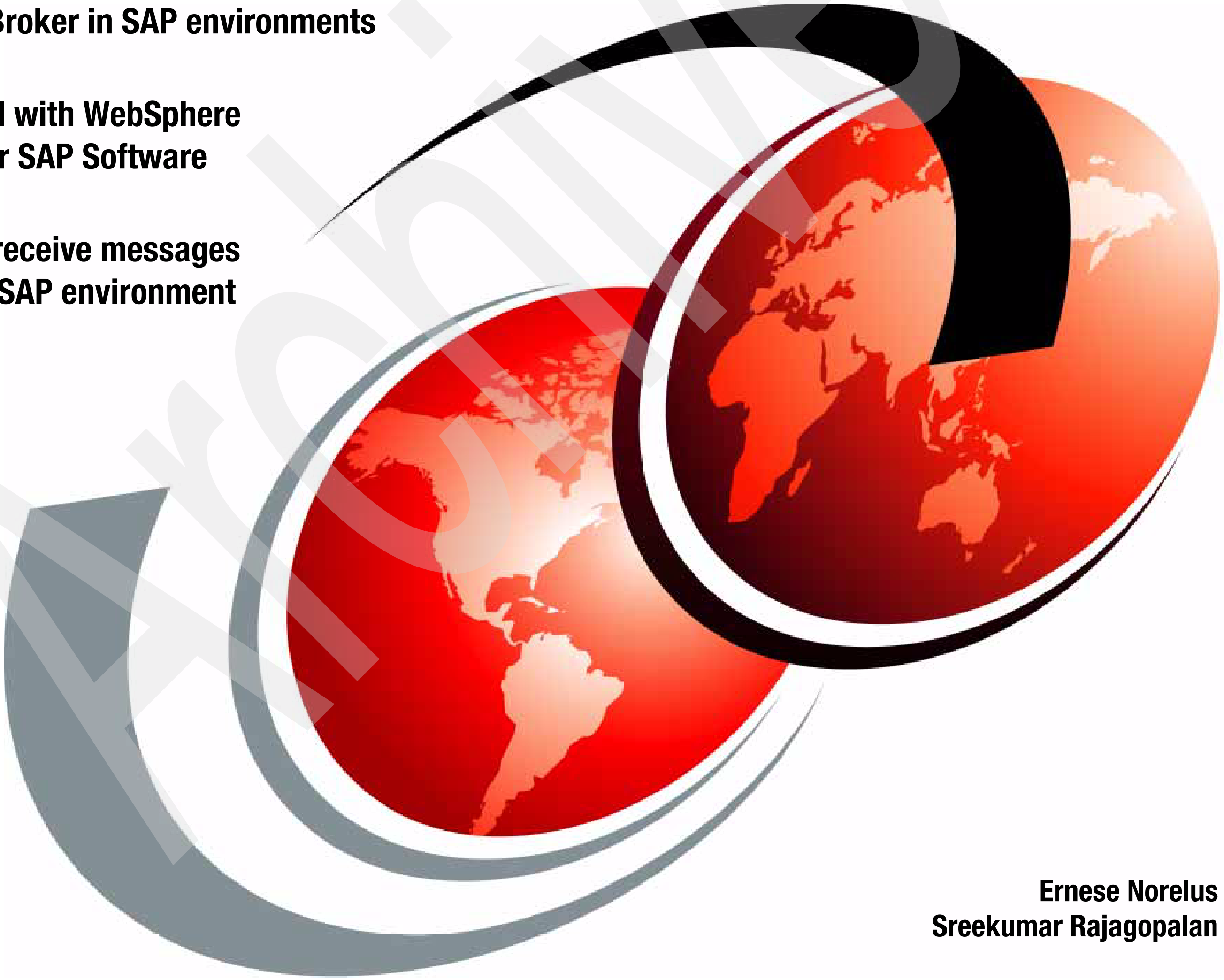
WebSphere Message Broker V7.0 Integration with WebSphere Adapter for SAP Software

Understand the value of WebSphere Message Broker in SAP environments

Get started with WebSphere Adapter for SAP Software

Send and receive messages from your SAP environment



Ernese Norelus
Sreekumar Rajagopalan

Redpaper

ibm.com/redbooks

International Technical Support Organization

WebSphere Message Broker V7.0 Integration with WebSphere Adapter for SAP Software

February 2010

REDP-4644-00

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

First Edition (February 2010)

This edition applies to Version 7.0 of WebSphere Message Broker.

This document created or updated on February 15, 2010.

Contents

Notices v
Trademarks vi

Preface vii
The team who wrote this paper vii
Now you can become a published author, too! vii
Comments welcome. viii

Chapter 1. Introduction. 1
1.1 Executive summary. 2
1.1.1 Intended audience 2
1.1.2 Prerequisites 2
1.2 Overview of WebSphere Adapter for SAP Software 2
1.3 WebSphere MQ 3
1.4 WebSphere Message Broker 3
1.5 WebSphere adapters 4
1.6 WebSphere Adapter for SAP Software 4
1.6.1 Outbound event processing 5
1.6.2 Inbound event processing. 5
1.7 Interacting with SAP Software. 6
1.8 Preparing the SAP environment 6
1.9 SAP Java Connector. 6
1.10 SAP JCo architecture 7
1.11 SAP JCo functions 7
1.12 SAP JCo installation 8

Chapter 2. The exercise: Developing message flows 11
2.1 SAP Inbound sample scenario 12
2.2 Configure the SAP source server 17
2.3 Launch the Command Console. 32
2.4 Deploy the bar file to the broker 34
2.5 Launch the IBM WebSphere Message Broker Explorer plug-in for WebSphere MQ Explorer 38
2.6 Create RFC connection from the SAP system to the broker. 41
2.7 Export the MASM05 project 53

Appendix A. Support information 57
Additional information resources 58
SAP terminology 59
Target system 60

Notices

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

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

Trademarks

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

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

developerWorks®
IBM®
Redbooks®
Redpaper™
Redbooks (logo) ®
WebSphere®
z/OS®

The following terms are trademarks of other companies:

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation 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 IBM® Redpaper™ publication describes how to get started using WebSphere® Adapter for SAP Software with WebSphere Message Broker. These products enable processes and components to be integrated to include the exchange of information with an SAP server, without special coding.

This paper shows how to use an adapter, an application component, to send requests to the SAP server or to receive events from the server. The adapter creates a standard interface to the applications and data on the SAP server so that the developer of the application component does not have to understand the lower level details (the implementation of the application or the data structures) on the SAP server.

The team who wrote this paper

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

Ernese Norelus is an IBM Certified Consulting IT Specialist with the ASEAN Software Services team in Singapore. In this role, he consults with clients to help define IT solutions to complex business problems. With more than 12 yeas of IT experience, Ernese publishes material within IBM providing best practice guidance with WebSphere and cross-brand products and has authored additional IBM Redbooks® publications. He holds a degree in Computer Science from the Université du Québec à Montréal (UQAM), Quebec, Canada.

Sreekumar Rajagopalan has worked with SAP since the late 1990s in various capacities, including project management and as a functional and technical expert. He is also a past speaker at 2002 Americas' SAP Users' Group (ASUG) annual conference. Since 2001, he has expanded his core SAP skills to the area of application integration. Currently, he is part of WebSphere sales team covering the Mid-Atlantic region of the United States.

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

Carla Sadtler
IBM International Technical Support Organization, Poughkeepsie Center

LindaMay Patterson
IBM International Technical Support Organization, Rochester Center

Linda Robinson
IBM International Technical Support Organization, Raleigh Center

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author - all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

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 papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an e-mail to:

  redbooks@us.ibm.com

- Mail your comments to:

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

1

Introduction

This chapter describes how to get started using WebSphere Adapter for SAP Software with WebSphere Message Broker.

It introduces you to the key products that enable you to send requests and receive events from SAP applications. The products introduced are:

- WebSphere Adapter for SAP Software
- WebSphere Message Broker
- WebSphere MQ
- WebSphere Adapters

1.1 Executive summary

SAP (a leading provider of global Enterprise Resource Planning (ERP) systems) and IBM (providing the leading software platform for e-business on demand) are cooperating to offer joint capabilities that permit seamless access to SAP applications and resources from portals, mobile applications, other SAP systems and a variety of enterprise applications.

In this paper we describe how to get started using WebSphere Adapter for SAP Software with WebSphere Message Broker, including how processes and components can be integrated to include the exchange of information with an SAP server, without special coding.

By using the adapter, an application component can send requests to the SAP server or receive events from the server. The adapter creates a standard interface to the applications and data on the SAP server so that the developer of the application component does not have to understand the lower-level details (the implementation of the application or the data structures) on the SAP server.

1.1.1 Intended audience

This paper demonstrates the value of IBM WebSphere Message Broker in an SAP environment. It is written for IBM and SAP technical sales personnel, customers, and partners. The paper assumes that readers have a working knowledge of SAP software and a basic familiarity with WebSphere Message Broker. For additional information about IBM WebSphere and SAP software, go to this Web site:

http://www.ibm.com/solutions/sap/us/en/landing/V397844G27598C13.html

1.1.2 Prerequisites

Information about how to install IBM WebSphere Message Broker, IBM WebSphere Message Broker Toolkit, and WebSphere MQ is not within the scope of this paper. Check the product documentation for installation instructions for these products.

In this paper, we assume you are familiar with WebSphere Message Broker and its related products:

- IBM WebSphere Message Broker 7.0
- IBM WebSphere Message Broker Toolkit 7.0
- IBM WebSphere Adapter for SAP Software
- IBM WebSphere MQ 7.0

To follow the step-by-step instructions for the technical examples described in this paper, you must have a development environment up and running for both IBM WebSphere Message Broker and IBM WebSphere MQ, and you must have access to an SAP environment.

1.2 Overview of WebSphere Adapter for SAP Software

With WebSphere Adapter for SAP Software you can create integrated processes that include the exchange of information with an SAP server, without special coding.

Using the adapter, an application component (the program or piece of code that performs a specific business function) can send requests to the SAP server (for example, to query a customer record in an SAP table or to update an order document) or receive events from the server (for example, to be notified that a customer record has been updated). The adapter creates a standard interface to the applications and data on the SAP server, so that the application component does not have to understand the lower-level details (the implementation of the application or the data structures) on the SAP server.

WebSphere Adapter for SAP Software complies with the Java™ Connector Architecture (JCA) 1.5, which standardizes the way in which application components, application servers, and Enterprise Information Systems (EIS), such as an SAP server, interact with each other.

The adapter, which you generate with the Adapter Connection wizard, uses a standard interface and standard data objects. The adapter takes the standard data object sent by the application component and calls the SAP function. The adapter then returns a standard data object to the application component. The application component does not have to deal directly with the SAP function; it is the SAP adapter that calls the function and returns the results.

For example, the application component that needed the list of customers would send a standard business object with the range of customer IDs to the SAP adapter. The application component would receive, in return, the results (the list of customers) in the form of a standard business object. The application component would have no need to know how the function worked or how the data was structured. The adapter would perform all the interactions with the actual SAP function.

Similarly, the client application might want to know about a change to the data on the SAP server (for example, a change to a particular customer). You can generate an adapter component that listens for such events on the SAP server and notifies client applications with the update. In this case, the interaction begins at the SAP server.

1.3 WebSphere MQ

WebSphere MQ offers various options for managing message exchange between service endpoints. WebSphere MQ has been a long-standing provider of reliable messaging with quality-of-service on interactions ranging from guaranteed, once-and-only-once delivery, to highly efficient best-effort delivery, to low-latency messaging. It also offers quality-of-service enhancements to SOAP-based or Representational State Transfer (REST) interactions, and to interactions that require file-based access to business information. The universal messaging connectivity infrastructure integrates with all of the ESB products that fully use the quality-of-service options offered by WebSphere MQ.

1.4 WebSphere Message Broker

WebSphere Message Broker offers universal any-to-any connectivity that supports a broad spectrum of message formats (from XML, industry standards, existing, byte arrays to comma separated lists), often adding ESB capability to existing messaging-based networks. Existing or packaged applications are made accessible as services in an SOA. WebSphere Message Broker exploits the reach and reliability of the WebSphere MQ messaging backbone, offers unique quality-of-service capabilities on z/OS®, and is optimized for high volume processing.

1.5 WebSphere adapters

The IBM connectivity portfolio includes a rich set of WebSphere adapters that enable interactions with applications and resources that are constructed using various programming models or interaction protocols. Adapters manage normalized access to application endpoint-specific data formats or APIs and a range of quality-of-service options. In addition, adapters provide service enablement for enterprise applications in terms of service virtualization, enabling applications to become connected regardless of their specific interaction protocols.

Various adapters are available, ranging from technology adapters, which support APIs and protocols, such as JDBC and FTP, to application adapters, which provide sophisticated integration for packages such as SAP applications. The adapters also assist the ESBs as plug-ins at the edges of ESB mediation processing, facilitating input from adapted applications to be put on the bus or enabling the bus to deliver requests to an adapter-enabled endpoint.

1.6 WebSphere Adapter for SAP Software

The WebSphere Adapter for SAP software is an application adapter that connects to SAP systems using the SAP Java interface called SAP Java Connector (JCo). It does so by making calls, modeled as business objects, to the SAP native interfaces, and passing data to and from the SAP system. The adapter supports SAP integration interfaces, such as BAPI and ALE, as well as SAP remote function call (RFC) function modules.

The adapter supports outbound processing (from the adapter to the SAP system) and inbound processing (from the SAP system to the adapter) of events.

Figure 1-1 on page 5 shows the high level flow and the components for inbound events from SAP. Inbound calls are supported by the Application Link Enabling (ALE) interface using asynchronous event notification. For the inbound call, the adapter acts as an RFC server and listens for ALE events from SAP applications. The adapter uses an event recovery table to manage the inbound events. With the Advanced Event Processing interface, the adapter processes events related to custom IDocs. The Synchronous callback interface provides a means to synchronously call RFC-enabled functions, such as Business Application Program Interface (BAPI) functions, that are registered with the adapter from the SAP server.

The adapter creates listener threads to the SAP system. The listener threads regularly poll for events. So, in a way, the adapter acts as a remote function call server to the SAP application. Whenever an event occurs in SAP, the event is pushed to the adapter through the event listeners. There is no event store within the SAP application. The adapter uses a persistent event recovery table to track the events. The adapter receives the event from SAP in IDoc table format and converts it to a business object before sending it to the endpoint.

Outbound calls are supported with the BAPI, ALE, and AEP interfaces using Advanced Business Application Programming (ABAP) handlers and Query Interface for SAP Software (QISS). With BAPI, the calls can be simple BAPI calls, BAPI using remote function calls, or multiple BAPI calls in a single interaction, referred to as BAPI unit of work. BAPI outbound calls have request and response interaction style. The ALE interface supports passing single or multiple IDocs. These are one-way calls where the IDocs are passed to the SAP application. With the AEP interface, the adapter makes use of the ABAP handlers, and with the QISS interface you can directly query the SAP application tables.

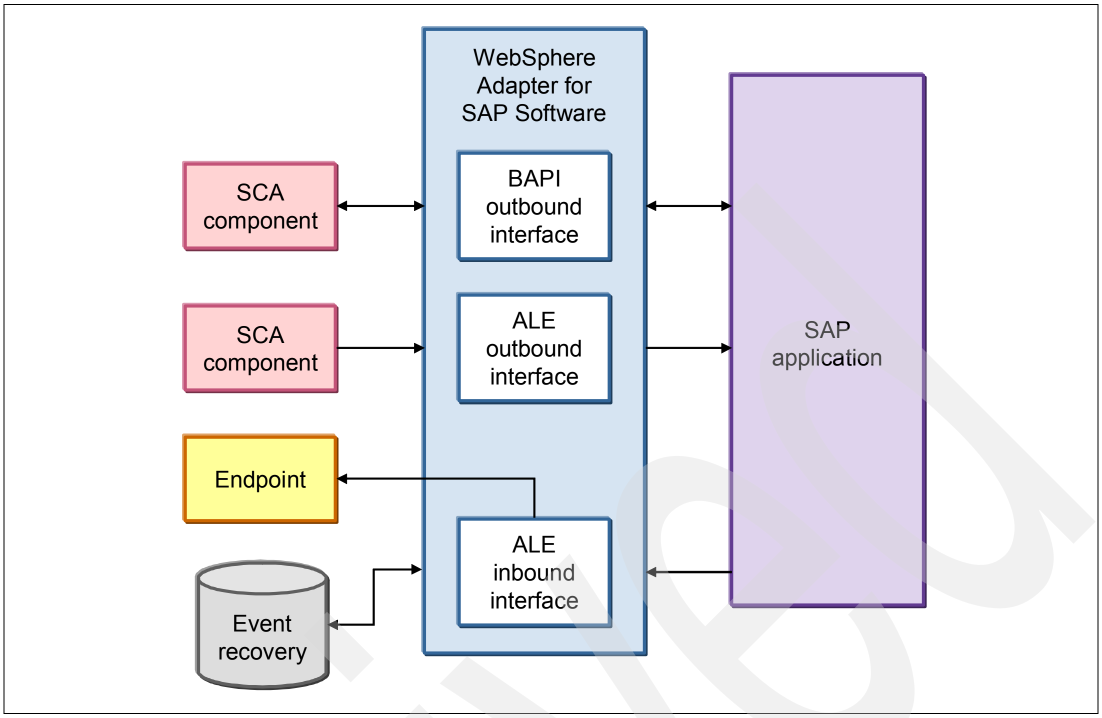


Figure 1-1 High level flow of inbound and outbound calls to SAP

1.6.1 Outbound event processing

With outbound support, a client can make calls to the adapter to perform a specific operation in the SAP system. The client requests a connection, which in turn is passed from the adapter to SAP.

Outbound event processing, which the adapter supports for the BAPI and ALE interfaces, consists of the following actions:

- A Service Component Architecture (SCA) component invokes an interaction with SAP.
- As a result of the invoked interaction, a business object that represents the SAP function call is passed from the component application to the adapter.
- The adapter extracts the elements from the business object, and by using the metadata information from the business object, recognizes the SAP interface to use (BAPI or ALE).
- Using the SAP JCo, the adapter converts the business object data to the appropriate SAP function call.
- The adapter executes the function on the destination SAP software system, sending the event data to SAP.

1.6.2 Inbound event processing

Inbound event processing, which the adapter supports for the ALE interface, consists of the following actions:

- The adapter spawns listener threads, acting as an RFC Server.
- Whenever an event occurs in SAP, the event is pushed to the adapter through the event listeners.

- In the event of an abrupt termination, the adapter can track and recover events by using the data source to persist the event state in an event recovery table.

The adapter supports container-managed sign on and basic authentication because the SAP back-end systems do not provide a re-authentication capability. Since the system does not support this capability the adapter cannot provide this capability either.

Finally, the adapter is able to use all SAP features that are available using the native RFC libraries and ALE processing, like enhanced connection handling, pooling, and expiration.

1.7 Interacting with SAP Software

WebSphere Adapter for SAP Software provides multiple ways to interact with applications and data on SAP servers. For outbound interactions, the adapter can issue RFCs through the Business Application Programming Interface (BAPI). The BAPI interface can work with an individual BAPI, with ordered sets of BAPIs, or with the BAPI result set interface.

The query interface for SAP Software enables retrieval of data from specific SAP application tables and data from an SAP table without using an RFC function or a BAPI. Using the Application Link Enabling (ALE) interface, your application can exchange data using SAP Intermediate Data structures (IDocs). In addition, by using the Advanced event-processing interface, you can send data to the SAP server to be processed by an Advanced Business Application Programming (ABAP) handler.

WebSphere Adapter for SAP Software provides three interfaces for extracting data from SAP and sending it to a service. The adapters and their functions are:

- Synchronous callback interface

  The adapter listens for events and receives notifications of RFC-enabled function calls.
- ALE inbound processing interface

  The adapter listens for events and receives one or more IDocs.
- qRFC interface

  The adapter can receive IDocs from a queue on the SAP server.

1.8 Preparing the SAP environment

To prepare the SAP environment, you need to install the most recent SAP Support Package for your version of SAP. You also need to set up a CPIC user account in the SAP application. This account must have the necessary privileges to manipulate the data that is required by the business objects that are supported by the connector.

1.9 SAP Java Connector

SAP Java Connector (SAP JCo) is a middleware component that enables the development of SAP-compatible components and applications in Java. SAP JCo supports communication with the SAP Server in both directions: inbound calls (Java calls ABAP) and outbound calls (ABAP calls Java).

SAP JCo can be implemented with desktop applications and with Web server applications. SAP JCo is used as an integrated component in the following applications:

- SAP Business Connector, for communication with external Java applications
- SAP Web Application Server, for connecting the integrated J2EE server with the ABAP environment

SAP JCo can also be implemented as a stand-alone component, for example to establish communication with the SAP system for individual online (Web) applications.

1.10 SAP JCo architecture

Figure 1-2 shows the technical schema of data conversion in SAP JCo (stand-alone version). Starting from a Java application, a Java method is forwarded via the JCo Java API and an additional middleware interface to RFC Middleware, where it is converted to an RFC (ABAP) call using the Java Native Interface (JNI) layer, and sent to the SAP system.

The same method is used in the other direction, where an RFC Call is converted to Java and forwarded to the Java application.

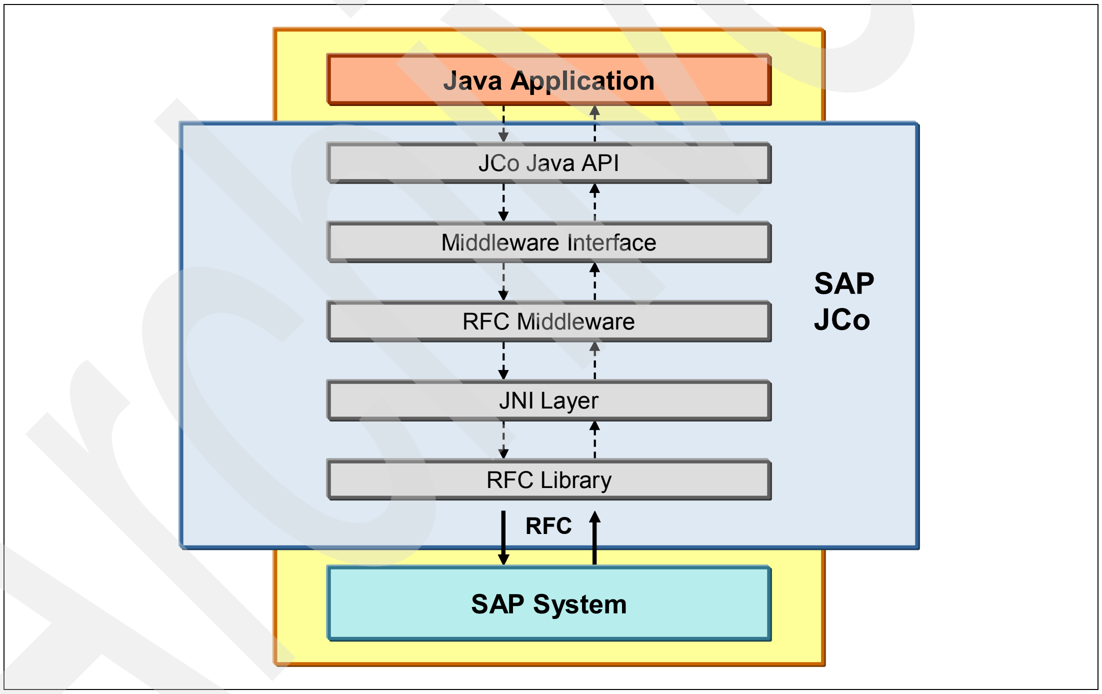


Figure 1-2 Technical schema of data conversion in SAP JCo

1.11 SAP JCo functions

The SAP JCo offers the following functions for creating SAP-capable Java applications:

- RFC middleware based on the JNI
- Support for SAP (R/3) systems of release 3.1H and later, and other mySAP components that have BAPIs or RFMs (Remote Function Modules)

- Execution of function calls inbound (Java client calls BAPI or RFM) and outbound (ABAP calls Java Server)
- Ability to use synchronous, transactional, and queued RFC
- Connection pooling (important for Web servers)
- Can be used on various platforms
- Extensive code page handling, including multi-byte languages

1.12 SAP JCo installation

This section contains the guidelines for installation of SAP JCo 3.0., which is a Java jar and a collection of DLLs that come with the SAP system.

Before you begin this procedure:

1. Download the SAP Java Connector from the SAP Service Marketplace Web site:

   `http://service.sap.com/connectors`

   > **Note:** You must have an SAP Service Marketplace user ID and password to access the download.

2. Review the installation information on the SAP Web site.

Because the components contain both packages and system-own class libraries, the system-own class libraries are platform specific.

These instructions apply to Windows32 or Linux® operating systems. The instructions for the installation of SAP JCo on other operating systems are included in the corresponding download files.

1. Create a directory, for example C:\SAP_JCO_3\, and extract the JCo.zip file into this directory.
2. Make sure that the file sapjco3.jar (in the SAP JCo main directory) is contained in the class path for all projects for which you want to use the SAP JCo.

For productive operation, the following files from the SAP JCo.zip file are necessary:

Microsoft® Windows®:	sapjco3.jar sapjco3.dll
Linux:	sapjco3.jar libsapjco3.so

Important: Note the additional information on the download page in SAPNet.

Prerequisites and Installation:

In order to connect to the SAP system the SAP Java Connector (JCo) is required. This component is not redistributed with the adapter because of license agreements and should be obtained separately from SAP Marketplace. Installation instructions for JCo are available on the SAP Web site.

`http://service.sap.com/connectors`

SAP highly recommends that you store these files in the same directory.

2

The exercise: Developing message flows

This chapter demonstrates the development of message flows for interacting with SAP Systems. It guides you through the steps to build a message flow to receive an IDoc from the SAP system. An “IDoc” is an intermediate document, a data exchange format used between SAP systems, and between SAP systems and external applications.

2.1 SAP Inbound sample scenario

When you are using WebSphere Message Broker inbound messaging, the SAP system must be configured so that it can uniquely identify the WebSphere Message Broker as a target destination. SAP creates several components (RFC Destinations, Ports, Logical Systems, Distribution Models, and Partner Profiles) in order to configure this connection. IDocs are used for asynchronous batch data transmission.

The ALE Inbound sample scenario demonstrates how the WebSphere Adapter for SAP Software can be used to receive events from the SAP Enterprise Information System (EIS) by taking a sample IDoc into consideration. The sample scenario shows how to configure the adapter; then it shows how to configure an endpoint in order to receive a SAP invoked IDoc asynchronous event.

You must install these products before you can deploy the adapter:

- IBM WebSphere Message Broker 7.0
- IBM WebSphere Message Broker Toolkit 7.0

After these products are installed, make sure you know the following information for accessing the SAP application:

- SAP User Name
- SAP Password
- SAP Host name (or IP address)
- SAP System number (usually 00)
- SAP Client number (usually 100)
- SAP JCO library

Follow these steps to get started:

1. Start the WebSphere Message Broker by selecting **Start** → **Programs** → **IBM WebSphere Message Broker Toolkit** → **WebSphere Message Broker Toolkit 7.0** (Figure 2-1).

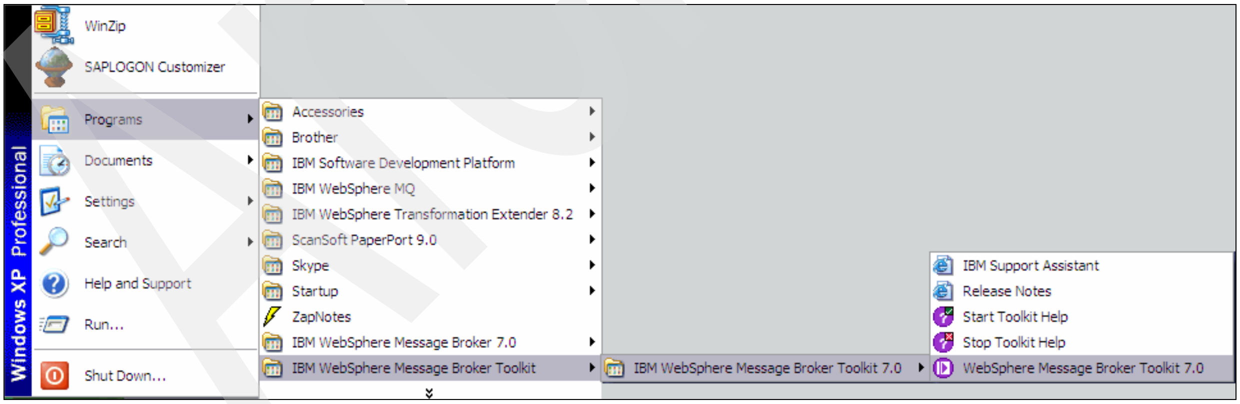
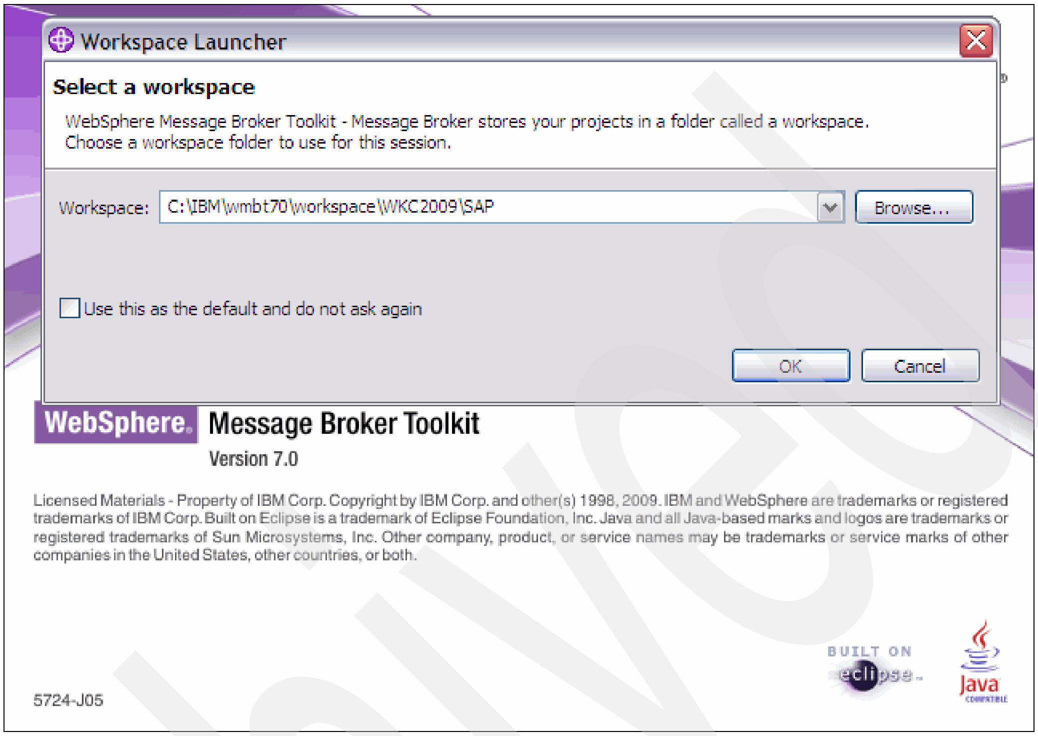


Figure 2-1 Starting WebSphere Message Broker Toolkit 7.0

2. Once the WebSphere Message Broker Toolkit 7.0 is launched (Figure 2-2) select the location where you want to build the demo; in our example this was C:\IBM\wmbt70\worspace\WKC2009\SAP. Click **OK**.



Select a workspace

WebSphere Message Broker Toolkit - Message Broker stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: C:\IBM\wmbt70\workspace\WKC2009\SAP

□ Use this as the default and do not ask again

Licensed Materials - Property of IBM Corp. Copyright by IBM Corp. and other(s) 1998, 2009. IBM and WebSphere are trademarks or registered trademarks of IBM Corp. Built on Eclipse is a trademark of Eclipse Foundation, Inc. Java and all Java-based marks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. Other company, product, or service names may be trademarks or service marks of other companies in the United States, other countries, or both.

Figure 2-2 Launching WebSphere Message Broker Toolkit

3. If this is the first time the WebSphere Message Broker is opened, the Welcome screen is displayed. Click **X** to close it (Figure 2-3).

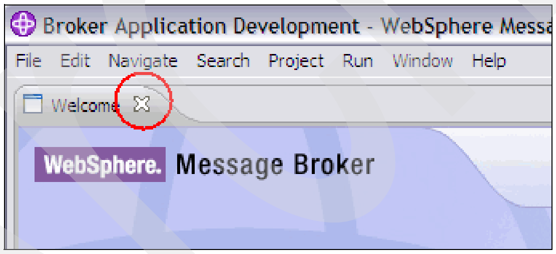
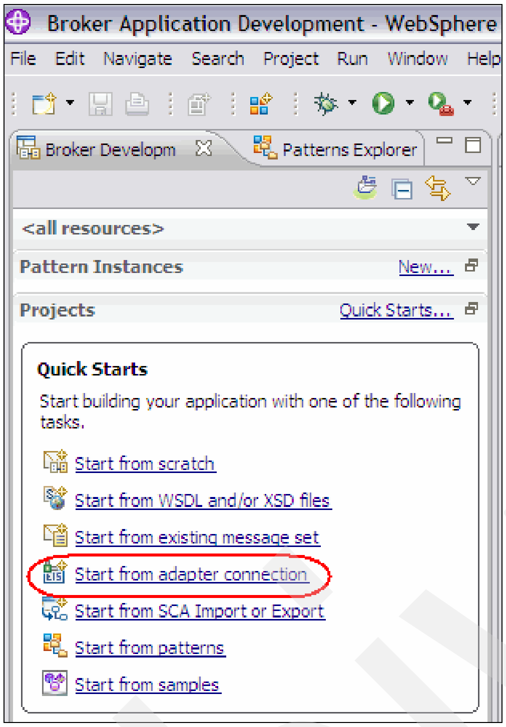

Broker Application Development - WebSphere Messa
File Edit Navigate Search Project Run Window Help
Welcome
WebSphere. Message Broker

Figure 2-3 Welcome window

4. Select the Broker Development tab located on the left side of the screen and click **Start from adapter connection** to start the adapter connection (Figure 2-4).

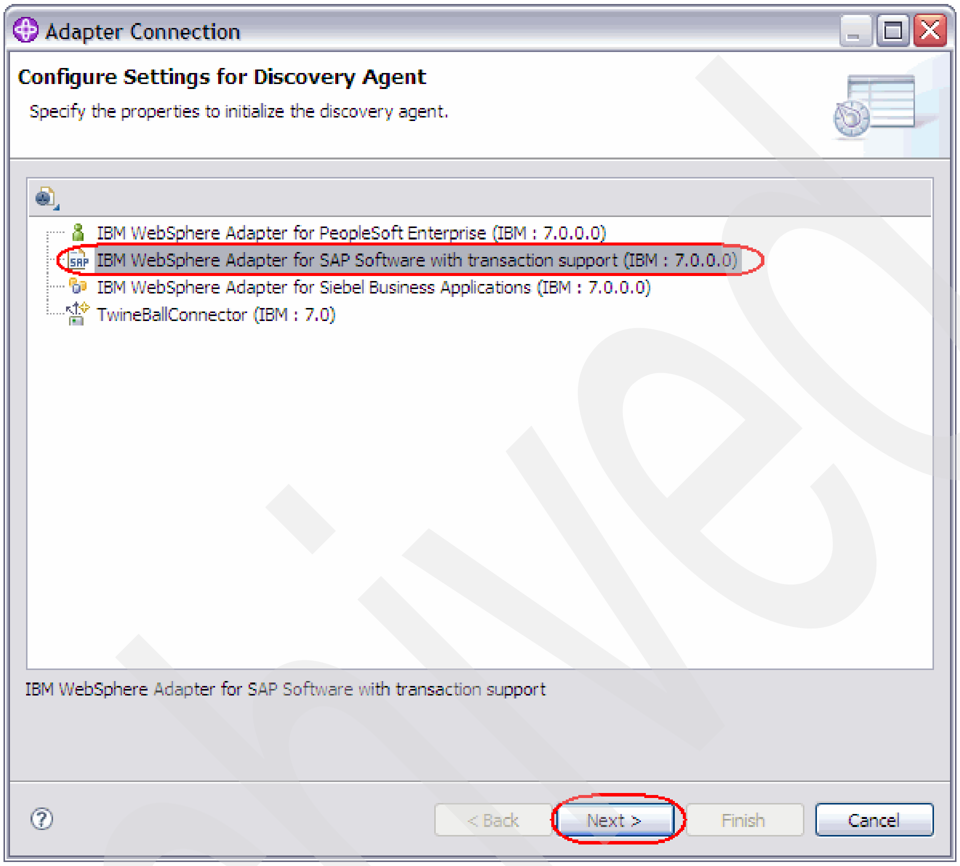

Broker Application Development - WebSphere
File Edit Navigate Search Project Run Window Help
Broker Developm
Patterns Explorer
<all resources>
Pattern Instances
New...
Projects
Quick Starts...
Quick Starts

Start building your application with one of the following tasks.

Start from scratch
Start from WSDL and/or XSD files
Start from existing message set
Start from adapter connection
Start from SCA Import or Export
Start from patterns
Start from samples

Figure 2-4 Start from adapter connection

5. The first panel in the wizard prompts you to select the Adapter Connection you want to use to discover a service. Select the resource adapter **IBM WebSphere Adapter for SAP Software with transaction support (IBM: 7.0.0.0)** and click **Next** to configure it from scratch (Figure 2-5).



Adapter Connection

Configure Settings for Discovery Agent

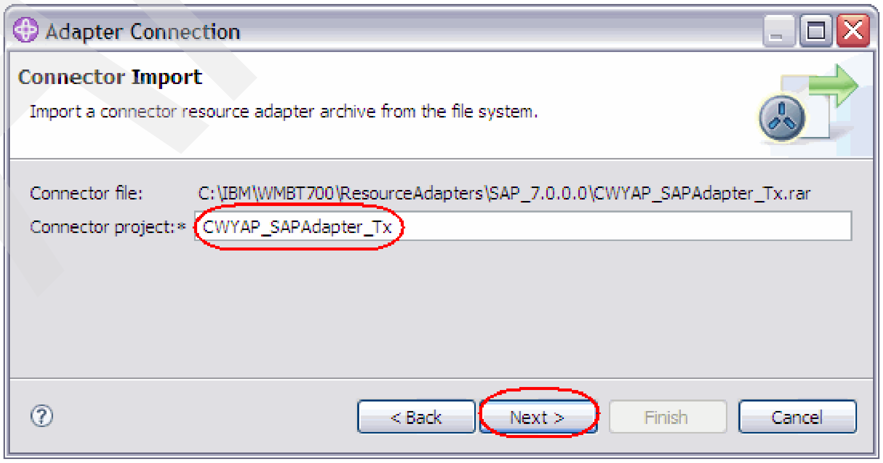
Specify the properties to initialize the discovery agent.

- IBM WebSphere Adapter for PeopleSoft Enterprise (IBM : 7.0.0.0)
- IBM WebSphere Adapter for SAP Software with transaction support (IBM : 7.0.0.0)
- IBM WebSphere Adapter for Siebel Business Applications (IBM : 7.0.0.0)
- TwineBallConnector (IBM : 7.0)

IBM WebSphere Adapter for SAP Software with transaction support

Figure 2-5 Configure settings for Discovery Agent

6. Specify the Connector project you want to use and click **Next**. The SAP adapter requires a connector resource adapter project, which contains references to the SAP-JCo that must be configured in each workspace that uses SAP Adapter projects. Provide the name for the connector resource project. The default name CWYAP_SAPAdapter_Tx is provided (Figure 2-6).



Adapter Connection

Connector Import

Import a connector resource adapter archive from the file system.

Connector file: C:\IBM\WMBT700\ResourceAdapters\SAP_7.0.0.0\CWYAP_SAPAdapter_Tx.rar

Connector project:* CWYAP_SAPAdapter_Tx

Figure 2-6 Specify connector

7. Set the location for the JCo on the local machine. This is required by the WebSphere Message Broker Toolkit for discovering RFCs and for using the adapter in the toolkit.

   In our example the files are located in directory C:\SAP_JCO_3\.

   To specify the directory in which SAP JCo is installed (Figure 2-7):

   a. Type the SAP JCo file path for your system into the appropriate field, or click **Browse** to select the file location.

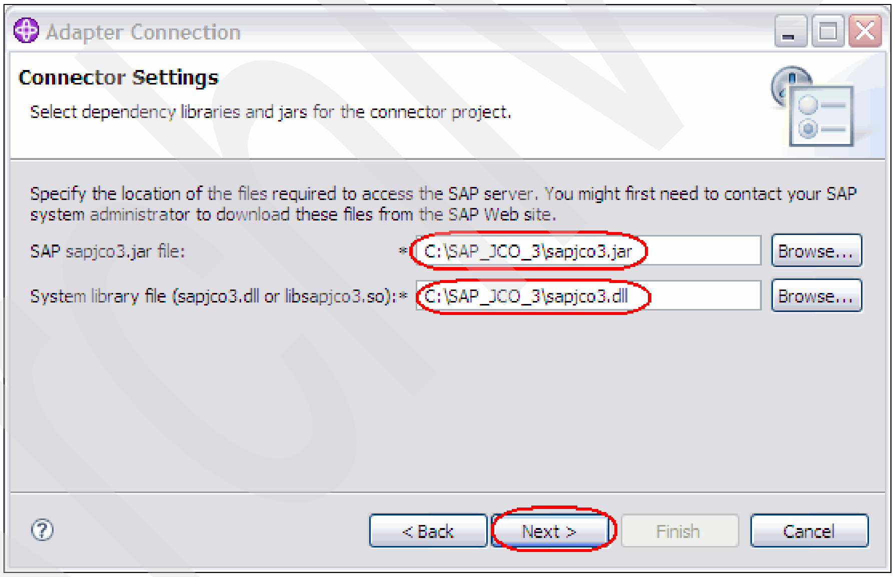
   b. The Installer checks the version of the SAP JCo file and continues if version 3.0 is required. The installer displays a warning if the required SAP JCo version is not installed and stops the installation. Restart the installation. Refer to the SAP documentation for information about the SAP JCo file installation.

   c. Make the following entries:

      - Connector Settings dialog: SAP sapjco3.jar file: C:\SAP_JCO_3\sapjco3.jar
      - System library file (sapjco3.dll or libsapjco3.so): C:\SAP_JCO_3\sapjco3.dlll

   d. Click **Next**.

> **Important:** The files sapjco3.jar and sapjco3.dll or libsapjco3.so are specific to the SAP system being used. You will need to obtain the dependency files for the SAP system you access.



Connector Settings

Select dependency libraries and jars for the connector project.

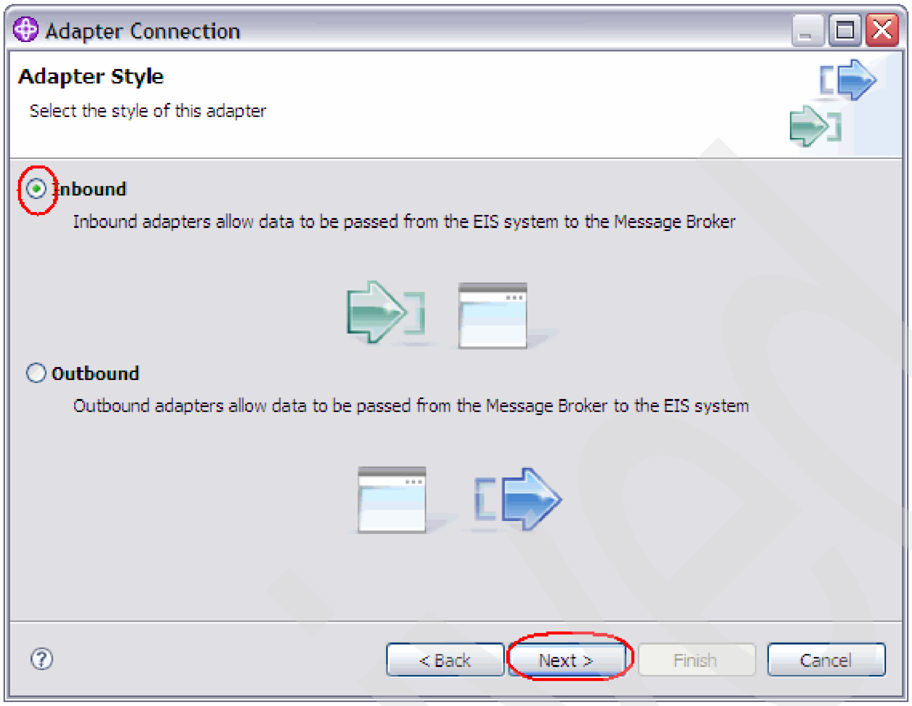
Specify the location of the files required to access the SAP server. You might first need to contact your SAP system administrator to download these files from the SAP Web site.

SAP sapjco3.jar file: * C:\SAP_JCO_3\sapjco3.jar

System library file (sapjco3.dll or libsapjco3.so): * C:\SAP_JCO_3\sapjco3.dll

Figure 2-7 Select dependency libraries and jars for the connector project

8. In the Adapter Style dialog (Figure 2-8), select **Inbound**. Click **Next**.


Adapter Connection

Adapter Style

Select the style of this adapter

● Inbound

Inbound adapters allow data to be passed from the EIS system to the Message Broker

○ Outbound

Outbound adapters allow data to be passed from the Message Broker to the EIS system

Figure 2-8 Establish Adapter Style

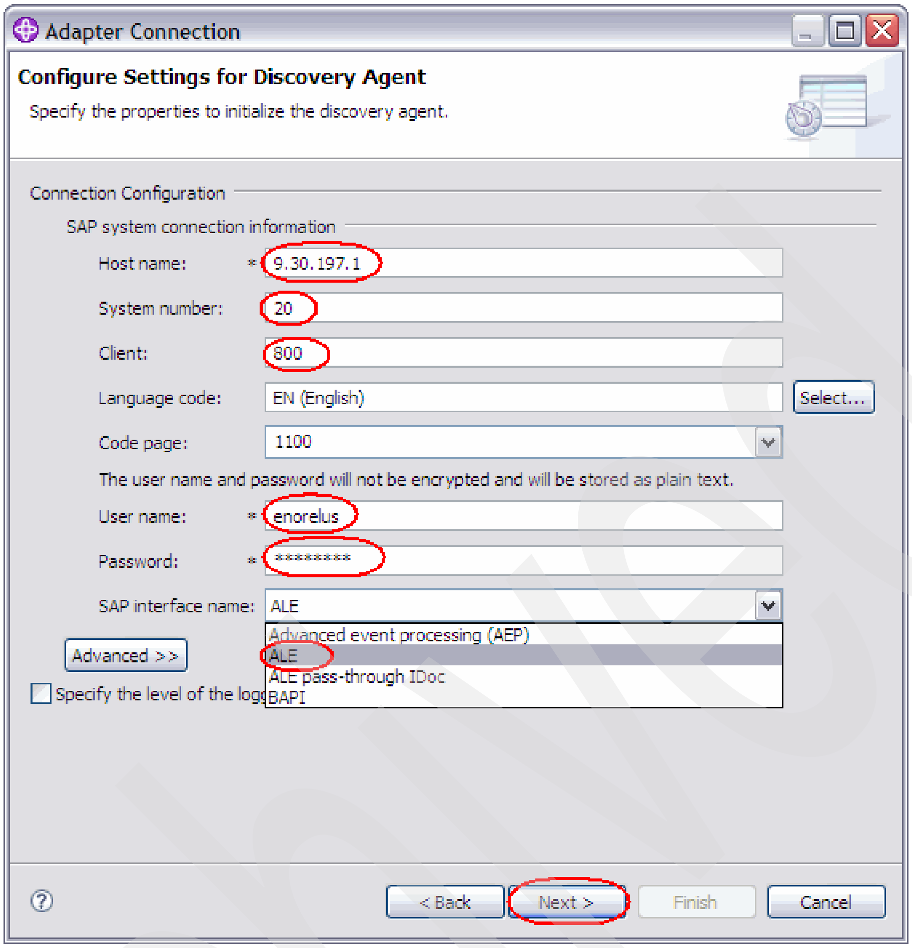
2.2 Configure the SAP source server

To connect to the SAP System you must specify the following information: username, password, Hostname, the system number, and which client of the SAP System you would like to connect to. Language code is set by default to English; it can be changed using the select button. Codepage is by default set to 1100; other values can be selected from the drop-down list. The SAP interface name to be selected is `ALE`.

1. Set the parameters for the data server by making the appropriate entries for your environment (Figure 2-9):

   – Host name: <*your system's hostname*>
   – System number: <*your system's number*>
   – Client: <*your system's client*>
   – User name: <*your user name*>
   – Password: <*your password*>
   – SAP Interface name: `ALE`

   Click **Next**.



Adapter Connection

Configure Settings for Discovery Agent

Specify the properties to initialize the discovery agent.

Connection Configuration

SAP system connection information

Host name: * 9.30.197.1

System number: 20

Client: 800

Language code: EN (English)

Code page: 1100

The user name and password will not be encrypted and will be stored as plain text.

User name: * enorelus

Password: * *********

SAP interface name: ALE

Advanced event processing (AEP)
ALE
ALE pass-through IDoc
BAPI

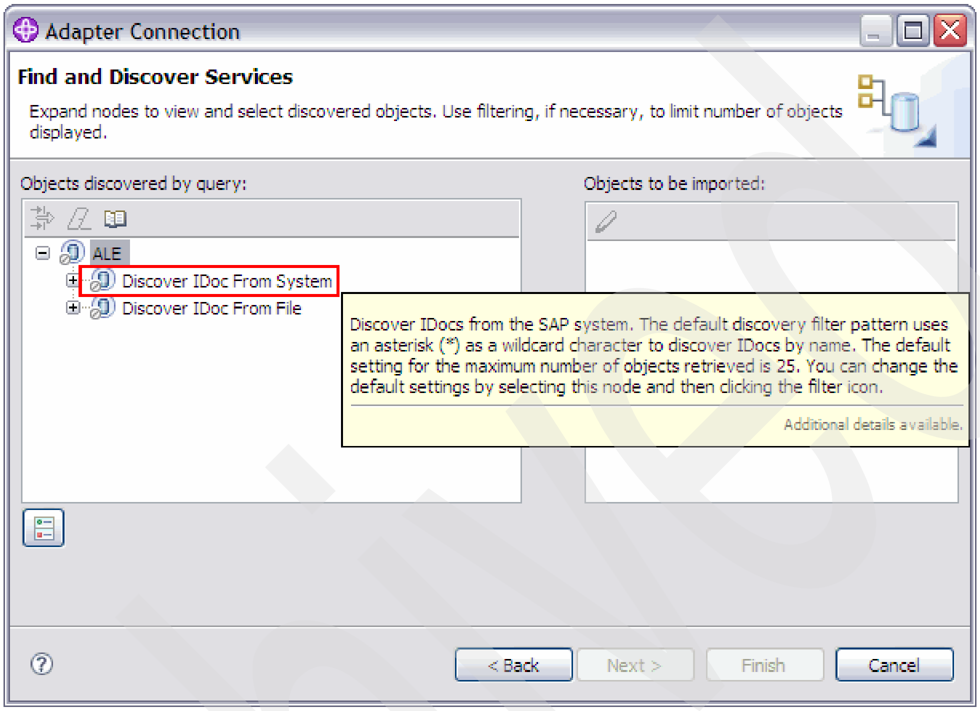
□ Specify the level of the log

Figure 2-9 Configure settings for Discovery Agent

2. Two categories of objects are available to be discovered under the ALE in the Find and Discover Services screen (Figure 2-10).

   a. Click **[+]** before the ALE node in Discover IDoc From System to expand this node.

   b. Click **Discover IDOC From System.**



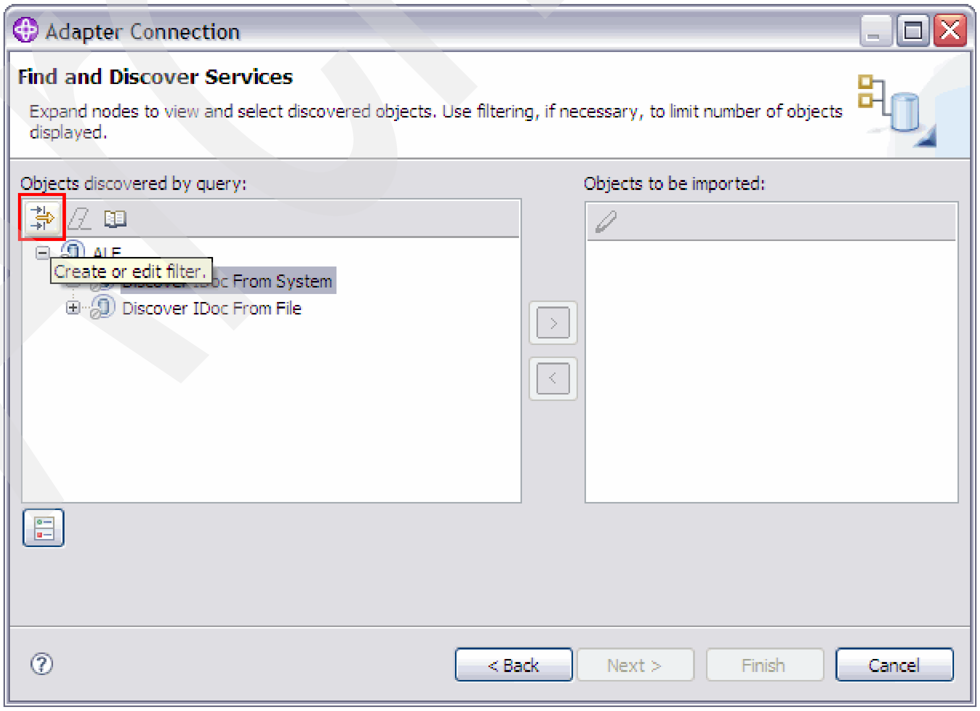
Find and Discover Services

Expand nodes to view and select discovered objects. Use filtering, if necessary, to limit number of objects displayed.

Discover IDocs from the SAP system. The default discovery filter pattern uses an asterisk (*) as a wildcard character to discover IDocs by name. The default setting for the maximum number of objects retrieved is 25. You can change the default settings by selecting this node and then clicking the filter icon.

Figure 2-10 Find and discover services

3. Click the **Create or edit filter** button (Figure 2-11).

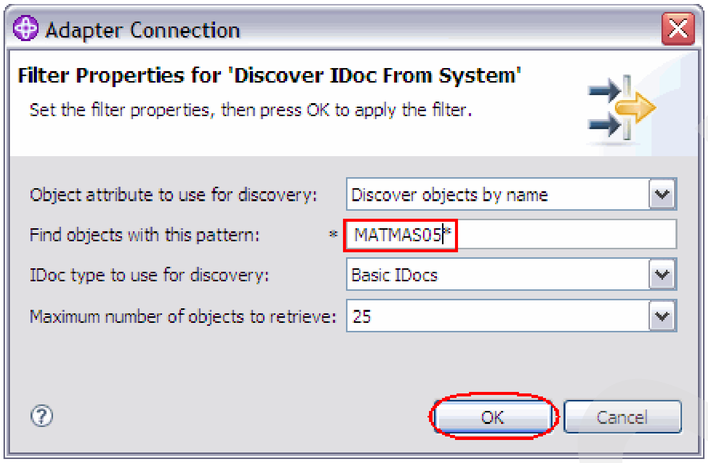


Find and Discover Services

Expand nodes to view and select discovered objects. Use filtering, if necessary, to limit number of objects displayed.

Figure 2-11 Select discovered objects

4. Enter `MATMAS05*` (the name of the ALE in the SAP system) in the Filter Properties for 'Discover IDoc From System' window. Click **OK** (Figure 2-12).



Adapter Connection

Filter Properties for 'Discover IDoc From System'

Set the filter properties, then press OK to apply the filter.

Object attribute to use for discovery: Discover objects by name

Find objects with this pattern: * MATMAS05*

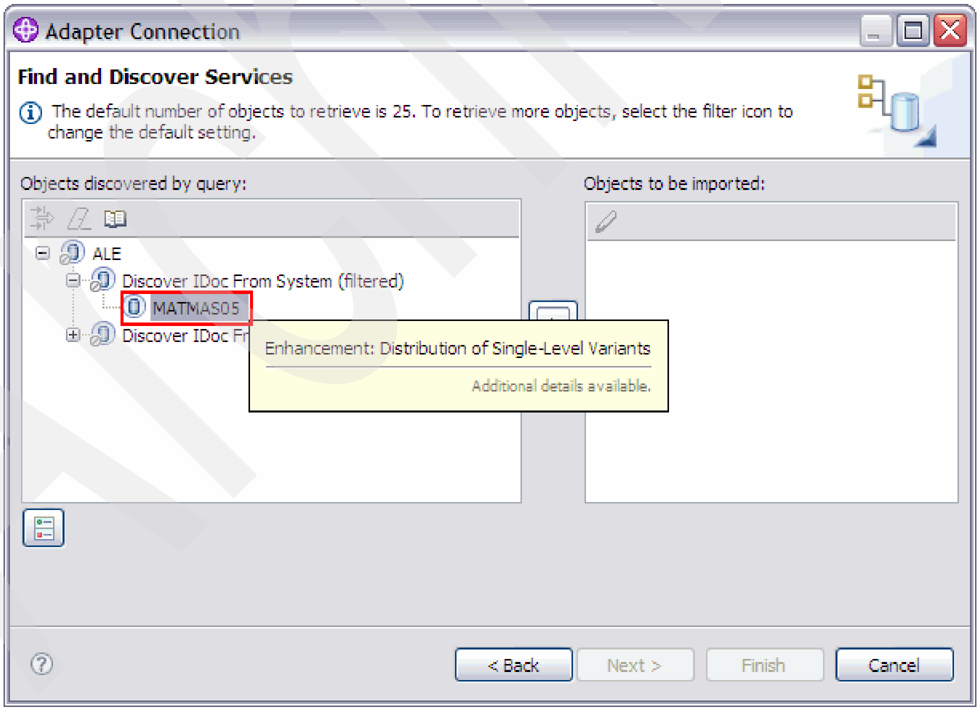
IDoc type to use for discovery: Basic IDocs

Maximum number of objects to retrieve: 25

Figure 2-12 Establish filter properties

5. Expand the Discover IDoc From System (filtered) node (Figure 2-13).

> **Note:** The query will now execute and may take a while to complete. You can expand the Discover IDoc From System element, but the object list will not be displayed until the query has been completed.



Adapter Connection

Find and Discover Services

The default number of objects to retrieve is 25. To retrieve more objects, select the filter icon to change the default setting.

Objects discovered by query:

- ALE
  - Discover IDoc From System (filtered)
    - MATMAS05
  - Discover IDoc Fr

Enhancement: Distribution of Single-Level Variants

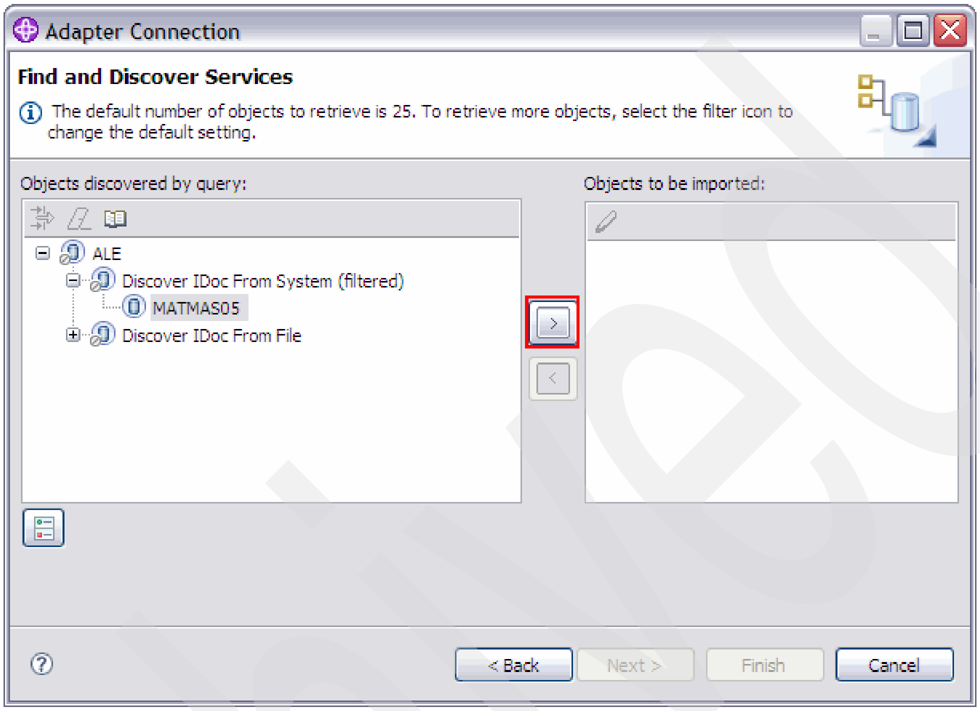
Additional details available.

Objects to be imported:

Figure 2-13 Expand Discover IDoc From System

6. Highlight **MATMAS05** in the left pane and click the right arrow button [>] (Figure 2-14).

Note: This operation will take a while to complete. Be patient.



Adapter Connection

Find and Discover Services

The default number of objects to retrieve is 25. To retrieve more objects, select the filter icon to change the default setting.

Objects discovered by query:

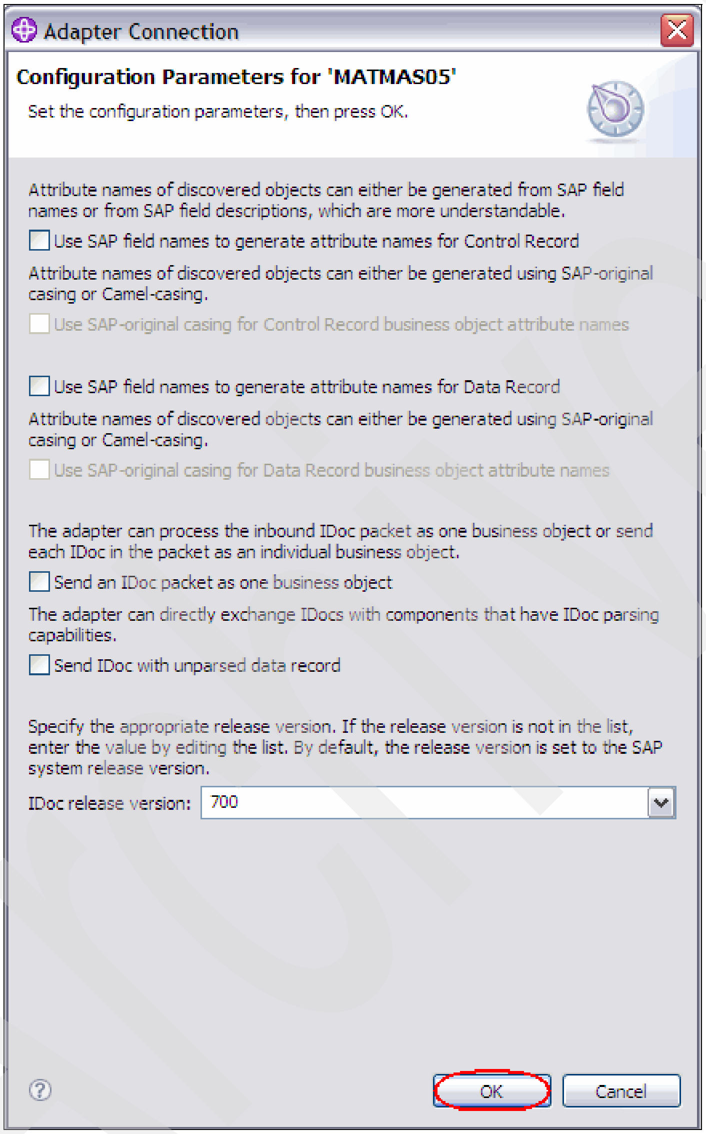
- ALE
  - Discover IDoc From System (filtered)
    - MATMAS05
  - Discover IDoc From File

Objects to be imported:

Figure 2-14 Determine objects to be imported

7. Accept the defaults on the Configuration Parameters for ‘MATMAS05 dialog’; click **OK** (Figure 2-15).

> **Note:** This window contains a series of options to play with the SAP System; in our case we leave all these options untouched, this is left for a possible discussion later.



Configuration Parameters for 'MATMAS05'

Set the configuration parameters, then press OK.

Attribute names of discovered objects can either be generated from SAP field names or from SAP field descriptions, which are more understandable.

□ Use SAP field names to generate attribute names for Control Record

Attribute names of discovered objects can either be generated using SAP-original casing or Camel-casing.

□ Use SAP-original casing for Control Record business object attribute names

□ Use SAP field names to generate attribute names for Data Record

Attribute names of discovered objects can either be generated using SAP-original casing or Camel-casing.

□ Use SAP-original casing for Data Record business object attribute names

The adapter can process the inbound IDoc packet as one business object or send each IDoc in the packet as an individual business object.

□ Send an IDoc packet as one business object

The adapter can directly exchange IDocs with components that have IDoc parsing capabilities.

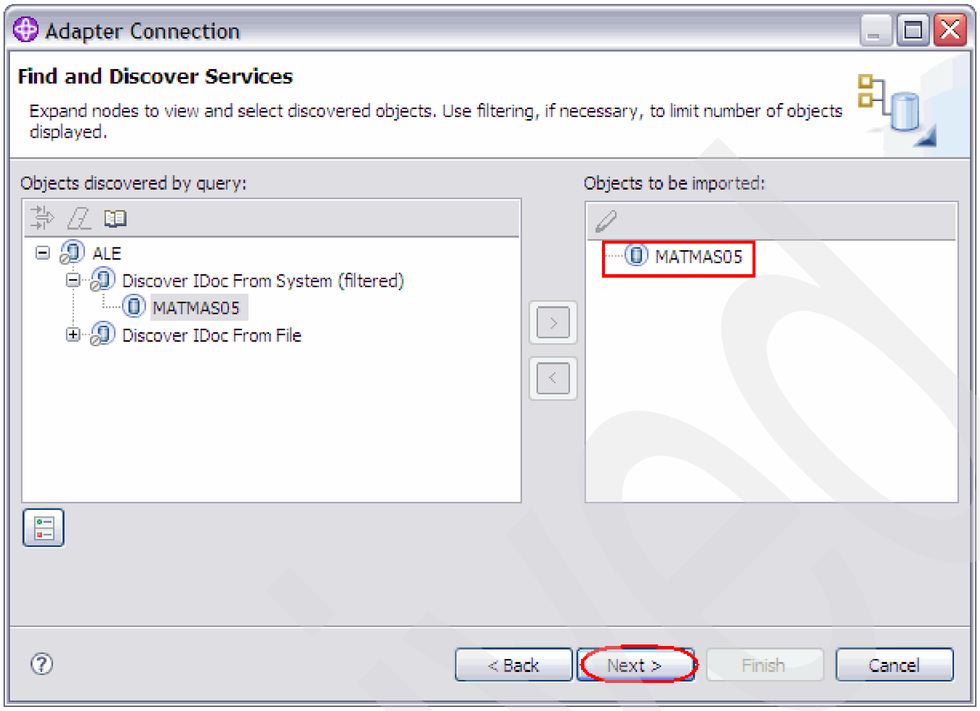
□ Send IDoc with unparsed data record

Specify the appropriate release version. If the release version is not in the list, enter the value by editing the list. By default, the release version is set to the SAP system release version.

IDoc release version: 700

Figure 2-15 Configure parameters

8. Select the **MATMAS05** from the Object to be imported. Click **Next** (Figure 2-16).



Adapter Connection

Find and Discover Services

Expand nodes to view and select discovered objects. Use filtering, if necessary, to limit number of objects displayed.

Objects discovered by query:

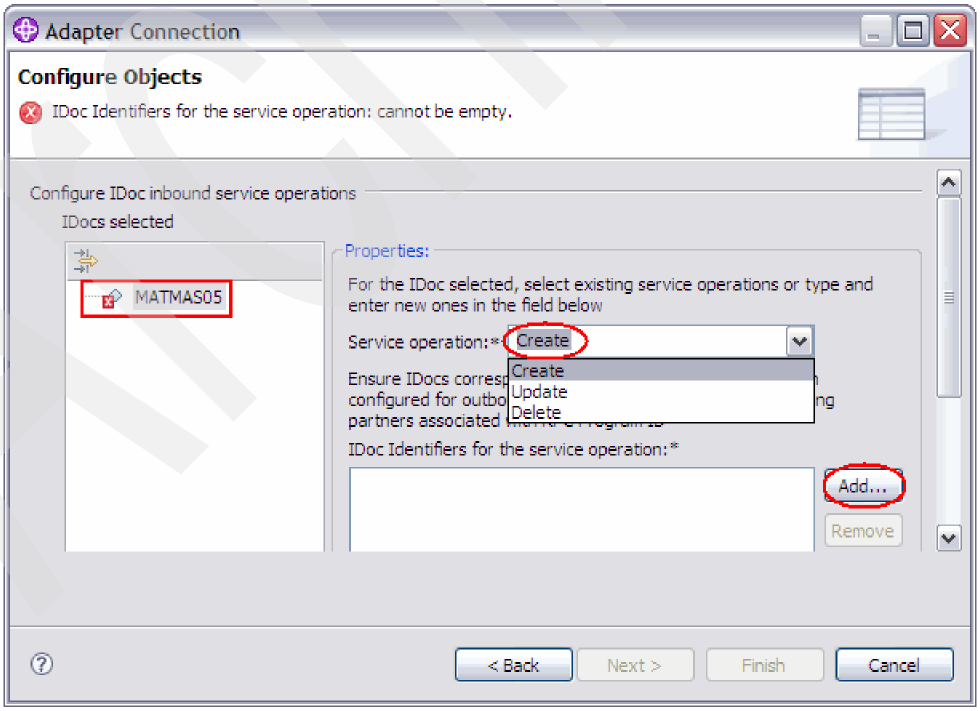
- ALE
  - Discover IDoc From System (filtered)
    - MATMAS05
  - Discover IDoc From File

Objects to be imported:

- MATMAS05

Figure 2-16   Expand nodes to view and select discovered objects

9. In the Configure Objects window, select **MATMAS05** from the IDocs selected section; in the properties field select **Create**, then click **Add** (Figure 2-17).



Adapter Connection

Configure Objects

IDoc Identifiers for the service operation: cannot be empty.

Configure IDoc inbound service operations

IDocs selected

MATMAS05

Properties:

For the IDoc selected, select existing service operations or type and enter new ones in the field below

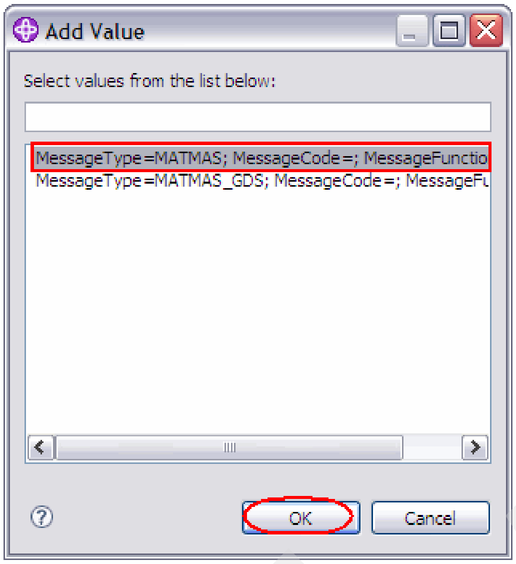
Service operation:* Create

Ensure IDocs corresp
configured for outbo
partners associated

IDoc Identifiers for the service operation:*

Figure 2-17   Create IDoc identifiers

10. When the Add Value window opens, select **MessageType=MATMAS; MessageCode= ...**. Click **OK** (Figure 2-18).

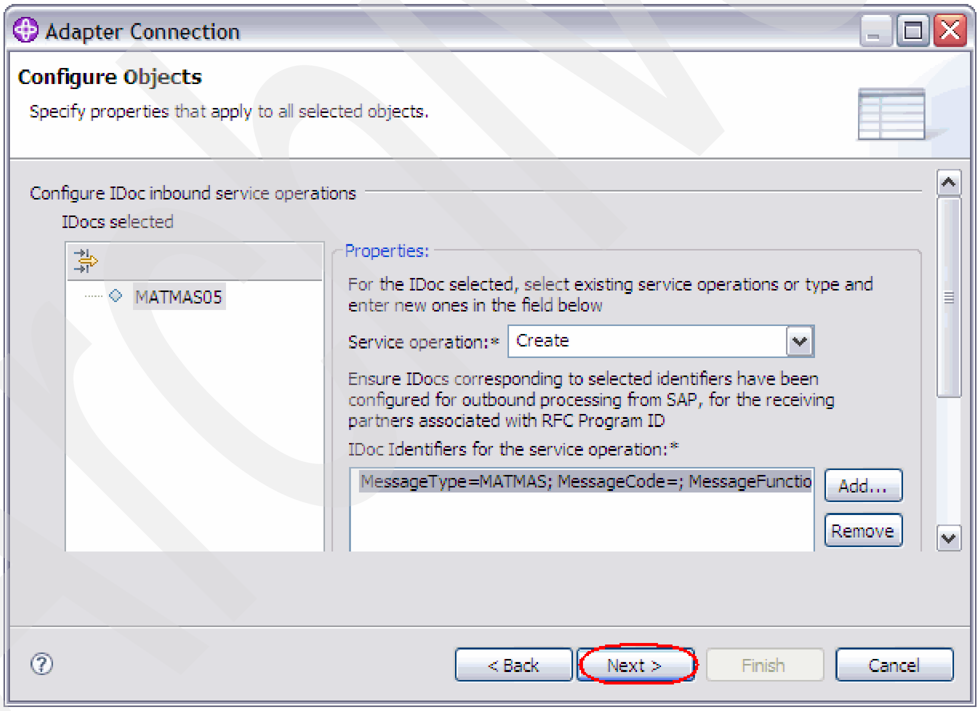


Select values from the list below:

MessageType=MATMAS; MessageCode=; MessageFunctio
MessageType=MATMAS_GDS; MessageCode=; MessageFu

Figure 2-18 Add values

11. This brings you back to the Configure Objects windows. Click **Next** (Figure 2-19).



Configure Objects

Specify properties that apply to all selected objects.

Configure IDoc inbound service operations

IDocs selected

MATMAS05

Properties:

For the IDoc selected, select existing service operations or type and enter new ones in the field below

Service operation:* Create

Ensure IDocs corresponding to selected identifiers have been configured for outbound processing from SAP, for the receiving partners associated with RFC Program ID

IDoc Identifiers for the service operation:*

MessageType=MATMAS; MessageCode=; MessageFunctio

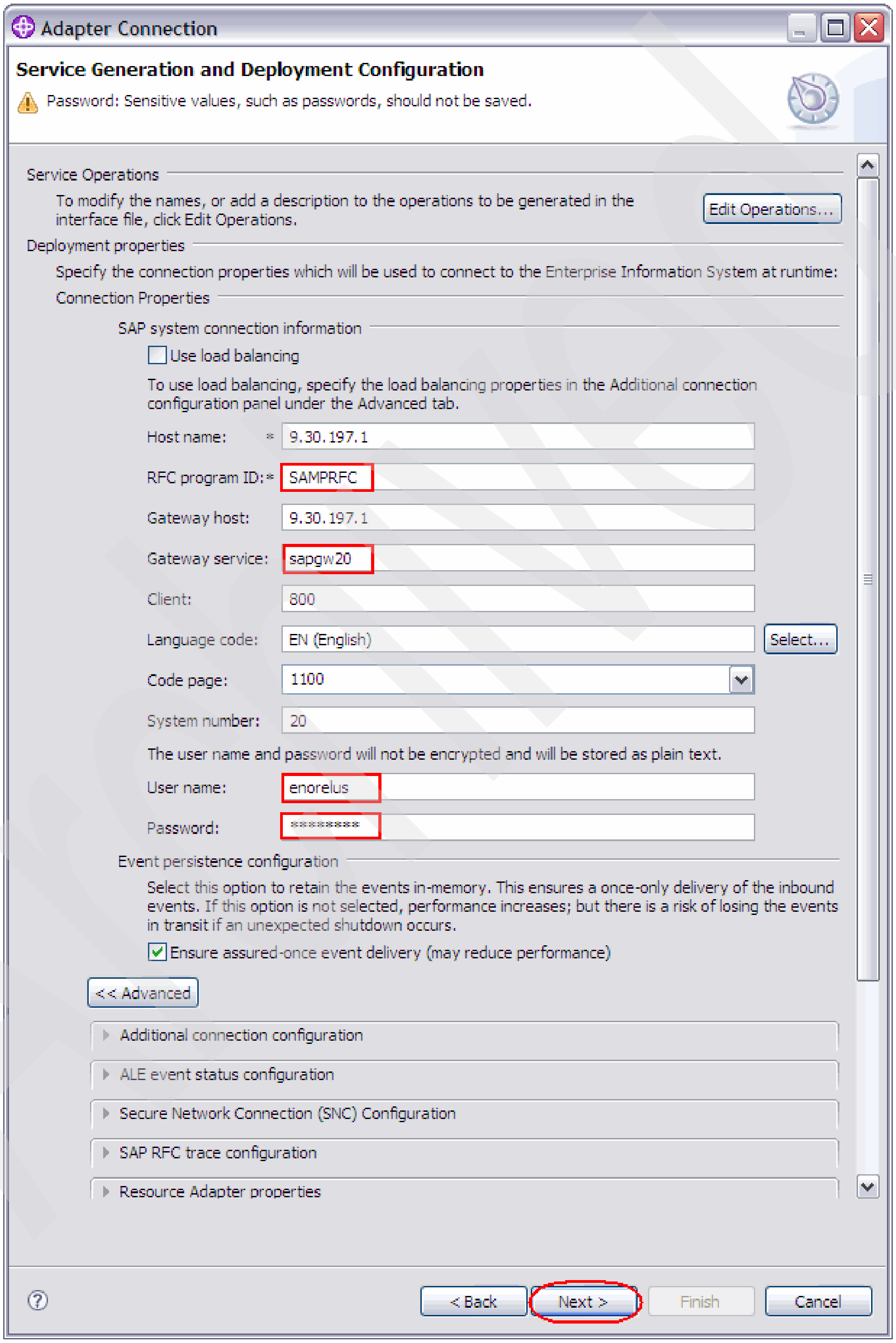
Figure 2-19 Specific properties for MATMAS05

12. On the Service Generation and Deployment Configuration screen (Figure 2-20) enter the connection information:

   – RFC program ID: `SAMPFRC`
   – Gateway service: <`sapgwXX`>
   – User name: <*your user name*>
   – Password: <*your password*>

   Click **Next.**

Requirement: This is where you set the Inbound Connection Properties with the SAP system values and the RFC program that was created for this purpose (transaction SM59). This part will be discussed later in this document



Adapter Connection

Service Generation and Deployment Configuration

⚠ Password: Sensitive values, such as passwords, should not be saved.

Service Operations

To modify the names, or add a description to the operations to be generated in the interface file, click Edit Operations.

Deployment properties

Specify the connection properties which will be used to connect to the Enterprise Information System at runtime:

Connection Properties

SAP system connection information

□ Use load balancing

To use load balancing, specify the load balancing properties in the Additional connection configuration panel under the Advanced tab.

Host name: * 9.30.197.1

RFC program ID:* SAMPRFC

Gateway host: 9.30.197.1

Gateway service: sapgw20

Client: 800

Language code: EN (English)

Code page: 1100

System number: 20

The user name and password will not be encrypted and will be stored as plain text.

User name: enorelus

Password: *********

Event persistence configuration

Select this option to retain the events in-memory. This ensures a once-only delivery of the inbound events. If this option is not selected, performance increases; but there is a risk of losing the events in transit if an unexpected shutdown occurs.

☑ Ensure assured-once event delivery (may reduce performance)

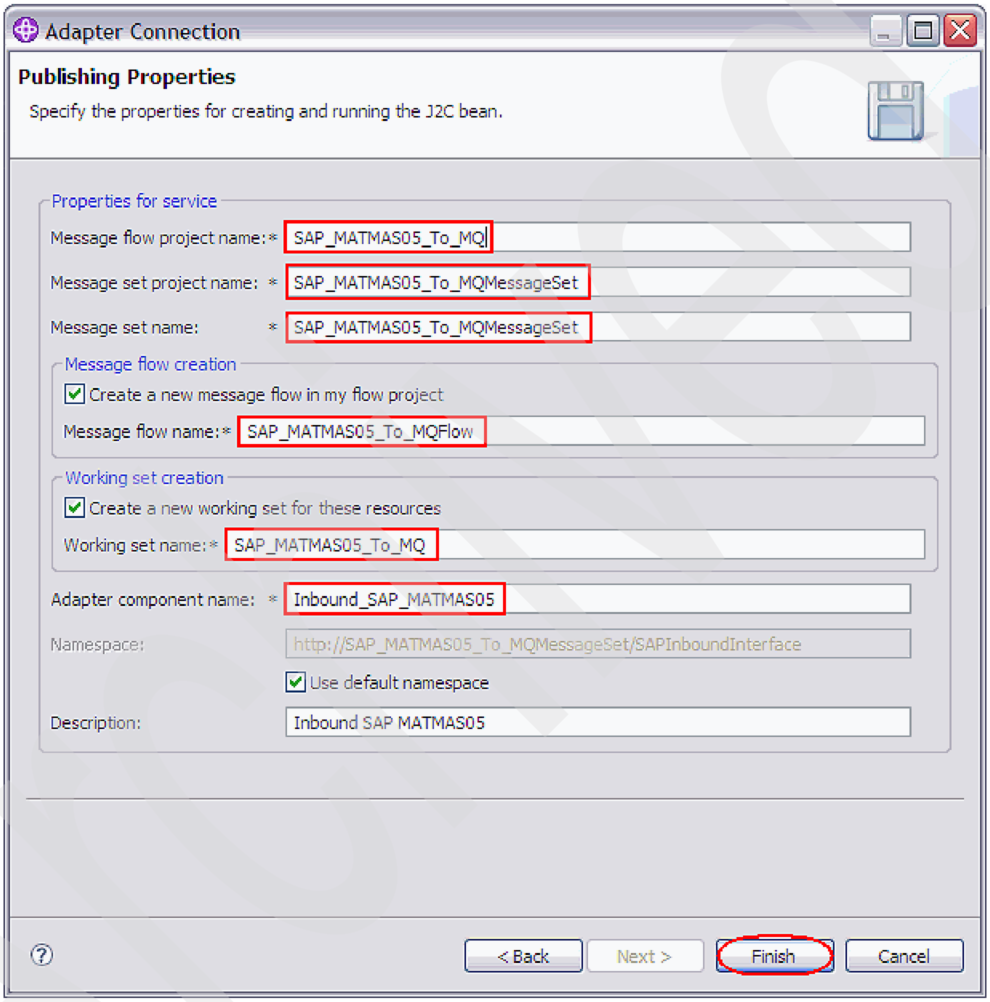
<< Advanced
Additional connection configuration
ALE event status configuration
Secure Network Connection (SNC) Configuration
SAP RFC trace configuration
Resource Adapter properties
Select...
< Back
Next >
Finish
Cancel

Figure 2-20 Set connection information

13. Make the following entries In the Publishing Properties window (Figure 2-21):

– Message flow project name: `SAP_MATMAS05_To_MQ`
– Message set project name: `SAP_MATMAS05_To_MQMessageSet`
– Message set name: `SAP_MATMAS05_To_MQMessageSet`
– Message flow name: `SAP_MATMAS05_To_MQFlow`
– Working set name: `SAP_MATMAS05_To_MQ`
– Adapter component name: `Inbound_SAP_MATMAS05`

Click **Finish**.



Adapter Connection

Publishing Properties

Specify the properties for creating and running the J2C bean.

Properties for service

Message flow project name: * SAP_MATMAS05_To_MQ

Message set project name: * SAP_MATMAS05_To_MQMessageSet

Message set name: * SAP_MATMAS05_To_MQMessageSet

Message flow creation

☑ Create a new message flow in my flow project

Message flow name: * SAP_MATMAS05_To_MQFlow

Working set creation

☑ Create a new working set for these resources

Working set name: * SAP_MATMAS05_To_MQ

Adapter component name: * Inbound_SAP_MATMAS05

Namespace: http://SAP_MATMAS05_To_MQMessageSet/SAPInboundInterface

☑ Use default namespace

Description: Inbound SAP MATMAS05

Figure 2-21 Specify properties for creating and running J2C bean

14. Examine the registry to verify that the following projects have been created (Figure 2-22):

– CWYAP_SAPAdapter_Tx
– SAP_MATMAS05_To_MQ
– SAP_MATMAS05_To_MQMessageSet

Expand [**+**] CWYAP_SAPAdapter_Tx to examine the newly created artifacts.

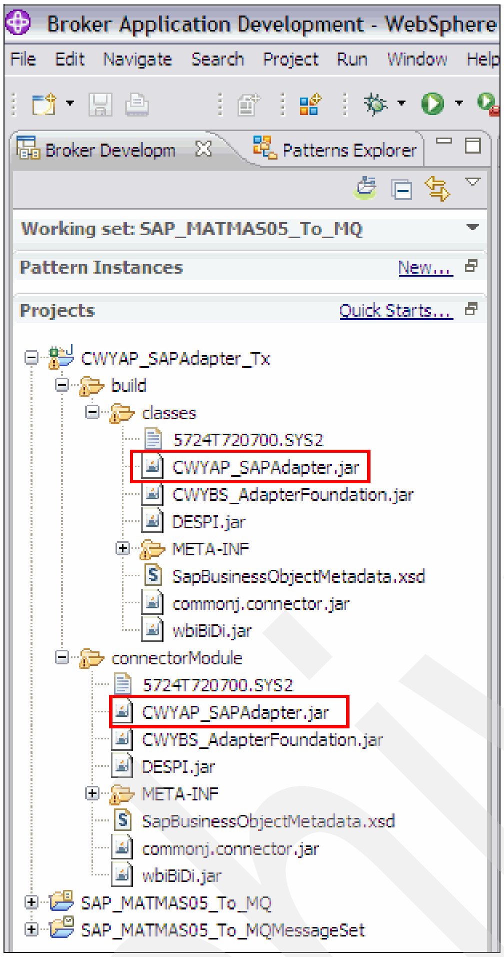


Figure 2-22 Verify what has been created

15. Expand [+] SAP_MATMAS05_To_MQ and navigate to the SAP_MATMAS05_To_MQFlow.msgflow. This is the flow that was created from the package. It is currently empty, but we will build a simple flow where we trigger a request from SAP MATMA05 to generate an XML output (Figure 2-23).

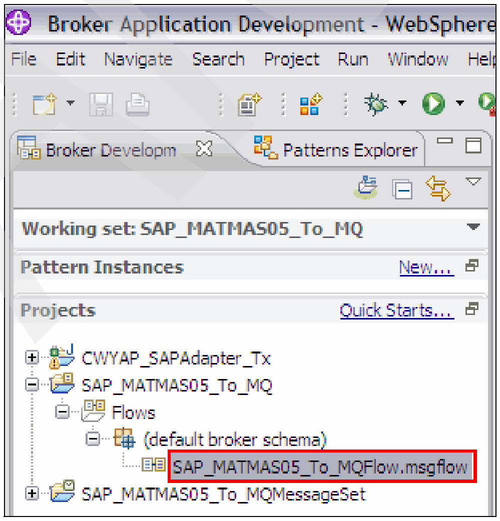

Broker Application Development - WebSphere
File Edit Navigate Search Project Run Window Help
Broker Developm
Patterns Explorer
Working set: SAP_MATMAS05_To_MQ
Pattern Instances
New...
Projects
Quick Starts...
CWYAP_SAPAdapter_Tx
SAP_MATMAS05_To_MQ
Flows
(default broker schema)
SAP_MATMAS05_To_MQFlow.msgflow
SAP_MATMAS05_To_MQMessageSet

Figure 2-23 Expand SAP_MATMAS05_To_MQ

16. Double-click the generated adapter component called Inbound_SAP_MATMAS05.inadapter to see the details (Figure 2-24). You can change the values for the adapter from here without having to run the wizard again.

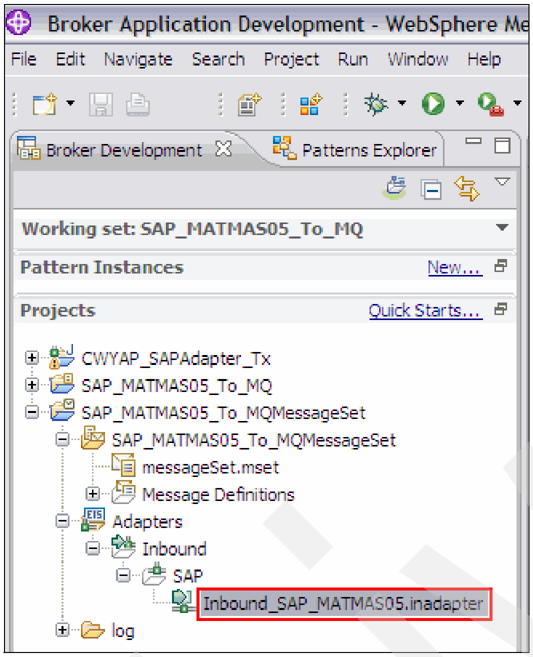

Broker Application Development - WebSphere Me
File Edit Navigate Search Project Run Window Help
Broker Development
Patterns Explorer
Working set: SAP_MATMAS05_To_MQ
Pattern Instances
New...
Projects
Quick Starts...
CWYAP_SAPAdapter_Tx
SAP_MATMAS05_To_MQ
SAP_MATMAS05_To_MQMessageSet
SAP_MATMAS05_To_MQMessageSet
messageSet.mset
Message Definitions
Adapters
Inbound
SAP
Inbound_SAP_MATMAS05.inadapter
log

Figure 2-24 See details of inbound adapter

17. Figure 2-25 on page 29 shows the object generated by the adapter wizard. The first result you see is the Message Set that has been created; this Message Set is the Object created from the SAP MATMAS05 structure. It includes all the message definitions for MATMAS05.

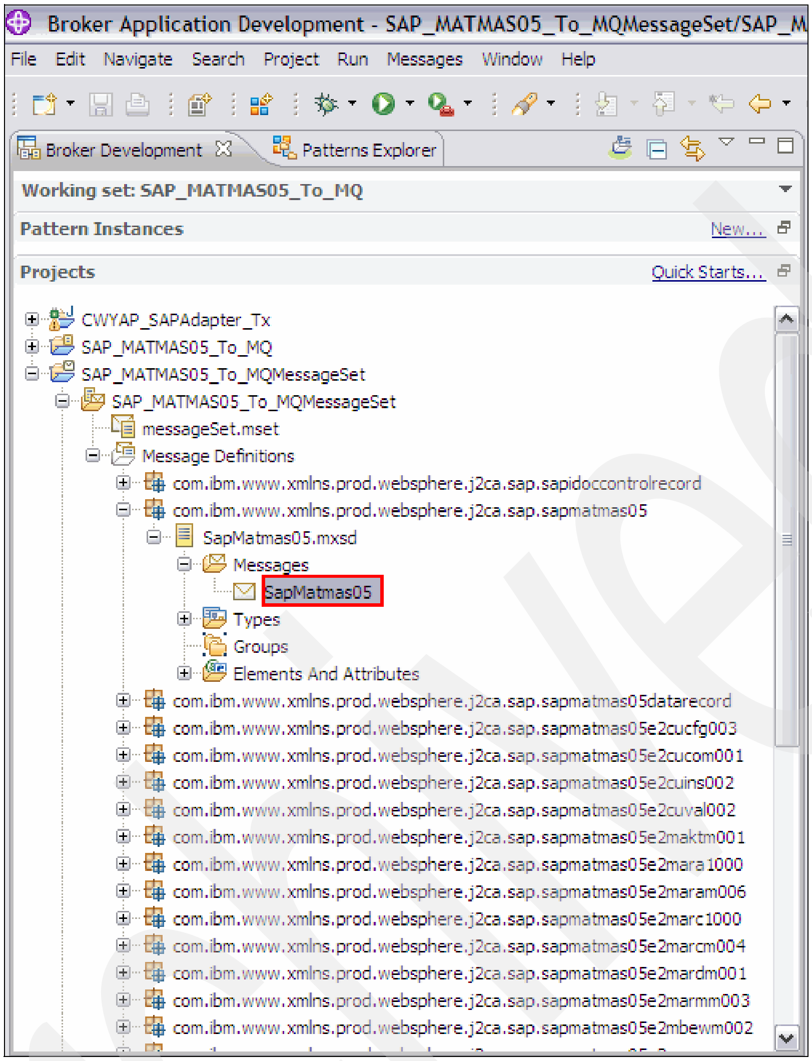

Broker Application Development - SAP_MATMAS05_To_MQMessageSet/SAP_M
File
Edit
Navigate
Search
Project
Run
Messages
Window
Help
Broker Development
Patterns Explorer
Working set: SAP_MATMAS05_To_MQ
Pattern Instances
New...
Projects
Quick Starts...
CWYAP_SAPAdapter_Tx
SAP_MATMAS05_To_MQ
SAP_MATMAS05_To_MQMessageSet
SAP_MATMAS05_To_MQMessageSet
messageSet.mset
Message Definitions
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapidoccontrolrecord
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05
SapMatmas05.mxsd
Messages
SapMatmas05
Types
Groups
Elements And Attributes
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05datarecord
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2cucfg003
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2cucom001
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2cuins002
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2cuval002
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2maktm001
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2mara1000
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2maram006
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2marc1000
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2marcm004
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2mardm001
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2marmm003
com.ibm.www.xmlns.prod.websphere.j2ca.sap.sapmatmas05e2mbewm002

Figure 2-25 Object generated by the adapter wizard

18. Click the **Inbound_SAP_MATMAS05.inadapter** adapter component and drag and drop it on the flow canvas (Figure 2-26). This create a message flow with a SAP Inbound node and a subflow.

    Click **Inbound_SAP_MATMAS05_Operations**.

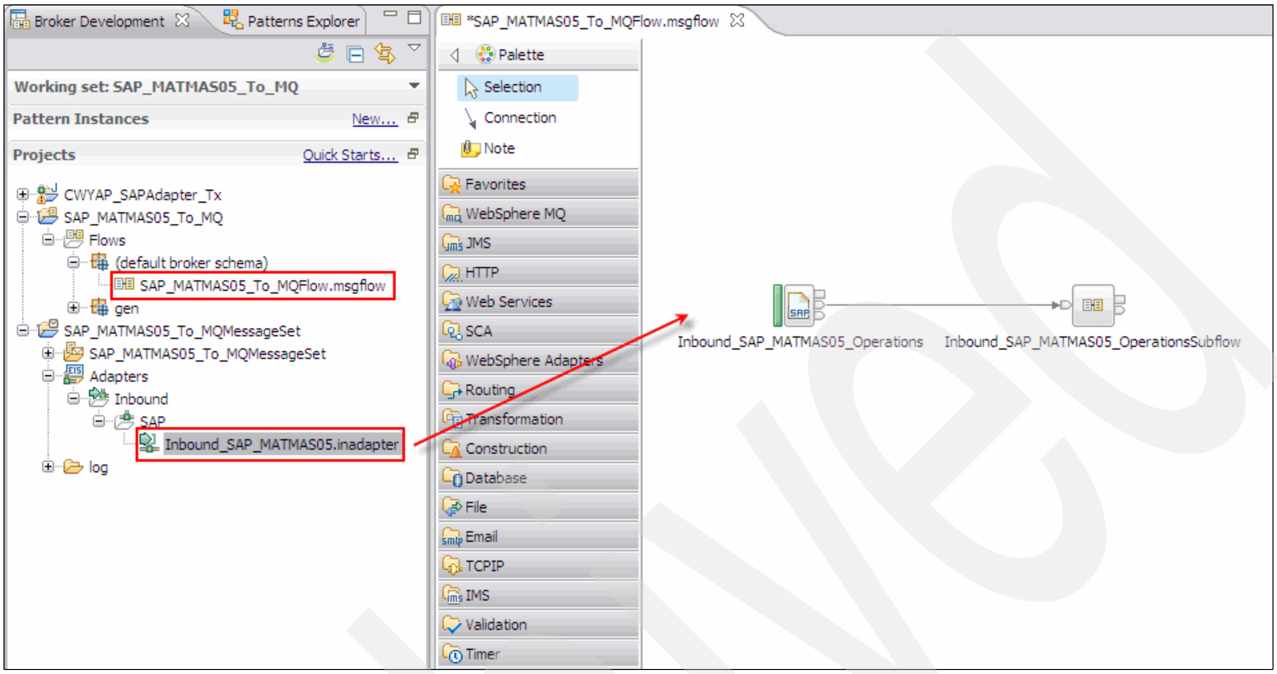
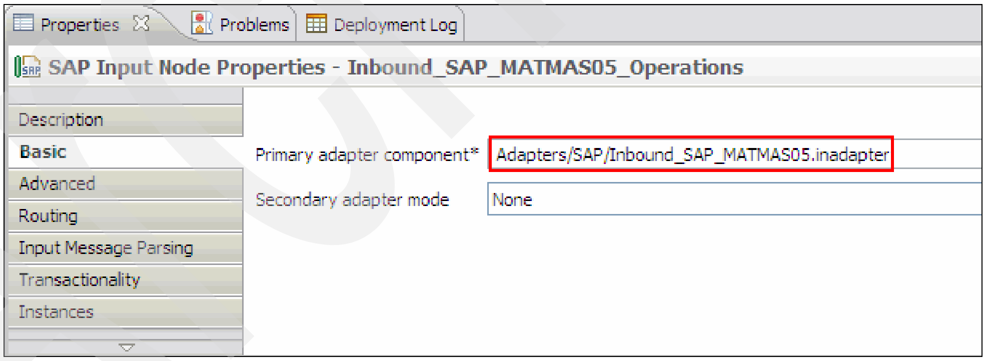


Figure 2-26 Create a message flow with a SAP Inbound node and a subflow

19. On the Properties tab, select **Basic**, which contains the Primary adapter component, here called Adapter/SAP/Inbound_SAP_MATMAS05.inadapter (Figure 2-27).



Primary adapter component* Adapters/SAP/Inbound_SAP_MATMAS05.inadapter

Secondary adapter mode None

Figure 2-27 View the adapter

20. Go back to the SAP_MATMAS05_To_MQFlow.msgflow and double-click Inbound_SAP_MATMAS05_OperationsSubflow to open the subflow (Figure 2-28).

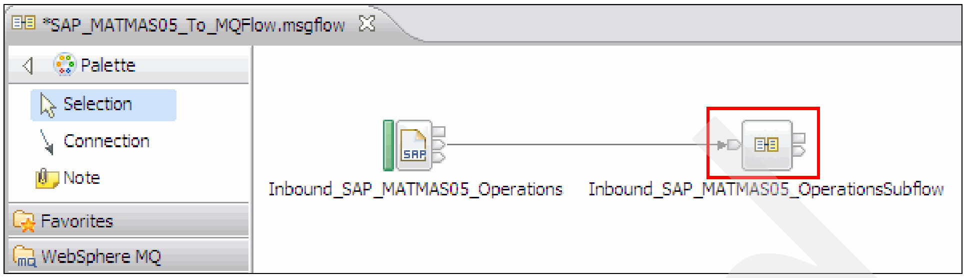

*SAP_MATMAS05_To_MQFlow.msgflow
Palette
Selection
Connection
Note
Favorites
WebSphere MQ
Inbound_SAP_MATMAS05_Operations
Inbound_SAP_MATMAS05_OperationsSubflow

Figure 2-28 Open the subflow

21. The subflow will appear (Figure 2-29).

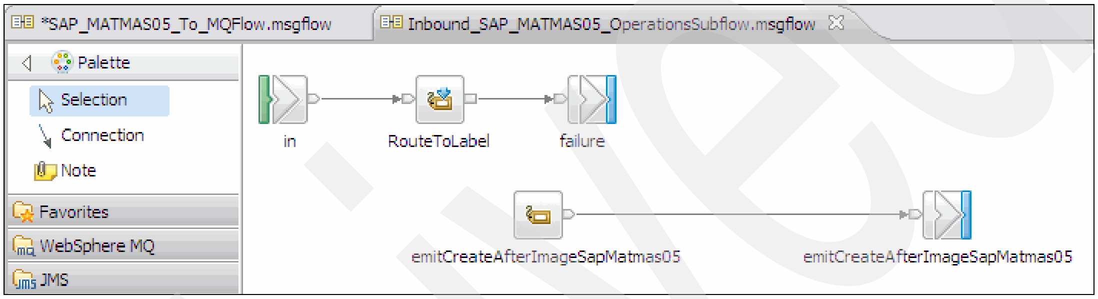

*SAP_MATMAS05_To_MQFlow.msgflow
Inbound_SAP_MATMAS05_OperationsSubflow.msgflow
Palette
Selection
Connection
Note
Favorites
WebSphere MQ
JMS
in
RouteToLabel
failure
emitCreateAfterImageSapMatmas05
emitCreateAfterImageSapMatmas05

Figure 2-29 The subflow

22. Go back to the SAP_MATMAS05_To_MQFlow.msgflow. From the Palette, select WebSphere MQ palette, select the MQOutput component, and drag and drop it on the flow canvas (Figure 2-30).

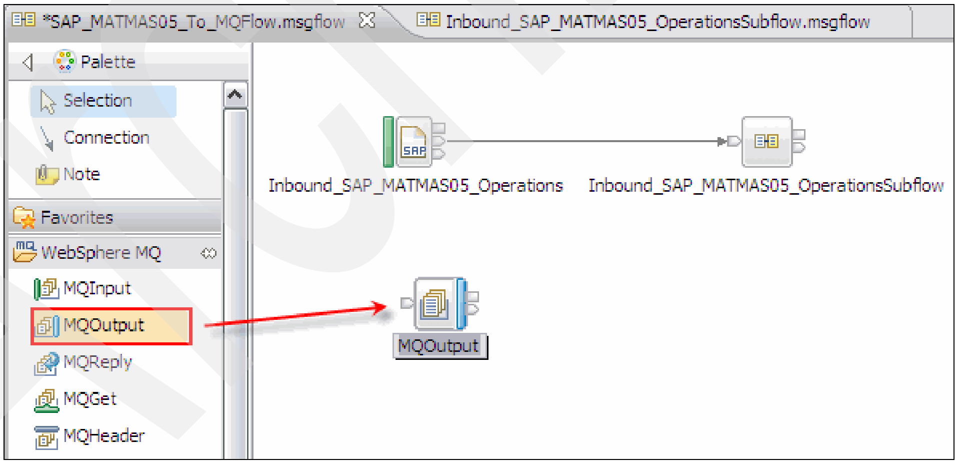
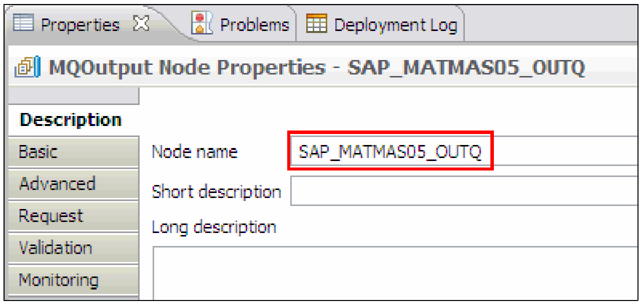

*SAP_MATMAS05_To_MQFlow.msgflow
Inbound_SAP_MATMAS05_OperationsSubflow.msgflow
Palette
Selection
Connection
Note
Favorites
WebSphere MQ
MQInput
MQOutput
MQReply
MQGet
MQHeader
Inbound_SAP_MATMAS05_Operations
Inbound_SAP_MATMAS05_OperationsSubflow
MQOutput

Figure 2-30 Choose MQ Output component

23. Double-click the newly selected MQOutput component. From the Properties tab click **Description** and enter Node name `SAP_MATMAS05_OUTQ` (Figure 2-31).



MQOutput Node Properties - SAP_MATMAS05_OUTQ

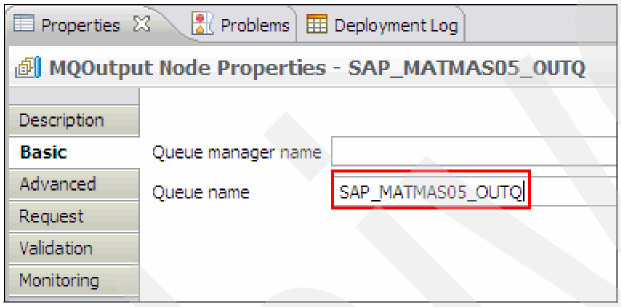
Node name: SAP_MATMAS05_OUTQ

Short description:

Long description:

Figure 2-31 View MQOutput component description

24. On the Basic tab, enter `SAP_MATMAS05_OUTQ` (Figure 2-32**)**; this is the name of the queue where the message will be put after the event from SAP has been triggered (later we show you how to create this queue from the IBM WebSphere Message Broker Explorer).



MQOutput Node Properties - SAP_MATMAS05_OUTQ

Queue manager name:

Queue name: SAP_MATMAS05_OUTQ

Figure 2-32 View output queue

25. Go back to the SAP_MATMAS05_To_MQFlow.msgflow (Figure 2-33); connect the Inbound_SAP_MATMAS05_OperationsSubflow to the SAP_MATMAS05_OUTQ MQOutput Queue.

    Press **Ctrl + s** to save the message flow. This concludes the development part of the exercise.

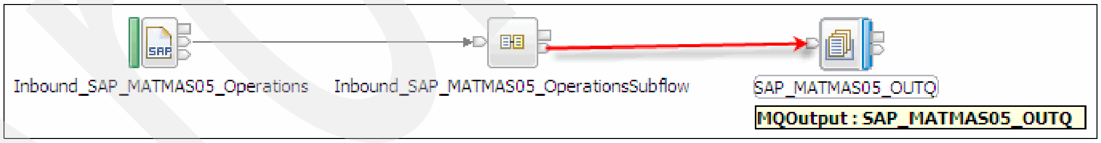

SAP
Inbound_SAP_MATMAS05_Operations
Inbound_SAP_MATMAS05_OperationsSubflow
SAP_MATMAS05_OUTQ
MQOutput : SAP_MATMAS05_OUTQ

Figure 2-33 Go to the message flow

This ALE Inbound sample scenario demonstrated how the WebSphere Adapter for SAP Software can be used to receive events from the SAP EIS by using a sample IDoc. The sample scenario shows how to configure the adapter from a wizard-based component and, once deployed, how to configure an endpoint in order to receive an SAP-invoked IDoc asynchronous event.

2.3 Launch the Command Console

All WebSphere Message Broker commands that have the “mqsi” prefix need to be run in the WebSphere Message Broker Command Console on Windows. This command prompt has been configured with the required environment for WebSphere Message Broker and is configured by default on Windows.

1. If you have only one installation of the runtime product, open a command console, and click **Start** → **Programs** → **IBM WebSphere Message Brokers 7.0** → **Command Console** (Figure 2-34).

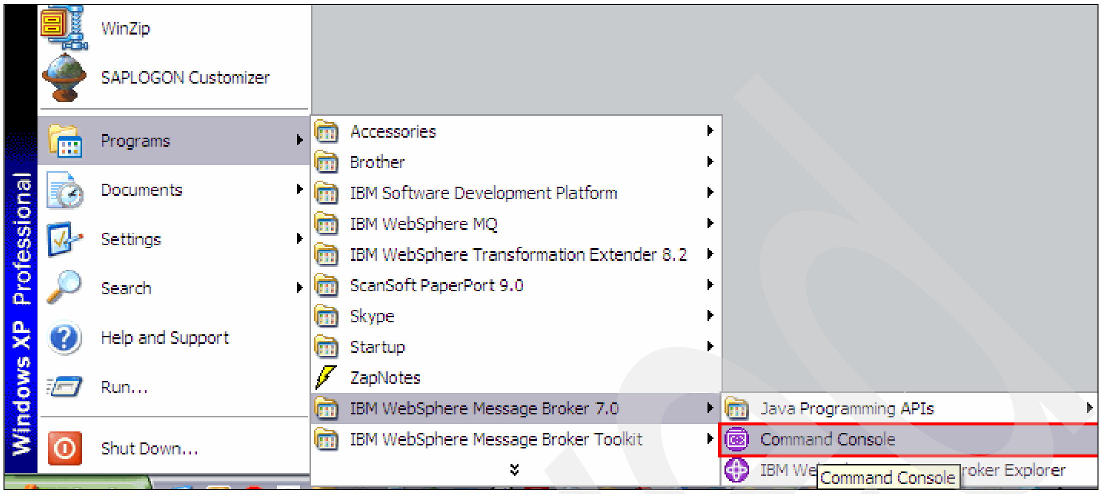


Figure 2-34 Start the command console

2. On Windows, the WebSphere message broker command console is essentially a command prompt with appropriate path and environment variables set.

   For successful deployment of message flows with SAP nodes, you must configure the WebSphere Message Broker runtime with the prerequisite SAP JCo libraries. These libraries are used by the underlying SAP adapter component to establish connections with the SAP system.

   You must run the following **mqsichangeproperties** commands from a command environment to set the location of the directories that contain the SAP JCo libraries.

   Run the following commands, changing the SAP client directory to match your platform and your choice of location:

   ```
   mqsichangeproperties MB7BROKER -c EISProviders -o SAP -n jarsURL -v C:\SAP_JCO_3
   ```

   ```
   mqsichangeproperties MB7BROKER -c EISProviders -o SAP -n nativeLibs -v C:\SAP_JCO_3
   ```

3. To verify that the properties have been set up correctly, run the following command:

   ```
   mqsireportproperties MB7BROKER -c EISProviders -o SAP -r
   ```

   The result of running this command is shown in Example 2-1.

   Example 2-1 EIS provider information

   ```
   EISProviders
     SAP
       jarsURL='C:\SAP_JCO_3'
       nativeLibs='C:\SAP_JCO_3'
   BIP8071I: Successful command completion.
   ```

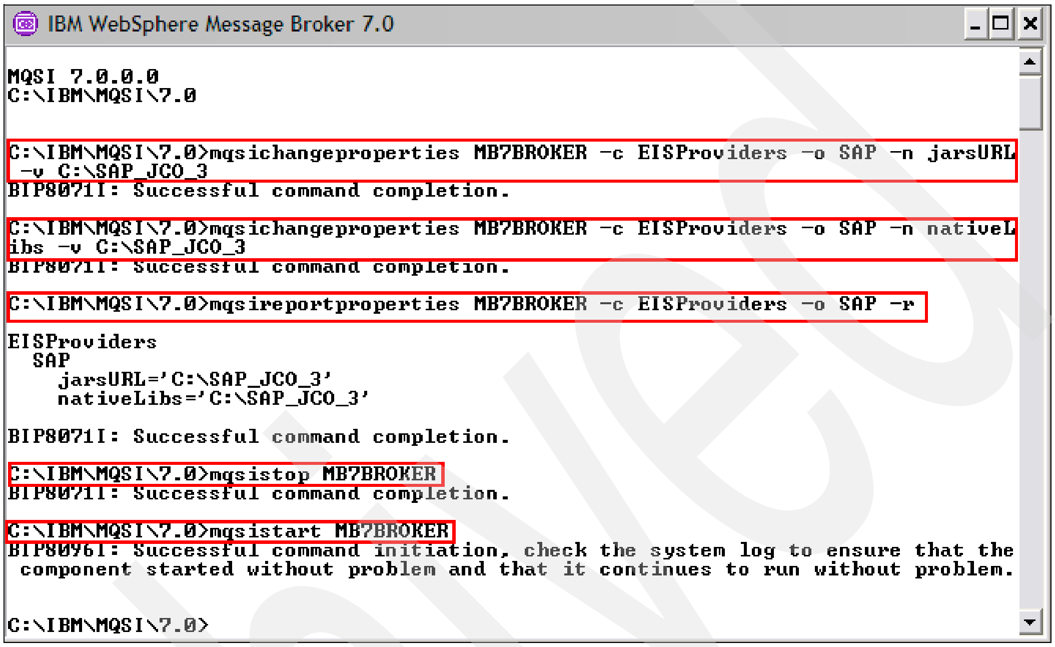
   This is a good sign, indicating that the JCo is running with no issues.

   The commands to restart WebSphere Message Broker execution group must be issued from the command console of the broker system and special authority is required.

Stop and restart the WebSphere Message Broker using these commands:

```
mqsistop MB7BROKER
mqsistart MB7BROKER
```

After each command you will receive the message `BIP80711: Successful command completion` (Figure 2-35).



```
MQSI 7.0.0.0
C:\IBM\MQSI\7.0

C:\IBM\MQSI\7.0>mqsichangeproperties MB7BROKER -c EISProviders -o SAP -n jarsURL
 -v C:\SAP_JCO_3
BIP8071I: Successful command completion.

C:\IBM\MQSI\7.0>mqsichangeproperties MB7BROKER -c EISProviders -o SAP -n nativeL
ibs -v C:\SAP_JCO_3
BIP8071I: Successful command completion.

C:\IBM\MQSI\7.0>mqsireportproperties MB7BROKER -c EISProviders -o SAP -r

EISProviders
  SAP
    jarsURL='C:\SAP_JCO_3'
    nativeLibs='C:\SAP_JCO_3'

BIP8071I: Successful command completion.

C:\IBM\MQSI\7.0>mqsistop MB7BROKER
BIP8071I: Successful command completion.

C:\IBM\MQSI\7.0>mqsistart MB7BROKER
BIP8096I: Successful command initiation, check the system log to ensure that the
 component started without problem and that it continues to run without problem.


C:\IBM\MQSI\7.0>
```

Figure 2-35 Executing commands and resulting messages

2.4 Deploy the bar file to the broker

To deploy a message flow, you have to create a package of deployable resources. One option is to create a broker archive (a .bar file).

The broker archive (.bar file) is the unit of deployment to the broker. It can contain any number of compiled message flows (.cmf) and message sets (.dictionary), and a single deployment descriptor. The .bar file is a zip format file that is used by both human users (for example, an administrator manually configuring archive deployment) and by automation. It serves to parameterize system objects like files, queues, Queue Managers, and databases without changing source code.

The .bar file can be saved and managed externally with a code versioning system. As a zip format archive, the .bar file can also contain any additional files you need, for example, the message flow source code or test message data, and thus serve as a means for code distribution. Only the .cmf and .dictionary files are added to the Execution Group, replacing any deployed objects with the same name.

Use the following steps to deploy the .bar file:

1. Click the **SAP_MATMAS05_To_MQFlow.msgflow** component and drag it into the default execution group in the Broker view (Figure 2-36).

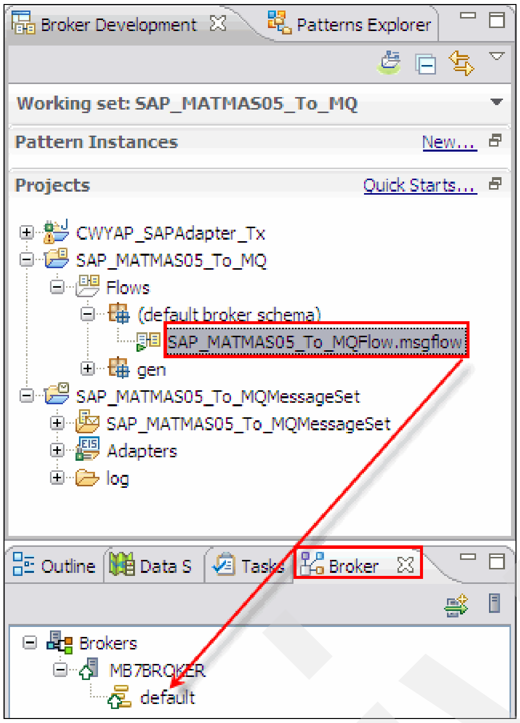
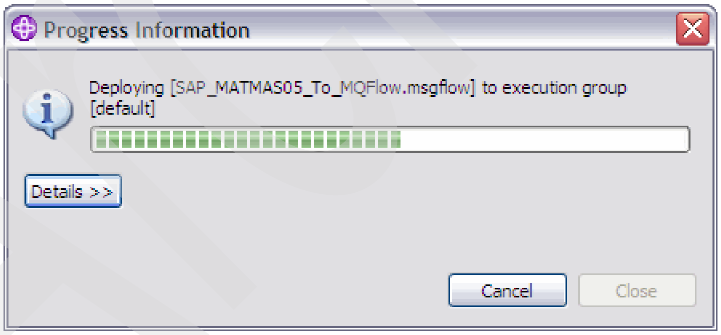


Figure 2-36 Drag SAP_MATMAS05 to Broker

Note: This deployment might take a while to complete; be patient.

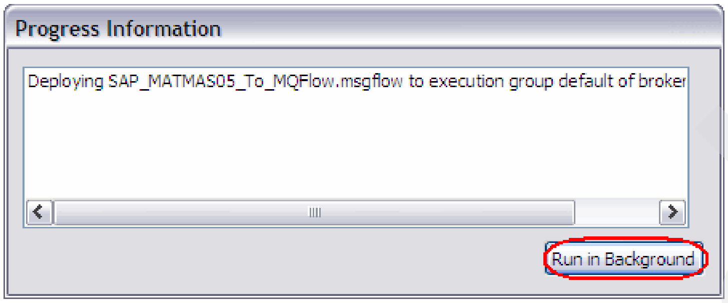
The Progress Information pop-up window (Figure 2-37) appears, indicating the bar file is being generated and deploying for SAP_MATMAS05_To_MQFlow.msgflow in to the default execution group.



Deploying [SAP_MATMAS05_To_MQFlow.msgflow] to execution group [default]

Figure 2-37 Deploying SAP_MATMAS05_To_MQFlow.msgflow

2. Click **Run in Background** (Figure 2-38) to have the process run in the background since this might take a short while.



Deploying SAP_MATMAS05_To_MQFlow.msgflow to execution group default of broker

Figure 2-38 View progress information

When the deployment has completed successfully, the message flows and other resources are displayed under the execution group. All the components have been deployed under the default execution group.

You are now ready to test.

3. Observe that a new folder has been created called GeneratedBarFiles (Figure 2-39). Click the [**+**] sign to expand this folder; notice under the SAP_MATMAS05_To_MQ directory there is a file called SAP_MATMAS05_To_MQFlow.msgflow.generated.bar.

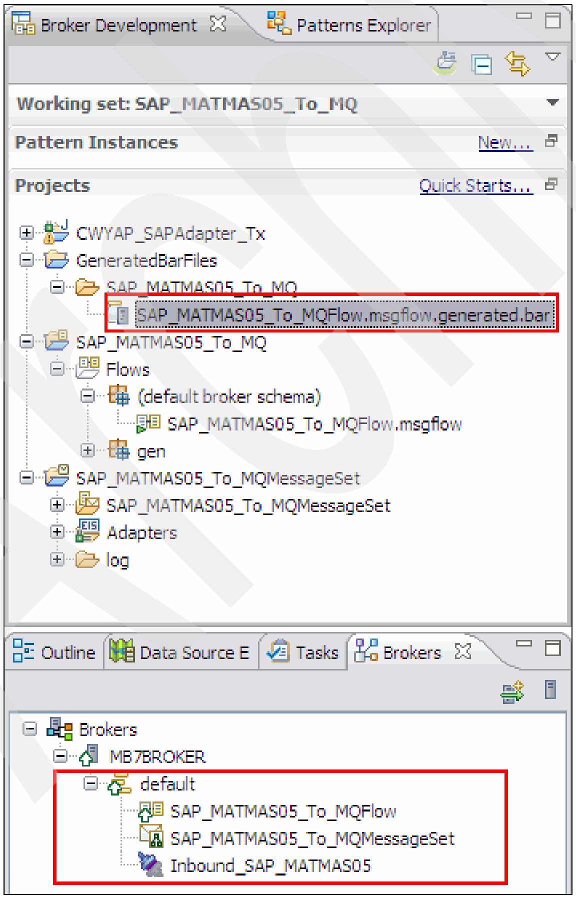
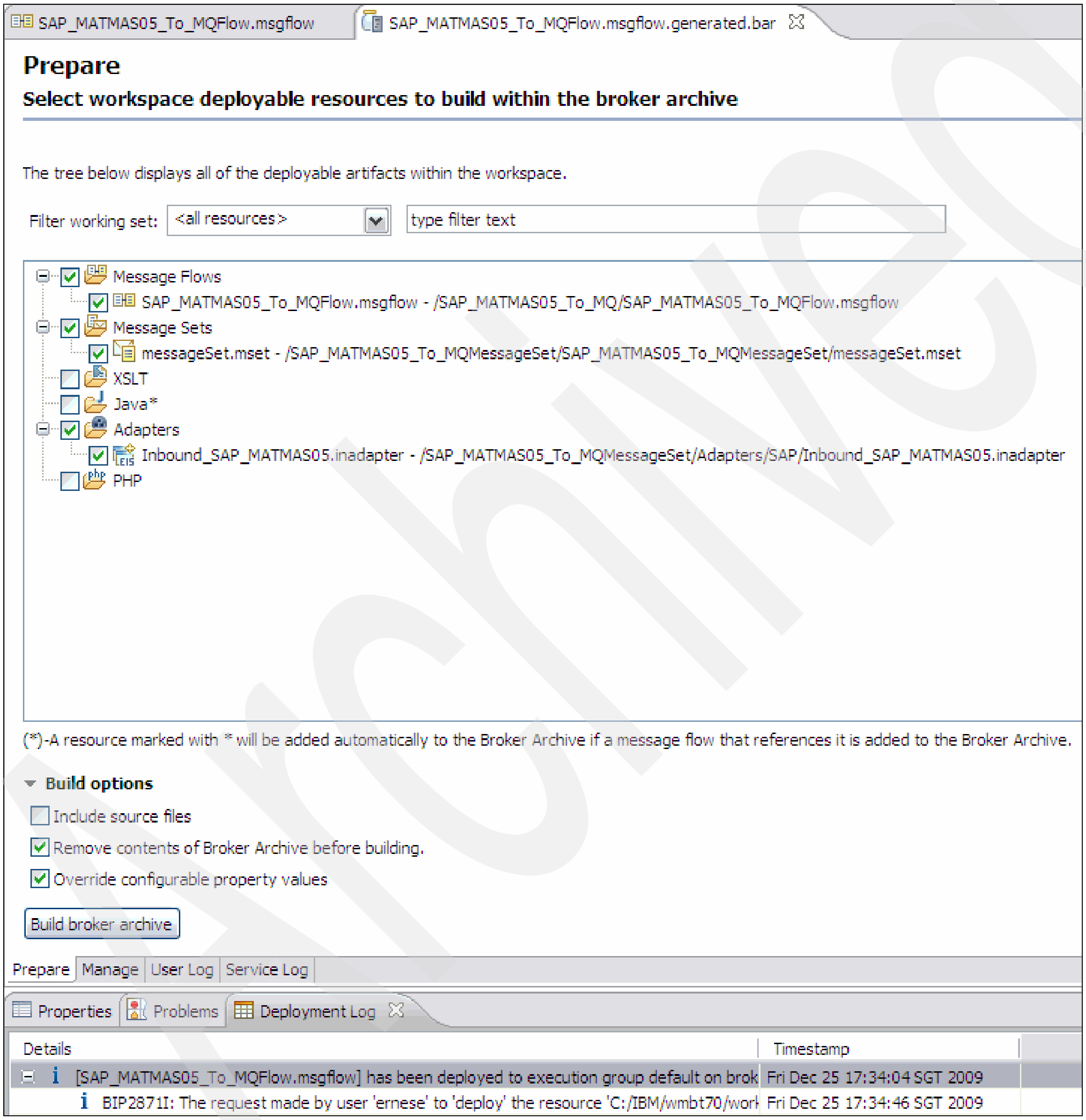


Figure 2-39 View generated components

4. Double-click the newly generated SAP_MATMAS05_To_MQFlow.msgflow.generated.bar file (Figure 2-40). It will open the Prepare tab, where you can observe all the deployable files, such as:

   – SAP_MATMAS05_To_MQFlow.msgflow
   – SAP_MATMAS05_To_MQMessageSet.mset
   – Inbound_SAP_MATMAS05.inadapter

   The Event Log in the Message Brokers Toolkit can be used to check that the broker has been successfully added to the domain.



SAP_MATMAS05_To_MQFlow.msgflow
SAP_MATMAS05_To_MQFlow.msgflow.generated.bar

Prepare

Select workspace deployable resources to build within the broker archive

The tree below displays all of the deployable artifacts within the workspace.

Filter working set: <all resources>   type filter text

- ☑ Message Flows
  - ☑ SAP_MATMAS05_To_MQFlow.msgflow - /SAP_MATMAS05_To_MQ/SAP_MATMAS05_To_MQFlow.msgflow
- ☑ Message Sets
  - ☑ messageSet.mset - /SAP_MATMAS05_To_MQMessageSet/SAP_MATMAS05_To_MQMessageSet/messageSet.mset
- □ XSLT
- □ Java*
- ☑ Adapters
  - ☑ Inbound_SAP_MATMAS05.inadapter - /SAP_MATMAS05_To_MQMessageSet/Adapters/SAP/Inbound_SAP_MATMAS05.inadapter
- □ PHP

(*)-A resource marked with * will be added automatically to the Broker Archive if a message flow that references it is added to the Broker Archive.

Build options

□ Include source files

☑ Remove contents of Broker Archive before building.

☑ Override configurable property values

Details	Timestamp
[SAP_MATMAS05_To_MQFlow.msgflow] has been deployed to execution group default on brok	Fri Dec 25 17:34:04 SGT 2009
BIP2871I: The request made by user 'ernese' to 'deploy' the resource 'C:/IBM/wmbt70/work	Fri Dec 25 17:34:46 SGT 2009

Figure 2-40 Observe all of the deployable files

2.5 Launch the IBM WebSphere Message Broker Explorer plug-in for WebSphere MQ Explorer

WebSphere MQ Explorer is an Eclipse-based graphical configuration tool that is provided with WebSphere MQ V7. In WebSphere MQ Explorer you can configure your queue managers and other objects. You can extend WebSphere MQ Explorer to manage other resources as well.

This plug-in enhances WebSphere MQ Explorer so that you can administer a WebSphere Message Broker network. The following features are available in the Navigator view after the plug-in is installed:

- View information and help in the Content view for each configuration manager, broker, execution group, and flow.
- Create and delete local brokers without using the command line.
- Start, stop, create, and delete brokers, execution groups, and message flows.
- Display and delete message flow resources in the Navigator view.
- Deploy a bar file to multiple execution groups in a single step.
- Display event log events from the selected Configuration Manager or broker in the Event Log view.

Perform the following steps to launch the WebSphere MQ Explorer and configure your queue managers:

1. Click **Start** → **Programs** → **IBM WebSphere Message Broker 7.0** → **IBM WebSphere Message Broker Explorer** (Figure 2-41).

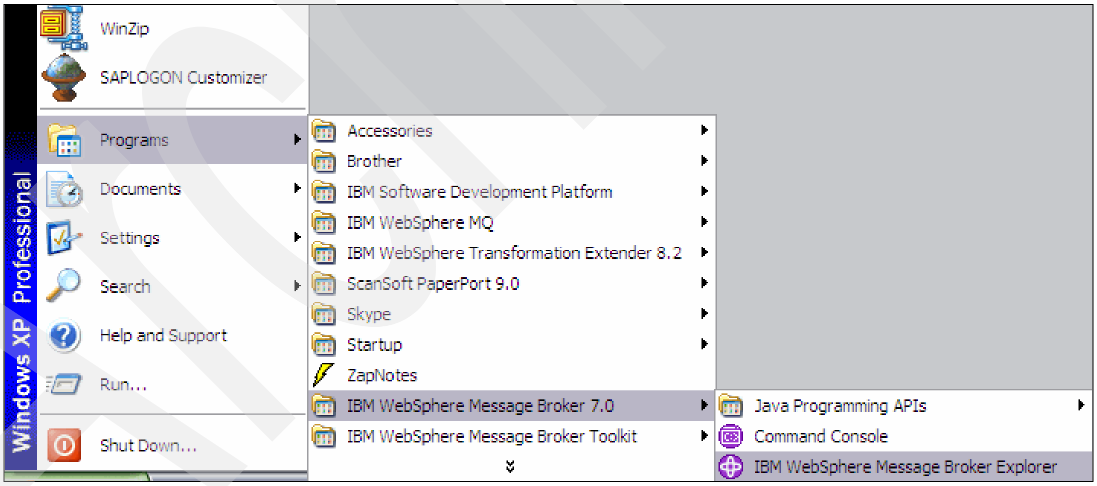


Figure 2-41 Start WebSphere Message Broker Explorer

2. Expand the Default WebSphere Message Broker v7.0 Queue Manager, click [+] MB7QMGR, and locate Queues (Figure 2-42).

   Under Broker, the MB7BROKER folder contains all the artifacts that have been deployed:

   – SAP_MATMAS05_To_MQFlow

   – Inbound_SAP_MATMAS05.inadapter

   – SAP_MATMAS05_To_MQMessageSet.xsdzip

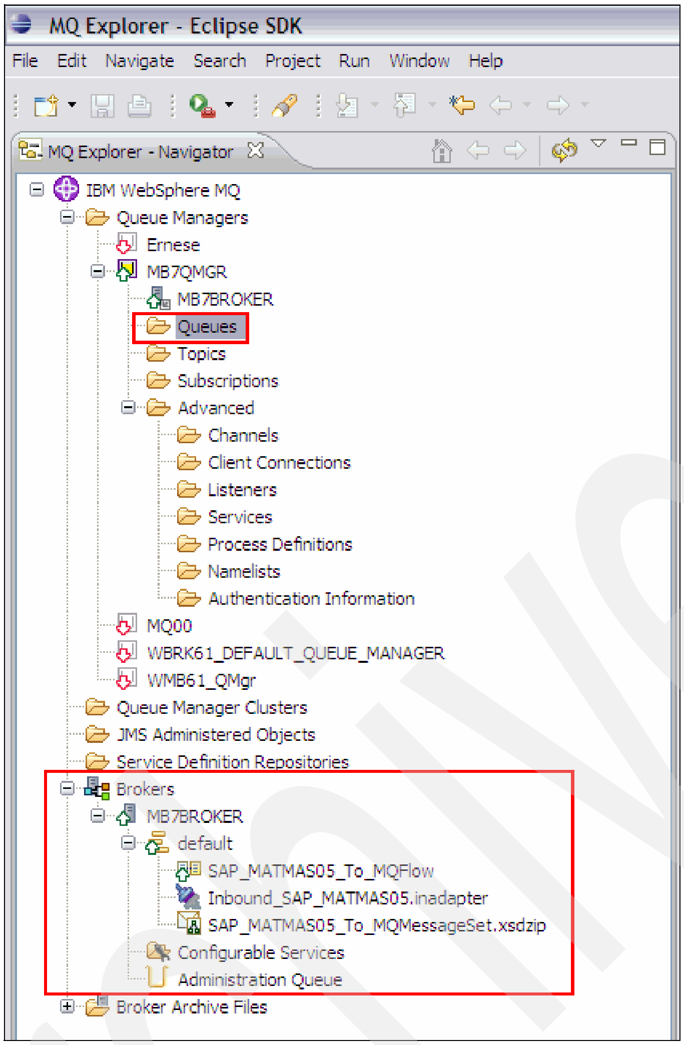


Figure 2-42 View the artifacts that have been deployed

3. Right-click **Queues** and select **New** → **Local Queue** (Figure 2-43).

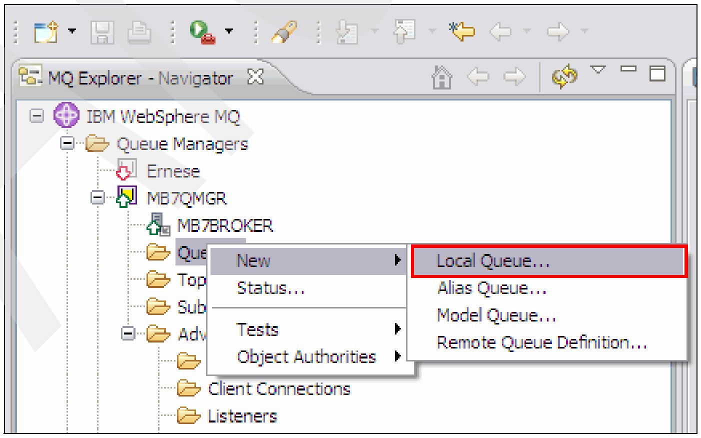
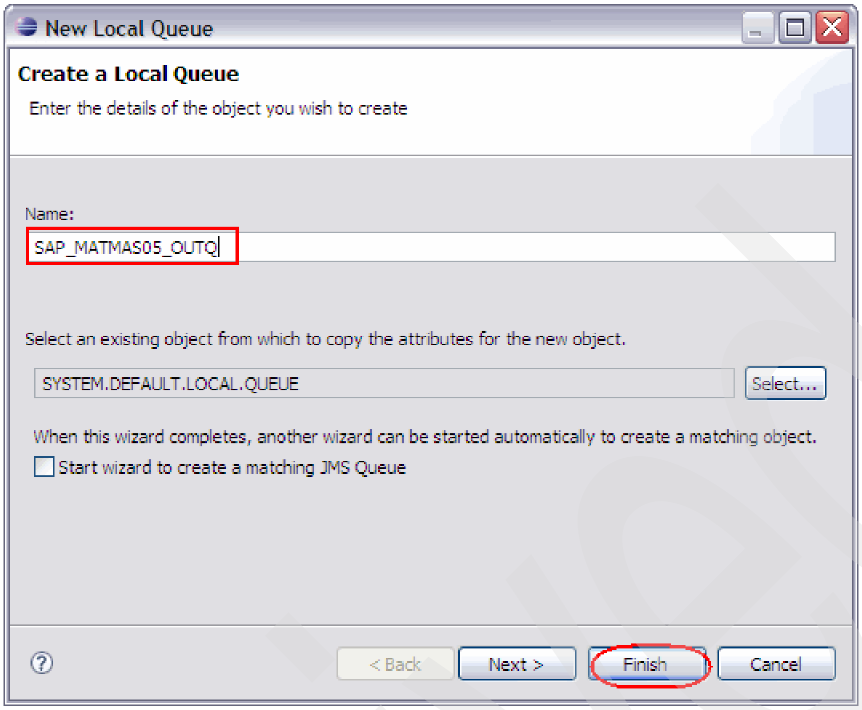


Figure 2-43 Select New -> Local Queue

4. Enter `SAP_MATMAS05_OUTQ` as the local queue name and click **Finish** (Figure 2-44).



Create a Local Queue

Enter the details of the object you wish to create

Name:
SAP_MATMAS05_OUTQ

Select an existing object from which to copy the attributes for the new object.

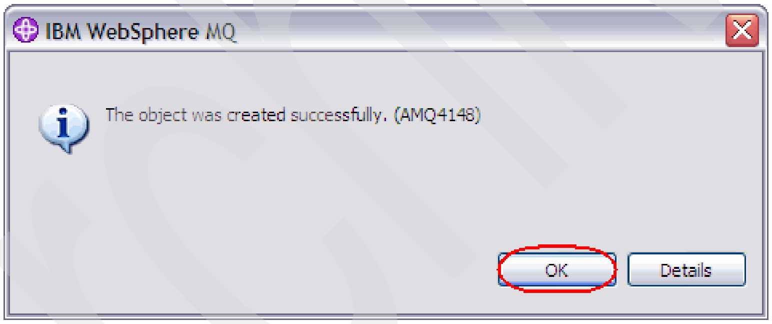
SYSTEM.DEFAULT.LOCAL.QUEUE

When this wizard completes, another wizard can be started automatically to create a matching object.

□ Start wizard to create a matching JMS Queue

Figure 2-44 Create a Local Queue

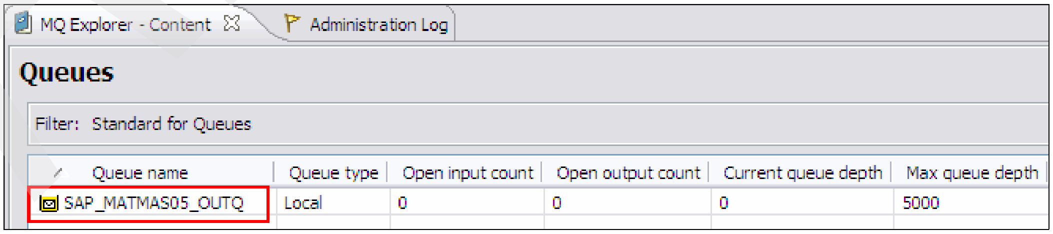
5. When the acknowledgement dialog is displayed, click **OK** (Figure 2-45).



The object was created successfully. (AMQ4148)

Figure 2-45 Message: successful object creation

6. View the newly created queue SAP_MATMAS05_OUTQ and verify that the Current queue depth is 0 (Figure 2-46).



Queues

Filter: Standard for Queues

Queue name	Queue type	Open input count	Open output count	Current queue depth	Max queue depth
SAP_MATMAS05_OUTQ	Local	0	0	0	5000

Figure 2-46 View the new queue

2.6 Create RFC connection from the SAP system to the broker

For testing we are going to use the SAP GUI. Perform the following steps to create and test an RFC connection from the SAP system to the broker:

1. Click **Start** → **Programs** → **SAP Front End** → **SAP Logon** (Figure 2-47).

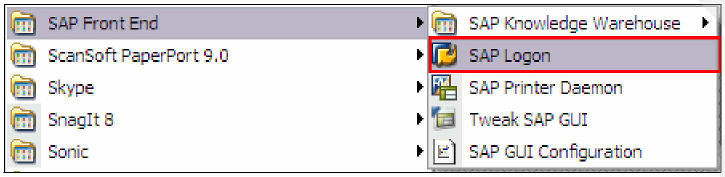


Figure 2-47 Start SAP logon

2. On the SAP Logon dialog highlight the appropriate system. In our case this is EC5[9.30.197.1] (Figure 2-48).

   Click **Log On**.

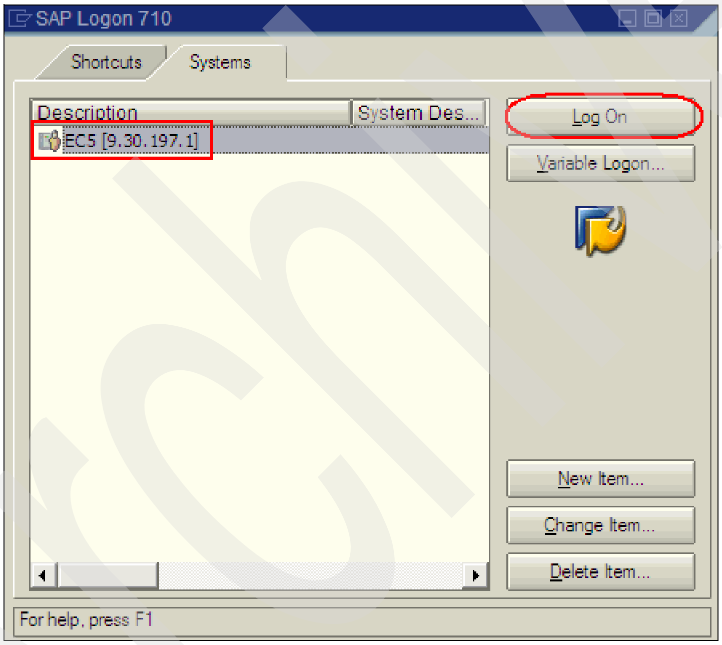
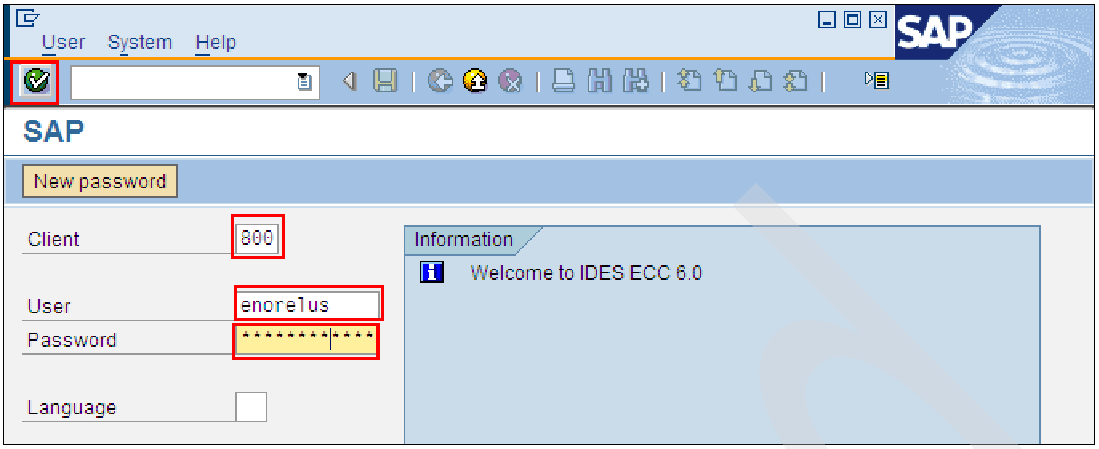


Figure 2-48 Log on to your system

3. Make the appropriate entries on the SAP Logon page (Figure 2-49):

   – Client: <*your client number*>
   – User: <*your user name*>
   – Password: <*your password*>

> **Important:** Enter the Client, User and Password and then press **Enter**. *Do not* click the “New Password” button. Doing so will cause the logon to fail and the SAP user account will be locked out of the SAP system. As you enter the Password the existing asterisks are not removed or replaced, which is a bit confusing, but you will see the point move in the Password text box.



Client 800

User enorelus

Password ************

Language

Information

Welcome to IDES ECC 6.0

Figure 2-49 Enter client information

4. After you have successfully logged on to the SAP system enter `/nSM59` in the text box. Click **Enter** (the green check mark) (Figure 2-50).

   SM59 is the SAP transaction code; this allows you to observe the Configuration of RFC connections.

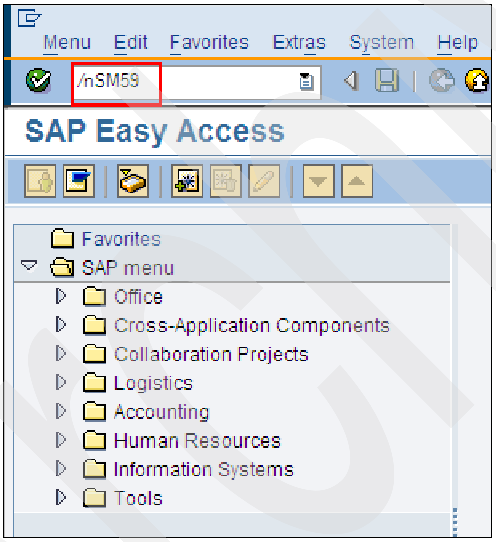
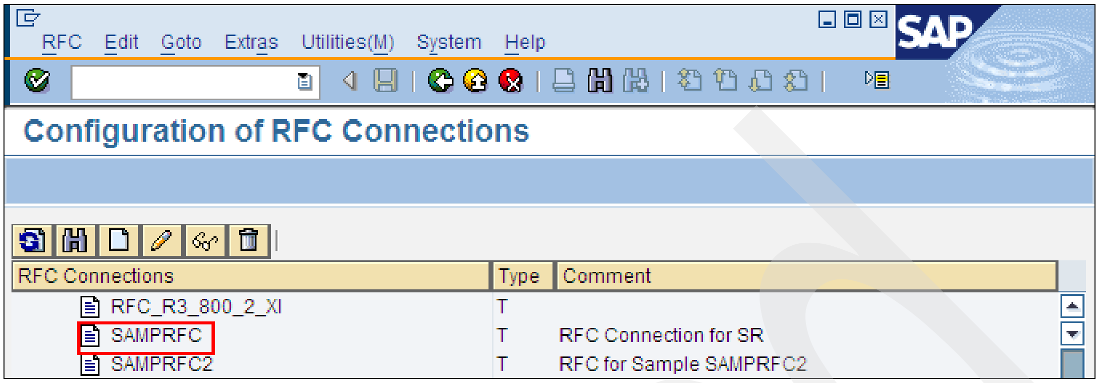


Figure 2-50 Observe SM59

5. Click **TCP/IP connections**, then locate and double-click **SAMPRFC**. The resulting window is shown in Figure 2-51.



Configuration of RFC Connections

RFC Connections	Type	Comment
RFC_R3_800_2_XI	T	
SAMPRFC	T	RFC Connection for SR
SAMPRFC2	T	RFC for Sample SAMPRFC2

Figure 2-51 Configure TCP/IP connection

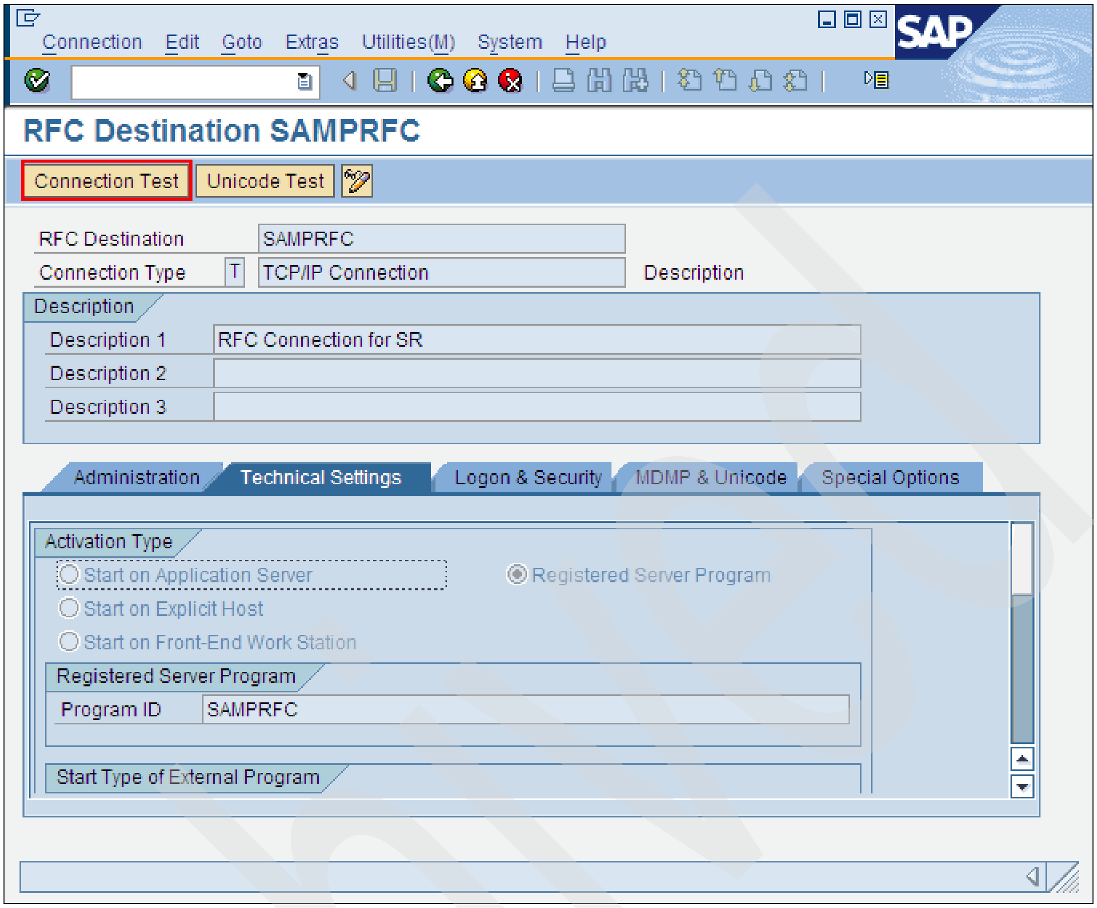
6. If not already done, set RFC Destination name to an appropriate value, for example SAMPRFC. Choose a name for the RFC Destination that is 8 characters or less and use the same name wherever SAMPRFC is used in our examples in this section.

   It is possible to use different names for the following steps. Consult the SAP systems documentation to understand exactly how to do this.

   For this exercise, use the same name for all created objects (Figure 2-52).

   a. Check that the connection type is set to **T**.

   b. Click **Connection Test**.



RFC Destination SAMPRFC

RFC Destination		SAMPRFC
Connection Type	T	TCP/IP Connection

Description

Description 1	RFC Connection for SR
Description 2	
Description 3	

Activation Type

○ Start on Application Server ● Registered Server Program

○ Start on Explicit Host

○ Start on Front-End Work Station

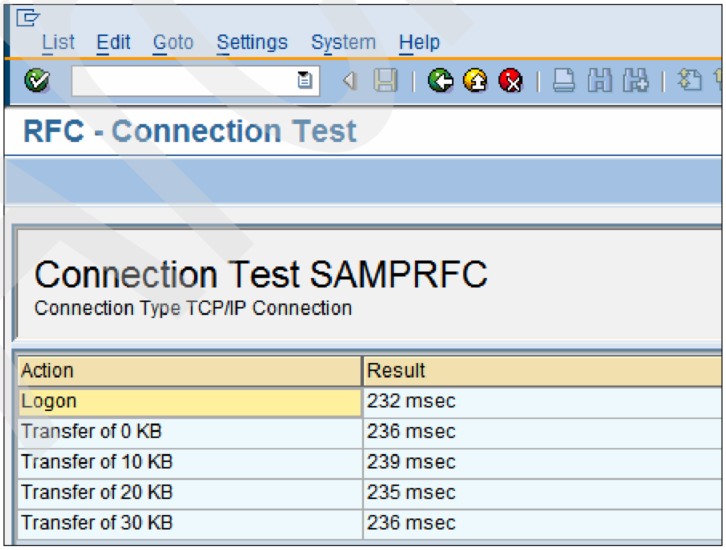
Registered Server Program

Program ID SAMPRFC

Start Type of External Program

Figure 2-52 Set connection type

You should get the screen shown in Figure 2-53; this indicates that you have established a connection and that your connectivity is set for the demo.



RFC - Connection Test

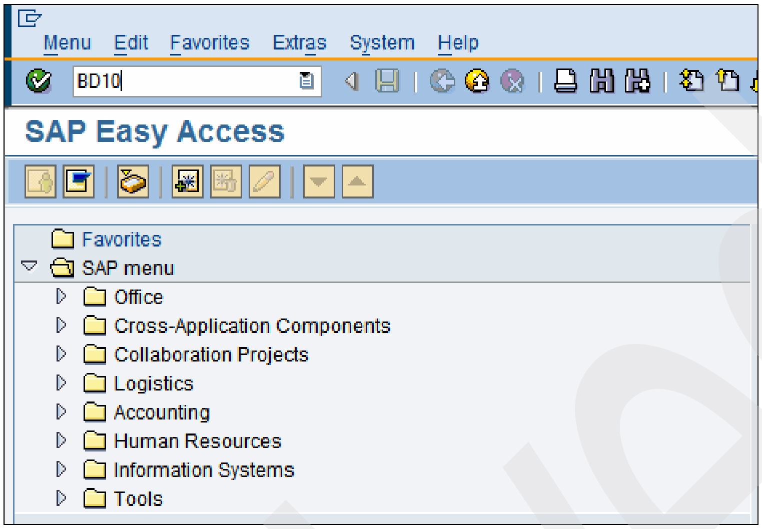
Connection Test SAMPRFC

Connection Type TCP/IP Connection

Action	Result
Logon	232 msec
Transfer of 0 KB	236 msec
Transfer of 10 KB	239 msec
Transfer of 20 KB	235 msec
Transfer of 30 KB	236 msec

Figure 2-53 Indicates setting connection

7. As shown in Figure 2-54, send a Material Master to a Local System by keying `/nBD10` (BD10 is for Send Material). The system distributes the material created centrally to the sales system and confirms that this event has happened.



BD10

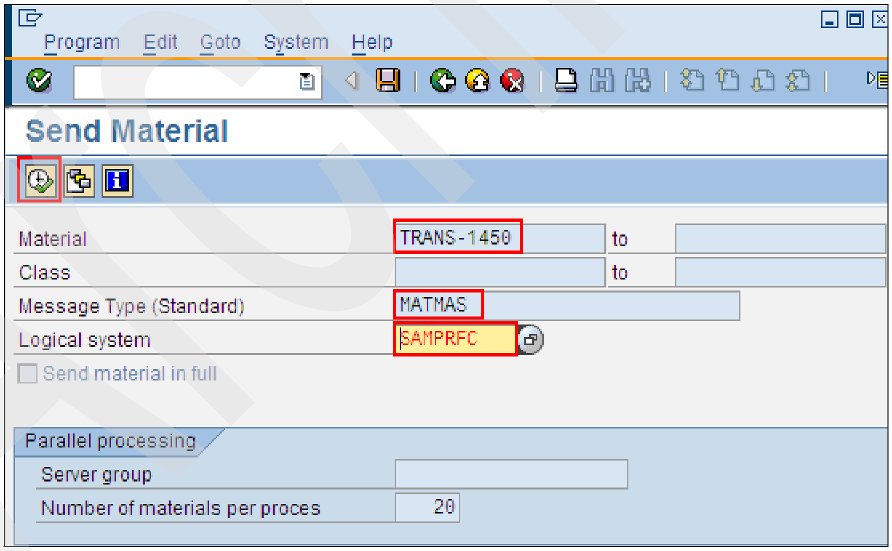
Figure 2-54 Send a Material Master to a Local System

8. To test the outbound partner profile from SAP (Figure 2-55):

   a. Search for and select a material with the ID: **TRANS-1450**

   b. Enter the Message Type (Standard): `MATMAS05`.

   c. Enter the Logical system you created: `SAMPRFC`



Material	TRANS-1450	to	
Class		to	
Message Type (Standard)	MATMAS		
Logical system	SAMPRFC		

□ Send material in full

Parallel processing

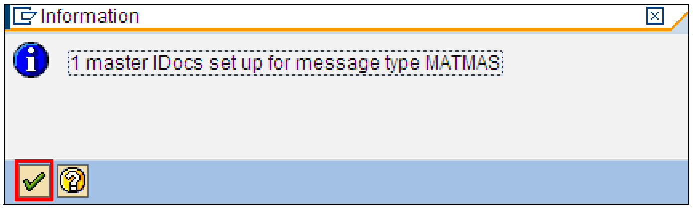
Server group	
Number of materials per proces	20

Figure 2-55 Test the outbound partner profile

9. When you send the material (message), you should receive a success confirmation in a pop-up window (Figure 2-56).

   The system creates a material IDoc.

   Click the **Green check mark**.

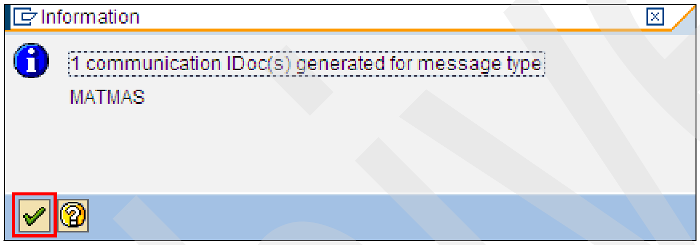


1 master IDocs set up for message type MATMAS

Figure 2-56 Message confirmation

10. The system creates a communication IDoc (Figure 2-57).

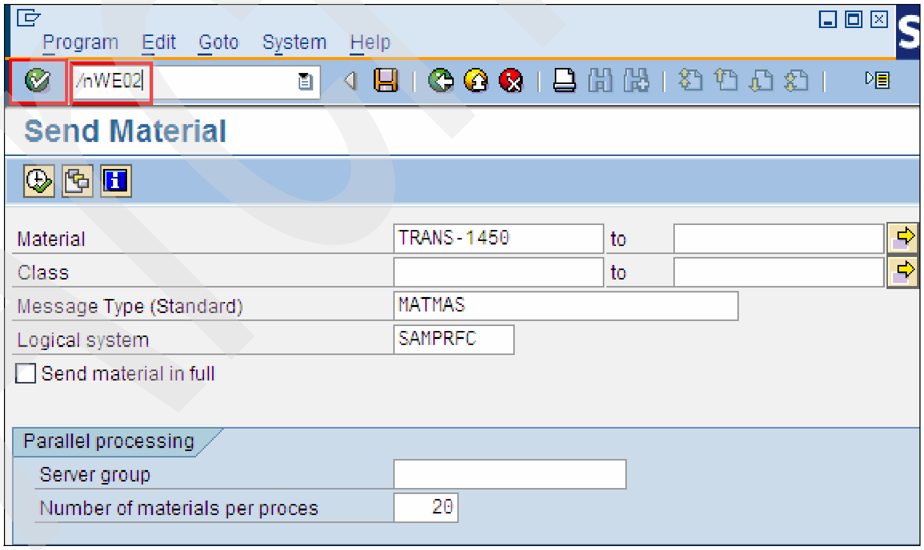
    Click the **Green check mark**.



1 communication IDoc(s) generated for message type MATMAS

Figure 2-57 Creating communication IDoc

11. Enter `/nWE02` (WE02 meaning Display IDOCS). This brings you to the Display IDOCS screen (Figure 2-58).



/nWE02

Send Material

Material	TRANS-1450	to	
Class		to	
Message Type (Standard)	MATMAS		
Logical system	SAMPRFC		

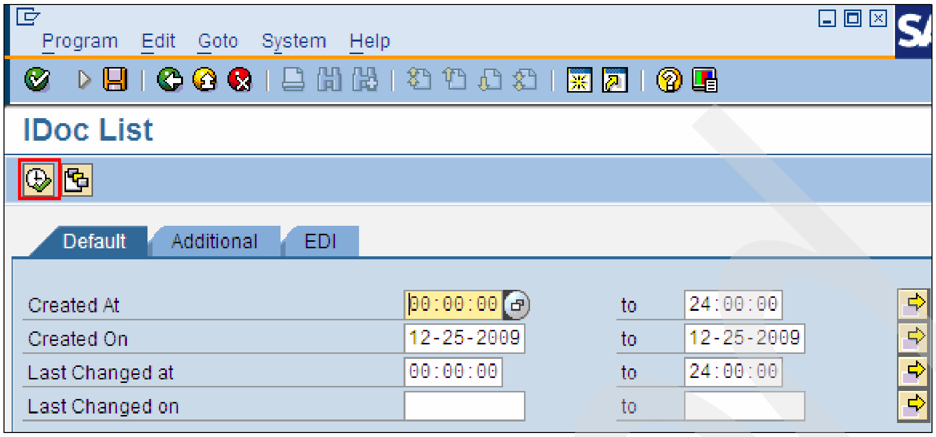
□ Send material in full

Parallel processing

Server group	
Number of materials per proces	20

Figure 2-58 Display IDocs

12. On the IDoc List screen click the **Green checklist** located just below the IDoc List header after you have navigated and gotten familiarized with the screen (Figure 2-59).

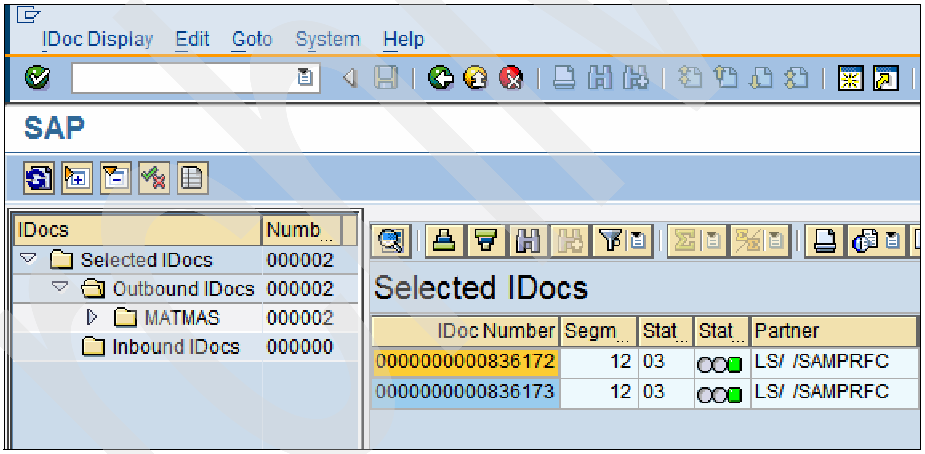


Created At	00:00:00	to	24:00:00
Created On	12-25-2009	to	12-25-2009
Last Changed at	00:00:00	to	24:00:00
Last Changed on		to	

Figure 2-59 IDoc List

13. The IDocs that have been created are listed. Browse and get acquainted with the SAP system.

Double-click IDoc Number **0000000000836173**. This brings you to the next screen(Figure 2-60).



IDocs	Numb...
Selected IDocs	000002
Outbound IDocs	000002
MATMAS	000002
Inbound IDocs	000000

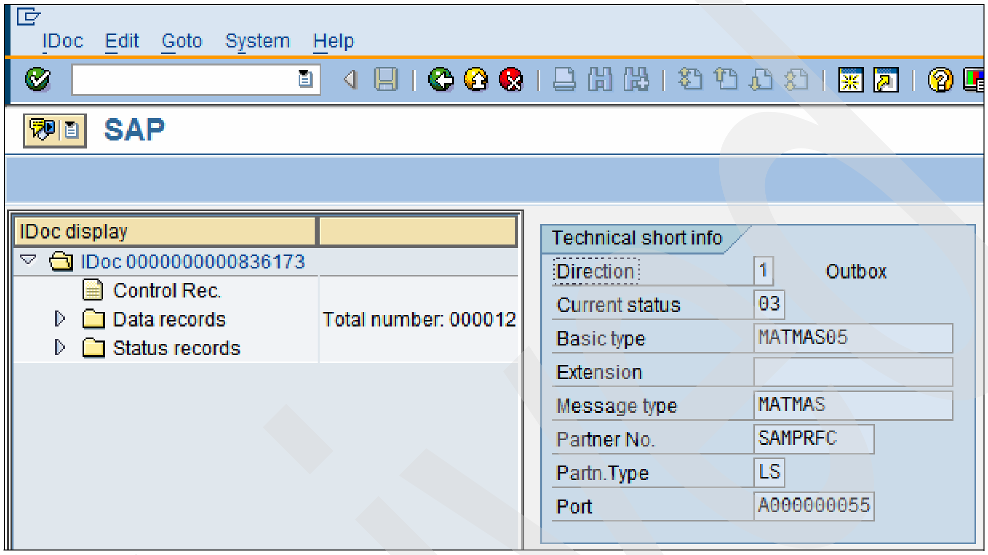
Selected IDocs

IDoc Number	Segm...	Stat...	Stat...	Partner
0000000000836172	12	03	○○■	LS/ /SAMPRFC
0000000000836173	12	03	○○■	LS/ /SAMPRFC

Figure 2-60 Browse selected IDocs

14. On the IDoc Display screen for <0000000000836173> notice the following fields in the Technical short info area (Figure 2-61):

- Direction: 1
- Current status: 03
- Basic type: MATMAS05
- Partner No. SAMPRFC



IDoc display

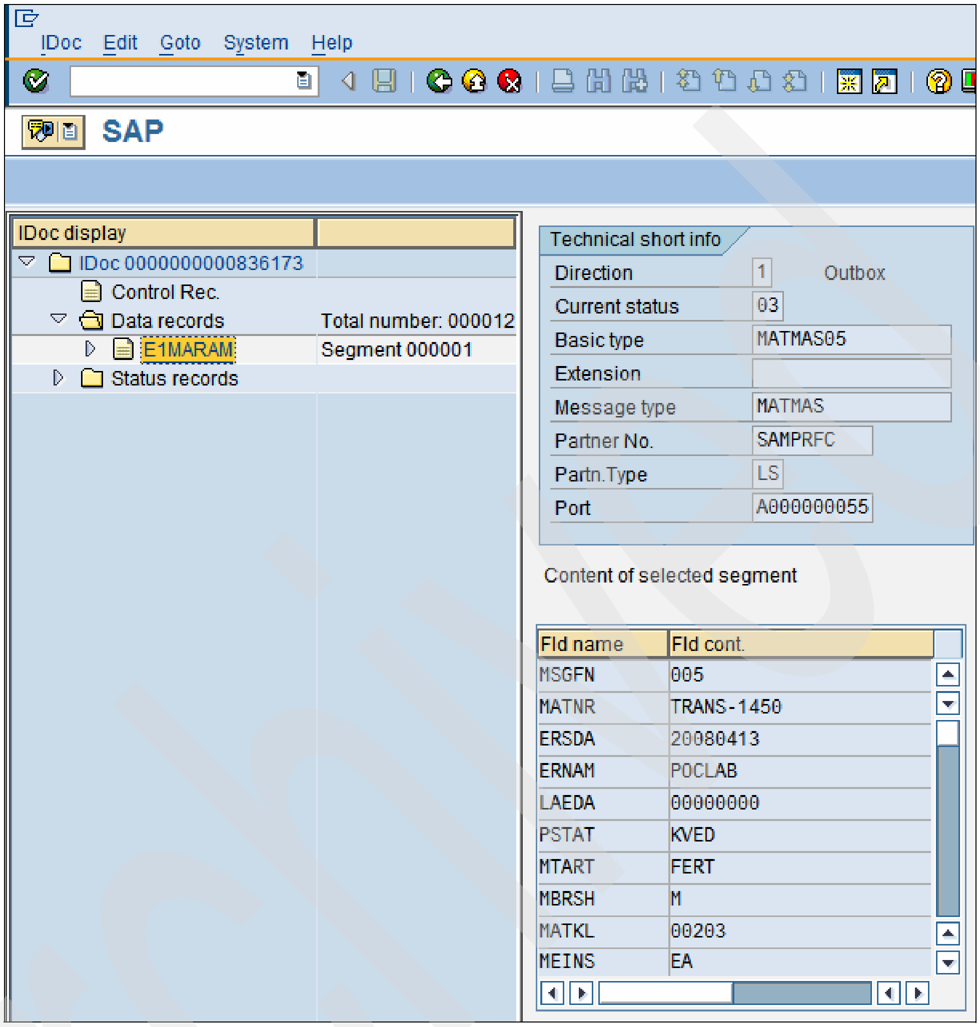
- IDoc 0000000000836173
  - Control Rec.
  - Data records — Total number: 000012
  - Status records

Technical short info

Direction	1	Outbox
Current status	03	
Basic type	MATMAS05	
Extension		
Message type	MATMAS	
Partner No.	SAMPRFC	
Partn.Type	LS	
Port	A000000055	

Figure 2-61 View technical details

15. Explore the Data records. Click **E1MARAM** to access details about the content of selected segment (Figure 2-62).



Direction	1	Outbox
Current status	03	
Basic type	MATMAS05	
Extension		
Message type	MATMAS	
Partner No.	SAMPRFC	
Partn.Type	LS	
Port	A000000055	

Fld name	Fld cont.
MSGFN	005
MATNR	TRANS-1450
ERSDA	20080413
ERNAM	POCLAB
LAEDA	00000000
PSTAT	KVED
MTART	FERT
MBRSH	M
MATKL	00203
MEINS	EA

Figure 2-62 Explore data records

16. Now that you have explored some of the SAP system related to the IDoc creation, it is time to exit the system. Click **System** → **Log off** (Figure 2-63).

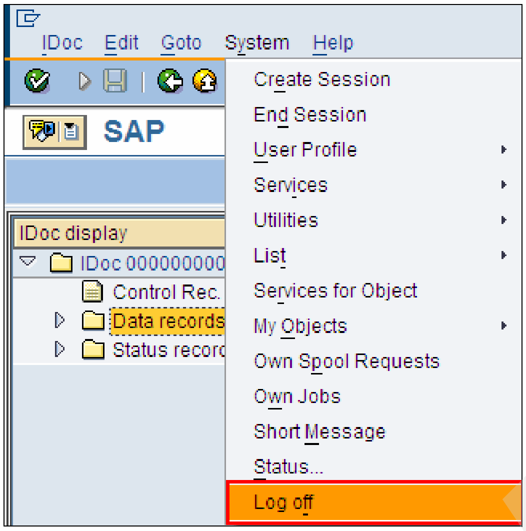
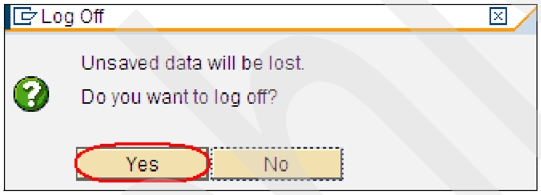


Figure 2-63 Exiting the system

17. Click **Yes** to log off the SAP system (Figure 2-64).



Unsaved data will be lost.

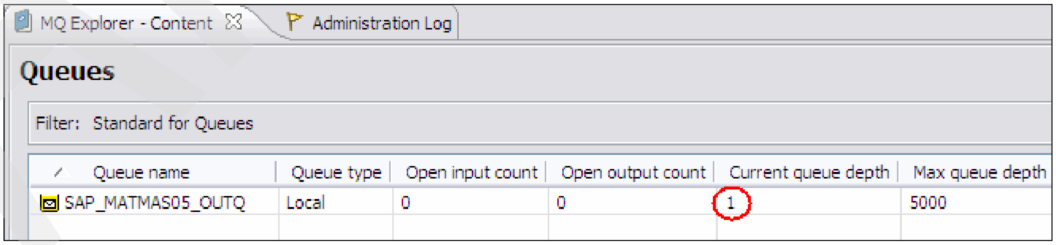
Do you want to log off?

Figure 2-64 Log off SAP system

18. Go back to the IBM WebSphere Message Broker Explorer and look for the newly created record. Select **Queue Managers** → **MB7QMGR** → **Queues** → **SAP_MATMAS05_OUTQ** (Figure 2-65).

Notice that the Current queue depth is 1, which means that the record is been created and is ready to be viewed.

The following steps describe how to use another tool to view the content of the message.



Filter: Standard for Queues

Queue name	Queue type	Open input count	Open output count	Current queue depth	Max queue depth
SAP_MATMAS05_OUTQ	Local	0	0	1	5000

Figure 2-65 Observe the current queue

19. Open **RFHUTIL**. Click the **RFHUtil** icon in the system Quick Launch toolbar (Figure 2-66).

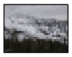


Figure 2-66 RFHUTIL

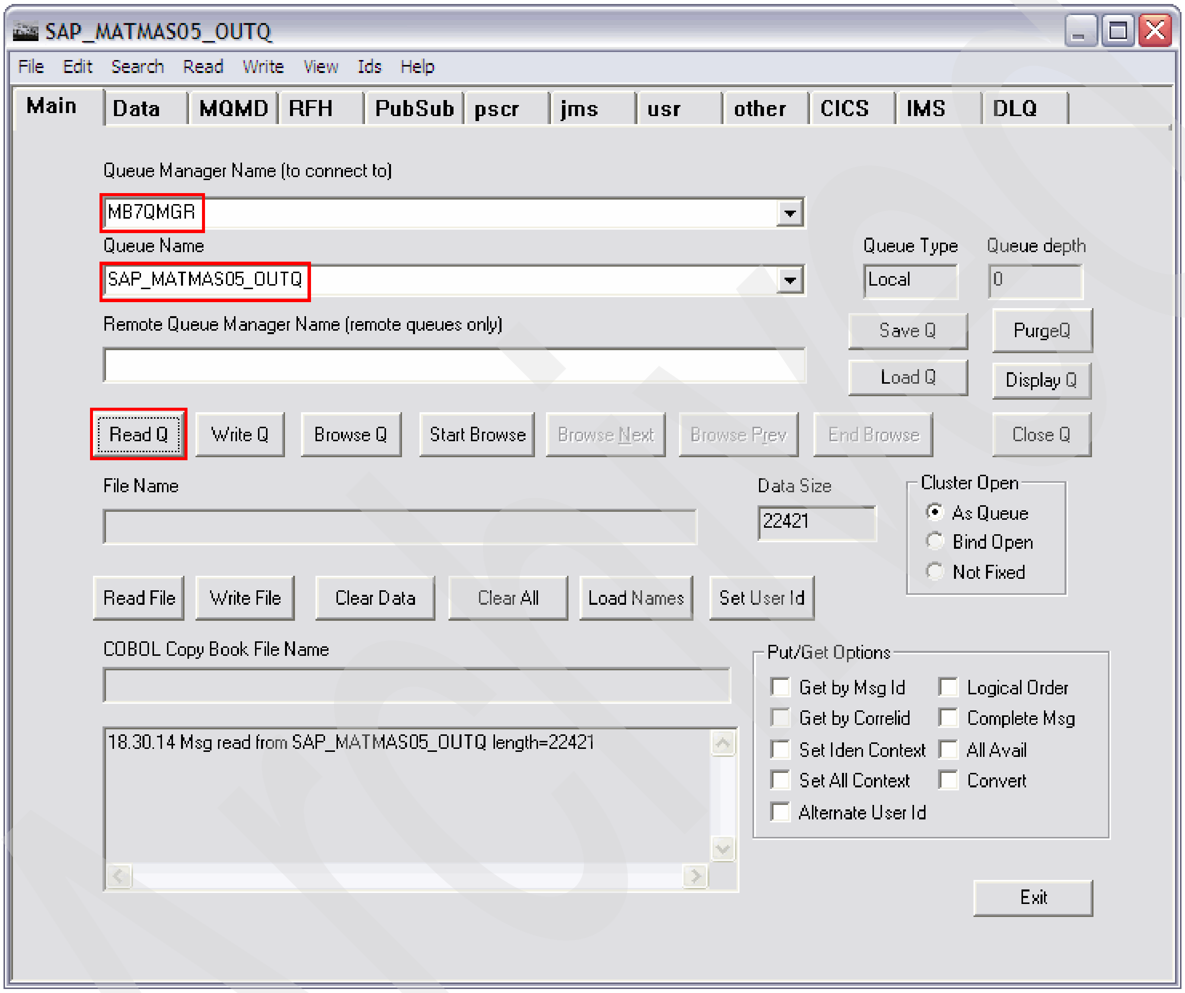
20. On the Main tab select:

- Queue Manager Name (to connect to): **MB7QMGR**
- Queue Name: **SAP_MATMAS05_OUTQ**

Click the **Read Q** button.

You should see the following status message at the bottom of the application panel (Figure 2-67), indicating an IDoc was successfully received from the SAP system:

Msg read from SAP_MATMAS05_OUTQ length=22421


SAP_MATMAS05_OUTQ
File Edit Search Read Write View Ids Help
Main Data MQMD RFH PubSub pscr jms usr other CICS IMS DLQ
Save Q PurgeQ
Load Q Display Q
Read Q Write Q Browse Q Start Browse Browse Next Browse Prev End Browse Close Q
File Name
Read File Write File Clear Data Clear All Load Names Set User Id
COBOL Copy Book File Name
Exit

Queue Manager Name (to connect to): MB7QMGR

Queue Name: SAP_MATMAS05_OUTQ

Queue Type: Local

Queue depth: 0

Remote Queue Manager Name (remote queues only):

Data Size: 22421

Cluster Open: ● As Queue ○ Bind Open ○ Not Fixed

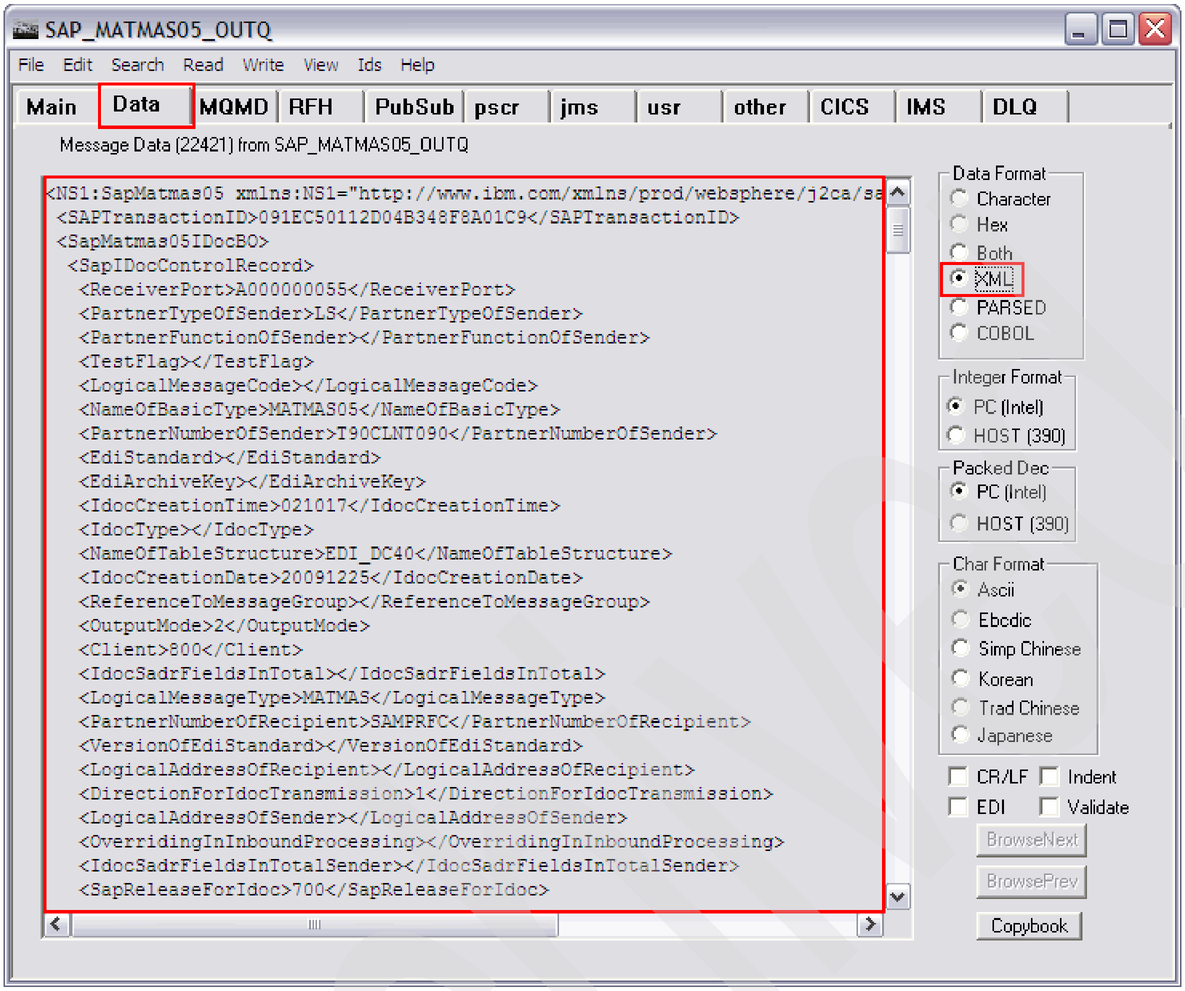
Put/Get Options: □ Get by Msg Id □ Logical Order □ Get by Correlid □ Complete Msg □ Set Iden Context □ All Avail □ Set All Context □ Convert □ Alternate User Id

```
18.30.14 Msg read from SAP_MATMAS05_OUTQ length=22421
```

Figure 2-67 Main tab

21. Click the **Data** tab and select **XML** in the Data Format area (Figure 2-68).

The IDoc MATMAS05 that has been received from SAP is displayed.


SAP_MATMAS05_OUTQ
File Edit Search Read Write View Ids Help
Main Data MQMD RFH PubSub pscr jms usr other CICS IMS DLQ
BrowseNext
BrowsePrev
Copybook

Message Data (22421) from SAP_MATMAS05_OUTQ

```
<NS1:SapMatmas05 xmlns:NS1="http://www.ibm.com/xmlns/prod/websphere/j2ca/sa
 <SAPTransactionID>091EC50112D04B348F8A01C9</SAPTransactionID>
 <SapMatmas05IDocBO>
  <SapIDocControlRecord>
   <ReceiverPort>A000000055</ReceiverPort>
   <PartnerTypeOfSender>LS</PartnerTypeOfSender>
   <PartnerFunctionOfSender></PartnerFunctionOfSender>
   <TestFlag></TestFlag>
   <LogicalMessageCode></LogicalMessageCode>
   <NameOfBasicType>MATMAS05</NameOfBasicType>
   <PartnerNumberOfSender>T90CLNT090</PartnerNumberOfSender>
   <EdiStandard></EdiStandard>
   <EdiArchiveKey></EdiArchiveKey>
   <IdocCreationTime>021017</IdocCreationTime>
   <IdocType></IdocType>
   <NameOfTableStructure>EDI_DC40</NameOfTableStructure>
   <IdocCreationDate>20091225</IdocCreationDate>
   <ReferenceToMessageGroup></ReferenceToMessageGroup>
   <OutputMode>2</OutputMode>
   <Client>800</Client>
   <IdocSadrFieldsInTotal></IdocSadrFieldsInTotal>
   <LogicalMessageType>MATMAS</LogicalMessageType>
   <PartnerNumberOfRecipient>SAMPRFC</PartnerNumberOfRecipient>
   <VersionOfEdiStandard></VersionOfEdiStandard>
   <LogicalAddressOfRecipient></LogicalAddressOfRecipient>
   <DirectionForIdocTransmission>1</DirectionForIdocTransmission>
   <LogicalAddressOfSender></LogicalAddressOfSender>
   <OverridingInInboundProcessing></OverridingInInboundProcessing>
   <IdocSadrFieldsInTotalSender></IdocSadrFieldsInTotalSender>
   <SapReleaseForIdoc>700</SapReleaseForIdoc>
```

Data Format: ○ Character ○ Hex ○ Both ● XML ○ PARSED ○ COBOL

Integer Format: ● PC (Intel) ○ HOST (390)

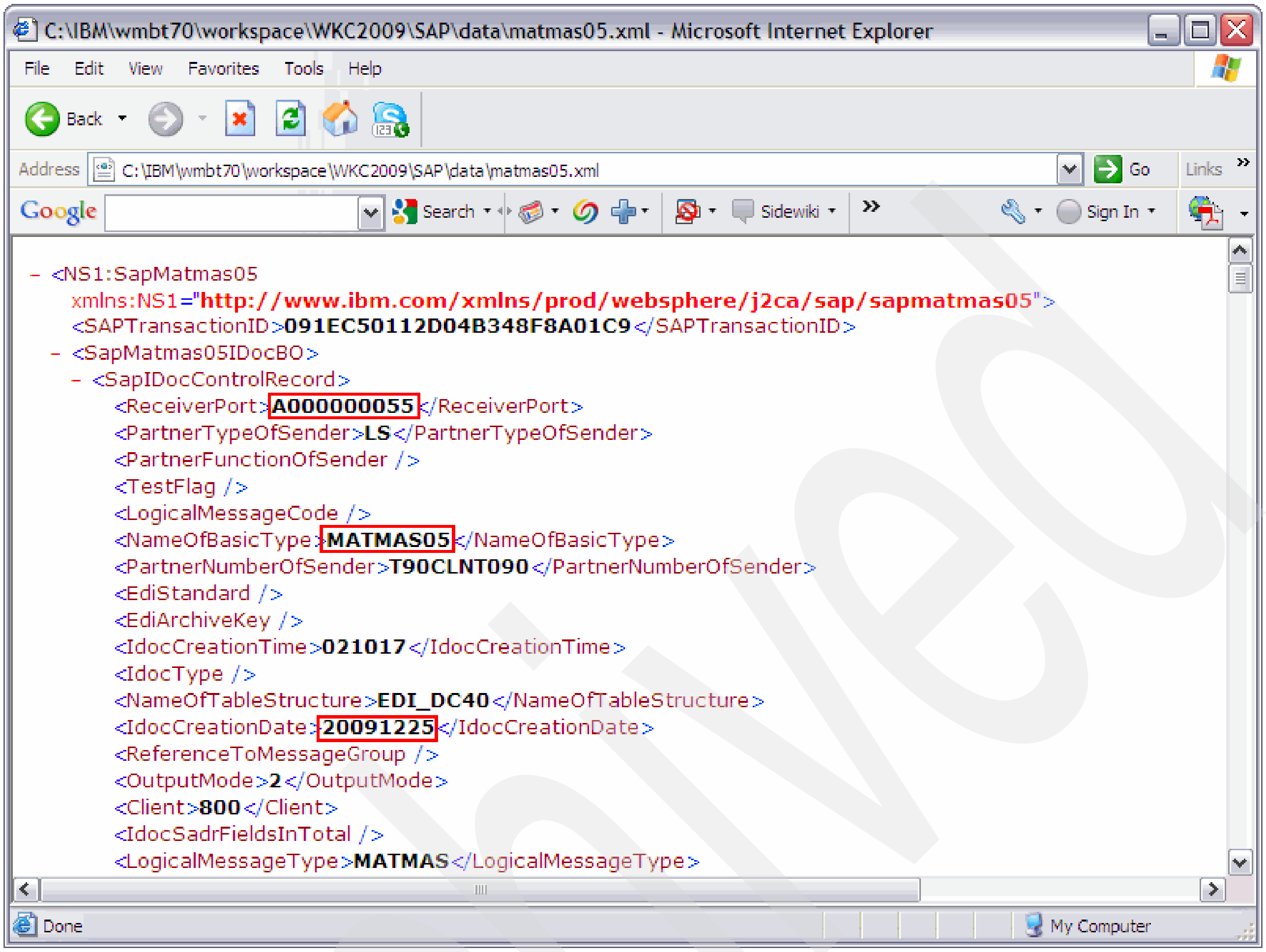
Packed Dec: ● PC (Intel) ○ HOST (390)

Char Format: ● Ascii ○ Ebcdic ○ Simp Chinese ○ Korean ○ Trad Chinese ○ Japanese

□ CR/LF □ Indent □ EDI □ Validate

Figure 2-68 View the Data tab

22.The view from the RFHUTIL tool is not very user friendly, but the content can be copied into a file and viewed from a browser (Figure 2-69).


C:\IBM\wmbt70\workspace\WKC2009\SAP\data\matmas05.xml - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Address
C:\IBM\wmbt70\workspace\WKC2009\SAP\data\matmas05.xml
Go
Links
Google
Search
Sidewiki
Sign In
Done
My Computer

```
- <NS1:SapMatmas05
    xmlns:NS1="http://www.ibm.com/xmlns/prod/websphere/j2ca/sap/sapmatmas05">
    <SAPTransactionID>091EC50112D04B348F8A01C9</SAPTransactionID>
  - <SapMatmas05IDocBO>
    - <SapIDocControlRecord>
        <ReceiverPort>A000000055</ReceiverPort>
        <PartnerTypeOfSender>LS</PartnerTypeOfSender>
        <PartnerFunctionOfSender />
        <TestFlag />
        <LogicalMessageCode />
        <NameOfBasicType>MATMAS05</NameOfBasicType>
        <PartnerNumberOfSender>T90CLNT090</PartnerNumberOfSender>
        <EdiStandard />
        <EdiArchiveKey />
        <IdocCreationTime>021017</IdocCreationTime>
        <IdocType />
        <NameOfTableStructure>EDI_DC40</NameOfTableStructure>
        <IdocCreationDate>20091225</IdocCreationDate>
        <ReferenceToMessageGroup />
        <OutputMode>2</OutputMode>
        <Client>800</Client>
        <IdocSadrFieldsInTotal />
        <LogicalMessageType>MATMAS</LogicalMessageType>
```

Figure 2-69 Viewing data in a browser

2.7 Export the MASM05 project

A project interchange archive is a compressed archive that contains your source projects and potentially, staging projects as well (if you choose to include them, but we recommend not including them).

Perform these steps to export the MASM05 project:

1. In WebSphere Message Broker, select **File** → **Export** to open the Export Wizard (Figure 2-70).

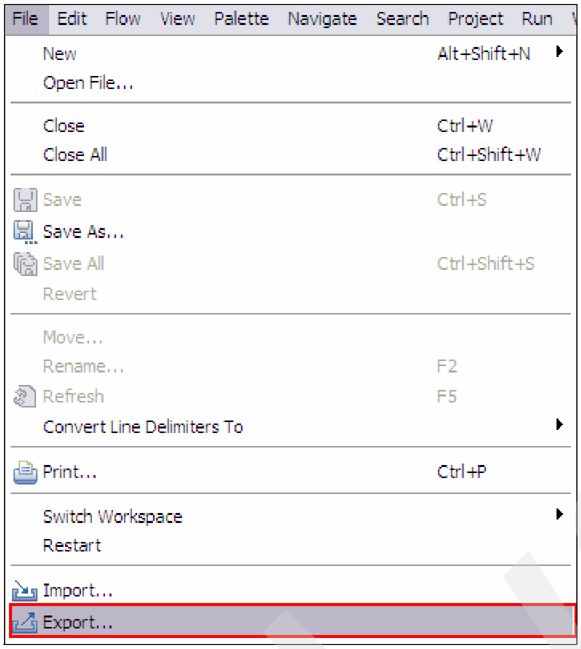
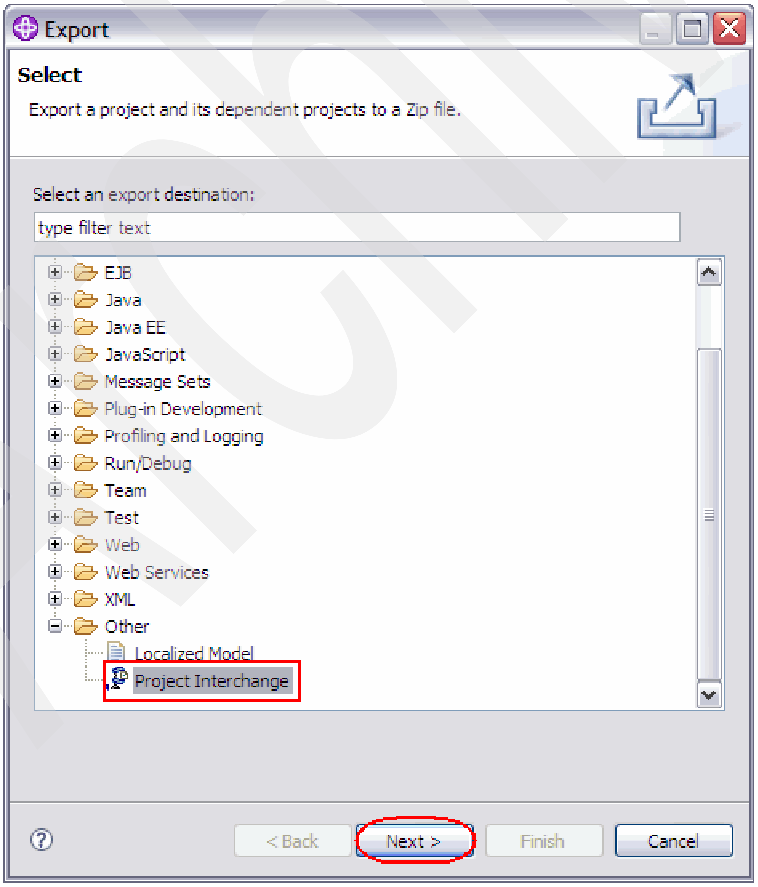


Figure 2-70 Start export

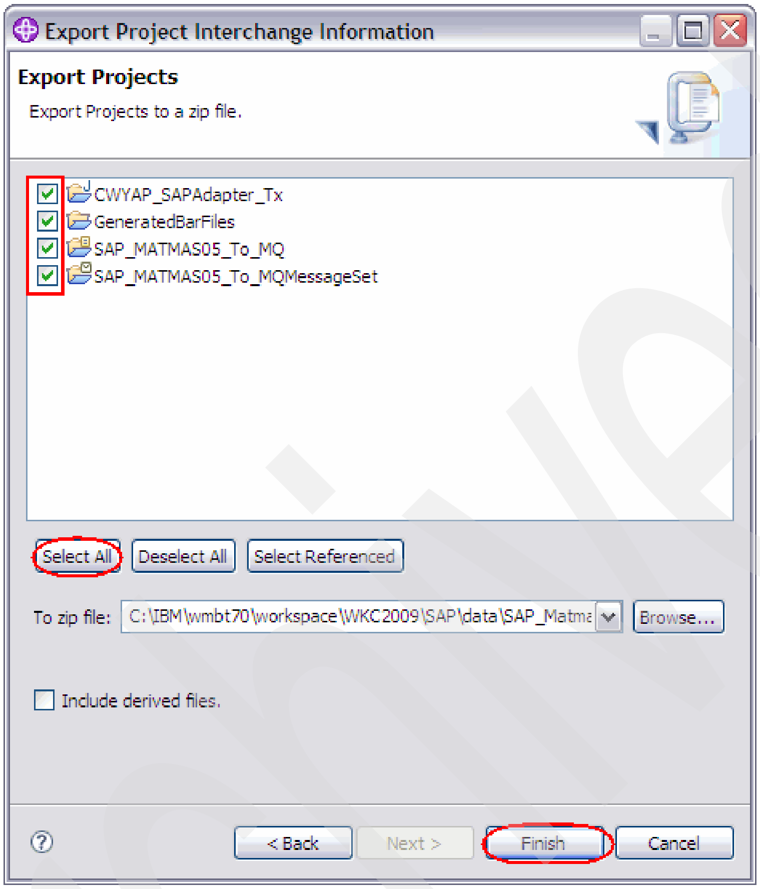
2. Select **Other** → **Project Interchange** and click **Next** (Figure 2-71).



Export a project and its dependent projects to a Zip file.

Figure 2-71 Select Project Interchange

3. In the Export Projects pane click **Select All**.
4. Enter a file name in the To ZIP file field (for example, C:\IBM\wmbt70\workspace\WKC2009\SAP\data\SAP_Matma05_PI.zip).
5. Click **Finish** to create the zip file.



Export Project Interchange Information

Export Projects

Export Projects to a zip file.

☑ CWYAP_SAPAdapter_Tx
☑ GeneratedBarFiles
☑ SAP_MATMAS05_To_MQ
☑ SAP_MATMAS05_To_MQMessageSet

To zip file: C:\IBM\wmbt70\workspace\WKC2009\SAP\data\SAP_Matma

□ Include derived files.

Figure 2-72 Export projects

A

Support information

This appendix provides:

- Additional information resources
- Definitions of key terms
- Target system details

Additional information resources

This section identifies Web sites that provide helpful information:

- SAP Java Connector:

  http://help.sap.com/saphelp_nw04/helpdata/en/6f/1bd5c6a85b11d6b28500508b5d5211/content.htm

- IBM InfoCenter details on installing the SAP Java Connector:

  http://publib.boulder.ibm.com/infocenter/lwm/v1r0/index.jsp?topic=/com.ibm.lwm.doc_1.0/inst_wp_sapjco.html

- Application Link Enabling (ALE) overview:

  http://www.sysdatait.com/ale.htm

- ALE Integration Technology (BC-MID-ALE):

  http://help.sap.com/saphelp_46c/helpdata/en/35/26b542afab52b9e10000009b38f974/content.htm

- From the *Oracle Application Server InterConnect Adapter for SAP R/3 Installation and User's Guide 10g (9.0.4)* - Application Link Enabling section:

  http://www.deakin.edu.au/its/dba/oracle-doco/9.0.4.1/9.0.4_doc_library/integrate.904/b10408/ale.htm

- IBM Redbooks publication *WebSphere MQ V6 Fundamentals*, SG24-7126:

  http://www.redbooks.ibm.com/abstracts/SG247128.html

- SAP Techies: SAP Support Ports - Application Link Enabling (ALE):

  http://www.saptechies.com/application-link-enabling-ale/

- IBM developerWorks® article: *Getting more out of the WebSphere SAP Adapter*

  http://www.ibm.com/developerworks/websphere/tutorials/0906_kumar/index.html

- SAP R/3 Gateway: Configuring an SAP R/3 Distribution Model

  http://documentation.softwareag.com/webmethods/sap231/pages/sapdist.htm

- Generate/Change RFC Connections:

  http://help.sap.com/saphelp_sm32/helpdata/en/b3/dd773dd1210968e10000000a114084/content.htm

- SAP Community Wiki: Establish Single Sign On between Portal (front end) and ESS6.0 (back end):

  http://wiki.sdn.sap.com/wiki/display/EP/8+Establish+Single+Sign+On+between+Portal+(front+end)+and+ESS6.0+(back+end)

WebSphere Message Broker (Samples and Tutorials)

- SAP Connectivity sample
- SAP call out to a synchronous system sample
- SAP callout to an asynchronous system sample

SAP terminology

This section gives you essential information about terminology used with SAP. If you need further help with SAP terminology, see the SAP Library Glossary located at this Web site:

http://help.sap.com/saphelp_46c/helpdata/En/35/2cd77bd7705394e10000009b387c12/frameset.htmhttp://help.sap.com/saphelp_46c/helpdata/En/35/2cd77bd7705394e10000009b387c12/frameset.htm

Key terms to know are:

- Advanced Business Application Programming (ABAP)

  A programming language designed for creating large-scale business applications.

  Outbound calls are supported with the BAPI interface, ALE interface, or Advanced Event Processing (AEP) interface using ABAP handlers and Query Interface for SAP Software (QISS).

- Application Link Enabling (ALE)

  ALE is an integration interface in the SAP Business Framework Architecture. It is a component-based architecture enabling software components from SAP and from other software vendors to communicate and be integrated with each other.

  ALE can integrate business processes between SAP Systems and external applications and between SAP Systems. Application systems are loosely coupled in an ALE integrated system and the data is exchanged asynchronously.

  ALE uses IDoc for data exchange.

- Business Application Program Interfaces (BAPIs)

  BAPIs are programming interface to access SAP Database from within SAP or other development platforms external to R/3 that support the Remote Function Call (RFC) protocol.

  The main objective of BAPI is to achieve integration between the SAP System and external applications, legacy systems, and so on.

  BAPIs are defined in the Business Object Repository (BOR) as methods of SAP Business Objects or SAP Interface Types and enable object-oriented access to Business Components (application components) in the SAP System.

- Enterprise resource planning (ERP)

  A software system for planning and automating enterprise-wide business processes.

- Intermediate document (IDocs)

  A data exchange format used between SAP systems and between SAP systems and external applications. IDocs represent SAP business objects as flat structures, proprietary format defined by SAP for business data transmission; IDocs are used for asynchronous batch data transmission.

- SAP Java Connector (SAP JCo)

  JCo is a toolkit that allows a Java application to communicate with any SAP system. The package supports both, Java to SAP system and SAP system to Java calls.

Target system

The target system can be any one of the following:

- SAP R/3 4.6C (running on Basis 4.6C)
- SAP R/3 4.7 (running on WAS 6.20)

WebSphere Message Broker V7.0 Integration with WebSphere Adapter for SAP Software

Redpaper™

Understand the value of WebSphere Message Broker in SAP environments

Get started with WebSphere Adapter for SAP Software

Send and receive messages from your SAP environment

This IBM Redbooks publication describes how to get started with using WebSphere Adapter for SAP Software with WebSphere Message Broker. These products enable processes and components to be integrated to include the exchange of information with an SAP server, without special coding.

This paper shows how to use the adapter, an application component, to send requests to the SAP server or to receive events from the server. The adapter creates a standard interface to the applications and data on the SAP server so that the developer of the application component does not have to understand the lower level details (the implementation of the application or the data structures) on the SAP server.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

REDP-4644-00