# Configuring Project Components for Oracle® Java CAPS Database Adapters

ORACLE®

# Contents

# Configuring Java CAPS Project Components for Database Adapters

The following topics provide instructions on how to configure Java CAPS project components for database adapters:

**What You Need to Do**

**What You Need to Know**

## Configuring Java CAPS Adapter Connectivity Map Properties

This topic describes how to set the connectivity map properties for Java CAPS Adapters.

All adapters contain a unique set of default configuration parameters. After the adapters are established and an External System is created in the Project's Environment, the adapter parameters are modified for your specific system. Adapter configuration parameters are modified from the following locations:

- **Connectivity Map**: These parameters most commonly apply to a specific component adapter, and may vary from other adapters (of the same type) in the Project.

- **CAPS Environment**: These parameters are commonly global, applying to all adapters (of the same type) in the Project. The saved properties are shared by all adapters in the External System window.

- **Collaboration or Business Process**: Adapter properties may also be set from your Collaboration or Business Process, in which case the settings will override the corresponding properties in the adapter's Connectivity Map configuration. Any properties that are not overridden retain their configured default settings.

Modifications to Connectivity Map configuration properties are made from the Adapter Properties Editor. A description of each property is displayed in the Description pane when it is selected. This provides a brief explanation of the required settings or options. The Comments pane provides an area to record notes and information regarding the currently selected property. These comments are saved when you close the editor.

## ▼ To Add an Adapter to the Connectivity Map

**1** **Open the Connectivity Map opened in the Connectivity Map Editor.**

**2** **In the Connectivity Map Editor toolbar, click the External Applications icon.**

A list of all available External Applications appears.

**3** **Click the External Application you want to add.**

An icon for the application appears on the toolbar to the right of the External Applications icon.

**4** **Select the new icon, and drag and drop it to the Connectivity Map editor canvas.**

The External Application appears on the canvas and in the project tree.

**5** **Drag a connection between the External Application and a Service, such as a Java Collaboration Definition (JCD) or Business Process.**

When you connect an External Application to a Service, the Connectivity Map Editor automatically assigns the appropriate adapter to the link (see the figure below). Each adapter is supplied with a template containing its default configuration properties, which are accessible from the Connectivity Map.

## ▼ To Configure Adapter Connectivity Map Properties

**1** **From the Connectivity Map, double click the adapter icon located in the link between the associated External Application and the Service.**

**2    If a dialog box appears requesting that you select the type of connection, make your selection and then click OK.**

This dialog box appears when there is more than one type of Adapter for the selected External Application. For example, some Adapters have a transactional mode and non-transactional mode.

The Properties Editor appears.

**3    From the Properties Editor, click on any folder to display the default configuration properties for that section.**

**4    Click on any property field to make it editable.**

**5    If an ellipsis appears next to a field, you can click the ellipsis button to open an editor for the field.**

This is useful for long field values.

**6    Make any necessary changes to the property values.**

**7    Click OK to close the Properties Editor and save the settings.**

**Note** – The available properties for the Applications Adapters are described in the following sections.

# DB2 Connect Adapter Inbound Connectivity Map Properties

The Inbound configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

## DB2 Connect Type 4 Adapter - Inbound Parameter Settings

The Parameter Settings section of the DB2 Connect Type 4 Adapter Inbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 1** DB2 Connect Adapter Inbound Type 4

| Name | Description | Required Value |
|------|-------------|----------------|
| Pollmilliseconds | The polling interval in milliseconds. | A valid numeric value. The default is 5000. |
| PreparedStatement | The Prepared Statement used for polling against the database. | The Prepared Statement must be the same Statement you created using the Database OTD Wizard. Only a SELECT Statement is allowed. Additionally, no place holders should be used. This is a SQL statement that cannot contain any input data (i.e. you cannot use "?" in the Prepared Query). |

## DB2 Connect Adapter - Inbound Parameter Settings

The Parameter Settings section of the DB2 Connect Adapter Inbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 2** DB2 Connect Adapter Inbound Type 4

| Name | Description | Required Value |
|------|-------------|----------------|
| Pollmilliseconds | The polling interval in milliseconds. | A valid numeric value. The default is 5000. |
| PreparedStatement | The Prepared Statement used for polling against the database. | The Prepared Statement must be the same Statement you created using the Database OTD Wizard. Only a SELECT Statement is allowed. Additionally, no place holders should be used. This is a SQL statement that cannot contain any input data (i.e. you cannot use "?" in the Prepared Query). |

# DB2 Connect Adapter Outbound Connectivity Map Properties

The Outbound configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

- "DB2 Connect Type 4 Adapter - Outbound JDBC Connector Settings" on page 9
- "DB2 Connect XA Adapter - Outbound JDBC Connector Settings" on page 9
- "DB2 Connect Adapter - Outbound JDBC Connector Settings" on page 10

# DB2 Connect Type 4 Adapter - Outbound JDBC Connector Settings

The JDBC Connector Settings section in the DB2 Connect Type 4 Adapter Outbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 3** DB2 Connector Type 4 Adapter - Outbound JDBC Connector Settings

| Name | Description | Required Value |
|---|---|---|
| Description | A description for the database. | A valid string. The default is DB2 Connect Type 4 Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface for a Type 4 driver. | A valid class name. The default is com.ibm.db2.jcc.DB2ConnectionPoolDataSource. |

# DB2 Connect XA Adapter - Outbound JDBC Connector Settings

The JDBC Connector Settings section in the DB2 Connect XA Adapter Outbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 4** DB2 Connect XA Adapter - Outbound JDBC Connector Settings

| Name | Description | Required Value |
|---|---|---|
| Description | A description for the database. | A valid string. The default is DB2 Connect XA Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface for a XA driver. | A valid class name. The default is COM.ibm.db2.jdbc.DB2XADataSource. |

# DB2 Connect Adapter - Outbound JDBC Connector Settings

The JDBC Connector Settings section in the DB2 Connect Adapter Outbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 5**    DB2 Connect Adapter - Outbound JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | Enter a description for the database. | A valid string. The default is DB2 Connect Connection Pool Datasource. |
| ClassName | Specify the Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is COM.ibm.db2.jdbc. DB2ConnectionPoolDataSource. |

# DB2 Connect non-Transactional Type 4 Adapter - Outbound JDBC Connector Settings

The JDBC Connector Settings section in the DB2 Connect non-Transactional Type 4 Adapter Outbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 6**    DB2 Connect non-Transactional Type 4 Adapter - Outbound JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | Enter a description for the database. | A valid string. The default is DB2 Connect Type 4 Connection Pool Datasource |
| ClassName | Specify the Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface for a non-transactional Type 4 driver. | A valid class name. The default is com.ibm.db2.jcc. DB2ConnectionPoolDataSource. |

# DB2 Connect Type 4 XA Adapter - Outbound JDBC Connector Settings

The JDBC Connector Settings section in the DB2 Connect Type 4 XA Adapter Outbound Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 7    DB2 Connect Type 4 XA Adapter - Outbound JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is DB2 Connect Type 4 XA Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface for a Type 4 XA driver. | A valid class name.<br><br>The default is `com.ibm.db2.jcc.DB2XADataSource`. |

## DB2 Connect non-Transactional Adapter - Outbound JDBC Connector Settings

The JDBC Connector Settings section in the Outbound DB2 Connect non-Transactional Adapter Outbound Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 8    DB2 Connect non-Transactional Adapter - Outbound JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is DB2 Connect non-Transactional Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface for a non-transactional driver. | A valid class name.<br><br>The default is `COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource`. |

# DB2 Adapter Inbound Connectivity Map Properties

The Inbound configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

## DB2 Adapter Inbound Properties for Windows and UNIX

The Parameter Settings section of the DB2 Adapter Inbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 9**    DB2 Adapter Inbound Connectivity Map Parameter Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Pollmilliseconds | The polling interval in milliseconds. | A valid numeric value. The default is 5000. |
| PreparedStatement | The Prepared Statement used for polling against the database. | The Prepared Statement must be the same Statement you created using the Database OTD Wizard. Only a SELECT Statement is allowed. Additionally, no place holders should be used. This is a SQL statement that cannot contain any input data (i.e. you cannot use "?" in the Prepared Query). |

# DB2 Adapter Inbound Properties for z/OS and AS/400

The Parameter Settings section of the DB2 Adapter Inbound Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 10**    DB2 Adapter Inbound Connectivity Map Parameter Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Pollmilliseconds | The polling interval in milliseconds. | A valid numeric value. The default is 5000. |
| PreparedStatement | The Prepared Statement used for polling against the database. | The Prepared Statement must be the same Statement you created using the Database OTD Wizard. Only a SELECT Statement is allowed. Additionally, no place holders should be used. This is a SQL statement that cannot contain any input data (i.e. you cannot use "?" in the Prepared Query). |

# DB2 Adapter Outbound Connectivity Map Properties

The Outbound configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

- "DB2 Adapter Outbound Properties for Windows and UNIX" on page 13
- "DB2 Adapter Outbound Properties for z/OS and AS/400" on page 13

# DB2 Adapter Outbound Properties for Windows and UNIX

The JDBC Connector Settings section of the DB2 Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 11**    Windows and UNIX Outbound Connectivity Map JDBC Connector Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Description | A description for the database. | A valid string. The default is DB2 Adapter Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `com.Sun.db2.jdbcx.db2.DB2DataSource`.<br><br>**Note** – Do not change this value. |

# DB2 Adapter Outbound Properties for z/OS and AS/400

The JDBC Connector Settings section of the DB2 Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 12**    z/OS and AS/400 Outbound Connectivity Map JDBC Connector Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Description | A description for the database. | A valid string. The default is DB2 Datasource For AS400/zOS. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `com.Sun.db2.jdbcx.db2.DB2DataSource`.<br><br>**Note** – Do not change this value. |

# Informix Adapter Connectivity Map Properties Configuration

When you connect an External Application to a Collaboration, NetBeans IDE automatically assigns the appropriate adapter to the link. Each adapter is supplied with a list of adapter connections (transaction support levels) from which to choose.

The properties for the Informix Adapter are divided into the following categories:

- "Informix Adapter — Outbound Adapter Properties" on page 14
- "Informix Adapter — Outbound XA Adapter Properties" on page 14
- "Outbound non-Transactional Adapter Properties" on page 15

For more information on Transaction Support Levels, see "Transaction Support Levels Between Different Versions" on page 15.

# Informix Adapter — Outbound Adapter Properties

The Outbound Adapter Properties include outbound parameters used by the external database.

TABLE 13    Informix Adapter Outbound Connectivity Map Properties

| Name | Description | Required Value |
| --- | --- | --- |
| Description | A description for the database. | A valid string. The default is INFORMIX Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `com.SeeBeyond.informix.jdbcx.informix.InformixDataSource`. |

# Informix Adapter — Outbound XA Adapter Properties

The Outbound XA Adapter Properties include outbound parameters used by the external database. Informix supports XA, a data source that provides connections that can participate in a distributed transaction. XA is a two-phase commit protocol that forms part of the JDBC 2.0 Standard Extension.

TABLE 14    Informix Adapter Outbound XA Connectivity Map Properties

| Name | Description | Required Value |
| --- | --- | --- |
| Description | Enter a description for the database. | A valid string. The default is INFORMIX XA Datasource. |
| ClassName | Specifies the Java class in the JDBC driver that is used to implement the XADataSource interface. | A valid class name. The default is `com.SeeBeyond.informix.jdbcx.informix.InformixDataSource`. |

# Outbound non-Transactional Adapter Properties

You can create Informix databases with or without logging enabled. If logging is disabled, then Non-Transactional mode must be used. Because data logs are not retained during Non-Transactional execution of SQL calls, data recovery is not possible during accidental or unscheduled shutdown of the database server.

Disabled logging also prevents transactions—enclosed in BEGIN-Tran and END-Tran statements—from occurring. This means that Non-Transactional mode cannot be used in XA (two-phase commit) transactions.

The Outbound non-Transactional Adapter Properties listed in the following table include outbound parameters used by the external database.

**TABLE 15**   Outbound Informix non-Transactional Adapter Connectivity Map Properties

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is INFORMIX non-Transactional Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the non-Transactional ConnectionPoolDataSource interface. | A valid class name. The default is `com.SeeBeyond.informix.jdbcx.informix.InformixDataSource`. |

# Transaction Support Levels Between Different Versions

The types of transaction support levels used in Java CAPS 5.1.X may be different from the support levels used in Java CAPS 6.3. Projects that are imported from a Java CAPS 5.1.X version can potentially display different results, depending on whether the 5.1.X Java Collaboration Definition (JCD) included multiple (insert/update/delete) operations. This only affects non-XA transactions. If you are using an XA transaction, then you can skip this section.

## Example:

In 5.1.X, five new records are to be inserted into a table. If the last record fails to insert (such as when a duplicate key exists), all previous records will have been inserted. This is the behavior of NoTransaction support.

In 6.0, five new records are to be inserted into a table. If one of the records fails to insert (such as when a duplicate key exists), the other four records will not be inserted. This is the behavior of the LocalTransaction.

In order to achieve the same result as in 5.1.X versions, you can choose the method below:

1. In the Connectivity Map, delete the link to the database external application, then reconnect the link and select NoTransaction.

2. Fill in the NoTransaction property for the database external system under the Environment.

3. Rebuild the Project.

---

**Note –** There are no changes when migrating from ICAN version 5.0.5 and Java CAPS 6.0.

---

Under the scenario noted above, if you want 6.0 behavior for a LocalTransaction, then set your adapter connection to be Outbound Informix non-Transactional Adapter (NoTransaction).

# JDBC/ODBC Adapter Connectivity Map Properties Configuration

When you connect an External Application to a Collaboration, NetBeans IDE automatically assigns the appropriate adapter to the link. Each adapter is supplied with a list of adapter connections (transaction support levels) from which to choose.

Transaction support levels provided by the JDBC Adapter include:

- "JDBC/ODBC Adapter — Outbound Adapter Properties" on page 16
- "JDBC/ODBC Adapter — Outbound non-Transactional Adapter Properties" on page 17
- "JDBC/ODBC Adapter — Outbound XA Adapter Properties" on page 17

For more information on Transaction Support Levels, refer to:

- "Transaction Support Levels Between Different Versions" on page 17

## JDBC/ODBC Adapter — Outbound Adapter Properties

The Outbound Adapter Properties include outbound parameters used by the external database.

**TABLE 16**   Outbound Adapter—JDBC Connector Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Description | The description of the database. | A valid string. |

# JDBC/ODBC Adapter — Outbound non-Transactional Adapter Properties

The Outbound non-Transactional Adapter Properties include outbound parameters used by the external database.

TABLE 17    Outbound non-Transactional Adapter—JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | The description of the database. | Any valid string. |

# JDBC/ODBC Adapter — Outbound XA Adapter Properties

The Outbound XA Adapter Properties include outbound parameters used by the external database.

TABLE 18    Outbound XA Adapter—JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | The description of the database. | Any valid string. |

# Transaction Support Levels Between Different Versions

The types of transaction support levels used in Java CAPS 5.1.X may be different from the support levels used in Java CAPS 6.3. Projects that are imported from a Java CAPS 5.1.X version can potentially display different results, depending on whether the 5.1.X Java Collaboration Definition (JCD) included multiple (insert/update/delete) operations. This only affects non-XA transactions. If you are using an XA transaction, then you can skip this section.

### Example:

In 5.1.X, five new records are to be inserted into a table. If the last record fails to insert (such as when a duplicate key exists), all previous records will have been inserted. This is the behavior of NoTransaction support.

In 6.0, five new records are to be inserted into a table. If one of the records fails to insert (such as when a duplicate key exists), the other four records will not be inserted. This is the behavior of the LocalTransaction.

In order to achieve the same result as in 5.1.X versions, you can choose the method below:

1. In the Connectivity Map, delete the link to the database external application, then reconnect the link and select NoTransaction.

2. Fill in the NoTransaction property for the database external system under the Environment.

3. Rebuild the Project.

The following charts identifies what transaction support levels changed between 5.0.5 and 6.0, and 5.1.3 and 6.0, respectively.

**Note –** there are no changes when migrating from ICAN version 5.0.5 and Java CAPS 6.0.

If you want 6.0 behavior for a LocalTransaction, then set your adapter connection to be Outbound JDBC/ODBC non-Transactional Adapter (NoTransaction).

# Oracle Adapter Inbound Connectivity Map Properties

The Parameter Settings section of the Inbound Oracle Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

| Name | Description | Required Value |
|------|-------------|----------------|
| Pollmilliseconds | The polling interval in milliseconds. | A valid numeric value. The default is 5000. |
| PreparedStatement | The Prepared Statement used for polling against the database. | The Prepared Statement must be the same Statement you created using the Database OTD Wizard. Only a SELECT Statement is allowed. Additionally, no place holders should be used. This is a SQL statement that cannot contain any input data (i.e. you cannot use "?" in the Prepared Query). |

# Oracle Adapter Outbound Connectivity Map Properties

The Outbound configuration parameters of the Oracle Adapter, accessed from the Connectivity Map, are organized into the following sections:

- "Oracle Adapter Outbound Properties" on page 19
- "Oracle Adapter Outbound non-Transactional Properties" on page 19

- "Oracle Adapter Outbound with XA Support Properties" on page 19

# Oracle Adapter Outbound Properties

The JDBC Connector Settings section of the Oracle Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is Oracle thin driver Connection Pool Datasource |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `oracle.jdbc.pool.OracleC6onnectionPoolDataSource`<br><br>**Note** – Do not change this value. |

# Oracle Adapter Outbound non-Transactional Properties

The Outbound non-Transactional Adapter Properties include outbound parameters used by the external database.

TABLE 19   Outbound non-Transactional Adapter—JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is Oracle thin driver non-Transactional Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `oracle.jdbc.pool.OracleConnectionPoolDataSource`. |

# Oracle Adapter Outbound with XA Support Properties

The JDBC Connector Settings section of the Oracle Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 20**    Outbound with XA Support JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is Oracle thin driver XA Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `oracle.jdbc.xa.client.OracleXADataSource`. |

# SQL Server Adapter Inbound Connectivity Map Properties

The **Parameter Settings** section of the SQL Server Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 21**    Inbound Adapter Connectivity Map Parameter Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Pollmilliseconds | Polling interval in milliseconds. | A valid numeric value. The default is 5000. |
| PreparedStatement | The Prepared Statement used for polling against the database. | The Prepared Statement must be the same Statement you created using the Database OTD Wizard. Only a SELECT Statement is allowed. Additionally, no place holders should be used. This is a SQL statement that cannot contain any input data (i.e. you cannot use "?" in the Prepared Query). |

# SQL Server Adapter Outbound Connectivity Map Properties

The Outbound configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

- "SQL Server Adapter — Outbound Properties" on page 20
- "SQL Server Adapter — Outbound non-Transactional Properties" on page 21
- "SQL Server Adapter — Outbound with XA Support Properties" on page 21

## SQL Server Adapter — Outbound Properties

The JDBC Connector Settings section of the SQL Server Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 22** Outbound Connectivity Map JDBC Connector Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Description | A description for the database. | A valid string. The default is SQLServer Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `com.Sun.sql server.jdbcx.sqlserver. SQLServerDataSource`. **Note** – Do not change this value. |

# SQL Server Adapter — Outbound non-Transactional Properties

The Outbound non-Transactional Adapter Properties include outbound parameters used by the external database.

**TABLE 23** Outbound non-Transactional Adapter—JDBC Connector Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Description | A description for the database. | A valid string. The default is Sqlserver driver non-Transactional Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is `com.SeeBeyond.sqlserver.jdbcx. sqlserver.SQLServerDataSource`. |

# SQL Server Adapter — Outbound with XA Support Properties

The JDBC Connector Settings section of the SQL Server Adapter Connectivity Map properties contains the top-level parameters displayed in the following table.

**TABLE 24** Outbound with XA Support JDBC Connector Settings

| Name | Description | Required Value |
| --- | --- | --- |
| Description | A description for the database. | A valid string. The default is SQLServer XA Datasource. |

**TABLE 24**  Outbound with XA Support JDBC Connector Settings  *(Continued)*

| Name | Description | Required Value |
|------|-------------|----------------|
| ClassName | The Java class in the JDBC driver that is used to implement the SQLServer XA Datasource interface. | A valid class name. The default is `com.Sun.sql server.jdbcx.sqlserver. SQLServerDataSource`.<br><br>**Note** – Do not change this value. |

# Sybase Adapter Connectivity Map Properties

When you connect an External Application to a Collaboration, the NetBeans IDE automatically assigns the appropriate adapter to the link. Each adapter is supplied with a list of adapter connections (transaction support levels) from which to choose.

Transaction support levels supported by the Sybase Adapter include:

- "Sybase Adapter — Outbound Properties" on page 22
- "Sybase Adapter — Outbound XA Properties" on page 22
- "Sybase Adapter — Outbound non-Transactional Properties" on page 23

For more information on Transaction Support Levels, refer to:

- "Transaction Support Levels Between Different Versions" on page 23

## Sybase Adapter — Outbound Properties

The Outbound Adapter Properties include outbound parameters used by the external database.

**TABLE 25**  Outbound Adapter—Outbound Connection Properties

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | Sybase Connection Pool Datasource. | A valid string. |
| ClassName | Displays the Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name.<br><br>The default is:<br><br>`com.SeeBeyond.jdbcx.sybase. sybaseDataSource`.<br><br>**Note** – Do not change this value. |

## Sybase Adapter — Outbound XA Properties

The Outbound XA Adapter Properties include outbound parameters used by the external database.

**TABLE 26**   Outbound Adapter—Outbound XA Connection Properties

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | The description of the database. | A valid string. |
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is: `com.SeeBeyond.jdbcx.sybase.sybaseDataSource`. **Note** – Do not change this value. |

# Sybase Adapter — Outbound non-Transactional Properties

The Outbound Sybase non-Transactional Adapter Properties include outbound parameters used by the external database.

**TABLE 27**   Outbound Adapter—Outbound non-Transactional Properties

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | The description of the database. | A valid string. |
| ClassName | Displays the Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name. The default is: `com.SeeBeyond.jdbcx.sybase.sybaseDataSource`. **Note** – Do not change this value. |

# Transaction Support Levels Between Different Versions

The types of transaction support levels used in Java CAPS 5.1.X may be different from the support levels used in Java CAPS 6.3. Projects that are imported from a Java CAPS 5.1.X version can potentially display different results, depending on whether the 5.1.X Java Collaboration Definition (JCD) included multiple (insert/update/delete) operations. This only affects non-XA transactions. If you are using an XA transaction, then you can skip this section.

## Example:

In 5.1.X, five new records are to be inserted into a table. If the last record fails to insert (such as when a duplicate key exists), all previous records will have been inserted. This is the behavior of NoTransaction support.

In 6.0, five new records are to be inserted into a table. If one of the records fails to insert (such as when a duplicate key exists), the other four records will not be inserted. This is the behavior of the LocalTransaction.

In order to achieve the same result as in 5.1.X versions, you can choose the method below:

1. In the Connectivity Map, delete the link to the database external application, then reconnect the link and select NoTransaction.

   a. Fill in the NoTransaction property for the database external system under the Environment.

   b. Rebuild the Project.

      **Note that there are no changes when migrating from ICAN version 5.0.5 and Java CAPS 6.0.**

      Under the scenario noted above, if you want 6.0 behavior for a LocalTransaction, then set your adapter connection to be Outbound Sybase non-Transactional Adapter (NoTransaction).

The Configuration properties window opens, displaying the default properties for the adapter.
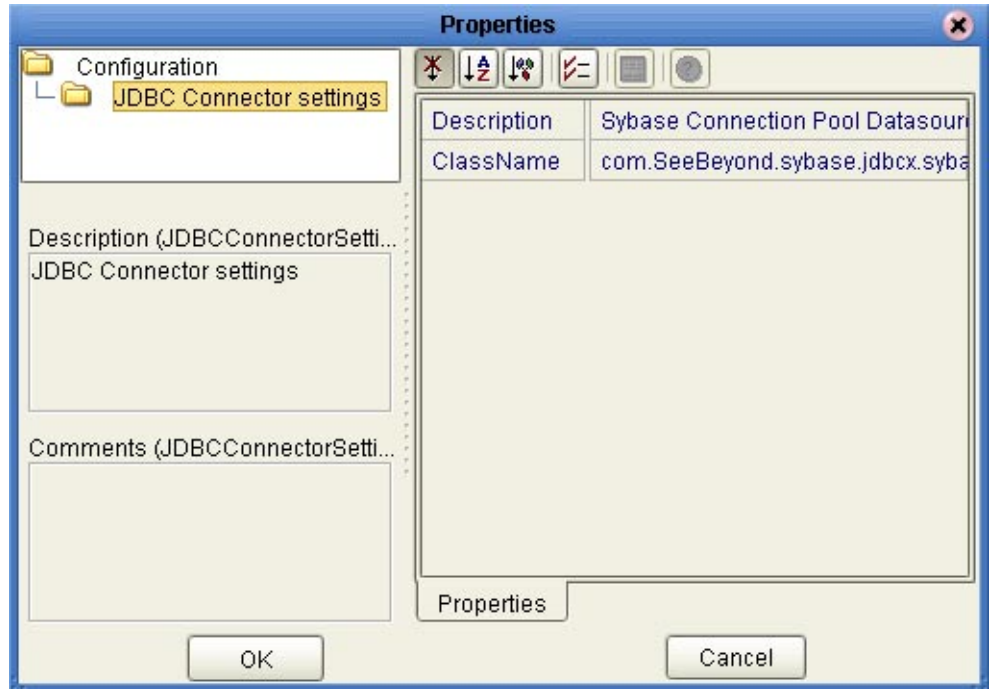
# VSAM Adapter Outbound Connectivity Map Properties

The VSAM configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

- "VSAM Adapter — Outbound Properties" on page 25
- "VSAM Adapter — Outbound non-Transactional Properties" on page 26

## VSAM Adapter — Outbound Properties

The VSAM Outbound Adapter Properties include outbound parameters used by the external database.

**TABLE 28** Outbound Adapter — JDBC Connector Settings

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | VSAM Connection Pool Datasource. | A valid string. |

**TABLE 28**   Outbound Adapter — JDBC Connector Settings     *(Continued)*

| Name | Description | Required Value |
|------|-------------|----------------|
| ClassName | The Java class in the JDBC driver that is used to implement the ConnectionPoolDataSource interface. | A valid class name.<br><br>The default is: com.cac.jdbc.ConnectionPoolDataSource<br><br>**Note –** Do not change this value. |

# VSAM Adapter — Outbound non-Transactional Properties

You can create VSAM databases with or without logging enabled. If logging is disabled, then Non-Transactional mode must be used. Because data logs are not retained during Non-Transactional execution of SQL calls, data recovery is not possible during accidental or unscheduled shutdown of the database server.

Disabled logging also prevents transactions—enclosed in BEGIN-Tran and END-Tran statements—from occurring. This means that Non-Transactional mode cannot be used in XA (two-phase commit) transactions.

The Outbound non-Transactional adapter Properties listed in the following table include parameters used by the external database.

**TABLE 29**   Outbound VSAM non-Transactional adapter Connectivity Map Properties

| Name | Description | Required Value |
|------|-------------|----------------|
| Description | A description for the database. | A valid string. The default is VSAM non-Transactional Connection Pool Datasource. |
| ClassName | The Java class in the JDBC driver that is used to implement the non-Transactional ConnectionPoolDataSource interface. | A valid class name. The default is com.cac.jdbc.ConnectionPoolDataSource. |