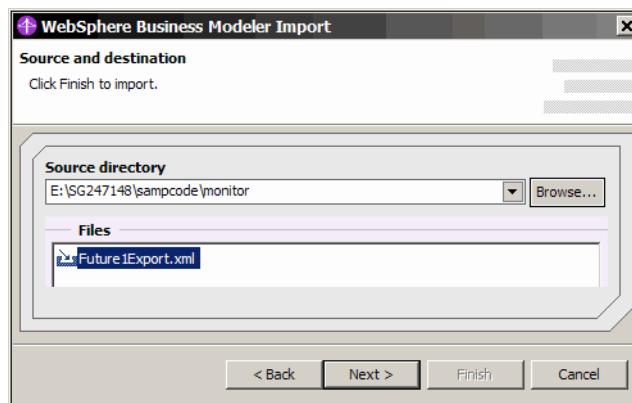
Figure 14-4   Modeler: Importing Monitor results (2)

► Click *display matching elements* and the list at the bottom is filled. Select *update processing durations* and *update probability on decisions*. Click *Finish* (Figure 14-5).
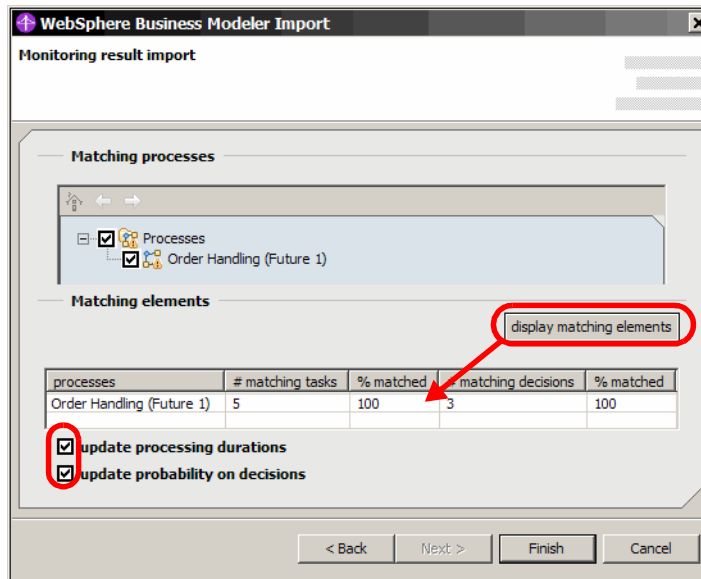
Figure 14-5   Modeler: Importing Monitor results (3)

► Processing starts and displays a dialog where you can see the details (Figure 14-6).
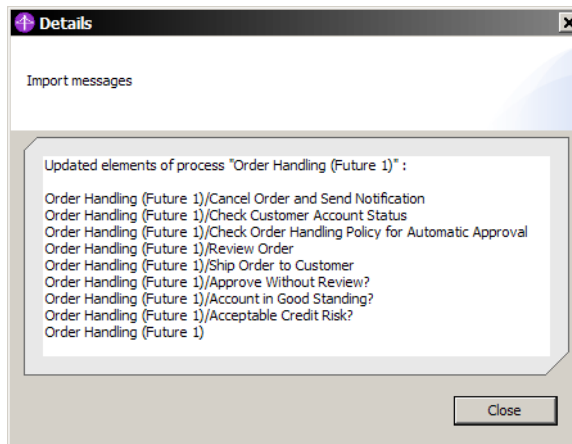
Figure 14-6   Modeler: Importing Monitor results (4)

Analyzing the Monitor measurements

To analyze the results we open the Order Handling (Future 1) process:

► We can see the new percentages for the decision blocks (Figure 14-7).
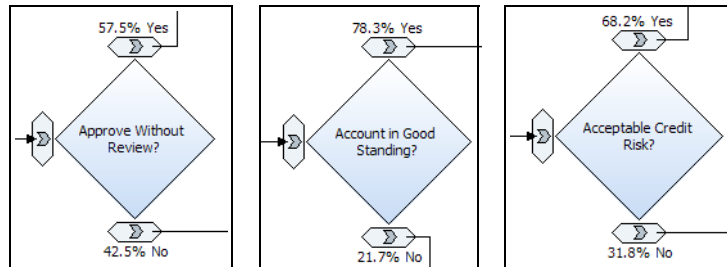
Figure 14-7   Decicion blocks with new percentage values

► We can see the task duration in the Attributes view (Figure 14-8) after selecting a task, for example, Check Order Handling Policy for Automatic Approval (1.131s) or Review Order (14m 12.123s).
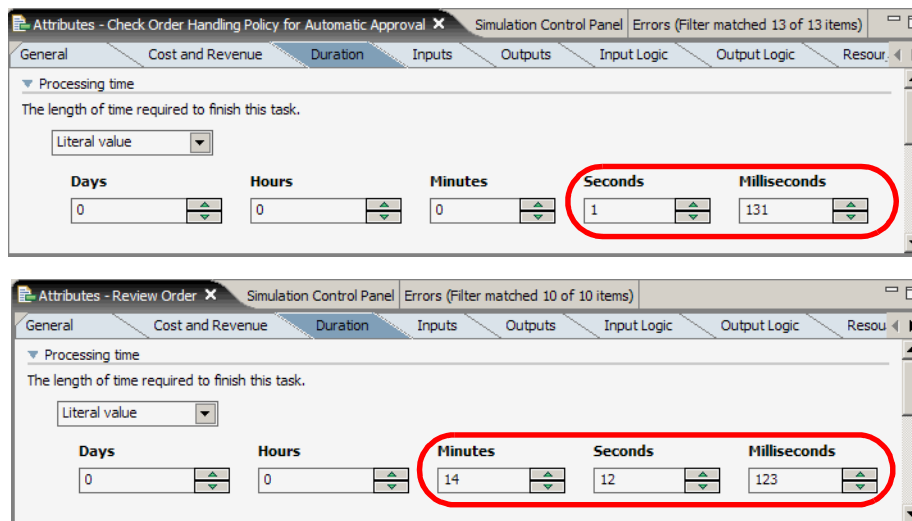
Figure 14-8   Activity duration measured by Monitor

► The duration of the process itself is also updated, using the value of the property `workingDuration="1194"` from the XML file.

Compare the measurements with the simulation

First we analyze the percentages:

► We did not reach the 65% automatic approval rate (57.5%).

► We missed the target of 85% account in good standing (78%).

► We missed the approval rate of 70% by the order manager (68%).

Next we analyze the durations:

► We got good results for the non-human tasks and reached about 1 second for each task.

► We came close to the 15 minutes for the order manager (14m 12s).

► We exceeded the shipping time of 16 minutes a little bit (17m).

► We exceeded the cancellation time of 2 minutes (2m 45s).

The measurements in the Monitor show that we still have too many orders reviewed by the order manager, which slows down the process and leads to a longer total time that an order spends in the system.

KPI analysis

From the KPI measurements in the Monitor we know:

► The percentage of shipped orders is only about 83% instead of 90%.

► The average duration exceeds the target of 3 days by a few hours.

Develop a strategy for improvements

Based on the Monitor measurements, the management of ClipsAndTacks decides to stream-line the business process in two ways.

Automatically approve more orders

To get more orders through the system without involvement of the order manager the limit for automatic approval must be raised. However, it could be dangerous to raise the limit for all customers, especially new ones.

The solution is to introduce a customer classification system:

► Regular customer
► Silver customers
► Gold customers

Gold and silver customers will have a higher limit for automatic approval of their orders. The business rule for automatic approval is changed as follows:

```
Order Handling Policy
============================
Default: Orders are reviewed by the system for automatic approval
============================
The order can be automatically approved without review if:
- the total order price is less than $750
- the total order price is less than $1250
  and the customer is classified as "Silver"
- the total order price is less than $1750
  and the customer is classified as "Gold"
============================
```

This new business rule does not change the flow of the order handling process. The new rule will be implemented in Integration Developer.

Automatically decline orders for customers with low credit rating

The checking of the customer account status sends orders back to the order manager for review. In most cases the order manager has declined the order based on the account status.

To get orders faster through the system management decides that orders that do not pass the customer account status check are declined immediately. This will further reduce the involvement of the order manager.

However, this is a change of the flow in the order handling process.

Creating the Future 2 process in the Modeler

The best way to change the model and keep the old model as well is to work with a new project:

► Export the project as a WebSphere Business Modeler project (.zip) file.

► Import the exported file into a new project name `Clips And Tacks Future 2`.

► Rename the process from `Order Handling (Future 1)` to `Order Handling (Future 2)`.

> **Note:** You can import the new model from:
>
> `SG247148\sampcode\model\Clips And Tacks Future 2.zip`
>
> See "Importing the current process model using the Modeler" on page 47 for instructions on how to import a model.

Change the business rule

The business rule is defined as an annotation (comment) in the Modeler. Therefore, change the annotation and you are done. The work will be performed in Integration Developer.

Change the percentage of automatically approved orders

Because of the customer classification (silver, gold) more orders will be automatically approved. For the simulation we have to change the percentages accordingly (Figure 14-9). We also change the percentages of the other decisions.
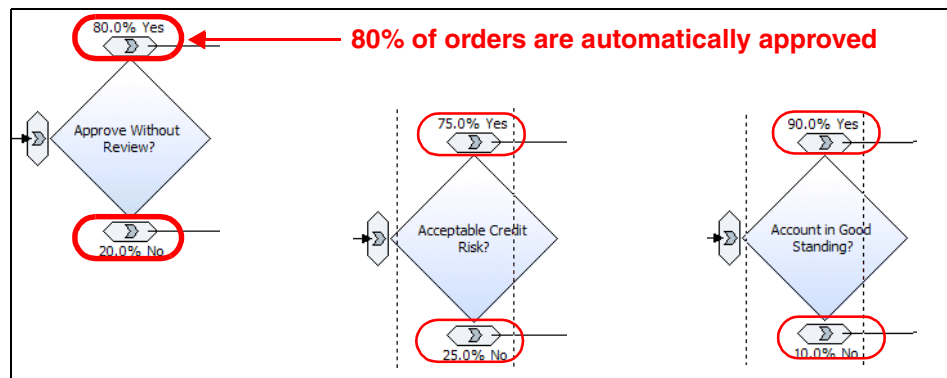
Figure 14-9   Automatic approval percentages

Change the process flow

The Check Customer Account Status activity verifies the credit rating of a customer. Based on that a decision is made to approve the order. ClipsAndTacks management has decided to immediately decline an order if the credit check is negative.

Therefore we have to change the flow and route an order from Account in Good Standing to Cancel Order and Send Notification, instead of routing the order to the order manager (Review Order).

The changes to the model are shown in Figure 14-10:

► Remove the merge in front of Review Order.

► Connect Approve Without Review directly to Review Order.

► Add a merge in front of Cancel Order and Send Notification.

► Connect Acceptable Credit Risk and Account in Good Standing to the new merge.
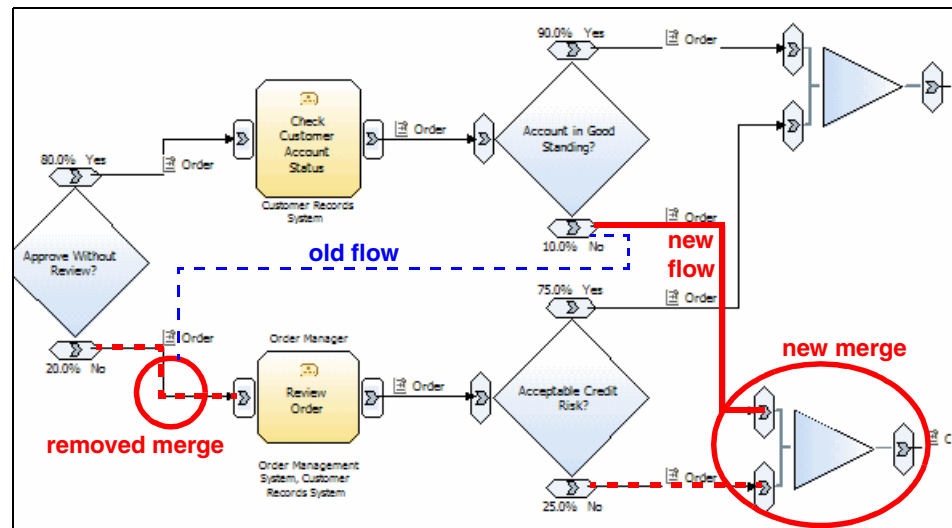
Figure 14-10   Changing the process flow

Changing a task to global

The Check Customer Account Status task is a local task. After discussions with the systems architect, who implemented the application in Integration Developer, the decision is made to change this task to a global task.

The reason for this change is to make the implementation more flexible, that is, easier to replace the implementation of this task when required.

To change the task to a global task follow these steps:

► Select the Check Customer Account Status task and *Convert to → Global task*.

► A dialog opens where the name could be changed. We leave the same name and click *Finish*.

The task is now displayed with double lines in the diagram and also appears as a separate entry in the Project Tree view (Figure 14-11).
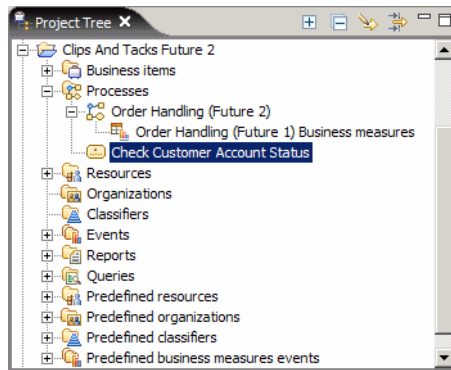
Figure 14-11   Project Tree view with global task

You can open the global task (Figure 14-12) and verify that all attributes are preserved (duration: 1 second, resources: bulk, technical specification: Java implementation).
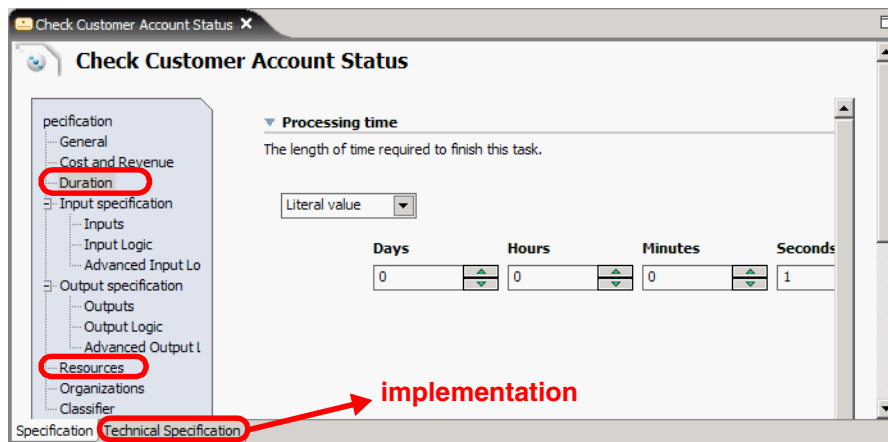
Figure 14-12   Attributes of a global task

Order Handling (Future 2)

The updated model is shown in Figure 14-13.

This diagram shows the future 2
ClipsAndTacks order handling process.

Order Handling Policy
=============================
Default: Orders are reviewed by the system
for automatic approval
=============================
The order can be automatically approved
without review if:
- the total order price is less than $750
- the total order price is less than $1250
and the customer is classified as "Silver"
- the total order price is less than $1750
and the customer is classified as "Gold"
=============================

Figure 14-13   Order Handling (Future 2) process diagram

Simulating the Future 2 process
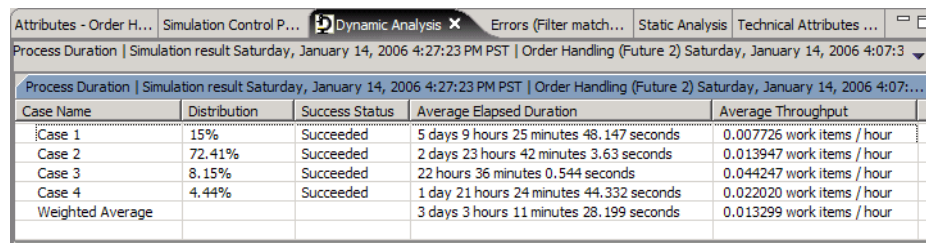
To simulate the process we have to verify all the information of duration, cost, and availability. We will use the same attributes as in "Populate the simulation environment" on page 128.

Then we create a simulation snapshot and populate its attributes with the same values as in "Simulating the Future 1 process" on page 131.

Finally we can run the simulation and create results.

Analyzing the Future 2 simulation results

The process duration analysis is shown in Figure 14-14.

Case Name	Distribution	Success Status	Average Elapsed Duration	Average Throughput
Case 1	15%	Succeeded	5 days 9 hours 25 minutes 48.147 seconds	0.007726 work items / hour
Case 2	72.41%	Succeeded	2 days 23 hours 42 minutes 3.63 seconds	0.013947 work items / hour
Case 3	8.15%	Succeeded	22 hours 36 minutes 0.544 seconds	0.044247 work items / hour
Case 4	4.44%	Succeeded	1 day 21 hours 24 minutes 44.332 seconds	0.022020 work items / hour
Weighted Average			3 days 3 hours 11 minutes 28.199 seconds	0.013299 work items / hour

Figure 14-14   Process duration results

► Case 1 are approved (order manager) and shipped orders.
► Case 2 are automatically approved and shipped orders.
► Case 3 are automatically approved but declined orders.
► Case 4 are non-approved and declined orders.

The total shipped orders is 87.41%, a little shy of 90%. The average time is a little over 3 days in a long simulation. These results show that it will be hard to reach the KPIs.

Update the Future 2 business measures

In general we leave the business measures the same as for Order Handling (Future 1).

Note: The name of the business measures does not change by renaming the process. It is still `Order Handling (Future 1) Business measures`.

The only change is the addition of the customer classification as a dimension so that dimensional analysis can be performed against the classification. For example, we will be able to track total order price against classification, or number of orders total/shipped/declined against classification.

Synchronize the business measures

Before making any changes to the business measures we have to synchronize the existing measures against the changed process:

► Make sure the business measures are not open for editing.

► Select the Order Handling (Future 2) process and *Synchronize* (context).

► A list of business measures to be synchronized is displayed. Click *OK*.

The the existing business measures (triggers, metrics, and so forth) are synchronized and show Order Handling (Future 2) as their process.

Open the *Order Handling (Future 2) → Order Handling (Future 1) Business measures*.

Setting the valid from date

The new model will only be valid once we complete the implementation and deploy the application to the server. Therefore, it is a good idea to change the valid from date:

► Select the *Diagram* tab.

► In the Attributes view, *General* tab, set the valid from date to a new date, for example:

```
Saturday, January 7, 2006 12:00:00 PM
```

Using customer classification for dimensional analysis

Follow the steps in "Adding a dimension" on page 171 to setup a dimension named *Classification Dimension* (see Figure 8-24 on page 172).

Next create a metric named *Classification*, of type `String`, with a description of `Customer classification` (see Figure 8-25 on page 172).

In the Value section click *Add* and create a trigger (Figure 14-15):

– Select *Inputs and outputs* as source category.
– Select the `Order Handling (Future 1)_Input Criterion` as source.

Use the expression builder to set the value to:

`Order Handling (Future 2).Input Criterion.Input.Customer.`**`Classification`**

In the Dimensional analysis and database schema settings section, select the `Classification` metric and select *Aggregation group in dimensional analysis*, set the maximum length to **8**, select *Set as part of the dimension key*, select the *Classification Dimension*, and set the aggregation group level to `0`.

> **Note:** Verify the `country` and `city` metrics. It is possible that the dimension key information is lost after synchronization.
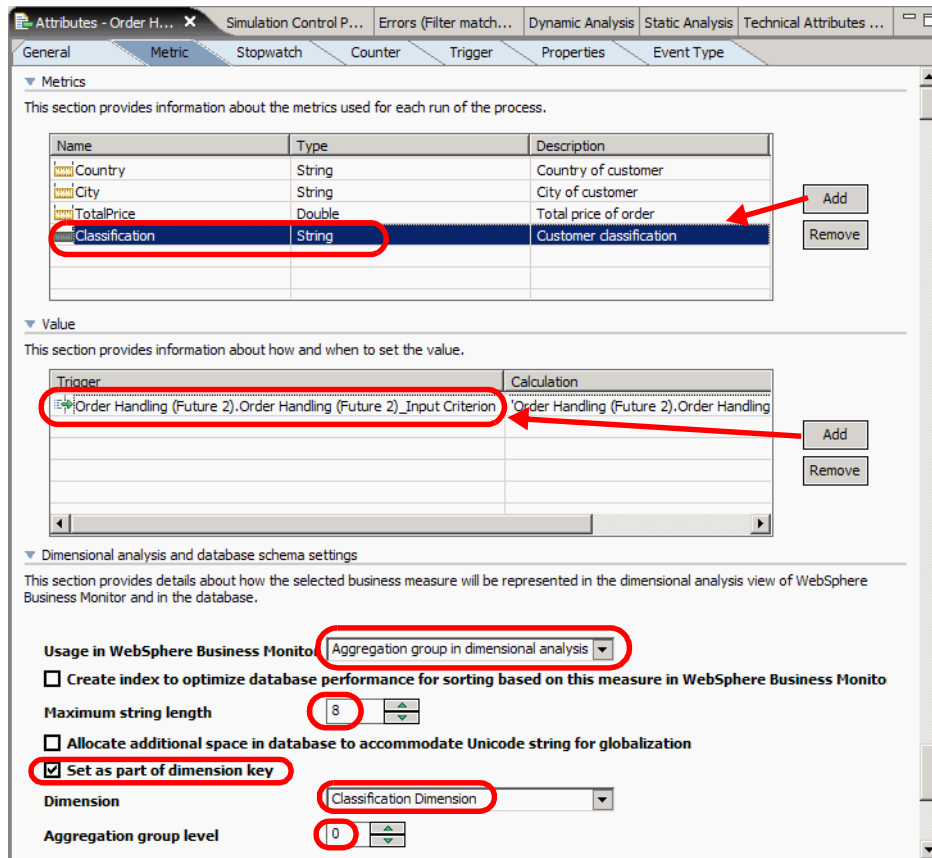
Figure 14-15   Classification as aggregation group for dimensional analysis

Save and close the business measures.

Export for Integration Developer and Monitor

Now we are ready to export the revised process to Integration Developer and Monitor. We follow the same process—with a small modification—as in "Export the process with the business measures model" on page 180.

Before exporting, verify that no errors are reported for the project when selecting *Modeling → Mode → WebSphere Process Server*.

To export the process for Integration Edition and Monitor, select the Clips And Tacks Future 2 project and *Export* (context):

► Select the type of export you want. Select *WebSphere Business Monitor and development tool* (see Figure 8-37 on page 181). Click *Next*.

► Define the destination and source information export (Figure 8-38 on page 182):

  – Target directory: `C:\SG247148\sampcode\model\export`
  – Monitor project version: `1`
  – Select *Export user defined event types to event catalog*
  – Click *Next*.

► Define the destination and source information (Figure 14-16):

  – Select *Module project name* and enter: `ClipsAndTacks2`
  – Select *Library project name* and enter: `ClipsAndTacks2Lib`
  – Select *Project Interchange name* and enter: `ClipsAndTacks2`
  – WID Workspace location: select the same target directory
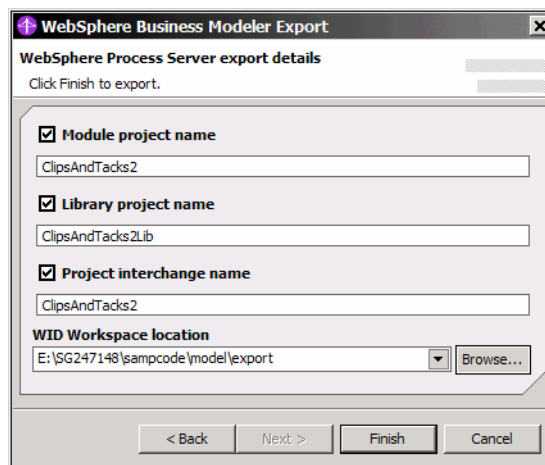
Figure 14-16   Export: WebSphere Process Server export details

► Click *Finish*.

We create both a module and a library. The library will contain the business items and the interfaces of the process and global task. We require the interface when implementing the global task in a separate module.

Exported files

The results of the export are the `ClipsAndTacks2.zip` file, the `Monitor.zip` file, and the `EventCatalog` directory.

Summary

In this chapter we started with the continuous improvement cycle by taking the Monitor measurements back into the Modeler to improve the business process. We implemented the model changes, simulated and analyzed the model, and exported the model for deployment to Integration Developer and Process Server and for new measurements by the Monitor.

15

Implementing the Future 2 process using WebSphere Integration Developer

In this chapter we describe how to implement the different tasks of the Future 2 order handling business process. We follow a similar approach as in Chapter 9, "Developing the application using WebSphere Integration Developer" on page 187, but with a few modifications:

► We use a separate module for the customer account status verification.

► We implement the business rule for automatic approval using templates.

► We only use the customized human task application instead of the BPC Explorer.

Import the Future 2 application

To import the revised application follow the steps described in "Import the model" on page 198. Select the `ClipsAndTacks2.zip` interchange file exported from the Modeler.

Wait until the workspace has been built and you should find these projects:

► `ClipsAndTacks2App`—Enterprise application

► `ClipsAndTacks2EJB`—EJB project with session beans

► `ClipsAndTacks2Web`—Web project (empty)

► `ClipsAndTacks2`—Module with BPEL, tasks, rules

► `ClipsAndTacks2EJBClient`—EJB client

► `ClipsAndTacks2Lib`—Library project with business items (`Businessitems.xsd`), and interfaces (`OrderHandlingFuture2Interface.wsdl` and `CheckCustomerAccountStatusInterface.wsdl`)

If you select the `ClipsAndTacks2` module and *Open Dependency Editor* you will seen that this project is dependent on the `ClipsAndTacks2Lib` project.

Note that an error is reported for the rule groups. We have to implement the business rule.

Assembly diagram

The assembly diagram is shown in Figure 15-1.
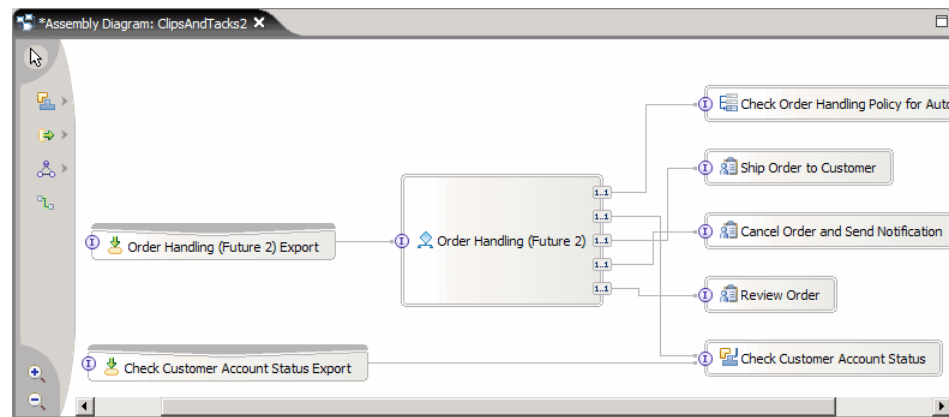
Figure 15-1   Order Handling (Future 2): Assembly diagram

Notice the two exports: one for the process and one for the global task (`Check Customer Account Status Export`).

Process diagram

The process diagram is shown in Figure 15-2.
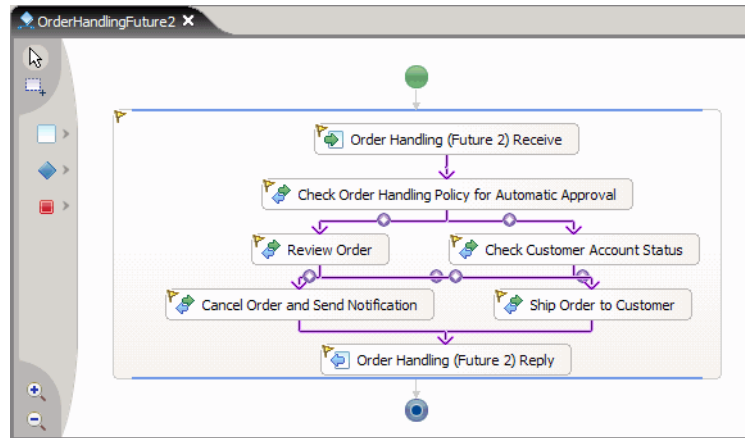
Figure 15-2   Order Handling (Future 2): Process diagram

In the properties view, leave *Automatically delete the process after completion* selected. We do not keep the finished processes.

Complete the application in Integration Developer

To complete the Future 2 application we follow steps similar to what we did in Chapter 9, "Developing the application using WebSphere Integration Developer" on page 187, with some changes to illustrate more functions of the process integration tools.

Use a separate module for the global task

We have one global task: Check Customer Account Status. To make this task independent of the business process, we can put it into a separate module. This enables us to change the implementation of the global task without affecting the business process application, or to change the business process in Modeler and bring it back into Integration Developer without affecting the implementation of the global task.

Create a module

To create a module for the global task, follow these steps:

▶ In the Business Integration view, select the `ClipsAndTacks2` module and *New → Module*.

▶ Enter `ClipsAndTacks2AccountStatus` as name and click *Finish*.

▶ Select the `ClipsAndTacks2AccountStatus` module and *Open Dependency Editor*. In the Libraries section click *Add* and select the `ClipsAndTacks2Lib` library. Save and close the editor.

The new module requires access to the business items and interfaces.

Create the assembly diagram

Open the assembly diagrams of both modules, `ClipsAndTacks2` and `ClipsAndTacks2AccountStatus`.

Perform these steps to update the two diagrams:

▶ In the **ClipsAndTacks2** diagram, delete the `Check Customer Account Status Export` and the `Check Customer Account Status` task.

▶ In the **ClipsAndTacks2AccountStatus** diagram drag the `CheckCustomerAccountStatus` interface from the `ClipsAndTacks2Lib` library into the diagram.

When prompted, select *Component with no Implementation Type* and click *OK*.

▶ A component named *Component1* appears in the diagram. Select the new component and in the Properties view change the name to `Check Customer Account Status`.

▶ Select the new component and *Export → SCA Binding*. This creates an export named `Check Customer Account StatusExport` (Figure 15-3).
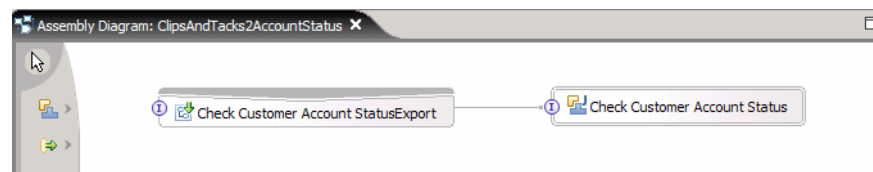
Figure 15-3   Global task assembly diagram

▶ Select the new component and *Generate Implementation → Java*. When prompted, select the default package and click *OK*.

The `CheckCustomerAccountStatusImpl.java` file opens. For now complete the code with this method:

```
        public DataObject InputCriterion(DataObject input) {
            return input;
        }
```

► Save and close the `ClipsAndTacks2AccountStatus` diagram.

► In the **ClipsAndTacks2** diagram, select the Order Handling (Future 2) process and *Wire References to New → Imports*. This creates an import named *CheckCustomerAccountStatusPartner* (Figure 15-4).
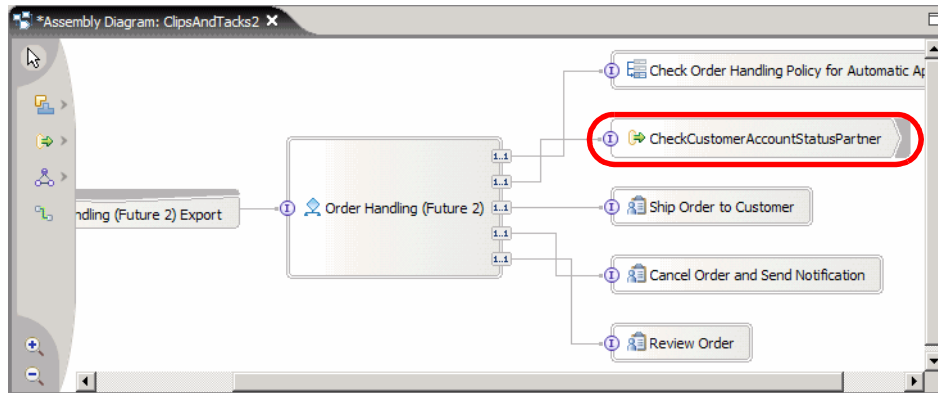
Figure 15-4   Wire references to imports

► Select the new import and *Generate Binding → SCA Binding*. The icon changes from 🡒 to 🡒 .

► Select the new import and in the Properties view, Binding tab, click *Browse*. In the selection dialog select the *Check Customer Account StatusExport* entry from the `ClipsAndTacks2AccountStatus` project (Figure 15-5).
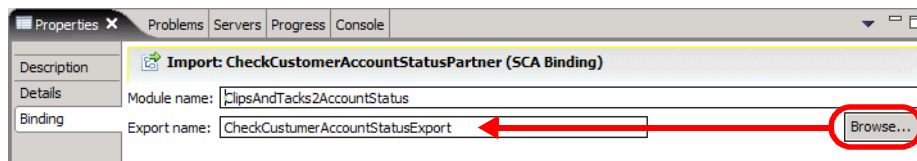
Figure 15-5   Selecting the SCA binding

► This completes the assembly diagram. Save and close the diagram.
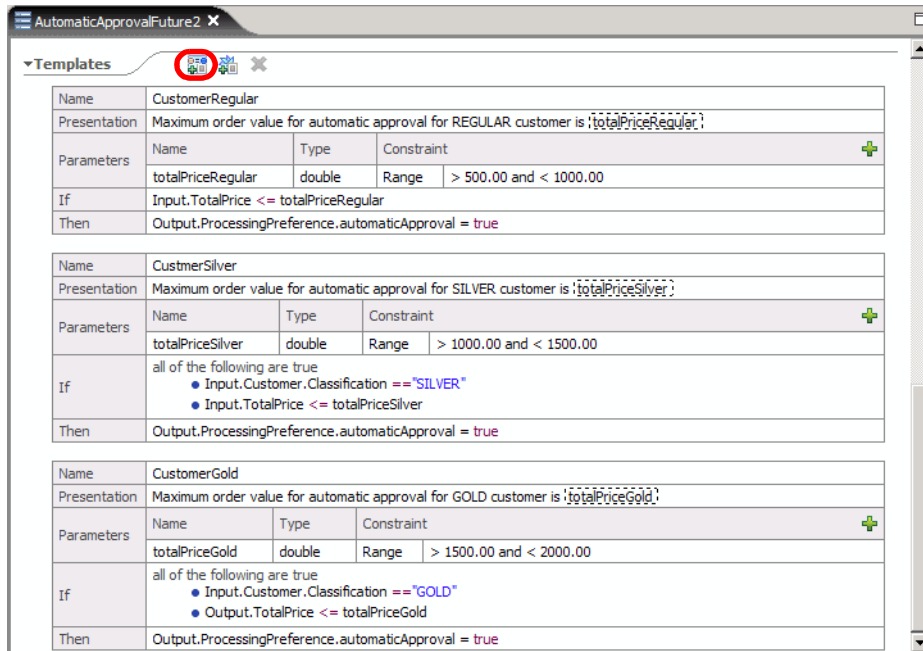
Implement the business rule

The business rule for automatic approval was formulated in the Modeler, taking into account the classification of the customer. Now we have to implement the rule.

We follow a similar approach as in "Implementing a business rule" on page 204, but we use templates as described in "Defining a business rule based on a template" on page 262. This enables us to use the Business Rules Manager to change the total price allowed for automatic approval.

To implement the business rule using templates follow these steps:

▶ Expand the *Business Logic* of `ClipsAndTacks2`. Open the rules group `CheckOrderHandligPolicyforAutomaticApproval`.

▶ Select the *InputCriterion* interface and click *Enter Destination*. Select *New RuleSet* and enter a name of `AutomaticApprovaleFuture2`. Click *Finish*.

▶ The `AutomaticApprovaleFuture2` rule opens.

▶ Add two action rules (same as in Future 1) and set the actions to:

    — `Output = Input`
    — `Output.processingPreference.automaticApproval = false`

▶ Add three *If Then* templates using the icon. Complete the templates as described in "Defining a business rule based on a template" on page 262 to get the result shown in Figure 15-6.

Figure 15-6 Business rule templates

    — Each template has a name, such as `CustomerRegular`.

- Each template has a variable for the total price.
- Each variable has a range limit.
- Each template has a presentation that uses the variable.
- The first template is for regular customers.
- The second template is for silver customers.
- The third template is for gold customers.
- Automatic approval is set if the order price is smaller than the variable.

► Add three *template* rules using the 🗋 icon. For each rule select one of the templates. Enter the initial value for the variable (total price). The final business rules are shown in Figure 15-7.
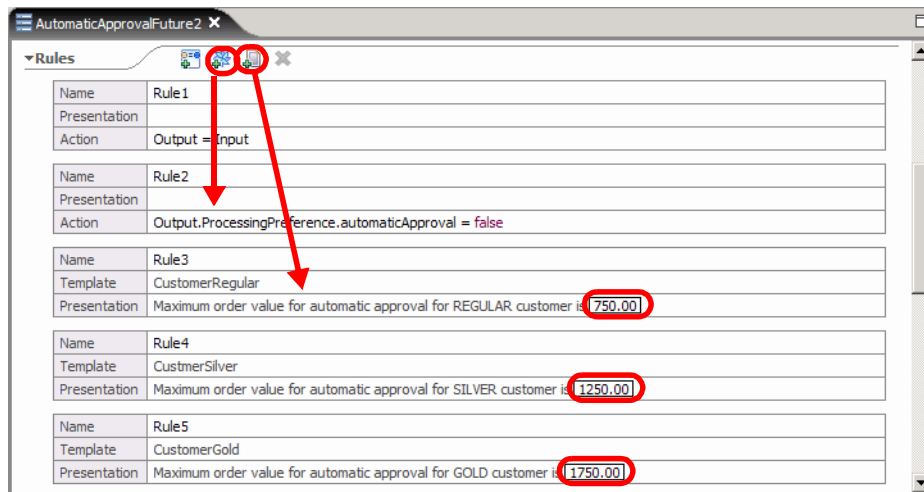
Figure 15-7   Business rule for automatic approval

► Save and close the business rule.

Implement the human tasks

In this application we only use our customized human tasks application described in "Implementing a human task application" on page 237. We will not activate the BPC Explorer.

To implement the human tasks follow these steps:

► Expand the *Business Logic* of `ClipsAndTacks2`. Open each human task, for example, `ReviewOrder`.

► Select *Potential Owner* (under Receiver settings) and in the Properties view set the verb to *Everybody*.

► Do not click the *BPC Explorer* icon 🔧 under Client settings.

Implement the global task using an external Web service

For the global task (Check Customer Account Status) we use the same approach as in "Implementing an external Web service" on page 234.

The Web service application, `ClipsAndTacksService`, has been imported and we can use the `CreditService.wsdl` file for the `ClipsAndTacks2AccountStatus` module.

To implement the global task follow these steps:

► Drag the `CreditService.wsdl` file from `ClipsAndTacksService` (*WebContent → WEB-INF → wsdl*) into the `ClipsAndTacks2AccountStatus` project.

► In the Business Integration view, expand the `ClipsAndTacks2AccountStatus` project and the `CreditService` is visible under Web Service Ports.

► Open the `ClipsAndTacks2AccountStatus` assembly diagram. See "Adding the Web service to the assembly diagram" on page 235 for detailed instructions.

► Select the `CreditService` and drag it into the diagram, next to the Check Customer Account Status activity. When prompted, select *Import with Web Service Binding*.

► Select the new component (`Import1`) and in the Properties view change the name to `Check Credit Import`.

► Select the wire icon and draw a connection from Check Customer Account Status to Check Credit Import. When prompted, Click *OK* to create a matching reference in the source node, and click *No* to use the WSDL interface (instead of converting to a Java interface).

The complete diagram is shown in Figure 15-8.
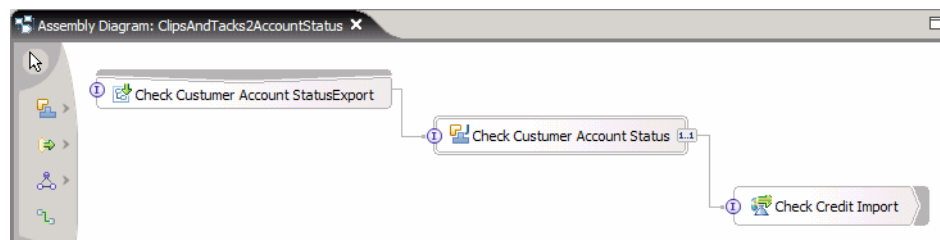
Figure 15-8   Check Customer Account Status assembly diagram

► Create a package with the name `com.clipstacks.credit` in the `ClipsAndTacks2AccountStatus` project.

► Import the `CreditRating.java` file into the new package from:

     `SG247148\sampcode\wid\creditRating`

▶ Open the `CreditRating.java` file and change the code to use the Web service:

```
static boolean useWebservice = true;
```

▶ Complete the Java code of `CheckCustomerAccountStatusImpl` with this logic:

```
public DataObject InputCriterion(DataObject input) {
    System.out.println("Check Customer Account Status Invoked");
    // create CreditRating bean
    com.clipstacks.credit.CreditRating creditRating =
                    new com.clipstacks.credit.CreditRating();
    // call CreditRating bean to update the BO
    DataObject orderOut = creditRating.calculateCreditRating(input);
    return orderOut;
}
```

Add a stand-alone reference to invoke the process

To be able to invoke the process from the Web front-end we define a stand-alone reference in the assembly diagram (see "Add a stand-alone reference to invoke the process" on page 216):

▶ Open the assembly diagram of `ClipsAndTacks2`.

▶ Click the arrow next to the *Import* icon ![icon], then select the *Stand-Alone References* icon ![icon], then click into the diagram.

▶ Link the stand-alone reference to the Order Handling (Future 2) process. Select the wire icon ![icon], then click on the reference and on the process.

When prompted, click *OK* to create a matching reference, and click *No* to create a Java interface instead of the WSDL interface (Figure 15-9).
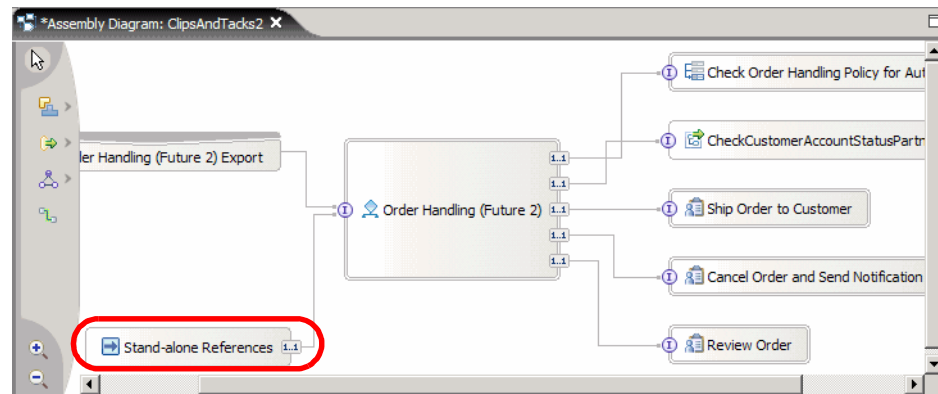
Figure 15-9   Order Handling (Future 2): Assembly diagram with stand-alone reference

The name of the Stand-alone reference is `OrderHandlingFuture2Partner`, visible in the Properties view.

Import the Web front-end application

To invoke the business process using a customer order we use the same Web front-end, `ClipsAndTacksFrontWeb`.

However, because the Future 1 application is still running in the real server in parallel to the new Future 2 application, we have to use a different context root. We also know that the Web application must be part of the enterprise application of the business process.

To implement the Web front-end follow these steps:

► In the Web perspective, select *Dynamic Web Projects* and *Import → WAR file* (context menu).

► Click *Browse* to locate the WAR file:

`SG247148\sampcode\wid\webfront\ClipsAndTacksFrontWeb.war`

► Change the Web project name to **ClipsAndTacks2FrontWeb**.

► Deselect *Add module to an EAR project*. Click *Finish*.

Attaching the Web front-end to the process application

The Web application invokes the process through the stand-alone reference. This is only possible if the Web application is part of the same enterprise application as the process.

To attach the Web application to a business process application follow these steps:

► In the Business Integration perspective select the `ClipsAndTacks` project and *Open Dependency Editor* (context menu).

► In the Dependencies editor, expand the J2EE section and click *Add*.

► Select the `ClipsAndTacks2FrontWeb` project and click *OK*.

► Select the `ClipsAndTacks2FrontWeb` project in the list and deselect *On Build Path*, but leave *Deploy with Module* selected.

► Save and close the Dependency editor.

► Open the `InvokeOrderHandling` Java code (in `com.clipstacks.sca`) and change the stand-alone reference name to:

`String standAloneReferenceName = "OrderHandlingFuture2Partner";`

Deploy and test the Future 2 application

The implementation is now complete and we can deploy and test the business process in the built-in Process Server:

► Start the Process Server and wait until it is ready.

► Select the server in the Servers view and *Add and remove projects*:

– Select these projects:

```
ClipsAndTacks2
ClipsAndTacks2AccountStatus
ClipsAndTacksHumanCustomEAR
ClipsAndTacksServiceEAR
```

– Click *Finish*.

Wait until all the applications are started in the server.

Test the application

See "Using the Web front-end" on page 229 for instructions on how to submit an order. Then see "Using the human task application" on page 242 for instructions how to handle the human tasks.

Note that you have to use the changed context root for the Web application:

```
http://localhost:9080/ClipsAndTacks2FrontWeb/index.jsp
```

Using the Business Rules Manager

You can dynamically update the business rule for automatic approval using the Business Rules Manager (see "Using the Business Rules Manager" on page 265):

► Select the server and *Launch → Business Rules Manager*.

You can also opne a browser and enter the URL:

```
http://localhost:9080/br/webclient
```

► Expand the sections and select *AutomaticApprovalFuture2* (Figure 15-10).

► Click *Edit*, then change some of the values and click *Save*.

► Select *Publish and Revert* and click *Publish* to make the new rule active.
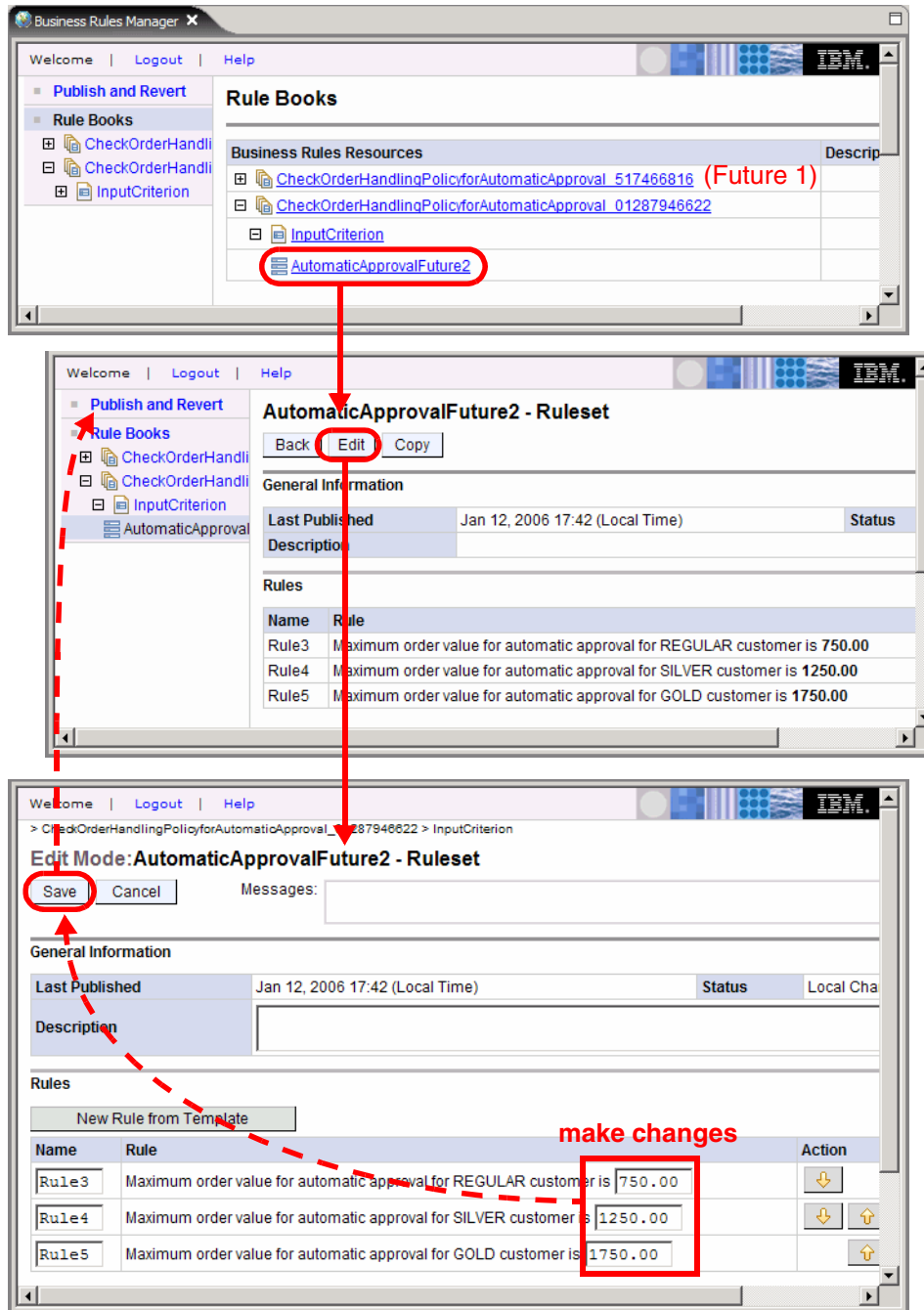
Figure 15-10 Using the Business Rules Manager

Install and run the application in Process Server

To deploy the application to a real process server follow the instructions in "Export application from Integration Developer" on page 252.

You have to export all the enterprise applications involved:

- ► `ClipsAndTacks2`
- ► `ClipsAndTacks2AccountStatus`
- ► `ClipsAndTacksHumanCustomEAR`
- ► `ClipsAndTacksServiceEAR`

The last two applications listed, `ClipsAndTacksHumanCustomEAR` and `ClipsAndTacksServiceEAR`, are unchanged and do not have to be exported and installed if they are already deployed to the Process Server.

Installing the application

Install the enterprise applications as described in "Install application in Process Server" on page 255.

Running the application

Run the application using these URLs:

```
http://localhost:9080/ClipsAndTacks2FrontWeb/
```

```
http://localhost:9080/ClipsAndTacksHumanCustomWeb/
```

Testing the business rule

To test the business rule using the customer classification, remember which customers have which classification:

```
Customer    Classification
12345       GOLD
11111       SILVER
22222       REGULAR
33333       GOLD
44444       SILVER
```

- ► Automatically approved orders:
  - – For a SILVER customer up to $1250
  - – For a GOLD customer up to $1750
- ► Such orders may still be declined by the customer credit check.

Changing the business rules

To start the Business Rules Manager on the Process Server open a browser and enter this URL:

```
http://localhost:9080/br/webclient
```

See Figure 15-10 on page 386 for instructions on how to change the rules.

Summary

In this chapter we described how to implement the Order Handling (Future 2) process in Integration Developer. We used different techniques as compared to the Future 1 process to illustrate more functions of Integration Developer and Process Server.

The bsuiness process is now ready for measurements using the Monitor.

16

Measuring the Future 2 process

In this chapter we implement the WebSphere Business Monitor for the improved Order Handling (Future 2) process.

Then we measure the business process application and verify that we can meet the key performance indicators.

To measure the Order Handling (Future 2) process we used the same technique as described in Chapter 13, "Measuring the Future 1 process" on page 329.

Setting up the Monitor

We setup the Monitor in the same way as described in "Business measures model deployment to Monitor" on page 300:

► Prepare for schema generation (using the Monitor.zip file produced by the Modeler for the Clips And Tacks Future 2 project).

► Generate the schema.

► Generate the tables in the State, Runtime, and History databases.

► Start the DB2 replicators.

► Deploy the business measures.

► Import the DB2 Cube Views.

► Configure the Dashboard Client with appropriate views for measuring the process. See "Configuration of the Monitor Dashboard" on page 313.

Measuring the Future 2 application

We used a script similar to the script described in "Measurement script" on page 330 to submit a number of orders to the Future 2 application.

We also tracked the process instances in the Monitor databases. Notice that new tables are generated for the Order Handling (Future 2) process.

Finally we used the Monitor Dashboard to view the KPIs. metrics, and we performed dimensional analysis of the process data.

Monitor Dashboard for the Future 2 application

In tis section we look at the different view we configured for our revised application.

Gauges view

Let's start with the Gauges view for the key performance indicators (Figure 16-1):

A first glance shows us that we reached our targets:

► Over 90% of orders have been shipped

► The average order fulfillment time is about 2.5 days
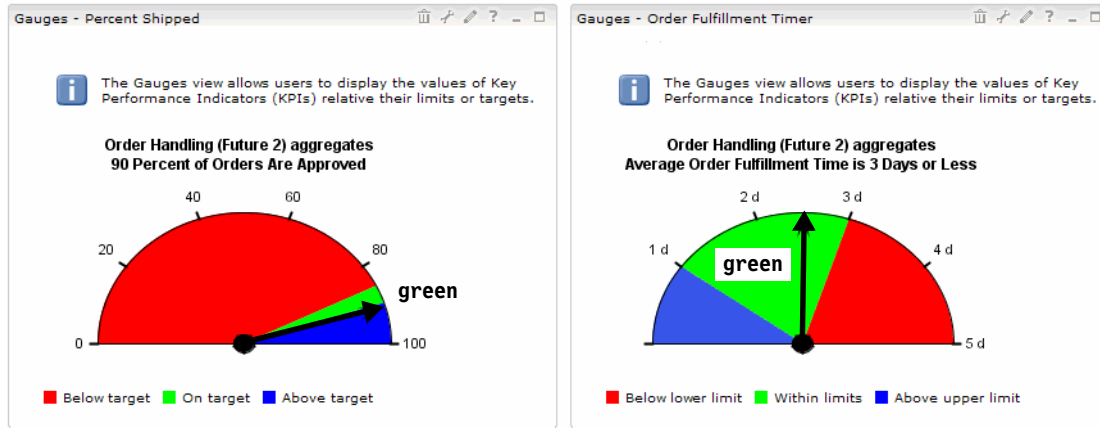
Figure 16-1   KPI gauges of revised process

KPIs and Scorecard views

The KPIs and Scorecards views shows us the exact figures (Figure 16-2). We shipped **91.2%** of the orders in an average time of **2 days 12 hours**.
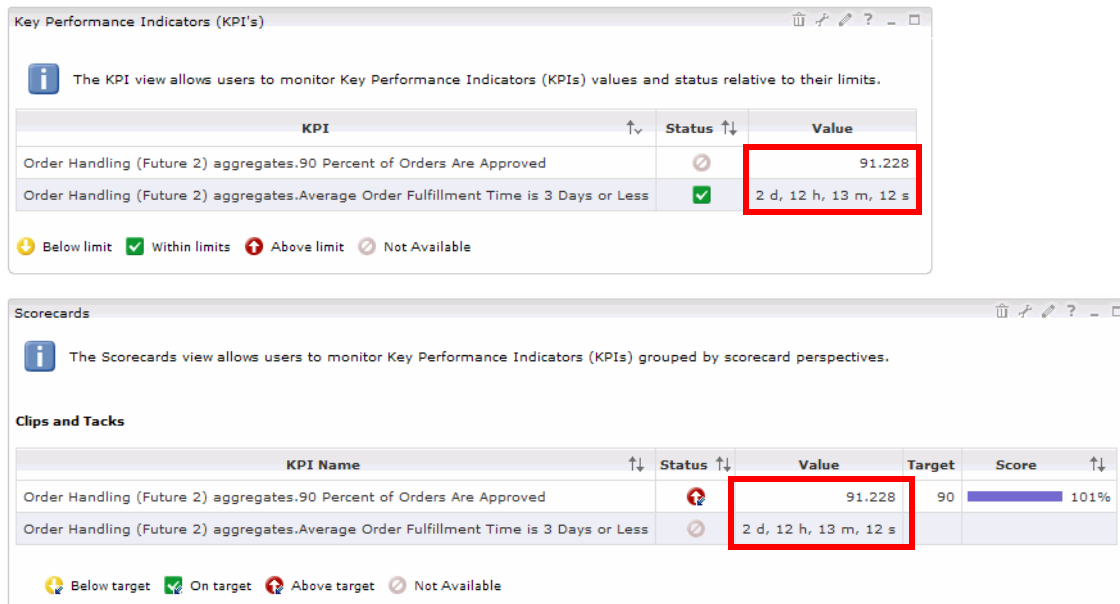
Figure 16-2   KPIs and Scorecards views

Dimensions view

In the revised application we added the customer classification (`REGULAR`, `SILVER`, `GOLD`) to the process model and defined this classification as a dimension. Therefore, we can now display our business measures as a function of the classification.

The Monitor Dashboards also enables us to configure views that shows multiple dimensions, such as location and classification.

Figure 16-3 shows us orders by customer classification and location. All the `GOLD` orders come from the USA and the others from Canada,
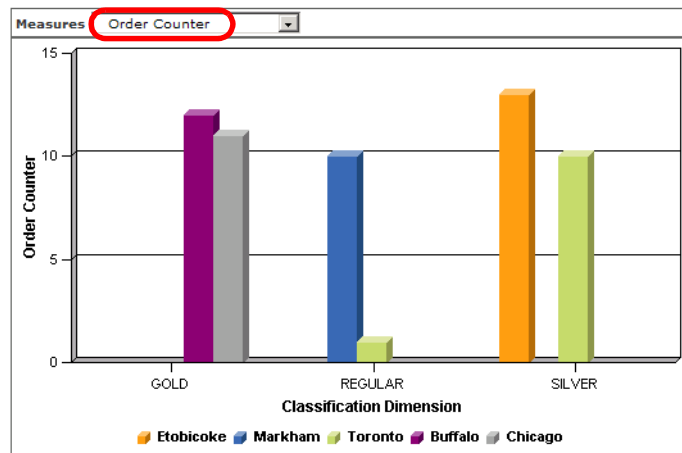
Location Drill-Down

Location Dimension ↑↓	Classification Dimension ↑↓	GOLD ↑↓	REGULAR ↑↓	SILVER ↑↓
Location Dimension ↑↓	57	23	11	23
Canada ↑↓	34		11	23
Etobicoke ↑↓	13			13
Markham ↑↓	10		10	
Toronto ↑↓	11		1	10
USA ↑↓	23	23		
Buffalo ↑↓	12	12		
Chicago ↑↓	11	11		

Figure 16-3   Orders by customer classification and location

Figure 16-4 shows us the same data but rearranged. We also tailored the colors according to the customer classification. To change a color, select a vertical bar and *Format Data Series* (context menu).
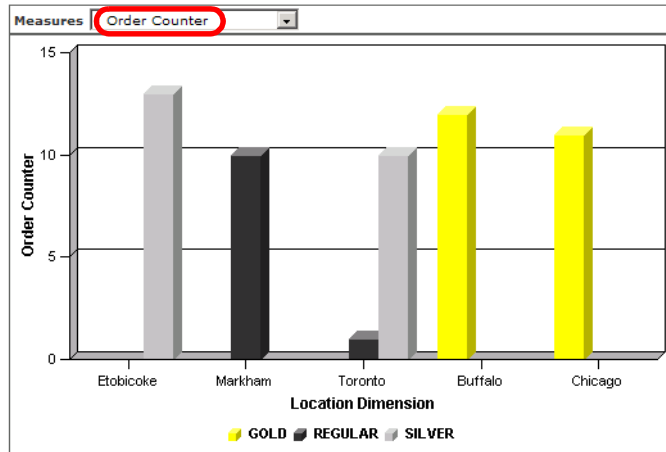
Figure 16-4   Orders by location and classification

Figure 16-5 shows the sum of the total price by location and classification.
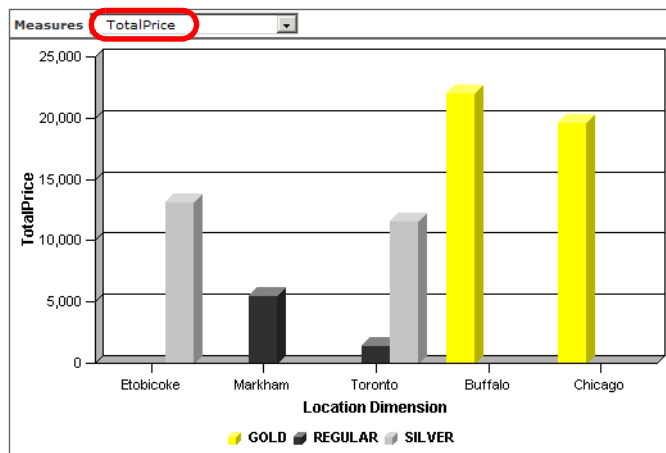
Figure 16-5   Sum of total price by location and classification

The Dimensions view can be configured with different graphics and also a number of different styles for each graphic.

Figure 16-6 shows the three order counters by classification. In the pull-down at the top we can further drill-down to the locations (country and city). In the table we can also drill-down by classification and location.

Location Dimension [Location Dimension ▾]

Chart Types: 3D Bar

Measures — Declined Order Counter, Order Counter, Shipped Order Counter

🟧 GOLD 🟦 REGULAR 🟩 SILVER

USA Drill-Down
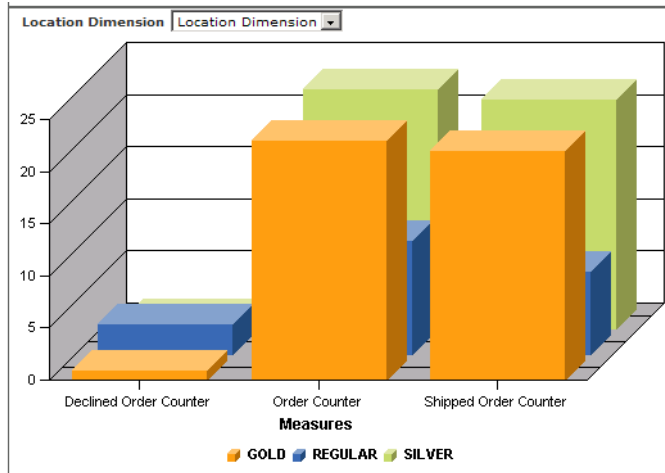
Location Dimension ↑↓	Classification Dimension ↑↓	Measures ↑↓	Declined Order Counter ↑↓	Order Counter ↑↓	Shipped Order Counter ↑↓
Location Dimension ↑↓	Classification Dimension ↑↓	57	5	57	52
	GOLD ↑↓	23	1	23	22
	REGULAR ↑↓	11	3	11	8
	SILVER ↑↓	23	1	23	22
Canada ↑↓	Classification Dimension ↑↓	34	4	34	30
	GOLD ↑↓				
	REGULAR ↑↓	11	3	11	8
	SILVER ↑↓	23	1	23	22
USA ↑↓	Classification Dimension ↑↓	23	1	23	22
	GOLD ↑↓	23	1	23	22
	REGULAR ↑↓				
	SILVER ↑↓				
Buffalo ↑↓	Classification Dimension ↑↓	12	0	12	12
	GOLD ↑↓	12	0	12	12
	REGULAR ↑↓				
	SILVER ↑↓				
Chicago ↑↓	Classification Dimension ↑↓	11	1	11	10
	GOLD ↑↓	11	1	11	10
	REGULAR ↑↓				
	SILVER ↑↓				

Figure 16-6   Orders total, shipped, and declined by classification

Figure 16-7 shows a pie chart of shipped orders by customer classification. We can select the measure in the top pull-down and we can drill-down by location.
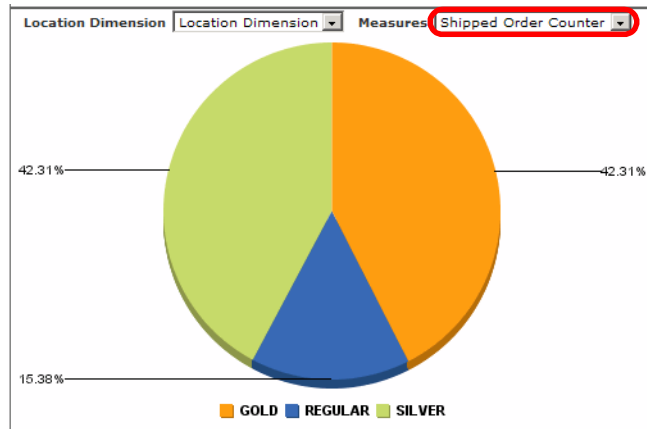
Chart Types: Pie

Figure 16-7   Shipped orders by classification

Figure 16-8 shows us the sum of the total cost as a pie chart by classification. Gold customers account for close to 60% of all orders.
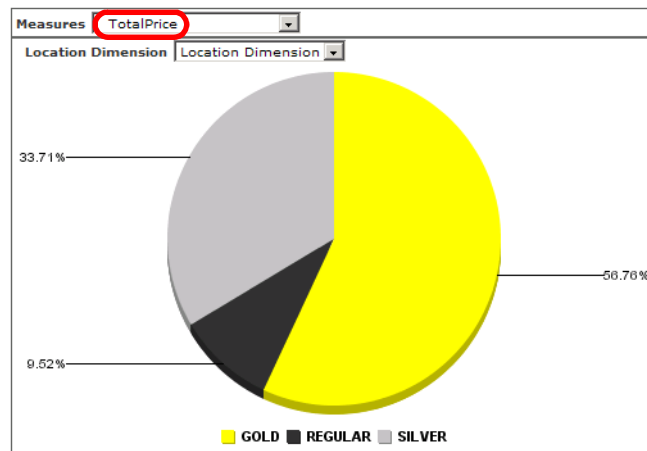
Figure 16-8   Total cost by classification

Figure 16-8 shows on what dates in January and February orders were created and how many of the orders were shipped.
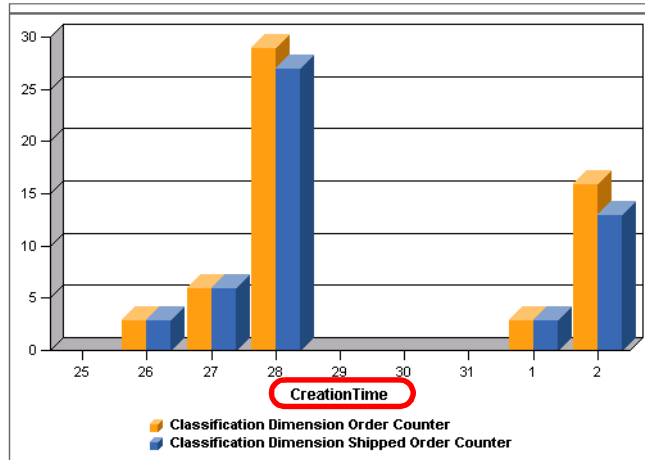
Figure 16-9   Orders created and shipped by date

As you can see there are many ways to display the accumulated data. The important part is to configure appropriate metrics and dimensions in the Modeler using the business measures editor.

Active Instances view

The Active Instances view is the same as discussed in "Active Instances view" on page 334. We do not show any instances here.

Summary

In this chapter we measured the revised application and used the Monitor Dashboard to verify that the application meets the expectations by exceeding both key performance indicators.

Using the customer classification as a dimension we also analyzed the order data to get a better understanding where the majority of orders are originating.

ClipsAndTacks management is very happy about their investment into an improved application that meets customer requirements and streamlines their operation.

Part 6

Appendixes

A
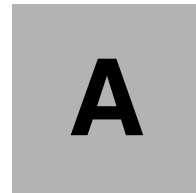
Installation of Modeler, Integration Developer, and Process Server

In this appendix we describe how to install the base products of the WebSphere Business Integration suite.

Installation of WebSphere Business Modeler V6

To install the WebSphere Business Modeler V6.0 follow these steps:

► Start the `setup.exe` from the installation CD or folder.

► In the Launchpad select *Install IBM WebSphere Business Modeler Advanced Version 6.0* (Figure A-1).
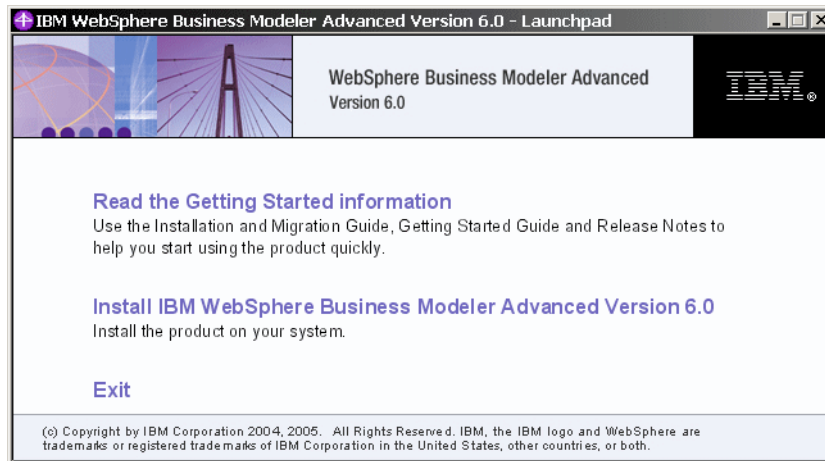
Figure A-1   WebSphere Business Modeler Launchpad

► In the Welcome page click *Next*.

► Accept the license agreement and click *Next*.

► Enter the name of the installation directory. The default is:

```
C:\Program Files\IBM\Modeler6
C:\Modeler60                                        <==== our choice
```

► Click *Next* in the summary panel.

► Wait for the installation to complete.

► Optionally select *Run WebSphere Business Modeler* and click *Next*.

► Click *Finish* to exit the installation wizard.

► Investigate if interim fixed or Fixpacks are available and install them.

► Start the Modeler using *Start → Programs → WebSphere Business Modeler → WebSphere Business Modeler*.

Select a location for the workspace, for example:

```
c:\Workspaces\Modeler60ClipsTacks
```

Installation of WebSphere Integration Developer V6

To install WebSphere Integration Developer V6.0.1 follow these steps:

► Start the `launchpad.exe` from the installation CD or folder.

► In the Launchpad select *View the Installation Guide* if you require help for installing the product.

► In the Launchpad select *Install IBM WebSphere Integration Developer V6.0.1* (Figure A-2).
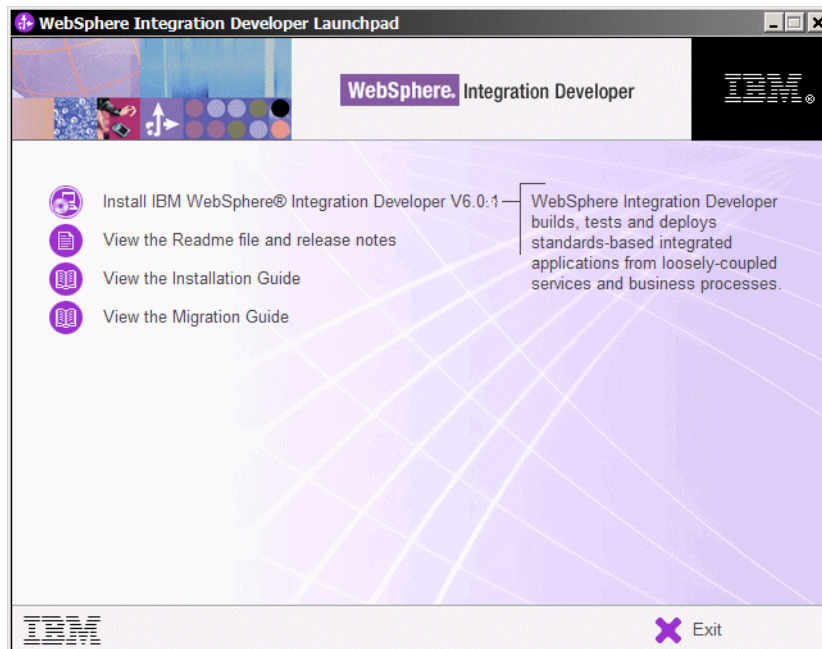
Figure A-2   WebSphere Integration Developer Launchpad

► In the Welcome panel click *Next*.

► Accept the license agreement and click *Next*.

► Enter the name of the installation directory. The default is:

```
C:\Program Files\IBM\WebSphere\ID\6.0
C:\WID601                                    <==== our choice
```

► In the features panel select *Integrated Test Environment* (Figure A-3).
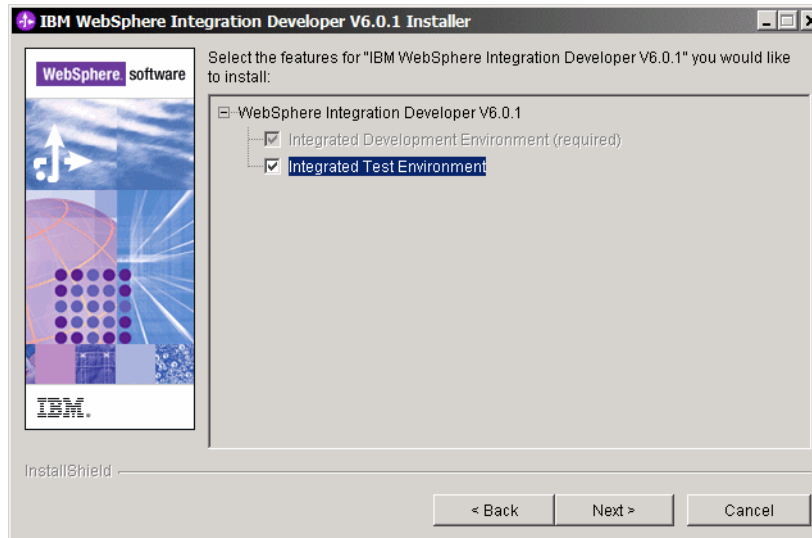
Figure A-3   Installing the integrated test environment

► Select the *WebSphere Process Server* profile (Figure A-4). For our scenario we do not require the WebSphere Enterprise Service Bus server.
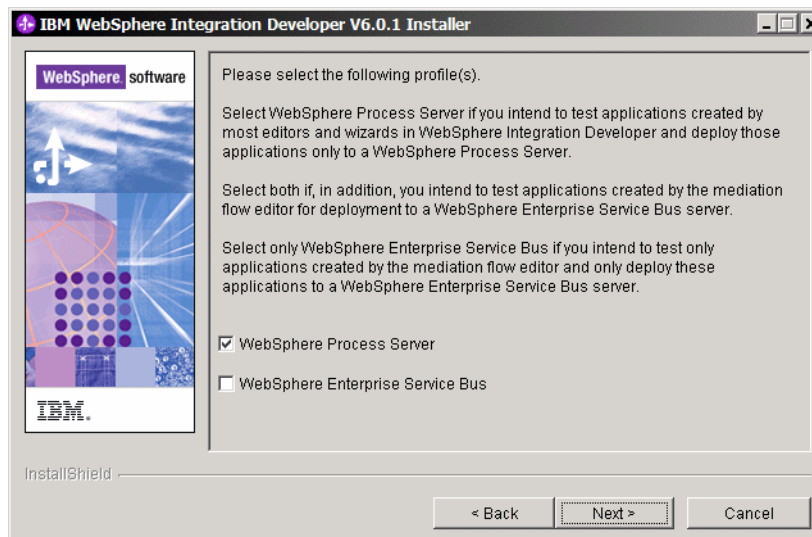
Figure A-4   Creating a profile for WebSphere Process Server

► Click *Next* in the summary panel.

► Be patient, the installation takes between 1.5 and 2 hours.

► When the installation is finished you receive a confirmation (Figure A-5). Click *Next*.
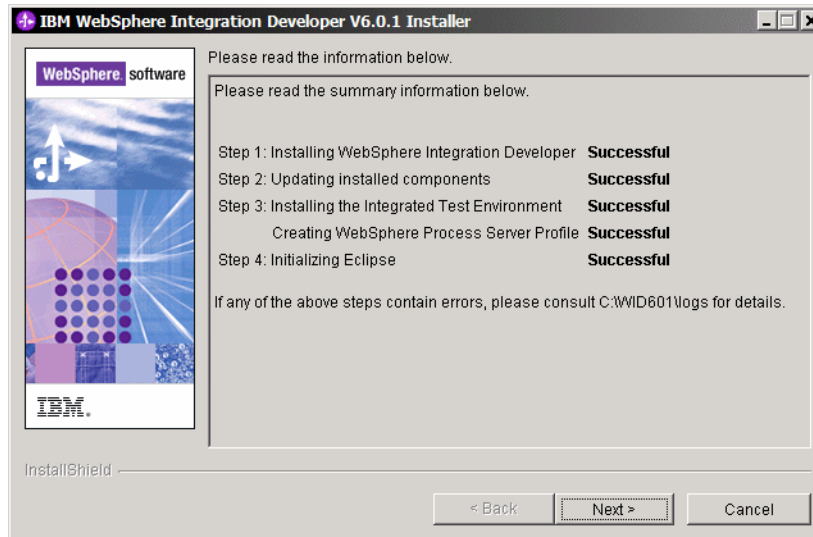
Figure A-5   Installation successful confirmation

► Select *View readme* and click *Next*. A browser is launched and the `readme.html` file is displayed. You can read the information.

► Read the summary panel (successful installation) and click *Next*.

► Select *Launch Rational Product Updater* to verify if product updates are available (Figure A-6). Click *Finish*.
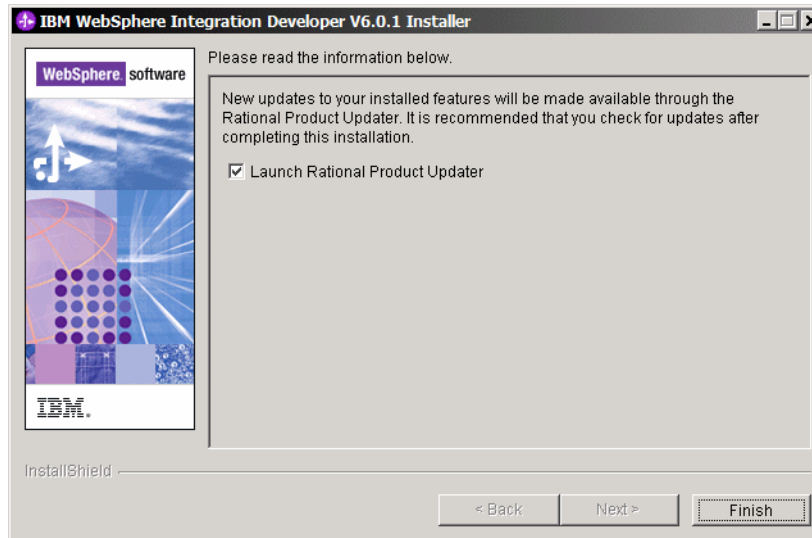
Figure A-6   Launch Rational Product Updater

► The Rational Product Updater opens. Select *IBM WebSphere Integration Developer* and click *Find Updates* (Figure A-7).
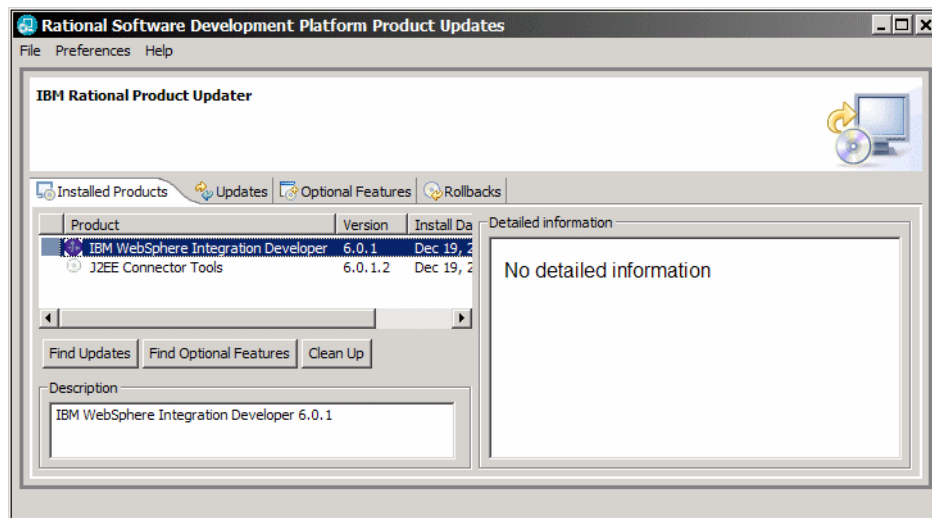
Figure A-7   Rational Product Updater

You have to be connected to the Internet to check for product updates.

> ► If there are product updates available, install them. Then close the Rational
> Product Updater.

> ► Click *Exit* in the Launchpad.

Interim fixes and fixpacks
Investigate if interim fixed or Fixpacks are available and install them.

Start WebSphere Integration Developer
You are now ready to start WebSphere Integration Developer. See "Start
Integration Developer" on page 191 for further information.

Installation of WebSphere Process Server V6

To install the WebSphere Process Server V6.0.1 follow these steps:

► Start the `launchpad.exe` from the installation CD or folder.

► In the Launchpad select *WebSphere Process Server installation* (Figure A-8).

Figure A-8   WebSphere Process Server: Launchpad

► Select *Launch the Installation Wizard for WebSphere Process Server for Multiplatforms* (Figure A-9).
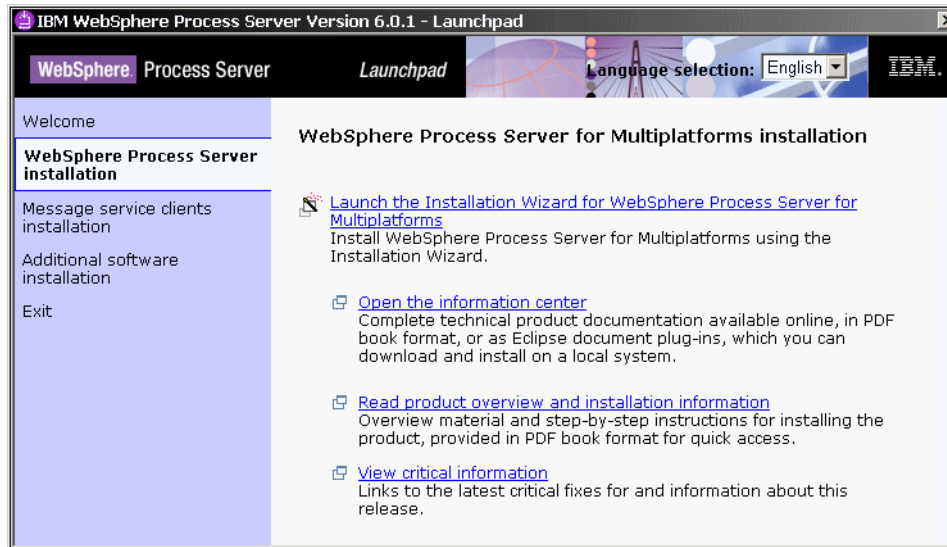
Figure A-9   WebSphere Process Server: Launch installation

► In the Welcome page click *Next*.

► Accept the license agreement and click *Next*.

► In the System prerequisite check panel click *Next*.

► If you install the server on the same machine as Integration Developer you get a warning about the existing test environment server. Click *Next*.

► Select *Install a new copy* (Figure A-10).
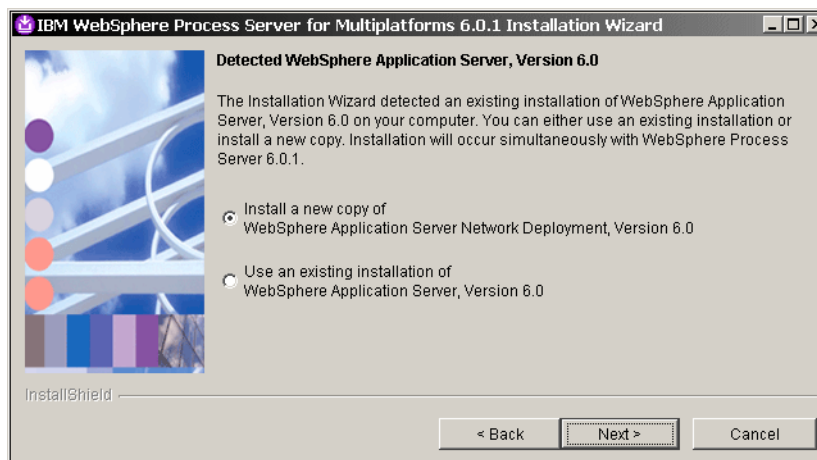
Figure A-10   WebSphere Process Server: Install new copy

▶ The warning about the existing server is shown again. Click *Next*.

▶ Enter the name of the installation directory. The default is:

```
C:\Program Files\IBM\WebSphere\ProcServer
C:\WPS601                                          <==== our choice
```

▶ Select *Custom installation* (Figure A-11).
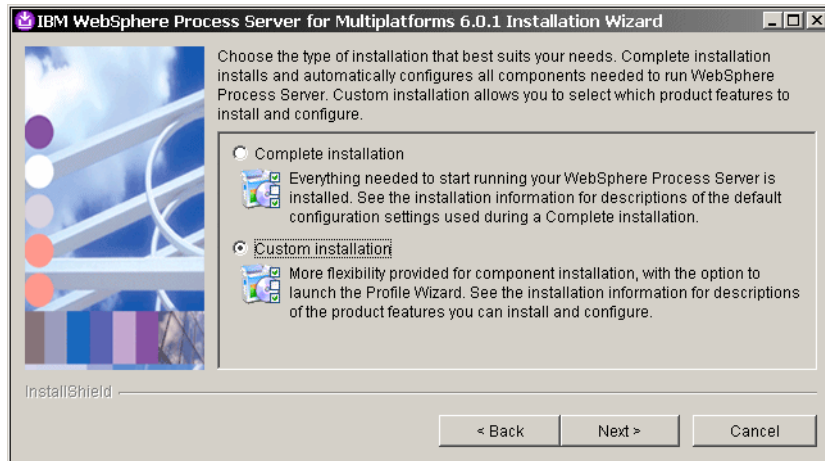
Figure A-11   WebSphere Process Server: Custom installation

▶ Select installation options (Figure A-12).
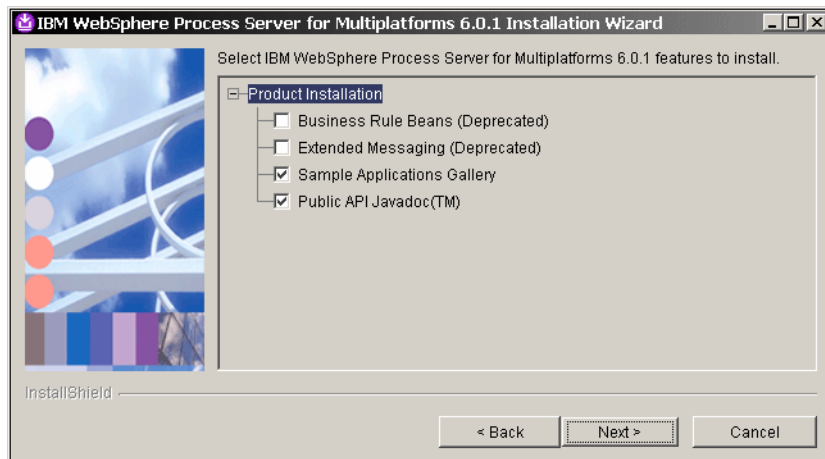
Figure A-12   WebSphere Process Server: Installation options

▶ Click *Next* in the summary panel.

► Wait for the installation to complete:

– Installation of WebSphere Application Server ND
– Installation of WebSphere Process Server
– Execution of several configuration commands

► Optionally select *Launch the Profile Wizard* and click *Next* (Figure A-13).
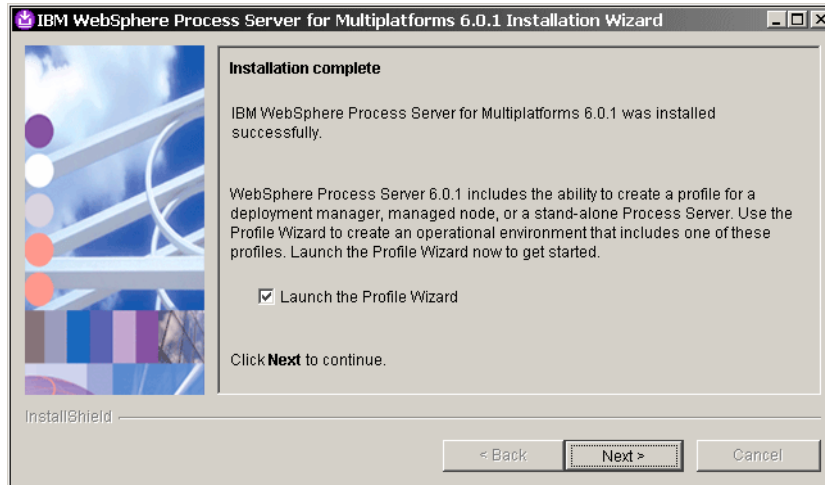
Figure A-13   WebSphere Process Server: Launch Profile Wizard

► The Profile Wizard starts (Figure A-14).
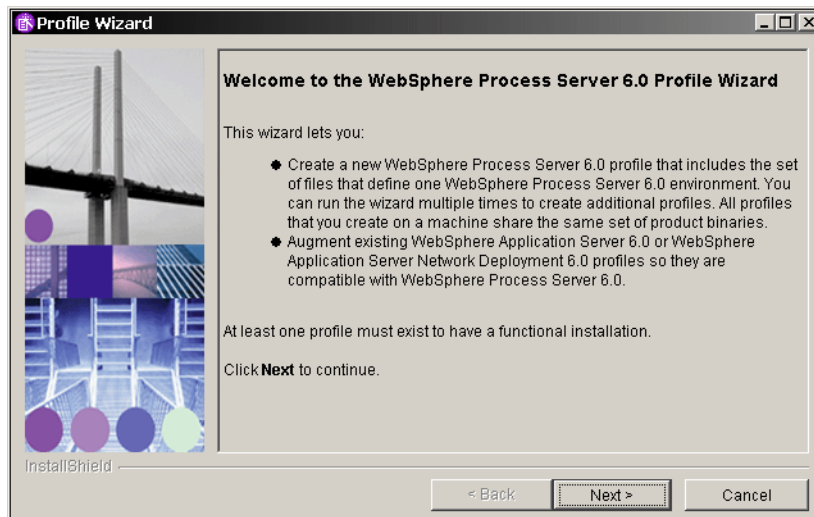
Figure A-14   WebSphere Process Server: Profile Wizard

► Select *STand-alone profile* and click *Next* (Figure A-15).
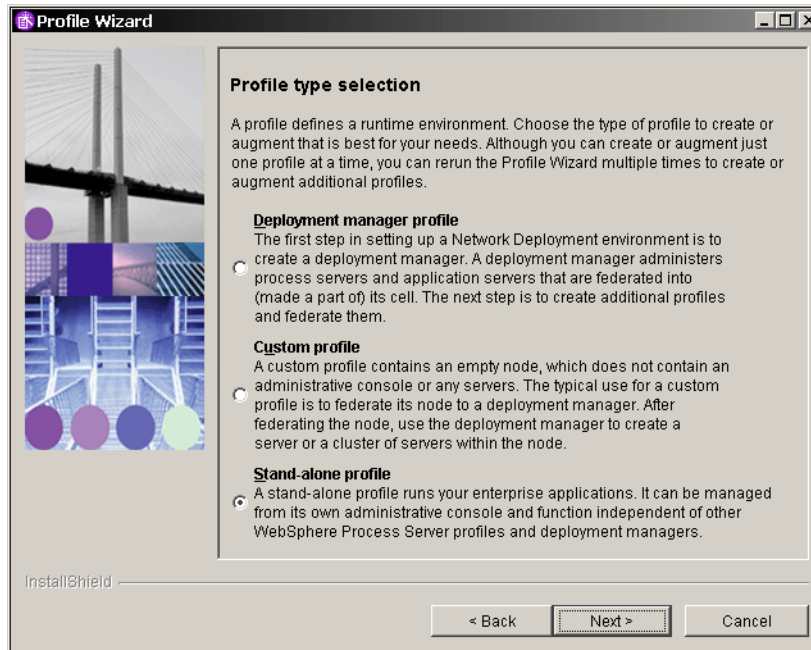
Figure A-15   WebSphere Process Server: Profile selection

► Enter the name of the profile, the default is `ProcSrv01`. Click *Next*.

► Accept the location of the profile directory, for example:

  `D:\WPS601\profiles\ProcServ01`

► Enter a node name, for example, WPSNode01, and the host name where the server runs. Click *Next*.

► Enter the port numbers. If you have other WebSphere servers on the machine the numbers are increased automatically. If you plan to only run one server at a time, then you can reset all the port numbers to their defaults (Figure A-16).
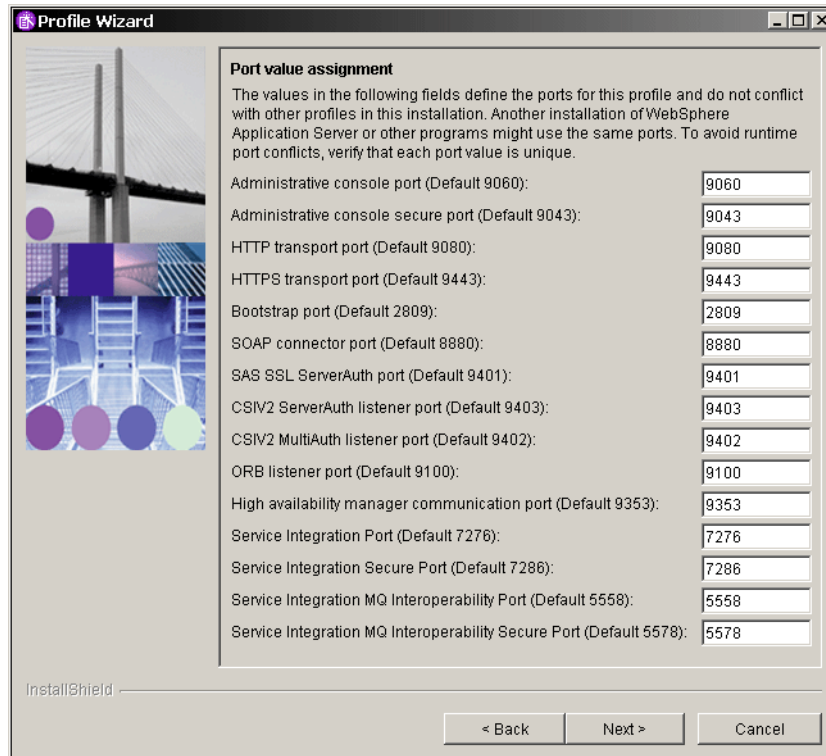
Figure A-16   WebSphere Process Server: Ports

► Select how to run the server, either as a service or only manual.

► Optionally enter user ID and password for the Service Component Architecture.

► Enter a user ID, password, and database product for the Comment Event Infrastructure Configuration.

► Optionally enter user ID and password for the Business Process Choreographer Configuration.

► Optionally create an Application Scheduler Configuration.

► Select *Create new (local) database* for the Process Server common database `WPRCSDB` and select the database product.

► Click *Next* in the summary panel.

► Optionally select *Launch the First Steps console*. From the First Steps console you can administer, start, and stop the server.

► Click *Finish* to exit the profile wizard.

In the Launchpad select *Additional software installation*. You have the options of installing (Figure A-17):

► IBM HTTP Server
► Web Server Plug-Ins
► Application Clients
► Application Server Toolkit
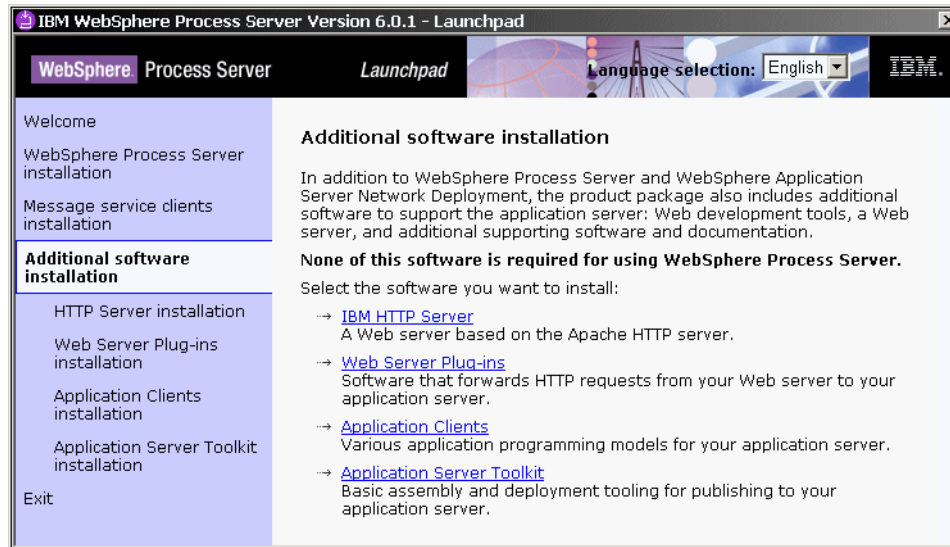
Figure A-17   WebSphere Process Server: Additional software

For the purpose of the ClipsAndTacks scenario we do not need any of these products.

Click *Exit* to terminate the Launchpad.

Interim fixes and fixpacks

Investigate if interim fixed or Fixpacks are available and install them.

Controlling the process server

You can start the process server by using *Start → Programs → IBM WebSphere → Process Server 6.0 → Profiles → ProcSrv01 → Start the server*.

From the same menu you can also *Stop the server*, start the *Administrative console*, or start the *First steps* menu (Figure A-18).
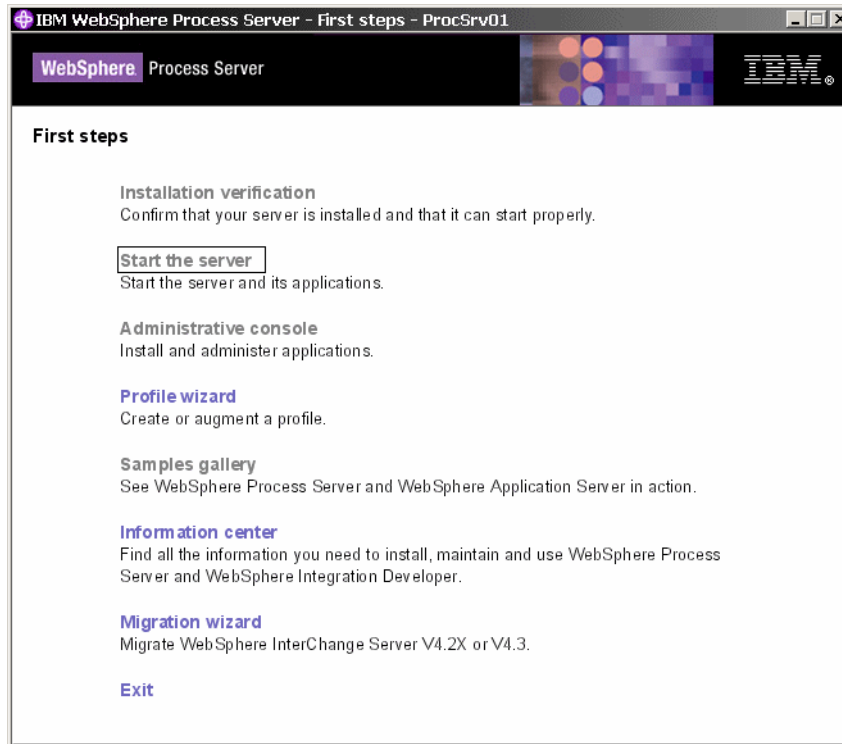
Figure A-18   WebSphere Process Server: First Steps

The First Steps menu can be used to verify the installation (which starts the server), start and stop the server, and start the administrative console.

Configuring the business process container

For business process execution the business process container must be installed using the administrative console or a JACL script.

Start the administrative console and perform these steps:

▶ Expand *Servers* → *Application Servers*.

▶ Click *server1*, then expand *Business process container setting settings* and click *Business process container*.

▶ You should see messages that the business process container is not installed.

▶ Under Additional Properties, click *Business process container installation wizard*.

► For JDBC Provider select *Cloudscape 5.1 (Cloudscape JDBC provider (XA))*. This action fills all the other fields. Click *Next*.

► For JMS select *Default messaging provider*.

  For JMS user ID and JMS API user ID enter the user ID and password of the user that starts the process server.

  For both security roles enter `Administrators` (for Windows systems).

  Click *Next*.

► Select *Create new JMS resources using default values*.

  Select *Select this check box to install the Business Process Choreographer Explorer*.

  If you want to monitor the server using the WebSphere Business Monitor, select *Enable Common Event Infrastructure logging for all processes running in this container*.

  Click *Next*.

► Read the summary. A reminder states: You must create the database and the tables yourself..... using the files in `WPS01\ProcessChoreographer`.

  Click *Finish*.

► The applications are installed. Wait for the messages:

```
Application BPEContainer_node_server1 installed successfully.
Application BPEExplorer_node_server1 installed successfully.
```

► Notice the remark:

  In order to use human task capabilities, the Human Task Manager must also be configured. The Human Task Manager configuration is here.

  Click *here*.

Configuring the human task container

Continue the process and install the human task container:

► Under Additional Properties, click *Human task container installation wizard*.

► For JMS select *Default messaging provider*.

  For JMS user ID and Escalation user ID enter the user ID and password of the user that starts the process server.

  For both security roles enter `Administrators` (for Windows systems).

  Click *Next*.

► Select *Create new JMS resources using default values*.

Optionally select *Mail session for Human Task Manager* (used for escalation).

If you want to monitor the server using the WebSphere Business Monitor, select *Enable Common Event Infrastructure logging*.

Click *Next*.

► Read the summary and click *Finish.*

► The application is installed. Wait for the message:

```
Application TaskContainer_node_server1 installed successfully.
```

Save the configuration and restart the server

Click *Save to Master Configuration*, then click *Save*.

Stop and restart the server, for example using the First Steps menu.

Creating the database for the process choreographer

For our purpose we use a Cloudscape database. Note that this is not supported in a real production environment.

The default database location is `<WPS-HOME>\profiles\ProcSrv01\databases`.

To create the `BPEDB` database follow these steps:

► Open a command window in the directory:

```
<WPS-HOME>\profiles\ProcSrv01\databases
```

► Copy the DDL file `createDatabaseCloudscape.ddl` from the `ProcessChoreographer` directory to the `databases` directory.

► Open the DDL file in WordPad and read the instructions, which tell you to run the command:

```
java  -Djava.ext.dirs=<WAS-HOME>\cloudscape\lib
      -Dij.protocol=jdbc:db2j: com.ibm.db2j.tools.ij
      createDatabaseCloudscape.ddl
```

► This creates the `BPEDB` database and tables.

B

Installation of WebSphere Business Monitor

This appendix provides the installation procedures for WebSphere Business Monitor V6.0.

We describe the system and software requirements necessary for installation and the steps involved for a successful installation of WebSphere Business Monitor V6.0.

WebSphere Business Monitor pre-requisites

This section summarizes the requirements for installation and related components required for WebSphere Business Monitor V6.

System requirements

System prerequisites for installing WebSphere Business Monitor are one of the following for both the Monitor Server and Monitor Dashboard:

► Windows 2000 Server, service pack 4
► Windows 2000 Advanced Server, service pack 4
► Windows Server 2003 Enterprise Edition service pack 1
► Windows Server 2003 Standard Edition service pack 1
► AIX® 5.2, maintenance package 5200-05
► AIX 5.3, with APAR IY58143

Note: The setup and installation for this redbook are based on Windows 2000 Server.

Software requirements

Figure B-1 shows the software pre-requisites for WebSphere Business Monitor.

Prerequisite Software	Monitor Databases	Dashboard Client[1]	Monitor Server [2]
IBM® DB2® UDB Database Server Version 8.2.1[6]	✓	✓	✓
DB2 Cube Views™ Version 8.2.1 [3]	✓ [3]	✓	
WebSphere Process Server Version 6.0 [4]			✓
WebSphere Application Server ND (WAS) Version 6.0.2.3		✓	
WebSphere Portal Version 5.1.0.2		✓	
IBM DB2 Alphablox Version 8.3 [5]		✓	

Figure B-1   Matrix of pre-requisite software requirements

Notes:

1. The **Dashboard Client** component can only be installed onto a machine that does not contain the pre-requisites. The Dashboard Client component and its prerequisites must be installed using only the Launchpad onto a "clean" machine that has no previous installation of the pre-requisites. Do not configure any the pre-requisites before the Dashboard Client component is installed. **Note**: Only Internet Explorer is supported; Firefox and Mozilla browsers are not supported.

2. The **Monitor Server** component can only be installed onto a machine that does not contain the pre-requisites. The Monitor Server component and its pre-requisites must be installed using only the Launchpad onto a clean machine that has no previous installation of the pre-requisites.

3. **DB2 Cube Views** must be installed on the machine where the historical database resides.

4. Although WebSphere Business Monitor V6.0 runs on WebSphere Process Server 6.0.0, it only supports applications running on WebSphere Process Server 6.0.1

5. If you are installing IBM **DB2 Alphablox** on a Windows 2003 system, prior to uninstallation, you must set the compatibility level of the file *Uninstall IBM DB2 Alphablox8.3.exe* to the value `Windows XP`.

6. On Windows platform, after installing DB2 using the WebSphere Business Monitor Launchpad, you must close the launchpad and all command windows or Explorer windows. Then you can restart the Launchpad and proceed with the remainder of the installation.

> **Important:** Before installing on Windows 2000, you should make the hostname of your machine 8 characters or less, if possible. This is important when you remote catalog the Monitor Server DB2 database of this machine to your Monitor Dashboard machine. Systems with more than 8 characters on Windows systems may cause conflicts in the remote catalog of the database system. Also, the WebSphere Application Server cell name can be too long which may cause problems on Windows systems.

Installation directories

Once you have all of the software pre-requisites and Monitor files on your system, you extract the WebSphere Business Monitor Launchpad to the parent directory of the pre-requisites. Assuming you extracted the Launchpad to a folder named `CDImage`, you should have this directory structure:

```
CDImage\Alphablox        DB2 Alphablox
CDImage\CUBE             DB2 Cube Views
CDImage\ESE              DB2 Universal Database
CDImage\Portal           WebSphere Portal
CDImage\Portal5102PTF    WebSphere Portal PTF
CDImage\ProcessServer    WebSphere Process Server
CDImage\WAS              WebSphere Application Server
```

This directory structure is suggested and optimal for a successful run of the Launchpad to install the pre-requisites correctly.

If your directory structure containing the installation files for the pre-requisites does not match this structure, the launchpad will prompt you to locate the folder where the pre-requisite files exist.

Refer to "Software requirements" on page 418 to understand which of these pre-requisites are required for the Monitor Server and which are required for the Monitor Dashboard.

Launchpad

The Launchpad is used to install the pre-requisites and WebSphere Business Monitor for both the Monitor Server and Monitor Dashboard.

> **Important:** The installation procedure in this section only discusses the use of the WebSphere Business Monitor Launchpad to install the pre-requisites.
>
> Installation of any of the pre-requisites without the use of the Launchpad is not supported.

Run the Launchpad by starting the batch file `launchpad.bat`. This starts a command window and the Launchpad opens (Figure B-2).

Figure B-2   WebSphere Business Monitor Launchpad

Note: Do not close the command window initially displayed during the installation procedure, as this will abort the WebSphere Business Monitor Launchpad.

Installing Monitor Server

First we install the Monitor Server on one of the Windows 2000 Server machines. The first step is to create the databases.

Create monitor databases

► Click *Create Databases* in the Launchpad, then select *Databases* and all four databases are selected (Figure B-3).
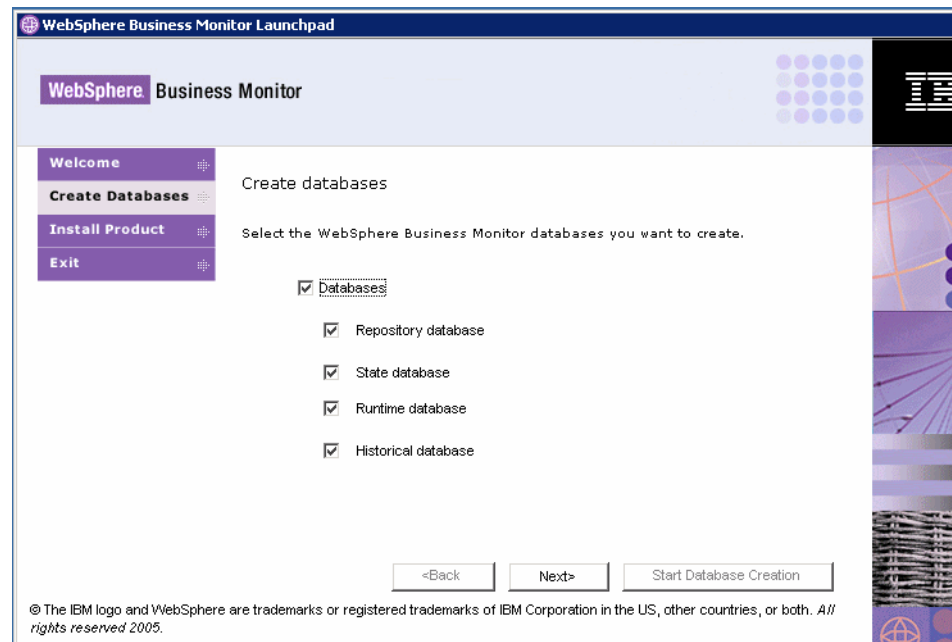
Figure B-3   Monitor Server installation: Create databases

> **Note:** This is the simplified topology for our ClipsAndTacks project scenario, where all of the run-time databases reside on the Monitor Server machine. As discussed in "WebSphere Business Monitor architecture overview" on page 291, this may not be the optimal topology designed for high-performance.

► Click *Next*.

► The launchpad verifies the existence of the pre-requisites of DB2 and DB2 Cubes. If the pre-requisites are not installed you are notified (Figure B-4).
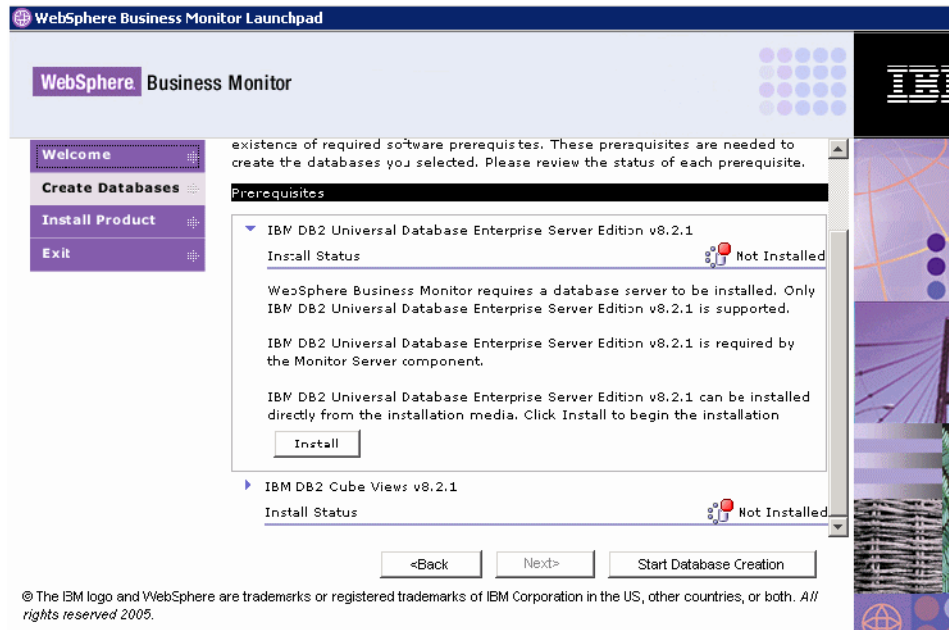
Figure B-4   Monitor Server installation: Pre-requisite verification

Install DB2 and DB2 Cube Views

► The Launchpad now installs both DB2 Enterprise Server Edition V8.2.1 and DB2 Cube Views V8.2.1.

> **Note:** You must ensure there is no **db2admin** user ID already defined in Windows. If it does exist, you must remove this user ID, or the installation fails.

The Launchpad guides you through the installation steps and when completed you should see these 2 pre-requisites installed (Figure B-5).
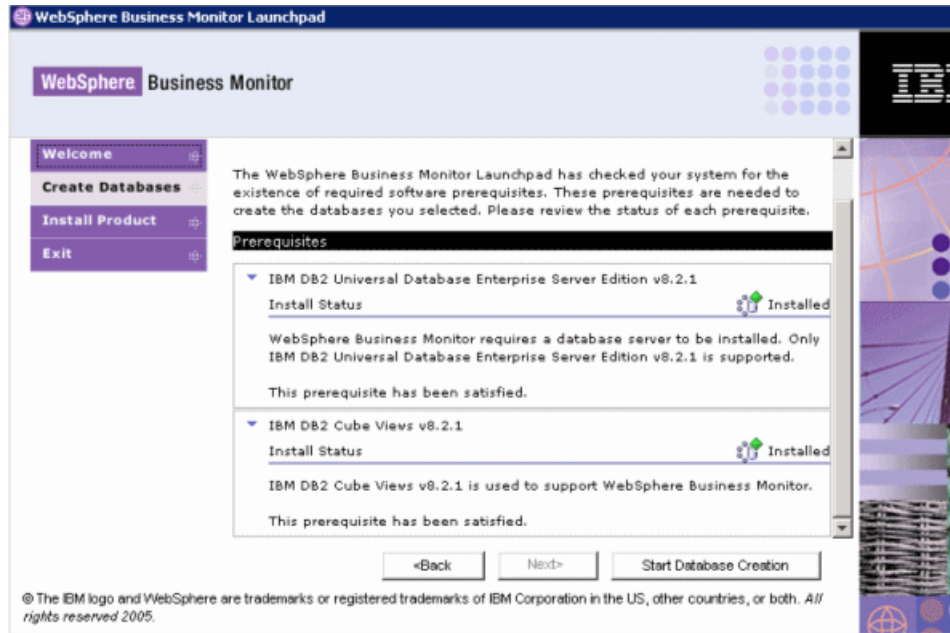
Figure B-5   Monitor Server installation: DB2 installed

► Click *Start Database Creation*.

  The Launchpad starts the WebSphere Business Monitor Installer.

► Accept the license agreement, click *Next* and accept the default installation directory:

    c:\IBM\WebSphere\Monitor

► Click *Next* and enter the database user ID (db2admin) and password, directories for backup and tablespace (which must already exist in the file system) for the STATE database (Figure B-6).

  Click *Next* and enter similar information for RUNTIME, REPOS, and HISTORY databases.
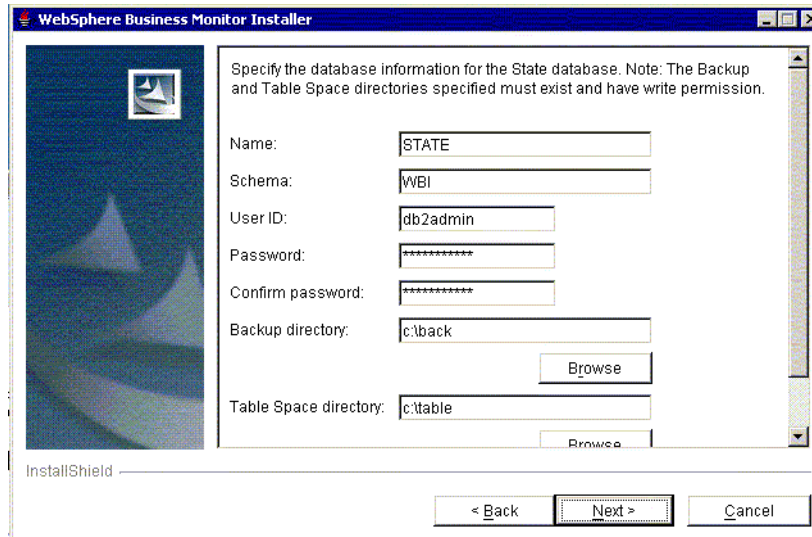
Figure B-6   Monitor Server installation: Database

► A summary panel is displayed before the databases are created (Figure B-7).
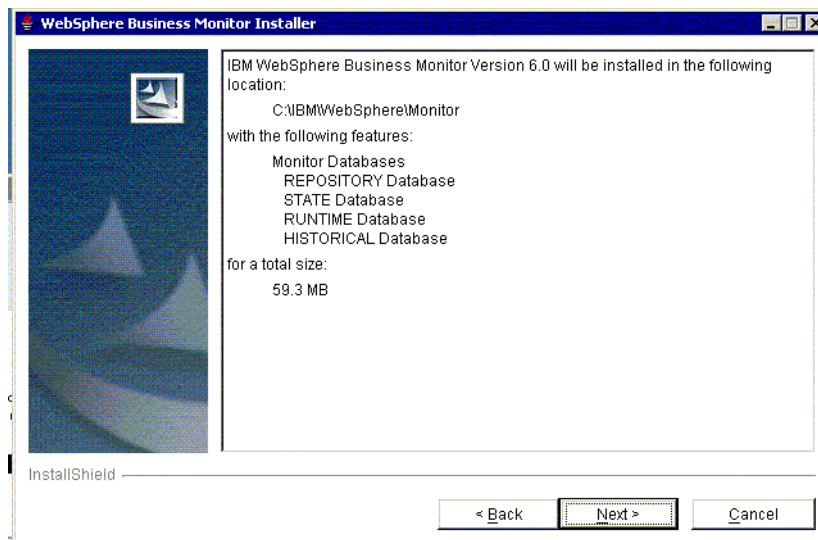
Figure B-7   Monitor Server installation: Database summary

► Click *Next* and the database creation scripts launches four command windows.

Do not close any of the four command windows when they appear or the database creation fails. When each database is created, the command window closes automatically.

► Click *Next* and the Launchpad verifies that all four databases have been created. A confirmation screen displays:

> *"InstallShield Wizard has successfully installed IBM WBI Monitor V6.0. Click Finish to exit the wizard."*

Note: This is somewhat misleading, because you have only created the databases and have not yet installed the Monitor Server.

► Click *Finish* to exit the database creation portion of the wizard.

► You return to the Launchpad where you click *Install Product*. Select *Monitor Server* and click *Next*.

Install Process Server

► The Launchpad verifies the pre-requisite for WebSphere Process Server v6.0.0.0 and because it is not installed yet, prompts you to install it before proceeding with the Monitor Server installation (Figure B-8).
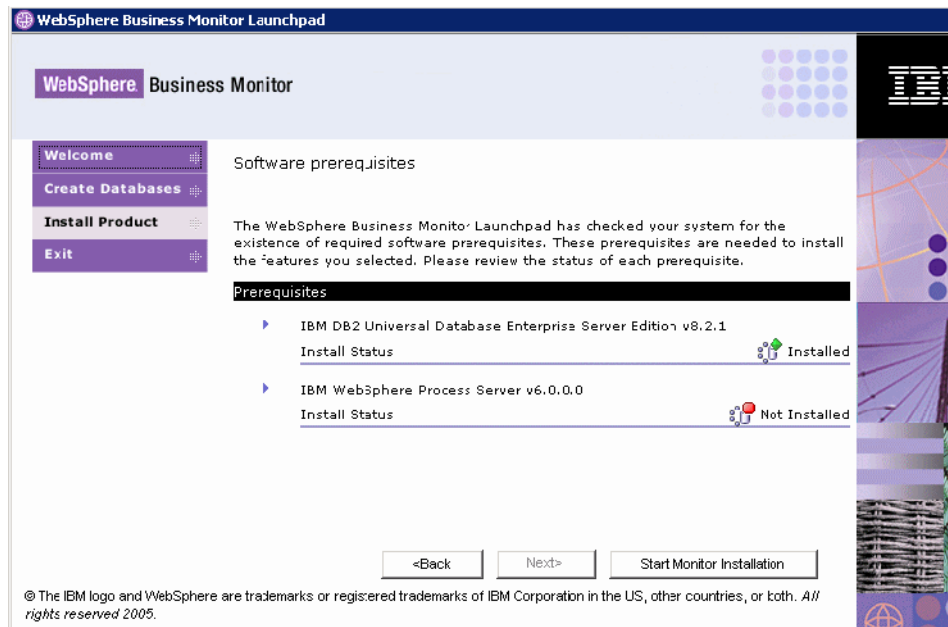
Figure B-8   Monitor Server installation: Process Server pre-requisites

▸ During the first part of the WebSphere Process Server v6.0.0.0 installation, you are prompted to enter the `CEI` database information. Use the same `db2admin` user credentials as before for DB2.

Installing WebSphere Process Server v6.0.0.0 will take a while, up to 45 minutes. When completed the status changes to installed.

Install Monitor Server

▸ Click *Next* to start the installation of Monitor Server. The Launchpad starts the InstallShield Window for WebSphere Business Monitor.

▸ The first panel is for the WebSphere Application Server information. You can keep the defaults, which are determined by your machine's hostname. You should have already defined a hostname for the system that is 8 characters or less. In this case, accept the defaults and click *Next*.

▸ The next panel requests information for the Adaptive Action Manager Catalog. Again, keep the defaults and enter the same database credentials as before for the user ID `db2admin`. Click *Next*.

▸ The summary is displayed (Figure B-9).
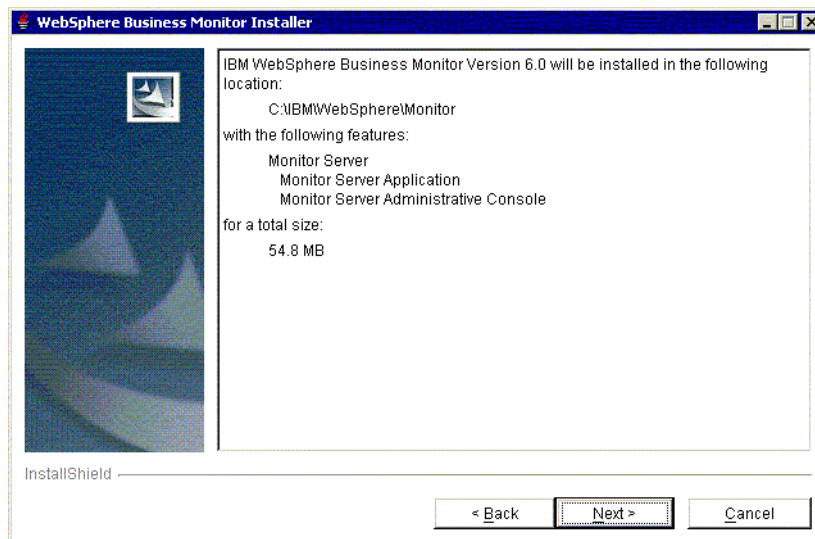
Figure B-9   Monitor Server installation: Monitor summary

▸ Click *Next*. The Monitor Server install will take up to 45 minutes to complete. When completed, you receive the confirmation:

"InstallShield Wizard has successfully installed IBM WBI Monitor V6.0. Choose Finish to exit the wizard".

▶ Click *Finish* and exit the Launchpad.

Monitor Server post-installation checkpoint

Before proceeding with the Monitor Dashboard installation, you should validate that the Monitor Server installation was successful:

▶ Start the DB2 Control Center and validate there are seven databases: AAMCAT, HISTORY, REPOS, RUNTIME, STATE, EVENT, WPRCSDB (Figure B-10).
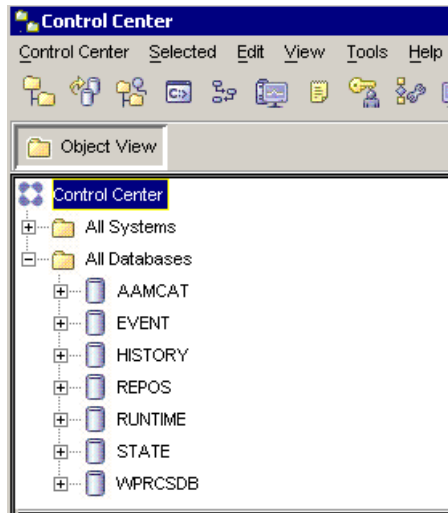
Figure B-10   Post installation: Monitor Server databases

▶ Open the REPOS database, and display the table DATABASE_CHARACTERICTICS. You should see some data that represents your system, for example, Windows, hostname, and so forth (Figure B-11).
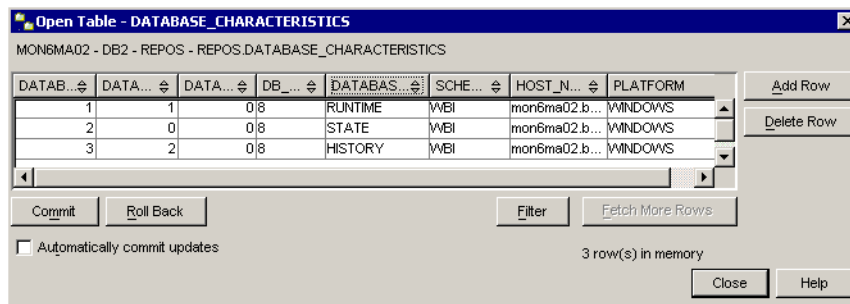
Figure B-11   Monitor Server Repository database table verification

►   If you do not find this information, do not continue with the installation as the server portion of Monitor did not install correctly.

►   Now verify that the WebSphere Application Server for Monitor Server is installed correctly by selecting *Start → Programs → IBM WebSphere → Process Server 6.0 → Profiles → monitor → Start the server*.

►   When the server has started, open the administrative console by selecting *Start → Programs → IBM WebSphere → Process Server 6.0 → Profiles → monitor → Administrative console*. The Enterprise Applications should be started (Figure B-12).
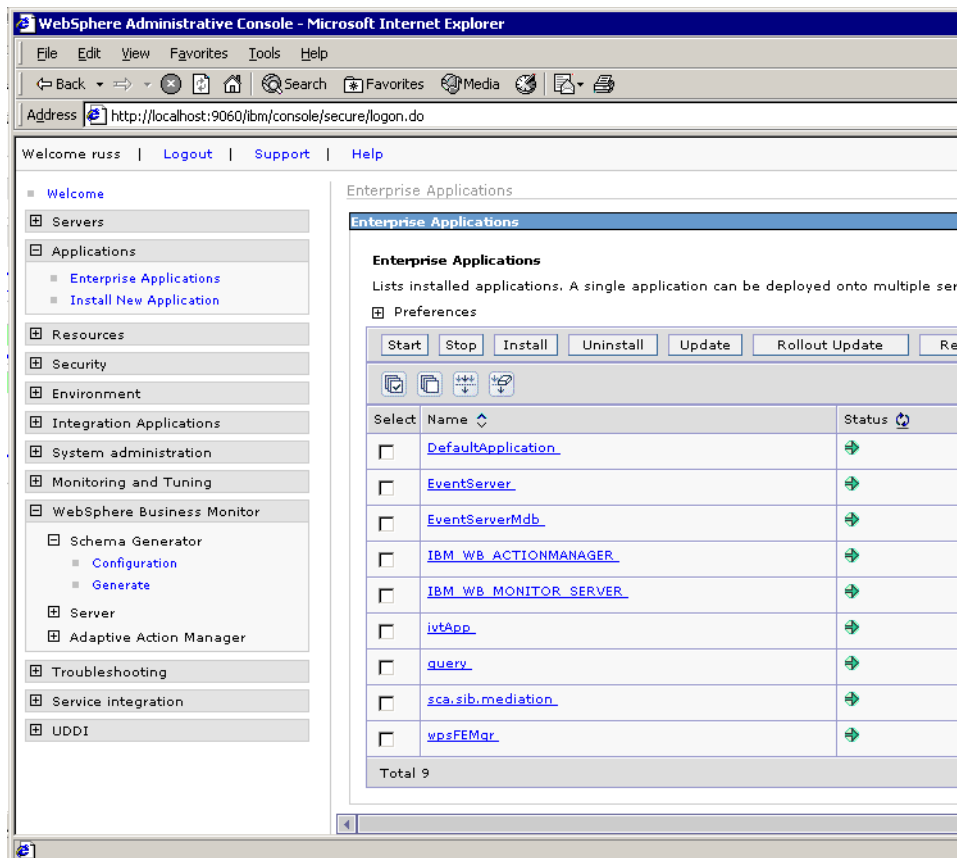
Figure B-12   Monitor Application Server Administrative Console

►   If the post Monitor Server installation validation is satisfactory, proceed to install the Monitor Dashboard on the other Windows 2000 Server machine.

Installing Monitor Dashboard

Note: Before starting, ensure that the hostname of the Monitor Dashboard machine is 8 characters or less.

Launchpad

▶ Start the Launchpad on the Monitor Dashboard machine (`launchpad.bat`).

▶ For the dashboard we do not create any databases because they are all remote cataloged from the Monitor Server machine. Click *Install Product* and select *Dashboard Client*. Click *Next*.

▶ The Launchpad verifies the pre-requisites for the Dashboard Client and displays that none of them are installed (Figure B-13).
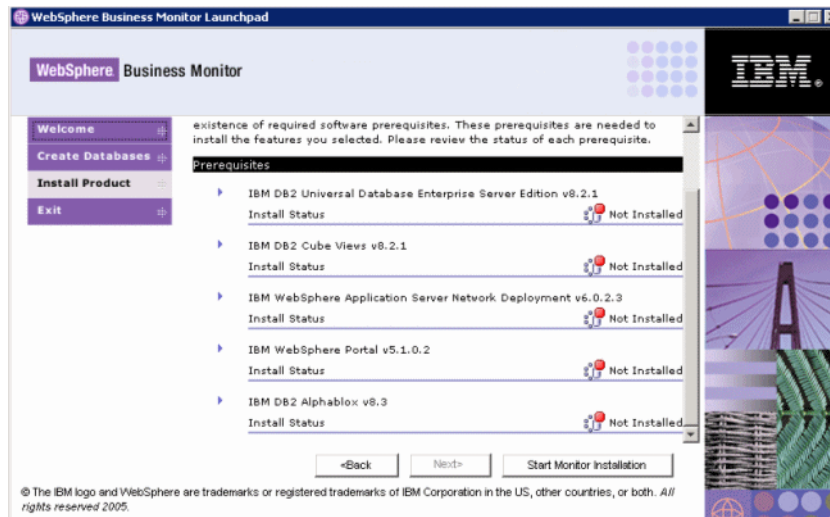
Figure B-13   Monitor Dashboard pre-requisites not installed

Install pre-requisites

▶ Starting with DB2 ESE v8.1.2, use the Launchpad to install to install each of the pre-requisite products for the Dashboard Client.

▶ **Notes**:

– Ensure you do not already have a `db2admin` user on this Windows 2000 Server or the installation fails.

– If your directory structure does not match what is outlined in "Installation directories" on page 419, the Launchpad will prompt you to locate the folder where the pre-requisite files exist.

This is particularly helpful during installation of the pre-requisites. In the case of the required fix pack Portal 5.1.0.2, you will not be prompted to locate the 5.1.0.2 files after the base Portal 5.1 is installed.

– As stated in "Software requirements" on page 418, item 6: On a Windows platform, after installing DB2, you should exit the Launchpad and close Windows Explorer windows and restart the Launchpad to proceed with the remaining pre-requisite installation.

► When you have completed installing all of the pre-requisites for the Dashboard Client, you get a confirmation (Figure B-14).
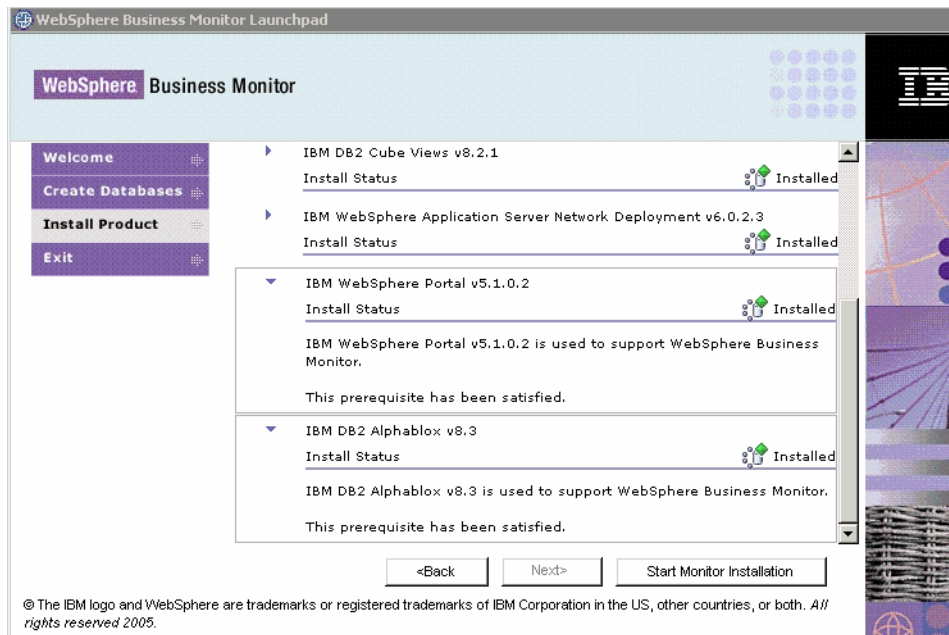
Figure B-14   Monitor Dashboard installation: Pre-requisites completed

Monitor Dashboard Client remote database configuration

Referring to Figure 12-5 in "ClipsAndTacks Monitor topology and configuration" on page 298, we have the run-time databases hosted on the Monitor Server machine for simplicity.

Therefore, before you use the Launchpad on the Monitor Dashboard machine to install the Monitor Dashboard Client, you have to catalog three remote DB2 databases from the Monitor Server machine:

► You can perform this task either with the DB2 Configuration Assistant or with the DB2 Command Line Processor.

► You have to catalog the `HISTORY`, `RUNTIME`, and `REPOS` databases remotely from the Monitor Server host machine.

► When done, open the DB2 Control Center on the Monitor Dashboard machine and verify that you can open the three databases from this machine. You are prompted for the `db2admin` credentials of the Monitor Server machine.

► For detailed instructions on how to catalog remote DB2 databases, refer to the DB2 ESE documentation on how to use the DB2 Configuration Assistant or DB2 Command Line Processor commands.

Note: Exit the Launchpad to perform the Monitor Dashboard Client remote database configuration. Catalog the databases, then restart the launchpad.

Monitor Dashboard Client installation procedure

In the Lauchpad click *Start Monitor Installation*. A command window opens and starts the installation.

> **Note:** Do not close the command window or any additional command windows that are launched in the background during the installation or installation fails.

► For our installation we first receive a warning that the repository database is not available (Figure B-15).
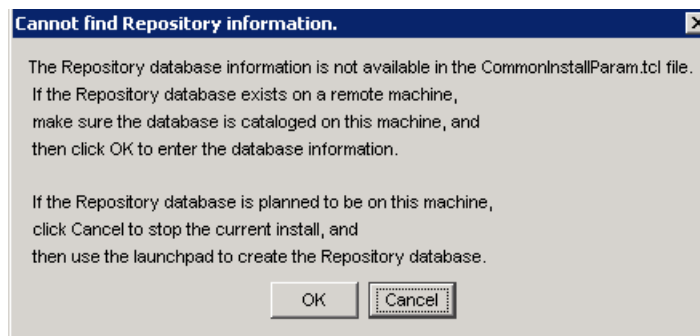
Figure B-15   Repository database not found on localhost

In our topology, the Monitor Dashboard Client uses the databases remotely on the Monitor Server machine, so this is expected.

► Click *OK* and enter the repository database information (Figure B-16).
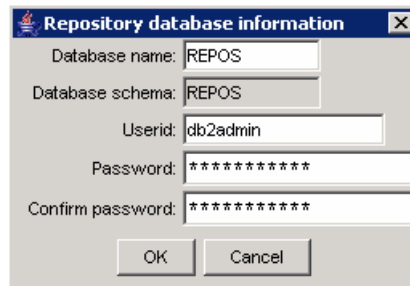
Figure B-16   Repository database information

► Click *OK* and the WebSphere Business Monitor Installer window opens.

► Accept the license agreement, click *Next* and accept the default installation directory:

    C:\IBM\WebSphere\Monitor

► Click *Next* and the portal settings are displayed (Figure B-17).
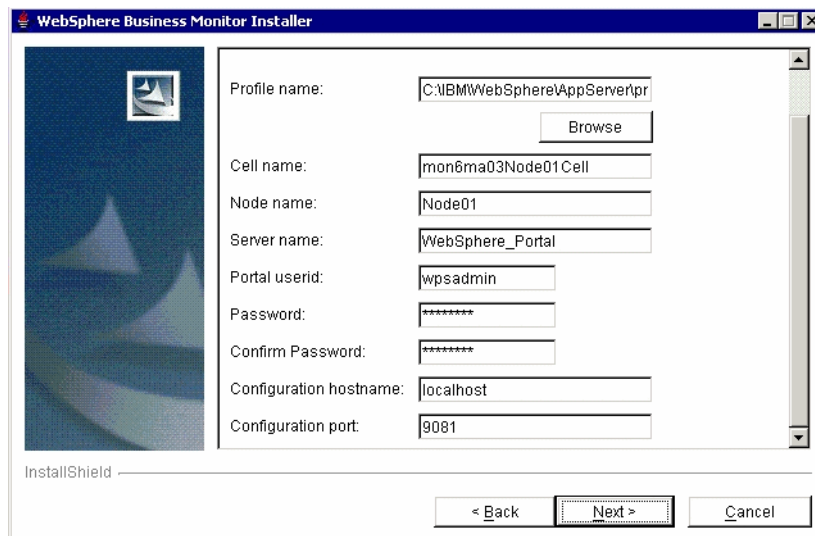
Figure B-17   Monitor Dashboard Client: Portal settings

► Accept the default settings and Portal user ID and password and click *Next*.

► The DB2 Alphablox settings are displayed (Figure B-18).
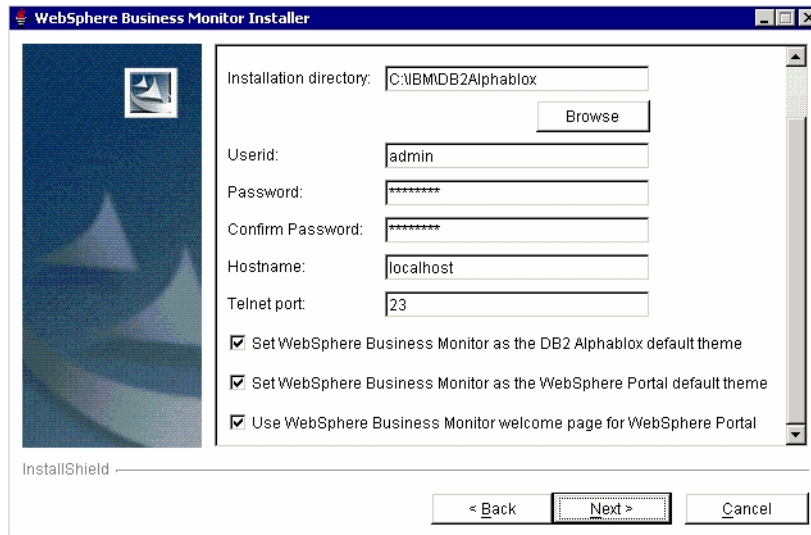
Figure B-18   Monitor Dashboard Client: DB2 AlphaBlox settings

► Click *Next* to proceed with the installation. It can take up to 1 hour, depending on your system. When completed, you get the confirmation (Figure B-19).
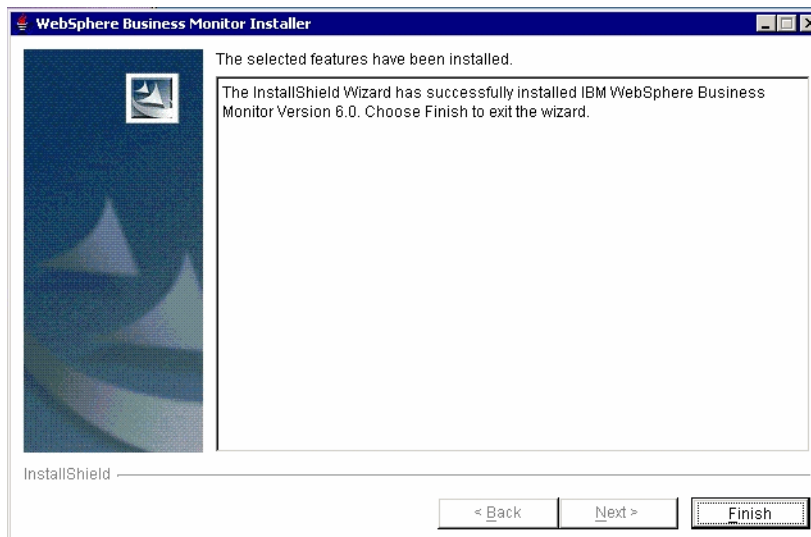
Figure B-19   Monitor Dashboard Client: Installation complete

► Click *Finish*.

Monitor Dashboard Client post-installation checkpoint

To verify that your installation was successful, start the Portal Server by selecting *Start → Programs → IBM WebSphere → Portal Server v5.1 → Start the Server*.

When the portal server is ready, start a browser with this URL (Figure B-20):

```
http://localhost:9081/wps/portal        <=== your port may be different
```
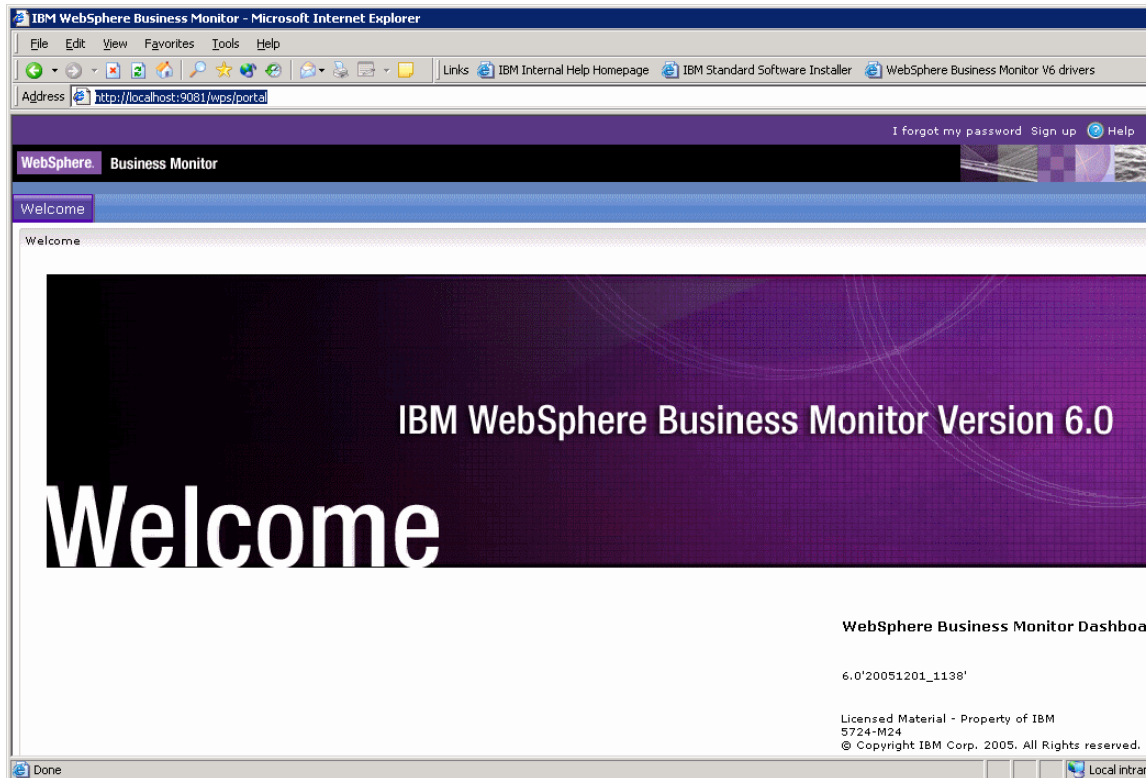

Figure B-20   Monitor Dashboard: Welcome screen

Monitor Server service integration bus

The Monitor Server communicates with the WebSphere Process Server that runs the ClipsAndTacks business process on another machine. This communication is performed through a service integration bus (SIB) that must be configured.

The SIB is the transport by which the common business events (CBE) are transmitted from the Process Server 6.0.1 to the Monitor Server.

SIB configuration for the Monitor Server

First we configure the SIB on the Monitor Server machine:

► Locate the file:

```
C:\IBM\WebSphere\Monitor\install\monsrv\configuration\crosscell\
                                         crossCellParameters.tcl
```

► Edit the file to reflect your environment for host and WebSphere configuration. You have to enter the cell, node, and hostname (or IP address) for both the Monitor Server and Process Server machines. A sample script is shown in Figure B-21.

```
📄 C:\IBM\WebSphere\Monitor\install\monsrv\configuration\crosscell\crossCellParameters.tcl    ▬

# --------------------------
# Monitor Server Parameters
# --------------------------
set MONITOR_SERVER_CELL_NAME    "mon6ma02Node01Cell"
set MONITOR_SERVER_NODE_NAME    "Node01"
set MONITOR_SERVER_SERVER_NAME  "server1"
set MONITOR_SERVER_HOSTNAME     "9.26.238.73"

# --------------------------
# Process Server Parameters
# --------------------------
set PROCESS_SERVER_CELL_NAME    "mon6ma01Node01Cell"
set PROCESS_SERVER_NODE_NAME    "mon6ma01Node01"
set PROCESS_SERVER_SERVER_NAME  "server1"
set PROCESS_SERVER_HOSTNAME     "9.26.238.116"
```

Figure B-21   Monitor Server: Cross cell parameter configuration

► Save the file and start both the Monitor Server and the process Server, if they are not started already.

> **Important:** Throughout the entire procedure of configuring the SIB on both the Monitor Server and Process Server machines, ensure that all of the tcl parameters are correct for your system prior to executing the wsadmin scripts.
>
> Once the SIB is configured for a given Application Server, the SIB configuration is stored persistently in the CEI database on the Process Server machine. If you execute the scripts with incorrect parameters, you will have to run a special cleanup procedure.

► From a command window on the Monitor Server, change the directory to:

```
C:\IBM\WebSphere\Monitor\install\monsrv\configuration\crosscell
```

► Execute this command:

```
C:\IBM\WebSphere\ProcServer\profiles\monitor\bin\wsadmin.bat
    -f C:\IBM\WebSphere\Monitor\install\monsrv\configuration\crosscell\
                                        configureMonitorCrossCell.tcl
```

Note: You execute wsadmin with the script configureMonitorCrossCell.tcl, even though you modified the file crossCellParameters.tcl.

► The SIB is configured on the Monitor Server machine.

► Restart the Monitor Server to allow these changes to take effect.

SIB configuration for the Process Server

Now we configure the SIB on the Process Server machine:

► Copy the crosscell folder (and containing files) from the Monitor Server machine to the Process Server machine:

```
From: C:\IBM\WebSphere\Monitor\install\monsrv\configuration\crosscell
To:   C:\crosscell                      (for example)
```

► Open a command window on the Process Server machine in the crosscell directory. Execute the command:

```
C:\IBM\WebSphere\ProcServer\profiles\monitor\bin\wsadmin.bat
    -f C:\IBM\WebSphere\Monitor\install\monsrv\configuration\crosscell\
                                        configureCrossCell.tcl
```

Notice that we are executing the file configureCrossCell.tcl (this is different from the Monitor Server machine).

► The SIB is configured on the Process Server machine.

► Restart the Process Server to allow these changes to take effect.

Service integration bus post-configuration checkpoint

After configuring the SIB for both the Monitor Server and Process Server, it is important to validate that the SIB is functional between the two hosts and that SIB messages are consistent:

► In the SystemOut.log on the Monitor Server, search for traces labeled SibMessage:

```
[12/10/05 0:24:36:734 EST] 000027ed SibMessage   I   [:] CWSIT0032I: The
inter-bus connection mon6ma02Node01Cell.mon6ma01Node01Cell.link from
messaging engine Node01.server1-MONITOR.mon6ma02Node01Cell.Bus in bus
MONITOR.mon6ma02Node01Cell.Bus to messaging engine
mon6ma01Node01.server1-MONITOR.mon6ma01Node01Cell.Bus in bus
MONITOR.mon6ma01Node01Cell.Bus started.
```

```
[12/10/05 0:24:37:969 EST] 000027f0 SibMessage    I
[MONITOR.mon6ma02Node01Cell.Bus:Node01.server1-MONITOR.mon6ma02Node01Cell.B
us] CWSIP0382I: messaging engine 88B26453C991E694 responded to subscription
request, Publish Subscribe topology now consistent.
```

► In the `SystemOut.log` on the Process Server, do the same:

```
[12/10/05 9:49:45:641 EST] 00000016 SibMessage    I
[MONITOR.mon6machine1Node01Cell.Bus:Node01.server1-MONITOR.mon6machine1Node
01Cell.Bus] CWSIP0212I: messaging engine
Node01.server1-MONITOR.mon6machine1Node01Cell.Bus on bus
MONITOR.mon6machine1Node01Cell.Bus is starting to reconcile the WCCM
destination and link configuration.
[12/10/05 9:49:45:656 EST] 00000016 SibMessage    I
[MONITOR.mon6machine1Node01Cell.Bus:Node01.server1-MONITOR.mon6machine1Node
01Cell.Bus] CWSIP0213I: messaging engine
Node01.server1-MONITOR.mon6machine1Node01Cell.Bus on bus
MONITOR.mon6machine1Node01Cell.Bus has finished reconciling the WCCM
destination and link configuration.
[12/10/05 9:49:45:703 EST] 00000016 SibMessage    I
[MONITOR.mon6machine1Node01Cell.Bus:Node01.server1-MONITOR.mon6machine1Node
01Cell.Bus] CWSID0016I: Messaging engine
Node01.server1-MONITOR.mon6machine1Node01Cell.Bus is in state Started.
```

Note: You may see SIB messages in the log that indicate communication is not established:

```
[12/10/05 0:29:37:531 EST] 0000002a SibMessage    W
[MONITOR.mon6ma02Node01Cell.Bus:Node01.server1-MONITOR.mon6ma02Node01Cell.B
us] CWSIP0381W: No Response received from messaging engine BF8AAD2A86FA1464
for subscription request message.
```

This happens in the case when one of the servers has stopped due to a restart, or if there is a network communication failure between the hosts. Do not be concerned with this condition, because the SIB will send keep-alive type requests periodically to re-establish communications.

C
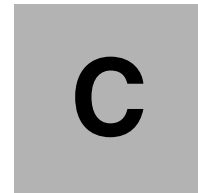
Additional material

This redbook refers to additional material that can be downloaded from the Internet as described below.

Locating the Web material

The Web material associated with this redbook is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

   ftp://www.redbooks.ibm.com/redbooks/SG247148

Alternatively, you can go to the IBM Redbooks Web site at:

   ibm.com/redbooks

Select the **Additional materials** and open the directory that corresponds with the redbook form number, SG24-7148.

Using the Web material

The additional Web material that accompanies this redbook includes the following files:

> *File name*   *Description*
>
> **sg247148code.zip**   Zipped code samples
> corrections7148.txt   Corrections to the redbook

System requirements for using the Web material

The following system configuration is recommended for our redbook scenario for Modeler, Integration Developer, and Process Server. Note that a Monitor installation requires multiple machines as described in Appendix B, "Installation of WebSphere Business Monitor" on page 417.

Hard disk space: For the base products you require:

- ► Modeler—About 1 GB (product and workspace)
- ► Integration Developer—About 5 GB
- ► Process Server—About 1.5 GB

Operating System: Windows XP/2000 or Linux®. We only used Windows when writing this redbook.

Processor: At least 2 GHz

Memory: 1.5 GB or better

Memory requirements by product

Typical memory requirements for working with our scenario are:

- ► Modeler: 400 MB

- ► Integration Developer: 500 - 600 MB

- ► Process server: 400 - 500 MB

To run Integration Developer together with the Process Server test environment you require a machine with at least 1.5 GB of memory, but 2 GB is better.

How to use the Web material

Unzip the contents of the Web material **sg247148code.zip** file onto your hardrive. This creates a directory named **SG247148** with a number of subdirectories.

```
sampcode                Main directory with sample code
  _setupCloudscape      Files to create and list the CLIPTACK database
  cloudscape            Files for Cloudscape
    databases\CLIPTACK  Copy of the initial CLIPTACK database
    datasource          JACL script to define the data source
  model                 Interchange files to import models per chapter
    export              Directory for your exports
      noMonitor         Export of Future1 without business measures
      withMonitor       Export of Future1 with Monitor data
      Future2           Export of Future2 with Monitor data
  monitor               Export data from Monitor
  wid                   Helper code for Integration Developer
    businessRuleManager Command file to install Business Rules Manager
    creditRating        Java code for customer credit rating check
    humantaskBPC        WAR file with human task customized JSPs
    humantaskSecurity   Groups, users, and code for human task security
    webfront            WAR file with Web front-end application
    webservice          EAR file with credit check Web service
    zInterchange        Interchange files of completed applications
  wps                   EAR files for installation in Process Server
```

The instructions on how to use these files are given in the respective chapters. Here we present only a short extract of major activities.

Creating the CLIPTACK database

The CLIPTACK database holds the customer and order information for thebusiness process.

The CLIPTACK database is automatically created by defining the data source in the server and testing the data source. This can be done under Integration Developer or in the real Process Server:

► In "Create a data source for the database" on page 194 we describe how to create the data source manually using the administrative console.

► In "Using a JACL script to define the data source" on page 253 we describe how to craete the data source using a JACL script.

Either method can be used. When the data source is defined, use the test data source facility in the administrative console and the database is created. To populate the database you have to run the CreateDatabaseServlet as described in "Initialize the database" on page 223.

Importing models

We provide the final business process model for each chapter that deals with the Modeler:

```
SG247148\sampcode\model\Clips And Tacks Chapter 4 Solution.zip
                        \Clips And Tacks Chapter 5 Solution.zip
                        \Clips And Tacks Chapter 6 Solution.zip
                        \Clips And Tacks Chapter 7 Solution.zip
                        \Clips And Tacks Chapter 8 Solution.zip
                        \Clips And Tacks Future 2.zip
```

These models can be imported into Modeler into separate Modeler projects as described in the respective chapters.

Importing application solutions

The completed applications of Integration Developer are provided in

```
SG247148\sampcode\wid\zInterchange\ClipsAndTacks-Interchange.zip
                                  \ClipsAndTacks2-Interchange.zip
                                  \ClipsAndTacksHumanCustom-Interchange.zip
                                  \ClipsAndTacksService-Interchange.zip
```

These interchange files can be imported into Integration Developer as described in "Importing the human task application" on page 238. Note that any existing code of the same projects is deleted first.

The completed exported enterprise applications (EAR files) are available in

```
SG247148\sampcode\wps\ClipsAndTacksApp.ear
                      \ClipsAndTacks2App.ear
                      \ClipsAndTacks2AccountStatusApp.ear
                      \ClipsAndTacksServiceEAR.ear
                      \ClipsAndTacksHumanCustomEAR.ear
```

These EAR files can be installed in a real Process Server for execution after configuring the server with the data source. See Chapter 10, "Deploying and running the application in Process Server" on page 251 for detailed instructions.

Abbreviations and acronyms

API	Application program interface		**HTML**	Hypertext Markup Language
ATM	Automatic teller machine		**HTTP**	Hypertext Transfer Protocol
BI	Business integration		**IBM**	International Business Machines
BPC	Business process container		**IDE**	Integrated development environment
BPEL	Business Process Execution Language		**IMS**	Information Management System
BPIA	Business process integration and automation		**IP**	Internet protocol
BPM	Business process management OR business performance management		**IT**	Information technology
			ITSO	Internatoinal Technical Support Organization
CBE	Common base event		**JAAS**	Java Authentication and Authorization Service
CEI	Common event infrastructure			
CICS	Customer Information Control System		**JACL**	Java command language
CPM	Corporate performance management		**JDBC**	Java Database Connectivity
			JMS	Java Messaging Service
CRM	Customer relationship management		**JNDI**	Java Naming and Directory Interface™
DAO	Data access object		**JSF**	JavaServer Faces
DB	Database		**JSP**	JavaServer Pages™
DDL	Data description language		**KPI**	Key performance indicator
DTO	Data transfer object		**LDAP**	Lightweight Directory Access Protocol
EAR	Eanterprise application archive			
EIS	Enterprise informaiton system		**LOB**	Line of business
EJB	Enterprise JavaBean		**LTPA**	Leightweight Third Party Authentication
EPM	Enterprise performance management			
			MOM	Message oriented middleware
ERP	Enterprise resource planning		**MQ**	Message queue
ESB	Enterprise service bus		**ND**	Network deployment
EST	Eatern Standard Time		**OASIS**	Organization for the Advancement of Structured Information Standards
GMT	Greenwihch Mean Time			
GUI	Graphical user interface		**OLAP**	Online analytical processing

OS	Operating system
PST	Pacific Standard Time
PTF	Program temporary fix
RMI	Remote method interface
SCA	Service Component Architecture
SDO	Service Data Object
SIB	Service integration bus
SMS	Short message service
SOA	Service oriented architecture
SOAP	Simple object access protocol
SOX	Sarbanes-Oxley
SQL	Structured query language
SSO	Single signon
TKIID	Task instance object identifier
UCM	Universal configuration management
URL	Universal resource locator
USD	US Dollar
WAR	Web application archive
WBI	WebSphere Business Integration
WSDL	Web Services Description Language
XML	Extensible Markup Language
XSD	XML schema definition

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see "How to get IBM Redbooks" on page 447. Note that some of the documents referenced here may be available in softcopy only.

- ► *Best Practices for Using WebSphere Business Modeler and Monitor*, REDP-4159
- ► *Getting Started with WebSphere Integration Developer and WebSphere Process Server,* SG24-7130
- ► *Patterns: WebSphere Process Server with Serial / Parallel Processes,* SG24-7205
- ► *Technical Overview of WebSphere Process Server and WebSphere Integration Developer,* REDP-4041
- ► *Build a Business Process Solution using Rational and WebSphere Tools*, SG24-6636
- ► *BPEL4WS Business Processes with WebSphere Business Integration: Understanding, Modeling, Migrating,* SG24-6381
- ► *Business Integration Management using WebSphere BI Modeler and Monitor A Real World Case Study,* SG24-7024
- ► *Business Performance Management . . . Meets Business Intelligence*, SG24-6340
- ► *WebSphere Version 6 Web Services Handbook Development and Deployment,* SG24-6461
- ► *WebSphere Application Server V6: Planning and Design*, SG24-6446
- ► *WebSphere Application Server V6: Security Handbook*, SG24-6316
- ► *WebSphere Application Server V6: System Management and Configuration Handbook*, SG24-6451
- ► *Rational Application Developer V6 Programming Guide*, SG24-6449
- ► *WebSphere Application Server V6: Scalability and Performance Handbook*, SG24-6392

► *WebSphere Application Server V6 Technical Overview*, REDP-3918

► *Software Configuration Management: A Clear Case for IBM Rational ClearCase and ClearQuest UCM*, SG24-6399

Online resources

These Web sites and URLs are also relevant as further information sources:

► WebSphere Business Integration products Web sites:

– WebSphere Business Modeler

`http://www.ibm.com/software/integration/wbimodeler/`

– WebSphere Integration Developer

`http://www.ibm.com/software/integration/wid/`

– WebSphere Process Server

`http://www.ibm.com/software/integration/wps/`

– WebSphere Business Monitor

`http://www6.ibm.com/software/integration/wbimonitor/`

– Information Center

`http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp`

► DeveloperWorks:

`http://www.ibm.com/developerworks/websphere`
`http://www.ibm.com/developerworks/websphere/zones/businessintegration/`

► Business Process Execution Language:

`http://www.ibm.com/developerworks/webservices/library/ws-bpel/`

► Service Component Architecture and Service Data Objects:

`http://www.ibm.com/developerworks/webservices/library/specification/ws-sca/`
`http://www.ibm.com/developerworks/library/specification/ws-sdo/`

► Business process management:

`http://www.research.ibm.com/journal/sj/412/leymann.html`
`http://www.bpmi.org/`
`http://en.wikipedia.org/wiki/Business_Process_Management`

► Business process integration and automation:

`http://bpia.zurich.ibm.com/`

► Scalable Vector Graphics (SVG) viewer:

`http://www.w3.org/Graphics/SVG/SVG-Implementations`

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Numerics
90 Percent of Orders Are Approved KPI   164, 169

A
action alert   299, 352
action manager   294
active instances   27
Active Instances view   153, 322, 334, 396
   details   335
   filter   336
activity duration   129
adaptive action manager   292, 294, 309
administrative console   192
   Monitor   300
   Process Server   413
advanced mode   47
aggregate metric   150
aggregate tables   332
aggregation function   149
alerts   309, 352
Alerts view   295, 352
   wiring   353
analysis
   dimensional   171
   simulation   74
animation   135
annotation   103
application
   change   258
   deployment   251
   development   21
   export   252
   implementation   188
   installation   255, 387
   programmer   35
   projects   199
   run   229, 256
   testing   208
   uninstall   258
assembly diagram   201, 216
   future 2   376
   Web service   235
automaticApproval   53

B
availability matrix   58, 127
AvailableCredit   53
Average Order Fulfillment is 3 Days or Less KPI
158, 162

benchmark   59
BOFactory   220
BPC Explorer   208
   configuration   226
   custom JSP   225
   custom JSPs   231
   run   257
   start   211
   tasks list   212
   terminate tasks   258
   work on   228
BPEDB database   415
BPEL
   see Business Process Execution Language
BPM
   see business process management
bulk resources   89
business analyst   34, 42
business flexibility   18
business innovation and optimization   4
business integration   21
Business Integration perspective   222
Business Integration view   200
business intelligence   14
business items   42, 44, 52
business measures   146
   changing   326
   future 2   369
   import   301, 307
   model   146, 148, 288
      create   154
      deployment   300
      export   180
      Monitor   293
   synchronize   370
   unload   327
Business Measures editor   148

future model   84

G
gauge   28
Gauges view   153, 341, 390
global
    security   281
    task   366, 377
governance   7
groups.props   268

H
hardware   37
History database   298
    tables   332
human task   20, 25, 113
    application   237
        run   242, 257
        security   277
    BPC Explorer   211
    configuration   207
    container   414
    duration time   63
    future 2   381
    implementation   113
    JSPs   225
    processing logic   227
    security   267, 274
        testing   276
Human Task Manager   414

I
implementation type   111
import
    model   47
    Monitor results   360
    project interchange file   198
improvement strategy   363
inbound event   152
individual resources   89
industry standards   17
integrated development environment   20
Integration Developer
    see WebSphere Integration Developer
interface map   20, 24

J
JACL script   253, 262
Java activity   205
Java implementation type   114
JavaScript   230
JavaServer Faces   26, 240
JDBC provider   194
jdbc/cliptack   196, 221, 253

K
keeping processes   202
key performance indicators   14, 27, 33, 148
    ClipsAndTacks   154
    implement   158
Key Performance Indicators view   340
KPI
    see key performance indicators
KPIs view   391

L
Late Average Order Shipped Event   176
LDAP   25, 267
    configuration   353
    query   310
    server   280
    staff plugin provider   280
    user registry   282
library project   376
lifecycle   6, 147
limit   149, 163
ListServlet   219
location dimension   172
LTPA   272

M
map   24
matrix   56
mediation   24
merge   95
metric   34, 150
Microsoft Visio   47
mode   47
model   16
    export   118
    future   84, 366
    import   47, 198
model_cv.xml   308

IBM ®

WebSphere Process Integration V6: Business Process Management Modeling through Monitoring

Redbooks

WebSphere Business Modeler

WebSphere Integration Developer

WebSphere Process Server

WebSphere Business Monitor

This redbook presents a business process management (BPM) "round trip" scenario, showing how a business can use a full business integration solution to complete the following tasks:

► Model and simulate a business process

► Develop and test an application to implement the business process

► Deploy and run the application on a server

► Monitor the application to observe pre-determined key performance indicators

► Import the observed data to make revisions to the original process model

The business scenario described in this document has been simplified to provide a full description of each stage of the BPM end-to-end process. To avoid an overly large and unwieldy document, the authors' focus is on specific tasks, elements, and details, and not on presenting all possible facets of a complex business process.

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

SG24-7148-00   ISBN