

**Not For Publication**

# **Sun Java System Application Server Platform Edition 9 Developer's Guide**



Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

Part No: 819-3659-11

Copyright 2006 Sun Microsystems, Inc.    4150 Network Circle, Santa Clara, CA 95054 U.S.A.    All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

---

Copyright 2006 Sun Microsystems, Inc.    4150 Network Circle, Santa Clara, CA 95054 U.S.A.    Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

(    P A R T I  
Development Tasks and Tools



P A R T I I

Developing Applications and Application  
Components



  
  
CHAPTER 7

# Using the Java Persistence API

---

Sun Java System Application Server support for the Java Persistence API includes all required features described in the Java Persistence Specification. Although officially part of the Enterprise JavaBeans Specification v3.0, also known as JSR 220 (<http://jcp.org/en/jsr/detail?id=220>), the Java Persistence API can also be used with non-EJB components outside the EJB container.

The Java Persistence API provides an object/relational mapping facility to Java developers for managing relational data in Java applications. For basic information about the Java Persistence API, see “Part Four: Persistence” in the [Java EE 5 Tutorial](http://java.sun.com/javaee/5/docs/tutorial/doc/index.html) (<http://java.sun.com/javaee/5/docs/tutorial/doc/index.html>).

This chapter contains Application Server specific information on using the Java Persistence API in the following topics:

- [“Specifying the Database” on page 8](#)
- [“Database Properties” on page 10](#)
- [“Configuring the Cache” on page 10](#)
- [“Setting the Logging Level” on page 10](#)
- [“Automatic Schema Generation” on page 11](#)
- [“Query Hints” on page 16](#)
- [“Changing the Persistence Provider” on page 16](#)
- [“Database Restrictions and Optimizations” on page 17](#)

---

**Note** – The default persistence provider in the Application Server is based on Oracle’s TopLink Essentials Java Persistence API implementation. All configuration options in TopLink Essentials are available to applications that use the Application Server’s default persistence provider.

---

## Specifying the Database

The Application Server uses the bundled Java DB (Derby) database by default. If the `transaction-type` element is omitted or specified as `JTA` and both the `jta-data-source` and `non-jta-data-source` elements are omitted in the `persistence.xml` file, Java DB is used as a JTA data source. If `transaction-type` is specified as `RESOURCE_LOCAL` and both `jta-data-source` and `non-jta-data-source` are omitted, Java DB is used as a non-JTA data source.

To use a non-default database, either specify a value for the `jta-data-source` element, or set the `transaction-type` element to `RESOURCE_LOCAL` and specify a value for the `non-jta-data-source` element.

If you are using the [default](#) persistence provider, the Application Server attempts to automatically detect the database based on the connection. You can specify the `toplink.platform.class.name` property to guarantee that the database is correct. For example:

```
<?xml version="1.0" encoding="UTF-8"?>
  <persistence xmlns="http://java.sun.com/xml/ns/persistence">
    <persistence-unit name="em1">
      <jta-data-source>jdbc/MyDB2DB</jta-data-source>
      <properties>
        <property name="toplink.platform.class.name"
          value="oracle.toplink.essentials.platform.database.DB2Platform"/>
      </properties>
    </persistence-unit>
  </persistence>
```

The following `toplink.platform.class.name` property values are allowed:

```
//Supported platforms
oracle.toplink.essentials.platform.DerbyPlatform
oracle.toplink.essentials.platform.oracle.OraclePlatform
oracle.toplink.essentials.platform.SQLServerPlatform
oracle.toplink.essentials.platform.DB2Platform
oracle.toplink.essentials.platform.SybasePlatform
oracle.toplink.essentials.platform.CloudscapePlatform
oracle.toplink.essentials.platform.MySQL4Platform
oracle.toplink.essentials.platform.PointBasePlatform
oracle.toplink.essentials.platform.PotgreSQLPlatform

//Others available
oracle.toplink.essentials.platform.InformixPlatform
oracle.toplink.essentials.platform.TimesTenPlatform
oracle.toplink.essentials.platform.AttunityPlatform
oracle.toplink.essentials.platform.HSQLPlatform
oracle.toplink.essentials.platform.SQLAnywherePlatform
oracle.toplink.essentials.platform.DBasePlatform
oracle.toplink.essentials.platform.DB2MainframePlatform
```

```
oracle.toplink.essentials.platform.AccessPlatform
```

To use the Java Persistence API outside the EJB container ([in Java SE mode](#)), do not specify the `jta-data-source` or `non-jta-data-source` elements [if the DataSource is not available](#). Instead, specify the provider element and any additional properties required by the JDBC driver or the database. If you are using the [default](#) persistence provider, specify the `toplink.platform.class.name` property and the `toplink.jdbc.*` properties. For example:

```
<?xml version="1.0" encoding="UTF-8"?>
  <persistence xmlns="http://java.sun.com/xml/ns/persistence">
    <persistence-unit name="em2">
      <provider>oracle.toplink.essentials.ejb.cmp3.EntityManagerFactoryProvider</provider>
      <transaction-type>RESOURCE_LOCAL</transaction-type>
      <non-jta-data-source>jdbc/MyDB2DB</non-jta-data-source>
      <properties>
        <property name="toplink.platform.class.name"
          value="oracle.toplink.essentials.platform.database.DB2Platform"/>
        <!-- JDBC connection properties -->
        <property name="toplink.jdbc.driver" value="org.apache.derby.jdbc.ClientDriver"/>
        <property name="toplink.jdbc.url"
value="jdbc:derby://localhost:1527/testdb;retrieveMessagesFromServerOnGetMessage=true;create=true;"/>
        <property name="toplink.jdbc.user" value="APP"/>
        <property name="toplink.jdbc.password" value="APP"/>
      </properties>
    </persistence-unit>
  </persistence>
```

For more information about `toplink` properties, see [“Database Properties” on page 10](#).

For a list of the JDBC drivers currently supported by the Application Server, see the *Sun Java System Application Server Platform Edition 9 Release Notes*. For configurations of supported and other drivers, see [“Configurations for Specific JDBC Drivers”](#) in *Sun Java System Application Server Platform Edition 9 Administration Guide*.

To change the persistence provider, see [“Changing the Persistence Provider” on page 16](#).

---

**Note** – To use BLOB or CLOB data types larger than 4 KB for persistence using the Inet Oraxo JDBC Driver for Oracle Databases, you must set the database’s `streamsToLob` property value to `true`.

---

## Database Properties

If you are using the default persistence provider, you can specify in the `persistence.xml` file the database properties listed at [Persistence Unit Extensions \(http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#persistence-xml\)](http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#persistence-xml).

For schema generation properties, see “Generation Options” on page 12. For query hints, see “Query Hints” on page 16.

## Configuring the Cache

You can configure whether caching occurs, the type of caching, the size of the cache, and whether client sessions share the cache. Caching properties for the default persistence provider are described in detail at [Extensions for Caching \(http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#TopLinkCaching\)](http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#TopLinkCaching).

## Setting the Logging Level

One of the default persistence provider’s database properties that you can set in the `persistence.xml` file is `toplink.logging.level`. For example, setting the logging level to `FINE` or higher logs all SQL statements. For details about this property, see [Extensions for Logging \(http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#TopLinkLogging\)](http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#TopLinkLogging).

You can also set the logging level globally in the Application Server in any of the following ways:

- Set a `module-log-levels` property using the `asadmin` command. For example:

```
asadmin set --user adminuser "server.log-service.module-log-levels.property.oracle\toplink\essentials"=FINEST
```

- Set a JVM option using the `asadmin` command. For example:

```
asadmin create-jvm-options --user adminuser -Dtoplink.logging.level=FINE
```

- Set a `module-log-levels` property using the Admin Console. Select the Application Server component, the Logging tab, and the Log Levels sub-tab. Scroll down to Additional Module Log Level Properties, select Add Property, type `oracle.toplink.essentials` in the Name field, and type the desired logging level in the Value field.

A logging level set in the `persistence.xml` file takes precedence over the global logging level.

# Automatic Schema Generation

The automatic schema generation feature of the Application Server defines database tables based on the fields or properties in entities and the relationships between the fields or properties. This insulates developers from many of the database related aspects of development, allowing them to focus on entity development. The resulting schema is usable as-is or can be given to a database administrator for tuning with respect to performance, security, and so on. This section covers the following topics:

- “Annotations” on page 11
- “Supported Data Types” on page 11
- “Generation Options” on page 12

---

**Note** – Automatic schema generation is supported on an all-or-none basis: it expects that no tables exist in the database before it is executed. It is not intended to be used as a tool to generate extra tables or constraints.

Deployment won't fail if all tables are not created, and undeployment won't fail if not all tables are dropped. This is done to allow you to investigate the problem and fix it manually. You should not rely on the partially created database schema to be correct for running the application.

---

## Annotations

The following annotations are used in automatic schema generation: `@AssociationOverride`, `@AssociationOverrides`, `@AttributeOverride`, `@AttributeOverrides`, `@Column`, `@DiscriminatorColumn`, `@DiscriminatorValue`, `@Embedded`, `@EmbeddedId`, `@GeneratedValue`, `@Id`, `@IdClass`, `@JoinColumn`, `@JoinColumns`, `@JoinTable`, `@Lob`, `@ManyToMany`, `@ManyToOne`, `@OneToMany`, `@OneToOne`, `@PrimaryKeyJoinColumn`, `@PrimaryKeyJoinColumns`, `@SecondaryTable`, `@SecondaryTables`, `@SequenceGenerator`, `@Table`, `@TableGenerator`, `@UniqueConstraint`, and `@Version`. For information about these annotations, see the Java Persistence Specification.

For `@Column` annotations, the `insertable` and `updatable` elements are not used in automatic schema generation.

For `@OneToMany` and `@ManyToOne` annotations, no `ForeignKeyConstraint` is created in the resulting DDL files.

## Supported Data Types

The following table shows mappings of Java types to SQL types when the default persistence provider and automatic schema generation are used.

TABLE 7-1 Java Type to SQL Type Mappings

Java Type	Java DB, Derby, CloudScape	Oracle	DB2	Sybase	MS-SQL Server	MySQL Server
boolean, java.lang.Boolean	SMALLINT	NUMBER(1)	SMALLINT	BIT	BIT	TINYINT(1)
int, java.lang.Integer	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
long, java.lang.Long	BIGINT	NUMBER(19)	INTEGER	NUMERIC(19)	NUMERIC(19)	BIGINT
float, java.lang.Float	FLOAT	NUMBER(19,4)	FLOAT	FLOAT(16)	FLOAT(16)	FLOAT
double, java.lang.Double	FLOAT	NUMBER(19,4)	FLOAT	FLOAT(32)	FLOAT(32)	DOUBLE
short, java.lang.Short	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
byte, java.lang.Byte	SMALLINT	NUMBER(3)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
java.lang.Number	DECIMAL	NUMBER(38)	DECIMAL(15)	NUMERIC(38)	NUMERIC(28)	DECIMAL(38)
java.math.BigInteger	BIGINT	NUMBER(38)	BIGINT	NUMERIC(38)	NUMERIC(28)	BIGINT
java.math.BigDecimal	DECIMAL	NUMBER(38)	DECIMAL(15)	NUMERIC(38)	NUMERIC(28)	DECIMAL(38)
java.lang.String	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
char, java.lang.Character	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)
byte[], java.lang.Byte[], java.sql.Blob	BLOB(64000)	LONG RAW	BLOB(64000)	IMAGE	IMAGE	BLOB(64000)
char[], java.lang.Character[], java.sql.Clob	CLOB(64000)	LONG	CLOB(64000)	TEXT	TEXT	TEXT(64000)
java.sql.Date	DATE	DATE	DATE	DATETIME	DATETIME	DATE
java.sql.Time	TIME	DATE	TIME	DATETIME	DATETIME	TIME
java.sql.Timestamp	TIMESTAMP	DATE	TIMESTAMP	DATETIME	DATETIME	DATETIME

## Generation Options

Schema generation properties or `asadmin` command line options can control automatic schema generation by the following:

- Creating tables during deployment
- Dropping tables during undeployment
- Dropping and creating tables during redeployment
- Generating the DDL files

---

**Note** – Before using these options, make sure you have a properly configured database. See “Specifying the Database” on page 8.

---

The following optional schema generation properties control the automatic creation of database tables at deployment. You can specify them in the `persistence.xml` file.

TABLE 7-2 Schema Generation Properties

Property	Default	Description
<code>toplink.ddl-generation</code>	none	Specifies whether tables and DDL files are created during deployment, and whether tables are dropped first if they already exist. Allowed values are <code>create-tables</code> , <code>drop-and-create-tables</code> , and <code>none</code> .  If you are using persistence outside the EJB container and would like to create the DDL files without creating tables, additionally define a Java system property <code>INTERACT_WITH_DB</code> and set its value to <code>false</code> .
<code>toplink.create-ddl-jdbc-file-name</code>	<code>createDDL.jdbc</code>	Specifies the name of the JDBC file that contains the DDL statements required to create the required objects (tables, sequences, and constraints) in the database.
<code>toplink.drop-ddl-jdbc-file-name</code>	<code>dropDDL.jdbc</code>	Specifies the name of the JDBC file that contains the DDL statements required to drop the required objects (tables, sequences, and constraints) from the database.
<code>toplink.application-location</code>	<code>.</code> for the current working directory	For persistence outside the EJB container, specifies the location where the DDL files are written.  For persistence within the EJB container, if this property is set, the value is ignored, and DDL files are written to one of the following locations, for applications and modules, respectively:  <i>domain-dir/generated/ejb/j2ee-apps/app-name</i> <i>domain-dir/generated/ejb/j2ee-modules/mod-name</i>

TABLE 7-2 Schema Generation Properties (Continued)

Property	Default	Description
<code>toplink.ddl-generation.output-mode</code>	both	<p>Specifies the DDL generation target if you are in Java SE mode, outside the EJB container. Values are as follows:</p> <ul style="list-style-type: none"> <li>■ <b>both</b> – Generates SQL files and executes them on the database. If <code>toplink.ddl-generation</code> is set to <code>create-tables</code>, then <code>toplink.create-ddl-jdbc-file-name</code> is written to <code>toplink.application-location</code> and executed on the database. If <code>toplink.ddl-generation</code> is set to <code>drop-and-create-tables</code>, then both <code>toplink.create-ddl-jdbc-file-name</code> and <code>toplink.drop-ddl-jdbc-file-name</code> are written to <code>toplink.application-location</code> and both SQL files are executed on the database.</li> <li>■ <b>database</b> – Executes SQL on the database only (does not generate SQL files). If <code>toplink.ddl-generation</code> is set to <code>create-tables</code>, then <code>toplink.create-ddl-jdbc-file-name</code> is executed on the database. It is not written to <code>toplink.application-location</code>. If <code>toplink.ddl-generation</code> is set to <code>drop-and-create-tables</code>, then both <code>toplink.create-ddl-jdbc-file-name</code> and <code>toplink.drop-ddl-jdbc-file-name</code> are executed on the database. Neither is written to <code>toplink.application-location</code>.</li> <li>■ <b>sql-script</b> – Generates SQL files only (does not execute them on the database). If <code>toplink.ddl-generation</code> is set to <code>create-tables</code>, then <code>toplink.create-ddl-jdbc-file-name</code> is written to <code>toplink.application-location</code>. It is not executed on the database. If <code>toplink.ddl-generation</code> is set to <code>drop-and-create-tables</code>, then both <code>toplink.create-ddl-jdbc-file-name</code> and <code>toplink.drop-ddl-jdbc-file-name</code> are written to <code>toplink.application-location</code>. Neither is executed on the database.</li> </ul>

The following options of the `asadmin deploy` or `asadmin deploydir` command control the automatic creation of database tables at deployment.

TABLE 7-3 The `asadmin deploy` and `asadmin deploydir` Generation Options

Option	Default	Description
<code>--createtables</code>	none	If <code>true</code> , causes database tables to be created for entities that need them. If <code>false</code> , does not create tables. If not specified, the value of the <code>toplink.ddl-generation</code> property in <code>persistence.xml</code> is used.

TABLE 7-3 The `asadmin deploy` and `asadmin deploydir` Generation Options    (Continued)

Option	Default	Description
<code>--dropandcreatetables</code>	none	<p>If <code>true</code>, and if tables were automatically created when this application was last deployed, tables from the earlier deployment are dropped and fresh ones are created.</p> <p>If <code>true</code>, and if tables were <i>not</i> automatically created when this application was last deployed, no attempt is made to drop any tables. If tables with the same names as those that would have been automatically created are found, the deployment proceeds, but a warning is thrown to indicate that tables could not be created.</p> <p>If <code>false</code>, the <code>toplink.ddl-generation</code> property setting in <code>persistence.xml</code> is overridden.</p>

The following options of the `asadmin undeploy` command control the automatic removal of database tables at undeployment.

TABLE 7-4 The `asadmin undeploy` Generation Options

Option	Default	Description
<code>--droptables</code>	none	<p>If <code>true</code>, causes database tables that were automatically created when the entities were last deployed to be dropped when the entities are undeployed. If <code>false</code>, does not drop tables.</p> <p>If not specified, tables are dropped only if the <code>toplink.ddl-generation</code> property setting in <code>persistence.xml</code> is <code>drop-and-create-tables</code>.</p>

For more information about the `asadmin deploy`, `asadmin deploydir`, and `asadmin undeploy` commands, see the *Sun Java System Application Server Platform Edition 9 Reference Manual*.

If one or more entities in the persistence unit are mapped to existing database tables and you use any of the `asadmin deploy` or `asadmin deploydir` options, the `asadmin` deployment options take precedence.

When `asadmin` deployment options and `persistence.xml` options are both specified, the `asadmin` deployment options take precedence.

The `asant` tasks `sun-appserv-deploy` and `sun-appserv-undeploy` are equivalent to `asadmin deploy` and `asadmin undeploy`, respectively. These `asant` tasks also override the `persistence.xml` options. For details, see [Chapter 3](#).

## Query Hints

Query hints are additional, implementation-specific configuration settings. You can use hints in your queries in the following format:

```
setHint("hint-name", hint-value)
```

For example:

```
Customer customer = (Customer)entityMgr.  
    createNamedQuery("findCustomerBySSN").  
    setParameter("SSN", "123-12-1234").  
    setHint("toplink.refresh", true).  
    getSingleResult();
```

For more information about the query hints available with the default provider, see [Query Hints](http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#QueryHints) (<http://www.oracle.com/technology/products/ias/toplink/jpa/resources/toplink-jpa-extensions.html#QueryHints>).

## Changing the Persistence Provider

---

**Note** – The previous sections in this chapter apply only to the default persistence provider. If you change the provider for a module or application, the provider-specific database properties, query hints, and schema generation features described in this chapter do not apply.

The verifier utility always uses the default provider to verify persistence settings. For information about the verifier utility, see “The verifier Utility” in *Sun Java System Application Server Platform Edition 9 Application Deployment Guide*.

---

You can change the persistence provider for an application in the manner described in the [Java Persistence API Specification](#).

First, install the provider. Copy the provider JAR files to the *domain-dir/lib* directory, and restart the Application Server. For more information about the *domain-dir/lib* directory, see “[Using the Common Class Loader](#)” on page \_\_\_\_\_. The new persistence provider is now available to all modules and applications in the domain. However, the *default* provider remains the same.

In your persistence unit, specify the provider and any properties the provider requires in the *persistence.xml* file. For example:

```
<?xml version="1.0" encoding="UTF-8"?>  
  <persistence xmlns="http://java.sun.com/xml/ns/persistence">  
    <persistence-unit name="em3">  
      <provider>com.company22.persistence.PersistenceProviderImpl</provider>  
      <properties>
```

```
<property name="company22.database.name" value="MyDB"/>
</properties>
</persistence-unit>
</persistence>
```

## Database Restrictions and Optimizations

This section discusses restrictions and performance optimizations that affect using the Java Persistence API.

- [“Case Insensitivity of Field Names” on page 17](#)
- [“Using a SERIAL Field in PostgreSQL” on page 17](#)
- [“Sybase Finder Limitation” on page 17](#)
- [“MySQL Database Restrictions” on page 18](#)

### Case Insensitivity of Field Names

In the Application Server, schema generation converts all field names to uppercase. Therefore, if you are using a case sensitive database, make sure all field names in the code and the database are in uppercase.

### Using a SERIAL Field in PostgreSQL

To use a SERIAL field in a PostgreSQL database, define the following in your entity class:

```
@Id
@GeneratedValue(strategy=GenerationType.IDENTITY)
```

The field created is of type SERIAL, and the implicit native sequence associated with this field is of the form *FullyQualifiedName\_IdFieldColumnName\_SEQ*.

### Sybase Finder Limitation

If a finder method with an input greater than 255 characters is executed and the primary key column is mapped to a VARCHAR column, Sybase attempts to convert type VARCHAR to type TEXT and generates the following error:

```
com.sybase.jdbc2.jdbc.SybSQLException: Implicit conversion from datatype
'TEXT' to 'VARCHAR' is not allowed. Use the CONVERT function to run this
query.
```

To avoid this error, make sure the finder method input is less than 255 characters.

## MySQL Database Restrictions

The following restrictions apply when you use a MySQL database with the Application Server for persistence.

- MySQL treats `int1` and `int2` as reserved words. If you want to define `int1` and `int2` as fields in your table, use `'int1'` and `'int2'` field names in your SQL file.
- When `VARCHAR` fields get truncated, a warning is displayed instead of an error. To get an error message, start the MySQL database in strict SQL mode.
- The order of fields in a foreign key index must match the order in the explicitly created index on the primary table.
- The `CREATE TABLE` syntax in the SQL file must end with the following line.

```
) Engine=InnoDB;
```

InnoDB provides MySQL with a transaction-safe (ACID compliant) storage engine having commit, rollback, and crash recovery capabilities.

- For a `FLOAT` type field, the correct precision must be defined. By default, MySQL uses four bytes to store a `FLOAT` type that does not have an explicit precision definition. For example, this causes a number such as 12345.67890123 to be rounded off to 12345.7 during an `INSERT`. To prevent this, specify `FLOAT(10,2)` in the DDL file, which forces the database to use an eight-byte double-precision column. For more information, see <http://dev.mysql.com/doc/mysql/en/numeric-types.html>.
- To use `||` as the string concatenation symbol, start the MySQL server with the `--sql-mode="PIPES_AS_CONCAT"` option. For more information, see <http://dev.mysql.com/doc/refman/5.0/en/server-sql-mode.html> and <http://dev.mysql.com/doc/mysql/en/ansi-mode.html>.
- MySQL always starts a new connection when `autoCommit=true` is set. This ensures that each SQL statement forms a single transaction on its own. If you try to rollback or commit an SQL statement, you get an error message.

```
javax.transaction.SystemException: java.sql.SQLException:
Can't call rollback when autocommit=true
```

```
javax.transaction.SystemException: java.sql.SQLException:
Error open transaction is not closed
```

To resolve this issue, add `relaxAutoCommit=true` to the JDBC URL. For more information, see <http://forums.mysql.com/read.php?39,31326,31404>.

- MySQL does not allow a `DELETE` on a row that contains a reference to itself. Here is an example that illustrates the issue.

```
create table EMPLOYEE (
    empId int NOT NULL,
    salary float(25,2) NULL,
```

```
mgrId int NULL,  
PRIMARY KEY (empId),  
FOREIGN KEY (mgrId) REFERENCES EMPLOYEE (empId)  
) ENGINE=InnoDB;  
  
insert into Employee values (1, 1234.34, 1);  
delete from Employee where empId = 1;
```

This example fails with the following error message.

```
ERROR 1217 (23000): Cannot delete or update a parent row:  
a foreign key constraint fails
```

To resolve this issue, change the table creation script to the following:

```
create table EMPLOYEE (  
  empId int NOT NULL,  
  salary float(25,2) NULL,  
  mgrId int NULL,  
  PRIMARY KEY (empId),  
  FOREIGN KEY (mgrId) REFERENCES EMPLOYEE (empId)  
  ON DELETE SET NULL  
) ENGINE=InnoDB;  
  
insert into Employee values (1, 1234.34, 1);  
delete from Employee where empId = 1;
```

This can be done only if the foreign key field is allowed to be null. For more information, see <http://bugs.mysql.com/bug.php?id=12449> and <http://dev.mysql.com/doc/mysql/en/innodb-foreign-key-constraints.html>.



PART III

Using Services and APIs

