



IBM Software Group

# IBM WebSphere® Adapters V6.0.2

## *WebSphere Adapter for Email V6.0.2*



@business on demand.

© 2007 IBM Corporation  
Updated March 9, 2007

This presentation will cover WebSphere Adapter for Email V6.0.2

## Agenda

- **Overview**
- Enterprise Service Discovery
- Configuration Properties
- Business Object Structure
- Outbound Operations
- Inbound Operations
- Summary and References

This section will provide an overview of the WebSphere Adapter for Email. Installation and deployment of the WebSphere Adapter for Email is also covered in a separate presentation common for all WebSphere Adapters.

## Overview: WebSphere Adapter for Email

- IBM WebSphere Adapter for Email implements the Java™ 2 Enterprise Edition (J2EE) Connector Architecture (JCA), version 1.5 specification
- Enables bi-directional connectivity for integration with Enterprise Information System applications that can communicate through Emails
- Sending and receiving mails to/from different mail servers using Email protocols
  - ▶ SMTP
  - ▶ IMAP
  - ▶ POP3



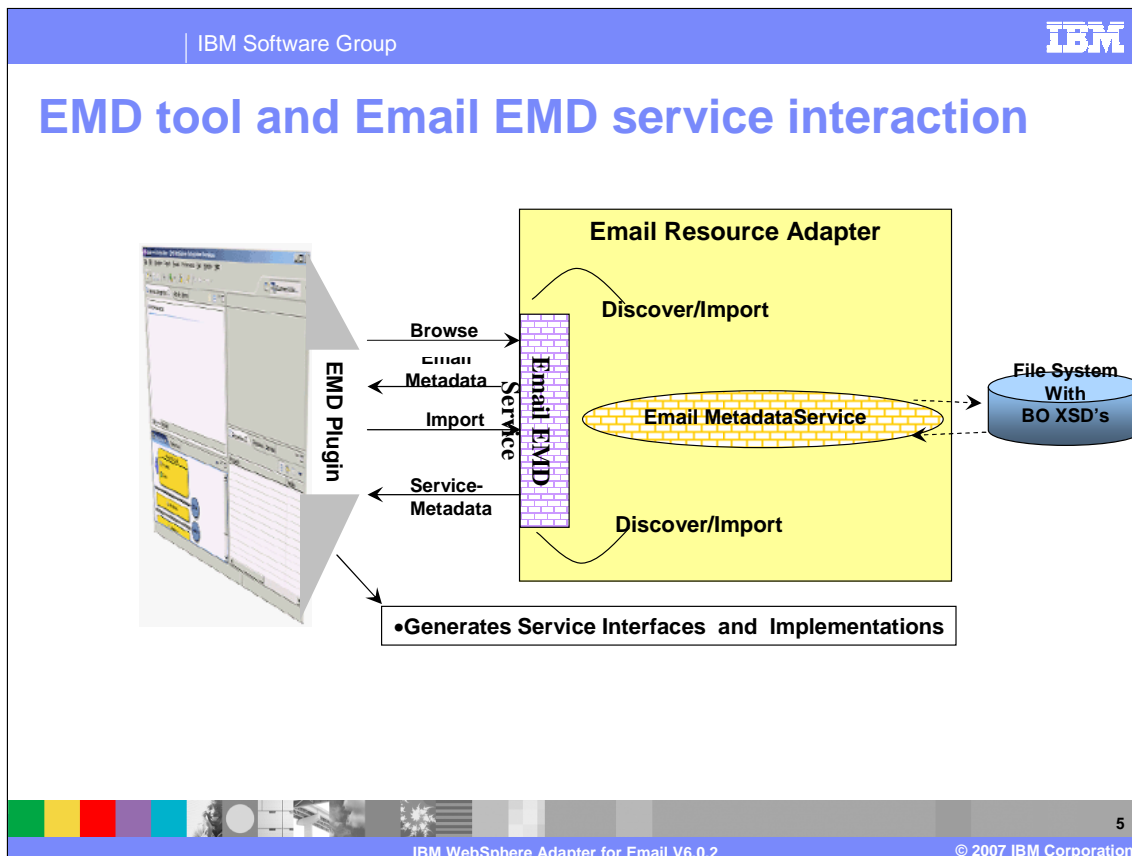
The IBM WebSphere Adapter for Email implements the JCA 1.5 specification, enabling bi-directional connectivity, both inbound and outbound, with those Enterprise Information System business applications that can communicate only through Emails. The Email Resource Adapter supports integration through sending and receiving mails to and from different mail servers using several Email protocols, including SMTP, IMAP and POP3.

## Agenda

- Overview
- **Enterprise Service Discovery**
- Configuration Properties
- Business Object Structure
- Outbound Operations
- Inbound Operations
- Summary and References



This section provides the configuration properties for WebSphere Adapter for Email. Common properties were covered in the WebSphere Adapter common details presentation. This section covers the Email specific attributes.



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

Email server EIS does not define a set of business entities and data structures for the Resource Adapter (RA) to work with. So you must pre-define the Business Objects in accordance with their business requirement, and store the XSDs in a local folder on the system where the EMD is running. This task must be done before the Email Enterprise Service (Metadata) Discovery tool is run.

EMD must discover the interfaces for the Email server Enterprise Information System (EIS), which does not explicitly expose any interfaces to the RA. The Enterprise Service (Metadata) Discovery mainly allows you to provide the ActivationSpec or Managed Connection Factory (MCF) properties, and the Resource Adapter properties as input to build the service description.

Once the necessary function and interaction style (whether outbound or inbound) is selected, the Enterprise Service (Metadata) Discovery tool will generate the artifacts for the outbound or inbound call. Artifacts include items such as the EIS SCA export or import components, the Business graph and business objects, the interfaces defined by WSDL and others needed by SCA clients to send an outbound or receive inbound requests.

## Agenda

- Overview
- Enterprise Service Discovery
- **Configuration Properties**
- Business Object Structure
- Outbound Operations
- Inbound Operations
- Summary and References



This section provides the configuration properties for WebSphere Adapter for Email. Common properties were covered In the WebSphere Adapter common details presentation. This section covers the Email specific attributes.

## Activation specification properties (for inbound) – Connection properties

Property	Description
Host	The host ip of the mail server.
Port	The port on which the mail server is listening.
Protocol	The protocol to be used for inbound communication with the mail server. This field has to be set to either IMAP or POP3.
Username	user id to be used for the IMAP/POP3 session.
Password	The password, for user id, to authenticate the IMAP/POP3 session.
PollFolders	The folder on which the adapter should poll.
MatchAllCriteria	This field will contain search criteria used as filtering criteria to selectively poll the Poll Folder. All criteria defined in this field will be ANDed for the search.

7

This table shows some of the ActivationSpec configuration properties for inbound calls. Host, port, protocol, username and password relate to the inbound connection properties. Poll Folders property is used to specify the list of folders that should be polled for Emails. For POP3 the default is INBOX, but in the case of IMAP protocol, you can specify multiple folders.

MatchAllCriteria property is used to specify filtering criteria to selectively poll the Poll Folder. All criteria defined in this field will be used for the search in AND mode.

## Activation specification properties - (cont.)

Property	Description
MatchSomeCriteria	This field will contain conditions which are used as filtering criteria to selectively poll the Poll Folder. The conditions specified under this property will be ORed for the search. Options for NOTing will also be provided.
InProgressFolder	The folder, on the file system, where the polled mails are first written to, in RFC822 format. The name of the RFC822 format file will correspond to <i>the Message-ID</i> of the mail. These details are retrieved from the Email header.
ArchiveFolder	The file-folder (on the local system where RA is running ) into which the successfully processed mails will be archived, in RFC822 format. If no Archive Folder is mentioned the Email RA will not archive the successfully processed Emails. The events are deleted from the InProgressFolder
FailedEventsFolder	The file-folder (on the local system where RA is running ) into which the un-successfully processed mails, or failed events, will be archived, in RFC822 format. If no Failed Events Folder is mentioned, the Email RA will not archive the un-successfully processed Emails.
ArchiveFileName	User can specify the pattern for the name of the Archive file. The pattern can be a list of Header names that are comma delimited.

8

This slide lists some more Activation specification properties. **MatchSomeCriteria** property is used to specify filtering criteria to selectively poll the Poll Folder. All criteria defined in this field will be ORed for the search. **InProgressFolder** specifies the folder on the file system, where the polled mails are first written to, in RFC822 format. **ArchiveFolder** and **FailedEventsFolder** are used to store successful and failed events.



## Activation specification properties for event recovery table

Property	Description
EP_DataSource_JNDIName EP_CreateTable EP_TableName EP_UserName EP_Password EP_SchemaName	Event Recovery Table information for inbound events.

This table shows more ActivationSpec configuration properties for inbound calls, related to the Event recovery table.

## Managed connection factory properties (for outbound)

Configuration Property	Details
Host	The host ip of the mail server.
Username	User id to be used for the SMTP session.
Password	The password, for user id, to authenticate the SMTP session.
Port	The port on which the mail server is listening.
Protocol	The protocol to be used for outbound communication with the mail server. For this release, only the SMTP protocol is supported for outbound.

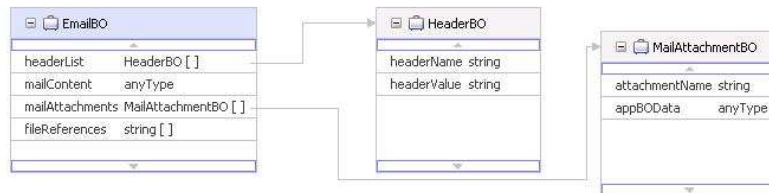
This table shows the managed connection factory configuration properties for outbound request. The properties host, port, protocol, username and password are used to create a connection to the EIS System. For version 6.0.2 release of WebSphere Email Adapter, only SMTP is supported for outbound requests.

## Agenda

- Overview
- Enterprise Service Discovery
- Configuration Properties
- **Business Object Structure**
- Outbound Operations
- Inbound Operations
- Summary and References

This section describes the structure of the business object used by the WebSphere Adapter for Email.

## BO hierarchy



The Email Resource Adapter uses the Email business object structure for both the inbound and the outbound communication. The functional properties are implemented as the properties of the business objects. The association information for each mime-type to a data-binding, and the configuration information for each data binding is stored as annotation in the Email BO structure, and in the BO-level annotations for the application BOs.

## Email BO

- Structure of the Email BO comprises of all details required by the RA during the inbound and the outbound.

Attribute Name	Type	Required	Description
headerList	HeaderBO[]	Yes	Will contain details of all the headers on the polled Email
mailContent	anyType	Yes	In the inbound context, this stores the content/data of the Email that was read and passed in by the adapter. In the outbound context, this contains the data that has to become the content of the mail and not the attachment of the mail. In a pass-through scenario, this will have the UnstructuredContent BO.
mailAttachments	AttachmentBO [ ]	No	Will contain content details for all the attachments of the Email
fileReferences	String[]	No	Will contain a list of files that needs to be attached to the Email. During outbound, the J2EE client specifies absolute paths of the files in this field. The adapter reads those files from local file system (where the adapter runs), and attaches them as attachments to the Email that is created during outbound. The property is not used during inbound. Note that only local files is <b>supported attaching</b> in this release.

13

Structure of the Email BO is comprised of all details required by the RA during the inbound and the outbound interactions.

- **headerList** is an array of HeaderBO which contains details of all the headers on the polled Email.
- **mailContent** field of the EmailBO is used to store the content/data of the Email that was read and passed in by the adapter.
- **mailAttachments** is an array of AttachmentBO that contains content details for all the attachments of the Email.
- **fileReferences** field is an array that contains a list of files that need to be attached. This field is only applicable to Outbound Operations.

## Header BO

- Used to store all standard(RFC 822) Email headers along with customized user headers
- The Head BO consists of a name and value pair.
  - ▶ Name of the header and value of the header
- Some of the Standard Email headers are listed here

Attribute Name	Required	Description
From	Yes	The From address of the Email. This field contains the identity of the person(s) who want this message to be sent.
To	Yes	The To addresses of the Email. This field contains the identity of the primary recipients of the Email.
Cc	No	The cc addresses of the Email. This field contains the identity of the secondary (informational) recipients of the message
Subject	No	This is intended to provide a summary, or indicate the nature, of the message .
Message-id	No	A message identifier pertains to exactly one instantiation of a particular message; subsequent revisions to the message should each receive new message identifiers

Header BO is used to store all standard (RFC 822) Email headers along with customized user headers and consists of a name value pair. The name field corresponds to the header name and value holds the value for the header. Apart from these standard headers, the RA also provides support to define custom headers, which can be defined and used within the enterprise. The name of the custom header and related information will be tracked by the Email RA with the help of the header List. Some of the supported Email headers are listed in the table.

## MailAttachment BO

- Corresponds to the attachments in the mail.
  - ▶ can be of any user-defined type
- During inbound, the attachments would be parsed and the contents would be sent out as this data object.
  - ▶ One attachment gets parsed into one MailAttachmentBO.
- During outbound, the J2EE client sets the data within this BO in the request. The same would become an attachment in the Email that gets created by the RA.

Attribute Name	Required	Description
attachmentName	Yes	Specify name of attachment
appBOData	Yes	The data comprises of the mail attachment content. anyType datatype can hold hexBinary or any SDO type of data. The data binding would decipher the hexBinary content as the UnstructuredContent BO.

MailAttachment BO corresponds to the attachments in the mail, and can be of any user-defined type. During inbound, the attachments are parsed and the contents sent out as this data object. Each attachment gets parsed into a MailAttachmentBO. During outbound, the J2EE client sets the data within this BO in the request, which becomes an attachment in the Email that is created by the RA.

## Agenda

- Overview
- Enterprise Service Discovery
- Configuration Properties
- Business Object Structure
- **Outbound Operations**
- Inbound Operations
- Summary and References

This section describes details about the outbound request support provided by WebSphere Adapter for Email.



## Email RA outbound support – High level flow

- A SCA component invokes an interaction with Mail Server
- Business Object representing the Email is passed from the SCA client to the Adapter
  - ▶ BO metadata contains properties that correspond to the method arguments and other application specific information
  - ▶ User can optionally attach files as attachments
  - ▶ BO is wrapped in the JCA CCI Record object
- Adapter extracts the elements from the BO
  - ▶ From the BO metadata, the Adapter composes an Email, attaches any specified files to the Email and deliver it to the mail server using SMTP.

The high level flow of an outbound call is described here. A J2EE Client sends the Email BO instance to the Email Resource Adapter (RA) during outbound. The BO's were generated by the Enterprise Metadata Discovery wizard within WebSphere Integration Developer. The Email RA composes an Email from the Email BO instance and sends it across to the mail server. If the mail creation on the mail server fails, an appropriate exception is thrown. You can optionally specify a set of files from the file system to be attached to the Email. The Email RA then attaches the specified files to the mail and deliver it to the mail server using SMTP. The content extracted from Email BO object is converted into a mail with attachments, and sent across to the specified mailbox, the details of which would exist in the request Email BO.

## Email RA outbound operations

- Create is the only operation supported
  - ▶ An Email is created from details provided within Email BO, and delivered to the mail server, to be sent to destination mail addresses.
  - ▶ If the mail is sent successfully, a null WBIRRecord is returned back to the J2EE client that generated the request.
  - ▶ If the mail sending fails for any reason, an appropriate exception is thrown back to the calling component

Create is the only supported operation. An Email is created from details provided within Email BO, and delivered to the mail server, to be sent to destination mail addresses. If the mail is sent successfully, a null WBIRRecord is returned to the J2EE client that generated the request. If the mail send fails for any reason, an appropriate exception is thrown back to the calling component.

## Email RA data transformation during outbound

- Receive the EmailBO instance as the request.
- Instantiate a corresponding EmailRecord object.
- Parse the array of HeaderBOs and populate the hashmap for headers
- Retrieve the Data Object from the mailContent attribute of Email BO.
  - ▶ This can be either an UnstructuredContent BO, or a specific BO, such as "Customer".
  - ▶ Parse the annotation on the respective BO, retrieve the mime-type and then call the appropriate DB.
  - ▶ Receive back the bytes/text corresponding to the content, and populate the same into the content field of the EmailRecord.
- Process the mail attachments array. For each attachment in the array, perform the following
  - ▶ Retrieve the object-type of the anyType appBOData within the MailAttachmentBO.
  - ▶ If the object-type is UnstructuredContent( in a pass-through scenario), an EmailAttachmentRecord is instantiated, and the content field will be filled with the bytes from appBOData.
  - ▶ If the object-type is a specific Data Object, then –
    - parse the annotation on the respective BO, retrieve the value of the content-type and then call the appropriate DB.
    - Get back the bytes/text from the DB and set the same on the content field of the EmailAttachmentRecord.
- Check for the file references. For each file reference, populate the array list in the EmailRecord with the file-reference paths. The RA would retrieve the list of files while processing the request, and load the files as mail attachments.
- Send out the populated EmailRecord instance to the RA.

19

IBM WebSphere Adapter for Email V6.0.2

© 2007 IBM Corporation

Listed here are the steps involved in outbound processing when data transformation is involved during outbound call. The resource adapter receives the populated EmailBO, which must be wrapped in a Record Object and an Email record object is instantiated. Resource adapter retrieves the header information from the HeaderBO and populates it in the record object. The mail content is retrieved from the mailContent attribute of Email BO. Depending on the type of data in the mail content, the appropriate data binding handler is invoked by passing the mail content data to the handler. Data binding handler processes the data and sends the bytes/text to the adapter which populates this information in the content field of Email record. Resource adapter processes the mail attachments by retrieving them from the MailAttachmentBO. An appropriate data binding handler is called which processes the data and returns it to the resource adapter which then populates the information to the **EmailAttachmentRecord**

If any files are listed to be attached to the Email, they are specified in the fileReferences field. Resource Adapter fetches the files from the local file system and attaches them to the Email Record. Then populated Email record is sent out to the mail server.

## Agenda

- Overview
- Enterprise Service Discovery
- Configuration Properties
- Business Object Structure
- Outbound Operations
- **Inbound Operations**
- Summary and References

This section describes details about the inbound operations support provided by WebSphere Adapter for Email.

## Email RA inbound support : High level flow

- Email RA polls on the mail folders in a mail box on a mail server.
- Inbound communication can be either using IMAP or POP3 protocols, as configured by user during Enterprise Service (Metadata) Discovery
- Email RA picks up unread mails, from the mail folders
- Each mail is transformed into an Email BO instance that is delivered to the end point
- The event persistence framework takes care of delivering the event only once to the endpoint.

21

IBM WebSphere Adapter for Email V6.0.2

© 2007 IBM Corporation

During Inbound Flow, the Email RA polls the mail folders in a mail box on a mail server. The configuration properties for mail server, mail box credentials and poll folder names come from the Activation Specification. The inbound communication between the adapter and the mail server can be over either IMAP or POP3 protocols, and only one protocol is supported at a time during inbound. The Email RA reads a mail from the mail server and writes the mail as an RFC822 format file to the InProgress folder. The InProgressFolder in Activation Specification is used to configure the path of the folder. The name of the RFC822 format file will be the Message-ID. These details are retrieved from the Email header and uniquely identify an Email. Once a mail is copied to the InProgress folder, the mail is explicitly marked as deleted on the Mail Server. Each mail is transformed into an Email BO instance that is delivered to the end point. The header fields of an Email are populated in the headerList of Email BO. Each attachment of the Email is transformed accordingly and populated into the attachments List in the EmailRecord. Each attachment name, content and mime-type information is captured.

The event persistence framework handles delivery of the event only once to the endpoint. In case of an application server failure, when the restart of adapter occurs, the adapter first processes all mail files in the InProgress folder before processing the mails on the mail server.

If the processing of the mail is successful, mail is moved into the Archive Folder, if configured, from the InProgress Folder

If the processing of mail is unsuccessful, the mail is moved into the Failed Events Folder, if configured, from the InProgress Folder.

## Email RA inbound support : IMAP

- Email RA polls on the PollFolders specified by user on mail server at regular intervals, and picks up all the unread mails as events.
  - ▶ If multiple Email folders specified, RA polls all the folders sequentially in the same poll cycle.
  - ▶ Email RA can be configured to provide Multiple Mailbox/MailServer support
- All the polled mails saved to the InProgress file-folder in the RFC822 format.
  - ▶ The name of the RFC822 format file will be the *Message-ID* of the polled Email.
- Each mail is transformed into an Email BO instance, and delivered to the end point.
- The event management framework takes care of delivering the event only once to the endpoint.
- Once a mail is processed, the mail is explicitly marked as deleted on the Mail Server, and archived or deleted from the InProgress folder

22

This slide provides the details of inbound support using IMAP protocol. The mail folder to be polled must be specified in the PollFolders property. The Email RA polls on the PollFolders for the specified user on the specified mail server at regular intervals, and retrieves all unread mails as events. You can specify multiple Email folders (such as in the case of IMAP) for a user's mail account within the PollFolders property, in which case, the multiple Email folders are polled sequentially in the same poll cycle. When mailboxes of multiple users must be polled from multiple servers, you must configure multiple Activation Specifications. You can also optionally specify search criteria to retrieve event mails, in which case all unread mails meeting the search criteria will be picked up for polling. When no search criteria are specified, all unread mails are picked up by default.

The Email RA writes all the polled mails to the InProgress file-folder, as files saved in the RFC822 format, with the name of the file being the *Message-ID* of the polled Email. Each mail is transformed into an Email BO instance and the event management framework ensures that the event is delivered only once to the end point. Once a mail is processed, it is explicitly marked as deleted on the Mail Server, and archived or deleted from the InProgress folder.

## Email RA inbound support : POP3

- Email RA polls on the PollFolders specified by user on mail server at regular intervals, and picks up all the unread mails as events.
  - ▶ If multiple Email folders specified, RA polls all the folders sequentially in the same poll cycle.
- All the polled mails saved to the InProgress file–folder in the RFC822 format.
  - ▶ The name of the RFC822 format file will be the *Message-ID* of the polled Email.
- Once the mail is read and stored in the InProgress folder, it is deleted from the mail server ( per the POP3 specifications).
- Each mail is transformed into an Email BO instance, and delivered to the end point.
- The event management framework takes care of delivering the event only once to the endpoint.
- Once a mail is processed, the mail will be archived or deleted from the InProgress folder.

23

This slide provides the details of inbound support using POP3 protocol.

The steps involved in processing Emails using the POP3 protocol are essentially the same as in the case of IMAP, the only difference being that you cannot specify multiple mail folders when using POP3 protocol. Archiving is a feature on inbound that you can choose to use to archive all the successful and unsuccessful events polled by the RA. It is based on the values for the properties provided for ArchiveFolder and the FailedEventsFolder. If the ArchiveFolder is specified, the successfully processed mails are moved into the ArchiveFolder from the InProgressFolder. If the value is left blank, the successfully processed mails are deleted from the InProgressFolder. If the FailedEventsFolder is specified, the mails that failed processing are moved into the FailedEventsFolder from the InProgressFolder. If the value is left blank, the failed mails are deleted from the InProgressFolder. When multiple event folders are specified and separated by commas in the PollFolders property (only applies to IMAP), the Resource Adapter creates folders with the same name as the poll folder under the specified ArchiveFolder and the FailedEventsFolders and archives the corresponding event files in these folders.

## Email RA data transformation during inbound

- Receive the EmailRecord
- Instantiate the EmailBO
- Parse the headers of the Email, and populate headerList
- Retrieve the mail content and content type.
- Based on content-type, call the appropriate DB and pass across the mail content.
- Retrieve back the BO and populate the same in the mailContent attribute of Email BO.
- Process the array of EmailAttachmentRecords. For each EmailAttachmentRecord, perform the following –
  - ▶ Retrieve the content-type of the attachment. This feature is provided by the Java Mail API.
  - ▶ Based on the content-type, pass across the content of the attachment to the corresponding mime-specific Data Binding.
  - ▶ Receive back the appropriate BO and populate the same in the mailContent attribute of the Email BO..
  - ▶ For the mime-types that do not have any specified association with the mime-specific data binding ( in a pass-through scenario), the content will not be parsed. A MailAttachmentBO is instantiated by the DB and the bytes are filled into the appBOData field as-is.
- DB will have an array of MailAttachmentBOs depending on number of the attachments to the mail.
- Send out the EmailBO to the endpoint

Shown here is a list of the steps involved in inbound processing when data transformation is involved.

The resource adapter receives the Record Object and an EmailBO object is instantiated. Resource adapter retrieves the header information and populates them to HeaderBO.

The mail content and content type are retrieved Email Record. Depending on the type of data in the mail content, the appropriate data binding handler is invoked by passing the mail content data to the handler. Data binding handler processes the data and sends the bytes/text to the adapter which populates this information in the mailContent field of EmailBO. If the Email record has any attachments, an appropriate data binding handler is called which processes the data and returns it to the resource adapter which then populates the information to the MailAttachmentBO. The populated EmailBO is then sent out to the endpoint



## Email RA inbound support : Event table

- The adapter can track and recover events in case of abrupt termination using the data source for persisting the event state in an event recovery table

Event Persistence table Column	Description
EVENTID	Message ID for the Email
STATUS	Event processing status. Possible values are "NEW", "IN PROGRESS", "COMMITTED", "FAILED"
XID	
POLLFOLDER	Not used

This table shows the components involved in end-to-end handling of events. The new event is pushed to the adapter and the adapter stores it in the persistent event recovery table, which provides failure recovery to reprocess the events. The schema of the event recovery table is shown. The Event ID is the unique message id. The valid status fields are new, in progress, committed and failed. A mail on the mail server that corresponds to the selected search criteria will be marked as NEW. Once the mail is copied from mail server to the local folder, it will be marked as IN PROGRESS. Once the event is sent out to the Function Selector, the event is deleted from the table.

## Agenda

- Overview
- Enterprise Service Discovery
- Configuration Properties
- Business Object Structure
- Outbound Operations
- Inbound Operations
- **Summary and References**

This section will provide a summary of the WebSphere Adapter for Email.

## Summary

- WebSphere Adapter for Email enables integration with SCA Applications and Enterprise Information System applications that can communicate only through Emails.
  - ▶ Inbound and Outbound support
- Provided support for Enterprise Service Discovery for discovering services
- Supported integration through sending and receiving mails to/from different mail servers

In summary, this presentation covered the details of WebSphere Adapter for Email. The WebSphere Adapter for Email enables integration with SCA business integration applications and Enterprise Information System applications through Emails. The adapter supports both inbound and outbound interaction. Enterprise service discovery is used for discovery of services and creating the service description. It is also used to specify values for custom adapter properties and discovery of business objects. The WebSphere Adapter for Email supports integration through sending and receiving mails to and from different mail servers.

## Reference information

- WebSphere Adapter for Email User Guide
- Java Connector Architecture
  - ▶ <http://java.sun.com/j2ee/connector/index.jsp>
- Enterprise MetaData Discovery
  - ▶ <http://www.ibm.com/developerworks/java/library/j-emd/>
- WebSphere Adapter Information Center
  - ▶ <http://www-306.ibm.com/software/integration/wbiadapters/library/infocenter/>
- WebSphere Process Integration Information Center
  - ▶ <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp>



Additional reference information can be found at these addresses.

## 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:

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.

