



IBM Software Group

IBM WebSphere® Adapters V6.0.2

WebSphere Adapter for Flat Files V6.0.2



@business on demand.

© 2007 IBM Corporation
Updated March 6, 2007

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

Agenda

- **Overview**
- Enterprise metadata support
- Configuration properties
- Supported operations
- Summary and references

This section will provide an overview of the WebSphere Adapter for Flat Files. Note that the installation and deployment of the WebSphere Adapter for Flat Files is also covered in a separate presentation common for all WebSphere Adapters.

Overview: WebSphere Adapter for Flat Files

- IBM WebSphere Adapter for Flat Files 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 only through files
- Events and Responses captured as files on the file system
- When data transformation is involved the BOs are pre-generated, otherwise a fixed structure is used.



The IBM WebSphere Adapter for Flat Files implements the JCA 1.5 specification. It enables bi-directional connectivity, both inbound and outbound, with those Enterprise Information System business applications that can communicate only through files. Events and responses are captured as files on the file system. The WebSphere Adapter for Flat Files supports a single business object structure for both inbound and outbound. The file is seen as a byte array in the business object. The adapter is not aware of the contents of the file, only the file itself.

Agenda

- Overview
- **Enterprise metadata support**
- Configuration properties
- Supported operations
- Summary and references



This section will provide a brief detail on Enterprise Metadata Support

Enterprise service discovery



Enterprise Service Discovery in WebSphere Integration Developer implements the Enterprise MetaData Discovery Specification. The Enterprise Service Discovery wizard steps you through configuration of the adapter properties, service descriptions, and business object discovery, and results in the generation of the artifacts required for integration with SCA applications. With the WebSphere Adapter for Flat Files, there is no need for “discovery” of business objects; there is only one, pre-defined structure for the business object for both inbound and outbound processing. The adapter views the file as the data, and does not look inside the contents of the file. Therefore; service discovery for the Flat Files Adapter is used mainly to specify values for the properties used in activation specification, managed connection factory, and adapter properties.

Agenda

- Overview
- Enterprise metadata support
- **Configuration properties**
- Supported operations
- Summary and references

This section provides details of configurations properties of the WebSphere Adapter for Flat Files. More details on these properties can be found in the WebSphere Adapter Information Center. The link is provided in the Summary and references section.

Activation specification properties

Property	Description
EventDirectory	<ul style="list-style-type: none"> Directory where events are stored by the EIS system
EventFileMask	<ul style="list-style-type: none"> Specifies the filter for the event files (alphanumeric characters, wildcard *.*)
ArchivingProcessed	<ul style="list-style-type: none"> Determines whether archiving is performed, true or false
ArchiveDirectory	<ul style="list-style-type: none"> Directory where the processed event files would be archived
SortEventFiles	<ul style="list-style-type: none"> Determines the sorting order of event files being polled (filename, timestamp, if blank – no sorting)
SplitCriteria	<ul style="list-style-type: none"> Value set based on SplittingFunctionClassName. Type may be delimiter or size
SplittingFunctionClassName	<ul style="list-style-type: none"> Class name of the control class for splitting
FilePassByReference	<ul style="list-style-type: none"> Determines loading of file content or information on directory Name and filename
IncludeEndBODelimiter	<ul style="list-style-type: none"> When this property is set to true, the delimiter value specified in the SplitCriteria property is also sent with the business object content for further processing.
DefaultObjectName	<ul style="list-style-type: none"> Specifies a wrapper object or a content-specific business object
EventContentType	<ul style="list-style-type: none"> Represents the mime type of the event file
FileContentEncoding	<ul style="list-style-type: none"> Reading and writing of the file will be performed based on encoding setting

Many of the properties and values specified for inbound processing are stored in the activation specification instance during runtime processing. These properties include SplitCriteria, in which file splitting is performed based on either a delimiter or on a file size using the value specified. The delimiter value is set in the Split Criteria property and the class used to split the event file is set in the SplittingFunctionClassName property of the activation specification. This reduces memory loading during event processing.

Activation specification properties (cont.)

Property	Description
EPDataSourceJNDIName	▪ JNDI name of the data source used to get the JDBC connection
EPEventTableName	▪ Table name for event persistence
EPDatabaseSchemaName	▪ Name of the database used by event persistence
EPDatabaseUserName	▪ User Name to connect to the JDBC connection from the data source
EPDatabaseUserPassword	▪ User Password for the JDBC connection from the data source
EPCreateTable	▪ Determines if the event persistence table gets created automatically or manually. If the value is true and the event persistence table does not exist, then the adapter creates the table. If the value is false, then the adapter does not create the table and user must manually create the table.
FailedArchiveExt	▪ Specifies the file extension used to archive BOs that failed
SuccessArchiveExt	▪ Specifies the file extension used to archive successfully processed BOs
OriginalArchiveExt	▪ Specifies the file extension used to archive the original event file

Here are some properties for the Event Persistence table. Specify the Event Table name and the data source JNDI name. All remaining Event Persistence properties are optional. The value of the EPCreateTable property will determine if the event persistence table gets created automatically or manually. The adapter creates the event persistence table if the value is true. If the value is false, you must manually create the table.

Archive Extension properties are used to specify the file extension used to store processed event files. All archived events in your configured Archive Directory are stored with a "PROCESSED" file extension. The extensions of the files for SUCCESS and FAILURE are configurable based on the activation specification properties FailedArchiveExt, OriginalArchiveExt, and SuccessArchiveExt.

Managed connection factory properties

Property	Description
OutputDirectory	<ul style="list-style-type: none">Output directory for outbound operations
StagingDirectory	<ul style="list-style-type: none">When specified, the file will first be written to the StagingDirectory before it is written to the OutputDirectory
OutputFileName	<ul style="list-style-type: none">If this property is specified, the file created in the output directory will have this name. (This value will be overridden if a value for OutputFileName is specified in the record object.)

Managed connection factory configuration properties are used at run time to create an outbound connection instance with an enterprise information system. In addition to the OutputDirectory and OutputFileName properties, the StagingDirectory property has been added for V6.0.2. Once this property is specified, files are created in the staging directory before they are moved to the configured directory. This is helpful when a third application starts reading the file before it is fully written. The enterprise service discovery wizard refers to these properties as managed connection properties and WebSphere Process Server refers to these as (J2C) connection factory properties.

Resource adapter properties

Property	Description
Adapter ID	<ul style="list-style-type: none">Identifies a deployment instance of the adapter. If you are using more than one instance of an adapter, the value of this property must be unique.
enableHASupport	<ul style="list-style-type: none">When the enableHASupport property is set to true, only one of the replicated adapter instances actively polls for events while other instances are in standby mode. If the enableHASupport property is set to false, all of the adapter instances replicated on cluster members actively poll for events. This may result in event duplication. Do not change the value of enableHASupport to false for single server environments.

Resource adapter properties consist of logging and tracing and activities specific to the adapter. The AdapterID property lets you identify a deployment instance of the adapter. If you are using more than one instance of an adapter, you must specify a unique id for this property. One of the properties available for high availability is the enableHASupport property which supports multiple adapter instances in clustered environment. If enableHASupport is true, only one of the replicated adapter instances actively polls for events while other instances are in standby mode. If enableHASupport is set to false, all of the adapter instances replicated on cluster members actively poll for events. This capability improves adapter performance and availability. You configure these properties using the enterprise service discovery wizard or the administrative console of the server.

Agenda

- Overview
- Enterprise metadata support
- Configuration properties
- **Supported operations**
- Summary and references

This section will take a closer look at the supported operations for both inbound and outbound processing.

Supported outbound operations



Here are the supported operations for outbound services.
create, append, delete, retrieve, and overwrite are fairly self-explanatory.
exists – returns true if the specified file exists, and
list – returns a list of all file names in the directory specified

Outbound parameter passing

- Protocol specific fields
 - ▶ Output directory
 - ▶ Staging directory
 - ▶ File name
- Specified by
 - ▶ Administratively configured (Output directory and Staging directory only)
 - ▶ InteractionSpec
 - ▶ Included in the Wrapper BO (Output directory, Staging directory and FileName)

13

WebSphere Adapter for Flat Files V6.0.2

© 2007 IBM Corporation

In the outbound request processing mode, the protocol-specific fields output directory, staging directory and file name can be specified in three ways :

1. The staging and output directories can be configured from the administration console of WebSphere Application Server, which is in the JCA ManagedConnectionFactory instance.
2. All three properties can be passed through the InteractionSpec, or
3. They can be included in the Wrapper Business Object.

The precedence of the parameters is: WrapperBO , InteractionSpec, and ManagedConnectionFactory.

Data transformation in outbound and inbound

- Pass Through
 - ▶ FlatFileUnstructuredRecord
- Data Transformation
 - ▶ Actual data object (that is, Customer)
 - ▶ DataBindingMapping
 - ▶ Content type
 - ▶ Content-specific data binding (Data Transformation Mapping)

Both Outbound and Inbound flows can be broadly classified into 2 flows, one that involves data transformation and another without it (pass-through).

When the data object does not have any annotation and the name of this data object is UnstructuredContent, the DataBinding does a pass-Through. It just instantiates the FTPUnstructuredRecord, sets the actual content (byte[] or as text) and sets the protocol specific information.

When the WrapperBG specifies the actual data object (for example: Customer) and the annotation is set to DataBindingMapping, a data transformation will occur. Based on the ContentType set in the annotation of the Customer, that particular content-specific data binding is invoked. The mapping between the ContentType and the content-specific data binding to be called is obtained from the annotation (Data Transformation Mapping) of the wrapper data object and the FTPUnstructuredRecord is transformed to the Customer object.

FlatFileUnstructuredRecord

Property	Description
directoryPath	<ul style="list-style-type: none">Determines to which directory the output operation is being done
fileName	<ul style="list-style-type: none">Determines the name of the file to be written
listOutput	<ul style="list-style-type: none">An array where the list of files in the output directory are stored during the LIST operation
doesFileExist	<ul style="list-style-type: none">Boolean specifying whether a file is present or not
fileContentEncoding	<ul style="list-style-type: none">Determines the encoding when the output files are written
IncludeEndBoDelimiter	<ul style="list-style-type: none">During outbound Create/Append/Overwrite operations the file content is appended with the value of IncludeEndBoDelimiter
chunkFileName	<ul style="list-style-type: none">The presence of this indicates that it is a chunked file
StagingDirectory	<ul style="list-style-type: none">Used for operations like create, overwrite and append
BONamespace	<ul style="list-style-type: none">NameSpace of the BOs used.

The fields of the FlatFileUnstructuredRecord are used to determine the status of some of the Outbound Operations. These values are mapped into the WrapperObjects and returned to the J2EE client.

Supported inbound operations and features



- Read operation
- EmitFlatFile for non DTF and emitCustomerWrapper etc for DTF
- Optional Features to split large event files
 - ▶ SplitBySize
 - If SplitCriteria contains an integer and the SplittingFunctionClassName is equated to the SplitBySize class then chunking happens based on the value provided for SplitCriteria.
 - ▶ SplitByDelimiter
 - If SplitCriteria contains the delimiter and the SplittingFunctionClassName is equated to the SplitByDelimiter class then chunking happens based on the value provided for SplitCriteria.
 - Common delimiters include comma (,), semicolon (;), quote (" '), braces ({}), or slashes (/ \).
 - ▶ Each Chunk represents a Business Object
 - ▶ Reassemble
 - Chunk information included "chunkFileName" of the Business Object
 - Includes event id
 - Event id is in the form of event file location and timestamp – including current chunk number and total number of chunks.
- File Pass By Reference
 - ▶ Appends with a time-stamp and is archived to the Archive Directory
 - ▶ File name and directory path are sent to the end-point
 - ▶ Used where the applications have the capability to load and process the files on their own

16

WebSphere Adapter for Flat Files V6.0.2

© 2007 IBM Corporation

The inbound operation supported is the read operation. It reads the file from the event directory. An optional feature for inbound processing includes the capability to split large event files into chunks. Configuration of this functionality is specified in the properties SplittingFunctionClassName and SplitCriteria. If the input file is larger than the SplitCriteria property, then the file is split into chunks based on the value size in bytes specified or based on the specified delimiter. Each chunk represents a business object. To reassemble the chunks, information is included in the chunkFileName attribute of the business object. This includes event id. The event id is in the form of event file location and timestamp, including the current chunk number and total number of chunks. Reassembly could be handled by business logic implemented in a Java component, for example.

For certain applications that have inbuilt capability to load the file content by themselves, there is an option with the Flat File Resource Adapter to pass the files by reference. This will especially be useful in cases with large business objects where the content transfer would take up a lot of memory.

Agenda

- Overview
- Enterprise metadata support
- Configuration properties
- Supported operations
- **Summary and references**

This section will provide a summary of the presentation.

Summary

- WebSphere Adapter for Flat Files enables integration with SCA Applications and Enterprise Information System applications that can communication only through files
 - ▶ Inbound and Outbound support
- Provided support for Enterprise Service Discovery for discovering services
- Flat File Business Object is a pre-defined structure for non DTF flows
 - ▶ Adapter is unaware of the content of the file for pass through scenarios
 - ▶ For DTF flows, the adapter can split files based on a configured delimiter and the BOs are sent to a configured databinding.
- Optional feature splits large event files based on size or a specified delimiter.

In this presentation, you learned that the WebSphere Adapter for Flat Files enables integration with SCA business integration applications and Enterprise Information System applications that can communicate only through files in a file system. The adapter supports both inbound and outbound interaction. Enterprise service discovery is used for discovery services and creates the service description with specified values for custom adapter properties. The business object supported by the Flat Files adapter is of a single, pre-defined structure and the adapter is unaware of the content of the file. An optional feature for inbound processing of large event files includes the capability to split files based on size in bytes or a specified delimiter for data transformation framework (DTF) flows.

Reference information

- WebSphere Adapter for Flat Files 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 may be found at these addresses.

Provide feedback on this module

Your feedback is valuable

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

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

[Click here to provide e-mail feedback](#)



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

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, JDBC, 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.