

## Transaction Service

### Summary

This service adopts and guides Spring's transaction service. There are several transaction services, but here, **DataSource Transaction Service**, **JTA Transaction Service** and **JPA Transaction Service** will be explained. In addition, we'll explain **Declaration Transaction Management** that can be utilized through setting and annotation for transaction utilization as well as **Programmatic Transaction Management** that can be written by directly calling API from program.

- [Declarative Transaction Management](#)
- [Programmatic Transaction Management](#)

### Description

#### DataSource Transaction Service

This can manage Local Transaction using DataSource. We'll explain how to set and use in the following example.

#### Configuration

```
<bean id="transactionManager"
class="org.springframework.jdbc.datasource.DataSourceTransactionManager">
<property name="dataSource" ref="dataSource"/>
</bean>
```

```
<bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">
<property name="driverClassName" value="com.mysql.jdbc.Driver"/>
<property name="url" value="dbc:mysql://db2:1621/rte"/>
<property name="username" value="rte"/>
<property name="password" value="xxx"/>
<property name="defaultAutoCommit" value="false"/>
</bean>
```

PROPERTIES	Description
driverClassName	jdbc driver
url	dburl
username	User name
password	Password
defaultAutoCommit	Auto commit setting

From above setting, dataSource was designated with the property of transactionManager and driver information, Url information required for this were designated. Provide transaction service based on the set dataSource. Apply after changing driverClassName, url, username and password according to site environment.

#### Sample Source

```
@Resource(name="transactionManager")
PlatformTransactionManager transactionManager;
...
TransactionStatus txStatus = transactionManager.getTransaction(txDefinition);
```

transactionManager can be utilized as shown above.

#### JTA Transaction Service

It is to support for Global Transaction management using JTA. For example, explain the setting method below. How to use is same as that of DataSource Transaction Service.

## Configuration

```
<tx:jta-transaction-manager/>
<jee:jndi-lookup id="dataSource" jndi-name="dbmsXADS" resource-ref="true">
<jee:environment>
    java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory
    java.naming.provider.url=t3://was:7002
</jee:environment>
</jee:jndi-lookup>
```

In above setting example, jndi-name and java.naming.factory.initial,java.naming.provider.url should be changed according to the environment of the site. It is not required to define bean separately for transactionManager, different from DataSource Transaction Service.

## JPA Transaction Service

The JPA Transaction service manages transaction using JPA EntityManagerFactory. Since JpaTransactionManager has dependence on EntityManagerFactory, it should be defined same as the EntityManagerFactory setting. Explain the setting method in the following example. How to use is same as DataSource Transaction Service.

## Configuration

```
<bean id="transactionManager" class="org.springframework.orm.jpa.JpaTransactionManager">
<property name="entityManagerFactory" ref="entityManagerFactory"/>
</bean>

<bean id="entityManagerFactory"
class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">
<property name="persistenceUnitName" value="OraUnit"/>
<property name="persistenceXmlLocation" value="classpath:META-INF/persistence.xml"/>
<property name="dataSource" ref="dataSource"/>
</bean>

<bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">
<property name="driverClassName" value="com.mysql.jdbc.Driver"/>
<property name="url" value="dbc:mysql://db2:1621/rte"/>
<property name="username" value="rte"/>
<property name="password" value="xxx"/>
<property name="defaultAutoCommit" value="false"/>
</bean>
```

PROPERTIES	Description
driverClassName	jdbc driver
url	dburl
username	User Name
password	Password
defaultAutoCommit	Auto commit setting
persistenceUnitName	persistenceUnitName
persistenceXmlLocation	XML location
dataSource	Data source

From above setting, it can be confirmed that entiyManagerFactory was designated with property of transactionManager; dataSource designated with property of entityManagerFactory and driver information,Url information required for this were designated. Provide transaction service under the dataSourcefoundation.

Apply after changing driverClassName,url,username,password according to site environment. In addition, persistenceUnitName andpersistenceXmlLocationinformation can be designated.

## Reference