JUDCon
JBoss Users & Developers Conference
Boston: 2010
About me

• Jesper Pedersen
  – Core developer, JBoss by Red Hat

• Project lead for
  – JBoss JCA – JCA container
  – Tattletale – software quality tool
  – JBoss Profiler 2 – profiler suite
  – Papaki – high-performance annotation scanner

• Chairman for Boston JBoss User Group
Overview

• What is JCA?
• JCA 1.6
• JBoss JCA
What is JCA?
What is JCA?

- The Java Connector Architecture (JCA) standard specifies how the Java Enterprise Edition platform connects to Enterprise Information Systems (EIS)
  - Database systems
  - Messaging system
  - Mainframe transaction processing
- Standard is created in JCP
What is JCA?

• JCA defines the contract between the JCA container and the EIS
  – Scalable, Transactional, Secure, Client API
• Resource Adapter (EIS vendors)
  – Protocol specific communication
  – Multiple AS vendors
• Resource Adapter (AS vendors)
  – Support multiple EISs
What is JCA?

- Resource adapters can be divided into
  - Outbound, Inbound and Bidirectional
  - Transactions
    - NoTransaction, LocalTransaction, XATransaction
  - Security
    - javax.security.auth.Subject
What is JCA?

• Examples
  – JDBC resource adapter (outbound)
    • Wraps JDBC API and adds connection pooling
  – Timer resource adapter (inbound)
    • Enables Quartz to invoke MessageEndpoints
  – JMS resource adapter (bidirectional)
    • Wraps JMS API (outbound part)
    • Enables message invocation on MDBs (inbound)
Java Connector Architecture 1.6
JSR-322
JCA 1.6

• Developed in JCP
  – Sun (Spec lead), Red Hat, Oracle, IBM
  – Ericsson, SAP, Tmax
• Focus on 4 key areas
  – Ease of Development
  – Generic Work context
  – Security Inflow
  – Misc enhancements
JCA 1.6

• Ease of Development
  – Make use of annotations
    • @ Connector
    • @ ConnectionDefinitions
    • ...
  – Make META-INF/ra.xml optional
  – Sparse metadata
  – Better defaults

@Connector
public class MyResourceAdapterImpl implements ResourceAdapter
JCA 1.6

• Generic Work Context (Message delivery / Work submission)
  – WorkContextProvider
  – TransactionContext, SecurityContext, HintsContext

interface WorkContextProvider
   List<WorkContext> getWorkContextProviders();

interface WorkContext
   String getName();
   String getDescription();
JCA 1.6

- Security Inflow
  - End-to-end security model
  - Message delivery
  - Work processing
  - Uses JSR-196: Java Authentication Service Provider Interface for Containers
  - JSR-196 Callback's used for establish caller identity
JCA 1.6

- Misc enhancements
  - Transaction support at run-time
  - Distributed work
  - Message endpoint timeouts
  - Retryable exceptions
  - Classloading
  - Bean validation (JSR-303)

```java
public class MyResourceAdapterImpl implements ResourceAdapter
@ConfigProperty @NotNull @Size(min=5, max=5)
private String myProperty;
```
JCA 1.6

- Standalone environment
  - Defines a JCA environment outside of an application server
  - Focus on outbound communication – inbound is optional
  - Includes transactions, security and bean validation
  - Up to vendor to choose the rest, like application component model
JBoss JCA
The JBoss JCA project aims to implement the JCA 1.6 specification

Three modes of operation
- Standalone (JCA 1.6 Chapter 3.5)
- Embedded
- Inside JBoss Application Server
JBoss JCA

- POJO based architecture
  - Light-weight
  - Fast
- Depends on frameworks also used in JBoss Application Server
- Supports two POJO kernels
  - Fungal
  - JBoss Microcontainer
JBoss JCA

• Configuration done through XML files
  – Thread pools
  – WorkManager
  – BootstrapContext
  – Deployer chain

• Setup different environments for the resource adapters
  – Bind a RA to a BootstrapContext through jboss-ra.xml
JBoss JCA

<!-- RA deployer -->
<bean name="RADeployer"
     interface="com.github.fungal.spi.deployers.Deployer"
     class="org.jboss.jca.deployers.fungal.RADeployer">
    <property name="ArchiveValidation">true</property>
    <property name="ArchiveValidationFailOnWarn">false</property>
    <property name="ArchiveValidationFailOnError">true</property>
    <property name="BeanValidation">true</property>
    <property name="PrintStream">
        <inject bean="JBossStdioContext" property="Out"/>
    </property>
    <property name="DefaultBootstrapContext">
        <inject bean="DefaultBootstrapContext"/>
    </property>
    <property name="TransactionManager">
        <inject bean="RealTransactionManager"/>
    </property>
</bean>
JBoss JCA

- Standalone environment
  - Kernel: Fungal / JBoss MC
  - Logging: JBoss Logging + JBoss Stdio
  - Naming: JBoss Naming
  - Threads: JBoss Threads
  - Bean validation: Hibernate Validator
  - Transactions: JBoss TS
  - Security: JBoss Security
  - Web: Jetty
JBoss JCA

- **Embedded use-cases**
  - Embed a JCA container inside an application
  - Easy unit testing of resource adapters

- **Embedded container**
  - Running In-VM
  - Built on standalone functionality
  - As easy as possible
  - Integration with ShrinkWrap and Arquillian
JBoss JCA

• Tools: Validator
  – Help resource adapter developers make their implementation spec compliant
  – Validates resource adapter against rules defined in the specification
    • ResourceAdapter, ManagedConnectionFactory, ManagedConnection, ...
  – Deployer, Standalone, Ant, Maven
JBoss JCA

- Tools: Code Generator
  - Help resource adapter developers get started
  - Generates code skeleton for
    - JCA 1.0, JCA 1.5, JCA 1.6
  - Based on user input / configuration
  - Provide implementation hints
Enough already! 

DEMO TIME
The future

• Enterprise level features
• Optional parts of the specification
  – f.ex. DistributedWorkManager
• OSGi/JCA
• More focus on usability
  – especially embedded
• Tools / Samples
Last thoughts

- JCA 1.6 adds key areas to the specification
- Included in the full profile of Java Enterprise Edition 6
- The JBoss JCA standalone / embedded distribution provides a way to get started today
Join our hackfest!

• Application Server hackfest
  – 7.30 PM -> 2.00 AM
  – Meet core JBoss developers and follow hackers
  – Get a quick introduction to several projects
  – Start hacking on your issue of interest :)
Links

• JCA 1.6 (JSR-322)

• JBoss JCA
  – http://www.jboss.org/jca

• Fungal
  – http://jesperpedersen.github.com/fungal/

• JBoss MC
  – http://www.jboss.org/jbossmc
Links

- ShrinkWrap
  - http://community.jboss.org/wiki/shrinkwrap
- Arquillian
  - http://www.jboss.org/arquillian