

Database Connection Pooling with JDBC

What is JDBC Connection Pooling?

JDBC Connection Pooling allows an application to establish a number of database connections and re-use them without requiring subsequent logins.

Connection Pools relieve the database of managing logins and logouts which are very expensive operations.

Typically, Developers enable Connection Pools, not DBA's. Developers and DBA's should coordinate.

Why is Connection Pooling Important?

Oracle is a multi-process application so the OS must reserve memory and create a process for each database login.

For every login, there “should be” a logout. Each of these operations takes time.

A Connection Pool can be used to control the number of simultaneous connections to the database.

How is Connection Pooling Enabled?

- WebLogic - Delivered with Oracle's Connection Pooling mechanism**
- Tomcat - Delivered with Apache's DBCP product**
- To enable Connection Pooling, an application may need little or no modification.**

How is Connection Pooling Enabled?

```
<Context path="/doug"  
    docBase="doug"  
    reloadable="true"  
    crossContext="true">
```

```
<Resource name="jdbc/myoracle"  
    auth="Container"  
    type="javax.sql.DataSource"  
    driverClassName="oracle.jdbc.OracleDriver"  
    url="jdbc:oracle:thin:@localhost:1521:xe"  
    username="user"  
    password="password"  
    maxActive="15"           Max Connections in the Pool  
    maxIdle="15"            Max Connections that can be idle in the pool  
    maxWait="3000"          Max time (ms) to wait for a connection from the pool  
    minIdle="5"             Min Connections in the pool  
    initialSize="5"/>      Number of connections to establish at startup
```

```
</Context>
```

So it's Easy, Right? Not Exactly.

Not all Connection Pools act as expected:

- **Initial values are not always evaluated**
- **When a pool is overloaded, the Pool does not always restart**
- **Some Connection Pools allow for a “validation query” to execute before every real query (huge time and resource waste)**

In general, Oracle's Pool is more consistent than DBCP

Demo

.Tomcat
.WebLogic

What are Connection Pooling Best Practices?

- **Use Connection Pooling whenever possible**
- **Understand how many connections your application really needs**
- **Turn off (or tune) Connection Validation**
- **Set the Min and Max Connection values to the same number**
- **Use a commercial product if possible**
- **TEST, TEST, TEST – don't assume its working**

Summary and Q&A

- **Contact - brian.hill@enkitec.com**
- **blog.enkitec.com – will post an entry with the slides**