



IBM Software Group

WebSphere® Message Broker V6.1

SAP adapter - Input node functionality



@business on demand.

© 2008 IBM Corporation
Updated January 29, 2008

This presentation covers the functionality of the WebSphere Adapter for SAP input node.

Agenda

- Overview
- Inbound operations
- Summary

The agenda for this presentation is shown here.

IBM Software Group 

Section

Overview

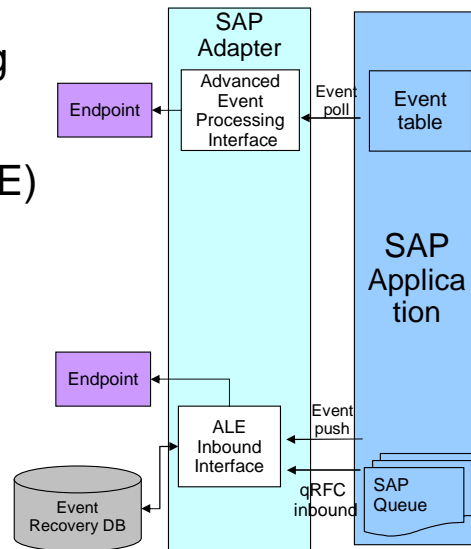
 3

SAP adapter - Input node functionality © 2008 IBM Corporation

This section provides an overview of the WebSphere Adapter for SAP inbound support

SAP adapter inbound support

- Inbound calls supported using these interfaces:
 - Application Link Enabling (ALE) using asynchronous event notification
 - ▶ For processing IDocs
 - ▶ Event recovery table is used for inbound event management
 - Advanced Event Processing
 - ▶ For processing custom IDocs



This diagram shows the high level flow and the components for 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 SAP applications. The adapter uses an event recovery table to manage the inbound events. With Advanced Event Processing interface, the adapter processes events related to custom IDocs.

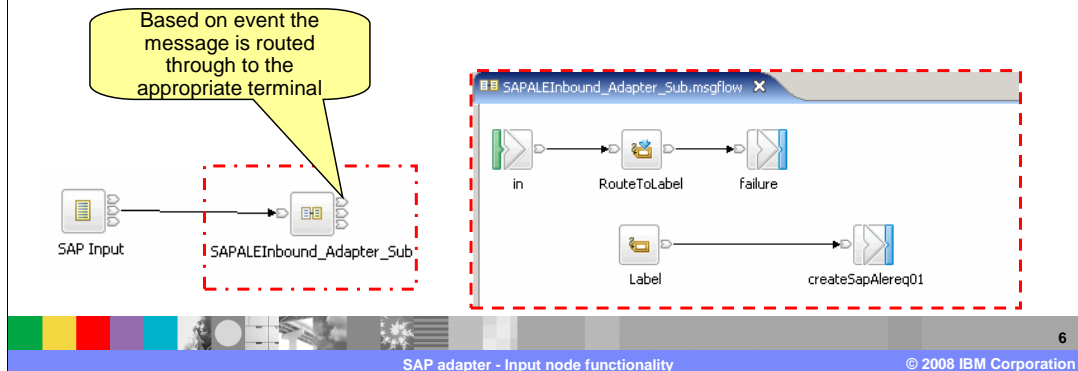
Section

SAP input node operations supported using Application Link Enabling interface

This section covers the inbound operations supported using only the ALE interface.

Inbound operations: High level flow (ALE only)

- Adapter acts as a Remote Function Call (RFC) Server by listening for ALE events from an SAP application
- Whenever an event occurs in SAP, the event is pushed to the adapter by way of the event listeners
- Adapter processes the event (message) and forwards the message to the rest of the message flow



The adapter creates a listener thread, which regularly polls for events, to the SAP system. In effect, the adapter acts as a Remote function call server to the SAP application. Whenever an event occurs in SAP, it is pushed to the adapter by way of the event listeners. There is no event store within the SAP application, so the adapter uses a persistent event recovery table to track events. The adapter receives the event from SAP as data in IDoc table format and converts the data into a message before sending it to the rest of message flow

ALE interface - Inbound IDoc scenarios

- SAP adapter supports processing
 - ▶ Single IDoc
 - ▶ Split IDoc Packets
 - ▶ Unsplit IDoc Packets
 - ▶ Unparsed IDocs & Packets

SAP adapter supports processing of single IDoc and IDoc packets. By default, the adapter splits IDoc packets into individual IDocs, but the adapter also provides functionality to process the whole IDoc packet without splitting. While processing the IDocs, the adapter parses the incoming IDoc structure and populates the IDoc content into the appropriate message. In addition, the adapter provides functionality to send the IDoc content without parsing if needed. These topics are covered in more detail in subsequent slides in this presentation.

Single IDoc or IDoc packet

- Event occurs in the SAP system
- IDoc or an IDoc packet corresponding to the event sent to the adapter
- Event processing
 - ▶ Single IDoc – IDoc is parsed and message is created using the message definition. Message sent to the rest of message flow
 - ▶ IDoc packet – IDoc packet is split into individual IDocs and populated in the message sent to the rest of message flow

The adapter listens for events occurring on the SAP system and when they occur the resulting IDoc or IDoc packet is sent to the adapter. If the incoming event relates to a single IDoc, the adapter parses the incoming IDoc and populates its contents to the appropriate message using the message definition created by the adapter connection wizard. If the incoming events relates to an IDoc packet, the adapter splits the IDoc packet into individual IDocs, populates the data in appropriate messages and sends the message through the rest of the message flow.

Single IDoc or IDoc packet message structure

- If error occurs during parsing of IDoc packet and
 - ▶ `IgnoreIDocPacketErrors = false`, adapter stops processing any further IDocs in the packet and reports errors to the SAP system

Structure	Type	Min Occurs	Max
SapAlereq01.mxsd			
Messages			
SapAlereq01	SapAlereq01		
SAPTransactionID	string	1	1
qRFCQueueName	string	1	1
SapAlereq01IDocBO	sapalereq01idocbo:SapAlereq01...	1	-1
DummyKey	string	1	1
SapIDocControlRecord	SapIDocControlRecord	1	1
SapAlereq01DataRecord	SapAlereq01DataRecord	1	1
Types			
Groups			
Elements and Attributes			

Content of IDoc parsed and stored here

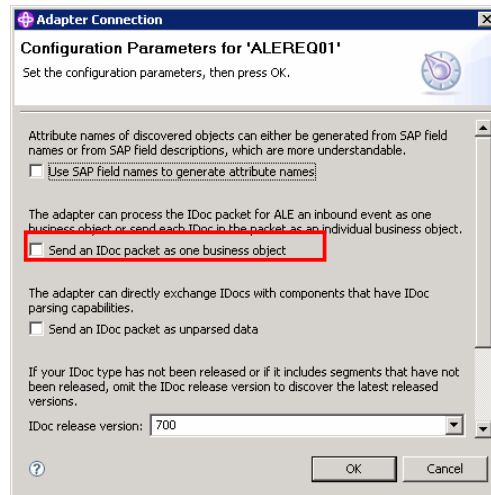
In case of IDoc packet, there are multiple IDocs populated in this structure

SAP adapter - Input node functionality © 2008 IBM Corporation 9

Shown here is the message definition created by the adapter connection wizard based on the IDocs you selected. The adapter uses this message definition to populate the IDoc data received from the SAP system. `IgnoreIDocPacketErrors` is a flag that you can set for each IDoc when you are using the Adapter Connection Wizard. If an error occurs during parsing of an IDoc in the IDoc packet and `IgnoreIDocPacketErrors` is set to false, the adapter stops processing any further IDocs in the packet and reports errors to the SAP system. If the flag is set to true, the adapter continues with processing the next IDoc in the packet.

Event processing for parsed non-split IDoc packet

- Parsed non-split IDoc packet
 - ▶ Option can be specified during discovery in Adapter Connection Wizard
 - ▶ IDoc Packet not split into individual IDocs
 - ▶ *IgnoreIDocPacketErrors* property is not valid
 - ▶ Failure during delivery requires whole IDoc packet to be resent



10

SAP adapter - Input node functionality

© 2008 IBM Corporation

Shown here is the optional flag that can be set to change the behavior of the adapter when processing an IDoc packet. The “Send an IDoc packet as one business object” flag is set at design time using the adapter connection wizard. If this flag is set (by selecting the box) the adapter does not split the incoming IDoc packets into individual IDocs, but treats the IDoc packet as a single entity, which is sent to the endpoint. One use case for this feature is for maintaining the transaction of data if inventory levels for a set of materials must be updated at once. This feature often does increase memory usage since the whole IDoc packet is stored in memory. The adapter can handle processing an IDoc packet as one unit of work if all the IDocs in the packet are of the same IDoc type. The event processing is similar to a single IDoc with respect to the event recovery mechanism. The IDoc packet is considered a single unit of work so a failure registered by the endpoint for any one IDoc fails the whole packet and requires all the IDocs in the IDoc packet to be resent to the endpoint.

Event processing for unparsed IDocs

- Adapter supports sending of events without parsing the data, based on the property set using Adapter Connection wizard.
 - ▶ Message contains the IDoc as byte array. Endpoint must handle the parsing and converting it to a appropriate format

Select this option if you want the adapter to not parse the IDoc contents

The adapter can process the IDoc packet for All found event as one business object or send each IDoc in the packet as 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 an IDoc packet as unparsed data

Structure	Type	Min Occurs	Max Occurs
SapAlereq01.msxd			
Messages			
SapAlereq01	SapAlereq01		
SAPTransactionID	string	1	1
qRFCQueueName	string	1	1
SapAlereq01IDocBO	sapalereq01idocbo:SapAlereq01IDocBO	1	-1
DummyKey	string	1	1
IDocData	hexBinary	1	1
SapIDocControlR	SapIDocControlR	1	1

The Data of the IDocs is unparsed and sent as-is

The adapter supports processing unparsed events for applications or components that have parsing capability. A value of "UNPARSEDIDOC" is set for the metadata property "Type" at the business object level. The value for this property is set by the Adapter Connection wizard and this flag determines the kind of message definition to generate (IDOC or UNPARSEDIDOC). If the endpoint requires the incoming IDoc not to be parsed, you can set the type of IDoc type to UNPARSED in the Adapter Connection Wizard and the message will contain the IDoc as a byte array.

ALE inbound events – Additional details

- Updating IDoc status
 - ▶ Adapter can be configured to update the IDoc status back to SAP for monitoring purposes
 - ▶ Done by setting **ALEUpdateStatus** property to true
 - ▶ Update is accomplished by updating a status IDoc called ALEAUD that the adapter sends to the SAP system, after the message is sent to the endpoint
 - ▶ Attributes to indicate success and failure are ALESuccessCode, ALESuccessText, ALEFailureCode, and ALEFailureText

Shown here are some additional details for ALE Inbound events. If the **ALEUpdateStatus** attribute is set to true in the adapter connection wizard panel, the adapter updates the IDoc status back to the SAP application for monitoring purposes. The update is accomplished by updating a status IDoc called ALEAUD, which the adapter sends to the SAP system after the event is sent to the endpoint. You can configure the adapter to update the IDoc status for the purpose of monitoring your IDoc processing. Setting the adapter configuration property **ALEUpdateStatus** to true indicates that an audit trail is required for all message types. The adapter updates the IDoc status of ALE business objects that are retrieved from the SAP system. The update is accomplished by updating a status IDoc called ALEAUD, which the adapter sends to the SAP system as an inbound IDoc event. After the event is sent to the endpoint, the adapter updates the status of the IDoc in SAP to indicate processing failure or success.

ALE inbound supported operations

- Adapter uses IDoc control record field to determine the operation
- Following fields in the IDoc control record are used to determine the operation:
 - ▶ Logical_message_type (MESTYP)
 - ▶ Logical_message_code (MESCOD)
 - ▶ Logical_message_function (MESFCT)
- Operation in the control record used to set the destination label names in the local environment tree.
 - ▶ Label information used by the Route to Label node to route it to appropriate node in the flow

The supported verbs for ALE inbound operations are create, Update and Delete. The adapter uses the IDoc control record field to determine the verb. The operation is then set on the message, before sending it to the rest of the flow. The values you set for the message type, code and function in the adapter connection wizard should match the values set in the incoming IDoc for successful processing of that IDoc.

Section

SAP input node supported operations using Advanced event processing interface

This section covers the operations supported using the Advanced Event Processing interface.

AEP interface – Inbound flow

- IDocs used for data transfer
- Adapters “polls” the event table on the EIS
- Whenever an event occurs in SAP, the event is placed in the event table
- Adapter retrieves the event and retrieves the IDoc info by using the ABAP handler
- IDoc is converted to message and sent to the endpoint

As with the ALE interface, the Advanced Event Processing (AEP) Interface also uses IDocs for data transfer, but the AEP interface is generally used for processing custom IDocs. The adapter polls on the event table on the SAP system, where new events are captured. This differs from the ALE interface event management where the event recovery store is external to the SAP system and events are pushed to the adapter. The adapter retrieves the information from these custom IDocs using the appropriate handler provided, and the retrieved IDoc information is populated into a message using the message definitions already created using the adapter connection wizard and returned to the message flow.

AEP interface – Event management

- You need to define event detection logic to identify events
- Event information is stored in the event table
- Event table schema defined by the adapter
- Transports provided with adapter to set up the event table
- WebSphere Business Integration station tool(transaction /n/CWLD/HOME_AEP) can be used to maintain
 - ▶ event queue
 - ▶ archive table
 - Resubmit events from archive table
 - Delete events from archive table
- EventTypeFilter property specified in the activation spec.
 - ▶ comma delimited list of IDoc types
 - ▶ only the types specified in the property are picked for processing

Shown here are the event management details for Advanced Event Processing (AEP) interface. You must provide the event detection logic on the SAP side to detect any events, which are then captured in the event table. The scripts for creating the event table and other infrastructure required for supporting AEP interface are provided as transports with the WebSphere Message Broker Toolkit. You can also make use of the WebSphere Business Integration Station tool to maintain both event and archive queues and you can specify event filters when you are generating definitions for the custom IDocs using the Adapter Connection Wizard. The IDoc names that you are interested in monitoring for events can be provided for the filter property. Only the types specified in the property are selected for processing

AEP interface – Event table

Name	Type	Description
event_id	NUMBER	Unique Eventid which is a Primary key for the table
object_name	STRING	Business graph name for WPS, and business object name for other brokers (WAS, WMB, WTX, DataStage) (not null)
object_key	STRING	Delimited String that contains keys for the business object (not null)
object_function	STRING	Operation corresponding to the event (Delete, Create, Update etc) (not null)
event_priority	NUMBER	Not Null; valid values are any positive integer
event_time	DATE	Date/time when event was generated
event_status	NUMBER	(In Progress, Success, Failure etc (not null) Values: 0 – Ready for poll 1 – delivered -1 – failed 2 – Event PreQueued 3 – Event InProgress 4 – Event Locked
Xid	STRING	Unique xid (transaction id) value for assured once delivery
event_user	STRING	User that has created the event.
event_comment	STRING	Description

17

Shown here is the schema of the event table that is created on the SAP system. Any events occurring on the IDocs that you are interested in should be detected and populated in this event table. The adapter then uses the information in this table to extract the corresponding IDoc using the handler.

Section

Summary and references

This section provides a summary and references.

Summary and references

- **Summary**
 - ▶ Discussed SAP input node's functionality and various interfaces supported in detail
- **References**
 - ▶ Information center
 - ▶ User guide
 - ▶ IBM Education Assistant

This presentation covered SAP Input node functionality and the various interfaces supported in detail. More information can be found in the user guide and the information center

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:

mailto:iea@us.ibm.com?subject=Feedback_about_WMBV61_SAPAdapter_InputNode.ppt

This module is also available in PDF format at: ..\\WMBV61_SAPAdapter_InputNode.pdf



You can help improve the quality of IBM Education Assistant content by providing feedback on this module using the link on this slide.

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

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 2008. 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.