

Beispiele zu Transaktionen:

Beispiel zur Verwendung der @TransactionAttribute Annotation und eines XML assembly-descriptors:

```
@Stateless
@TransactionAttribute( TransactionAttributeType.REQUIRES_NEW )
public class ServerSideOperatonTestBean implements ServerSideOperatonTestRemote {
    [...]
}

<assembly-descriptor>
  <container-descriptor>
    <trans-attribute>REQUIRES_NEW
  </trans-attribute>
  <method>
    <ejb-name>ServerSideOperatonTestBean</ejb-name>
    <method-name>addItem</method-name>
  </method>
  <method> ... </method>
</container-descriptor>
</assembly-descriptor>
```

Beispiel zu CMT unter Verwendung der @TransactionAttribute Annotation mit einer SessionBean:

Stateless:

```
@Stateless
@TransactionAttribute( TransactionAttributeType.REQUIRES_NEW )
public class ServerSideOperatonTestBean implements ServerSideOperatonTestRemote {
    [...]
}
```

Stateful:

```
@Stateful
public class ServerSideOperatonTestBean implements ServerSideOperatonTestRemote {
    [...]

    @TransactionAttribute( TransactionAttributeType.REQUIRES_SUPPORTS )
    public void createCustomer(Customer customer) {
        [...]
    }
}
```

Beispiel einer Persistence Unit, die in der persistence.xml mit der jta-data-source definiert wird:

Da die Business Logik in einer Stateless Session Bean implementiert ist, wird das persistence.xml file im /META-INF Verzeichnis der gepackten EJB JAR-Datei enthalten sein. CMT verwendet JTA Transaktionen:

```
<persistence>
  <persistence-unit name="CustomerService">
    <jta-data-source>jdbc/CustomerDS</jta-data-source>
  </persistence-unit>
</persistence>
```

Hier ein Beispiel eines Methodenaufrufs einer transaktionalen Methode innerhalb derselben SFSB:

```
@Stateful
public class ServerSideOperatonTestBean implements ServerSideOperatonTestRemote {
    [...]

    @TransactionAttribute( TransactionAttributeType.REQUIRED )
    public void firstMethod() {
        secondMethod;           // call from inside the same SFSB does NOT start
    }                           // a new transaction

    @TransactionAttribute( TransactionAttributeType.REQUIRES_NEW )
    public void secondMethod() {
        [...]
    }
}
```

Hier ein typisches Beispiel zu BMT (explizite Methoden – Aufrufe in einem try{} catch{} Handler):

```
@Stateful
@TransactionManagement( TransactionManagementType.BEAN )
public class ServerSideOperatonTestBean implements ServerSideOperatonTestRemote {
    @Resource
    SessionContext ejbContext;

    public void methodUsingTransactions() {
        boolean error = false;
        UserTransaction uTx = ejbContext.getUserTransaction();
        try {
            uTx.begin();
            [...]
            if (error) {
                uTx.rollback();
            }
            else {
                uTx.commit();
            }
        } catch (Exception e) {
            [...]
        }
    }
}
```

Hier die Methoden des UserTransaction Interface:

```
public interface javax.transaction.UserTransaction {

    public abstract void begin();
    public abstract void commit();
    public abstract int getStatus();
    public abstract void rollback();
    public abstract void setRollbackOnly();
    public abstract void setTransactionTimeout(int seconds);
}
```