

Configuring DataSources in Tests

InitialContext properties

You can configure data sources from within your test case (avoiding the need for an `openejb.xml` entirely) like so:

```
Properties p = new Properties();
p.put(Context.INITIAL_CONTEXT_FACTORY, "org.apache.openejb.client.LocalInitialContextFactory");

p.put("myDataSource", "new://Resource?type=DataSource");
p.put("myDataSource.JdbcDriver", "org.apache.derby.jdbc.EmbeddedDriver");
p.put("myDataSource.JdbcUrl", "jdbc:derby:derbyDB;create=true");
p.put("myDataSource.JtaManaged", "true");

Context context = new InitialContext(p);
```

See [Embedded Configuration](#) for details on properties and overrides.

See [Containers and Resources](#) for a full list of supported Resource types and their properties.

Note on <jta-data-source> and <non-jta-data-source>

When configuring DataSources to be used by `persistence.xml` files, the DataSource supplied for `<jta-data-source>` is typically identical to the `<non-jta-data-source>`, but with the `JtaManaged` property set differently. Keeping with our philosophy to free you up from redundant configuration, we will happily auto-create a missing `jta-data-source` or `non-jta-data-source` based upon the supplied DataSource.

In the example above, a new DataSource would be generated as an exact copy but with the name `"myDataSourceUnmanaged"` and its `JtaManaged` flag set to `false`. If the supplied DataSource was not `JtaManaged`, then the generated DataSource would be called `"myDataSourceJta"` and have its `JtaManaged` flag set to `true`.

When relying on this functionality it is not necessary to specify the name of the generated DataSource in the `persistence.xml` file.