



IBM Software Group

WebSphere® Process Server V6.0.2
WebSphere Integration Developer V6.0.2
WebSphere Enterprise Service Bus V6.0.2

WebSphere Adapter V6.0.2 Overview



@business on demand.

© 2006 IBM Corporation
Updated December 22, 2006

This presentation provides an overview of the IBM WebSphere Adapters for version 6.0.2.

Agenda

- General Overview of IBM WebSphere Adapters
 - ▶ Recap of V6.0 Adapters
 - ▶ New and Enhanced Adapters
- Enterprise Metadata (Service) Discovery
- Business Objects
- Generic Configuration for all Adapters
- Common Problem Determination
- Summary



This presentation covers a general overview of the IBM WebSphere adapters. It starts with a recap of the previously available adapters and then introduces the new adapters. The presentation also reviews the enhancements to the existing adapters, and then reviews the enterprise service discovery, showing the common sequence of installing and configuring an adapter. Then it discusses business objects and goes through some of the generic configuration for all adapters. At the end, there will be an overview of problem determination.

Section

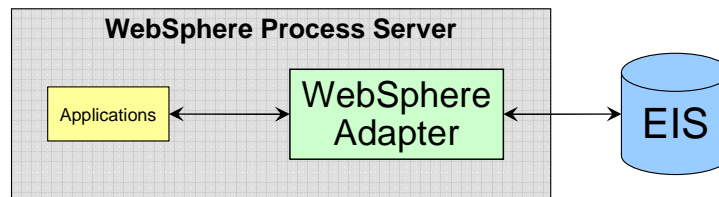
General overview of WebSphere Adapters



The next section covers the general overview of WebSphere Adapters.

Overview – Available in V6.0

- IBM WebSphere Adapters – currently available
 - ▶ IBM WebSphere Adapter For Flat Files V6.0
 - ▶ IBM WebSphere Adapter for JDBC™ V6.0
 - ▶ IBM WebSphere Adapter for SAP Applications V6.0
 - ▶ IBM WebSphere Adapter for PeopleSoft Enterprise V6.0
 - ▶ IBM WebSphere Adapter for Siebel Business Applications V6.0
- Implement J2EE Connector Architecture (JCA 1.5)
- Integrated with WebSphere Process Server and WebSphere Enterprise Service Bus V6.0
- Use Adapter Foundation Classes that enhance JCA 1.5 and encapsulate many common adapter functions



Here is the list of adapters that are available for version 6.0. They include the Flat File and JDBC technology adapters along with the SAP, PeopleSoft, and Siebel application adapters. These are all based on a set of foundation classes that enhance the JCA 1.5 specification. These foundation classes contain many new features for the 6.0.2 release.

What's new for WebSphere Adapters?



- **WebSphere Adapter for FTP**
 - ▶ Bi-directional integration to systems through exchange of remote flat files, supports FTPS (FTP over SSL)
 - ▶ Features include
 - Support for data transformation using structured business objects
 - Pass through using unstructured business objects
 - Inbound files parsed into individual business objects based on configured delimiter
 - Passing file by reference (only filename and directory name will be sent to endpoint)
- **WebSphere Adapter for e-mail**
 - ▶ Bi-directional integration to systems through exchange of e-mails using e-mailing systems
 - ▶ Features include inbound support for IMAP, POP3 and outbound support for SMTP protocols, support for data transformation of mail content and attachments, mail archiving on local system



The FTP and e-mail adapters have been added to the technology adapters for the 6.0.2 release. The FTP adapter provides bi-directional transfer of files to and from a remote EIS. It includes many new features for data transformation, inbound file parsing and passing a file by reference. The Email adapter allows exchange of information using an Email system. It includes support for IMAP, POP3, outbound support for SMTP, data transformation of mail contents and attachments and mail archiving on the local system.

What's new for WebSphere Adapters?



- WebSphere Adapter for JD Edwards EnterpriseOne
 - ▶ Outbound support to JD Edwards EnterpriseOne
 - ▶ Leverages the EnterpriseOne Dynamic Java Connector APIs to support business function and XML List calls
- WebSphere Adapter for Oracle E-Business Suite
 - ▶ Bi-directional integration with Oracle E-Business Suite using WebSphere Adapter for JDBC
 - ▶ Samples provided including
 - Outbound events for creating and retrieving customer information from Oracle EBS
 - Inbound events for retrieving customer data generated or modified in Oracle EBS,
 - Outbound event to process an Oracle EBS API call



There are also two new application adapters. The JD Edwards application provides outbound support to the JD Edwards EnterpriseOne environment. It supports business functions and XML list calls through the dynamic Java connector APIs provided by JD Edwards. The Oracle e-business application provides bi-directional integration using the JDBC adapter. Several samples are provided for creating and retrieving customer information.

What's enhanced for WebSphere Adapters?



- WebSphere Adapter for Flat Files
 - ▶ Bi-directional integration to systems through exchange of local flat files
 - ▶ New features include
 - Supporting for data transformation using structured business objects
 - Pass through using unstructured business objects
 - Inbound files parsed into individual business objects based on configured delimiter
 - Passing file by reference (only filename and directory name will be sent to endpoint)
- WebSphere Adapter for JDBC
 - ▶ Bi-directional integration to relational data sources supporting JDBC 2.0. Access tables, views, and stored procedures
 - ▶ New features include
 - Provided CLOB and BLOB data type support
 - Enhanced custom event query support for Inbound
 - Provided user-specified SQL processing support
 - Enhanced stored procedures support
 - Used WebSphere Process Server Datasource for database connectivity



Enhancements to the flat file technology adapter include data transformation, inbound file parsing based on a delimiter and passing a file by reference. The JDBC adapter has been enhanced to support CLOB and BLOB data, custom event query for inbound, user specified SQL processing, stored procedures and using the WebSphere datasource for database connectivity.

What's enhanced for WebSphere Adapters?

- WebSphere Adapter for Siebel Business Applications
 - ▶ Bi-directional integration to Siebel Business Applications by leveraging the Siebel Java Data Bean to support Application Service Interfaces (ASIs), including Integration Objects, Business Services, and Business Objects/Components (new)
- WebSphere Adapter for SAP
 - ▶ Provides integration to applications running on SAP Application Servers
 - ▶ Leverages the SAP interfaces for Business Application Programming Interfaces (BAPIs) and Intermediate Documents (IDocs) over Application Link Enabling (ALE)
 - ▶ Provides Hierarchical Dynamic Retrieval (HDR) of data from custom tables in SAP without requiring the development of custom BAPIs or IDocs to access the data
- WebSphere Adapter for PeopleSoft not updated
- Runtime support for Z/OS platform



Using Siebel Java Data Bean application service interfaces the Siebel Adapter can access integration objects, business services and business objects or components. The SAP adapter leverages the SAP interfaces for business application programming interfaces and intermediate documents over application link enabling. It also provides hierarchical dynamic retrieval of data from custom tables in SAP without requiring the development of custom BAPIS or IDocs. The PeopleSoft adapter was not updated for the 6.0.2 release but the existing 6.0 version should run on 6.0.2 without any problems.

New features

- **Data Transformation Framework**
 - ▶ Allows for transformation of Business Objects to and from semi-structured formats like XML and fixed-width.
 - ▶ Provides for user-configured data binding
 - Specify during Enterprise Service Discovery
 - Provided XMLBoSerializerDataBinding or your own custom data binding
- **High Availability Support – Failover**
 - ▶ Will allow adapter instances to be marked as Singleton within a cluster. At any given point of time only one adapter instance would be looking for events from an event store.
 - ▶ New “Enable HA Support” property on adapter
 - ▶ Requires enhanced support from WebSphere ND (updated fix pack)



There are many new features that are provided in the foundation classes that will be covered in the next few slides. The data transformation framework is enabled for the Flat File, FTP, and E-mail adapters. This allows the transformation of business objects to and from a semi-structured format, like xml or fixed-width, into a structure that the business object of the client application in WebSphere Process Server will understand. The user-configured data binding can be selected during enterprise service discovery using either the XMLBoSerializerDataBinding that is provided, or a custom data binding created by a developer.

High availability support or failover has been added for all adapters. This will allow multiple adapters to register with the HA manager and if the current adapter or WebSphere Process Server process fails, the HA manager can select a new instance of the adapter to handle incoming notifications. Only one instance of an adapter will be active at any given point in time to receive notifications although all instances of the registered adapters will be running. The adapter becomes HA capable by setting the “enable HA support” property. HA support for adapters requires either WebSphere version 6.0.2 or a fix pack added to 6.0.1.

The HA support and the data transformation framework will be covered in more detail in another presentation.

New features

- **Additional Filtering Capabilities**
 - ▶ **EventTypeFilter**
 - Delimited list of event types the adapter should deliver. Events not in the list will not be delivered. If empty (null) all events are delivered. Default null
 - ▶ **FilterFutureEvents**
 - If True, adapter inspects timestamp on each event and compares to system time. If event time is ahead, event will not be delivered. Default false
- **Continue Polling After Event Processing Exception**
 - ▶ **StopPollingOnError**
 - Currently, WebSphere Adapters stop polling when an exception occurs during event processing
 - StopPollingOnError is new activation spec property to allow continued polling
 - Aligns behavior with WebSphere Adapters. Default false



Additional filtering capabilities have been added for the EventTypeFilter and FilterFutureEvents. The EventTypeFilter allows a delimited list of event types that the adapter will deliver to the client application. If FilterFutureEvents is set to true then the adapter will not deliver events that are ahead of the system time. The StopPollingOnError has been added to allow polling to continue polling when an exception occurs. The default is false which works as the adapters did in the 6.0 version, stopping polling when the exception occurs.

New features

- Automatically retry inbound connection if EIS is offline
 - ▶ **RetryInterval**
 - If non-negative, and adapter encounters error related to inbound connection to the EIS, adapter will continue to reattempt a connection after the specified interval until a connection is re-established. Default 60000 ms
 - ▶ **RetryLimit**
 - Specifies number of times the foundation classes should attempt to retry the connection to the underlying EIS until giving up. Value of 0 will cause unlimited attempts. Default 0 (unlimited)



The adapters will now automatically retry the inbound connection if the EIS is offline. The `RetryInterval` and `RetryLimit` specify how often and how many retries should occur.

New features

- **Multiple Endpoint Support**
 - ▶ **Delivery of the same event to multiple applications**
 - Ensure all properties in the individual Activation Specs are identical
 - An adapter treats Activation Specs that contain different values as having independent events which will lead to competition among different endpoints and can cause lost events
 - Adapter will broadcast a copy of the event to all “active” subscribed endpoints
 - For true publish/subscribe functionality, better to employ features provided in underlying ESB architecture
 - Another usage of this support is to run Enterprise Metadata Discovery multiple times, specifying EventTypeFilter property for different event types each time
 - Multiple exports in same module
 - Using Event Type Filtering in this way will result in one Activation Spec for Customer, one Activation Spec for Order, for example



The adapters are now capable of delivering the same event to multiple endpoints. To accomplish this, all properties for the individual activation specs must be identical. The adapter will broadcast a copy of the event to all active subscribed endpoints. By running ESD multiple times and specifying a different event type each time, the same event can be filtered by the adapter after it is broadcast to all active subscribed endpoints.

New features

- Improved First Failure Data Capture (FFDC) Support
 - ▶ provide information on objects of interest when exceptions occur
- Provide support for Performance Monitoring Infrastructure (PMI)



The adapters now support FFDC when exceptions occur and provide performance monitoring using PMI.

New features

- **Event Management Framework**
 - ▶ **Assured Event Delivery**
 - Previously, two tables were needed, event distribution table (EDT) and the event store. EDT is deprecated.
 - Now, single event table contains all of the information (new XID column)
 - Easier configuration using WebSphere data source for configuration (JNDI name, username and password for data source, table name)
 - Siebel: need to add new XID Fields into event table



The event management framework has removed the use of the event distribution table. Only the event table is required to insure once and once only event delivery. The single event table now contains a XID column. The configuration is easier using the WebSphere datasource for the JNDI name, username and password for the data source and table name. For Siebel, the new XID fields will need to be added into the event table.

Section

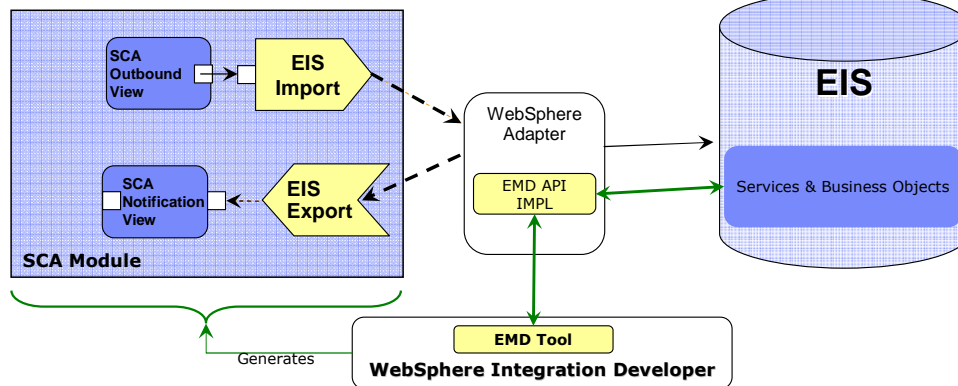
Enterprise metadata (service) discovery



The next section covers the Enterprise Metadata Discovery function. Enterprise Metadata Discovery is a joint specification from IBM and BEA.

Enterprise metadata (or service) discovery

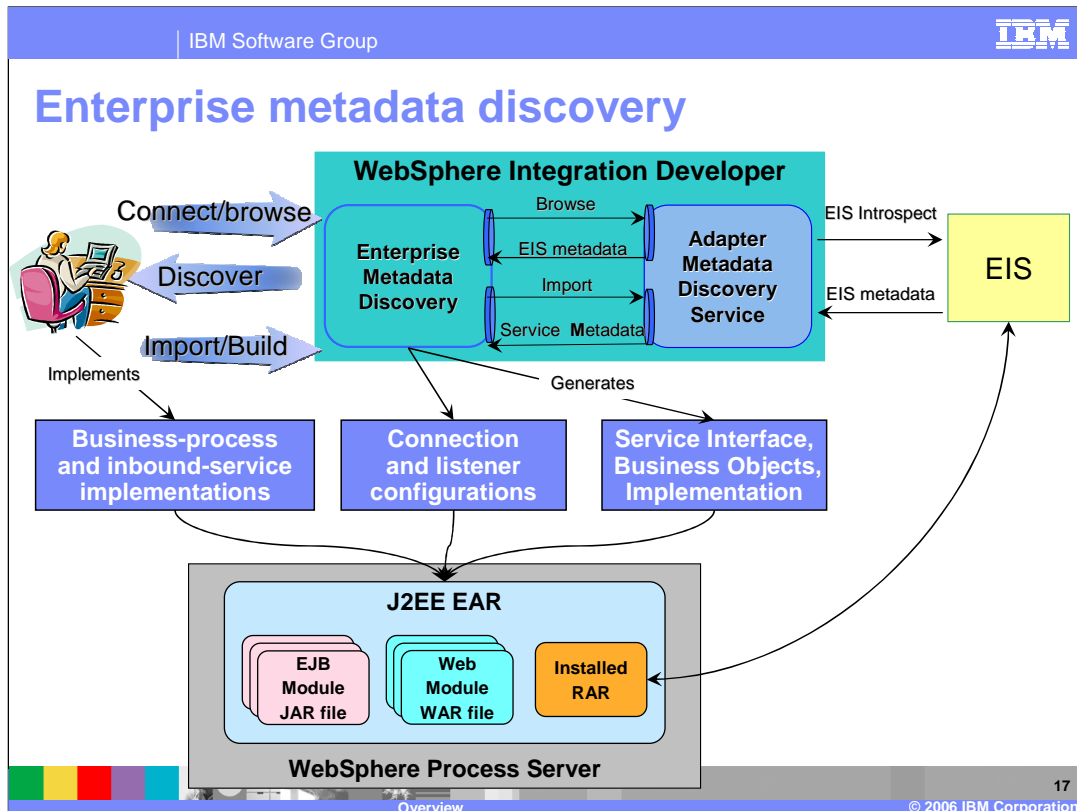
Overview – The Solution: Enterprise Service Discovery API



- Define a common API that adapters can use to expose their services and business objects to tools for the generation of J2CA based applications.
- Implementation of the API is a simple extension to the capabilities of the J2CA resource adapter.
- Allows a single tool environment to support development for multiple EIS systems through their Adapters
- Transforms application development from a top-down activity to a discovery exercise.

The diagram shows the different components of the Adapter and the WebSphere Integration Developer Tool that make the EMD solution. The EMD Tool within WebSphere Integration Developer calls the EMD APIs in the Adapter to introspect the backend EIS. Based on the selection made in the EMD wizards, the appropriate SCA components for EIS Import or EIS Export are generated, along with the Business Objects.

EMD defines a common API for adapters to use to expose the EIS services and business objects. The EMD specification is an extension to the JCA specification. Using the EMD Tool makes developing business applications that interact with EIS a simple discovery process, rather than coding to the J2C APIs.



Enterprise Metadata Discovery (EMD) is a tool that allows generation of SCA components and business objects from the business services and integration objects in the EIS. The tool uses the EMD classes within the adapter to introspect the EIS.

The business objects contain the metadata corresponding to the EIS business services, and are used by the adapter.

Using the business objects, the EMD generates SCA constructs for the adapter. Using the SCA constructs, the developer can create the SCA components to interact with the EIS. The SCA components can then be used in a business process flow.

EMD component is equivalent to ODA (Object Discovery Agent) from legacy adapters. It is a little more than ODA since, besides Business Objects, it also generates the artifacts needed by SCA like Import/Export file and WSDLs.

Interaction styles

- There are two types of interactions:
 - ▶ **Outbound:** Client initiated
 - ▶ **Inbound:** EIS-initiated
- There can be two modes of interaction for each interaction style
 - ▶ Request-Response – For Outbound interaction
 - Request/Response interaction takes a request and returns a response
 - ▶ One-way - For Inbound interaction
 - Takes a request, but does not return a response
- Interaction style is specified during discovery service within Integration Developer
 - The generated request-response method for outbound has both input and output specified, whereas for One-way method for inbound interaction has only input arguments specified

An outbound request or interaction is initiated by a client to the EIS. The client could be an SCA component within the same module or some other module, or it could be some external client like a JSP.

An inbound event notification is initiated by the backend EIS based on some event on the EIS business object, like create, update or delete. The adapter gets the event either by push or pull, based on how the specific adapter interacts with the EIS. It then fetches the business object from EIS before sending that to the listener that will handle the event.

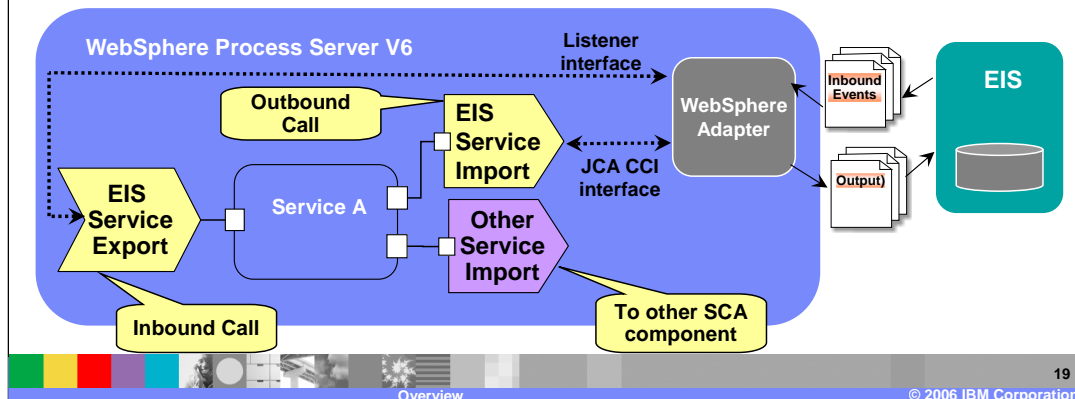
The outbound interaction is a request-response interaction, where usually the copy of the business service is returned.

The inbound interaction is one way only where the event gets passed to the adapter.

The interaction style (inbound or outbound) is selected during the discovery service process within WebSphere Integration Developer.

EIS import and export services

- EIS Import (for outbound) and EIS Export (for inbound) SCA components provide a uniform view of the EIS services external to the module
- Business Objects are generated by service discovery and are used by SCA components and the adapter
- This allows components to communicate with the variety of external EIS systems using a consistent SCA programming model
- Interacting with the EIS systems through the use of SCA components, Business Objects, and adapters fit the goals and the vision of SOA solutions



The Enterprise Service Discovery Tool in WebSphere Integration Developer creates an EIS Import SCA component for an outbound request, and creates EIS Export SCA component for an inbound event request. The Business Objects for the outbound or inbound requests are also created. Using the SCA components for the adapter, they can be wired with other SCA components to create a business application.

In the diagram, the SCA component representing “Service A” is wired with the Adapter EIS Export and EIS Import component. Also shown is the wiring from Service A to other SCA components through the Import. The implementation of Service A could be BPEL or any other support implementation like POJO, Human task and so on.

For SCA clients, the adapter functionality is exposed through the EIS Import and EIS Export SCA components.

Development: General steps

- Import the WebSphere Adapter RAR file with WebSphere Integration Developer
- Run the Enterprise Service (Metadata) Discovery tool
 - ▶ Discovers Service and Data
 - ▶ Generates Business Objects, EIS SCA components, import and export files, WSDL and other required artifacts
- Assemble SCA Solution using the EIS SCA components
- Export as an Enterprise Application Archive (EAR) file and install to WebSphere Process Server

20

Overview

© 2006 IBM Corporation

The high level steps of the development of business applications using adapters is shown here.

The Adapter RAR file is imported in WebSphere Integration Developer. The developer runs the Enterprise Service Discovery tool making the appropriate selection in the wizards. Based on whether outbound or inbound request was selected, the tool generates EIS Import SCA component for outbound, EIS Export SCA component for inbound, along with the Business Objects, WSDL files and other artifacts.

The EIS SCA components can then be wired with other SCA components that make up the business application.

The SCA modules bundled in a J2EE EAR file is then exported from WebSphere Integration Developer and deployed to a WebSphere Process Server for execution.

Other presentations will go into the details of these steps.

Section

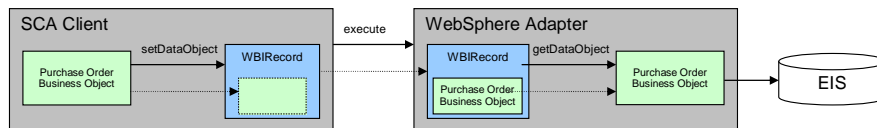
Business objects



The next section covers the Business Objects and how they are used as the core component in the WebSphere Adapter architecture.

Business objects

- For outbound operations, Business Objects are used to describe the structure and content of arguments to functions on the EIS client interface or the EIS business objects
- However, CCI client and adapter exchange data through the Record model, as part of JCA specification
 - *Record execute(InteractionSpec ispec, Record inRecord)*
- Business Object model is not compatible with Record model
- Solution: Adapter Foundation classes provide a Record implementation that wraps the Business Object to allow it to be passed between the client and the resource adapter



22

Overview

© 2006 IBM Corporation

The SCA client uses Business Object to pass data between the client and the WebSphere Adapter. For an outbound request, the Business Object describe the function or the business service to call in the backend EIS. The Business Object also contains metadata representing the arguments to be passed to the EIS function.

However, the J2C specification does not understand Business Object. They data between a JCA client and JCA Adapter is passed using the common client Interface Record object. Hence the Business Object model is not compatible with the JCA Record object. The problem is solved by wrapping the Business Object inside the Record object and passing the Record containing the wrapped Business Object to the adapter. The diagram shows the interaction between the SCA client and the Adapter. The Adapter extracts the Business Object from the Record object and determines the EIS function to invoke, passing the parameters that are in the Business Object as metadata.

The data structure for all WebSphere adapters for outbound requests is a business object wrapped in a WBIRRecord implementation.

The parameters of the operation, for any J2CA adapter, are defined through an adapter-specific InteractionSpec instance; this class can contain 0..n properties that specify details about the operation to perform. In the case of WebSphere Adapters, it provides a default WBIRInteractionSpec class which has one property: "FunctionName". It is in this property that invoking components should set the operation to perform (note: this is different than the verb which is defined in the actual business object). All adapters are strongly encouraged to use this InteractionSpec class.

Advantages of using business objects

- Business Object model is based on the open-standard service data object (SDO) model, and provides extensions and additional functionality to make this model more robust
- Business Object model aligns well with the larger WebSphere Service-Oriented Architecture (SOA) strategy
- Business Object model provides support for tracking changes at both object- and property-level to better enable delta update functionality



The Business Object model and the Service Component Architecture are the core components of the Business process application following the Service Oriented Architecture principles.

Section

Common configuration for all adapters



The next section covers the common configuration that applies to all the WebSphere Adapters.

Resource adapter configuration

- JCA related configuration artifacts are created as a result of service discovery process in WebSphere Integration Developer
- Properties may be modified while still in Integration Developer, or after the application has been deployed to the WebSphere Process Server through Administrative Console or wsadmin
 - ▶ ResourceAdapter
 - global configuration properties
 - ▶ ManagedConnectionFactory
 - properties for a target EIS instance (outbound)
 - examples: hostname, port
 - ▶ ConnectionSpec
 - connection instance properties (outbound)
 - examples: user name, password, language
 - ▶ InteractionSpec
 - per request configuration properties (outbound)
 - examples: operation to be invoked, return values required
 - ▶ ActivationSpec
 - properties configurable by a specific endpoint (inbound)
 - examples: durability, business objects of interest



The developer uses the Enterprise Service Discovery wizards to create the outbound or inbound request to and from backend EIS. The JCA properties for the Resource Adapter, Managed Connection Factory for outbound request and Activation Spec for inbound request are already created based on the input to the service discovery wizard.

Once the application containing the RAR is deployed, the administrator can modify the properties using the Administrative console or the command line "wsadmin" tool.

Connection properties

Inbound Connection Properties

<i>Name</i>	<i>Description</i>
Resource Adapter	Resource Adapter and its custom properties
ActivationSpec	Adapter Activation Spec.

Outbound Connection Properties

<i>Name</i>	<i>Description</i>
Resource Adapter	Resource Adapter and its custom properties
Managed Connection Factory	Adapter Managed Connection Factory

Each Adapter may have their own additional properties

As indicated in the previous page, the properties for the Inbound request are in the Resource Adapter and the ActivationSpec, whereas the properties for the Outbound request are in the Resource Adapter and the Managed Connection Factory.

Common adapter configuration properties for both inbound and outbound

Name	Description
Hostname	Name of the machine hosting the EIS. (Name or IP address)
Port	Port on which the EIS instance is listening.
Username	User name for logging into the EIS for inbound events
Password	User password for authorizing the user to retrieve events from the EIS
Timeout	Amount of time in milliseconds to wait for a response from the EIS
EISEncoding	Used if adapter needs to convert unicode to/from native encoding like in case of file handling etc.. This should specify the native encoding to use.

This table lists the common properties for outbound and inbound requests needed for the Enterprise Service Discovery tool to connect to the EIS. These are also used as the attributes on the Resource Adapter. The EISEncoding is used to convert to and from the Unicode native encoding.

Common adapter logging and tracing properties

<i>Name</i>	<i>Required</i>	<i>Description</i>
AdapterID	Yes	The id of this deployment of the adapter
LogFilename	No	The full path of the log file
LogNumberOfFiles	No	The number of log files to use. When a log file reaches its maximum size it will start using another log file. Default = 1
LogFileMaxSize	No	Size of the log files in kilobytes. Default = No maximum size
TraceFilename	No	The full path of the trace file
TraceNumberOfFiles	No	The number of trace files to use. When a trace file reaches its maximum size it will start using another trace file. Default = 1
TraceFileMaxSize	No	Size of the trace files in kilobytes. Default = No maximum size

The logging and tracing properties for all the WebSphere Adapters are listed here. The file name, maximum file size and the number of history log files are the main options for files for logging and tracing.

Common ActivationSpec attributes for inbound events

Name	Required	Description
PollPeriod	Yes	The rate (in milliseconds) at which to poll the EIS event store for new inbound events. Default: 500
PollQuantity	Yes	# of events per poll cycle for each endpoint. Default: 1
DeliveryType	No	Determines the order in which the events will be published. One at a time (ORDERED) or all at once (UNORDERED). Default: ORDERED
StopPollingOnError	No	If True, adapter stops polling when it encounters an error during polling. If False, adapter logs exception and continues polling. Default False
EventTypeFilter	No	Delimited list of event types the adapter should deliver. Events not in the list will not be delivered. If empty (null) all events are delivered. Default null
FilterFutureEvents	Yes	If True, adapter inspects timestamp on each event and compares to system time. If event time is ahead, event will not be delivered. Default false

For adapters that poll the EIS system for new events, the poll period and number of events per poll cycle can be customized. StopPollingOnError allows the adapter to continue polling when an error occurs. The eventTypeFilter is a delimited list of event types that an adapter will deliver. FilterFutureEvents will restrict delivery of events that are ahead of the system time.

Common ActivationSpec attributes for inbound events

Name	Required	Description
AssuredOnceDelivery	Yes	If True, adapter stores XID information in the EventStore. If False, will not. True required for 'assured once event delivery'. MDB must be transactional. Default True.
RetryInterval	No	If non-negative, and adapter encounters error related to inbound connection to the EIS, adapter will continue to reattempt a connection after the specified interval until a connection is re-established. Default 60000 ms
RetryLimit	No	Specifies number of times the foundation classes should attempt to retry the connection to the underlying EIS until giving up. Value of 0 will cause unlimited attempts. Default 0 (unlimited)

AssuredOnceDelivery enables the use of the XID information in the event store. This must be true for once and once only delivery. The retry interval and limit specify how connection retry to the EIS are handled.

Section

Problem determination: Common for all WebSphere adapters



The next section covers common problem determination for all WebSphere Adapters.

Problem determination: Log and trace files

- WebSphere Process Server
 - ▶ <WPS_Home>\profiles\<profile>\logs\<server>\SystemOut.log
 - ▶ <WPS_Home>\profiles\<profile>\logs\<server>\SystemErr.log
 - ▶ <WPS_Home>\profiles\<profile>\logs\ffdc\xxxxxx.log

- Adapter Logs and Traces Files
 - ▶ Messages and Exceptions logged in these files
 - ▶ Log and Trace file location and levels specified during EMD
- EMD log, specified during EMD
 - ▶ Default: <WID_workspace>\.metadata\<Adapter>MetadataDiscovery.log
- WebSphere Integration Developer
 - ▶ <WID_workspace>\.metadata\.log



The location of the log and trace files are specified here.

The WebSphere Process Server log files are the System out and System error log files in the profile logs directory of the server.

Then there are the adapter log and trace files. The trace file locations are specified using the log and trace file attributes on the adapter, as discussed before. In addition, the trace files location and trace strings are specified in the Administrative console of the Process Server for the server's change log and trace file option.

While running the Enterprise Metadata or Service Discovery tool, the log file is within the WebSphere Integration Developer workspace in the directory specified.

The overall log file for WebSphere Integration Developer is in the workspace metadata log file.

Problem determination: Logging

- User messages and Language translated

Logging Level	Content
Fatal	Task cannot continue. Component cannot function.
Severe	Task cannot continue. Component can still function. This also includes conditions that indicate an impending fatal error, i.e., reporting on situations that strongly suggest that resources are on the verge of being depleted.
Warning	Potential error or impending error. This also includes conditions that indicate a progressive failure, i.e., the potential leaking of resources.
Audit	Significant event affecting server state or resources
Info	General Information outlining overall task progress
Config	Configuration change or status
Detail	General Information detailing subtask progress

The different logging levels are specified on this page. These messages are translated and they appear in the WebSphere Process Server System out log files.

Problem determination: Tracing

- Service level information for IBM Support
 - Not translated
 - Specified at the Adapter level through its custom properties

Tracing Level	Content
Fine	Trace info – general trace, plus method entry / exit / return values
Finer	Trace info – detailed trace
Finest	<ul style="list-style-type: none">▪ Trace info – most detailed trace▪ Includes all detail needed to debug problems

The trace strings are more geared towards the IBM Support. The different tracing levels are specified on this page. These are specified at the Adapter through its custom properties.

Section

Service update process

The next section covers common configuration that applies to all the WebSphere Adapters.

Service update process

- All adapters are part of WebSphere Integration Developer V6.0.2
 - ▶ (WebSphere Integration Developer Root)/Resource Adapters
 - ▶ Adapters are not listed as installed components
 - ▶ Adapters provided to WebSphere Integration Developer as a ZIP containing all
- Adapter fix pack aligns with WebSphere Integration Developer
 - ▶ Use Rational® Product Updater
- Adapter fix pack does not align with WebSphere Integration Developer
 - ▶ Delivered as IFIX by WebSphere Integration Developer per WebSphere Integration Developer process
 - ▶ Adapters IFIX delivered as a ZIP containing all

The adapters are now being delivered as part of WebSphere Integration Developer version 6.0.2 in the WebSphere Integration Developer root/resource adapter directory. The adapters are not listed as installed components within WebSphere Integration Developer. The adapters are provided to WebSphere Integration Developer as a zip file which contains all of the adapters in the directory structure needed. Fix packs will be applied using the Rational Product Updater if the adapter fix pack aligns with the release of a WebSphere Integration Developer fix pack. If it does not align then the IFIX will be delivered by WebSphere Integration Developer using the WebSphere Integration Developer process.

Section

Summary and reference

The next section will provide the Summary and References.

Summary

- This presentation covered an overview of the new and enhanced IBM WebSphere Adapters
 - ▶ Enterprise Metadata or Service Discovery and creation of EIS SCA components
 - ▶ Business Objects
 - ▶ Common configuration properties
 - ▶ Problem Determination log and trace levels and file locations
- Adapter Foundation classes provide the foundation needed for adapter implementation



This presentation covered an overview of the new and enhanced IBM WebSphere Adapters for the 6.0.2 release. You have reviewed the changes in the enterprise service discovery that is now provided as part of the WebSphere Integration Developer. Discussing the changes in support of business objects and the common configuration property changes. The common problem determination by using the log and trace files provided details of how to configure the different levels of messages that can be provided. Much of the common functionality has been moved into the adapter foundation classes.

References

- Java Connector Architecture
 - ▶ <http://java.sun.com/j2ee/connector/index.jsp>
- Enterprise MetaData Discovery Whitepaper
 - ▶ <http://www.ibm.com/developerworks/java/library/j-emd/>



This page lists all the references that can be helpful for additional resources.

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 Rational WebSphere

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

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 NONINFRINGEMENT. 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 2006. 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.

