

High Level Steps of an MVP migration to JBOSS EAP7

Hour	Summary	Recommendations
0 - 1	<p>Evaluation</p> <ul style="list-style-type: none">• Evaluation and understanding of current project structure, what is needed and what isn't for the MVP migration.• Understand XML files (web.xml, persistence.xml etc) which may need to change due to upgrading.• Run mvn dependency:tree to help examine dependencies.• Evaluate current build and if it is hiding implicit dependencies and how dependencies are drawn in.• Understand the target plugin libraries which will be needed. <p>Links https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/7.1/html/development_guide/using_maven_with_eap https://mvnrepository.com/</p>	<p>Mark TODO places in files to look down the line</p> <p>If in doubt search maven repo <lib name> https://mvnrepository.com/</p> <p>Opt to find libraries within JBOSS EAP 7 first, and catalog them for future reference. If not aware have a look in modules directory in a JBOSS 7 installation.</p>
1 - 2	<p>Maven Build Changes - mvn clean successful</p> <p>Incrementally change pom.xml with below and run and mvn clean after each step which will highlight</p>	<p>The mvn clean simply helps ensure integrity of maven project files.</p>

	<p>issues along the way. We do not need to compile yet as the clean will first draw out issues to resolve</p> <ul style="list-style-type: none"> • change the version mvn versions:set -DnewVersion=2.2.1-eap7 • Change the JVM to point to Java 8 and compatible maven version • Source code/compiler to 1.8 • Upgrade Maven plugin versions • Then look at the order of compiling and work way through • Comment out dependencies which cannot be found in this case OSGI etc. • Add JBOSS EAP pom.xml to inherit lib versions 	<p>Easier to comment out dependencies which do not fit the mould in Java 8/EAP 7 then revisit later</p> <p>Maven with EAP 7 https://access.redhat.com/documentation/en-us/red_hat_iboss_enterprise_application_platform/7.1/html/development_guide/using_maven_with_eap</p> <p>OpenJPA to Hibernate is fraught with potential problems depending on the code and annotations used. In this case only JPA annotations were used which were compatible. However different queries can result. Best to do side by side testing with SQL output for comparison. Currently working on a framework which helped a previous complex upgrade.</p> <p>Try to centralise dependencies consistently in another pom (ideally) or at the top level project</p>
	<p>Ouput 1. - Result from mvn clean 1st run.</p> <pre>[INFO] DayTrader :: Performance Benchmark Sample 2.2.1-eap7 SUCCESS [0.140 s] [INFO] DayTrader :: Modules SUCCESS [0.002 s] [INFO] DayTrader :: Modules - Util SUCCESS [0.002 s] [INFO] DayTrader :: Modules - Beans SUCCESS [0.004 s] [INFO] DayTrader :: Modules - Entities SUCCESS [0.003 s] [INFO] DayTrader :: Modules - Core SUCCESS [0.003 s] [INFO] DayTrader :: Modules - EJB3 SUCCESS [0.004 s] [INFO] DayTrader :: Modules - WS Application Client SUCCESS [0.002 s] [INFO] DayTrader :: Modules - SOAP SUCCESS [0.004 s] [INFO] DayTrader :: Modules - Web SUCCESS [0.003 s]</pre>	

	[INFO] DayTrader :: Modules - Quote Streamer SUCCESS [0.003 s] [INFO] DayTrader :: Modules - JSON Proxy SUCCESS [0.003 s] [INFO] DayTrader :: Assemblies SUCCESS [0.003 s] [INFO] DayTrader :: Assemblies for JavaEE SUCCESS [0.003 s] [INFO] DayTrader :: Assemblies - JavaEE WAR SUCCESS [0.003 s] [INFO] DayTrader :: Assemblies - JavaEE EAR SUCCESS [0.003 s] [INFO] Daytrader plans SUCCESS [0.003 s] [INFO] DayTrader :: Assemblies for Web containers SUCCESS [0.003 s] [INFO] DayTrader :: Assemblies - Web + Direct JDBC SUCCESS [1.610 s] [INFO] DayTrader :: Assemblies - Web + Direct JPA 2.2.1-eap7 SUCCESS [0.004 s]	
2 - 4	Maven Build Changes - mvn install <ul style="list-style-type: none"> • Change logging to slf4j - this is an implicit dependency within JBOSS EAP • Upgrade Junit library to 4.x • Review code and mark TODOs for review • DayTrader Beans compiled but key review of code showed Serializable without implementing a serializable uid • Entities required the hibernate dependency • Change persistence xml header and provider implementation. Comment out openjpa properties for now. • Core and EJB required Java EE libs which are provided by JBOSS EAP7 and should be marked as provided and not included in any EAR or WAR file. • 	Custom Iterators in Java 5 or 6 can more than likely be removed - a nice thing to do extending java objects such as HashMap or List is not ideal, better to rewrite/wrap or use concurrent instead on own synchronisation code - maintenance Serialization uid is important in marshalling and unmarshalling where implements Serializable { For example add this private static final long serialVersionUID = 7L; Hibernate dependency <dependency> <groupId>org.hibernate</groupId> <artifactId>hibernate-core</artifactId> </dependency> For persistence xml change <provider>org.apache.openjpa.persistence.PersistenceProviderImpl</pr vider> to

		<p><provider>org.hibernate.ejb.HibernatePersistence</provider></p> <p>Persistence.xml - Change header from <persistence xmlns="http://java.sun.com/xml/ns/persistence" version="1.0"> to <persistence xmlns="http://xmlns.jcp.org/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd" version="2.1"></p> <p>Persistence.xml - comment these in first iteration - revisit <property name="openjpa.DynamicEnhancementAgent" value="false"/> <property name="openjpa.BrokerImpl" value="non-finalizing" /> <property name="openjpa.jdbc.SynchronizeMappings" value="buildSchema(ForeignKeys=true)"/> <property name="openjpa.Sequence" value="table(Table=OPENJPASEQ, Increment=100)"/></p> <p>Core and ejb modules required following dependency <dependency> <groupId>javax</groupId> <artifactId>javaee-api</artifactId> <version>7.0</version> <scope>provided</scope> </dependency></p>
--	--	--

	<p>Output 2.</p> <p>mvn install - util compiles and beans</p> <p>[INFO] -----</p> <p>[INFO] Reactor Summary:</p> <p>[INFO]</p> <p>[INFO] DayTrader :: Performance Benchmark Sample 2.2.1-eap7 SUCCESS [6.186 s]</p> <p>[INFO] DayTrader :: Modules SUCCESS [0.011 s]</p> <p>[INFO] DayTrader :: Modules - Util SUCCESS [1.564 s]</p> <p>[INFO] DayTrader :: Modules - Beans SUCCESS [0.241 s]</p> <p>[INFO] DayTrader :: Modules - Entities FAILURE [0.267 s]</p> <p>[INFO] DayTrader :: Modules - Core SKIPPED</p> <p>[INFO] DayTrader :: Modules - EJB3 SKIPPED</p> <p>[INFO] DayTrader :: Modules - WS Application Client SKIPPED</p> <p>[INFO] DayTrader :: Modules - SOAP SKIPPED</p> <p>[INFO] DayTrader :: Modules - Web SKIPPED</p> <p>[INFO] DayTrader :: Modules - Quote Streamer SKIPPED</p> <p>[INFO] DayTrader :: Modules - JSON Proxy SKIPPED</p> <p>[INFO] DayTrader :: Assemblies SKIPPED</p> <p>[INFO] DayTrader :: Assemblies for JavaEE SKIPPED</p> <p>[INFO] DayTrader :: Assemblies - JavaEE WAR SKIPPED</p> <p>[INFO] DayTrader :: Assemblies - JavaEE EAR SKIPPED</p> <p>[INFO] Daytrader plans SKIPPED</p> <p>[INFO] DayTrader :: Assemblies for Web containers SKIPPED</p> <p>[INFO] DayTrader :: Assemblies - Web + Direct JDBC SKIPPED</p> <p>[INFO] DayTrader :: Assemblies - Web + Direct JPA 2.2.1-eap7 SKIPPED</p> <p>[INFO] -----</p>	
4 - 5	<p>Webservices Rewrite</p> <ul style="list-style-type: none"> Created a new module webservices for rewrite of webservices. 	<p>WSDL will not generally need to change however this generally re-generate Java libraries with tools. The implementation will also have to be rewritten.</p>

	<ul style="list-style-type: none"> Followed procedure to regenerate code from a wsdl file https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/7.1/html/developing_web_services_applications/developing_jax_ws_web_services change wsdl location wsdlLocation = "classpath:TradeServices.wsdl" 	<p>Commands to produce files</p> <pre>\$ EAP_HOME/bin/wsconsume.sh -k -o ../resources/EchoService.wsdl \$./jboss-eap-7.1/bin/wsconsume.sh -k -o ../resources/TradeServices.wsdl</pre> <p>Using same wsdl as contract first helps to ensure not to break existing interfaces.</p>
	<p>Output</p> <p>Could not find log4j.properties or log4j.xml configuration, logging to console.</p> <p>Loading FrontEnd jaxws ...</p> <p>Loading DataBinding jaxb ...</p> <p>wsdl2java -compile -exsh false -d /Users/bfarr/projects/minimigration/daytrader-2.2.1-eap7/daytrader-2.2.1/modules/webservice/src/main/java/. -verbose -classdir /Users/bfarr/projects/minimigration/daytrader-2.2.1-eap7/daytrader-2.2.1/modules/webservice/src/main/java/. -allowElementReferences file:/Users/bfarr/projects/minimigration/daytrader-2.2.1-eap7/daytrader-2.2.1/modules/webservice/src/main/java/./resources/TradeServices.wsdl</p> <p>wsdl2java - Apache CXF 3.1.12.redhat-1</p>	
5 - 6	WebService dependencies	

	<ul style="list-style-type: none"> • Servlets are calling webservice • Change TradeServices in core to remove Remote references, this will • Remove SOAP and WSAPPCLient to be rewritten - Create an implementation class and rewrite • See this link for more information https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/7.1/html/developing_web_services_applications/developing_jax_ws_web_services#ws_clients 	
	<pre> [ERROR] /Users/bfarr/projects/minimigration/daytrader-2.2.1-eap7/daytrader-2.2.1/modules/web/src/main/java/org/apache/geronimo/samples/daytrader/web/TradeServletAction.java:[24,1] package org.apache.geronimo.samples.daytrader.soap does not exist [ERROR] /Users/bfarr/projects/minimigration/daytrader-2.2.1-eap7/daytrader-2.2.1/modules/web/src/main/java/org/apache/geronimo/samples/daytrader/web/TradeServletAction.java:[51,43] cannot find symbol [ERROR] symbol: class TradeWebSoapProxy [ERROR] location: class org.apache.geronimo.samples.daytrader.web.TradeServletAction [ERROR] -> [Help 1] [ERROR] </pre> <p>After refactoring proxy client</p> <pre> [INFO] ----- [INFO] Reactor Summary: [INFO] [INFO] DayTrader :: Performance Benchmark Sample 2.2.1-eap7 SUCCESS [0.442 s] </pre>	

	<pre> [INFO] DayTrader :: Modules SUCCESS [0.012 s] [INFO] DayTrader :: Modules - Util SUCCESS [1.392 s] [INFO] DayTrader :: Modules - Beans SUCCESS [0.224 s] [INFO] DayTrader :: Modules - Entities SUCCESS [0.270 s] [INFO] DayTrader :: Modules - Core SUCCESS [0.664 s] [INFO] DayTrader :: Modules - EJB3 SUCCESS [0.843 s] [INFO] DayTrader :: Modules - webservices SUCCESS [0.286 s] [INFO] DayTrader :: Modules - SOAP SUCCESS [0.240 s] [INFO] DayTrader :: Modules - Web SUCCESS [0.446 s] [INFO] DayTrader :: Modules - Quote Streamer SUCCESS [0.237 s] [INFO] DayTrader :: Modules - JSON Proxy FAILURE [0.113 s] [INFO] DayTrader :: Assemblies SKIPPED [INFO] DayTrader :: Assemblies for JavaEE SKIPPED [INFO] DayTrader :: Assemblies - JavaEE WAR SKIPPED [INFO] DayTrader :: Assemblies - JavaEE EAR SKIPPED [INFO] Daytrader plans SKIPPED [INFO] DayTrader :: Assemblies for Web containers SKIPPED [INFO] DayTrader :: Assemblies - Web + Direct JDBC SKIPPED [INFO] DayTrader :: Assemblies - Web + Direct JPA 2.2.1-eap7 SKIPPED [INFO] ----- </pre>	
6 -7	Maven Build Changes - mvn install <ul style="list-style-type: none"> • Add jee 7 dependency for JSON Proxy • day-trader.war details <ul style="list-style-type: none"> remove geronimo-web.xml and jboss-web.xml remove webservices.xml remove TradeSerives_mapping.xml, TradeServicesClient_mapping.xml 	<p>JBOSS EAP 7 uses deployment descriptor XML files to help customise any mapping and class loading issues. The -web.xml files are redundant and no longer of any value (but need to examine each of them)</p> <p>EAR Plugin for EAP7 was changed from and to</p> <pre> <groupId>org.apache.maven.plugins</groupId> <artifactId>maven-ear-plugin</artifactId> <configuration> <displayName>DayTrader EAR</displayName> </pre>

	<ul style="list-style-type: none"> • Note JMS Queues needed jms/TradeBrokerQueue jms/TradeStreamerTopic • Note Datasources needed to setup jdbc/TradeDataSource • DayTrader ear - pom.xml change ear plugin • 	<pre> <description>DayTrader Stock Trading Performance Benchmark Sample</description> <version>5</version> <modules> <javaModule> to to <configuration> <displayName>DayTrader EAR</displayName> <description>DayTrader Stock Trading Performance Benchmark Sample</description> <version>7</version> <defaultLibBundleDir>lib</defaultLibBundleDir> <fileNameMapping>no-version</fileNameMapping> <skinnyWars>true</skinnyWars> </configuration> </pre> <p>Recommend to use skinny wars as this will shrink the size of the EAR file as common dependencies between WAR and EAR libs are packaged only once.</p>
7 - 8	Build Completes !	
	<p>Examine and test packaged EAR</p> <ul style="list-style-type: none"> • Now examine the libs within the ear to understand where some libs are implicitly provided by JBOSS 7 or through a module • Another way to find packaged errors is simply to try and deploy it in EAP 7 	<p>Recommend to use Jboss deployment structure for dependency management of JBOSS EAP7 provided libs. https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/7.1/html/development_guide/class_loading_and_modules</p> <p>For the most part its not needed but good practice as it. See 3.9.1.</p>

	<ul style="list-style-type: none"> • needed to remove duplicate persistence.xml files • Create the JMS topics and queues needed https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/7.0/html/configuring_messaging/getting_started#intro_messaging_config For our MVP add to standalone.xml • Changed datasources JNDI names to compliant with jboss style datasources • Use H2 database for our MVP - Add properties for H2 database 	<p>Implicit Module Dependencies do not need to be included, however ensure they are not packaged in your EAR https://access.redhat.com/articles/2158031</p> <p>To check JBOSS EAP7 libs easiest to check JBOSS 7 directory Look at directory at modules/layers/system/base</p> <p>can see error on deployment is with ejb-jar delete ejb-jar.xml - not required remove this</p> <pre><plugins> <plugin> <groupId>org.apache.maven.plugins</groupId> <artifactId>maven-ejb-plugin</artifactId> <configuration> <generateClient>true</generateClient> </configuration> </plugin> </plugins></pre> <p>Links to use https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/7.1/html-single/developing_ejb_applications/index</p>
--	---	---

		<p>https://developers.redhat.com/blog/2017/06/14/coding-ejb-clients-for-jboss-eap-7-1/</p> <p>JNDI names in JBOSS for datasources</p> <pre><jta-data-source>java:jboss/datasources/TradeDataSource</jta-data-source> <non-jta-data-source>java:jboss/datasources/NoTxTradeDataSource</non-jta-data-source></pre> <p>Add properties for H2 database</p> <pre><property name="connection.driver_class" value="org.h2.Driver"/> <property name="hibernate.dialect" value="org.hibernate.dialect.H2Dialect"/> <property name="hibernate.hbm2ddl.auto" value="create-drop"/> <property name="hibernate.show_sql" value="true" /></pre>
	<p>list under the target directory following libs look questionable to have in there given they are probably already provided within JBOSS EAP 7.</p> <p>antlr-antlr-2.7.7.redhat-7.jar com.fasterxml-classmate-1.3.3.redhat-1.jar com.sun.xml.bind-jaxb-core-2.2.11.jar com.sun.xml.bind-jaxb-impl-2.2.11.jar dom4j-dom4j-1.6.1.redhat-7.jar org.apache.geronimo.specs-geronimo-jta_1.1_spec-1.1.1.jar</p>	

<p>org.apache.cxf-cxf-core-3.1.12.redhat-1.jar org.apache.neethi-neethi-3.0.3.redhat-1.jar org.apache.cxf-cxf-rt-bindings-soap-3.1.12.redhat-1.jar org.apache.ws.xmlschema-xmlschema-core-2.2.1.redhat-1.jar org.apache.cxf-cxf-rt-bindings-xml-3.1.12.redhat-1.jar org.codehaus.woodstox-stax2-api-3.1.4.redhat-1.jar org.apache.cxf-cxf-rt-databinding-jaxb-3.1.12.redhat-1.jar org.codehaus.woodstox-woodstox-core-asl-4.4.1.jar org.apache.cxf-cxf-rt-frontend-jaxws-3.1.12.redhat-1.jar org.hibernate-hibernate-core-5.1.10.Final-redhat-1.jar org.apache.cxf-cxf-rt-frontend-simple-3.1.12.redhat-1.jar org.hibernate.common-hibernate-commons-annotations-5.0.1.Final-redhat-2.jar org.apache.cxf-cxf-rt-ws-addr-3.1.12.redhat-1.jar org.hibernate.javax.persistence-hibernate-jpa-2.1-api-1.0.0.Final-redhat-2.jar org.apache.cxf-cxf-rt-ws-policy-3.1.12.redhat-1.jar org.javassist-javassist-3.20.0.GA-redhat-3.jar org.apache.cxf-cxf-rt-wsdl-3.1.12.redhat-1.jar org.jboss-jandex-2.0.3.Final-redhat-1.jar org.jboss.logging-jboss-logging-3.3.1.Final-redhat-1.jar org.ow2.asm-asm-5.0.4.jar wsdl4j-wsdl4j-1.6.3.redhat-2.jar xml-resolver-xml-resolver-1.2.0.redhat-11.jar</p> <p>Output from deploy ERROR [org.jboss.msc.service.fail] (MSC service thread 1-5) MSC000001: Failed to start service jboss.deployment.subunit."daytrader-ear-2.2.1-eap7.ear"."org.apache.geronimo.daytrader.modules-daytrader-ejb3-2.2.1-eap7.jar".PARSE: org.jboss.msc.service.StartException in service jboss.deployment.subunit."daytrader-ear-2.2.1-eap7.ear"."org.apache.geronimo.daytrader.modules-daytrader-ejb3-2.2.1-eap7</p>

.jar".PARSE: WFLYSRV0153: Failed to process phase PARSE of subdeployment
"org.apache.geronimo.daytrader.modules-daytrader-ejb3-2.2.1-eap7.jar" of deployment "daytrader-ear-2.2.1-eap7.ear"
at org.jboss.as.server.deployment.DeploymentUnitPhaseService.start(DeploymentUnitPhaseService.java:172)
at org.jboss.msc.service.ServiceControllerImpl\$StartTask.startService(ServiceControllerImpl.java:2032)
at org.jboss.msc.service.ServiceControllerImpl\$StartTask.run(ServiceControllerImpl.java:1955)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
at java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:624)
at java.lang.Thread.run(Thread.java:748)
Caused by: org.jboss.msc.service.ServiceNotFoundException: Service service
jboss.ejb.default-resource-adapter-name-service not found
at org.jboss.msc.service.ServiceContainerImpl.getRequiredService(ServiceContainerImpl.java:669)
at
org.jboss.as.ejb3.deployment.processors.MessageDrivenComponentDescriptionFactory.getDefaultResourceAdapterName(MessageDrivenComponentDescriptionFactory.java:274)
at
org.jboss.as.ejb3.deployment.processors.MessageDrivenComponentDescriptionFactory.processMessageBeans(MessageDrivenComponentDescriptionFactory.java:154)
at
org.jboss.as.ejb3.deployment.processors.MessageDrivenComponentDescriptionFactory.processAnnotations(MessageDrivenComponentDescriptionFactory.java:81)
at
org.jboss.as.ejb3.deployment.processors.AnnotatedEJBComponentDescriptionDeploymentUnitProcessor.processAnnotations(AnnotatedEJBComponentDescriptionDeploymentUnitProcessor.java:57)
at
org.jboss.as.ejb3.deployment.processors.AbstractDeploymentUnitProcessor.deploy(AbstractDeploymentUnitProcessor.java:76)
at org.jboss.as.server.deployment.DeploymentUnitPhaseService.start(DeploymentUnitPhaseService.java:165)
... 5 more

	<p>13:25:35,920 ERROR [org.jboss.as.controller.management-operation] (DeploymentScanner-threads - 1) WFLYCTL0013: Operation ("deploy") failed - address: ([("deployment" => "daytrader-ear-2.2.1-eap7.ear")]) - failure description: {"WFLYCTL0080: Failed services" => {"jboss.deployment.subunit.\"daytrader-ear-2.2.1-eap7.ear\".\"org.apache.geronimo.daytrader.modules-daytrader-ejb3-2.2.1-eap7.jar\".PARSE" => "WFLYSRV0153: Failed to process phase PARSE of subdeployment \"org.apache.geronimo.daytrader.modules-daytrader-ejb3-2.2.1-eap7.jar\" of deployment \"daytrader-ear-2.2.1-eap7.ear\" Caused by: org.jboss.msc.service.ServiceNotFoundException: Service service jboss.ejb.default-resource-adapter-name-service not found"}}}</p> <p>Caused by: org.jboss.as.server.deployment.DeploymentUnitProcessingException: WFLYEE0092: No message destination with name dt-ejb.jar#jms/TradeBrokerQueue for binding java:module/env/jms/TradeBrokerQueue"},</p>	
8 - 9	<p>JNDI name changes</p> <ul style="list-style-type: none"> • Change datasource from "java:comp/env/jdbc/TradeDataSource" to "java:jboss/datasources/TradeDataSource"; • "java:comp/env/jms/QueueConnectionFactory" to java:/ConnectionFactory • 	Advice - better to break and fix it, ie. prefer to remove and reinstate
	<p>10:46:30,065 ERROR [org.apache.geronimo.samples.daytrader.util.Log] (ServerService Thread Pool -- 105) Error: TradeDirect:init -- error on JNDI lookups of DataSource -- TradeDirect will not work javax.naming.NameNotFoundException: env/jdbc/TradeDataSource -- service jboss.naming.context.java.module."daytrader-ear-2.2.1-eap7"."org.apache.geronimo.daytrader.assemblies.javaee-daytrader-war-2.2.1-eap7".env.jdbc.TradeDataSource</p>	

	<p>EJB injection</p> <ul style="list-style-type: none"> • Use CDI container dependency injection <p>Rather than <code>trade = (TradeServices)</code> <code>context.lookup("java:comp/env/ejb/TradeSLSBBean");</code> use <code>@EJB</code> <code>private TradeSLSBRemote tradeSLSBRemote;</code></p>	<p>Commented code out like this to look up EJBs for now</p> <pre> /* try { Class c = Class.forName("org.apache.geronimo.samples.daytrader.ejb3.TradeSLSBRemote"); if ((trade == null) (!(trade.getClass().isAssignableFrom(c)))) { InitialContext context = new InitialContext(); try { trade = (TradeServices) context.lookup("java:comp/env/ejb/TradeSLSBBean"); } catch (Exception ex) { Log.error("TradeAction:createTrade - Lookup of TradeSLSBRemote failed!!!"); trade = (TradeServices) context.lookup("TradeSLSBBean"); } } } catch (Exception e) { Log.error("TradeAction:TradeAction() Creation of Trade EJB 3 failed\n" + e); re = new RuntimeException(e); } */ </pre>
10 - 12	Problem - TradeAction	

	<p>TradeAction is a problem as its not in the right area, doing a class.forName should bne got ridden of</p> <p>when introducing EJB3 it caused a maven cyclic dependency, so this needs to be refactored</p> <p>This clearly belongs in the web application jar given all references that use it are in web and web already relies on core</p> <p>In this case we did some refactoring of dependencies.</p>	
--	--	--