Developing Connectors Using J2EE Connector Architecture

Valerie Pressley
Hewlett-Packard
Review – J2EE Architecture
Background

Web-driven environment: pros and cons
- Benefit: access to information
- Problem: how to leverage information in EISs

Web-driven environment: tools
- J2EE
  - Provides access to databases – JDBC
  - What about EISs? – J2CA
J2CA Terminology

- Application Server
- Application Component
- Connector/Resource Adapter
- Enterprise Information System (EIS)
J2CA Architecture
Connection Mgmt. Interfaces
ConnectionFactory

- Called by the application component
- Creates Connection instances
Connection

- Application-level connection handle
- Used by a component to access an EIS
LocalTransaction

- Used to demarcate local transactions
- Created with Connection method `getLocalTransaction`
- Created with ManagedConnection method `getLocalTransaction`
ManagedConnectionFactory

- Creates managed connections
- Managed by the application server
- Creates ConnectionFactory instances
ManagedConnection

- Physical connection to the EIS
- Creation initiated by the application server
- Created with the ManagedConnectionFactory method createManagedConnection
XAResource

- JTA Mapping of the standard XA interface
- Manages two-phase commits
- Created with the ManagedConnection method getXAResource
Contracts

- Application contract
- Container-component contract
- System contracts
- EIS-Specific contract
Application Contract

- API that an application uses for EIS access
- May use the Common Client Interface (CCI)
- May use EIS-specific API (e.g. JDBC)
Container-Component Contract

- Contract between component and container hosting the component
- Defines services available to component (e.g. JNDI)
System Contracts

- Enable the application server to interact with the resource adapter in a standard manner
  - Connection management
  - Transaction management
  - Security
EIS-Specific Contract

- Between the resource adapter and the EIS
- Varies depending on the type of EIS
Retrieving a Connection

1. getConnection()
2. ConnectionManager.allocateConnection

3. createManagedConnection

The application server looks up a set of possible connections from the connection pool.

4. <<create>>
5. addConnectionEventListener
6. (optional) setLogWriter

7. getXAResource
8. Transaction.xaResource(XAResource)
9. XAResource.start(XID)

10. getConnection(Subject, ConnectionRequestInfo)

11: return javax.resource.cci.Connection
12: return javax.resource.cci.Connection
Step 1

- Application component calls getConnection()
- Method of ConnectionFactory class
- ConnectionFactory found using JNDI lookup
Step 2

- ConnectionFactory calls allocateConnection()
- Method of ConnectionManager
- ConnectionFactory was passed a ConnectionManager instance by the application server when created
- Application server is responsible for fulfilling the connection request
Step 3

- Application server looks in the connection pool for connections that can handle the request
  - If one is found, it is returned to the resource adapter
  - If not, the application server calls createManagedConnection()
  - Method of ManagedConnectionFactory
Step 4

- The ManagedConnectionFactory returns a ManagedConnection to the application server
Step 5

- The application server calls `addConnectionEventListener()` method of ManagedConnection
Step 6

- Application calls `setLogWriter()`
- Method of ManagedConnection
- Dependent on application server logging
Step 7

- Application server calls `getXAResource()`
- Method of `ManagedConnection`
Step 8

- Application calls enlistResource()
- Method of Transaction
- Making use of the application server’s transaction manager
Step 9

- Transaction manager calls start()
- Method of XAResource
- Or method of LocalTransaction
Step 10

- Application server calls `getConnection()`
- Method of `ManagedConnection`
- Returns the application-level connection
Step 11

- Application server returns the connection handle to the resource adapter
- To the ConnectionFactory
Step 12

- The resource adapter returns the connection handle to the application component
Sample Code

Common Client Interface (CCI) Code

- Connection Interface
  - SampleConnection Class
- ConnectionFactory Interface
  - SampleConnectionFactory Class
Connection Code

```java
package connector;

import javax.resource.ResourceException;
import javax.resource.cci.*;

public class SampleConnection implements Connection {
    private SampleManagedConnection mc;

    public SampleConnection() {
        System.out.println("In SampleConnection constructor...");
    }

    public void close() throws ResourceException {
        System.out.println("In SampleConnection.close()...");
    }

    public Interaction createInteraction() throws ResourceException {
        System.out.println("In SampleConnection.createInteraction()...");
        return null;
    }
}
```
Connection Code

```java
public LocalTransaction getLocalTransaction() throws ResourceException {
    System.out.println("In SampleConnection.getLocalTransaction()...");
    return null;
}

document.content

public ConnectionMetaData getMetaData() throws ResourceException {
    System.out.println("In SampleConnection.getMetaData()...");
    return null;
}

document.content

public ResultSetInfo getResultSetInfo() throws ResourceException {
    System.out.println("In SampleConnection.getResultSetInfo()...");
    return null;
}
```
ConnectionFactory Code

```java
package connector;

import java.io.Serializable;
import javax.naming.Reference;
import javax.resource.ResourceException;
import javax.resource.spi.*;
import javax.resource.cci.*;

public class SampleConnectionFactory implements ConnectionFactory, Serializable {
    private ConnectionManager cm;
    private ManagedConnectionFactory mcf;

    public SampleConnectionFactory(ManagedConnectionFactory mcf, ConnectionManager cm) {
        System.out.println("In SampleConnectionFactory constructor...");
        this.mcf = mcf;

        if (cm == null)
            this.cm = new SampleConnectionManager();
        else
            this.cm = cm;
    }
}
```
ConnectionFactory Code

```java
public Connection getConnection() throws ResourceException
{
    System.out.println("In SampleConnectionFactory.getConnection()...");
    return (Connection) cm.allocateConnection(mcf, null);
}

public Connection getConnection(ConnectionSpec properties) throws ResourceException
{
    System.out.println("In SampleConnectionFactory.getConnection(ConnectionSpec)...");
    return (Connection) cm.allocateConnection(mcf, null);
}

public ResourceAdapterMetaData getMetaData() throws ResourceException
{
    System.out.println("In SampleConnectionFactory.getMetaData()...");
    return new SampleResourceAdapterMetaData();
}
```
ConnectionFactory Code

```java
public RecordFactory getRecordFactory() throws ResourceException {
    System.out.println("In SampleConnectionFactory.getRecordFactory()..."olec);    return null;
}

public Reference getReference() {
    System.out.println("In SampleConnectionFactory.getReference()..."olec);    return null;
}

public void setReference(Reference reference) {
    System.out.println("In SampleConnectionFactory.setReference()..."olec);
}
```
Sample Code

Managed Environment

- ManagedConnectionFactory
  - SampleManagedConnectionFactory Class
- ManagedConnection
  - SampleManagedConnection Class
ManagedConnectionFactory Code

package connector;

import java.io.PrintWriter;
import java.io.Serializable;
import java.util.Iterator;
import java.util.Set;
import javax.resource.ResourceException;
import javax.resource.spi.*;
import javax.security.auth.Subject;

public class SampleManagedConnectionFactory implements ManagedConnectionFactory, Serializable {

    public SampleManagedConnectionFactory() {
        System.out.println("In SampleManagedConnectionFactory constructor...");
    }

    public Object createConnectionFactory() throws ResourceException {
        System.out.println("In SampleManagedConnectionFactory.createConnectionFactory...");
        return new SampleConnectionFactory(this, null);
    }
}
ManagedConnectionFactory Code

```java
public Object createConnectionFactory(ConnectionManager cm) throws ResourceException {
    System.out.println("In SampleManagedConnectionFactory.createConnectionFactory(ConnectionManager)...");
    return new SampleConnectionFactory(this, cm);
}

custom ManagedConnection createManagedConnection(Subject subject, ConnectionRequestInfo info) {
    System.out.println("In SampleManagedConnectionFactory.createManagedConnection()...");
    return new SampleManagedConnection(this, "test");
}

custom ManagedConnection matchManagedConnections(Set connectionSet, Subject subject, ConnectionRequestInfo info) throws ResourceException {
    System.out.println("In SampleManagedConnectionFactory.matchManagedConnections()...");
    return null;
}
```
ManagedConnectionFactory Code

```java
public ManagedConnection matchManagedConnections(Set connectionSet, Subject subject, ConnectionRequestInfo info)
    throws ResourceException
{
    System.out.println("In SampleManagedConnectionFactory.matchManagedConnections()...");
    return null;
}

public void setLogWriter(PrintWriter out) throws ResourceException
{
    System.out.println("In SampleManagedConnectionFactory.setLogWriter");
}

public PrintWriter getLogWriter() throws ResourceException
{
    System.out.println("In SampleManagedConnectionFactory.getLogWriter");
    return null;
}
```
public boolean equals(Object obj)
{
    if(obj == null)
        return false;
    if(obj instanceof SampleManagedConnectionFactory)
    {
        int hash1 = ((SampleManagedConnectionFactory)obj).hashCode();
        int hash2 = hashCode();
        return hash1 == hash2;
    }
    else
    {
        return false;
    }
}

public int hashCode()
{
    return 1;
}
package connector;

import java.io.PrintWriter;
import java.util.*;
import javax.resource.cci.Connection;
import javax.resource.NotSupportedException;
import javax.resource.ResourceException;
import javax.resource.spi.*;
import javax.security.auth.Subject;
import javax.transaction.xa.XAResource;
import javax.resource.resource.spi.*;
import javax.security.auth.Subject;
import javax.transaction.xa.XAResource;

public class SampleManagedConnection implements ManagedConnection
{
    private SampleConnectionEventListener sampleListener;
    private String user;
    private ManagedConnectionFactory mcf;
    private PrintWriter logWriter;
    private boolean destroyed;
    private Set connectionSet;
ManagedConnection Code

SampleManagedConnection(ManagedConnectionFactory mcf, String user)
{
    System.out.println("In SampleManagedConnection constructor...");
    this.mcf = mcf;
    this.user = user;
    connectionSet = new HashSet();
    sampleListener = new SampleConnectionEventListener(this);
}

public void addConnectionEventListener(ConnectionEventListener listener)
{
    System.out.println("In SampleManagedConnection.addConnectionEventListener");
    sampleListener.addConnectorListener(listener);
}
ManagedConnection Code

```java
public void associateConnection(Object connection) {
    System.out.println("In SampleManagedConnection.associateConnection()...");
}

public void cleanup() {
    System.out.println("In SampleManagedConnection.cleanup()...");
}

public void destroy() {
    System.out.println("In SampleManagedConnection.destroy()...");
    destroyed = true;
}
```
ManagedConnection Code

```java
public Object getConnection(Subject subject, ConnectionRequestInfo connectionRequestInfo)
    throws ResourceException    {
    System.out.println("In SampleManagedConnection.getConnection(Subject, ConnectionRequestInfo)"");
    SampleConnection sampleCon = new SampleConnection();
    addConnection(sampleCon);
    return sampleCon;
}

public LocalTransaction getLocalTransaction()
{
    System.out.println("In SampleManagedConnection.getLocalTransaction()...");
    return null;
}

public PrintWriter getLogWriter() throws ResourceException
{
    System.out.println("In SampleManagedConnection.getLogWriter()...");
    return logWriter;
}
```
ManagedConnection Code

```java
ManagedConnectionFactory getManagedConnectionFactory()
{
    System.out.println("In
SampleManagedConnection.getManagedConnectionFactory");
    return mcf;
}

public ManagedConnectionMetaData getMetaData()
throws ResourceException
{
    System.out.println("In SampleManagedConnection.getMetaData()...");
    return new SampleManagedConnectionMetaData(this);
}

public XAResource getXAResource()
throws ResourceException
{
    System.out.println("In SampleManagedConnection.getXAResource()...");
    return null;
}
```
ManagedConnection Code

```java
boolean isDestroyed()
{
    System.out.println("In SampleManagedConnection.isDestroyed()...");
    return destroyed;
}

void removeConnection(SampleConnection sampleCon)
{
    System.out.println("In SampleManagedConnection.removeConnection()...");
    connectionSet.remove(sampleCon);
}

public void removeConnectionEventListener(ConnectionEventListener listener)
{
    System.out.println("In SampleManagedConnection.removeConnectionEventListener()...");
    sampleListener.removeConnectorListener(listener);
}

void sendEvent(int eventType, Exception ex)
{
    System.out.println("In SampleManagedConnection.sendEvent(int)...");
    sampleListener.sendEvent(eventType, ex, null);
}
```
ManagedConnection Code

```java
void sendEvent(int eventType, Exception ex, Object connectionHandle)
{
    System.out.println("In SampleManagedConnection.sendEvent(int, Exception, Object)...");
    sampleListener.sendEvent(eventType, ex, connectionHandle);
}

public void setLogWriter(PrintWriter out) throws ResourceException
{
    System.out.println("In SampleManagedConnection.setLogWriter(PrintWriter)");
    logWriter = out;
}
```
Sample Application

- Application Component
  - JSP
  - Servlet
  - EJB

- Resource Adapter
  - .rar file
Application Component Code

```html
<%@ page import="javax.naming.*, javax.resource.cci.ConnectionFactory, javax.resource.cci.Connection" %>
<html>
<head>
  <title>Connector example</title>
</head>
<body bgcolor="#ffffff">
  InitialContext initCtx = null;
  Connection conn = null;
  try {
    initCtx = new InitialContext();
    ConnectionFactory cf = (ConnectionFactory)
      initCtx.lookup("SampleConnector");
    conn = cf.getConnection();
    // Perform operations using the connection.
    // Available operations are determined by the resource adapter.
    conn.close();
  }
  catch (NamingException ne) {
    System.out.println("Error with context: " + ne);
  }

  <h2>Performed a lookup and got a connection</h2>
</body>
</html>
```
Final Steps

- Package resource adapter (.rar)
- Deploy resource adapter
  - Standalone
  - Bundled
- Run application
Console Output

```plaintext
C:\WINNT\system32\cmd.exe - startweblogic

<May 10, 2005 7:22:10 AM EDT> <Notice> <WebLogicServer> <BEA-000331> <Started WebLogic Admin Server "myserver" for domain "ovsidomain" running in Development Mode>
<May 10, 2005 7:22:10 AM EDT> <Notice> <WebLogicServer> <BEA-000360> <Server started in RUNNING mode>
<May 10, 2005 7:22:10 AM EDT> <Notice> <WebLogicServer> <BEA-000355> <Thread "ListenThread.Default" listening on port 7001, ip address *.*>
In SampleManagedConnectionFactory.getConnection()... 
In SampleManagedConnectionFactory.matchManagedConnections()...
In SampleManagedConnectionFactory.createManagedConnection()...
In SampleManagedConnection constructor...
In SampleConnectionEventListener constructor...
In SampleManagedConnection.addConnectionEventListener...
In SampleManagedConnection.getConnectionSubject.()...
In SampleManagedConnection constructor...
In SampleManagedConnection.addConnection()...
In SampleConnection.close()...
In SampleManagedConnection.destroy()...
```