



IBM Software Group

IBM WebSphere® Adapters V6.0.2

WebSphere Adapter V6.0.2 for SAP



@business on demand.

© 2007 IBM Corporation
Updated March 12, 2007

This presentation will focus on the WebSphere Adapter for SAP V6.0.2

Agenda

- SAP terminology
- Overview
- Installation
- Configuration properties
- Enterprise service discovery
- Transaction and security
- Migration
- Summary

The agenda for this presentation is shown here.

Section

SAP terminology

This section will cover SAP terminology.

SAP terminology

- **IDoc (SAP Intermediate Document)**

- ▶ 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

- **SAPJCo:**

- ▶ The SAP Java Connector (SAPJCo) is a toolkit that allows a Java application to communicate with any SAP System
- ▶ The package supports both, Java™ to SAP System and also SAP System to Java™ calls

Shown on this slide and the next two slides is some common SAP terminology that will help you to better understand the SAP adapter.

SAP terminology (cont.)

- **Business Application Program Interfaces (BAPI)**
 - ▶ BAPI's 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

BAPI's are programming interface to access SAP Database from within SAP or other development platforms external to R/3 that support the Remote Function Call protocol

SAP terminology (cont.)

▪ ALE (Application Link Enabling)

- ▶ Application Link Enabling (ALE) is an integration interface in SAP's Business Framework Architecture, 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 also 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

6

WebSphere Adapter V6.0.2 for SAP

© 2007 IBM Corporation

Application Link Enabling (ALE) is an integration interface in SAP's Business Framework Architecture, a component-based architecture enabling software components from SAP and from other software vendors to communicate and be integrated with each other.

ALE uses IDoc for data exchange

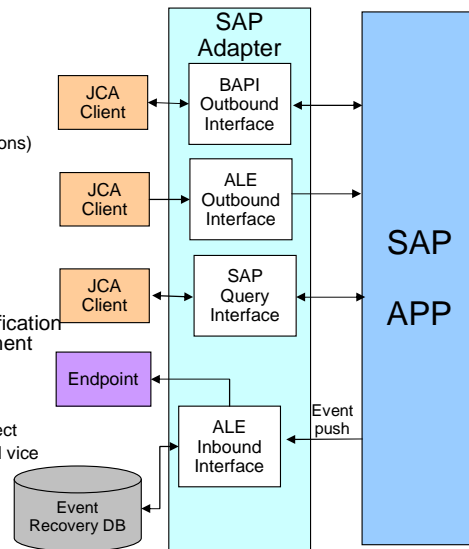
Section

Overview

This section provides an overview of the WebSphere Adapter for SAP.

SAP adapter outbound and inbound support

- Outbound calls supported using the following interfaces:
 - BAPI interface
 - ▶ Single BAPI call
 - ▶ BAPI / RFC
 - ▶ Support multiple BAPI calls in a single interaction (BAPI Transactions)
 - ALE interface
 - ▶ Single IDoc
 - ▶ IDoc packets (collection of IDocs)
 - SQI interface
 - ▶ Retrieve Application table data
- Inbound calls with ALE only using Asynchronous event notification
 - Event Recovery table is used for inbound event management
- SAP functions are modeled as Business Object
 - ▶ SCA clients wrap the Business Object within J2C CCI Record object
 - ▶ Metadata in Business Object used to make SAP function calls and vice versa



8

WebSphere Adapter V6.0.2 for SAP

© 2007 IBM Corporation

The diagram on this page shows the high level flow and the components for the outbound and Inbound calls to and from SAP. The outbound calls are supported with the BAPI interface, ALE interface and SAP Query Interface (SQI). The SQI is new interface that is added in V6.0.2 .

The Adapter uses the SAP Component Interface JAR file to communicate with the SAP application.

With BAPI, they can be simple BAPI calls, or BAPI using Remote function call (RFC) or can be multiple BAPI calls in a single interaction, referred to as BAPI transactions. BAPI outbound calls have request and response interaction style. The ALE interface supports passing single or multiple IDocs. These are only one-way calls where the IDoc(s) are passed to the SAP application.

With SQI interface you can directly query the SAP application tables.

The inbound calls are supported by the ALE interface only using Asynchronous event notification. For the inbound call, the Adapter acts as an RfcServer and listens for ALE events from the SAP Application. The adapter uses an event recovery table to manage the inbound events. More details on outbound and inbound calls are discussed later in this presentation. Similar to other WebSphere Adapters, SAP functions are modeled as Business Objects. The SCA clients wrap the Business Object with the J2C CCI Record object. The parameters and details of the SAP function call are contained as metadata within the Business Object.

Section

Installation

This section will provide the steps in installation and deployment of the WebSphere Adapter for SAP. Note that this section is also covered in a separate presentation common for all WebSphere Adapters.

Importing the adapter

- Import SAP Adapter RAR into WebSphere Integration Developer – this creates a J2EE connector RAR project in the workspace
- Add EIS external dependencies, “sapjco.jar”, to the Java Build path of the Connector project
 - ▶ For WebSphere Process Server: <WPS_INSTALL>\lib directory
 - ▶ For WebSphere Integration Developer: add the to Resource Adapter project’s Java Build Path
- Add the external SAP native libraries required for the adapter
 - ▶ For WebSphere Process Server: <WPS_INSTALL>\bin directory
 - ▶ For WebSphere Integration Developer: <WID_INSTALL>\eclipse\jre\bin
 - ▶ Libraries are:
 - librfc2.dll (or .so)
 - MSVCP71.dll (or .so)
 - Msvcr71.dll (or .so)
 - Sapjcorfc.dll (or .so)

Import the RAR file into your workspace using the WebSphere Integration Developer tool. This creates a J2EE connector project in your workspace. Add the EIS external dependency, “sapjco.jar”, to the java build path of the RAR project. You also need to add external SAP native libraries to the connector project class path.

Deployment in WebSphere Process Server

- Add the SAP dependent jar files and DLLs to the WebSphere Process Server class path
 - ▶ sapco.jar: <WPS_INSTALL>\lib directory
 - ▶ DLLs: <WPS_INSTALL>\bin directory
- Install EAR file in WebSphere Process Server using the Administrative Console or command line tool “wsadmin”
 - ▶ For most install steps, defaults are fine
- If needed, modify any RAR properties (from Your application -> Connector Modules -> CWYEP_SAPAdapter.rar)
 - ▶ Custom RAR properties
 - ▶ Managed Connection Factory for outbound connection to EIS
 - ▶ Activation Spec for inbound event from EIS
- Start the Enterprise application using Administrative console or wsadmin

This page outlines the high level flow of deploying the business process application built in WebSphere Integration Developer to a WebSphere Process Server. Adapter's external dependencies must be satisfied. The J2C authentication alias for the outbound request must be specified. The database that holds the Event table must be created for adapters that require them. SAP Adapter requires it, whereas for the other adapter, the table is optional.

Add necessary SAP dependent jar files and DLLs to the WebSphere Process class path. Installation of the application containing the WebSphere Adapter for SAP to the WebSphere Process Server is similar to installing any other enterprise application. Either the administrative console or command line tool, wsadmin, can be used for application installation. Once installed, the administrator can modify the adapter properties. Last task is to start the application, either in the console or through wsadmin.

Section

Configuration SAP properties

Common properties were covered In the WebSphere Adapter common details presentation. This section covers the SAP specific attributes. More details on these properties can also be found in the WebSphere Adapter Information Center. The link is provided in the Summary and references section.

Managed connection factory properties (for outbound)

Configuration Property	Details
Client	SAP Client number under which the resource adapter logs in, often 100
Username	User account on the SAP application.
Password	Password
Language	The default is E , for English.
ApplicationServerHost	When configuring the resource adapter to run without load balancing, specifies the IP address or the name of the application server that the resource adapter logs in to. In both cases, the resource adapter assumes that the name of the gateway host is the same as the value specified for this property.
SystemNumber	System number of the application server. The value is a two-digit number, often 00 - Default is 00.
GatewayHost	Host where the gateway service is running.
GatewayService	Gateway server identifier; often sapgw00. The 00 is the system number of the server running the SAP Gateway (typically an application server) and may not be 00 if you have more than one. The default is sapgw00.

Managed Connection Factory configuration properties are used at run time to create an outbound connection instance with an enterprise information system. Once the Managed Connection Factory properties are created, they are stored in the deployment descriptor. These properties are essential for SAP applications. ApplicationServerHost property specifies the IP address or the name of the application server that the adapter logs in to. GatewayHost and GatewayService are needed to identify where the gateway service is running.

Managed connection factory properties (for outbound) – cont.

Configuration Property	Details
Group	When configuring the resource adapter for load balancing, specifies the name of the logon group that represents a group of application servers.
MessageServerHost	When configuring the resource adapter for load balancing, specifies the name of the message server.
SAPSystemID	When configuring the resource adapter for load balancing, specifies the logical name of the SAP system, which is also known as R3name.
ABAPDebug	This property specifies whether the resource adapter invokes the ABAP Debugger for the appropriate function module when the resource adapter begins processing a business object. When this property is set to true, the resource adapter opens the ABAP Debugger. A dialog user with proper user authorizations is required for debugging. You can add breakpoints only after the debugger opens. Important: This property should always be set to false in a production environment. The default value is false.
PartnerCharset	This property is used to specify a PartnerCharset encoding. When populated with an encoding, it will be used. Otherwise the default encoding will be used. To use the default encoding, leave this property value blank
RFCTraceOn	This property specifies whether or not to generate a text file detailing the RFC activity for each listener thread. You can specify a value of true or false. A value of true activates tracing, which generates a text file. It is recommended that you use these text files in a development environment only, because the files can grow rapidly. The default is false.

This table shows additional managed connection factory configuration properties for outbound request. The ABAP Debugger is available when ABAPDebug property is set to true. The adapter will invoke the ABAP Debugger for the appropriate function module when the adapter begins processing a business object. When configuring the resource adapter for load balancing, the Group property specifies the name of the logon group that represents a group of application servers.

Activation specification properties (for inbound) – connection properties

Property	Description
GatewayHost	SAP Gateway Host where the gateway service is running.
GatewayService	Gateway server identifier; often sapgw00 . The 00 is the system number of the server running the SAP Gateway (typically an application server) and may not be 00 if you have more than one. The default is sapgw00.
RfcProgramID	Program identifier that the RFC Server program will register under
Client	SAP Client number under which the resource adapter logs in, often 100
Username	Name of the resource adapter's user account on the SAP application.
Password	Password for the resource adapter's user account on the SAP application.
Language	Language in which the resource adapter logs in. The default is E , for English.
ApplicationServer Host	When configuring the resource adapter to run without load balancing, specifies the IP address or the name of the SAP application server

Enterprise service discovery connection properties include outbound and inbound connection properties required for performing metadata discovery and bidirectional configuration. You configure these properties using the enterprise service discovery wizard when you initially deploy the adapter.

Activation specification properties - connection properties (cont.)

Property	Description
SystemNumber	System number of the application server. The value is a two-digit number, often 00. The default is 00.
Group	When configuring the resource adapter for load balancing, specifies the name of the logon group that represents a group of application servers.
MessageServerHost	When configuring the resource adapter for load balancing, specifies the name of the message server.
SAPSystemID	When configuring the resource adapter for load balancing, specifies the logical name of the SAP system, which is also known as R3name.
PartnerCharset	This property is used to specify a PartnerCharset encoding. When populated with an encoding, it will be used. Otherwise the default encoding will be used. To use the default encoding, leave this property value blank
RFCTraceOn	This property specifies whether or not to generate a text file detailing the RFC activity for each listener thread. You can specify a value of true or false. A value of true activates tracing, which generates a text file. It is recommended that you use these text files in a development environment only, because the files can grow rapidly. The default is false.

16

The table shows additional Activation Specification configuration properties for inbound calls. They include necessary properties to make a connection to the SAP system, such as the system number of the application server, logon group, message server host, and so on.

Activation specification properties for error recovery table

Property	Description
EP_DataSource_JNDIName EP_CreateTable EP_TableName EP_UserName EP_Password EP_SchemaName	Event Recovery Table information for inbound events. SAP does not have its own event recovery mechanism. Hence, persistent event recovery table is required for SAP adapter.

Here is the highlight of Activation Specification configuration properties for inbound calls, related to the Event recovery table. Since SAP does not have its own event recovery mechanism, the Persistence Event table is required. These are properties required for inbound events.

Activation specification properties for IDoc processing

Property	Description
IgnoreIDocPacketErrors	<p>If adapter encounters an error while processing the IDoc packet, it can behave in two different ways depending upon this configuration property.</p> <p>When this property is set to false adapter will stop processing further IDocs in that packet and report error to the SAP application.</p> <p>When this property is set to true adapter logs an error and continues processing the rest of the IDocs in that packet.</p>

The table shows more Activation Specification configuration properties for inbound calls related to how the IDoc is to be processed by the adapter. If the adapter encounters an error while processing the IDoc packet, it can behave in one of two ways, depending on the IgnoreIDocPacketErrors configuration property. If the IgnoreIDocPacketErrors property is set to false, the adapter stops processing any additional IDocs in the packet and reports errors to the SAP system. If the IgnoreIDocPacketErrors property is set to true, the adapter logs an error and continues processing the rest of the IDocs in the packet. The status of the transaction is marked 3 (InProgress). In this case, the adapter log shows the IDoc numbers that failed, and you must resubmit those individual IDocs separately. You must also manually maintain these records in the event recovery table.

Activation specification properties for ALE properties

Property	Description
AleUpdateStatus	Specifies whether an audit trail is required for all message types. This property must be set to true to cause the connector to update a standard SAP status code after the ALE Module has retrieved an IDoc object for event processing.
AleSelectiveUpdate	Specifies which IDoc Type and MessageType combinations are to be updated when the connector is configured to update a standard SAP status code. Valid only if AleUpdateStatus=True The syntax for this property is: <i>IDocType:MessageType[,IDocType:MessageType [,...]]</i> where a colon (:) delimiter separates each IDoc Type and MessageType, and a comma (,) delimiter separates entries in a set. Example below illustrates two sets. In the example, MATMAS03 and DEBMAS03 are the IDocs, and MATMAS and DEBMAS are the message types: MATMAS03:MATMAS,DEBMAS03:DEBMAS
AleStatusMsgCode	If required, specifies the message code to use when the connector posts the ALEAUD Message IDoc (ALEAUD01). Configure this message code in the receiving Partner Profile. Valid only if AleUpdateStatus=True

This table shows more Activation Specification configuration properties for inbound calls, related to ALE update status that is sent back to the SAP application for monitoring purposes. To monitor your IDoc processing, you can configure the adapter to update the IDoc status by setting the ALEUpdateStatus to true. Once it is true, ALESelectiveUpdate property specifies which IDoc type and MessageType combinations are to be updated. Finally, ALEStatusMsgCode will specify the message code to use when the adapter posts the Message IDoc.

Activation specification properties for ALE properties (cont.)

Property	Description
AleSuccessCode	Specifies the success status code for Application Document Posted. You must specify a value for this property (52 or 53) to cause the connector to update the SAP success status code after the ALE Module has retrieved an IDoc object for event processing. SAP converts this value to status 41 (Application Document Created in Receiving System).
AleFailureCode	Specifies the status code for dispatch failure. You must specify a value for this property (68 or 58) to cause the connector to update the SAP failure status code after the ALE Module has retrieved an IDoc object for event processing. SAP converts this value to 40.
AleSuccessText	Specifies the descriptive text for successful Application Document Posted. Specifying a value for this property is optional, even when you set AleUpdateStatus to true.
AleFailureText	Specifies the descriptive text for dispatch failure. Specifying a value for this property is optional, even when you set AleUpdateStatus to true.

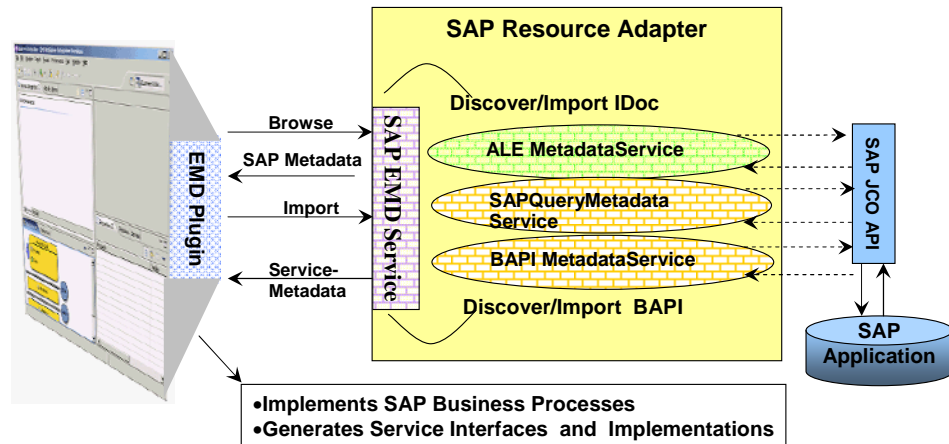
The table shows more Activation Specification configuration properties for inbound calls, related to ALE update status that is send back to the SAP application for monitoring purposes.

Section

Enterprise metadata discovery

This section will cover Enterprise Service Discovery.

EMD tool and SAP EMD service interaction



This diagram shows the interaction between the Enterprise Metadata (Service) Discovery (EMD) tool in WebSphere Integration Developer and the EMD support classes within the SAP adapter. It also shows the end to end flow of interaction from the tool to the SAP adapter and to the SAP application to create the artifacts for the outbound and inbound requests.

The EMD tool working with the EMD classes in the adapter will do the following steps:

- Discover and import SAP metadata definitions and other artifacts related to BAPI and ALE interfaces.
- Generate Business Object for each method in the business service selected. (Business Objects will contain properties that correspond to the method arguments. Application specific information is set in the Business Objects).
- Generate EIS Import (for outbound) or Export (for inbound) SCA components that can then be wired to other business components and create your business process.
- Provide connection descriptions for the inbound and outbound calls.

Based on whether BAPI ,ALE or SQL is selected, the appropriate functions with the adapter are invoked to introspect the SAP application for the available functions. Using the EMD tool in WebSphere Integration Developer, you can introspect the SAP application through the metadata that the adapter will fetch from the SAP application.

Once the necessary function and interaction style (outbound or inbound) is selected, the EMD tool will generate the artifacts for the outbound or inbound call. Generated artifacts include the EIS SCA export or import components, the Business graph and business objects, the interfaces defined by WSDL and other artifacts needed for SCA clients to send an outbound or receive inbound requests.

Section

Transaction and security

The next section covers the transaction and security support.

Transaction and security

Transaction

- ▶ SAP application support transaction for BAPI calls only
- ▶ Adapter provides local transaction support for the BAPI interface using the following BAPI calls
 - BAPI_TRANSACTION_COMMIT
 - BAPI_TRANSACTION_ROLLBACK
- ▶ Adapter provides local transaction support for the ALE interface when using the transactional .rar file
 - JCA client needs to handle the persistence of Transaction ID

Security

- ▶ Adapter supports container managed sign-on and basic authentication. It does not support re-authentication
- ▶ For outbound or inbound, use the WebSphere Process Server J2C Authentication Alias to specify user id and password to connect to the EIS

SAP application supports transaction for BAPI calls only through the use of BAPI_TRANSACTION_COMMIT and BAPI_TRANSACTION_ROLLBACK calls. In this version, the adapter also provides local transaction support for the ALE interface when using the transactional RAR file. The JCA client will need to determine how to store the SAP transaction ID and how to relate the SAP transaction ID with the data being sent to the adapter. If the JCA client does not send a SAP transaction ID with the business object, the adapter will return one after executing the transaction. If the JCA client has a SAP transaction ID, it needs to populate the SAP transaction ID property with that value before executing the transaction.

On the security front, for the outbound or inbound request, the J2C Authentication Alias within the WebSphere Process Server can be used. The authentication alias name can be specified in WebSphere Integration Developer. The administrator must have that J2C Authentication Alias defined within the Process Server with its user id and the password to authenticate the SAP application. Pre-defined J2C Authentication Alias “**SCA_Auth_Alias**” can be used for authentication.

Section

Migration

The next section covers the details on migrating to WebSphere Adapter for SAP V6.0.2

Migration

- Import the Application EAR into WebSphere Integration Developer if it is not already imported.
- Replace the Adapter
 - Delete the connector project that is used by the application. Import the new SAP Adapter RAR file and add the third-party dependencies and add new imported adapter project into your EAR application.
- Test the Application and re-deploy
- Migrating Event Recovery Table
- Event recovery feature has changed. For event recovery mechanism adapter now uses data source configured through administrative console
 - Option 1:
 - Configure new datasource that the adapter would use for ALE event recovery.
 - Stop new events from being triggered and complete the existing flows for inbound ALE events.
 - Upgrade the adapter and start the application.
 - Option 2:
 - Configure new datasource that the adapter would use for ALE event recovery. Create the event recovery table in the data source.
 - Transfer entries from the existing event recovery table to the new event recovery table. You may want to use an existing tool or write scripts to achieve this.
 - Use the following mapping between earlier table and the new event recovery table.

Event Persistence Table Column	Column Type and Size	Deprecated Table Column	Deprecated Type and Size	Column
EVNTID	VARCHAR(255)	TID	VARCHAR(255)	
EVNTSTAT	INTEGER	Status	VARCHAR(255)	
XID	VARCHAR(255)	--- NA ---	-- NA --	
BQTOTAL	INTEGER	NumIDocs	INTEGER	
--- NA ---		NumIDocsProcessed	INTEGER	
BQPROC	INTEGER	CurIDoc	INTEGER	
EVNTDATA	VARCHAR(255)	--- NA ---	-- NA --	

26

WebSphere Adapter V6.0.2 for SAP

© 2007 IBM Corporation

Event recovery feature has changed in V602; the adapter now uses data source configured through the WebSphere Process Server administrative console. For data source configuration, see the WebSphere Process Server documentation.

You have two options for migrating the event recovery table:

Option 1:

- First, Configure the new data source that the adapter will use for ALE event recovery. The adapter can automatically create event-recovery tables for the configured data source database if the activation spec property “EP_CreateTable” is set to true. If “EP_CreateTable” activation spec property is set to false, adapter will not create the event recovery table, you must create the event recover table manually or that table must already exist.
- Next, Stop new events from being triggered and complete the existing flows for inbound ALE events.
- Finally, Upgrade the adapter and start the application.

Option 2:

- First, Configure the new data source that the adapter will use for ALE event recovery, and create the event recovery table in the data source.
- Next, Transfer entries from the existing event recovery table to the new event recovery table. You may want to use existing tool or write scripts to achieve this.
- Finally, Use the table shown on the bottom right for mapping properties between the previous table and the new event recovery table.

Migration

- In the imported application EAR, update the export service description as follows (*this can be achieved by modifying <InterfaceName>.export file using the editor in WebSphere Integration Developer*).
- Delete the following configurations.
 - ▶ EDTDatabaseName
 - ▶ EDTDriverName
 - ▶ EDTSchemaName
 - ▶ EDTTableName
 - ▶ EDTUserName
 - ▶ EDTUserPassword
- Add the following configuration properties and set proper values for them.
 - ▶ AssuredOnceDelivery
 - ▶ EP_DataSource_JNDIName
 - ▶ EP_SchemaName
 - ▶ EP_TableName
 - ▶ EP_UserName
 - ▶ EP_Password

For the existing application EAR, you can update the export service description by deleting some configurations which are now deprecated. They are listed here. Then add the listed configuration properties and set proper values for them.

Section

Summary and references

The next section covers the summary and references.

Summary and references

▪ Summary

- ▶ Discussed SAP Adapter in details covering the architecture, Enterprise Service Discovery, configuration properties
- ▶ Looked at transaction and security

▪ References

- ▶ Information Center:

http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/topic/com.ibm.wsadapters602.doc/welcome_wsa602.html

- ▶ User Guide
- ▶ IBM Education Assistant

In summary, this presentation covered the details of WebSphere Adapter for SAP including the architecture, its enterprise service discovery and configuration properties. The presentation has also provided information on transaction and security. Last but not least, more information on WebSphere Adapter for SAP can be found in the User Guide and the Information Center for the Adapter.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

[Click to send e-mail feedback](#)



You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

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

IBM WebSphere

J2EE, Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

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

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.