



# BEA Liquid Data for WebLogic™

## Frequently Asked Questions

# Copyright

Copyright © 2003 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.

## Frequently Asked Questions

<b>Part Number</b>	<b>Date</b>	<b>Software Version</b>
N/A	October 2002	1.0
N/A	April 2003	1.1

---

# Contents

## About This Document

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

## 1. Frequently Asked Questions

XML Query Language .....	1-1
Where can I find more information on XQuery? .....	1-1
How does XQuery compare to SQL?.....	1-2
How does XQuery compare to XPath and XSLT? .....	1-2
Does XQuery contain Business Logic?.....	1-3
Does Liquid Data Use XSLT? If not, how do you support data mapping and transformation? .....	1-3
Why is XQuery preferable to XSLT for XML-to-XML transformations? .....	1-3
Data Sources and Data Access .....	1-4
How does Liquid Data integrate data from relational databases? .....	1-4
Can the Liquid Data Server connect to non-relational data sources? .....	1-5
XML Files .....	1-5
Web Services.....	1-5
Application Views.....	1-5
Other Data Sources .....	1-5
What support does the Liquid Data have for “Web Services?” .....	1-6
Is the platform bi-directional? How do you update backend sources? .....	1-6
Does Liquid Data support RosettaNet, cXML, ebXML, FiXML? .....	1-6

---

How does Liquid Data integrate with hierarchical database sources? .....	1-6
How does Liquid Data handle VSAM/ISAM files from mainframe systems? 1-7	
Liquid Data Server and Technology .....	1-7
Does the platform cache query results? .....	1-7
What security (authentication/access control) policies are built into the platform? .....	1-7
Is there support for hierarchical views? .....	1-8
Is there a repository for views? .....	1-8
Architecture, System Configuration, Deployment .....	1-8
What platforms does Liquid Data support? .....	1-8
Performance and Security .....	1-9
How does the Liquid Data handle access control and authentication? .....	1-9
Administration and Graphical Tools .....	1-9
Does Liquid Data support data mining of stored meta-data? .....	1-9
Does Liquid Data support searches of stored meta-data? .....	1-9
IDI Applications and Complementary Technologies .....	1-10
What are some of the common characteristics of IDI applications (or alternately what applications are enabled by Liquid Data?) .....	1-10
How does the Liquid Data IDI offering complement EAI solutions? .....	1-10

## **Index**

---

# About This Document

This document provides answers to frequently asked questions about BEA Liquid Data for WebLogic™.

This document covers the following topics:

- [Chapter 1, “Frequently Asked Questions,”](#)

## What You Need to Know

This document is intended mainly for users who have questions about the Liquid Data product.

## 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

The following BEA Liquid Data documents contain information that is relevant to using Liquid Data:

- Installing Liquid Data
- Administering Liquid Data
- Using the Data View Builder

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](mailto: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 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](http://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
<b>boldface text</b>	Indicates terms defined in the glossary.
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.

---

<b>Convention</b>	<b>Item</b>
<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
<b>monospace boldface text</b>	Identifies significant words in code.  <i>Example:</i> void <b>commit</b> ( )
<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.
...	<p>Indicates one of the following in a command line:</p> <ul style="list-style-type: none"> <li>■ That an argument can be repeated several times in a command line</li> <li>■ That the statement omits additional optional arguments</li> <li>■ That you can enter additional parameters, values, or other information</li> </ul> <p>The ellipsis itself should never be typed.</p> <p><i>Example:</i></p> <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 Frequently Asked Questions

This topic provides answers to some frequently asked questions about BEA Liquid Data for WebLogic™. It contains the following sections:

- [XML Query Language](#)
- [Data Sources and Data Access](#)
- [Liquid Data Server and Technology](#)
- [Architecture, System Configuration, Deployment](#)
- [Performance and Security](#)
- [Administration and Graphical Tools](#)
- [IDI Applications and Complementary Technologies](#)

## XML Query Language

### Where can I find more information on XQuery?

Refer to the topic [XQuery Links](#) in “Liquid Data Concepts” in our *Product Overview*. A complete list of links to W3C XQuery specifications and documentation is provided, along with more discussion about the use of XQuery in Liquid Data.

### **How does XQuery compare to SQL?**

XQuery brings “SQL-like” querying power to applications that require access, selection, integration and transformation of information from one or more XML sources. XQuery is to XML what SQL has been to relational data. However, the combination of XQuery and XML has significant advantages compared to the combination of SQL and relational data. SQL is primarily a language that developers use to retrieve data from relational sources. XQuery provides retrieval as well as advanced processing capabilities. XQuery lets developers access multiple XML data sets (from distributed, disparate XML sources) and integrate, manipulate, transform, and filter them in order to make them fit the needs of the target application. XQuery and XML release the developer and the application from the bounds of tuple-based processing and allow him to morph a wide range of input data into the structure that the application requires. This is particularly important for Web-based applications that need to convey data to the user in a form that is most easy to digest. Most often the nested and flexible structure of XML matches the user needs better than the flat relational structures returned by SQL.

### **How does XQuery compare to XPath and XSLT?**

Before XQuery, developers used the XPath query language to locate items in an XML document and the Extensible Stylesheet Language Transformation engine (XSLT) to transform XML information to XHTML or transform one XML format to another XML format. For instance, a developer could use XPath to retrieve a customer record from an XML file and XSLT to package the name and phone number of the customer in a new XML structure. But XPath and XSLT work only on XML files. If one wants to work on other types of sources he will need to write code that will access the sources and formulate the XML structure that will be passed to XPath or XSLT. Furthermore, both the “selection-oriented” XPath and the “transformation-oriented” XSLT are only small parts of the full solution, which requires efficient selection, join, integration, and transformation to the application needs. XQuery addresses all the data needs of the Web-based application: XQuery statements can access multiple sources, select information from them, join it, and transform it to meet the application needs.

Furthermore, XQuery is a declarative language that is amenable to query optimization - of the form that SQL query processors employ. Advanced optimizers, such as are provided with Liquid Data, optimize the given XQuery to achieve efficient access to the sources, focused selection of data, and quick integration and transformation.

## Does XQuery contain Business Logic?

No, XQuery encapsulates the Internet Data Integration Logic required for accessing multiple, disparate data sources in real-time. XQuery lets developers access multiple XML data sets (from distributed, disparate XML sources) and integrate, manipulate, transform, and filter them in order to make them fit the needs of the target application.

## Does Liquid Data Use XSLT? If not, how do you support data mapping and transformation?

XSLT is not used for mappings and transformations of source data. The XQuery language used by our product contains the integration logic, mappings and transformations associated with a distributed XQuery. XQuery statements can access multiple disparate sources, select information from them, join it, and transform it to meet the application needs.

A GUI tool, Data View Builder, provides drag-and-drop functionality to create XQueries using underlying data sources. The Liquid Data Server parses and executes an XQuery at run-time producing XML results.

## Why is XQuery preferable to XSLT for XML-to-XML transformations?

XSLT has been traditionally used for building XML->HTML transformations. Some attempts have been made to leverage XSLT for building XML to XML transformations also. However, XQuery is the preferable option.

Without XQuery, if the developer uses XSLT, the developer needs the following:

1. A wrapper/adaptor system to access the sources and get XML output. For example, if it is Oracle, one needs to use the XSQL system to get XML out of the tables.
2. XSLT must transform the XML file/message received by the wrapper/adaptor into the XML that the application requires.

The Pain points associated with this approach are:

- The developer has to use two disconnected systems and two programming paradigms.
- XSLT is particularly cumbersome for join, aggregation, and non-trivial selections.
- XSLT is also inefficient and it is impossible to optimize XSLT transformations for better run-time performance

The XQuery approach used by the Liquid Data Server provides the developer the ability to specify data access and transformation in a unique concise XQuery statement. The Benefits for the developer are as follows:

- One system to work with.
- Powerful for transformations that involve joins, aggregation, and non-trivial selections.
- The Liquid Data Server performs optimization of the joint data access and transformation task. The queries can be optimized and fine tuned for superior run-time performance.

## **Data Sources and Data Access**

### **How does Liquid Data integrate data from relational databases?**

The Liquid Data server accepts relational databases as data sources. Multiple schemas from the same database can also be easily integrated. The Liquid Data server can access any RDBMS that is accessible through a JDBC driver. The relational translator in Liquid Data also exports an XML Schema that allows the use of the Query Builder for specifying transformation and mappings of the data found in the relational database.

## **Can the Liquid Data Server connect to non-relational data sources?**

The Liquid Data server can integrate data from XML files, Web Services, packaged ERP, CRM, SCM, and legacy applications.

### **XML Files**

XML files can be easily defined and accessed by the Liquid Data server for integration. By specifying the XML file source and schema in the server configuration, schemas are associated with XML files and queries optimized for accessing them.

### **Web Services**

The Web Services translator in Liquid Data tightly integrates SOAP calls into the server query engine. XML data generated from the calls get fed directly into the query engine for transformation, navigation and processing. Automatic registration of SOAP calls associated with Web Services that provide WSDL descriptions is possible with the Data Integration Suite.

### **Application Views**

Liquid Data supports the use of several Enterprise Information Systems (EIS) and custom applications as Application View data sources. Liquid Data interoperates with the BEA WebLogic Integration Application Integration (AI) layer through the Liquid Data Application View interface. Liquid Data can retrieve information through commercial application-specific adapters such as SAP, Siebel, PeopleSoft, Oracle Financial and so on which are certified to work with BEA WebLogic Integration.

### **Other Data Sources**

Access to data sources other than those defined above, such as APIs for packaged applications (ERP, CRM, SCM etc.) can be facilitated through Application Views using BEA WebLogic Integration (WLI) application integration (AI). Liquid Data leverages BEA WebLogic Integration adapters to connect to legacy systems.

Access to other data sources can also be achieved through custom functions; that is, user-defined functions written in Java. Writing these functions is fairly simple and guidelines on creating them and integrating them with the Liquid Data server are available in the product documentation. These functions can be directly integrated into complex queries built in Liquid Data.

### **What support does the Liquid Data have for “Web Services?”**

The Web Services translator (or SOAP translator) in Liquid Data tightly integrates SOAP calls into the server query engine. XML data generated from the calls get fed directly into the query engine for transformation, navigation and processing. Automatic registration of SOAP calls associated with Web Services that provide WSDL descriptions is possible with the Data Integration Suite.

### **Is the platform bi-directional? How do you update backend sources?**

Bidirectional source updates are not supported in the Technical Preview version of Liquid Data.

### **Does Liquid Data support RosettaNet, cXML, ebXML, FiXML?**

Liquid Data supports all XML document formats, including industry or function-specific formats like RosettaNet, cXML, ebXML, and FiXML.

### **How does Liquid Data integrate with hierarchical database sources?**



Liquid Data integrates hierarchical database sources through the WebLogic Server JDBC facilities. For more information on how to configure databases as data sources in Liquid Data, see [Configuring Access to Relational Databases](#) in the *Liquid Data Administration Guide*.

## **How does Liquid Data handle VSAM/ISAM files from mainframe systems?**

Liquid Data can manage files from mainframe systems through the WebLogic Server JDBC facilities.

# **Liquid Data Server and Technology**

## **Does the platform cache query results?**

Liquid Data provides the option to enable caching on query results for stored queries. See [Configuring the Query Cache](#) in the *Liquid Data Administration Guide*.

## **What security (authentication/access control) policies are built into the platform?**

Liquid Data uses BEA WebLogic Server Compatibility Security for implementation of security on various components of Liquid Data by means of access control lists for users and groups. Using the security features in the WLS Administration Console, you can configure access to administrative privileges, stored queries, data sources, the Data View Builder tool, the Liquid Data server repository, and more. For information, see [Implementing Security](#) in the *Liquid Data Administration Guide*.

### **Is there support for hierarchical views?**

Yes. Users can build varying level of nested data views. For more information, refer to the discussion of [data views](#) in “Liquid Data Concepts” in the Product Overview, and to the topic [Using Data Views as Data Sources](#) in *Building Queries and Data Views*.

### **Is there a repository for views?**

Yes. Data views are stored in Liquid Data server file-based repository. For information, see the topic [Using Data Views as Data Sources](#) in *Building Queries and Data Views* and [Managing the Liquid Data Server Repository](#) in the *Liquid Data Administration Guide*.

## **Architecture, System Configuration, Deployment**

### **What platforms does Liquid Data support?**

Liquid Data is a 100% Java platform built on BEA WebLogic Platform 7.0, and is not dependent on any specific hardware or operating system configuration. The current version of Liquid Data runs on Sun Solaris and Windows NT/2000 platforms. For detailed information on supported platforms, see [Supported Platforms and Databases](#) in the *Liquid Data Installation Guide*.

# Performance and Security

## How does the Liquid Data handle access control and authentication?

Liquid Data implements security based on BEA WebLogic Server Compatibility Security. It relies on the WLS JAAS authentication and Access Control List (ACL) based authorization. For information, see [Implementing Security](#) in the *Liquid Data Administration Guide*.

## Administration and Graphical Tools

### Does Liquid Data support data mining of stored meta-data?

Data Mining applications need an integrated view of enterprise data. Liquid Data can provide such integrated views of mission-critical data in real-time. Though the Liquid Data server does not contain specific data mining algorithms, it can be easily integrated with other vendor tools which support data mining capabilities.

### Does Liquid Data support searches of stored meta-data?

The Liquid Data server fully supports metadata repositories built on top of all relational databases and XML files. There are no search capabilities provided in the current version of the platform.

# IDI Applications and Complementary Technologies

## What are some of the common characteristics of IDI applications (or alternately what applications are enabled by Liquid Data?)

Common characteristics of IDI applications include:

- Need for real-time, integrated data from multiple, disparate data sources
- Re-usable, secure, multi-source, enterprise data assets driving mission-critical web-based applications
- Seamless beyond-the-firewall access for customers, partners, suppliers

## How does the Liquid Data IDI offering complement EAI solutions?

EAI solutions focus on

- Transaction-based application integration
- Process-based workflow solutions
- Application-to-application messaging

The IDI offering from BEA is complementary to EAI solutions and offers a universal data integration platform to aggregate, transform and filter from multiple backend data sources. The secure, re-usable XML views created by BEA can be easily integrated with a workflow or process flow tool from EAI vendors.

BEA is focused on solving the challenges of Internet Data Integration (which consume upto 70% of a typical integration project budget). By employing a "declarative" query-based approach to integrate disparate, multi-source data, BEA can expedite the task of deploying new composite applications such as internal portals, self-service customer and supplier portals.



---

# Index

## **C**

customer support contact information iii-vii

## **D**

documentation, where to find it iii-v

## **P**

printing product documentation iii-vi

## **R**

related information iii-vi

## **S**

support  
    technical iii-vii