

Oracle9iAS TopLink

CMP-Specific Release Notes

Release 2 (9.0.3)

August 2002

Part No. B10161-01

These release notes include information on using Oracle9iAS TopLink Release 2 (9.0.3) with the following CMPs:

- [TopLink CMP for BEA WebLogic Server](#)
- [TopLink CMP for IBM WebSphere Server](#)

Note: Effective with the release of Oracle9iAS TopLink Release 2 (9.0.3), Oracle9iAS TopLink no longer provides CMP support for the HP Application Server.

1 TopLink CMP for BEA WebLogic Server

Oracle9iAS TopLink CMP for BEA WebLogic Server is compatible with BEA WebLogic Server 6.1 GA and above, and with BEA WebLogic Server 7.0 GA and above. Entity beans deployed with TopLink with BEA WebLogic Server adheres to either the EJB 1.1 or EJB 2.0 specification.

1.1 EJB 2.0 Support

Oracle9iAS TopLink provides support for EJB 2.0 CMP entity beans. TopLink's implementation is based on the EJB 2.0 support provided in BEA WebLogic Server 6.1 and 7.0.

Some specifics of EJB 2.0 that are supported are:

- Local interfaces
- Generation of the concrete subclass of your abstract bean.
- EJB-QL
- ejbSelects
- ejbHome methods

ORACLE®

Copyright © 2002 Oracle Corporation.
All Rights Reserved.

Oracle is a registered trademark, and Oracle9i is a trademark or registered trademark of Oracle Corporation. Other names may be trademarks of their respective owners.

- Automatic management of bi-directional relationships
- Mapping Workbench support for EJB 2.0
 - Initializing project from `ejb-jar.xml`
 - Support for finders

1.2 Installation Changes

You need to set the value of **JAVA_HOME** in the `<INSTALL_DIR>\setenv.cmd` file (for example, `c:\jdk1.3.1`).

Ensure `<INSTALL_DIR>\setenv.cmd` has the correct values for the **WLS61_HOME** (for example, `C:\bea\wlserver6.1`), or **WLS70_HOME** (for example, `C:\bea\weblogic700\server`). If the application servers are installed on your machine before you install TopLink, they are automatically detected by the TopLink installer and the variables set correctly.

The main structure now looks like:

```
<ORACLE_HOME>
  toplink
    doc
    package_rename
    examples
    core
      lib
        toplink.jar
    wls_cmp
      persistence.install
      lib
        tl_wlsx.jar
```

1.3 Examples

To familiarize yourself with the CMP functionality in TopLink CMP for BEA WebLogic Server, we recommend you explore the TopLink Examples.

1.4 Version Change for XML Deployment Descriptors

When using BEA WebLogic Server 6.1, the `<type-version>` specified in the `weblogic-ejb-jar.xml` should be **4.0**. The `<type-identifier>` should be **TopLink_CMP_1_1** for EJB 1.1 entity beans and **TopLink_CMP_2_0** for EJB 2.0 entity beans.

```
<persistence-descriptor>
  <persistence-type>
    <type-identifier>TopLink_CMP_2_0</type-identifier>
```

```

        <type-version>4.0</type-version>
        <type-storage>META-INF/toplink-ejb-jar.xml</type-storage>
    </persistence-type>
    <persistence-use>
        <type-identifier>TopLink_CMP_2_0</type-identifier>
        <type-version>4.0</type-version>
    </persistence-use>
</persistence-descriptor>

```

When using BEA WebLogic Server 7.0, the `<type-version>` specified in the `weblogic-ejb-jar.xml` should be **4.5**. The `<type-identifier>` should be **TopLink_CMP_1_1** for EJB 1.1 entity beans and **TopLink_CMP_2_0** for EJB 2.0 entity beans.

```

<persistence-descriptor>
    <persistence-use>
        <type-identifier>TopLink_CMP_2_0</type-identifier>
        <type-version>4.5</type-version>
        <type-storage>META-INF/toplink-ejb-jar.xml</type-storage>
    </persistence-use>
</persistence-descriptor>

```

For more information on deployment, refer to the [Oracle9iAS TopLink CMP for BEA WebLogic Server Guide](#). The `weblogic-ejb-jar.xml` file is used for both EJB 1.1 and EJB 2.0 beans.

1.5 Collection Set Methods

1.5.1 Dependent Object Collections

Unlike Enterprise JavaBeans, TopLink dependent persistent objects can be sent back and forth between a client and the server. When these objects are serialized, they can cause the cache to lose the identity of the objects or attempt to cache duplicate identical objects. To avoid potential problems, you use the bean set methods when adding dependent objects to relationship collections. This enables TopLink to handle merging of objects in the cache.

Example: Adding a dependent object

```

addPhoneNumber(PhoneNumber phone) {
    Collection phones = this.getPhoneNumbers();
    Vector newCollection = new Vector();
    newCollection.addAll(phones);
    newCollection.add(phone);
    this.setPhones(newCollection);
}

```

1.6 JDK Support

In Oracle9iAS TopLink Release 2 (9.0.3), only JDK 1.3 .jars are provided since BEA WebLogic Server 6.1 runtime requires the use of JDK 1.3. However, the TopLink JAR files will work properly with JDK 1.2 should you choose to use that version for your development.

1.7 Known Limitations

1.7.1 Relationship Management

We currently do not support automatic management of uni-directional relationships.

1.7.2 Limitations on Instantiation Policy

Custom instantiation policies that provide custom methods for creating entities do not work with WebLogic 6.1 entities. This is due to the fact that WebLogic subclasses the bean class and uses this subclass for persistence purposes. The "class dictionary" method does work.

1.7.3 Proxy Indirection

The Proxy Indirection feature (introduced in TopLink for Java 3.0 for JDK 1.3) can only be used on relationships between beans and dependent objects, and between regular Java objects. Proxy Indirection cannot be used to fault in entity beans.

1.7.4 Problems with ValueHolders

ValueHolders, including those used through transparent indirection and proxy indirection, will not be valid when serialized to the client. It is important that ValueHolders be used on server-side components only (entity beans) or that they are instantiated before being serialized to the client.

1.7.5 Using TopLink for Java APIs Directly

Access to lower-level TopLink for Java Foundation Library APIs are limited when using the TopLink CMP with BEA WebLogic Server. Direct manipulation of the TopLink **Session**, **UnitOfWork**, and other APIs is not encouraged when CMP is being used. Direct access to these APIs may compromise cache security and integrity, among other problems. Some TopLink Events cannot be used for database access when used within a JTS transaction. The JTS/JTA architecture does not permit database access during the commit process. Consequently, TopLink Events cannot be used to read or write during the commit.

1.7.6 Native Sequencing and Primary Key Allocation

If native sequencing is used on a database that does not support pre-allocation, primary keys will not be populated until the end of the transaction. This means that the `postCreate()` method will not have a properly initialized primary key. This can cause problems if the `create` method is called within a long-running transaction. This is not an issue for applications that use table-based sequencing or for databases that support sequence pre-allocation. Sybase and SQL Server do not support sequence pre-allocation, whereas Oracle does.

2 TopLink CMP for IBM WebSphere Server

Oracle9iAS TopLink CMP for IBM WebSphere Server is an extension of the TopLink for Java Foundation Library product and shares all of that product's core functionality. TopLink CMP for IBM WebSphere Server provides container-managed persistence (CMP) for Enterprise JavaBeans (EJBs) deployed in the IBM WebSphere Application Server Version 4.0. It also supports development with WebSphere Studio Application Developer. In addition, TopLink for WebSphere Server Foundation Library provides all of the object-relational persistence facilities, as documented in the [Oracle9iAS TopLink Foundation Library Guide](#).

TopLink CMP for IBM WebSphere Server's support allows you to have complex mappings from entity beans to relational database tables, and to model relationships between entity beans and between beans and regular Java objects. The TopLink for Java Foundation Library provides a rich set of querying options, and allows queries to be defined at the bean-level rather than at the database level.

Oracle9iAS TopLink CMP for IBM WebSphere Server supports the EJB 1.1 specification as defined by Sun Microsystems. Knowledge of the EJB specification, WebSphere Application Server 4.0, and optionally IBM WebSphere Studio Application Developer 4.0 is assumed.

2.1 Installation Notes

TopLink JARs must be placed in the WebSphere Applications Extensions (AEX) classpath and the server restarted. The jars to be copied to `<WebSphere_Install>\lib\app` are:

- `<INSTALL_DIR>\core\lib\toplink.jar`
- `<INSTALL_DIR>\core\lib\antlr.jar`
- `<INSTALL_DIR>\core\lib\xerces.jar`
- `<INSTALL_DIR>\was_cmp\lib\tl_wasx.jar`

- `<INSTALL_DIR>\hsq1\lib\hsqldb.jar` (required to run the examples only)
- `<INSTALL_DIR>\hsq1\lib\hsq1_ds.jar` (required to run the examples only)

For more information, see the [Oracle9iAS TopLink CMP for IBM WebSphere Server Guide](#).

2.2 Examples

To familiarize yourself with the CMP functionality in TopLink CMP for IBM WebSphere Server, we recommend you explore the [TopLink examples](#). The Single bean example is a simple example that demonstrates some of the setup issues such as setting up the `toplink-ejb-jar.xml` file, licensing, and **CLASSPATH**.

2.3 Known Limitations

- When installing a new version of an EJB module, the Application Server into which the EJB module is installed must be shutdown and restarted in order to clear the TopLink state associated with the previous version of the EJB module.
- `OptimisticLockExceptions` thrown by TopLink are masked by the server. Instead the client will receive a `TransactionRolledbackException` or a `RollbackException` depending on whether the transaction was begun manually or not.