

Transitioning to Oracle Universal Connection Pool (UCP)

Starting with Oracle Database 11g Release 11.1.0.7, we have released the new Oracle Universal Connection Pool for JDBC. For more details, see UCP Users Guide @

http://www.oracle.com/technology/tech/java/sqlj_jdbc/UCP_dev_guide.pdf

The transition affects applications that use

- (i) `OracleDataSource` and
- (ii) `OracleConnectionCacheManager` in the `oracle.jdbc.pool` package.

UCP Code samples are provided on the download site below to illustrate the changes.

http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/ucp.html

The following instructions will help you replace Implicit Connection Cache APIs with UCP APIs:

1. Since UCP is not colocated with the JDBC driver, applications should include **ucp.jar** in their classpath.
2. Packages: A selective set of APIs related to Implicit Connection Cache has changed. Applications switching to UCP should use the new UCP packages instead.

Replace:

```
import oracle.jdbc.pool.*;
```

With:

```
import oracle.ucp.jdbc.*;
import oracle.ucp.admin.*;
```

3. `OracleDataSource` -- cache/pool creation:

Replace:

```
OracleDataSource ods = new OracleDataSource();
ods.setConnectionCachingEnabled(true);
ods.setConnectionCacheName("ICC_Name");
```

With:

```
PoolDataSource pds = PoolDataSourceFactory.getPoolDataSource();
pds.setConnectionFactoryClassName("oracle.jdbc.pool.OracleDataSource");
pds.setConnectionPoolName("UCP_Name");
```

Other data source properties like user, password, url, etc. should be set appropriately. See UCP sample code `JDBCConnectionPoolAccess.java`.

4. `OracleDataSource` -- setting cache/pool properties:

Replace:

```
java.util.Properties props = new java.util.Properties ();
props.setProperty("MinLimit", "20");
props.setProperty("MaxLimit", "100");
props.setProperty("InitialLimit", "50");
props.setProperty("InactivityTimeout", "60");
// similarly for other properties
ods.setConnectionCacheProperties(props);
```

With:

```
pds.setMinPoolSize(20);
pds.setMaxPoolSize(100);
pds.setInitialPoolSize(50);
pds.setInactiveConnectionTimeout(60);
// similarly for other properties
See UCP sample code ConnectionPoolTimeoutProperties.java.
```

5. OracleDataSource -- Fast Connection Failover and RCLB:

Replace:

```
// The following line is necessary when using remote-ONS subscription
ods.setONSConfiguration("<REMOTE_ONS_CONFIGURATION_STRING>");
ods.setFastConnectionFailoverEnabled(true);
```

With:

```
// The following line is necessary when using remote-ONS subscription
pds.setONSConfiguration("<REMOTE_ONS_CONFIGURATION_STRING>");
pds.setFastConnectionFailoverEnabled(true);
See UCP sample code FCFSample.java.
```

6. OracleDataSource -- connection attributes/labeling:

See ICC sample code ImplicitCache04.java.
See UCP sample code ConnectionLabelingSample.java.

7. OracleConnectionCacheManager -- cache/pool administration and statistics:

Replace:

```
OracleConnectionCacheManager occm =
    OracleConnectionCacheManager.getConnectionCacheManagerInstance();
int usedCount = occm.getNumberOfActiveConnections("ICC_Name");
int availableCount = occm.getNumberOfAvailableConnections("ICC_Name");
```

With:

```
// This is in oracle.ucp.admin package
UniversalConnectionPoolManager mgr =
    UniversalConnectionPoolManagerImpl.getUniversalConnectionPoolManager();
UniversalConnectionPool pool = mgr.getConnectionPool("UCP_Name");
UniversalConnectionPoolStatistics stats = pool.getStatistics();
int usedCount = stats.getBorrowedConnectionsCount("ICC_Name");
int availableCount = stats.getAvailableConnectionsCount("ICC_Name");
```

See UCP sample code UniversalConnectionPoolManagerSample.java.