



BEA Liquid Data for WebLogic™

Getting Started

Release: 1.0
Document Date: October 2002
Revised:

Copyright

Copyright © 2002 BEA Systems, Inc. All Rights Reserved.

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

BEA, Jolt, Tuxedo, and WebLogic are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Campaign Manager for WebLogic, BEA eLink, BEA Manager, BEA WebLogic Commerce Server, BEA WebLogic Enterprise, BEA WebLogic Enterprise Platform, BEA WebLogic Express, BEA WebLogic Integration, BEA WebLogic Personalization Server, BEA WebLogic Platform, BEA WebLogic Portal, BEA WebLogic Server, BEA WebLogic Workshop, BEA Liquid Data for WebLogic, and How Business Becomes E-Business are trademarks of BEA Systems, Inc.

All other trademarks are the property of their respective companies.

Getting Started

Part Number	Date	Software Version
N/A	October 2002	1.0

Contents

About This Document

What You Need to Know	v
e-docs Web Site	vi
How to Print the Document	vi
Contact Us!	vii
Documentation Conventions	vii

1. Getting Started with Liquid Data: A Basic Example

Step 1. Start the Samples Server.....	1-2
Step 2. Start the WLS Administration Console.....	1-3
Step 3. Configure Order Query Data Sources	1-4
Configure a Relational Database Data Source Description for the Wireless Data	1-5
Create and Deploy the JDBC Connection Pool	1-5
Create and Deploy the JDBC Data Source	1-7
Configure the Relational Database Data Source Description	1-9
Configure an XML File Data Source Description for the BroadBand Data 1-11	
Step 4. Start the Data View Builder and Verify the Data Sources You Configured Show Up in Builder Toolbar	1-13
Step 5. Construct the Query.....	1-14
View a Demo	1-15
Open the Data Source Schemas You Need	1-16
Add a Target Schema from the Server Repository	1-16
Project the Output by Mapping Source to Target Schema.....	1-17
Define the Source Conditions	1-21
Create a Join Between the CUSTOMER_IDs in CUSTOMER and	

CUSTOMER_ORDER within Wireless	1-22
Create a Join between Wireless CUSTOMER_ID and Broadband CUSTOMER_ID.....	1-22
Use a Function to Retrieve a Customer ID Matching a Given Constant . 1-22	
Step 6. View and Test the Query	1-24
Switch to Test View	1-24
Understanding the Generated XQuery	1-25
Run the Query.....	1-26
Step 7. Save the Project	1-29
Next Steps.....	1-29

Index

About This Document

This document provides quick start steps that explain how to start the BEA Liquid Data for WebLogic™ server, bring up the WebLogic Server Administration Console and use it to configure some data sources, use the Data View Builder to map source and target schemas and construct a query, and run the query against the server.

This document covers the following topics:

- [“Step 1. Start the Samples Server” on page 1-2](#)
- [“Step 2. Start the WLS Administration Console” on page 1-3](#)
- [“Step 3. Configure Order Query Data Sources” on page 1-4](#)
- [“Step 4. Start the Data View Builder and Verify the Data Sources You Configured Show Up in Builder Toolbar” on page 1-13](#)
- [“Step 5. Construct the Query” on page 1-14](#)
- [“Step 6. View and Test the Query” on page 1-24](#)

What You Need to Know

This document is intended mainly for business users/developers and XML data designers who will be using Liquid Data to design distributed data solutions and construct queries. System Administrators will also find the initial steps useful as a snapshot example of the configuration process.

e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation or go directly to the “e-docs” Product Documentation page at <http://e-docs.bea.com>.

How to Print the Document

You can print a copy of this document from a Web browser, one file at a time, by using the File—>Print option on your Web browser.

A PDF version of this document is available on the Liquid Data documentation Home page on the e-docs Web site (and also on the documentation CD). You can open the PDF in Adobe Acrobat Reader and print the entire document (or a portion of it) in book format. To access the PDFs, open the Liquid Data documentation Home page, click the PDF files button and select the document you want to print.

If you do not have the Adobe Acrobat Reader, you can get it for free from the Adobe Web site at <http://www.adobe.com/>.

Related Information For more information in general about Java and XQuery, refer to the following sources.

- The Sun Microsystems, Inc. Java site at:
<http://java.sun.com/>
- The World Wide Web Consortium XML Query section at:
<http://www.w3.org/XML/Query>

For more information about BEA products, refer to the BEA documentation site at:
<http://edocs.bea.com/>

Contact Us!

Your feedback on the BEA Liquid Data documentation is important to us. Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the Liquid Data documentation.

In your e-mail message, please indicate that you are using the documentation for the BEA Liquid Data for WebLogic 1.0 release.

If you have any questions about this version of Liquid Data, or if you have problems installing and running Liquid Data, contact BEA Customer Support through BEA WebSupport at www.bea.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Item
boldface text	Indicates terms defined in the glossary.
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.

Convention	Item
<i>italics</i>	Indicates emphasis or book titles.
monospace text	Indicates code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard. <i>Examples:</i> #include <iostream.h> void main () the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float
monospace boldface text	Identifies significant words in code. <i>Example:</i> void commit ()
<i>monospace italic text</i>	Identifies variables in code. <i>Example:</i> String <i>expr</i>
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. <i>Examples:</i> LPT1 SIGNON OR
{ }	Indicates a set of choices in a syntax line. The braces themselves should never be typed.
[]	Indicates optional items in a syntax line. The brackets themselves should never be typed. <i>Example:</i> buildobjclient [-v] [-o name] [-f <i>file-list</i>]... [-l <i>file-list</i>]...

Convention	Item
	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.
...	Indicates one of the following in a command line: <ul style="list-style-type: none"> ■ That an argument can be repeated several times in a command line ■ That the statement omits additional optional arguments ■ That you can enter additional parameters, values, or other information The ellipsis itself should never be typed. <i>Example:</i> <pre>buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...</pre>
.	Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.



1 Getting Started with Liquid Data: A Basic Example

These Getting Started steps walk you through a simple example to illustrate the general process of configuring some data sources and constructing a query. This example shows how to build a query similar to the Order Query used in our BEA Liquid Data for WebLogic™ “Avitek Sample.” Following through this process using our example query is an easy and fun way to get familiar with how to configure data sources, construct data views, and run the generated queries in Liquid Data.

Please note that this topic is meant to serve as an example only to familiarize you with the most common tasks of configuration and query building—it is not comprehensive or required. For more advanced query examples, see the [Query Cookbook](#) in *Building Queries and Data Views* and the [Samples](#) readme files and project files that are installed with Liquid Data. For details on how to set up and manage Liquid Data and configure data sources, see the Liquid Data [Administration Guide](#). For information on building queries and using the [Data View Builder](#), see [Building Queries and Data Views](#).

The following sections are included here:

- [Step 1. Start the Samples Server](#)
- [Step 2. Start the WLS Administration Console](#)
- [Step 3. Configure Order Query Data Sources](#)
- [Step 4. Start the Data View Builder and Verify the Data Sources You Configured Show Up in Builder Toolbar](#)

- [Step 5. Construct the Query \(View a Demo\)](#)
- [Step 6. View and Test the Query](#)
- [Step 7. Save the Project](#)
- [Next Steps](#)

Step 1. Start the Samples Server

If you have not already done so, start the Liquid Data Samples server.

If you have just installed Liquid Data and this is the first time running the Samples server after installation, be sure to first populate the Samples database by choosing the following menu option:

Start—>Programs—>BEA WebLogic Platform 7.0—>Liquid Data for WebLogic 1.0—>Liquid Data Samples—>First-Time Samples Configuration

Wait for the Samples setup to complete. Now you are ready to start the server.

- To start the Samples server on Windows, choose the following menu option:

Start—>Programs—>BEA WebLogic Platform 7.0—>Liquid Data for WebLogic 1.0—>Liquid Data Samples—>Launch Samples Server

- To start the Samples server on UNIX, run the `startWebLogic.sh` command in the following domain:

```
BEA_Home/WL_HOME/liquiddata/samples/config/ld_samples/startWeblogic.sh
```

Note: For more information on setting up and starting the Samples server, see [Post-Installation Tasks](#) in *Installing Liquid Data*. For more information on starting Liquid Data servers in the various preconfigured domains, see [Starting and Stopping the Server](#) in the *Liquid Data Administration Guide*.

Step 2. Start the WLS Administration Console

To start the WLS Administration Console for the Liquid Data Samples server running on your local machine, type the following URL in a Web browser address field:

`http://localhost:7001/console/`

Login to the console by providing the following default username and password for the Samples server.

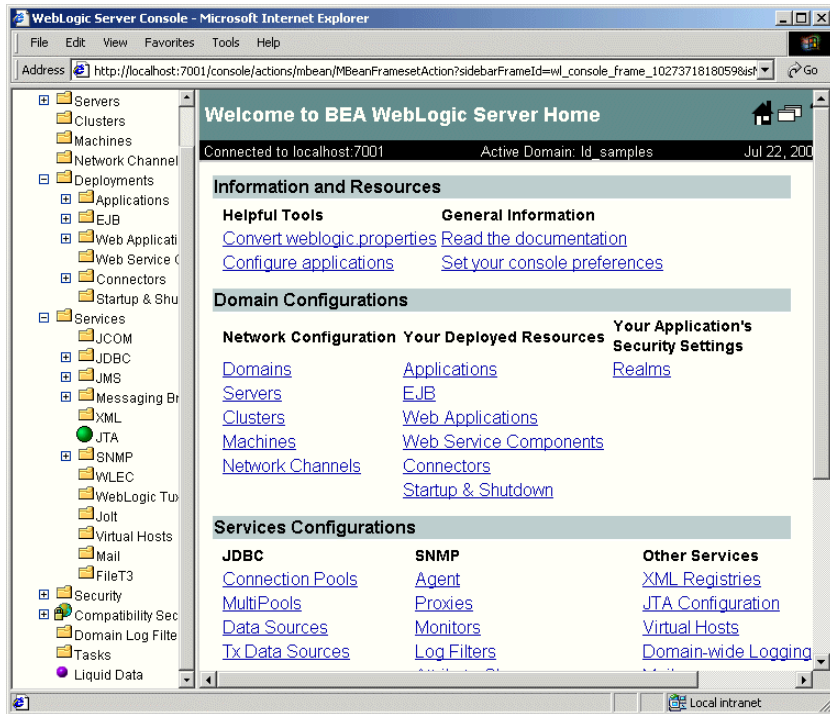
Table 1-1 User Name and Password for Samples WLS Administration Console

Field	Defaults
Username	<code>system</code>
Password	<code>security</code>

The Console Home Page is displayed.

1 Getting Started with Liquid Data: A Basic Example

Figure 1-1 Samples WLS Administration Console



Note: For more information on using the Administration Console, see starting the [Using the WLS Administration Console](#) in the *Liquid Data Administration Guide*.

Step 3. Configure Order Query Data Sources

This section describes how to configure the two data sources used in the Order Query in the Sample:

- An Relational Database Liquid Data data source description for customer and order based on a Wireless customer database

- An XML Liquid Data data source description for customer and order information based on a BroadBand customer database

Configure a Relational Database Data Source Description for the Wireless Data

Configuring a Relational Database Liquid Data source description consists of first creating a JDBC connection pool and JDBC data source, and then configuring the Liquid Data data source using the JDBC configuration you just created. Each of these tasks is described here:

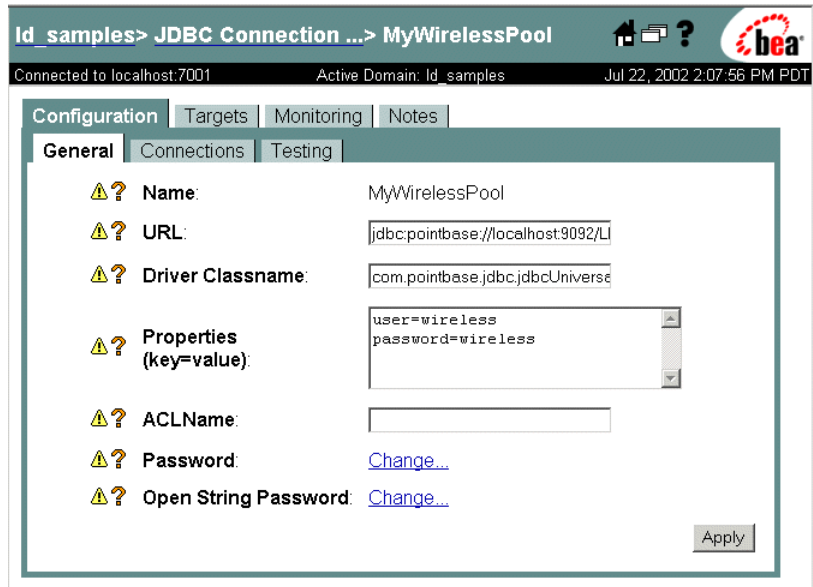
- [Create and Deploy the JDBC Connection Pool](#)
- [Create and Deploy the JDBC Data Source](#)
- [Configure the Relational Database Data Source Description](#)

Note: For detailed information on configuring Relational Database data sources in Liquid Data, see [Configuring Access to Relational Databases](#) in the *Liquid Data Administration Guide*.

Create and Deploy the JDBC Connection Pool

1. In the left pane, expand the Services node.
2. Expand the JDBC node.
3. Click on Connection Pools.
A table of existing connection pools, if any, is shown.
4. Click the “Configure a new JDBC connection pool” text link to display the JDBC pool General configuration tab.

Figure 1-2 JDBC Connection Pool General Configuration Tab



5. On the General tab, provide information about the JDBC connection pool you want to create as described in the following table.

Table 1-2 JDBC Connection Pool Configuration for PB-WL Data Source

Field	Value to Enter
Name	MyWirelessPool This is the name of the JDBC Connection Pool. You can choose any meaningful name.
URL	jdbc:pointbase://localhost:9092/LDDB This is the URL for the database. In this case, we are using the PointBase database installed with the Samples.
Driver Classname	com.pointbase.jdbc.jdbcUniversalDriver This is the driver classname for the database. In this case, we are using the driver for the PointBase database installed with the Samples.

Table 1-2 JDBC Connection Pool Configuration for PB-WL Data Source

Field	Value to Enter
Properties	<pre>user=wireless password=wireless</pre> <p>User name and password based on security set up on the database. Those provided here are the default username and password for the Samples database. You can add these as comma separated <i>attribute=value</i> pairs or one per line as shown above.</p>

6. Click Create.

The new JDBC connection pool you created is shown in the table.

7. In the table of connection pools, click on the name of the new JDBC connection pool you just created.

The Configuration and Monitoring tabs for that pool are displayed.

8. Click on the Configuration tab.

9. Click on the Connections tab and set the Maximum Capacity to 10. (This is an optional step, but we suggest setting the Maximum Capacity on the JDBC connection pool to some number greater than 1.)

10. Click on the Targets tab.

The name of your Liquid Data server should be listed under Available Servers.

11. Select the Liquid Data server in Available and click the right arrow button to move the server into the Chosen list.

12. Click Apply.

Create and Deploy the JDBC Data Source

1. In the left pane, expand the Services node.
2. Expand the JDBC node.
3. Click on Data Sources.

A table of existing data sources, if any, is shown.

1 Getting Started with Liquid Data: A Basic Example

4. Click the “Configure a new JDBC Data Source connection pool” text link to display the JDBC data source General configuration tab.

Figure 1-3 JDBC Data Source General Configuration Tab

The screenshot shows a web application interface for configuring a JDBC data source. The browser address bar displays 'Id_samples> JDBC Data Sources> Create a new JDBCDataSource...'. The page title is 'JDBC Data Sources' and the breadcrumb is 'Id_samples > JDBC Data Sources > Create a new JDBCDataSource...'. The page is connected to localhost:7001, active domain is Id_samples, and the date is Jul 22, 2002 2:10:13 PM PDT. The configuration tab is selected, showing fields for Name (MyWirelessDS), JNDI Name (MyWirelessDS), Pool Name (MyWirelessPool), Row Prefetch Enabled (unchecked), Row Prefetch Size (48), and Stream Chunk Size (256 bytes). A Create button is at the bottom right.

5. On the General tab, provide information about the JDBC data source you want to create as described in the following table.

Table 1-3 JDBC Data Source Configuration for PB-WL Data Source

Field	Value to Enter and Explanation
Name	MyWirelessDS This is a logical name we'll use for the JDBC data source.
JNDI Name	MyWirelessDS In this example we make the JDBC Data Source Name and JDBC JNDI Name the same. The JNDI name will be used in the Liquid Data data source configuration (as described in “Configure the Relational Database Data Source Description”).
Pool Name	MyWirelessPool The Pool Name must match the name of the Connection Pool created in “Create and Deploy the JDBC Connection Pool” on page 1-5.

6. Click Create.

The new JDBC data source you created is shown in the table.

7. In the table of JDBC data sources, click on the name of the new JDBC data source you just created.

The Configuration and Monitoring tabs for that JDBC data source are displayed.

8. Click on the Configuration tab.

9. Click on the Targets tab.

The name of your Liquid Data server should be listed under Available Servers.

10. Select the Liquid Data server in Available and click the right arrow button to move the server into the Chosen list.

11. Click Apply.

Configure the Relational Database Data Source Description

Now you are ready to add a description of your Relational Database data source to the Liquid Data Server registry. To do this:

1. In the left pane, click the Liquid Data node.
2. In the right pane, click the Configuration tab.
3. Click the Data Sources tab.
4. Click the Relational Databases tab.
5. Click the “Configure a new Relational Database source description” text link to display Configure tab.

1 Getting Started with Liquid Data: A Basic Example

Figure 1-4 Liquid Data Relational Databases Configuration Tab on the Liquid Data Node

Id_samples > Liquid Data > Configuration > Data Sources > Relational Databases > Configure a Relational Database Data Source Description

Connected to localhost:7001 Active Domain: Id_samples Mon Jul 22 14:14:03 PDT 2002

Configure

? Name: * MyWireless-LD-DS

? Data Source Name: * MyWirelessDS

? User Name:

? Password:

? Schema: WIRELESS

? Catalog:

? Table Pattern:

? Timeout:

? Shared Connection:

? Isolation Level:

? Maximum Connections:

Create

6. Fill in the fields for your data source as described in the following table. (You need only fill in the fields detailed in the following table—leave other fields blank.)

Table 1-4 Liquid Data Source Description for PB-WL Data Source

Field	Value to Enter
Name	MyWireless-LD-DS This is the logical name of the database—it is a Liquid Data source description used to register the Relational Database data source with the Liquid Data server. You can choose any meaningful name. For this example, use the name we suggest here.

Table 1-4 Liquid Data Source Description for PB-WL Data Source

Field	Value to Enter
Data Source Name	MyWirelessDS This is the database name; it must match the JNDI name for the JDBC data source created in “Create and Deploy the JDBC Data Source” on page 1-7.
Schema	WIRELESS The schema name given here must match the name of a schema stored in your database. In this case, we are using the WIRELESS schema provided in the Samples database. Note: The way in which you define schema varies depending on the type of database you are using. For more detailed information, see Configuring Access to Relational Databases in the <i>Liquid Data Administration Guide</i> .

7. Click Create.

You can click on Relational Databases in the breadcrumbs path at the top of the console to see the relational database you added displayed in the summary table.

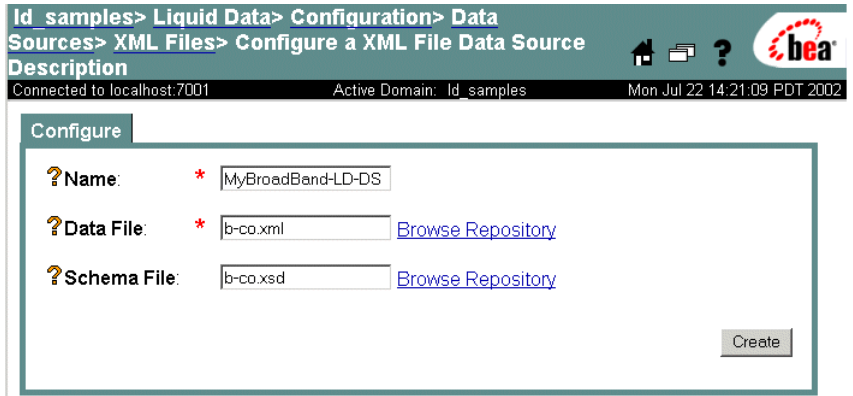
Configure an XML File Data Source Description for the BroadBand Data

Configuring an XML File data source description in Liquid Data simply consists of configuring it as a resource using the Liquid Data node in the WLS Administration Console:

1. In the left pane, click the Liquid Data node.
2. In the right pane, click the Configuration tab.
3. Click the Data Sources tab.
4. Click the XML Files tab.
5. Click the “Configure a new XML data source description” text link to display Configure tab.

1 Getting Started with Liquid Data: A Basic Example

Figure 1-5 XML Configuration Tab on the Liquid Data Node



6. Fill in the fields for your data source as described in the following table.

Table 1-5 Liquid Data Data Source Description for My-PB-WL Data Source

Field	Value to Enter
Name	MyBroadBand-LD-DS This is the name of the Liquid Data data source description used to register the XML data source with the Liquid Data server. You can choose any meaningful name.
Data File	b-co.xml This is the XML data file name. It must correspond to an XML data file stored in the Liquid Data server repository. (The example file name provided above does reside in the Samples server repository.)
Schema	b-co.xsd Schema for the XML file. The schema you specify here must be in the Liquid Data server Repository. The schema used for this example is provided in the Repository along with the samples.

7. Click Create.

You can click on XML Files in the breadcrumbs path at the top of the console to see the file you added displayed in the summary table.

Note: For detailed information on configuring XML data sources in Liquid Data, see [Configuring Access to XML Files](#) in the Liquid Data *Administration Guide*.

Step 4. Start the Data View Builder and Verify the Data Sources You Configured Show Up in Builder Toolbar

To start the Data View Builder and connect to the Samples server, do the following:

1. Start the Data View Builder.

- On a Windows platform, choose the menu item:
Start—>Programs—>BEA WebLogic Platform 7.0—>Liquid Data for WebLogic 1.0—>Launch Data View Builder

You can also start the Data View Builder by double-clicking on the file:
`BEA_Home\WL_HOME\liquiddata\DataViewBuilder\bin\DVBUILDER.cmd`

- On a UNIX platform, open a command window and change directories into the following directory:

`BEA_Home/WL_HOME/liquiddata/DataViewBuilder/bin`

Enter the following command at the prompt to start the Data View Builder:

```
sh DVBUILDER.sh
```

2. Connect to the Samples server.

- a. Enter the following address in the Server URI field on the Login window. (No username or password is required here. Leave those fields blank.)

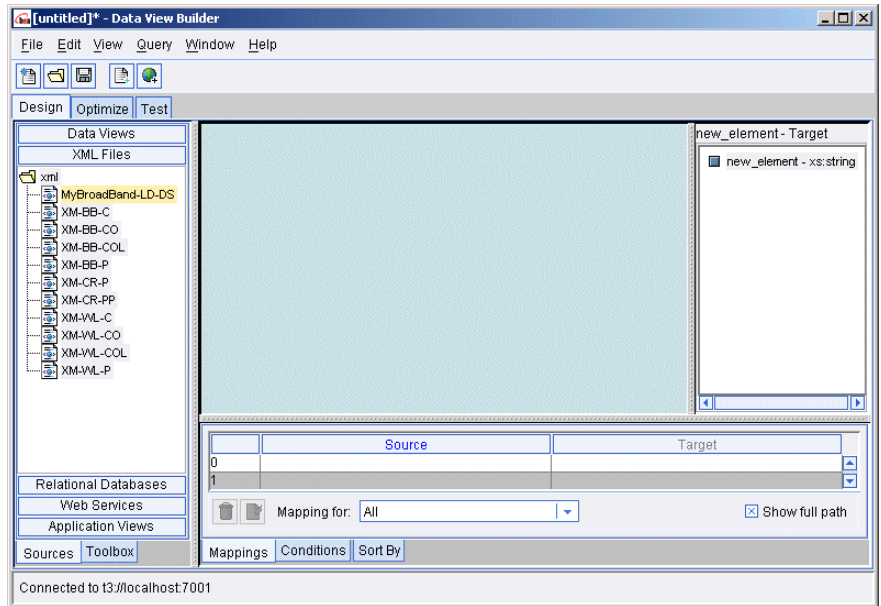
```
t3://localhost:7001
```

- b. Click the Login button.

The Data View Builder workspace and tools are displayed. You can click on the buttons on the left Navigation panel to view the various types of data sources configured on the Liquid Data server to which your Data View Builder is connected. In this case, we are connected to the Samples server.

1 Getting Started with Liquid Data: A Basic Example

Figure 1-6 Starting the Data View Builder



3. The data sources you just configured on the Samples server should be available in the Data View Builder. To find them, click the buttons on the left for the appropriate data source types:
 - Click Relational Databases to view the configured relational databases. This list should include your new Relational Database data source: `MyWireless-LD-DS`
 - Click XML Files to view the configured XML data sources. This list should include your new XML data source: `MyBroadBand-LD-DS`

Step 5. Construct the Query

The Data View Builder allows you to construct queries (or *data views*) by dragging and dropping elements or *nodes* in source and target schemas for the data sources with which you are working.

You can drag and drop elements among source schemas and use other tools (such as constants and functions) to express the query statement. This is also referred to as *defining source conditions*.

You can also drag and drop elements from your source schemas to the target schema to shape and define the structure of the query result. This is also referred to as *mapping*.

As you work in Design mode with the data elements and attributes in source and target schemas using drag-and-drop actions to join or map elements, Data View Builder is generating the XQuery for the query in the background. You can also use the Data View Builder to run the query and view the results in Test Query mode—as a way of testing queries as you build them.

The query you are about to build here is similar to the Order Query featured in the Liquid Data Avitek Sample. Given a customer with orders stored in two data sources (“Wireless” orders in a relational database and “Broadband” orders in an XML file), this query will return the order information for that customer.

To construct our sample Order Query, you need to do the following:

- [Open the Data Source Schemas You Need](#)
- [Add a Target Schema from the Server Repository](#)
- [Project the Output by Mapping Source to Target Schema](#)
- [Define the Source Conditions](#)

All of the tasks described in this section are accomplished using the Design tab in the Data View Builder. For more information on using the tools and features on the Design tab, see the Design tab in *Building Queries and Data Views*.

View a Demo

Constructing the “Order Query” Demo... If you are looking at this documentation online, you can click the “Demo” button to see a viewlet demo showing how to build the conditions and create the mappings described in this example. This demo previews the steps described in detail in the following sections.

Open the Data Source Schemas You Need

To open the data sources you need to construct the data view, simply navigate to them on the Builder Toolbar—>Sources tab and double-click on them:

1. Click the Design tab.
2. On the Builder Toolbar, click the Sources tab (on the bottom of the left vertical panel).
3. Open the data source schemas from the Builder toolbar as follows:
 - Click Relational Databases and double-click on `MyWireless-LD-DS` data source to open the associated XML schema.
 - Click XML Files and double-click on `MyBroadBand-LD-DS` data source to open the associated XML schema.

The XML schemas for the each of the data sources are displayed.

Note: You can open multiple data source schemas and move them around on the workspace as needed.

Add a Target Schema from the Server Repository

The target schema we will use for the Order Query is in a file called `customerOrderReport.xsd` which is available in the Samples server repository.

To add the target schema:

1. Choose the menu command File—>Set Target Schema.

This brings up a file browser. Use the file browser to navigate into the server Repository, which includes several schemas available for use.

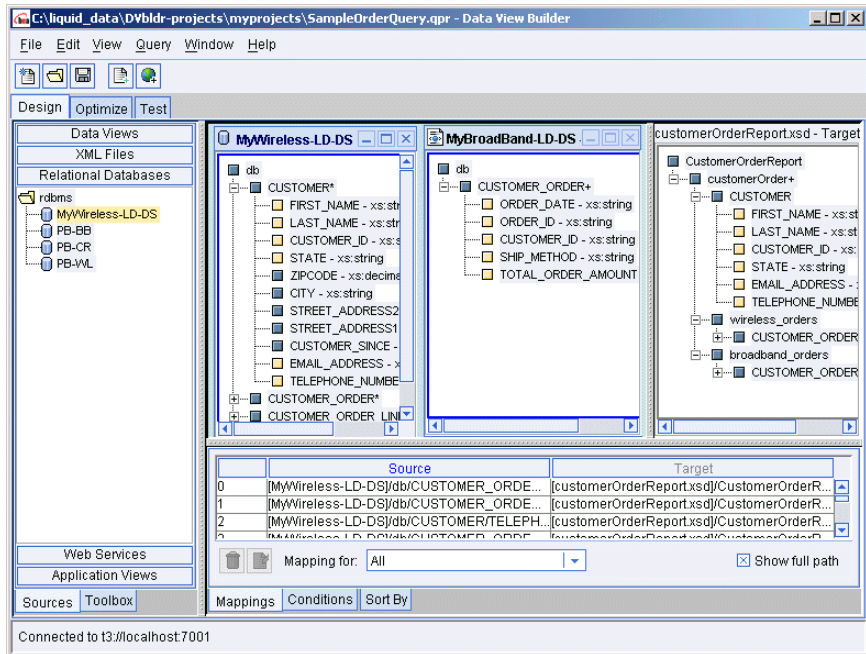
2. Choose `customerOrderReport.xsd` and click Open.

`customerOrderReport.xsd` is displayed as the target schema.

Note: You can open only one target schema at a time and it is always docked to the right side of the workspace.

At this point, your Data View Builder workspace contains two data source schemas and a target schemas you plan to work with as shown in the following figure.

Figure 1-7 Data View Builder with Source and Target Schemas

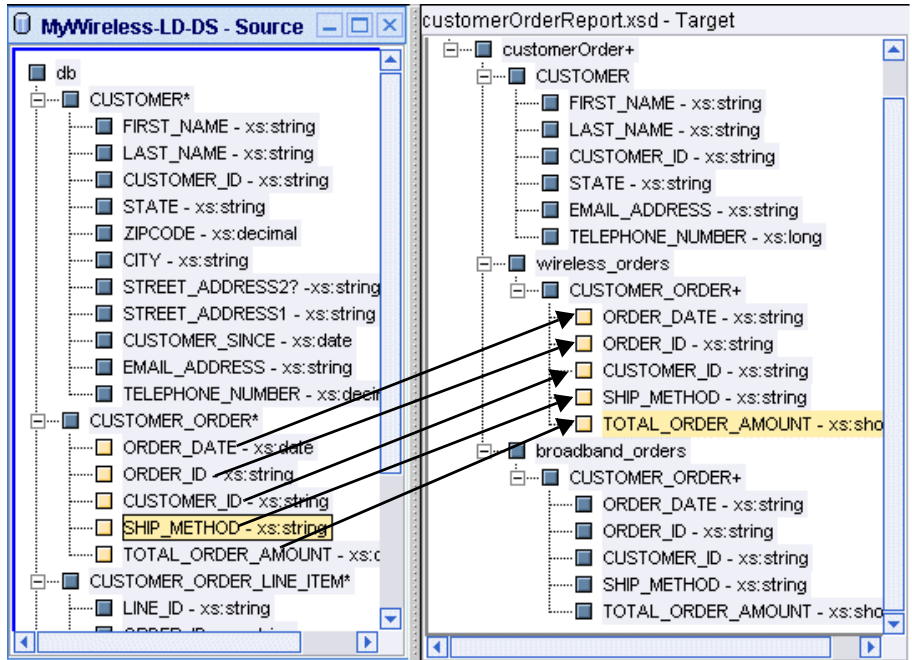


Project the Output by Mapping Source to Target Schema

Note: To expand the XML element trees in source and target schemas select the top element, right mouse-click to get the pop-up menu, and choose Expand.

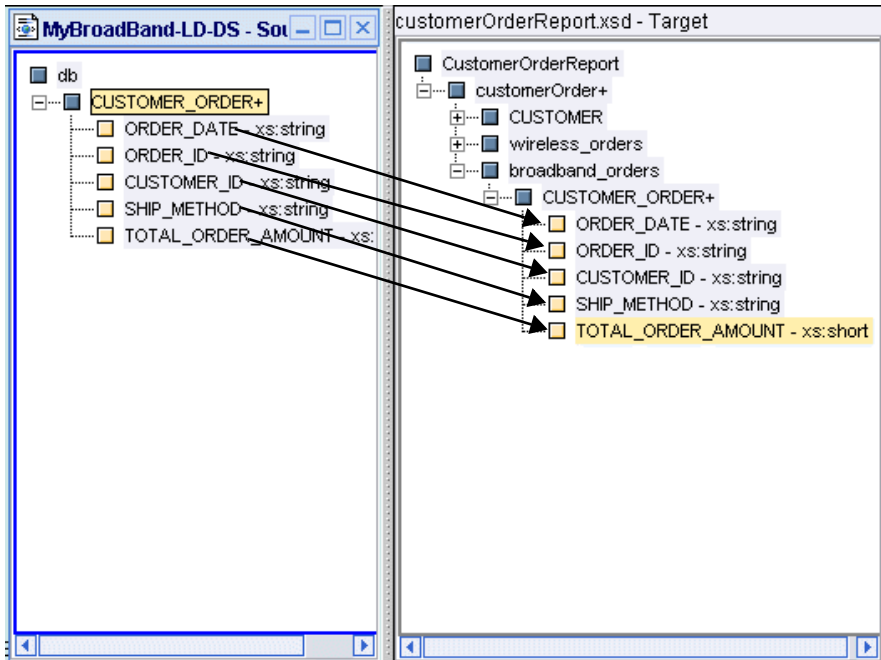
- The Order Query target schema includes a CUSTOMER_ORDER format for both Wireless and Broadband. So, we'll start by mapping the CUSTOMER_ORDER elements shown in each data source to the corresponding CUSTOMER_ORDER elements in the target schema as follows:
 - Drag and drop the sub-elements contained in the Wireless data source CUSTOMER_ORDER* into the corresponding sub-elements in the target schema wireless_orders CUSTOMER_ORDER+.

Figure 1-8 Mapping Wireless CUSTOMER_ORDER to Target Schema



- Drag and drop the sub-elements contained in the BroadBand data source CUSTOMER_ORDER* into the corresponding sub-elements in the target schema wireless_orders CUSTOMER_ORDER+

Figure 1-9 Mapping Broadband CUSTOMER_ORDER to Target Schema



In both cases, the sub-elements to map are ORDER_DATE, ORDER_ID, CUSTOMER_ID, SHIP_METHOD and TOTAL_ORDER_AMOUNT.

Note: As you drag and drop elements from source to target, the Mappings tab is automatically displayed in the lower part of the workspace. The mapped elements are highlighted in the schema windows and each mapping is recorded on the Mappings tab showing Source and Target details. If you need to delete a mapping, select it on the Mappings tab by clicking in the Mapping ID field to highlight the row in which the Source/Target mapping is shown and then click the Trashcan icon.

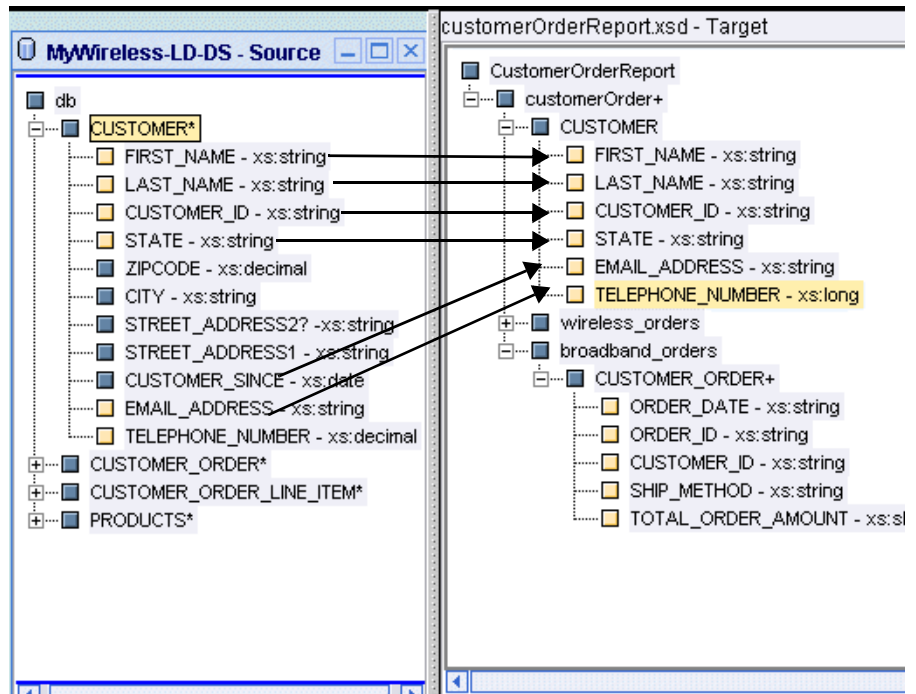
2. For this example, we use the Wireless data source schema to format the CUSTOMER element in the target schema. So we need to map the CUSTOMER elements shown in the Wireless data source to the CUSTOMER elements in the target.

1 Getting Started with Liquid Data: A Basic Example

To do this, drag and drop the sub-elements contained in the Wireless data source CUSTOMER* into the corresponding sub-elements in the target schema CUSTOMER.

Note: Note that you do not necessarily need to have a target for every source element. In our example, we create source-to-target mappings for every node that exists in our target CUSTOMER schema.

Figure 1-10 Mapping Wireless CUSTOMER to Target Schema CUSTOMER



At this point, you have completed the source to target mappings and the target schema is ready.

The source-to-target mappings you created are shown on the Mappings tab.

Figure 1-11 Source-to-Target Mappings

	Source	Target
0	[MyWireless-LD-DS]/db/CUSTOMER_ORDER/O...	[customerOrderReport.xsd]/CustomerOrderRep...
1	[MyWireless-LD-DS]/db/CUSTOMER_ORDER/O...	[customerOrderReport.xsd]/CustomerOrderRep...
2	[MyWireless-LD-DS]/db/CUSTOMER_ORDER/C...	[customerOrderReport.xsd]/CustomerOrderRep...
3	[MyWireless-LD-DS]/db/CUSTOMER_ORDER/S...	[customerOrderReport.xsd]/CustomerOrderRep...
4	[MyWireless-LD-DS]/db/CUSTOMER_ORDER/T...	[customerOrderReport.xsd]/CustomerOrderRep...
5	[MyBroadBand-LD-DS]/db/CUSTOMER_ORDER...	[customerOrderReport.xsd]/CustomerOrderRep...
6	[MyBroadBand-LD-DS]/db/CUSTOMER_ORDER...	[customerOrderReport.xsd]/CustomerOrderRep...
7	[MyBroadBand-LD-DS]/db/CUSTOMER_ORDER...	[customerOrderReport.xsd]/CustomerOrderRep...
8	[MyBroadBand-LD-DS]/db/CUSTOMER_ORDER...	[customerOrderReport.xsd]/CustomerOrderRep...
9	[MyBroadBand-LD-DS]/db/CUSTOMER_ORDER...	[customerOrderReport.xsd]/CustomerOrderRep...
10	[MyWireless-LD-DS]/db/CUSTOMER/FIRST_NA...	[customerOrderReport.xsd]/CustomerOrderRep...
11	[MyWireless-LD-DS]/db/CUSTOMER/LAST_NAME	[customerOrderReport.xsd]/CustomerOrderRep...
12	[MyWireless-LD-DS]/db/CUSTOMER/CUSTOME...	[customerOrderReport.xsd]/CustomerOrderRep...
13	[MyWireless-LD-DS]/db/CUSTOMER/STATE	[customerOrderReport.xsd]/CustomerOrderRep...
14	[MyWireless-LD-DS]/db/CUSTOMER/EMAIL_AD...	[customerOrderReport.xsd]/CustomerOrderRep...
15	[MyWireless-LD-DS]/db/CUSTOMER/TELEPHO...	[customerOrderReport.xsd]/CustomerOrderRep...
16		
17		

Mapping for: All Show full path

Mappings Conditions Sort By

Define the Source Conditions

The final step in constructing the data view for our example query is to define some source conditions. This is the way in which you actually pose the “question” (query) or filter designed to get at the data in a particular way. In this example, we want to find out which customers (whose customer IDs match a given constant) have orders in both BroadBand and Wireless.

- Create a Join Between the CUSTOMER_IDs in CUSTOMER and CUSTOMER_ORDER within Wireless
- Create a Join between Wireless CUSTOMER_ID and Broadband CUSTOMER_ID
- Use a Function to Retrieve a Customer ID Matching a Given Constant

Create a Join Between the CUSTOMER_IDs in CUSTOMER and CUSTOMER_ORDER within Wireless

In the Wireless data source, drag and drop the sub-element CUSTOMER_ID (from CUSTOMER) to CUSTOMER_ID (subelement in CUSTOMER_ORDER) within the same data source schema window.

This creates a join between CUSTOMER and CUSTOMER_ORDER. The syntax of this join is displayed on the first row on Conditions tab:

Note: As you drag and drop elements from source element to source element, the Conditions tab is automatically displayed in the lower part of the workspace. The mapped elements are highlighted in the schema windows and each source condition is recorded on the Conditions tab. If you need to delete a condition, select it on the Conditions tab by clicking in the Condition ID field to highlight the row in which the condition is shown and then click the Trashcan icon.

Create a Join between Wireless CUSTOMER_ID and Broadband CUSTOMER_ID

Drag and drop the Wireless element CUSTOMER/CUSTOMER_ID onto the Broadband data source element CUSTOMER/CUSTOMER_ID.

The syntax of this Join is displayed on the next row on the Conditions tab.

Note: If you ran the query at this point, you would get a query result showing all customers who have orders in both BroadBand and Wireless. In the next step, however, we refine the query a little further.

Use a Function to Retrieve a Customer ID Matching a Given Constant

You can use a function to create filter that matches a given customer.

1. If the Conditions tab is not already selected in the lower panel, click the Conditions tab.
2. On the Builder Toolbar (left navigation panel) click the Toolbox tab, then click Functions.
3. In the Functions panel, expand the Operators list and find the Equals function (eq).

4. Drag and drop the eq function from the Builder Toolbar into the first empty row in under Conditions on the lower part of the workspace.

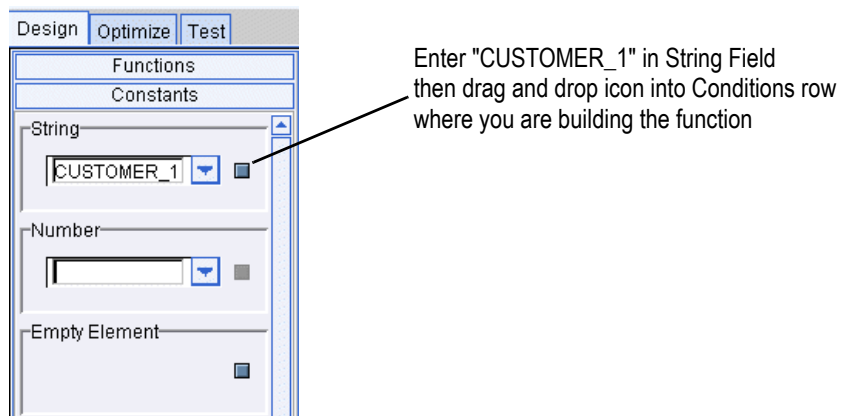
If you have been following along this will be the third row since it's the third source condition you are setting.

This causes the Functions Editor to open automatically. By dragging the eq function into the conditions you get a “placeholder” equation in the Functions Editor in the form of:

```
(anyValue1 - xset:anyValue eq anyValue2- xset:anyValue)
```

5. Drag the Wireless CUSTOMER_ID (a sub-element of CUSTOMER) into the left side of the equation (onto the first placeholder, “anyValue1 - xset:anyValue”).
6. On the left navigation panel, click Constants.
7. In the Constants String field, type CUSTOMER_1. Drag and drop the constants icon (to the right of the field) into the right side of the equation (onto the remaining placeholder, “anyValue2- xset:anyValue”).

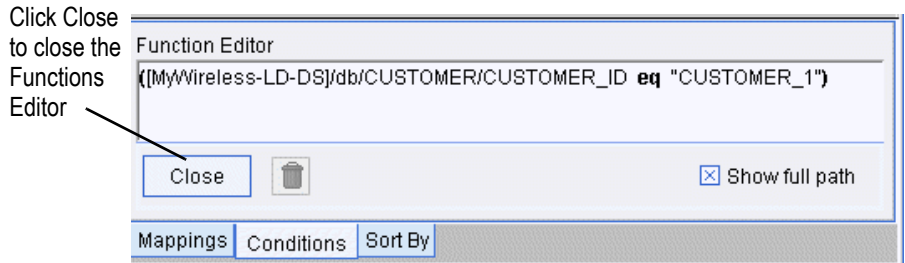
Figure 1-12 Using a String Constant in a Function



Note: Make sure that you have the left navigation panel expanded out far enough to the right to be able to see the arrow icons to the left of the Constants fields.

The syntax of this statement is displayed in the Functions Editor.

Figure 1-13 Equation for a Source Condition Shown in Functions Editor

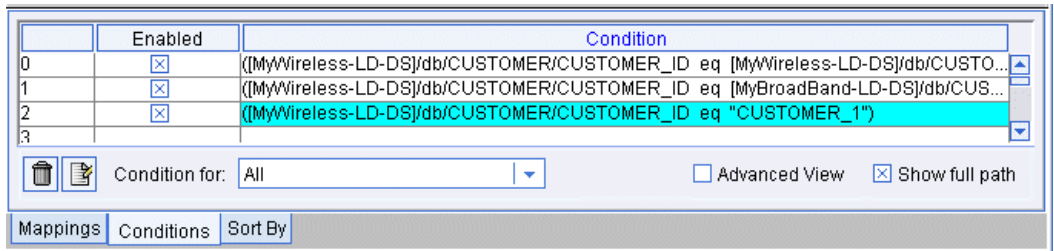


You can close the Functions Editor by clicking the Close button on it.

You have set up all the source conditions. They are reflected on the Conditions tab.

Note: To see the full text of the source conditions statements, you may have to expand the Conditions area by clicking and dragging the left or right borders.

Figure 1-14 Source Conditions



Step 6. View and Test the Query

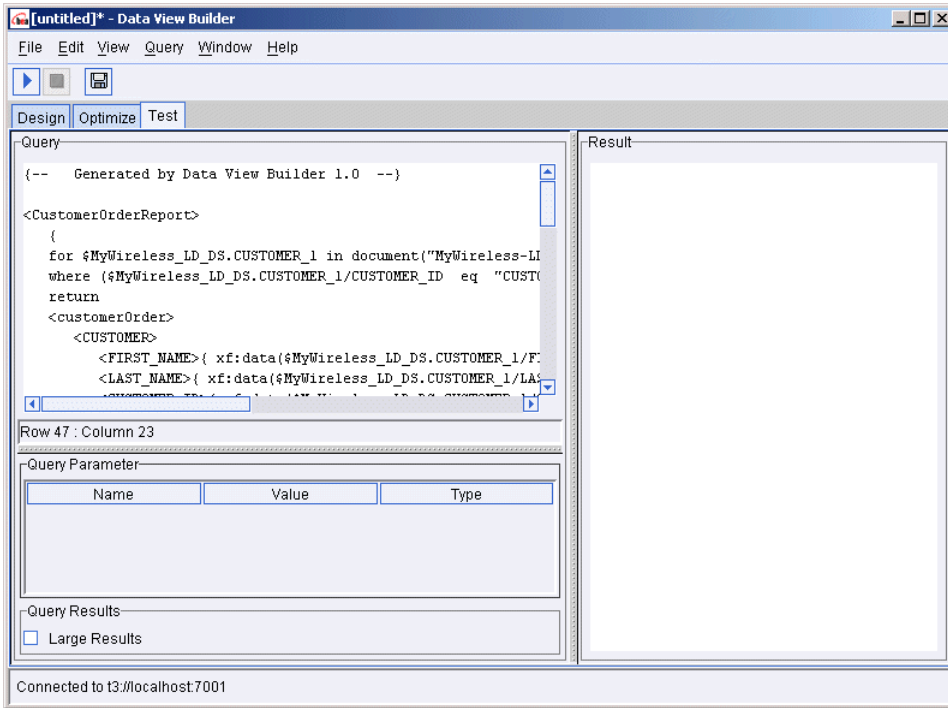
Switch to Test View

Now you are ready to switch to the Test tab where you can look at the XQuery syntax of the generated query you just constructed, and then *run* the query to test it.

Click the Test tab at the top of the workspace.

The Test tab is displayed as shown in the following figure.

Figure 1-15 Test Tab



Understanding the Generated XQuery

The Data View Builder generates the following XQuery for the query you constructed with drag-and-drop tools.

Listing 1-1 Generated XQuery for Getting Started Query

```
{--      Generated by Data View Builder 1.0--}
<CustomerOrderReport>
{
for $MyWireless_LD_DS.CUSTOMER_1 in document("MyWireless-LD-DS")/db/CUSTOMER
where ($MyWireless_LD_DS.CUSTOMER_1/CUSTOMER_ID eq "CUSTOMER_1")
return
<customerOrder>
<CUSTOMER>
<FIRST_NAME>{ xf:data($MyWireless_LD_DS.CUSTOMER_1/FIRST_NAME) }</FIRST_NAME>
<LAST_NAME>{ xf:data($MyWireless_LD_DS.CUSTOMER_1/LAST_NAME) }</LAST_NAME>
}
</CustomerOrderReport>
```

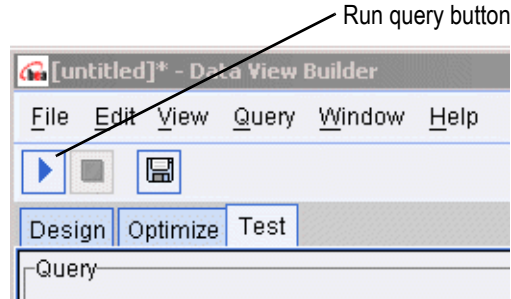
1 Getting Started with Liquid Data: A Basic Example

```
<CUSTOMER_ID>{ xf:data($MyWireless_LD_DS.CUSTOMER_1/CUSTOMER_ID) }</CUSTOMER_ID>
<STATE>{ xf:data($MyWireless_LD_DS.CUSTOMER_1/STATE) }</STATE>
<EMAIL_ADDRESS>{ xf:data($MyWireless_LD_DS.CUSTOMER_1/EMAIL_ADDRESS) }</EMAIL_ADDRESS>
<TELEPHONE_NUMBER>{ cast as
xs:long(xf:data($MyWireless_LD_DS.CUSTOMER_1/TELEPHONE_NUMBER)) }</TELEPHONE_NUMBER>
</CUSTOMER>
  <wireless_orders>
  {
    for $MyWireless_LD_DS.CUSTOMER_ORDER_2 in document("MyWireless-LD-DS")/db/CUSTOMER_ORDER
    where ($MyWireless_LD_DS.CUSTOMER_1/CUSTOMER_ID eq
$MyWireless_LD_DS.CUSTOMER_ORDER_2/CUSTOMER_ID)
    return
    <CUSTOMER_ORDER>
    <ORDER_DATE>{ cast as xs:string(xf:data($MyWireless_LD_DS.CUSTOMER_ORDER_2/ORDER_DATE))
}</ORDER_DATE>
    <ORDER_ID>{ xf:data($MyWireless_LD_DS.CUSTOMER_ORDER_2/ORDER_ID) }</ORDER_ID>

    <CUSTOMER_ID>{ xf:data($MyWireless_LD_DS.CUSTOMER_ORDER_2/CUSTOMER_ID) }</CUSTOMER_ID>
    <SHIP_METHOD>{ xf:data($MyWireless_LD_DS.CUSTOMER_ORDER_2/SHIP_METHOD) }</SHIP_METHOD>
    <TOTAL_ORDER_AMOUNT>{ xf:data($MyWireless_LD_DS.CUSTOMER_ORDER_2/TOTAL_ORDER_AMOUNT)
}</TOTAL_ORDER_AMOUNT>
    </CUSTOMER_ORDER>
  }
</wireless_orders>
  <broadband_orders>
  {
    for $MyBroadBand_LD_DS.CUSTOMER_ORDER_3 in document("MyBroadBand-LD-DS")/db/CUSTOMER_ORDER
    where ($MyWireless_LD_DS.CUSTOMER_1/CUSTOMER_ID eq
$MyBroadBand_LD_DS.CUSTOMER_ORDER_3/CUSTOMER_ID)
    return
    <CUSTOMER_ORDER>
    <ORDER_DATE>{ xf:data($MyBroadBand_LD_DS.CUSTOMER_ORDER_3/ORDER_DATE) }</ORDER_DATE>
    <ORDER_ID>{ xf:data($MyBroadBand_LD_DS.CUSTOMER_ORDER_3/ORDER_ID) }</ORDER_ID>
    <CUSTOMER_ID>{ xf:data($MyBroadBand_LD_DS.CUSTOMER_ORDER_3/CUSTOMER_ID) }</CUSTOMER_ID>
    <SHIP_METHOD>{ xf:data($MyBroadBand_LD_DS.CUSTOMER_ORDER_3/SHIP_METHOD) }</SHIP_METHOD>
    <TOTAL_ORDER_AMOUNT>{ cast as
xs:decimal(xf:data($MyBroadBand_LD_DS.CUSTOMER_ORDER_3/TOTAL_ORDER_AMOUNT)) }</TOTAL_ORDER_AMOUNT>
    </CUSTOMER_ORDER>
  }
</broadband_orders>
</customerOrder>
}
</CustomerOrderReport>
```

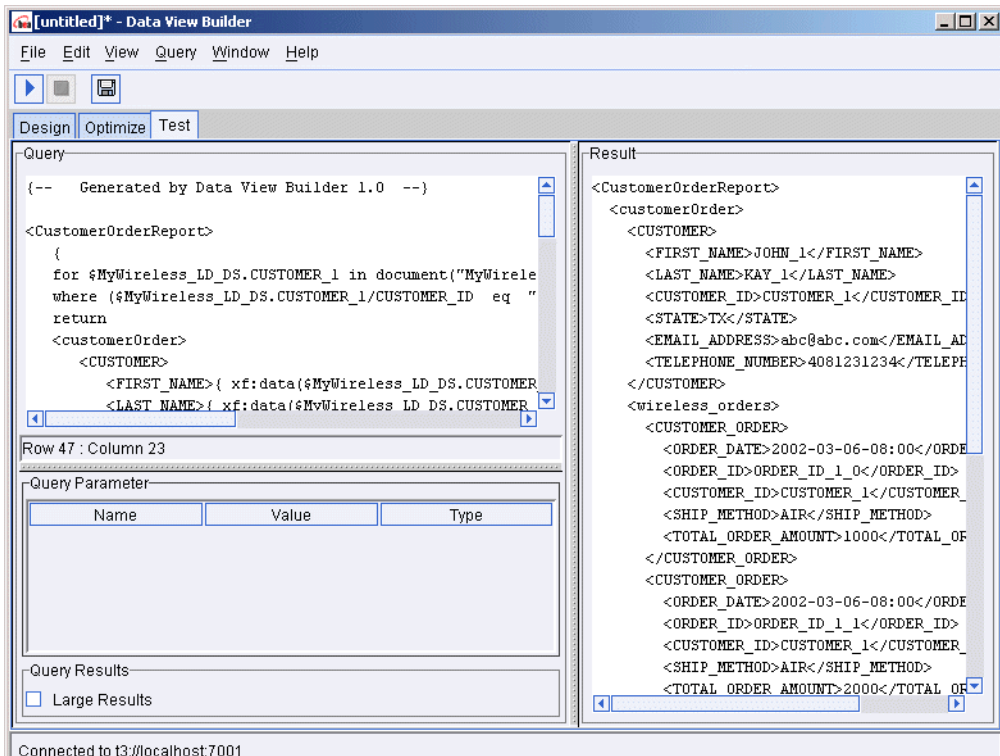
Run the Query

When you are ready to run the query, click the Run query button on the toolbar in the upper left.



The Liquid Data server process the query and the query result is displayed as shown in the following figure and code listing.

Figure 1-16 Query Result on Test Tab



1 Getting Started with Liquid Data: A Basic Example

Listing 1-2 XML for the Query Result

```
<CustomerOrderReport>
  <customerOrder>
    <CUSTOMER>
      <FIRST_NAME>JOHN_1</FIRST_NAME>
      <LAST_NAME>KAY_1</LAST_NAME>
      <CUSTOMER_ID>CUSTOMER_1</CUSTOMER_ID>
      <STATE>TX</STATE>
      <EMAIL_ADDRESS>abc@abc.com</EMAIL_ADDRESS>
      <TELEPHONE_NUMBER>4081231234</TELEPHONE_NUMBER>
    </CUSTOMER>
    <wireless_orders>
      <CUSTOMER_ORDER>
        <ORDER_DATE>2002-03-06-08:00</ORDER_DATE>
        <ORDER_ID>ORDER_ID_1_0</ORDER_ID>
        <CUSTOMER_ID>CUSTOMER_1</CUSTOMER_ID>
        <SHIP_METHOD>AIR</SHIP_METHOD>
        <TOTAL_ORDER_AMOUNT>1000</TOTAL_ORDER_AMOUNT>
      </CUSTOMER_ORDER>
      <CUSTOMER_ORDER>
        <ORDER_DATE>2002-03-06-08:00</ORDER_DATE>
        <ORDER_ID>ORDER_ID_1_1</ORDER_ID>
        <CUSTOMER_ID>CUSTOMER_1</CUSTOMER_ID>
        <SHIP_METHOD>AIR</SHIP_METHOD>
        <TOTAL_ORDER_AMOUNT>2000</TOTAL_ORDER_AMOUNT>
      </CUSTOMER_ORDER>
    </wireless_orders>
    <broadband_orders>
      <CUSTOMER_ORDER>
        <ORDER_DATE>2002-04-09</ORDER_DATE>
        <ORDER_ID>ORDER_ID_1_0</ORDER_ID>
        <CUSTOMER_ID>CUSTOMER_1</CUSTOMER_ID>
        <SHIP_METHOD>AIR</SHIP_METHOD>
        <TOTAL_ORDER_AMOUNT>1000.00</TOTAL_ORDER_AMOUNT>
      </CUSTOMER_ORDER>
      <CUSTOMER_ORDER>
        <ORDER_DATE>2002-04-09</ORDER_DATE>
        <ORDER_ID>ORDER_ID_1_1</ORDER_ID>
        <CUSTOMER_ID>CUSTOMER_1</CUSTOMER_ID>
        <SHIP_METHOD>AIR</SHIP_METHOD>
        <TOTAL_ORDER_AMOUNT>1500.00</TOTAL_ORDER_AMOUNT>
      </CUSTOMER_ORDER>
    </broadband_orders>
  </customerOrder>
</CustomerOrderReport>
```

Congratulations! You have successfully built and run a query using Liquid Data.

Note: If you want to switch back to the Design mode again, click the Design View tab on the toolbar.

Step 7. Save the Project

We suggest you save the project file at this point if you have not done so already.

To save the project for the first time:

1. Choose the menu option File—>Save Project. The Save dialog is displayed.
2. Navigate to the directory in which you want to save your project and enter a name for it in the File Name field.
3. Click Save.

The file is saved as a Data View Builder project with a `.qpr` extension.

Note: Saving the project file is not the same as saving a query. For more information on working with and saving projects, see [Working with Projects](#) in the *Building Queries and Data Views*. For information on saving a query as a *stored query*, see [Saving a Query](#) also in *Building Queries and Data Views*.

Next Steps

The process you just completed with these Getting Started steps is a thin slice of the Liquid Data picture meant to help familiarize you with some of the basics. If you are ready to learn more, try the following topics:

- See the Liquid Data [Product Overview](#) for an end-to-end description of the product and how to use it, including designing your business solution, using security features, setting up clustering, creating the presentation layer, understanding the recommended deployment scenarios, and using Liquid Data with other BEA products like BEA WebLogic Portal, BEA Application Integration, and BEA WebLogic Workshop.

1 *Getting Started with Liquid Data: A Basic Example*

- See [Building Queries and Data Views](#) for a detailed GUI overview and description of Data View Builder features, the complete set of tasks you can accomplish with the Builder, a full “Query Cookbook”, and lots more.
- See the Liquid Data [Administration Guide](#) for information about how to configure all the data source types supported Liquid Data supports including Application Integration (AI), how to configure Security, how to set up monitoring and reporting.

Index

- A**
- Administration Console
 - logging in 1-3
 - starting 1-3
 - URL 1-3
 - where to find more information 1-30
- C**
- conditions, creating 1-21
- D**
- Data View Builder
 - starting 1-13
 - URL 1-13
 - demo on how to construct query 1-15
 - Design tab 1-14
- F**
- functions editor
 - closing 1-24
 - opening 1-23
 - working in 1-24
- G**
- generated XQuery, viewing on the Test tab 1-25
- J**
- JDBC connection pool, configuring in WLS 1-5
 - JDBC data source, configuring in WLS 1-7
- L**
- logging in to the Administration Console 1-3
- Q**
- query result, viewing on the Test tab 1-27
- R**
- relational database
 - configuring as Liquid Data data source 1-9
 - steps for configuring as Liquid Data data source 1-5
 - result, viewing on the Test tab 1-27
 - running a query 1-26
- S**
- Samples server, starting 1-2
 - source schemas
 - mapping elements to target schema 1-17
 - opening/adding to a project 1-16
 - starting
 - Administration Console 1-3
 - Data View Builder 1-13

Samples server 1-2
string constant 1-23

T

target schema
 configuring 1-17
 mapping 1-17
 opening/adding from server Repository
 1-16
Test tab 1-24
testing a query 1-24

U

URL
 Administration Console 1-3
 Data View Builder 1-13

X

XML file, configuring as Liquid Data data
 source 1-11
XML query result, viewing on Test tab 1-28
XQuery, viewing on Test tab 1-25