

git_comments:**git_commits:**

1. **summary:** TOMEE-2050 handling txmgr error for ut usage in @Transactional
message: TOMEE-2050 handling txmgr error for ut usage in @Transactional

github_issues:**github_issues_comments:****github_pulls:****github_pulls_comments:****github_pulls_reviews:****jira_issues:**

1. **summary:** JTA 1.2 compliance: Calling UserTransaction methods from @Transactional bean
description: JTA 1.2, section 3.7: {code} If an attempt is made to call any method of the UserTransaction interface from within the scope of a bean or method annotated with @Transactional and a Transactional.TxType other than NOT_SUPPORTED or NEVER, an IllegalStateException must be thrown. {code} Instead TomEE throws: {code} 13:38:03.710 [http-nio-8080-exec-7] ERROR java.lang.Throwable - javax.transaction.NotSupportedException: Nested Transactions are not supported 13:38:03.713 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.geronimo.transaction.manager.TransactionManagerImpl.begin(TransactionManagerImpl.java:157) 13:38:03.716 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.geronimo.transaction.manager.TransactionManagerImpl.begin(TransactionManagerImpl.java:152) 13:38:03.737 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.webbeans.intercept.AbstractInvocationContext.directProceed(AbstractInvocationContext.java:113) 13:38:03.758 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.webbeans.intercept.AbstractInvocationContext.proceed(AbstractInvocationContext.java:106) 13:38:03.761 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.webbeans.intercept.InterceptorInvocationContext.proceed(InterceptorInvocationContext.java:67) 13:38:03.766 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.openejb.cdi.transactional.InterceptorBase.intercept(InterceptorBase.java:67) 13:38:03.770 [http-nio-8080-exec-7] ERROR java.lang.Throwable - at org.apache.openejb.cdi.transactional.RequiredInterceptor.intercept(RequiredInterceptor.java:35) {code}

jira_issues_comments:

1. **body:** Why case do you have, we have a test validating we get {code} java.lang.IllegalStateException: Can't use UserTransaction from @Transaction call {code}
label: test
2. Servlet: {code} @WebServlet("/userTransaction") public final class CdiTransactionaltest extends HttpServlet { @Resource UserTransaction transaction; @Inject TransactionalBean transactionalBean; @Override protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException { final PrintWriter out = resp.getWriter(); resp.setContentType("text/plain"); //other test cases here try { out.println("Testing with UserTransaction inside transactional bean..."); transactionalBean.testWithTxRequiredAndUserTransaction(); } catch (Exception ex) { out.println("Exception: " + ex); ex.printStackTrace(out); } } {code} Transactional bean: {code} @Default public class TransactionalBean { @Resource UserTransaction userTransaction; //other test methods here @Transactional(TxType.REQUIRED) public void testWithTxRequiredAndUserTransaction() throws Exception { userTransaction.begin(); userTransaction.commit(); } } {code} Stack trace from test case: {code} Testing with UserTransaction inside transactional bean... Exception: javax.transaction.NotSupportedException: Nested Transactions are not supported javax.transaction.NotSupportedException: Nested Transactions are not supported at org.apache.geronimo.transaction.manager.TransactionManagerImpl.begin(TransactionManagerImpl.java:157) at org.apache.geronimo.transaction.manager.TransactionManagerImpl.begin(TransactionManagerImpl.java:152) at org.example.TransactionalBean.testWithTxRequiredAndUserTransaction(TransactionalBean.java:43) at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method) at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57) at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) at java.lang.reflect.Method.invoke(Method.java:606) at org.apache.webbeans.intercept.AbstractInvocationContext.directProceed(AbstractInvocationContext.java:113) at org.apache.webbeans.intercept.AbstractInvocationContext.proceed(AbstractInvocationContext.java:106) at org.apache.webbeans.intercept.InterceptorInvocationContext.proceed(InterceptorInvocationContext.java:67) at org.apache.openejb.cdi.transactional.InterceptorBase.intercept(InterceptorBase.java:67) at org.apache.openejb.cdi.transactional.RequiredInterceptor.intercept(RequiredInterceptor.java:35) at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method) at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57) at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) at java.lang.reflect.Method.invoke(Method.java:606) at org.apache.webbeans.component.InterceptorBean.intercept(InterceptorBean.java:136) at org.apache.webbeans.intercept.InterceptorInvocationContext.proceed(InterceptorInvocationContext.java:63) at

org.apache.webbeans.intercept.DefaultInterceptorHandler.invoke(DefaultInterceptorHandler.java:139) at
org.example.TransactionalBean\$\$OwbInterceptProxy0.testWithTxRequiredAndUserTransaction(org/example/TransactionalBean.java)
at org.example.CdiTransactionaltest.doGet(CdiTransactionaltest.java:81) at
javax.servlet.http.HttpServlet.service(HttpServlet.java:622) {code} Attaching zip with sample app (war + source)

3. this exact case is tested and passing

4. Well, but the attached WAR fails with the exception mentioned above when executed on vanilla TomEE. Just drop the ROOT.war in webapps and request <http://localhost:8080/>

5. which is normal since you handle the main tx outside @Tx

6. No, I do not: {code} @WebServlet("/userTransaction") public final class CdiTransactionaltest extends HttpServlet { @Inject TransactionalBean transactionalBean; @Override protected void doGet(HttpServletRequestRequest req, HttpServletResponse resp) throws ServletException, IOException { final PrintWriter out = resp.getWriter(); resp.setContentType("text/plain"); try { out.println("Testing with UserTransaction inside transactional bean..."); transactionalBean.testWithTxRequiredAndUserTransaction(); } catch (Exception ex) { out.println("Exception: " + ex); ex.printStackTrace(out); } } } {code} The main transaction is initiated and managed by the transactional intervceptor

7. Oki, then sample.zip was not the intended one ;). Reproduced