JBoss JCA

Connecting Your
Enterprise Information Systems

jboss.org

JBoss JCA

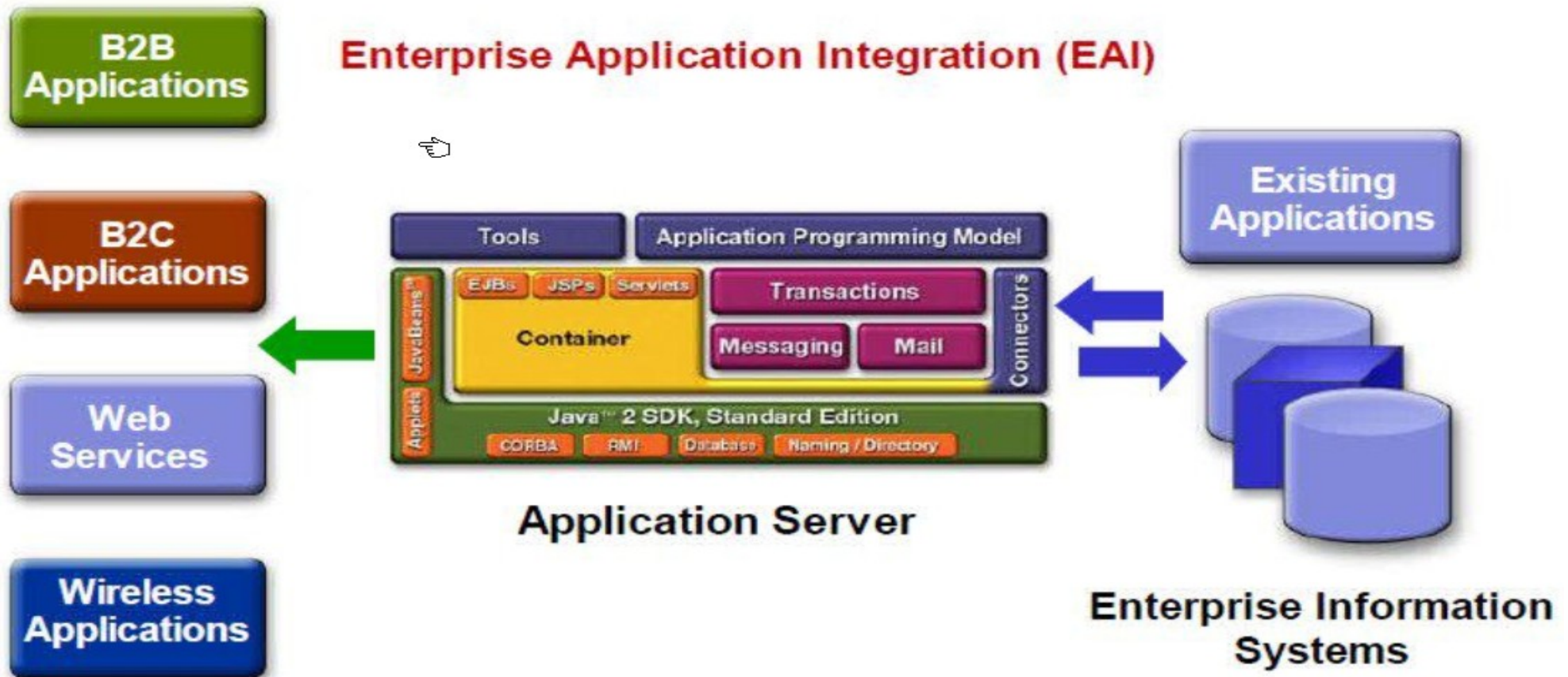- What is JCA ?
- JCA 1.6
- JBoss JCA

JBoss JCA

What is JCA ?

JBoss 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 systems
    - Mainframe transaction processing
- Standard is created in JCP
    - JSR-322: JCA 1.6
    - Part of the full-profile of Java EE 6

JBoss JCA – What is JCA ?

JBoss JCA – 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
    - Support multiple AS vendors
- Resource adapter (AS vendors)
    - Support multiple EISs

JBoss JCA – Key Components

- ResourceAdapter

  – Represents the resource adapter instance

    void endpointActivation(MessageEndpointFactory endpointFactory, ActivationSpec spec)

    void endpointDeactivation(MessageEndpointFactory endpointFactory, ActivationSpec spec)

    XAResource[] getXAResources(ActivationSpec[] specs)

    void start(BootstrapContext ctx)

    void stop()

JBoss JCA – Key Components

- ## ConnectionManager

    – Connection request to the application
    server

Object allocateConnection(ManagedConnectionFactory mcf,
ConnectionRequestInfo cri)

JBoss JCA – Key Components

- ManagedConnectionFactory / ManagedConnection
  - – Physical connection to EIS

```
Object
createConnectionFactory(ConnectionManager cm)

ManagedConnection
createManagedConnection(javax.security.auth.Subject subject,
                        ConnectionRequestInfo cri)

ManagedConnection
matchManagedConnections(java.util.Set connectionSet,
                        javax.security.auth.Subject subject,
                        ConnectionRequestInfo cri)

Object
getConnection(javax.security.auth.Subject subject,
              ConnectionRequestInfo cri)
```

JBoss JCA – Key Components

- ## ConnectionFactory / Connection
  - – A connection in EE
  - – Specific to EIS or use javax.resource.cci

Connection getConnection()

Connection getConnection(ConnectionSpec properties)

Interaction createInteraction()

JBoss JCA – Key Components

- ## MessageEndpointFactory / MessageEndpoint

  - Message delivery

MessageEndpoint createEndpoint(XAResource xaResource)

MessageEndpoint createEndpoint(XAResource xaResource, long timeout)

void afterDelivery()

void beforeDelivery(java.lang.reflect.Method method)

JBoss JCA – Key Components

- ## WorkManager

  – ### Work processing

void doWork(Work work)

void doWork(Work work, long startTimeout,
            ExecutionContext execContext, WorkListener workListener)

void scheduleWork(Work work)

void scheduleWork(Work work, long startTimeout,
                ExecutionContext execContext,
                WorkListener workListener)

long startWork(Work work)

long startWork(Work work, long startTimeout,
             ExecutionContext execContext,
             WorkListener workListener)

JBoss JCA

Java Connector Architecture 1.6
JSR-322

JBoss JCA – JCA 1.6

- Developed in JCP – JSR-322 / EE6
    - Sun (Spec Lead)
    - Red Hat, Oracle, IBM
    - Ericsson, SAP and Tmax
- Focus on 4 key areas
    - Ease of Development
    - Generic Work Context
    - Security Inflow
    - Misc enhancements

JBoss JCA – JCA 1.6

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

@Connector

public void MyResourceAdapter implements ResourceAdapter

JBoss JCA – JCA 1.6

- Generic Work Context – Message delivery / Work submission
  - WorkContextProvider
  - TransactionContext, SecurityContext, HintsContext

```
interface WorkContextProvider

    List<WorkContext> getWorkContexts();


interface WorkContext

    String getName()

    String getDescription()
```

JBoss JCA – JCA 1.6

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

JBoss JCA – JCA 1.6

- Misc enhancements
  - Transaction support at run-time
  - Distributed work
  - Message endpoint timeouts
  - Retryable exceptions
  - Classloading
  - Bean validation (JSR-303) integration
  - Standalone environment
- Knock yourself out – 592 pages of fun :)

JBoss JCA

JBoss JCA

JBoss JCA – JBoss JCA

- The JBoss JCA container is an implementation of the JCA 1.6 specification

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

JBoss JCA – JBoss JCA

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

JBoss JCA – JBoss JCA

- Tools: Validator
  - Validates resource adapter against rules defined in the specification
  - Deployer, Standalone, Ant, Maven
- Tools: Code generator
  - Generates code skeleton for
    - JCA 1.0
    - JCA 1.5
    - JCA 1.6
  - Based on user input / configuration

JBoss JCA – JBoss JCA

```java
import org.jboss.jca.embedded.EmbeddedJCA;

// Embedded JCA container
EmbeddedJCA embedded = new EmbeddedJCA();

// Startup
embedded.startup();

URL archive = getURL("myresourceadapter.rar");

try {
  embedded.deploy(archive);
} catch (Throwable t) {
  // Handle throwable
} finally {
  embedded.undeploy(archive);
}

// Shutdown
embedded.shutdown();
```

JBoss JCA

The future

JBoss JCA – The future

- JCA 1.5 profile
- Enterprise level features
- JCA / OSGi
- Optional parts of the JCA specification
- More focus on usability
    - Embedded especially
- Powerful database services
- Tools / Samples

JBoss JCA

Links

JBoss JCA - Links

- JCA 1.6 (JSR-322)
    - http://jcp.org/en/jsr/detail?id=322
- Website
    - http://jboss.org/jca
- JIRA
    - https://jira.jboss.org/jira/browse/JBJCA
- Forum
    - http://community.jboss.org/en/jbossas/jca