Service Component Architecture JCA Binding Specification Version 1.1

Working Draft 02

7 August 2008

Specification URIs:
This Version:
   http://docs.oasis-open.org/sca-bindings/sca-jcabinding-1.1-spec-WD-02.doc
   http://docs.oasis-open.org/sca-bindings/sca-jcabinding-1.1-spec-WD-02.pdf (Authoritative)
Previous Version:
   http://docs.oasis-open.org/sca-bindings/sca-jcabinding-1.1-spec-WD-01.doc
   http://docs.oasis-open.org/sca-bindings/sca-jcabinding-1.1-spec-WD-01.pdf (Authoritative)
Latest Version:
   http://docs.oasis-open.org/sca-bindings/sca-jcabinding-1.1-spec-WD-02.doc
   http://docs.oasis-open.org/sca-bindings/sca-jcabinding-1.1-spec-WD-02.pdf (Authoritative)

Latest Approved Version:

Technical Committee:
   OASIS Service Component Architecture / Bindings (SCA-Bindings) TC

Chair(s):
   Simon Holdsworth, IBM

Editor(s):
   Anish Karmarkar, Oracle
   Piotr Przybylski, IBM

Related work:
   This specification replaces or supercedes:
      • Service Component Architecture JCA Binding Specification Version 1.00 20 September 2007
   This specification is related to:
      • Service Component Architecture Assembly Model Specification Version 1.1
      • Service Component Architecture Policy Framework Specification Version 1.1

Declared XML Namespace(s):
   http://docs.oasis-open.org/ns/opencsa/sca/200712

Abstract:
   This document presents bindings describing access and connectivity to the services provided by
   the Enterprise Information System (EIS).
   This version of the document describes JCA Bindings thus narrowing connectivity down to the
   connectivity to the EIS system external to the SCA system, based on the Java EE Connector
   Architecture specification and implemented in Java.
Further specification is necessary to define EIS Bindings between different SCA runtimes within SCA system, for example J2EE and EIS based runtimes.

The binding specified in this document applies to the composite’s references and services.

The connection to exchange data with the EIS is characterized by two sets of configuration parameters, the connection and interaction parameters. The former set determines the location of the target system the latter determines characteristics that need to be specified to invoke one specific service available at the endpoint. JCA Binding model captures these parameters as separate sets to allow their reuse and reconfiguration.

**Status:**

This document was last revised or approved by the OASIS Service Component Architecture / Bindings (SCA-Bindings) TC on the above date. The level of approval is also listed above. Check the “Latest Version” or “Latest Approved Version” location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee’s email list. Others should send comments to the Technical Committee by using the “Send A Comment” button on the Technical Committee’s web page at http://www.oasis-open.org/committees/sca-bindings/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/sca-bindings/ipr.php).

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/sca-bindings/.
# Table of Contents

1 Introduction.......................................................................................................................... 5  
   1.1 Terminology .................................................................................................................... 5  
   1.2 Normative References...................................................................................................... 5  
   1.3 Non-Normative References .............................................................................................. 5  
2 Operation Selection and Data Binding.................................................................................. 6  
3 JCA Binding............................................................................................................................. 7  
4 Binding Properties.................................................................................................................. 10  
5 Additional Binding Configuration Data .................................................................................. 12  
6 Examples.................................................................................................................................. 13  
   6.1 Minimal JCA Binding ....................................................................................................... 13  
   6.2 Existing resources ........................................................................................................... 13  
   6.3 Resource Creation .......................................................................................................... 13  
   6.4 Existing Resources specified in the definition file ......................................................... 14  
7 Conformance.......................................................................................................................... 15  
   A. JCA Binding Schema ......................................................................................................... 16  
   B. Java EE Connector Architecture ...................................................................................... 20  
      B.1 Introduction ................................................................................................................ 20  
      B.2 Selected JCA CCI Interfaces ....................................................................................... 20  
   C. Acknowledgements .......................................................................................................... 22  
   D. Non-Normative Text ......................................................................................................... 23  
   E. Revision History ............................................................................................................... 24
1 Introduction

This document presents bindings describing access and connectivity to the services provided by the Enterprise Information System (EIS).

This version of the document describes JCA Bindings thus narrowing connectivity down to the connectivity to the EIS system external to the SCA system, based on the J2EE Connector Architecture specification and implemented in Java.

Further specification is necessary to define EIS Bindings between different SCA runtimes within SCA system, for example J2EE and EIS based runtimes.

The JCA Bindings are applicable to the composite’s references and services.

The connection to exchange data with the EIS is characterized by two sets of configuration parameters, the connection and interaction parameters. The former set determines the location of the target system the latter determines characteristics that need to be specified to invoke one specific service available at the endpoint. JCA Binding model captures these parameters as separate sets to allow their reuse and reconfiguration.

1.1 Terminology


[WSDL] WSDL Specification

WSDL 1.1: http://www.w3.org/TR/wsd1

WSDL 2.0: http://www.w3.org/TR/wsd120/

Any SCA runtime that claims to support this binding MUST abide by the requirements of this specification.

Within this specification, the following conformance targets are used:

- XML document elements and attributes, including binding.jms and its children, and bindingType
- The SCA runtime – this refers to the implementation that provides the functionality to support the SCA specifications, including that specific to the JCA binding as well as other SCA capabilities
- JCA objects, including ConnectionFactories and ActivationSpecs
- WSDL documents

1.2 Normative References


[WSDL] WSDL Specification

WSDL 1.1: http://www.w3.org/TR/wsd1

WSDL 2.0: http://www.w3.org/TR/wsd120/
1.3 Non-Normative References

TBD  TBD
2 Operation Selection and Data Binding

The operation selection is a process used to identify the operation of the service that should be invoked.
The data binding is a process of data conversion between EIS and runtime formats.

This specification does not define default behavior for the operation selection nor data binding. This choice had been made because the interfaces describing data exchanged between JCA adapter and its client are specific to a particular adapter and, unlike JMS, cannot be used in a generic manner.

The JCA Binding implementation however, MUST use or provide the operation selection and data binding functionality. To support multiple adapters in a generic fashion, the binding MUST expose a mechanism for selecting or specifying the implementations of data bindings or operation selector. This protocol MUST be followed by this functionality providers. This can be achieved in a variety of ways, for example by providing the metadata information or specific pluggability points. This document does not specify such mechanism, it is left to the binding implementation.
3  JCA Binding

The JCA binding element is defined by the following schema:

```xml
<binding.jca connectionInfo="QName">
    <initialContextFactory="xs:anyURI"/>
    <jndiURL="xs:anyURI"/>
    <name="NCName"/>
    <requires="{1}QName"/>
    <uri="xsd:anyURI"/>
</binding.jca>

<jca.outbound.connection managed="xs:boolean" ?>
    <resourceAdapter name="NMTOKEN" type="NMTOKEN"/>
    <connection name="NMTOKEN" type="NMTOKEN" create="string"/>
    <resAuth>Container|Application</resAuth>?
</jca.outbound.connection>?
</jca.inbound.connection>
<jca.inbound.connection>
    <resourceAdapter name="NMTOKEN" type="NMTOKEN"/>
</jca.inbound.connection>?
</jca.outbound.interaction>
<jca.outbound.interaction>
    <connectionSpec type="NMTOKEN"/>
    <interactionSpec type="NMTOKEN"/>
</jca.outbound.interaction>?
</jca.inbound.interaction>
<jca.inbound.interaction>
    <listener type="NMTOKEN"/>
    <inboundOperation name="NMTOKEN" nativeOperation="NMTOKEN"/>
</jca.inbound.interaction>?
</binding.jca>
```

The `binding.jca` element has the following attributes:

- `/binding.jca/@uri` the binding’s uri attribute allows for the specification of the endpoint. For the reference, it defines the endpoint allowing connecting to the target EIS by providing JNDI name
under which the ConnectionFactory is located. For the service, the uri defines the endpoint to
allow the EIS system to connect to the SCA system by defining the JNDI lookup name of the
ActivationSpec, for example uri="java:comp/env/eis/TRAN_EIS". The the binding’s uri attribute is
mutually exclusive with the connectionInfo element as well as jca.inbound.connection or
jca.outbound.connection elements
- /binding.jca/@connectionInfo identifies the jca.binding element present in the definitions
document and whose child or children (one or more of jca.inbound.connection,
jca.outbound.connection, jca.inbound.interaction, jca.outbound.interaction) are used to define
characteristics of connection and interaction characteristics for this binding.
- /binding.jca/@initialContextFactory – the name of the JNDI initial context factory.
- /binding.jca/@jndiURL – the URL for the JNDI provider.
- /binding.jca/@name - as defined in the SCA Assembly specification in Section 9, “Binding”
- /binding.jca/@requires - as defined in the SCA Assembly specification in Section 9, “Binding”
- /binding.jca/jca.outbound.connection defines the outbound connection characteristics, MUST
NOT be specified for services
- /binding.jca/jca.outbound.connection/@managed attribute that determines whether the
interaction with the EIS system should be performed in the managed or non-managed mode. If
the value is true (default), the JNDI name is used to obtain connection to the EIS and use adapter
in the managed mode. If the value is false, the connection information is used to invoke adapter
in the non-managed mode i.e. by creating instance of the ManagedConnectionFactory and using
it to create Connection. For the full description of the managed and non-managed mode refer to
section 6.9 of [JCA15]
- /binding.jca/jca.outbound.connection/resourceAdapter – specifies name, type and properties
of the Resource Adapter Java bean. There may be a restriction, depending on the deployment
platform, about specifying properties of the RA Java Bean. This element is only valid in the
managed mode.
- /binding.jca/jca.outbound.connection/resourceAdapter/@type – the fully qualified name of
the class implementing the JCA ResourceAdapter interface
- /binding.jca/jca.outbound.connection/resourceAdapter/@name – the optional name that
uniquely identifies the existing instance of the resource adapter.
- /binding.jca/jca.outbound.connection/resourceAdapter/property element contains the subset
of the properties of the Resource Adapter Java Bean that need to be set in order to access
specified EIS service. The full list of Resource Adapter properties can be obtained by
introspecting the Java Bean.
- /binding.jca/jca.outbound.connection/connection element specifies the properties of the
connection factory used to create connections to the required service endpoint.
- /binding.jca/jca.outbound.connection/connection/@type – the fully qualified name of the
class implementing the JCA ManagedConnectionFactory interface
- /binding.jca/jca.outbound.connection/connection/@name – if the create attribute is never, the
name uniquely identifies existing instance of the managed connection factory. If create attribute is
always, the name MUST be unique within domain.
- /binding.jca/jca.outbound.connection/connection/property element contains the subset of the
properties of the Managed Connectoin Factory Java Bean that need to be set in order to access
specified EIS service. The full list of Managed Connectoin Factory properties can be obtained by
introspecting the Java Bean.
- /binding.jca/jca.outbound.connection/connection/@create attribute indicates whether the
element containing the attribute should be created when the containing composite is deployed.
Valid values are “always”, “never” and “ifnotexist”. The default value is “ifnotexist”. It is an error if
the attribute value is “always” and the element with the given name already exists.
- /binding.jca/jca.outbound.connection/connection/resAuth element specifies the
authentication mechanism used by the resource adapter in the managed environment
- Vendor specific extensions allow to customize the model to support the specific runtime
characteristics, for example pool size or maximum number of connections
• `/binding.jca/jca.outbound.interaction` defines characteristics of the outbound interaction, MUST NOT be specified for the services

• `/binding.jca/jca.outbound.interaction/connectionSpec` identifies the name of the class implementing javax.resource.cci.ConnectionSpec interface and the set of connectionSpec properties to be specified when creating a connection, a client level connection properties e.g. user name or password. The ConnectionSpec object is used in several patterns that justify its definition in the interaction binding.

• `/binding.jca/jca.outbound.interaction/interactionSpec` type specifies the name of the class implementing javax.resource.cci.InteractionSpec interface. The interaction specified outside of all operation applies to all the operations defined

• `/binding.jca/jca.outbound.interaction/operation` element gathers characteristics of one operation of the service, the data bindings of the inbound and outbound arguments as well as interaction type and the properties.

• `/binding.jca/jca.inbound.connection` defines the inbound connection characteristics, MUST NOT be specified for the references

• `/binding.jca/jca.inbound.connection/resourceAdapter`— specifies name, type and properties of the Resource Adapter Java bean. There may be a restriction, depending on the deployment platform, about specifying properties of the RA Java Bean. This element is only valid in the managed mode.

• `/binding.jca/jca.inbound.connection/resourceAdapter@type`— the fully qualified name of the class implementing the ResourceAdapter interface

• `/binding.jca/jca.inbound.connection/resourceAdapter@name`— the optional name that uniquely identifies the existing instance of the resource adapter.

• `/binding.jca/jca.inbound.connection/activationSpec` element specifies the name of the class implementing javax.resource.spi.ActivationSpec interface and its properties.

• `/binding.jca/jca.inbound.connection/activationSpec@type`— the fully qualified name of the class implementing the ActivationSpec interface

• `/binding.jca/jca.inbound.connection/activationSpec@name`— if the create attribute is never, the name uniquely identifies existing instance of the activation spec. If create attribute is always, the name MUST be unique within domain.

• `/binding.jca/jca.inbound.connection/activationSpec@create` attribute indicates whether the element containing the attribute should be created when the containing composite is deployed. Valid values are “always”, “never” and “ifnotexist”. The default value is “ifnotexist”. It is an error if the attribute value is “always” and the element with the given name already exists.

• `/binding.jca/jca.inbound.interaction` defines characteristics of the inbound interaction, MUST NOT be specified for the references

• `/binding.jca/jca.inbound.interaction/listener` type specifies the listener interface supported by this group of interactions. If the listener is not specified, it is assumed to be a listener implementing javax.resource.cci.MessageListener interface from the JCA specification

• `/binding.jca/jca.inbound.interaction/inboundOperation` element maps the name of the EIS event received by ResourceAdapter to the name of the operation of the Service.

Extensibility - the JCA Bindings provide an extensibility mechanism that allows further customization of the bindings with the vendor specific attributes or elements using extensibility element in the schema as follows:

• `<anyAttribute namespace="##any" processContents="lax"/>

• `<any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
4 Binding Properties

The JCA Binding may contain properties necessary to interact with the EIS system, properties that are, however, not related to the service location or type of services available. Such properties SHOULD be configurable but SHOULD NOT require overwriting connection or interaction elements. Examples of such properties are user ID or password.

The binding.jca contains connectionInfo element that specifies the name of the binding.jca element in the definition file.

```xml
<reference name="EISHelloWorldReference">
  <binding.jca uri="java:comp/env/eis/EISMCF"
              connectionInfo="JCA_Services">
    </binding.jca>
</reference>
```

This element may contain the interaction properties, for example properties of the connectionSpec.

```xml
<definitions targetNamespace="http://acme.com"
             xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200712">
  <binding.jca name="JCA_Services">
    <jca.outbound.interaction
        <connectionSpec name="FAConnectionSpec">
        <property name="group">GROUP1</property>
        <property name="userid">SYSAD</property>
        <property name="password">SYSAD</property>
    </connectionSpec>

    <property name="password" source="$password"/>

    </jca.outbound.interaction>
  </binding.jca>
</definitions>
```

In the example above, the connectionSpec element specifies all the properties it overwrites in place and needs to be updated when there is a need to modify any of the properties. This could be inefficient at times and the method of passing properties from the bindings is defined. To get the value from the bindings, the property specifies the source attribute as follows.

```xml
<jca.outbound.interaction
    <connectionSpec
        name="connector.file.outbound.FAConnectionSpec">
        <property name="group">GROUP1</property>
        <property name="userid">SYSAD</property>
        <property name="password" source="$password"/>
    </connectionSpec>
</jca.outbound.interaction>
```

The property value is the specified in the binding element that refers to the element in the definitions file.

```xml
<reference name="JCAHelloWorldReference">
  <binding.jca uri="java:comp/env/eis/MCF"
              connectionInfo="JCA_Services">
    <property name="password">SYSAD</property>
  </binding.jca>
</reference>
```

The properties can also be specified by the composite, in that case the reference or service would contain the source attribute pointing to the property of the composite:
The indirection level of the binding, required even if the property value is specified in the composite prevents introducing hidden dependencies between the composite and definitions file.
5 Additional Binding Configuration Data

SCA runtime implementations may provide additional configuration that is associated with a JCA Binding, for example to overwrite binding properties like user name or password. The specification of such configuration is SCA runtime-specific and is outside of the scope of this document.
6 Examples

6.1 Minimal JCA Binding

The minimal JCA Binding only contains the binding’s uri attribute with JNDI name of the connection factory. It allows to obtain the Connection to execute request against EIS using adapter. Since no interaction properties are specified, it is assumed that Resource Adapter accepts the null values for the invocation methods.

```xml
<!-- JCA reference, connection is configured in JNDI context -->
<reference name="EISHelloWorldReference">
  <binding.jca uri="java:comp/env/eis/EISMCF"/>
</reference>
```

6.2 Existing resources

The sample reference with the JCA Binding, the binding’s uri attribute specifies the existing resource - the JNDI name under which the connection factory object is located. The interaction properties are specified explicitly in the inlined jca.outbound.interaction element.

```xml
<reference name="EISHelloWorldReference">
  <binding.jca uri="java:comp/env/eis/EISMCF">
    <jca.outbound.interaction>
      <connectionSpec name="FAConnectionSpec">
        <property name="userid">SYSAD</property>
      </connectionSpec>
      <interactionSpec name="FAInteractionSpec">
        <operation name="hello">
          <interactionSpec>
            <property name="dir">temp</property>
            <property name="fileMode">read</property>
          </interactionSpec>
        </operation>
      </interactionSpec>
    </jca.outbound.interaction>
  </binding.jca>
</reference>
```

6.3 Resource Creation

The following sample presents the reference with JCA bindings where the connection resources do not exist and need to be created.

```xml
<reference name="JCAHelloWorldReference">
  <binding.jca>
    <jca.outbound.connection managed="true">
      <resourceAdapter name="connector.file.FAResourceAdapter">
        <property name="logDrive">D</property>
      </resourceAdapter>
    </jca.outbound.connection>
  </binding.jca>
</reference>
```
6.4 Existing Resources specified in the definition file

This sample shows the resources specified in the definitions file and referred to by the binding elements.

The definitions file contains the following

```xml
<definitions targetNamespace="http://acme.com"
    xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200712">
  <binding.jca name="JCA_Inbound">
    <jca.inbound.connection>
      <resourceAdapter name="FAResourceAdapter">
        <property name="logDrive">D</property>
      </resourceAdapter>
      <activationSpec name="FAActivationSpec">
        <property name="directory_type">temp</property>
        <property name="drive">C</property>
      </activationSpec>
    </jca.inbound.connection>
  </binding.jca>
</definitions>
```

The service with the JCA Bindings uses the connectionInfo attribute to identify the resources in the definition file

```xml
<service name="JCAHelloWorldService">
  <binding.jca connectionInfo="JCA_Inbound">
    <jca.inbound.interaction>
      <listener>MyInboundListener</listener>
      <inboundOperation name="hello" nativeOperation="TXPN"/>
      <inboundOperation name="bye" nativeOperation="ETXPRN"/>
    </jca.inbound.interaction>
  </binding.jca>
</service>
```
7 Conformance

The XML schema available at the namespace URI, defined by this specification, is considered to be authoritative and takes precedence over the XML Schema defined in the appendix of this document.
A. JCA Binding Schema

<?xml version="1.0" encoding="UTF-8"?>
<!-- (c) Copyright SCA Collaboration 2006, 2007 -->
<schema xmlns="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200712"
xmns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200712"
elementFormDefault="qualified">
<include schemaLocation="sca-core.xsd"/>
<complexType name="JCABinding">
<complexContent>
<extension base="sca:Binding">
<sequence>
<element name="jca.outbound.connection" type="sca:JCAOutboundConnection"
minOccurs="0" />
<element name="jca.inbound.connection" type="sca:JCAInboundConnection"
minOccurs="0" />
<element name="jca.outbound.interaction" type="sca:JCAOutboundInteraction"
minOccurs="0" />
<element name="jca.inbound.interaction" type="sca:JCAInboundInteraction"
minOccurs="0" />
<element name="property" type="sca:Property" minOccurs="0"
maxOccurs="unbounded" />
<any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
</sequence>
<attribute name="connectionInfo" type="anyURI" use="optional" />
<attribute name="initialContextFactory" type="anyURI" use="optional"/>
<attribute name="jndiURL" type="anyURI" use="optional"/>
<anyAttribute namespace="##any" processContents="lax" />
</extension>
</complexContent>
</complexType>
<complexType name="JCAOutboundConnection">
<sequence>
<element name="resourceAdapter" type="sca:ResourceAdapter"/>
<complexType name="sca:Connection">
    <sequence>
        <element name="resAuth" type="sca:ResAuth" minOccurs="0" />
        <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
</complexType>

<complexType name="JCAInboundConnection">
    <sequence>
        <element name="resourceAdapter" type="sca:ResourceAdapter" />
        <element name="activationSpec" type="sca:ActivationSpec" />
        <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
</complexType>

<complexType name="JCAOutboundInteraction">
    <sequence>
        <element name="connectionSpec" type="sca:ConnectionSpec" minOccurs="0" />
        <element name="interactionSpec" type="sca:InteractionSpec" minOccurs="0" />
        <element name="operation" type="sca:Operation" minOccurs="0" />
        <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
</complexType>

<complexType name="JCAInboundInteraction">
    <sequence>
        <element name="listener" type="string" minOccurs="0" />
        <element name="inboundOperation" type="sca:InboundOperation" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
</complexType>

<complexType name="ResourceAdapter">
    <sequence>
        <element name="property" type="sca:Property" minOccurs="0" maxOccurs="unbounded" />
        <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
    </sequence>

    <attribute name="name" type="NMTOKEN" use="optional" />
    <attribute name="type" type="NMTOKEN" use="required" />
</complexType>

<complexType name="Connection">
    <sequence>
        <element name="property" type="sca:Property" minOccurs="0" maxOccurs="unbounded" />
        <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
    </sequence>

    <attribute name="name" type="NMTOKEN" use="optional" />
</complexType>
<complexType name="InteractionSpec">
  <attribute name="type" type="NMTOKEN" use="required" />
  <attribute name="create" type="sca:CreateResource" use="optional" default="ifnotexist" />
  <anyAttribute namespace="##any" processContents="lax" />
</complexType>

<complexType name="ActivationSpec">
  <sequence>
    <element name="property" type="sca:Property" minOccurs="0" maxOccurs="unbounded" />
    <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </sequence>
  <attribute name="name" type="NMTOKEN" use="optional" />
  <attribute name="type" type="NMTOKEN" use="required" />
  <attribute name="create" type="sca:CreateResource" use="optional" default="ifnotexist" />
  <anyAttribute namespace="##any" processContents="lax" />
</complexType>

<complexType name="Operation">
  <sequence>
    <element name="interactionSpec" type="sca:InteractionSpec" minOccurs="0" />
    <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </sequence>
  <attribute name="name" type="NMTOKEN" use="required" />
  <attribute name="nativeOperation" type="string" use="required" />
  <anyAttribute namespace="##any" processContents="lax" />
</complexType>

<complexType name="InboundOperation">
  <sequence>
    <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </sequence>
  <attribute name="name" type="NMTOKEN" use="required" />
  <attribute name="nativeOperation" type="string" use="required" />
  <anyAttribute namespace="##any" processContents="lax" />
</complexType>

<complexType name="ConnectionSpec">
  <sequence>
    <element name="property" type="sca:Property" minOccurs="0" maxOccurs="unbounded" />
    <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </sequence>
  <attribute name="type" type="NMTOKEN" use="required" />
  <anyAttribute namespace="##any" processContents="lax" />
</complexType>

<complexType name="InteractionSpec">
  <sequence>
    <element name="property" type="sca:Property" minOccurs="0" maxOccurs="unbounded" />
    <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </sequence>
  <attribute name="type" type="NMTOKEN" use="required" />
  <anyAttribute namespace="##any" processContents="lax" />
</complexType>

<element name="binding.jca" type="sca:JCABinding"
substitutionGroup="sca:binding" />
</schema>
B. Java EE Connector Architecture

B.1 Introduction

The connector architecture specification defines set of contracts that allow interoperability of the resource adapters and application server environments. The specification also defines set of client interfaces that can be optionally supported by the adapter and allow the use of adapter functionality by the application clients. The following figure illustrates the relationships of these interfaces.

---

The SPI defines the following management contracts that give adapter consistent view of the infrastructure provided by the server and give server consistent view of all the adapters thus helping with integration of adapters and servers.

- Lifecycle management allows application server to control the startup of the adapter and notification to allow it to shutdown in an orderly fashion.
- Work management allows the adapter to use the server resources such as threads in an efficient way and allows server to manage system resources appropriately.
- Connection management lets the server control the pooling, reusing and caching of the physical connections to the EIS system thus allowing for better scalability.
- Transactions allow the server to control EIS resource managers and provide application clients with the transactional access to external resources.
- Security contract allow for secure access to the EIS systems with security information configured and provided by the application server.
- Message inflow contract allows Resource Adapter to deliver events initiated by the EIS system to the application component executing on the application server.
- Transaction inflow contract allow the application server to participate and execute in the context of the transaction initiated by the EIS system.

The CCI defines set of interfaces to access EIS functionality, through the resource adapter, from the application client. The CCI also provides access to some of the SPIs for transactions and security management to allow for executions of clients running in the non-managed mode, without the presence of the Application Server.

B.2 Selected JCA CCI Interfaces

Record
public interface javax.resource.cci.Record
    extends java.lang.Cloneable, java.io.Serializable {

    public String getRecordName();
    public void setRecordName(String name);
    public void setRecordShortDescription(String description);
    public String getRecordShortDescription();
    public boolean equals(Object other);
    public int hashCode();
    public Object clone() throws CloneNotSupportedException;
}

Interaction

public interface javax.resource.cci.Interaction {

    public Connection getConnection();
    public void close() throws ResourceException;
    public boolean execute(InteractionSpec ispec,
        Record input, Record output) throws ResourceException;
    public Record execute(InteractionSpec ispec,
        Record input) throws ResourceException;
}

MessageListener

interface javax.resource.cci.MessageListener {

    Record onMessage(Record inputData) throws ResourceException;
}

C. Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

[Participant Name, Affiliation | Individual Member]

[Participant Name, Affiliation | Individual Member]
D. Non-Normative Text
E. Revision History

[optional; should not be included in OASIS Standards]

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2008-01-16</td>
<td>Anish Karmarkar</td>
<td>Applied the OASIS template + related changes to the Submission</td>
</tr>
<tr>
<td>2</td>
<td>2008-08-06</td>
<td>Piotr Przybylski</td>
<td>Updates for consistency with JMS Binding and to resolve the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BINDING-13</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BINDING-14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BINDING-28</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BINDING-30</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BINDING-32</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BINDING-38</td>
</tr>
</tbody>
</table>