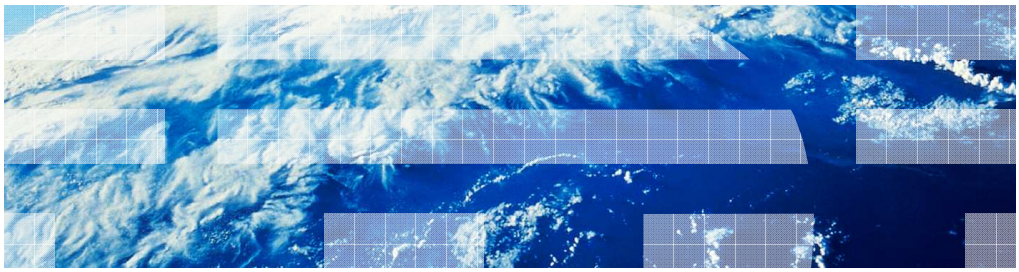
IBM WebSphere Adapters V7.0

WebSphere Adapter for SAP – Overview



This presentation will focus on the WebSphere® Adapter for SAP V7.0 overview.

Table of contents

The agenda for this presentation is shown here.

Section

SAP terminology

This section will cover SAP terminology.

SAP terminology (1 of 3)

- **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 SAP system to Java calls

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

SAP terminology (2 of 3)

- **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.

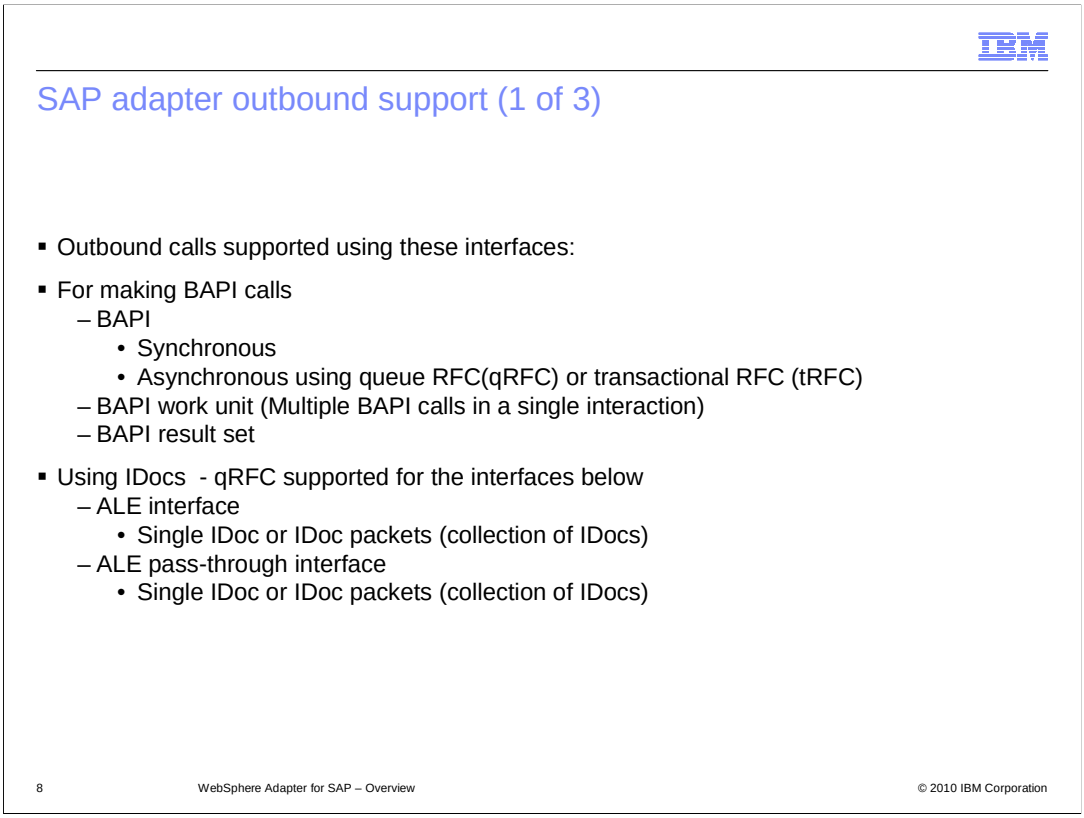
SAP terminology (3 of 3)

- **ALE (Application Link Enabling)**
  - an integration interface in SAP's Business Framework Architecture.
  - 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
  - uses IDoc for data exchange

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.

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



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

With BAPI, outbound can be simple BAPI calls, or BAPI using remote function calls, or multiple BAPI calls in a single interaction, referred to as a BAPI work unit. BAPI outbound calls have request and response interaction style. Using the BAPI result set interface, you can configure the adapter to return combination of output from two different BAPI calls. The adapter also supports transactional and queued RFC support for BAPI calls. With the Advanced Event processing interface, the adapter makes use of the Advanced Business Application Programming (ABAP) handlers.

With the QISS interface, you can directly query the SAP application tables.

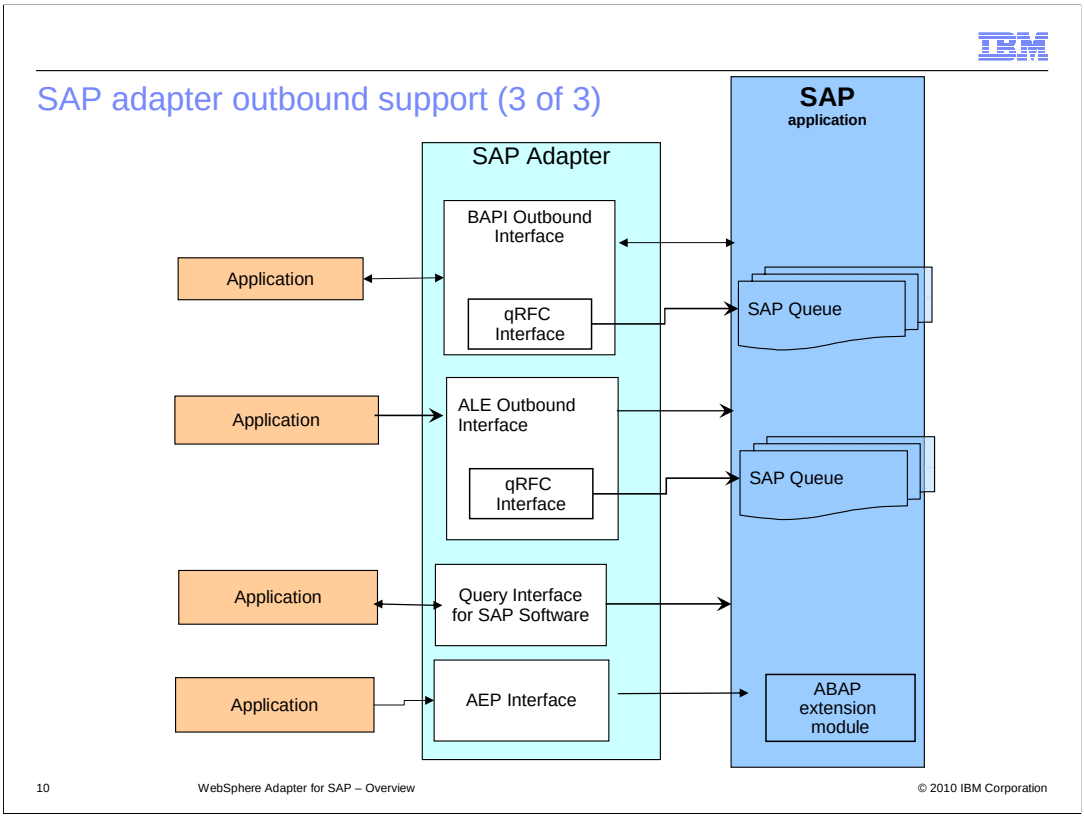
SAP adapter outbound support (2 of 3)

- Outbound calls supported using these interfaces (continued)
- Using IDocs - qRFC supported for the interfaces below
  - ALE interface
    - Single IDoc or IDoc packets (collection of IDocs)
  - ALE pass-through interface
    - Single IDoc or IDoc packets (collection of IDocs)
- Advanced event processing (AEP) interface
  - Custom IDocs and ABAP handlers
- Query interface for SAP software (QISS) interface
  - Retrieve application table data

The ALE interface supports passing single or multiple IDocs, which are one-way calls where the IDocs are passed to the SAP application.

With the Advanced Event processing interface, the adapter makes use of the Advanced Business Application Programming (ABAP) handlers.

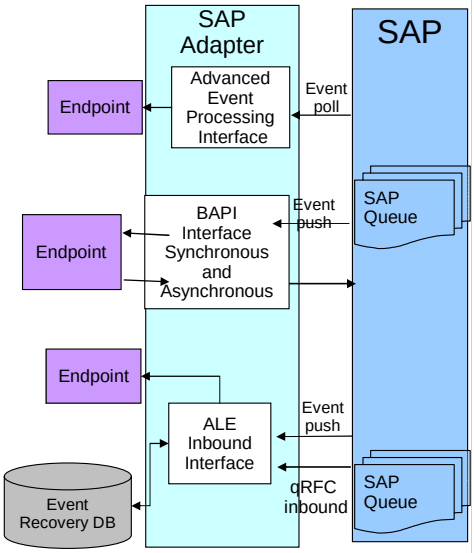
With the QISS interface, you can directly query the SAP application tables and retrieve the data in the hierarchical form you define.



This diagram shows the high level flow and the components in outbound calls. The various interfaces that the adapter supports for outbound are shown in the diagram above. You can use the adapter to make BAPI calls, exchange data using IDocs over ALE interface, interact directly with application tables using query interface for SAP software interface or can call ABAP handlers when using the advanced event processing interface.

SAP adapter inbound support

- Inbound calls supported using these interfaces:
- Using IDocs - qRFC supported for the interfaces below
  - – ALE interface
    - Single IDoc or IDoc packets (collection of IDocs)
  - – ALE pass-through interface
    - Single IDoc or IDoc packets (collection of IDocs)
- BAPI interface
  - – Synchronous
  - – Asynchronous using tRFC
  - – Asynchronous using qRFC
- Advanced event processing interface
  - – For processing custom IDocs



This diagram shows the high level flow and the components for the inbound events from SAP.

Inbound calls are supported by the ALE interface using asynchronous event notification. For the inbound call, the Adapter acts as an RFC server and listens for ALE events from the SAP Application and the adapter uses an event recovery table to manage the inbound events.

For ALE outbound and inbound operations qRFC is supported for the ALE interface in V6.0.2.1. Client applications can specify a queue to which IDocs are delivered, to ensure the order in which the IDocs are delivered and processed by an SAP application.

The Synchronous call back interface introduced in V6.0.2.1 is used to monitor any events related to the invocation on BAPI call on. In V6.2 synchronous call back interface has been renamed to BAPI interface. The adapter supports transactional RFC (tRFC) and queue RFC (qRFC). So adapter now supports both synchronous and asynchronous processing of events. More information on tRFC and qRFC support is covered in the inbound presentation.

With the Advanced Event Processing interface, the adapter processes any events related to custom IDocs using the Advanced Business Application Programming (ABAP) handlers.

More details on outbound and inbound calls are discussed in a different presentation.

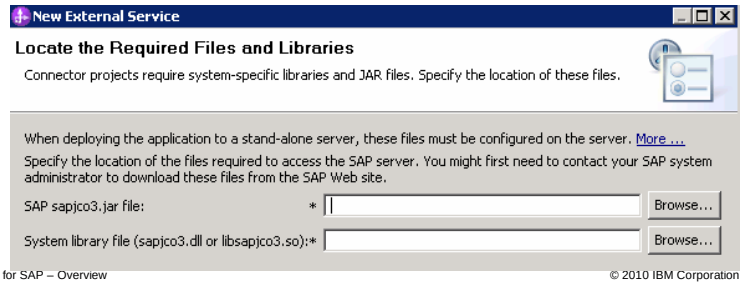
Section

Configuring dependent files

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.

Specify dependencies (1 of 2)

- With V7.0 adapter supports connectivity through SAP jco3 library
- During configuration, you are prompted for sapjco3.jar and sapjco3.dll or libsapjco3.so files in external service wizard
- At runtime
  - place sapjco3.jar and sapjco3.dll or libsapjco3.so file under <WPS_INSTALL_DIR>\bin
  - place sapjco3.jar file under <WPS_INSTALL_DIR>\lib\ext
  - CWYAP_SAPAdapterExt.jar shipped with adapter in WebSphere Integration Developer.
    - place it in <WPS_INSTALL_DIR>\lib\ext



The adapter requires the dependant files to communicate with the Enterprise Information System (EIS).

With V7.0 the adapter uses SAP JCo3 library to connect to the SAP system. CWYAP_SAPAdapterExt.jar is packaged with the adapter in the WebSphere Integration Developer. This jar file is required for running adapter in JCo3 environment. During runtime, the jar needs to be in application server class path.

At the design time, you can now browse and select the dependent files in the external service wizard. For runtime usage of these dependent files by adapter, you need to place the files in the locations shown in the slide.

Specify dependencies (2 of 2)

- Select the SAP Adapter entry in the External Service Wizard
- Add EIS external dependencies, “sapjco.jar
  - For WebSphere Process Server: <*WPS_INSTALL*>\lib directory
  - For WebSphere Integration Developer: You are prompted in External service wizard
- Add the external SAP native libraries required for the adapter
  - For WebSphere Process Server : <*WPS_INSTALL*>\bin directory
  - For WebSphere Integration Developer : You are prompted in External service wizard
  - Libraries are:
    - librfc2.dll (or .so)
    - Sapjcorfc.dll (or .so)

The adapter requires the dependant files to communicate with the Enterprise Information System (EIS). At the design time, you can now browse and select the dependent files in the external service wizard. For runtime usage of these dependent files by adapter, you need to place the files in the locations shown in the slide.

Deployment in WebSphere Process Server

- Add the SAP dependent jar files and DLLs to the WebSphere Process Server class path as shown in previous slide
- 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.

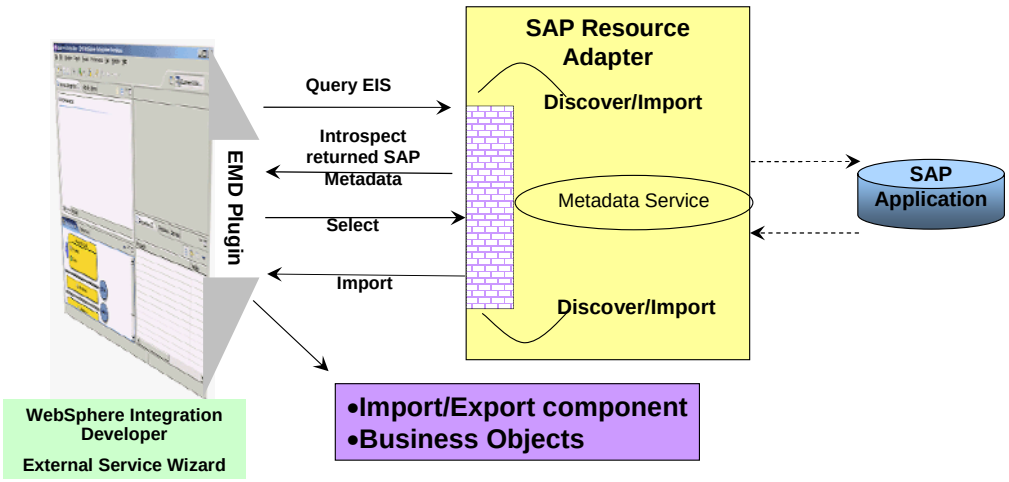
Add necessary SAP dependent jar files and DLLs to the WebSphere Process class path. Installation of the application containing the WebSphere Adapter for SAP to the WebSphere Process Server is similar to installing any other enterprise application. Either the administrative console or command line tool,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

Section

Enterprise metadata discovery

This section will cover Enterprise Service Discovery.

SAP adapter metadata discovery

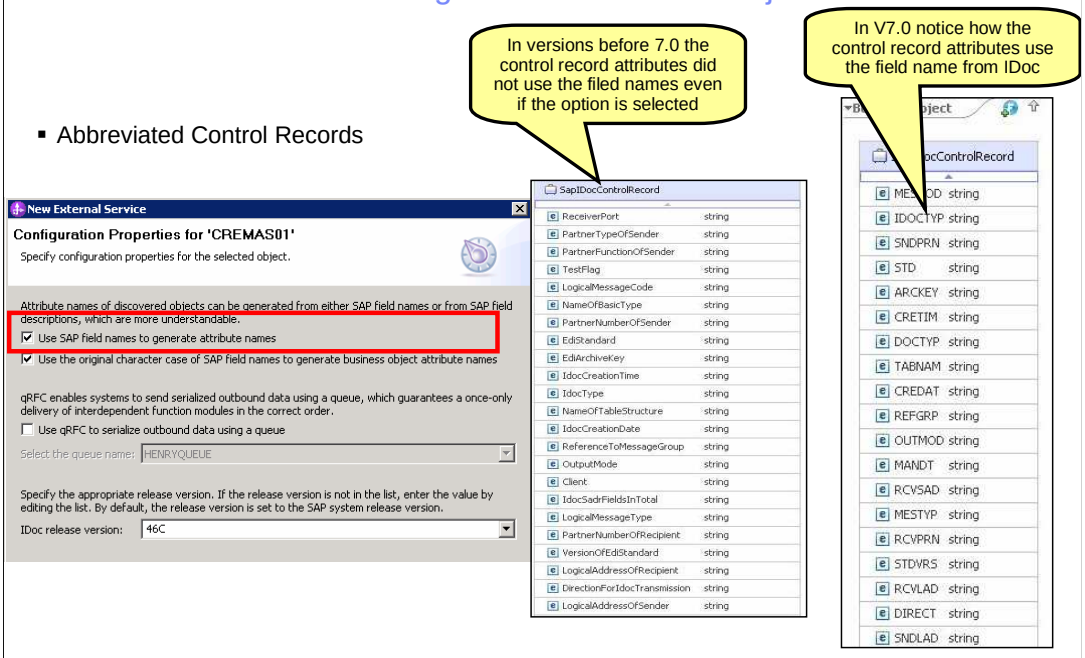


This diagram shows the interaction between the External Service wizard in WebSphere Integration Developer and the Enterprise Metadata Discovery (EMD) support classes within the SAP adapter. It also shows the end to end flow of interaction from the tool to the adapter to the SAP application to create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

The wizard lets you discover and import SAP metadata definitions and other artifacts related to BAPI, ALE and other interfaces. Based on your selection of metadata, business objects containing properties that correspond to the method arguments are generated. Application specific information is set in the business objects. The import and export components along with other artifacts are also generated based on the interactions style you chose.

Control record attributes in generated business objects

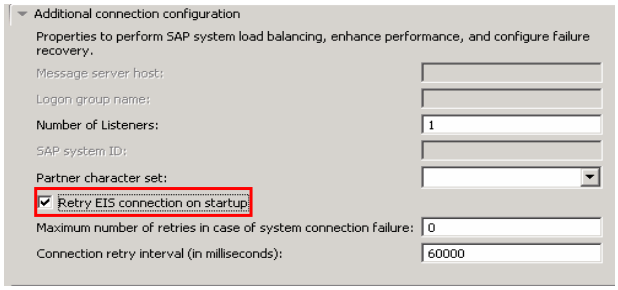
- Abbreviated Control Records



In prior versions selecting the option to “use SAP field names to generate attribute names” for the business objects only resulted in SAP field names being used for the data record and not control record. With V7.0 if you enable this option, both control and data record attributes in the business object will use SAP IDoc field names.

Retry connection

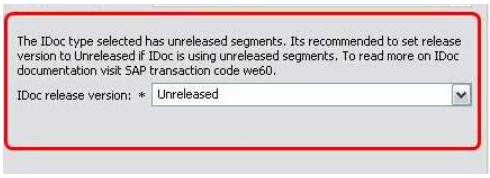
- Support connection retry to SAP system during application start
  - Connection retried at application startup time
  - Available in V6.2 for most of the adapters
  - Support added for SAP adapter in V7.0



Retry connection is an option that you can enable when configuring the adapter using the external service wizard. When enabled, during application start, the adapter will retry connecting to the SAP system based on the number you configure for maximum number of retried property.

Unreleased segments support

- Provide option for selecting unreleased segments
  - notified that IDoc has unreleased segment
  - previous version of the adapter
    - You have to manually set the release version to empty value (“ ”)



In V7.0 when an IDoc that has unreleased segments discovered, label information and the tool tip changed automatically to give more information about unreleased segments. Release version is automatically set to unreleased as shown in the screen capture here. External service wizard generates the schema definitions using unreleased segment definitions. In previous versions you have to manually set the IDoc release version attribute to a specific value.

Section

Transaction and security

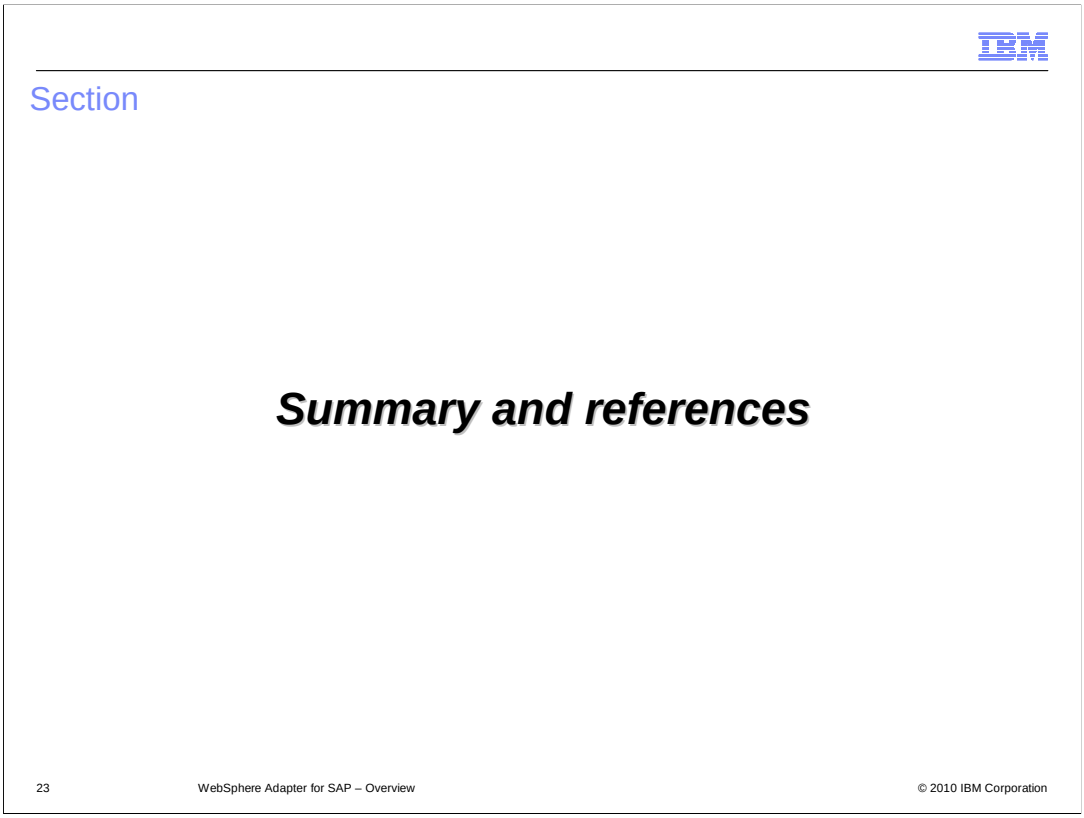
The next section covers the transaction and security support.

Transaction and security

- Transaction
  - SAP application support transaction for BAPI calls only
  - Adapter provides local transaction support for the BAPI interface using these 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 an SAP transaction ID with the business object, the adapter will return one after executing the transaction. If the JCA client has an 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.



The next section covers the summary and references.

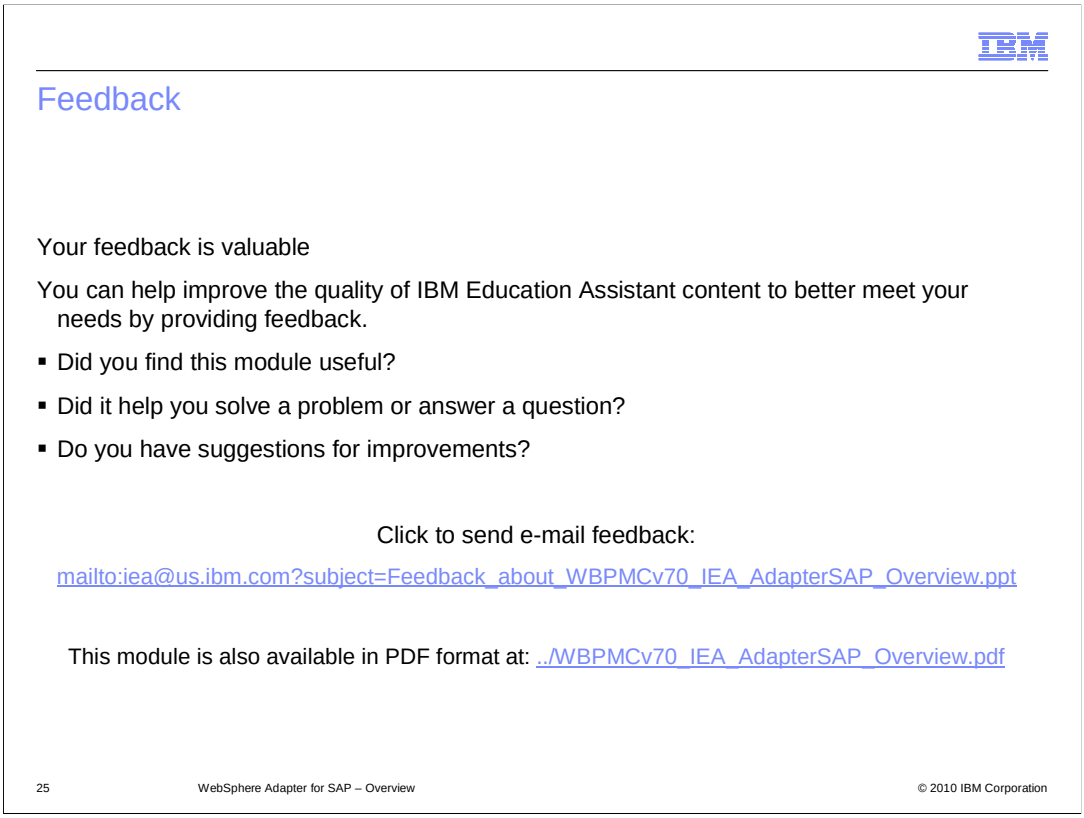
Summary and references

- Summary
  - Discussed SAP adapter in details covering the architecture and enterprise service discovery
  - Looked at transaction and security
- References
  - Information center:

http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/topic/com.ibm.wbit.620.help.adapter.emd.ui.doc/topics/tcreatecmps.html

  - User guide
  - IBM Education Assistant

In summary, this presentation covered the overview of WebSphere Adapter for SAP including the architecture and enterprise service discovery. The presentation has also provided information on transaction and security. More information on WebSphere Adapter for SAP can be found in the user guide and the information center for the adapter. Details about inbound and outbound functionality are covered in separate presentations.



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

Trademarks, disclaimer, and copyright information