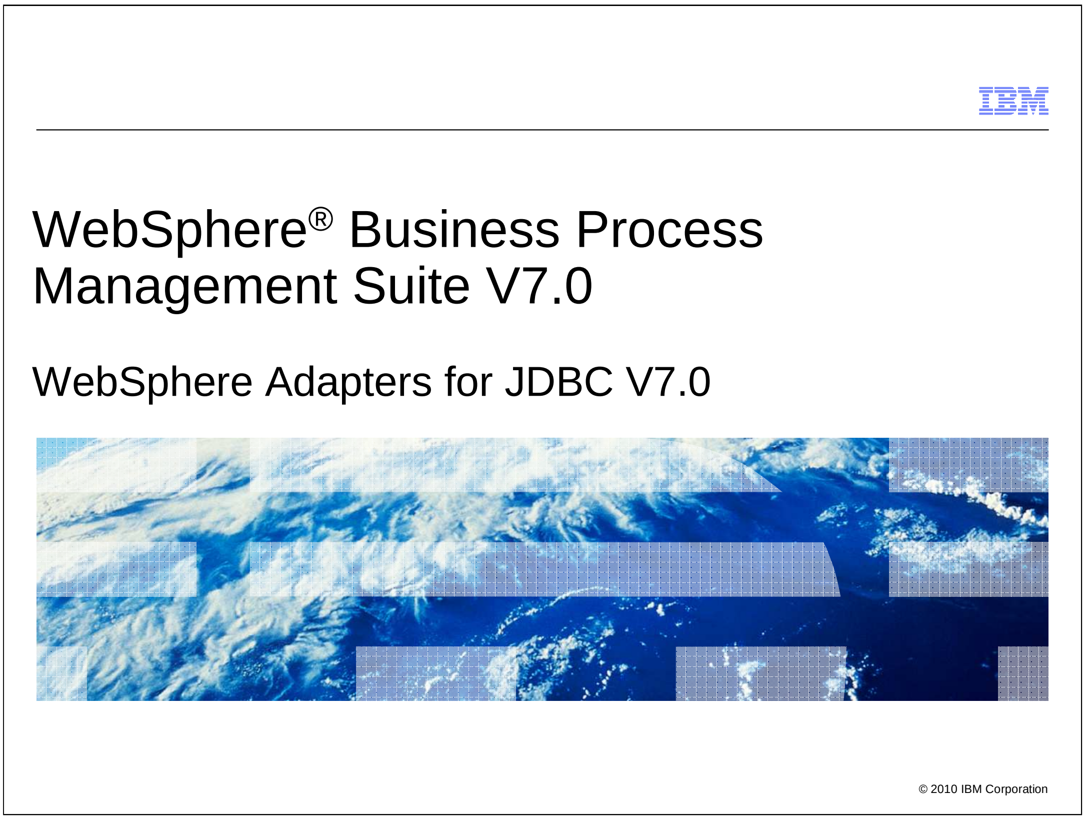
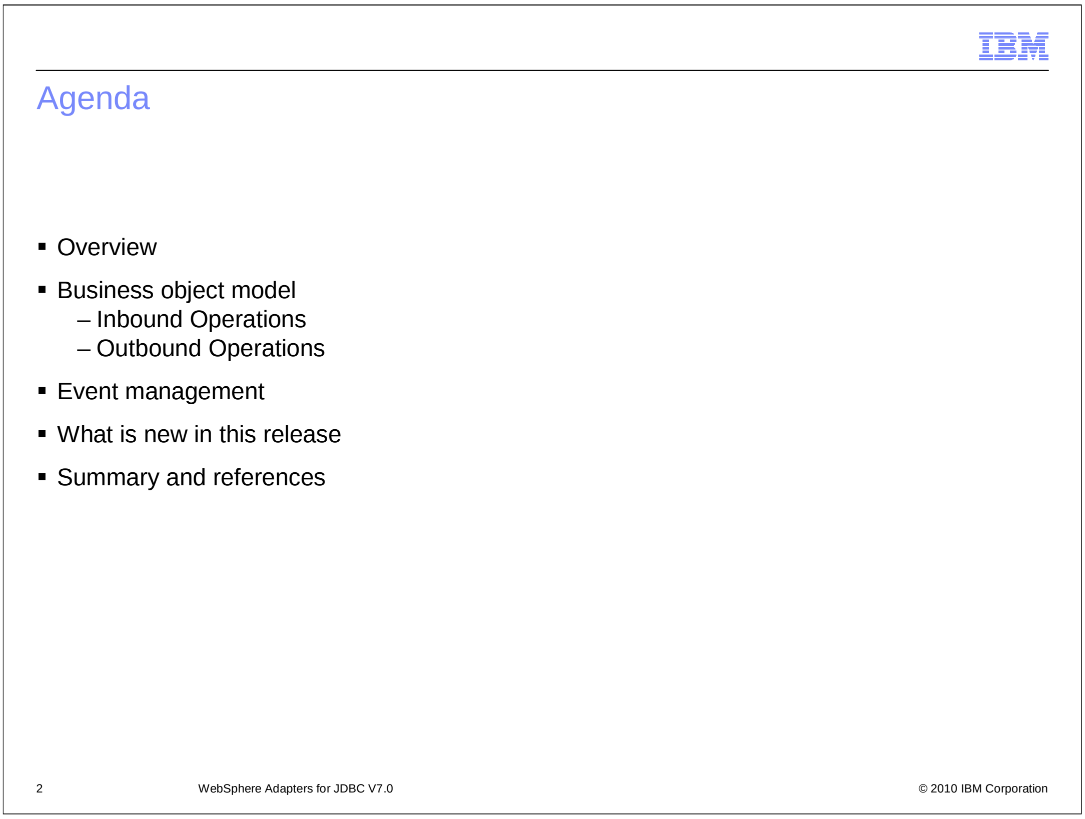


WebSphere® Business Process Management Suite V7.0

WebSphere Adapters for JDBC V7.0

This presentation provides a general overview of the IBM WebSphere Adapters V7.0



Agenda

- Overview
- Business object model
  - Inbound Operations
  - Outbound Operations
- Event management
- What is new in this release
- Summary and references

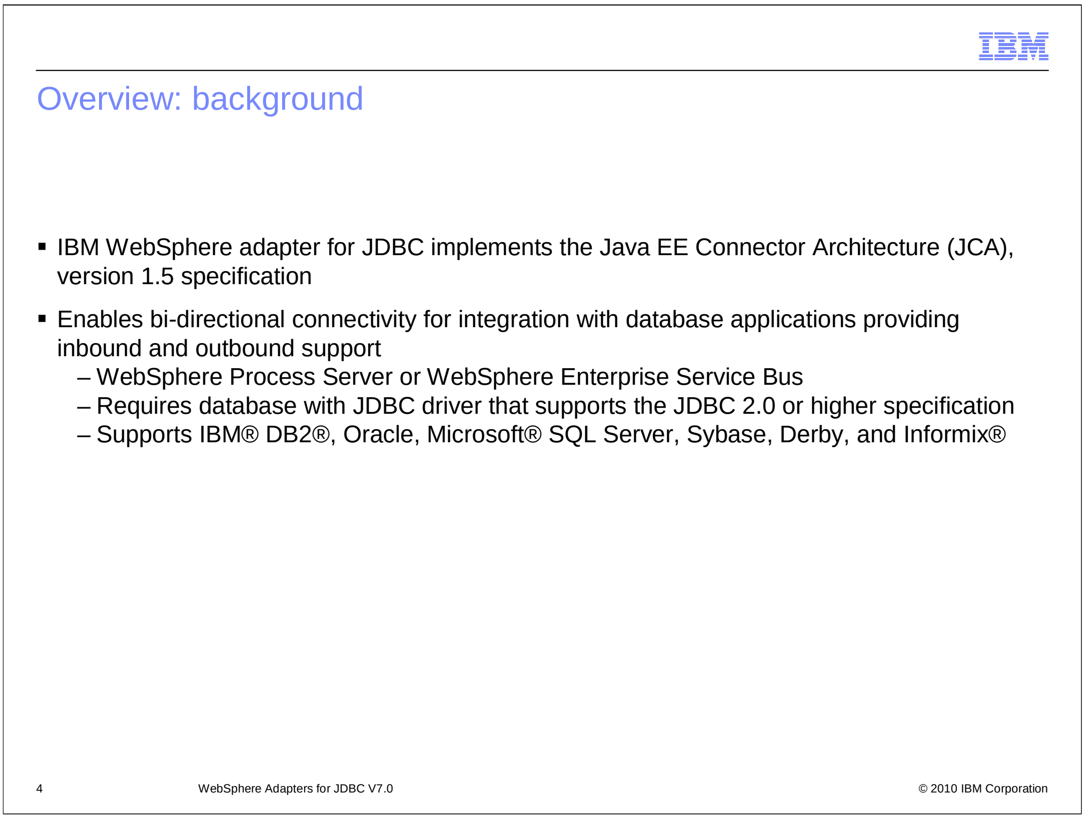
2 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

This presentation covers details of the IBM WebSphere Adapters. It starts with reviewing the new enhancements in the development and deployment process of JDBC Adapter. The last section provides information about event management in inbound processing. For more information about the new and improved external service wizard, refer to the JDBC demonstrations.

Section

Overview

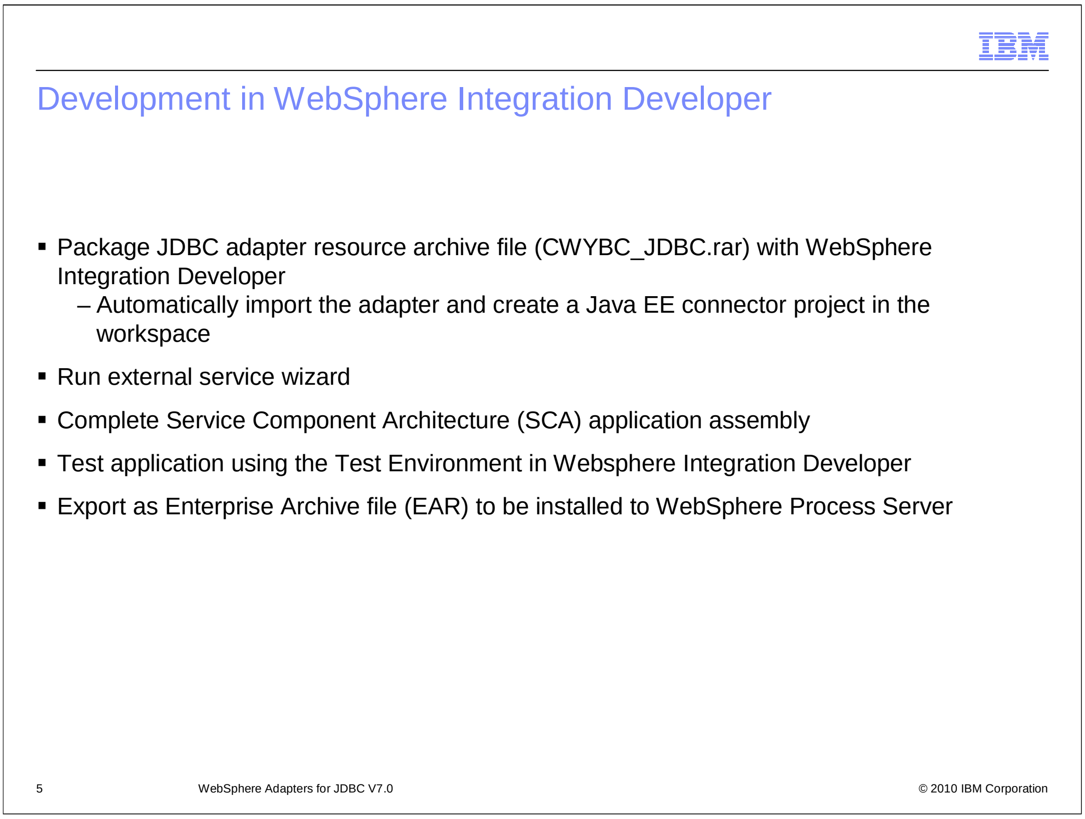
This section provides an overview of the new enhancements in deployment of the WebSphere Adapter for JDBC V7.0



Overview: background

- IBM WebSphere adapter for JDBC implements the Java EE Connector Architecture (JCA), version 1.5 specification
- Enables bi-directional connectivity for integration with database applications providing inbound and outbound support
  - WebSphere Process Server or WebSphere Enterprise Service Bus
  - Requires database with JDBC driver that supports the JDBC 2.0 or higher specification
  - Supports IBM® DB2®, Oracle, Microsoft® SQL Server, Sybase, Derby, and Informix®

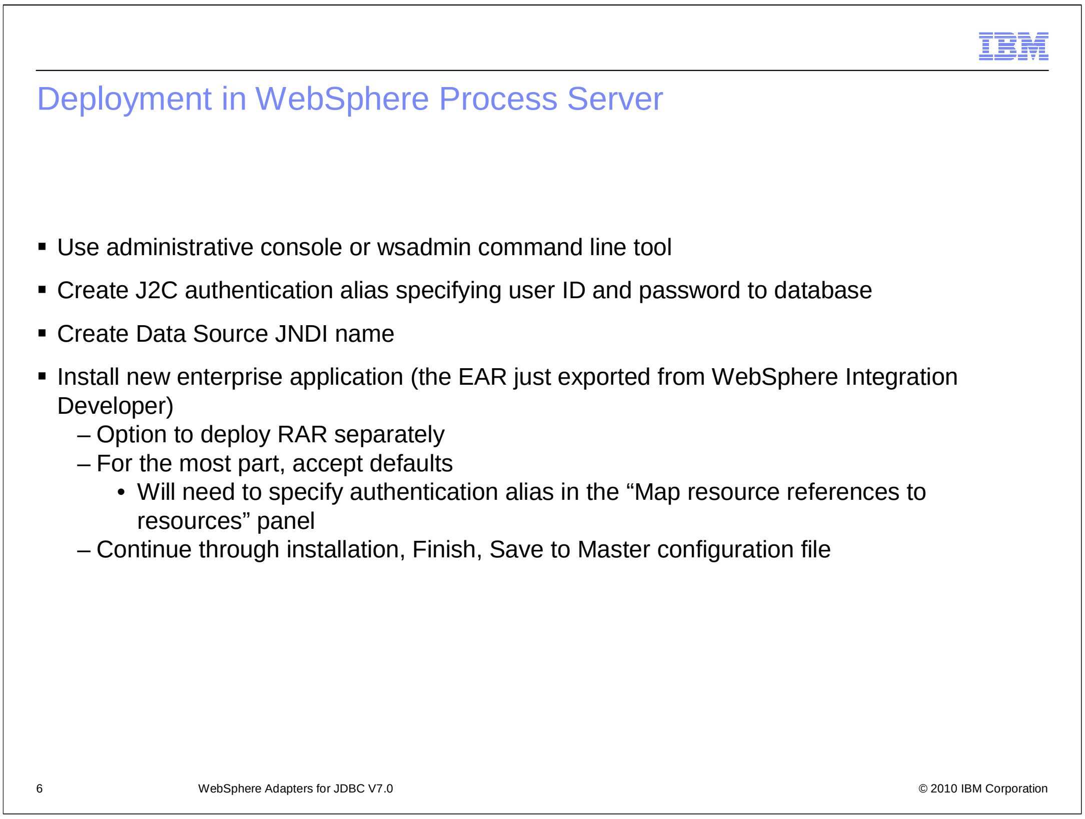
The IBM WebSphere Adapter for JDBC implements the JCA 1.5 specification and enables two-way communication, both inbound and outbound, between an application running on WebSphere Process Server or WebSphere Enterprise Service Bus and a database. The adapter provides a standard interface that integrates with diverse database software vendors and versions; it supports any database server with a Java™ Database Connectivity (JDBC) driver that supports the JDBC 2.0 or later specification. Examples of such servers IBM® DB2®, Oracle, Microsoft® SQL Server, Sybase, Derby, and Informix®. The adapter uses business objects to exchange data between the application and the database, so the application does not need to use the JDBC application programming interface (API).



Development in WebSphere Integration Developer

- Package JDBC adapter resource archive file (CWYBC_JDBC.rar) with WebSphere Integration Developer
  - Automatically import the adapter and create a Java EE connector project in the workspace
- Run external service wizard
- Complete Service Component Architecture (SCA) application assembly
- Test application using the Test Environment in Websphere Integration Developer
- Export as Enterprise Archive file (EAR) to be installed to WebSphere Process Server

You can import the JDBC Adapter RAR file, which is available in the Resource Adapters directory, into WebSphere Integration Developer. Importing this file creates a Java EE connector project in your workspace. Using the external service wizard, you then enter the adapter configuration information and select the service type, either inbound or outbound. Next, you select the particular operations, the logging and tracing file names, inbound and archive subdirectory names, and other adapter specific properties. Upon completion of the external service wizard, you are returned to the business integration perspective. There is a module containing the necessary adapter artifacts for integration and assembly with other Service Component Architecture, or SCA, components. When you complete your application assembly, the application can be tested and debugged within the WebSphere test environment. It then exported as an Enterprise Archive, or EAR file, and ultimately installed into a stand-alone WebSphere Process Server or WebSphere Enterprise Service Bus runtime.

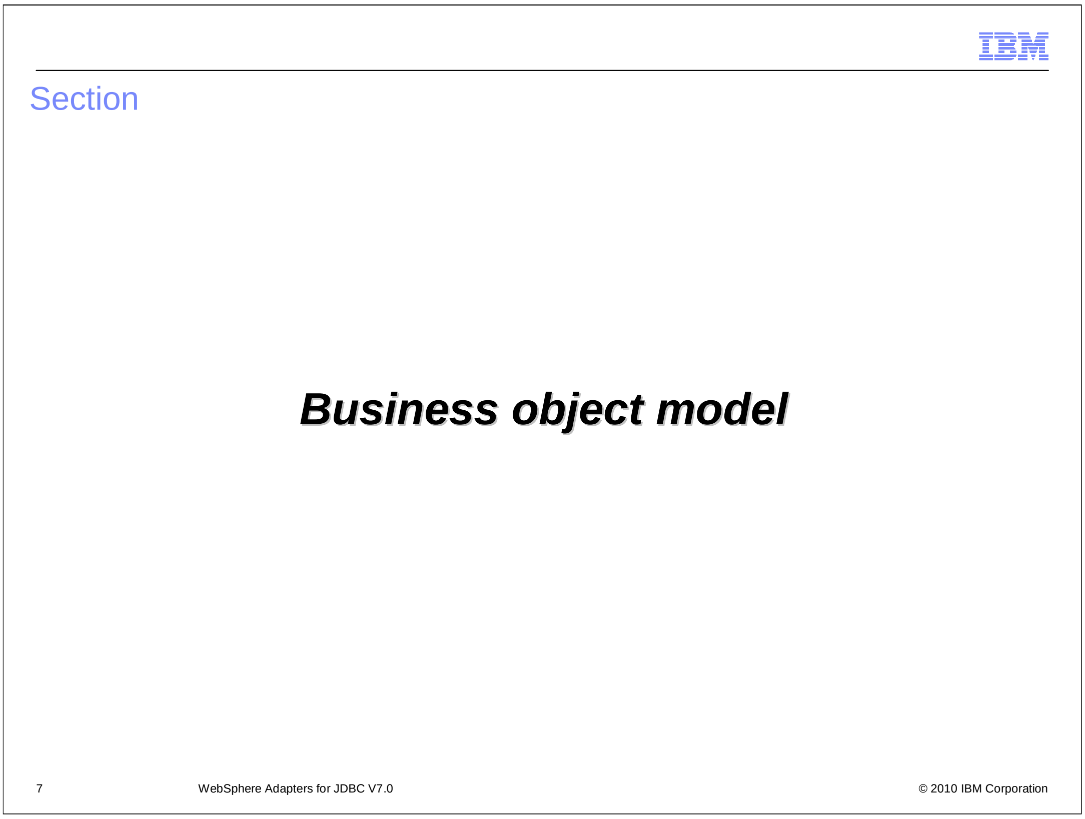


Deployment in WebSphere Process Server

- Use administrative console or wsadmin command line tool
- Create J2C authentication alias specifying user ID and password to database
- Create Data Source JNDI name
- Install new enterprise application (the EAR just exported from WebSphere Integration Developer)
  - Option to deploy RAR separately
  - For the most part, accept defaults
    - Will need to specify authentication alias in the “Map resource references to resources” panel
  - Continue through installation, Finish, Save to Master configuration file

6 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

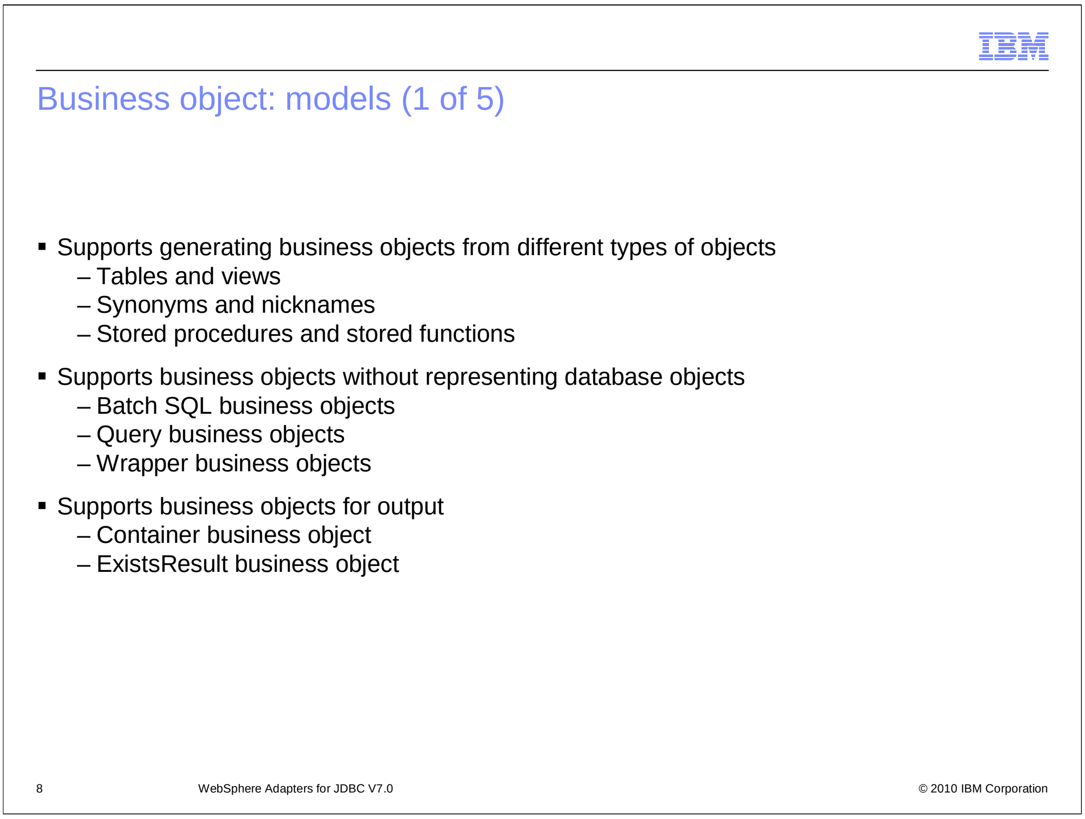
Installation of the application containing the WebSphere Adapter for JDBC to the WebSphere Process Server, is very similar to installing any other enterprise application. However, in the case of outbound communications with the JDBC adapter, you can use the J2C authentication alias which contains the user ID and password that are used to connect to the application database. After creating the Data Source JNDI name, you install the EAR file that you just exported from WebSphere Integration Developer, into the server. You can choose to deploy the RAR file separately, or include it with the application in the EAR file. In general, you can go with the installation defaults, but on the panel that maps resource references to resources, you will have to specify the J2C authentication alias.



Section

Business object model

This section provides an overview of the business object models.



Business object: models (1 of 5)

- Supports generating business objects from different types of objects
  - Tables and views
  - Synonyms and nicknames
  - Stored procedures and stored functions
- Supports business objects without representing database objects
  - Batch SQL business objects
  - Query business objects
  - Wrapper business objects
- Supports business objects for output
  - Container business object
  - ExistsResult business object

8 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

WebSphere® Adapter for JDBC uses business objects to represent tables and views in the database and the results of database queries, stored procedures, and stored functions. Business objects can also create a hierarchy of objects from your database and group unrelated tables. Some business objects do not represent data objects including batch SQL business object which represent a series of user-defined insert, update, and delete statements, or query business objects which represent a user-defined SQL query to run against the database. Last but not least, wrapper business objects which let you group unrelated table and view objects into a single business object, and allow you to group multiple stored procedures into a single business object.

The adapter also use business objects for output. They include container business object which contains the output from a RetrieveAll operation, and ExistsResult business object which contains the output from an Exists operation.

IBM

Business object: models (2 of 5)

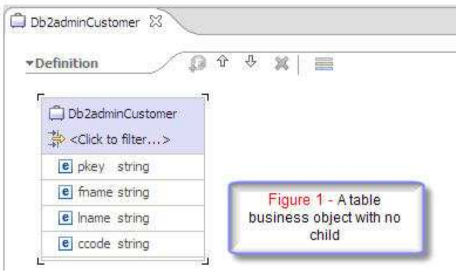


Figure 1 - A table business object with no child

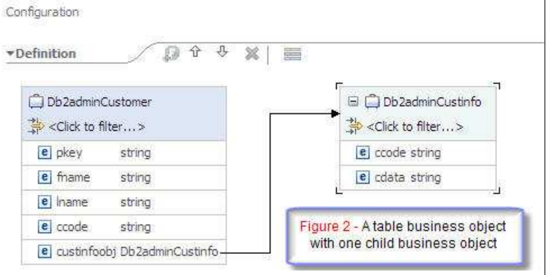


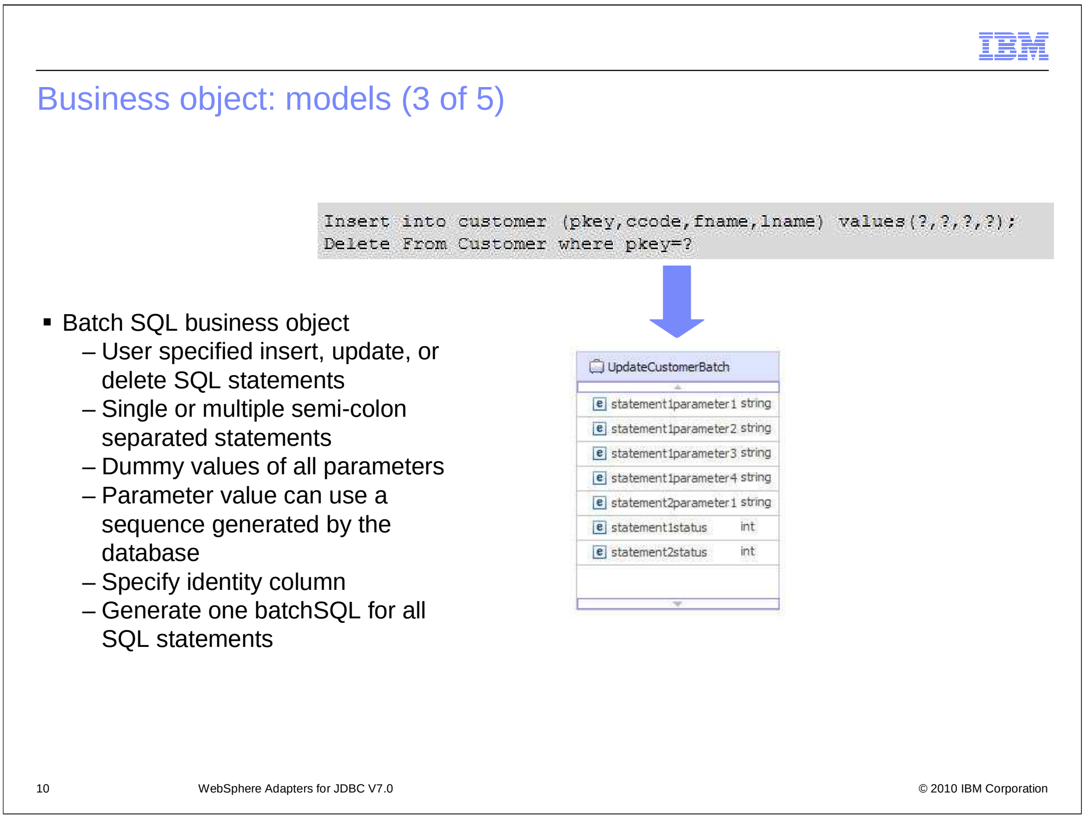
Figure 2 - A table business object with one child business object

- Table or view business object
  - Represents by a simple attribute of the table or view business object
  - Simple attributes within same business objects must be stored in same database tables
  - Must always have a primary key
    - Prompt by external wizard for primary key if the corresponding database table does not have primary key
  - Supports multiple composite, or multiple, primary keys
- Stored procedure and stored function business object
  - Input and output parameters have corresponding attributes in business object
  - Attributes of array or structure have corresponding child business object

Each column in the table or view is represented by a simple attribute of the table or view business object. A simple attribute is an attribute that represents a single value, such as a String, Integer, or Date. Other attributes represent a child business object or an array of child business objects. Simple attributes within the same business object cannot be stored in different database tables; however, some situations are possible. For example, the database table can have more columns than the corresponding business object has simple attributes; that is, some columns in the database are not represented in the business object. Only those columns needed for your application's processing of the business object must be included in your design. Or the business object can have more simple attributes than the corresponding database table has columns; that is, some attributes in the business object are not represented in the database. The attributes that do not have a representation in the database either have no application-specific information, are set with default values, or are parameters for stored procedures or stored functions. Last but not least, the business object can represent a view that spans multiple database tables.

A table business object must always have a primary key, even if the corresponding database table does not have a primary key. In that case, the external service wizard prompts for primary key information when discovering and configuring that business object. The adapter uses that column specified in the primary key attribute when it retrieves table business objects. The adapter supports tables that have composite, or multiple, primary keys. If a database table has one or more primary keys, the wizard sets the primary key property for those columns in the table business object.

In a business object for a stored procedure or stored function, all the input and output parameters of the stored procedure or stored function have corresponding attributes in the business object. If any of the input or output parameters is of a complex type, such as an array or structure, then the corresponding business object attribute is a child business



Business object: models (3 of 5)

```
Insert into customer (pkey,ccode,fname,lname) values(?,?,?,?);
Delete From Customer where pkey=?
```

UpdateCustomerBatch	
statement1parameter1	string
statement1parameter2	string
statement1parameter3	string
statement1parameter4	string
statement2parameter1	string
statement1status	int
statement2status	int

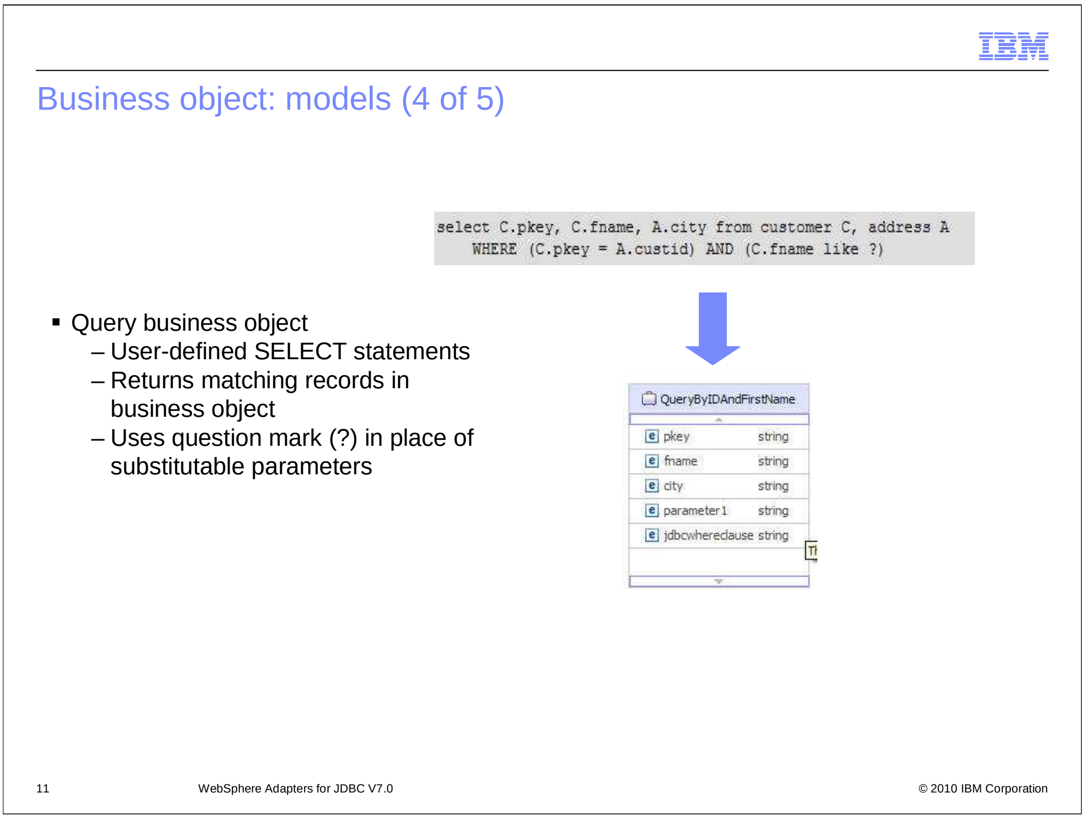
- Batch SQL business object
  - – User specified insert, update, or delete SQL statements
  - – Single or multiple semi-colon separated statements
  - – Dummy values of all parameters
  - – Parameter value can use a sequence generated by the database
  - – Specify identity column
  - – Generate one batchSQL for all SQL statements

10 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

The BatchSQL business object allows you to supply user-specified Insert, Update, and Delete SQL Statements. More advanced users who have a good knowledge of databases and who understand SQL statements will benefit from this new feature. It allows you to directly perform these operations on database entities without having to use table business objects. The external wizard asks to specify single or multiple, semi-colon separated Insert, Update, Delete SQL statements, and the types and dummy values of all parameters are included in the specified SQL statements. You can also specify whether there is an Identity column, which is auto-generated, in the SQL Statement and whether any parameter value should use the sequence generated by the database. The JDBC external service wizard processing will generate a single BatchSQL business object for the all the specified SQL statements, including the parameters that correspond to each SQL statement. At runtime, this BatchSQL business object will process each of the specified Insert, Update and Delete SQL Statements.

If you want to generate a batchSQL business object from a set of specified Select statements, you need to select the "BatchSQL business objects" check box in the Query Filter Properties window. Then you can set the "BatchSQL business object count" to the maximum number of batchSQL business objects that are generated when you click “Execute Query’. When you look in the navigation tree, you find one top level node labeled “BatchSQL Statements”. And by expanding this node, you find child nodes labeled “BatchSQL Statement 1”, “BatchSQL Statement 2” and so on. These statements have a dependency on the value for "BatchSQL business object count". The child nodes are all selectable, and you can select any of them to generate a batchSQL business object.

There are three types of attributes associated with batchSQL business objects. One type



Business object: models (4 of 5)

```
select C.pkey, C.fname, A.city from customer C, address A
   WHERE (C.pkey = A.custid) AND (C.fname like ?)
```

- Query business object
  - User-defined SELECT statements
  - Returns matching records in business object
  - Uses question mark (?) in place of substitutable parameters

QueryByIDAndFirstName

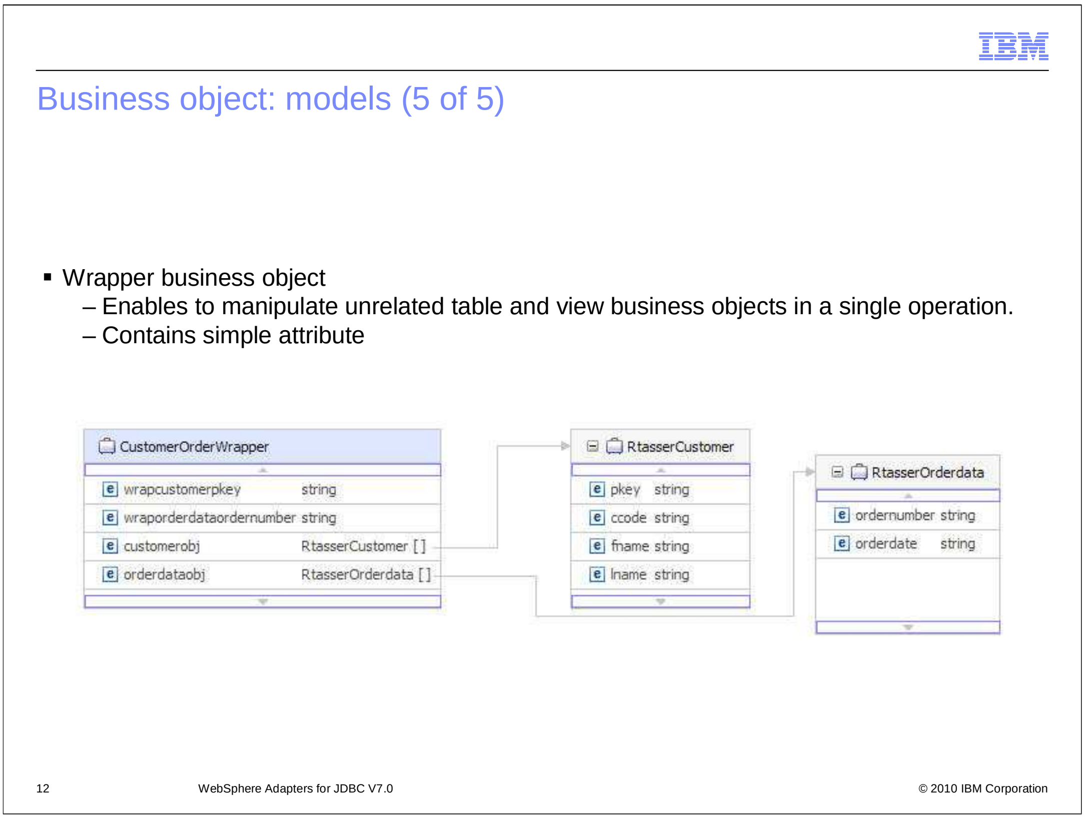
pkey	string
fname	string
city	string
parameter1	string
jdbcwhereclause	string

11 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

Query business objects run a user-defined SELECT statement against the database and return the matching records in business objects.

The external service wizard helps you build query business objects that run user-defined SELECT statements against the database. You specify the SELECT statement, using ? (the question mark) in place of any substitutable parameters in the SELECT statement. The wizard then provides an area where you specify the data type of each parameter and provide a sample value. The sample value must match data in the database because the wizard uses the SELECT statement's results to create the query business object.

Before you save the configuration of the query in the wizard, you validate it. When you validate, the wizard runs the SELECT statement using the sample values. After obtaining the result set, the wizard analyzes the metadata to obtain the column name and column type of all columns. For each column of the returned result set, the wizard generates one corresponding attribute in the query business object. For each parameter in the WHERE clause, the wizard generates one jdbcwhereclause attribute in the query business object and sets this attribute's default value to be the WHERE clause. These attributes are used to generate one dynamic WHERE clause at run time to replace the default WHERE clause.



IBM

Business object: models (5 of 5)

- Wrapper business object
  - Enables to manipulate unrelated table and view business objects in a single operation.
  - Contains simple attribute

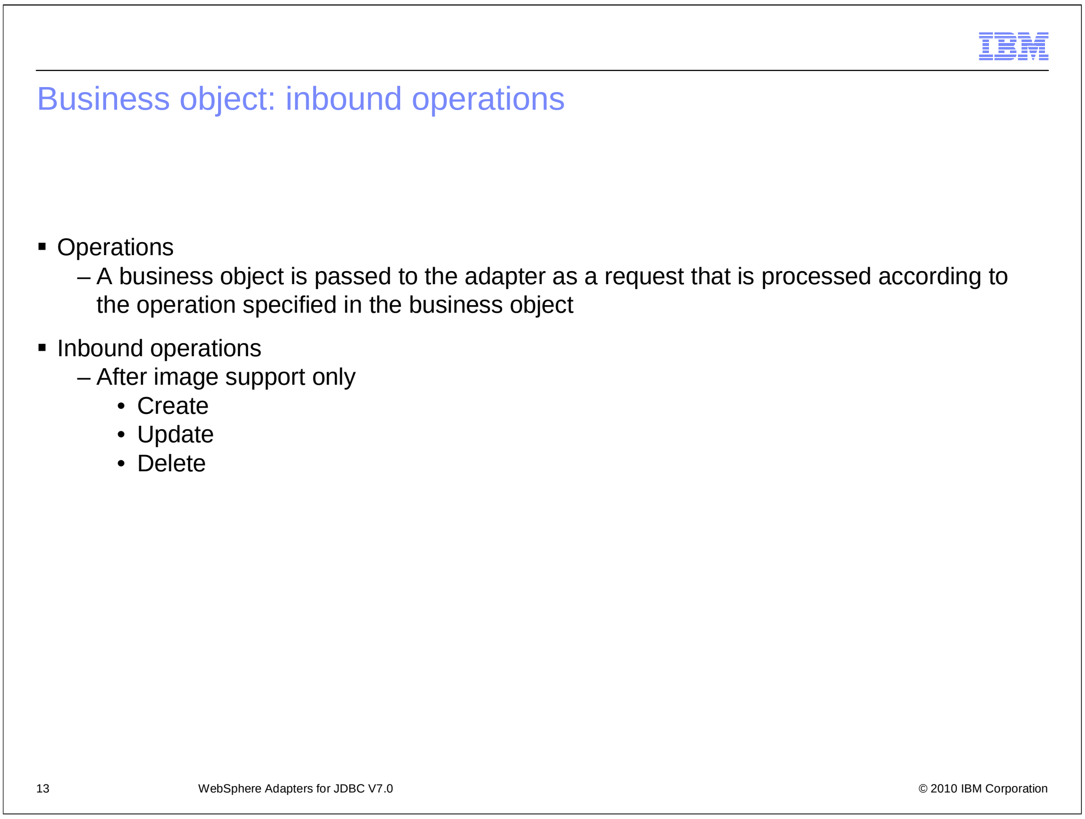
CustomerOrderWrapper	
wrapcustomerpkey	string
wraporderdataordernumber	string
customerobj	RtasserCustomer [ ]
orderdataobj	RtasserOrderdata [ ]

RtasserCustomer	
pkey	string
ccode	string
fname	string
lname	string

RtasserOrderdata	
ordernumber	string
orderdate	string

12 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

A wrapper business object enables you to manipulate unrelated table and view business objects in a single operation. The wrapper business object contains a simple attribute for the primary key of each child business object. The name of the field is the string "wrap", followed by the database table name and the column name of the primary key of the table. The wrapper business object also contains a complex attribute for each table business object. The name of the attribute is the table name with the string "obj" appended. The type of the complex attribute is the name of the corresponding table business object.



Business object: inbound operations

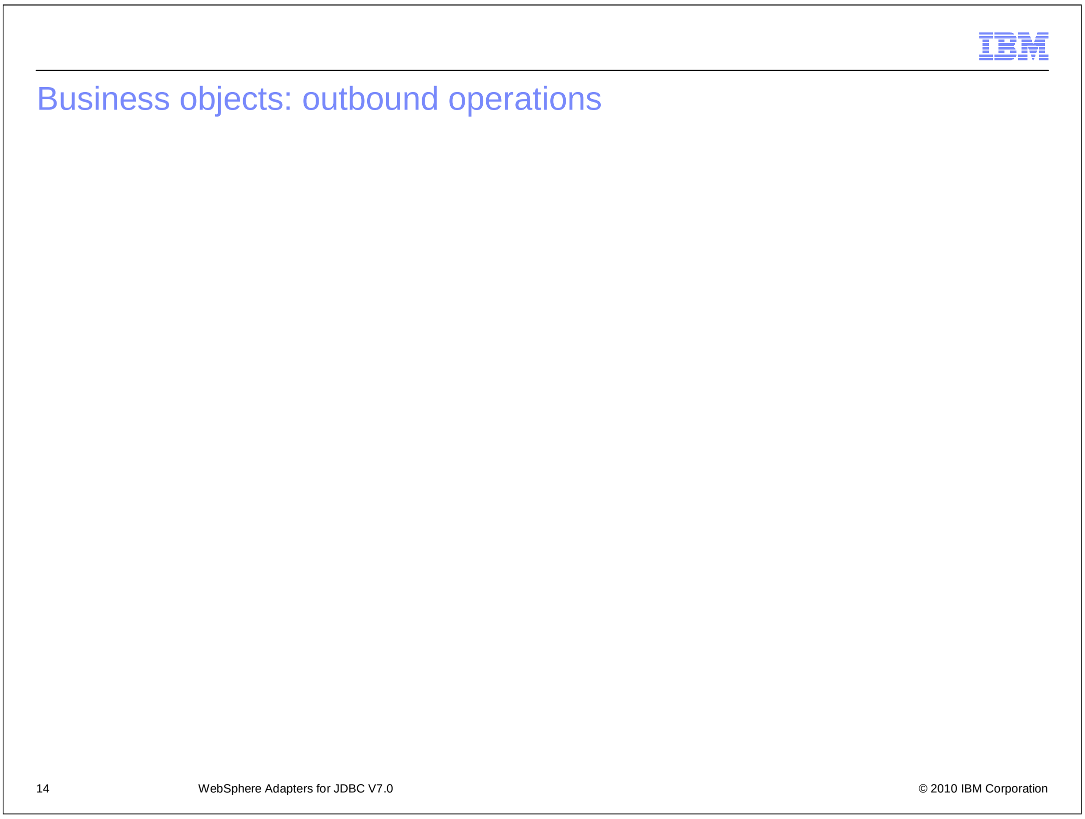
- Operations
  - A business object is passed to the adapter as a request that is processed according to the operation specified in the business object
- Inbound operations
  - After image support only
    - Create
    - Update
    - Delete

13 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

A business object’s content is related to the amount and purpose of the information conveyed by the business object. There are two categories of business object content, “After image” and “Delta”. After image content represents the state of the business object after all changes have been made. This type of content is indicated by the presence of a top-level verb in the business graph of the business object.

Delta content represents the changes that have happened to the business object content. This type of content is indicated by the absence of a top-level verb.

The operations specified in the business object indicate what type of processing is being requested. Business objects are passed to the adapter and processed according to the operations they specify. In the case of inbound operations, only after-image support is available for the operations, which include Create, Update, and Delete.



Business objects: outbound operations

14 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

In the case of outbound operations, after-image support is available for the Create, Update, and Delete operations. Delete operations can include both physical and logical deletes.

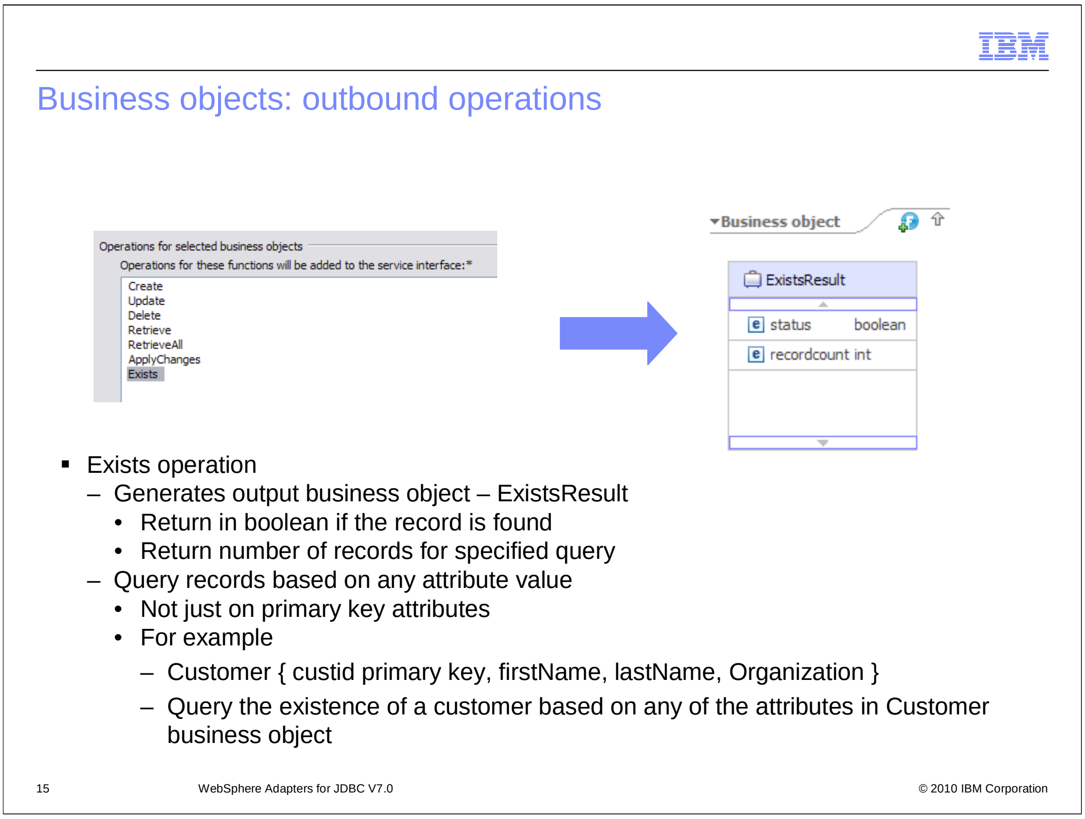
For Delta support, outbound operations also include the ApplyChanges operation.

Outbound supports other operations as well, retrieve, retrieveAll, execute, and exists. The retrieve operation, when passed a hierarchical business object, starts processing by first making a copy of the top level business object without any of its children. It then recursively retrieves the child business objects starting with the parent of the incoming business object.

RetrieveAll is an operation that will direct the JDBC adapter to retrieve an array of business objects from the database. The value of the ResultSetLimit property determines the number of records to return. This value can be set during the external service processing or by specifying the value for MaxRecords on the Configure Objects panel for the outbound ServiceType.

The Execute operation is for both Stored Procedures and for user defined SQL statements. In support of Stored procedures, the Execute operation generates the required stored procedure that corresponds to the stored procedure definition in the database. For user defined SQL statements, this operation generates the required BatchSQL business object based on user defined SQL statement or statements.

The Exists operation determines whether the database table contains records that match



Business objects: outbound operations

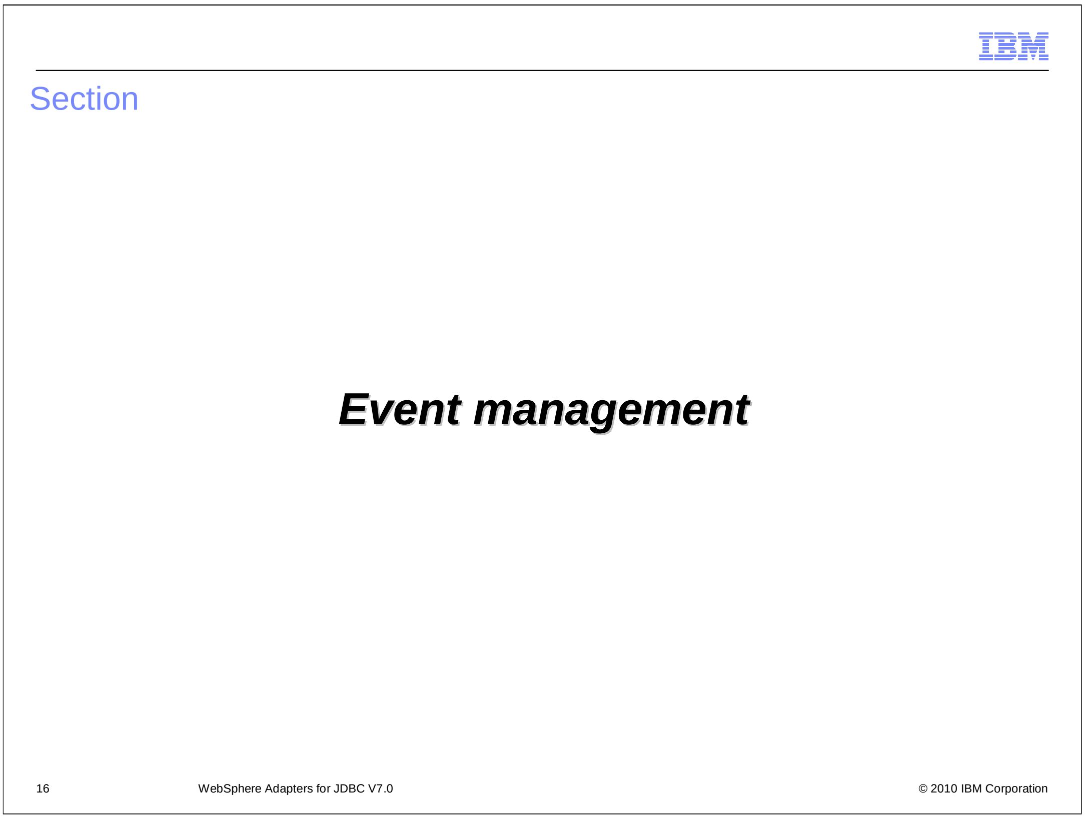
Operations for selected business objects
Operations for these functions will be added to the service interface:*
Create
Update
Delete
Retrieve
RetrieveAll
ApplyChanges
Exists
Business object
ExistsResult
status boolean
recordcount int

- Exists operation
  - Generates output business object – ExistsResult
    - Return in boolean if the record is found
    - Return number of records for specified query
  - Query records based on any attribute value
    - Not just on primary key attributes
    - For example
      - Customer { custid primary key, firstName, lastName, Organization }
      - Query the existence of a customer based on any of the attributes in Customer business object

15 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

Continued from previous slide, the results of the Exist operation is returned in a different business object then the input business object. It actually generates a new business object called ExistsResults, which used the output result based on query criteria set on the attributes of input business object. The business object ExistsResult has two attributes. Status attribute returns with boolean value if the record is found in the database and recordcount attribute returns the number of records found in the databases for the specified query. The default value for record count attribute is 0.   The screen capture illustrates the structure of the ExistsResult business object returned from an Exists operation.

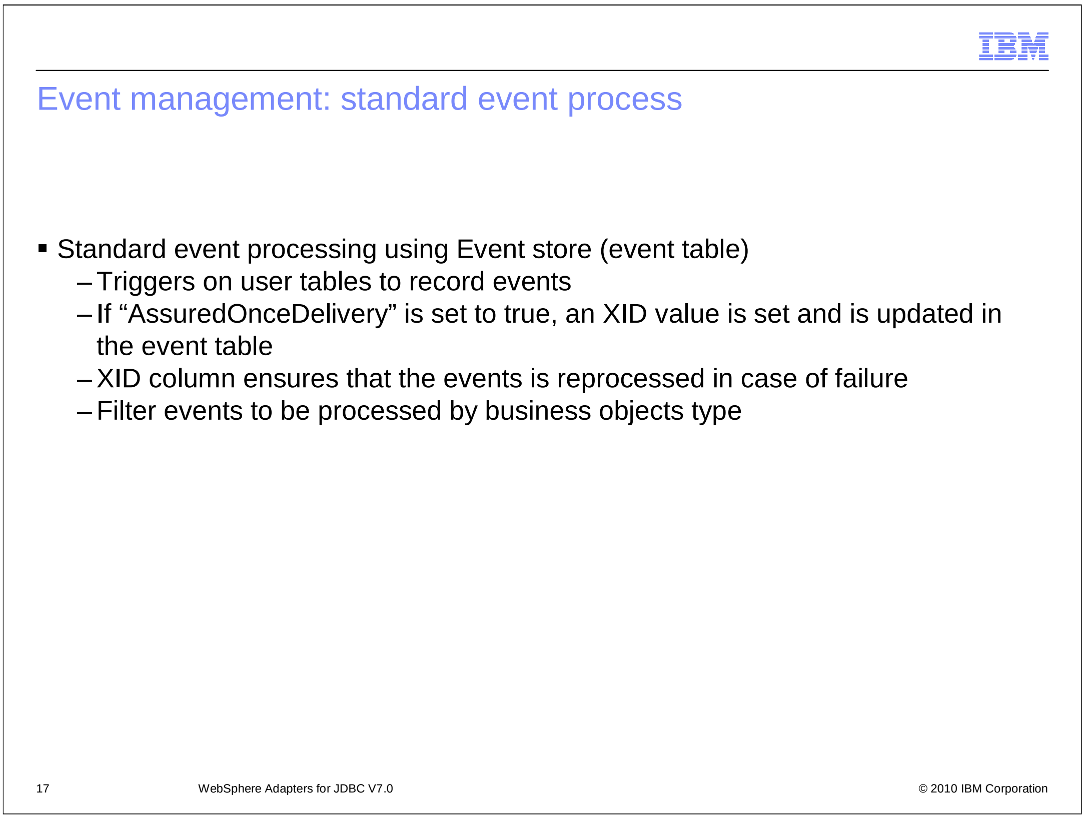
The Exists operation also supports both key and non-key attributes in the selection criteria. It sends that object as an input to query a record/s in the database. For example, if user has generated a business object for a Customer table which has a primary key and non-primary keys attributes, then user can query the existence of a customer based on any of the attributes in Customer business object. And not just query based on primary key attributes only.



Section

Event management

This section provides an overview of the event management of the WebSphere Adapter for JDBC V7.0

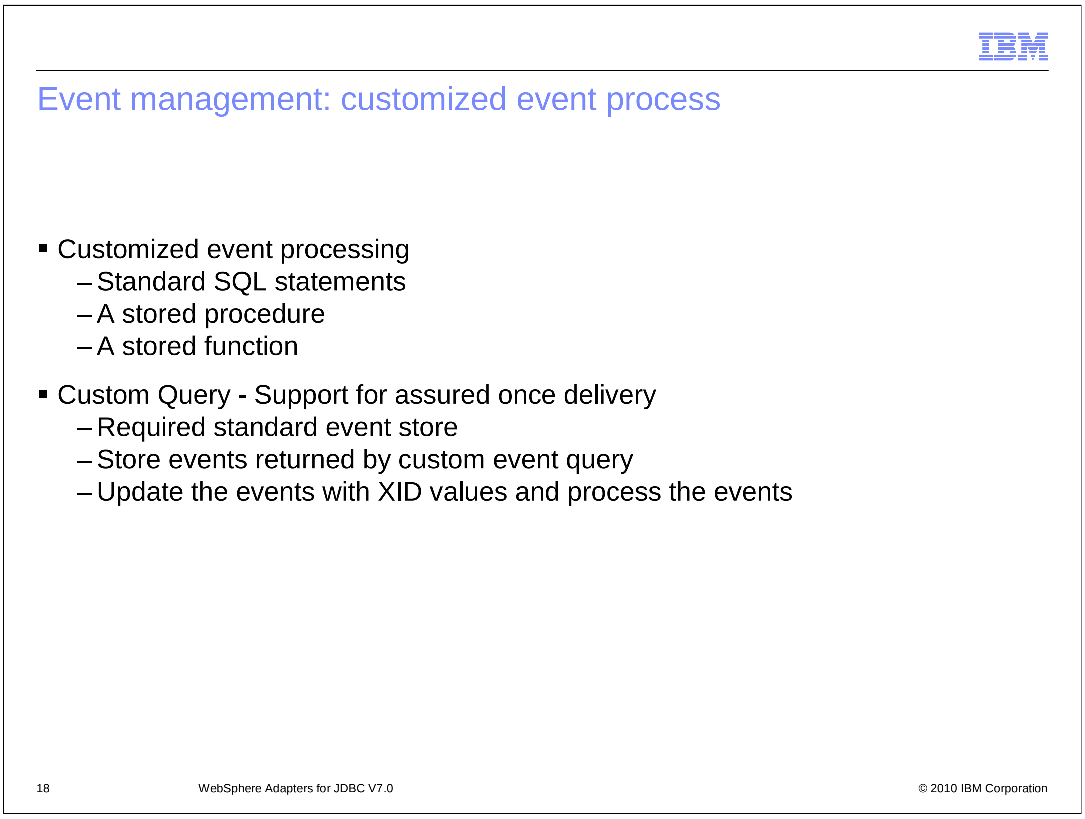


Event management: standard event process

- Standard event processing using Event store (event table)
  - Triggers on user tables to record events
  - If “AssuredOnceDelivery” is set to true, an XID value is set and is updated in the event table
  - XID column ensures that the events is reprocessed in case of failure
  - Filter events to be processed by business objects type

Asynchronous event delivery is supported either by having an event table (event store) or by using custom queries. For standard event processing, the event table in the EIS is populated by the application for any changes that occur in your tables. It can be updated by placing triggers on your tables that record events in the event table, that correspond to updates to your table. If the ‘AssuredOnceDelivery’ property is set to true, an XID value is set for each event in the event store. After each event is picked up for processing, the XID value for that event is updated in the event table. Each event is then delivered to its corresponding endpoint and subsequently deleted from the event table. Before the event is delivered to the endpoint, the database connection might be lost or the application might be stopped, resulting in the event not being processed completely. The XID column ensures that the event is re-processed and sent to the endpoint. Once the database connection is re-established or the adapter starts up again, it first checks for events in the event table that have a value in the XID column. It then processes these events first and then polls the other events during the poll cycles.

The adapter can also filter the events to be processed by business object type, or timestamp. The filter is set in the EventFilterType property specified in the adapter foundation classes. This property has a comma-delimited list of business object types and only the types specified in this property is picked for processing. If no value is specified for the property, no filter is applied and all the events are picked up for processing. If the FilterFutureEvents property in the adapter foundation classes is set to true, the adapter will filter events based on their timestamp. The adapter will retrieve the system-time in each poll cycle and compare it to the timestamp on each event. If the event is set to occur in the future, it is not picked up for processing until that future time.



Event management: customized event process

- Customized event processing
  - Standard SQL statements
  - A stored procedure
  - A stored function
- Custom Query - Support for assured once delivery
  - Required standard event store
  - Store events returned by custom event query
  - Update the events with XID values and process the events

18 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

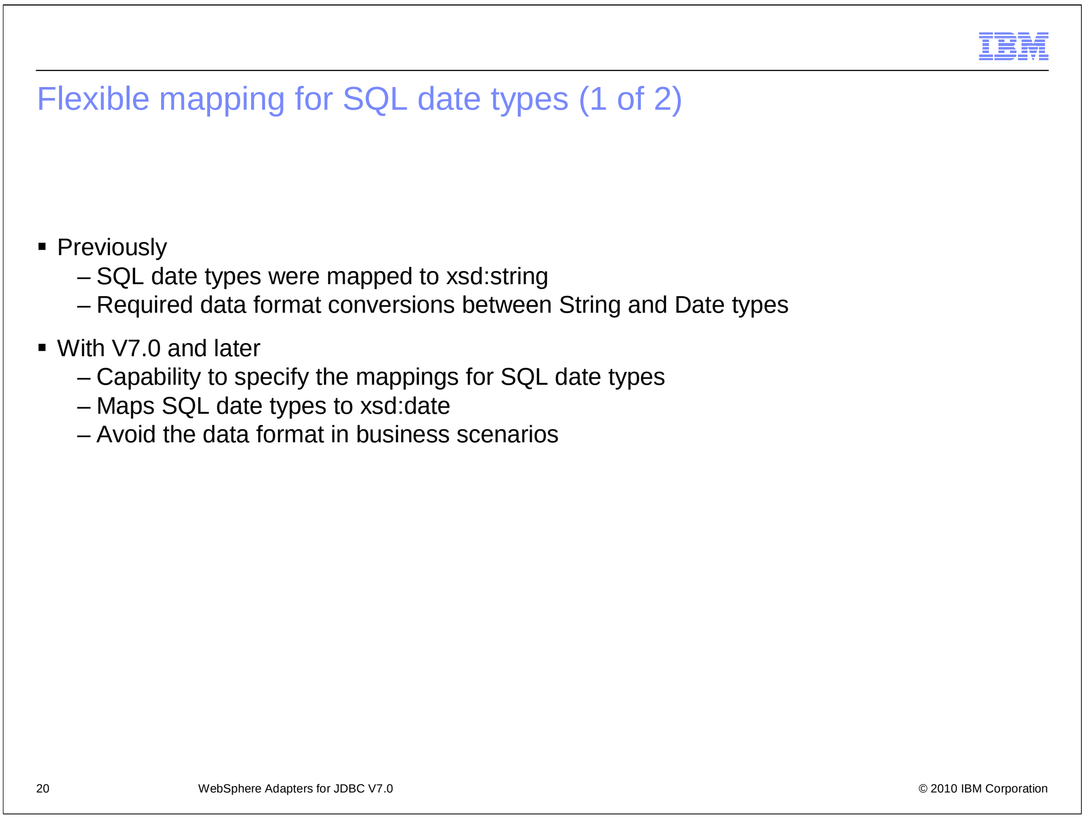
For customized event processing, you can either enter a standard SQL statement, a stored procedure or a stored function. All three cases will return a ResultSet that will have the data for these columns in this order: event_id, object_key, object_name and object_function. The adapter will construct a Business object for each event and then deliver it to the end points that have a subscription for this specific business object. The adapter will also support custom Update and Delete queries for custom event processing. The custom update and delete queries are started after each event is processed, will use the event ID as an input parameter. The update query is used to ensure that the same record does not get picked up for processing during subsequent poll cycles. The delete query is used in case records need to be deleted after each event is delivered.

If you want to have support for assured-once delivery in the case of a custom query, you will need to have the standard event store created. That allows XID to be able to be stored in that event store. The adapter will store the events returned by the custom event query in the event store, update those events with XID values and process the events as it does in the case of standard event processing. In this type of scenario, the event table does not have auto-generated event ID values. Also, when using a custom query, there is no support for event filtering.

Section

What is new in this release

This section includes new features that enhance the business flexibility, user experience, and performance of the WebSphere Adapter for JDBC V7.0

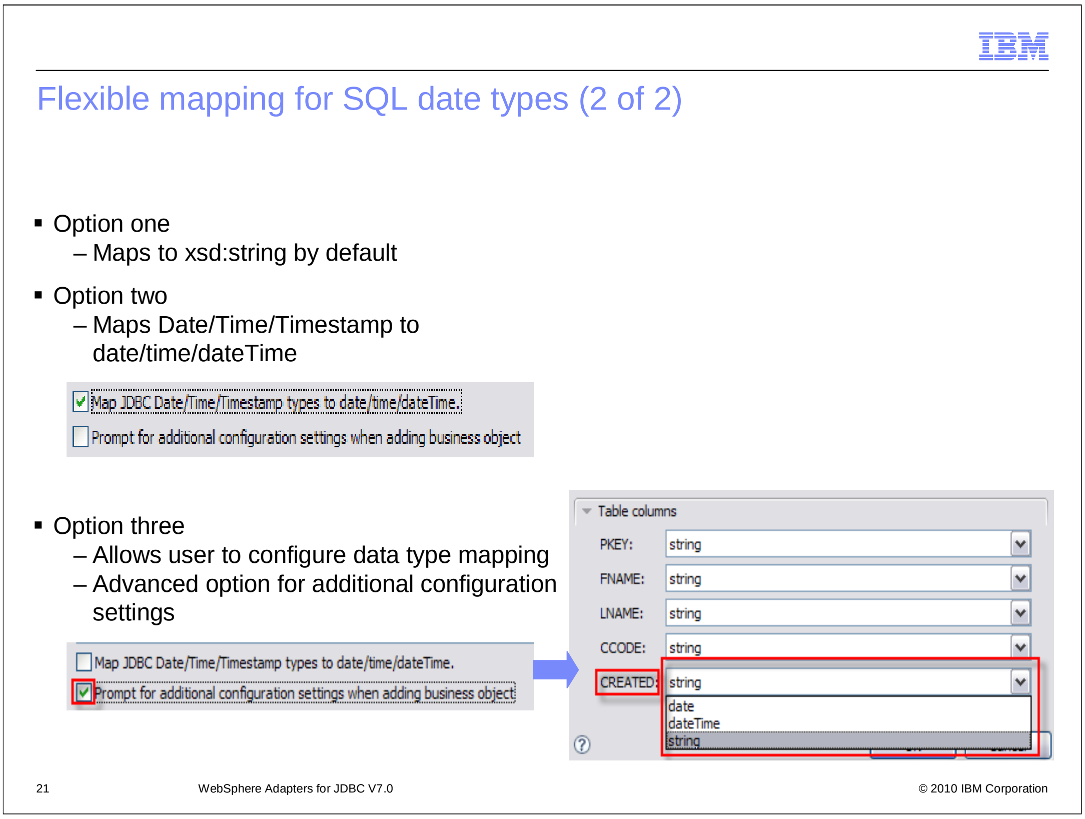


Flexible mapping for SQL date types (1 of 2)

- Previously
  - SQL date types were mapped to xsd:string
  - Required data format conversions between String and Date types
- With V7.0 and later
  - Capability to specify the mappings for SQL date types
  - Maps SQL date types to xsd:date
  - Avoid the data format in business scenarios

20 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

Previously, the SQL date types are mapped to the string data type for business object by default. That means it requires some data format conversions between string and date types. With V7.0 and later, the JDBC adapter is capable to map SQL data types Date, Time, and Timestamp to string or date, time, dateTime XSD types as required in the business objects. Thus, you can avoid additional data format in your business scenarios.



Flexible mapping for SQL date types (2 of 2)

- Option one
  - Maps to xsd:string by default
- Option two
  - Maps Date/Time/Timestamp to date/time/dateTime

☑ Map JDBC Date/Time/Timestamp types to date/time/dateTime.
□ Prompt for additional configuration settings when adding business object

- Option three
  - Allows user to configure data type mapping
  - Advanced option for additional configuration settings

□ Map JDBC Date/Time/Timestamp types to date/time/dateTime.
☑ Prompt for additional configuration settings when adding business object

Table columns

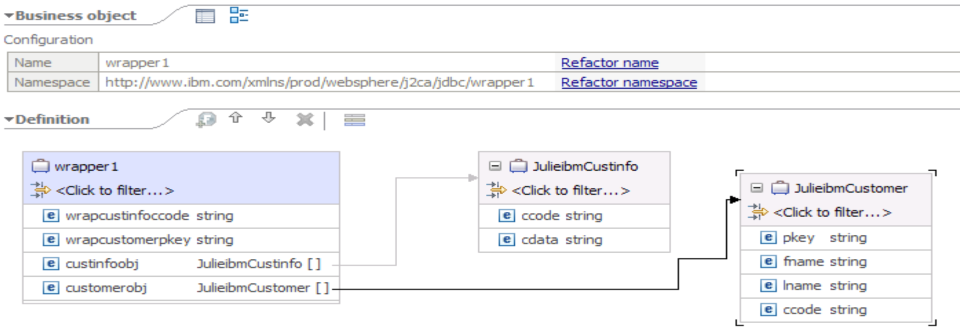
PKEY:	string
FNAME:	string
LNAME:	string
CCODE:	string
CREATED:	string

date
dateTime
string

21 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

Continue from previous slide, the JDBC adapter provide flexible mapping option for SQL date types. The table, stored procedure, and stored function objects with the date and timestamp data types are mapped to the string data type by default. If you want to map these objects to the actual data types that are supported by the JDBC driver such as the date and datetime data types, select the Map JDBC Date/Timestamp types to date/dateTime check box. However, the default data type mapping differs based on the different JDBC driver versions. Thus, you can map the Date to dateTime data type instead of mapping it to Date. In such cases, the appropriate data type must be manually selected in the Specify the Configuration Properties for 'object' window. If you want to customize the format of the Date and Timestamp data types in the Application Info section of the Properties view, clear the Map JDBC Date/Timestamp types to date/dateTime check box. Also, ensure that the data types are mapped to the default string data type in the Specify the Configuration Properties for 'object' window.

Wrapper business object - inbound



Business object

Configuration

Name	wrapper1	Refactor name
Namespace	http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/wrapper1	Refactor namespace

Definition

wrapper1
<Click to filter...>
wrapcustinfoccode string
wrapcustomerpkey string
custinfoobj JulieibmCustinfo [ ]
customerobj JulieibmCustomer [ ]

JulieibmCustinfo
<Click to filter...>
ccode string
cdata string

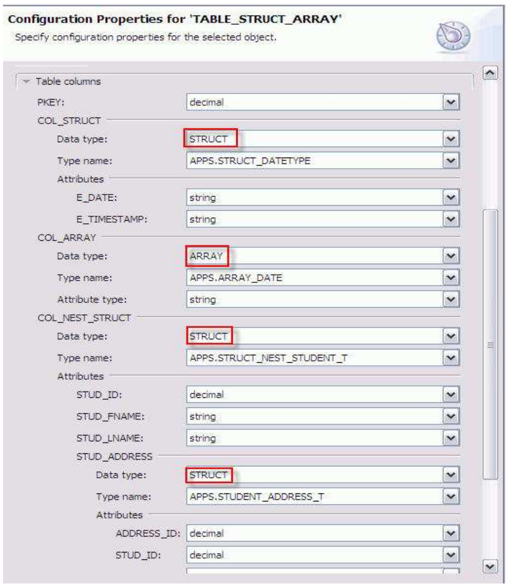
JulieibmCustomer
<Click to filter...>
pkey string
fname string
lname string
ccode string

- Previously
  - – Inbound can only encapsulate one business object for one single event
- With V7.0 and later
  - – Supports wrapper business object
  - – Allows business object which can have multiple child business objects
  - – Obtains multiple business objects for inbound events
    - •Same or different source tables

The JDBC adapter now supports wrapper business object for inbound. This new feature allows inbound event object to have multiple child business object with only one event entry. It also allows business object to have more than one table as source together. They are used to build relationship between two tables in database with no parent-child relationship and use the wrapper as the inbound event object.

User-defined types in table business object - inbound

- Previously
  - Only support simple data type
- In V7.0 and later
  - Support complex types (STRUCTS/ARRAY) in table business object
  - Apply to Oracle database only
  - Require no extra stored procedure



Configuration Properties for 'TABLE_STRUCT_ARRAY'

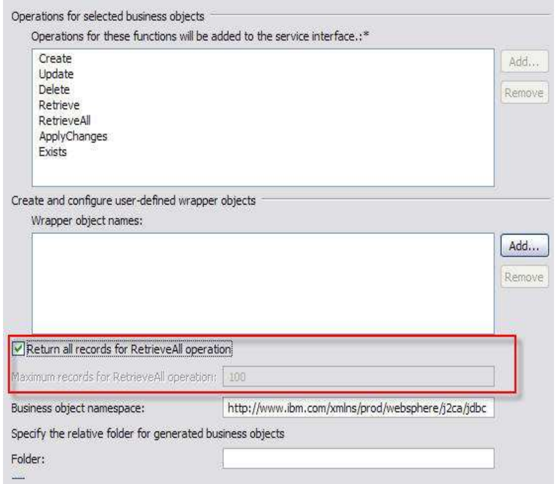
Specify configuration properties for the selected object.

```
Table columns
  PKEY:                 decimal
  COL_STRUCT
    Data type:          STRUCT
    Type name:          APPS.STRUCT_DATETYPE
    Attributes
      E_DATE:           string
      E_TIMESTAMP:      string
  COL_ARRAY
    Data type:          ARRAY
    Type name:          APPS.ARRAY_DATE
    Attribute type:     string
  COL_NEST_STRUCT
    Data type:          STRUCT
    Type name:          APPS.STRUCT_NEST_STUDENT_T
    Attributes
      STUD_ID:          decimal
      STUD_FNAME:       string
      STUD_LNAME:       string
      STUD_ADDRESS
        Data type:      STRUCT
        Type name:      APPS.STUDENT_ADDRESS_T
        Attributes
          ADDRESS_ID:   decimal
          STUD_ID:      decimal
```

In previous version, the JDBC adapter only supports simple data type. With V7.0 and later, the adapter supports complex data types such as array, structure, or nested structured in the query result of the business object. However, the adapter does not support these complex types as parameters in batch and query business object. This new feature applies to Oracle database only. The type name and sub attribute details are also automatically discovered and displayed. The adapter processes these data types as child business objects of the table business object.

Return records for RetrieveAll operation - outbound

- Previously
  - Configure positive integer for RetrieveAll operation
  - Throw MatchesExceedLimitFault
    - Returned recorded exceeds the number records configured by user
- With V7.0 and later
  - Checkbox to allow all records for RetrieveAll operation



Operations for selected business objects

Operations for these functions will be added to the service interface.:*

Create
Update
Delete
Retrieve
RetrieveAll
ApplyChanges
Exists

Create and configure user-defined wrapper objects

Wrapper object names:

☑ Return all records for RetrieveAll operation

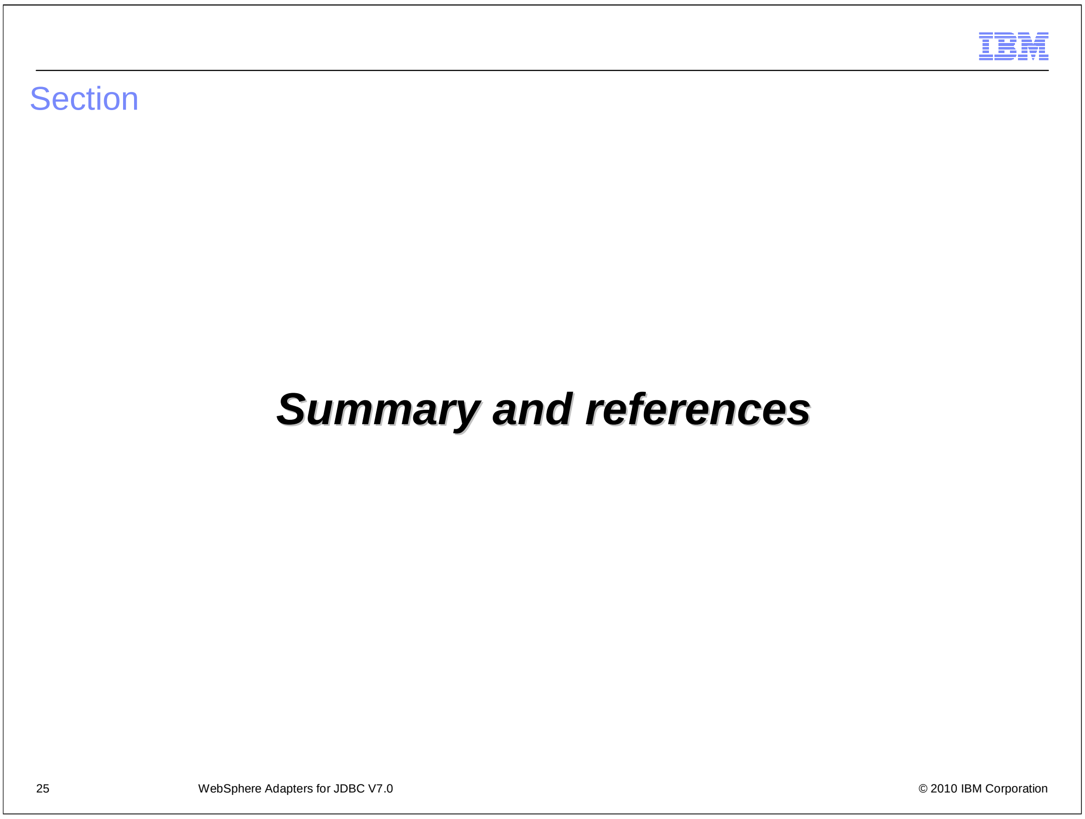
Maximum records for RetrieveAll operation: 100

Business object namespace: http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc

Specify the relative folder for generated business objects

Folder:

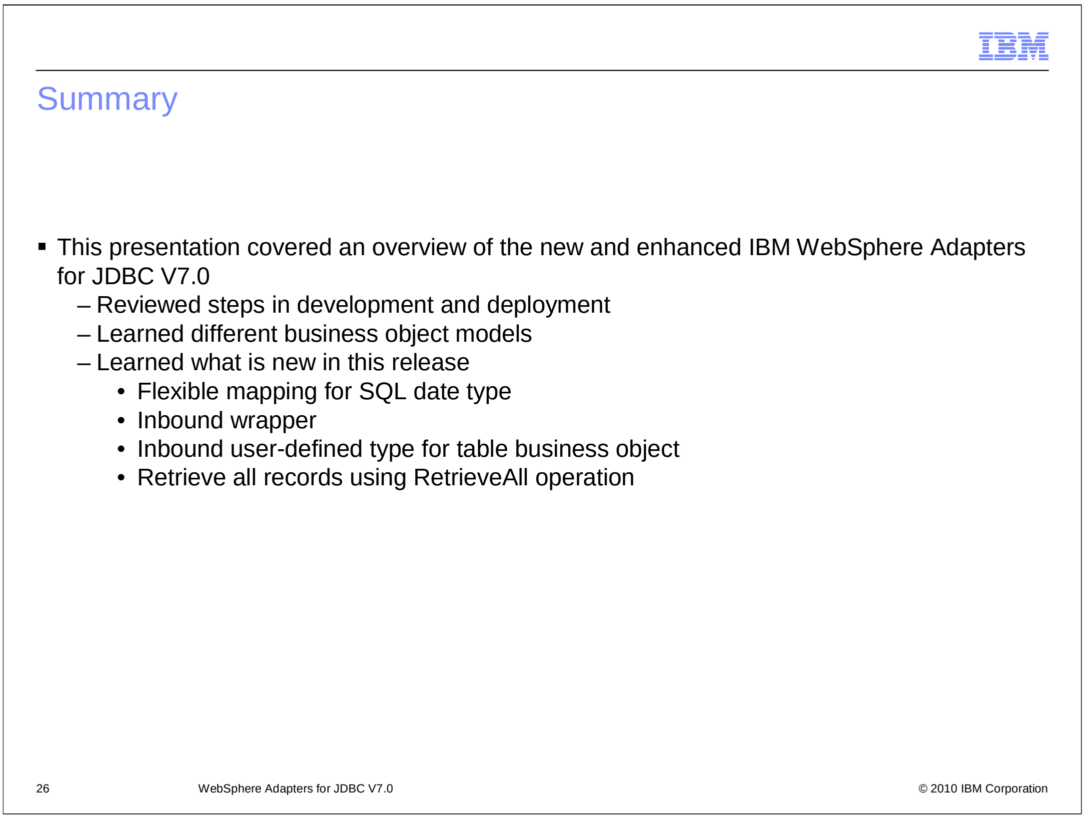
With previous version, user must configure a positive integer to retrieve all records using RetrieveAll operation. Otherwise, if the returned records exceed the number of records configured by the user, the business fault MatchesExceedLimitFault is thrown. In this latest version, the RetrieveAll operation allows to return all records by selecting the check box during configuring business object in external service wizard. This operation also supports empty result set as a non-exception condition.



Section

Summary and references

This section provides a summary of the WebSphere Adapter for JDBC V7.0



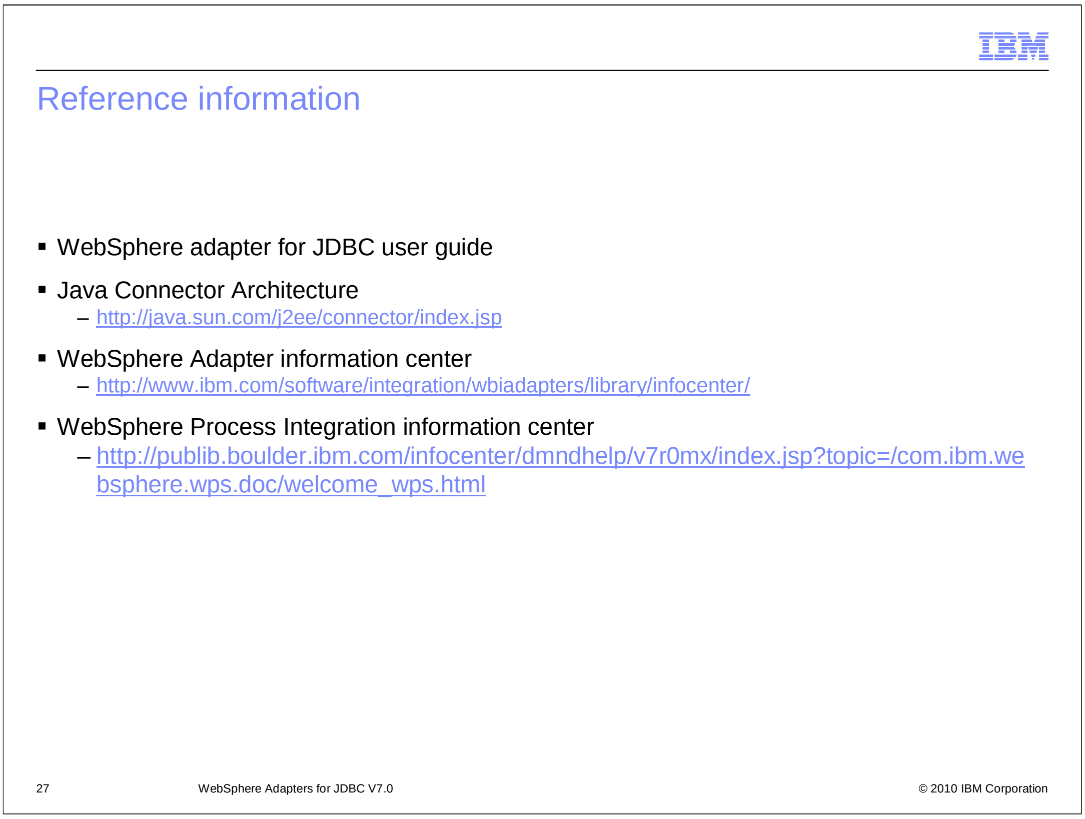
Summary

- This presentation covered an overview of the new and enhanced IBM WebSphere Adapters for JDBC V7.0
  - Reviewed steps in development and deployment
  - Learned different business object models
  - Learned what is new in this release
    - Flexible mapping for SQL date type
    - Inbound wrapper
    - Inbound user-defined type for table business object
    - Retrieve all records using RetrieveAll operation

26 WebSphere Adapters for JDBC V7.0 © 2010 IBM Corporation

This presentation covered an overview of the new and enhanced IBM WebSphere Adapter for JDBC V7.0 release. You have reviewed how to develop and deploy JDBC adapter with different business object models that JDBC adapter offers. You have also learned about flexible mapping for SQL date type, new inbound wrapper business object to allow multiple records in one event entry, and user-defined type for table business object. Finally, there is an option of retrieving all records for RetrieveAll operation without specifying the positive integers for returned records.

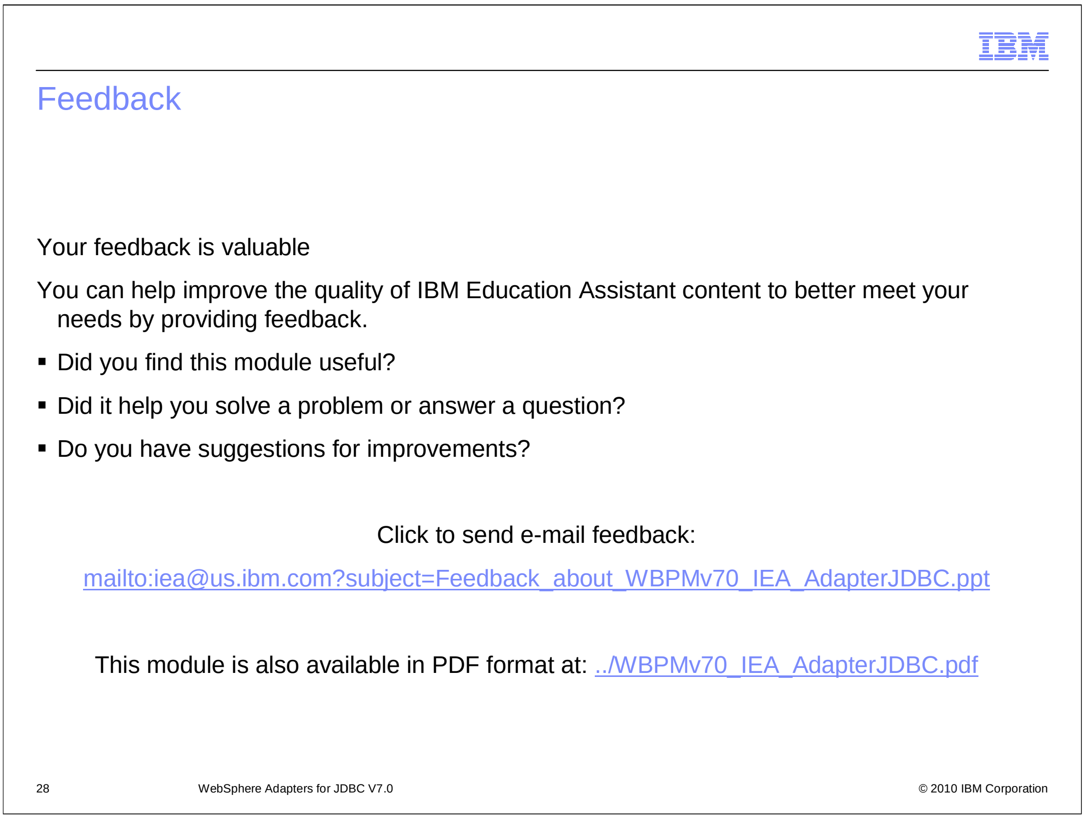
A separate demonstration is available that illustrates the features and functions of the newly enhanced external service wizard.



Reference information

- WebSphere adapter for JDBC user guide
- Java Connector Architecture
  - http://java.sun.com/j2ee/connector/index.jsp
- WebSphere Adapter information center
  - http://www.ibm.com/software/integration/wbiadapters/library/infocenter/
- WebSphere Process Integration information center
  - http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r0mx/index.jsp?topic=/com.ibm.websphere.wps.doc/welcome_wps.html

Additional reference information can be found at these URLs.



Feedback

Your feedback is valuable

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

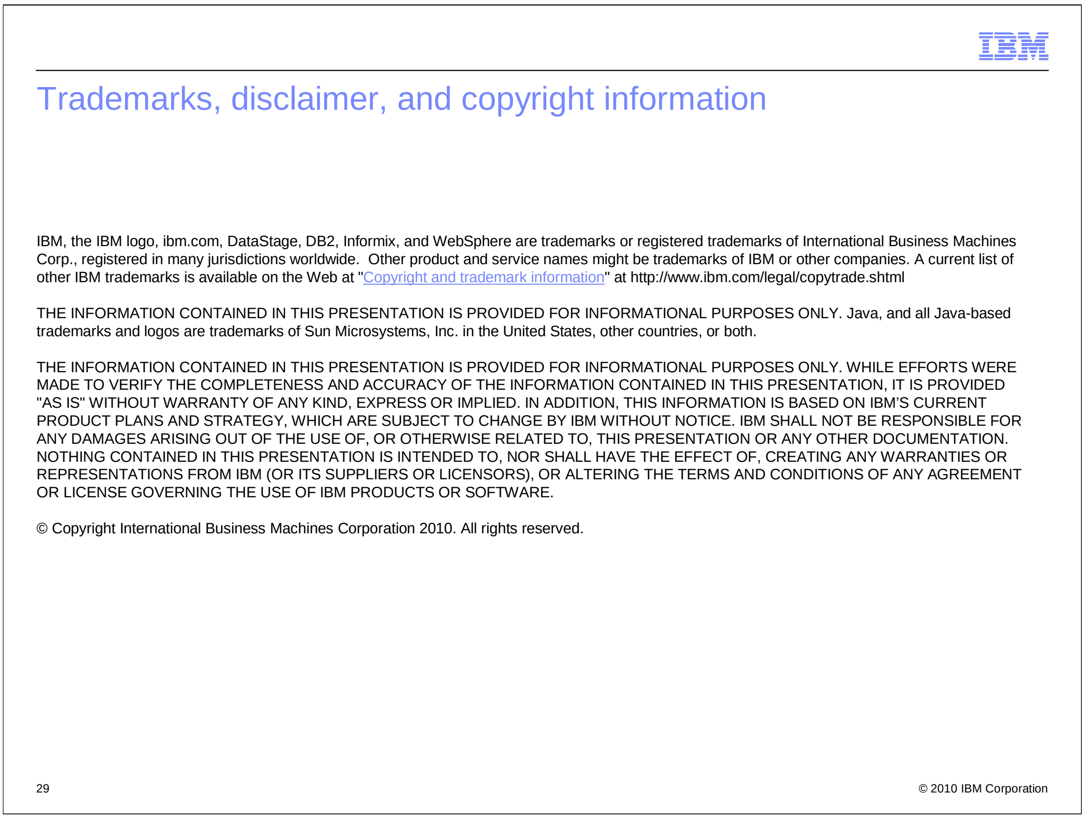
- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WBPMv70_IEA_AdapterJDBC.ppt

This module is also available in PDF format at: ../WBPMv70_IEA_AdapterJDBC.pdf

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



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, DataStage, DB2, Informix, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. Java, and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.