



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/>.

Notices

Copyright © OASIS® 2007, 2008. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS", [insert specific trademarked names and abbreviations here] are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

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

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, <http://www.ietf.org/rfc/rfc2119.txt>, IETF RFC 2119, March 1997.

[JCA15] Java Connector Architecture Specification Version 1.5

<http://java.sun.com/j2ee/connector/>

[WSDL] WSDL Specification

WSDL 1.1: <http://www.w3.org/TR/wsdl>

WSDL 2.0: <http://www.w3.org/TR/wsdl20/>

.

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

[RFC2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, <http://www.ietf.org/rfc/rfc2119.txt>, IETF RFC 2119, March 1997.

[JCA15] Java Connector Architecture Specification Version 1.5

<http://java.sun.com/j2ee/connector/>

[WSDL] WSDL Specification

WSDL 1.1: <http://www.w3.org/TR/wsdl>

WSDL 2.0: <http://www.w3.org/TR/wsdl20/>

44 **1.3 Non-Normative References**

45 **TBD TBD**

46 2 Operation Selection and Data Binding

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

49
50 This specification does not define default behavior for the operation selection nor data binding. This
51 choice had been made because the interfaces describing data exchanged between JCA adapter and its
52 client are specific to a particular adapter and, unlike JMS, cannot be used in a generic manner.

53
54 The JCA Binding implementation however, MUST use or provide the operation selection and data binding
55 functionality. To support multiple adapters in a generic fashion, the binding MUST expose a mechanism
56 for selecting or specifying the implementations of data bindings or operation selector. This protocol MUST
57 be followed by this functionality providers. This can be achieved in a variety of ways, for example by
58 providing the metadata information or specific pluggability points. This document does not specify such
59 mechanism, it is left to the binding implementation.

60
61
62

3 JCA Binding

The JCA binding element is defined by the following schema:

```
<binding.jca connectionInfo="QName"?
  initialContextFactory="xs:anyURI"?
  jndiURL="xs:anyURI"?
  name="NCName"?
  requires="listofQName"?
  uri="xsd:anyURI"?>

  <jca.outbound.connection managed="xs:boolean" ?>
    <resourceAdapter name="NMTOKEN" type="NMTOKEN">
      <property name="NMTOKEN" type="NMTOKEN">*
    </resourceAdapter>?
    <connection name="NMTOKEN"? type="NMTOKEN" create="string"?>
      <property name="NMTOKEN" type="NMTOKEN">*
    </connection>
    <resAuth>Container|Application</resAuth>?
    <!-- Vendor specific extensions -->
  </jca.outbound.connection>?

  <jca.inbound.connection>
    <resourceAdapter name="NMTOKEN"? type="NMTOKEN">
      <property name="NMTOKEN" type="NMTOKEN">*
    </resourceAdapter>
    <activationSpec name="NMTOKEN"? type="NMTOKEN" create="string"?>
      <property name="NMTOKEN" type="NMTOKEN">*
    </activationSpec>
    <!-- Vendor specific extensions -->
  </jca.inbound.connection>?

  <jca.outbound.interaction>
    <connectionSpec type="NMTOKEN">
      <property name="NMTOKEN" type="NMTOKEN">*
    </connectionSpec>?
    <interactionSpec type="NMTOKEN">
      <property name="NMTOKEN" type="NMTOKEN">*
    </interactionSpec>?
    <operation name="NMTOKEN">
      <interactionSpec type="NMTOKEN"?>
        <property name="NMTOKEN" type="NMTOKEN">*
      </interactionSpec>?
    </operation>*
    <!-- Vendor specific extensions -->
  </jca.outbound.interaction>?

  <jca.inbound.interaction>
    <listener type="NMTOKEN" />?
    <inboundOperation name="NMTOKEN" nativeOperation="NMTOKEN">*
    </inboundOperation>
  </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

- 117 under which the ConnectionFactory is located. For the service, the uri defines the endpoint to
118 allow the EIS system to connect to the SCA system by defining the JNDI lookup name of the
119 ActivationSpec, for example `uri="java:comp/env/eis/TRAN_EIS"`. The the binding's uri attribute is
120 mutually exclusive with the connectionInfo element as well as `jca.inbound.connection` or
121 `jca.outbound.connection` elements
- 122 • **/binding.jca/@connectionInfo** identifies the `jca.binding` element present in the definitions
123 document and whose child or children (one or more of `jca.inbound.connection`,
124 `jca.outbound.connection`, `jca.inbound.interaction`, `jca.outbound.interaction`) are used to define
125 characteristics of connection and interaction characteristics for this binding.
 - 126 • **/binding.jca/@initialContextFactory** – the name of the JNDI initial context factory.
 - 127 • **/binding.jca/@jndiURL** – the URL for the JNDI provider.
 - 128 • **/binding.jca/@name** - as defined in the SCA Assembly specification in Section 9, "Binding"
 - 129 • **/binding.jca/@requires** - as defined in the SCA Assembly specification in Section 9, "Binding"
 - 130 • **/binding.jca/jca.outbound.connection** defines the outbound connection characteristics, MUST
131 NOT be specified for services
 - 132 • **/binding.jca/jca.outbound.connection/@managed** attribute that determines whether the
133 interaction with the EIS system should be performed in the managed or non-managed mode. If
134 the value is true (default), the JNDI name is used to obtain connection to the EIS and use adapter
135 in the managed mode. If the value is false, the connection information is used to invoke adapter
136 in the non-managed mode i.e. by creating instance of the ManagedConnectionFactory and using
137 it to create Connection. For the full description of the managed and non-managed mode refer to
138 section 6.9 of [JCA15]
 - 139 • **/binding.jca/jca.outbound.connection/resourceAdapter** – specifies name, type and properties
140 of the Resource Adapter Java bean. There may be a restriction, depending on the deployment
141 platform, about specifying properties of the RA Java Bean. This element is only valid in the
142 managed mode.
 - 143 • **/binding.jca/jca.outbound.connection/resourceAdapter/@type** – the fully qualified name of
144 the class implementing the JCA ResourceAdapter interface
 - 145 • **/binding.jca/jca.outbound.connection/resourceAdapter/@name** – the optional name that
146 uniquely identifies the existing instance of the resource adapter.
 - 147 • **/binding.jca/jca.outbound.connection/resourceAdapter/property** element contains the subset
148 of the properties of the Resource Adapter Java Bean that need to be set in order to access
149 specified EIS service. The full list of Resource Adapter properties can be obtained by
150 introspecting the Java Bean.
 - 151 • **/binding.jca/jca.outbound.connection/connection** element specifies the properties of the
152 connection factory used to create connections to the required service endpoint.
 - 153 • **/binding.jca/jca.outbound.connection/connection/@type** – the fully qualified name of the
154 class implementing the JCA ManagedConnectionFactory interface
 - 155 • **/binding.jca/jca.outbound.connection/connection/@name** – if the create attribute is never, the
156 name uniquely identifies existing instance of the managed connection factory. If create attribute is
157 always, the name MUST be unique within domain.
 - 158 • **/binding.jca/jca.outbound.connection/connection/property** element contains the subset of the
159 properties of the Managed Connectoin Factory Java Bean that need to be set in order to access
160 specified EIS service. The full list of Managed Connectoin Factory properties can be obtained by
161 introspecting the Java Bean.
 - 162 • **/binding.jca/jca.outbound.connection/connection/@create** attribute indicates whether the
163 element containing the attribute should be created when the containing composite is deployed.
164 Valid values are "always", "never" and "ifnotexist". The default value is "ifnotexist". It is an error if
165 the attribute value is "always" and the element with the given name already exists.
 - 166 • **/binding.jca/jca.outbound.connection/connection/resAuth** element specifies the
167 authentication mechanism used by the resource adapter in the managed environment
 - 168 • Vendor specific extensions allow to customize the model to support the specific runtime
169 characteristics, for example pool size or maximum number of connections

- 170 • **/binding.jca/jca.outbound.interaction** defines characteristics of the outbound interaction, MUST
- 171 NOT be specified for the services
- 172 • **/binding.jca/jca.outbound.interaction/connectionSpec** identifies the name of the class
- 173 implementing javax.resource.cci.ConnectionSpec interface and the set of connectionSpec
- 174 properties to be specified when creating a connection, a client level connection properties e.g.
- 175 user name or password. The ConnectionSpec object is used in several patterns that justify its
- 176 definition in the interaction binding.
- 177 • **/binding.jca/jca.outbound.interaction/interactionSpec** type specifies the name of the class
- 178 implementing javax.resource.cci.InteractionSpec interface. The interaction specified outside of all
- 179 operation applies to all the operations defined
- 180 • **/binding.jca/jca.outbound.interaction/operation** element gathers characteristics of one
- 181 operation of the service, the data bindings of the inbound and outbound arguments as well as
- 182 interaction type and the properties.
- 183 • **/binding.jca/jca.inbound.connection** defines the inbound connection characteristics, MUST
- 184 NOT be specified for the references
- 185 • **/binding.jca/jca.inbound.connection/resourceAdapter** – specifies name, type and properties
- 186 of the Resource Adapter Java bean. There may be a restriction, depending on the deployment
- 187 platform, about specifying properties of the RA Java Bean. This element is only valid in the
- 188 managed mode.
- 189 • **/binding.jca/jca.inbound.connection/resourceAdapter@type** – the fully qualified name of the
- 190 class implementing the ResourceAdapter interface
- 191 • **/binding.jca/jca.inbound.connection/resourceAdapter/@name** – the optional name that
- 192 uniquely identifies the existing instance of the resource adapter.
- 193 • **/binding.jca/jca.inbound.connection/activationSpec** element specifies the name of the class
- 194 implementing javax.resource.spi.ActivationSpec interface and its properties.
- 195 • **/binding.jca/jca.inbound.connection/activationSpec/@type** – the fully qualified name of the
- 196 class implementing the ActivationSpec interface
- 197 • **/binding.jca/jca.inbound.connection/activationSpec/@name** – if the create attribute is never,
- 198 the name uniquely identifies existing instance of the activation spec. If create attribute is always,
- 199 the name MUST be unique within domain.
- 200 • **/binding.jca/jca.inbound.connection/activationSpec/@create** attribute indicates whether the
- 201 element containing the attribute should be created when the containing composite is deployed.
- 202 Valid values are “always”, “never” and “ifnotexist”. The default value is “ifnotexist”. It is an error if
- 203 the attribute value is “always” and the element with the given name already exists.
- 204 • **/binding.jca/jca.inbound.interaction** defines characteristics of the inbound interaction, MUST
- 205 NOT be specified for the references
- 206 • **/binding.jca/jca.inbound.interaction/listener** type specifies the listener interface supported by
- 207 this group of interactions. If the listener is not specified, it is assumed to be a listener
- 208 implementing javax.resource.cci.MessageListener interface from the JCA specification
- 209 • **/binding.jca/jca.inbound.interaction/inboundOperation** element maps the name of the EIS
- 210 event received by ResourceAdapter to the name of the operation of the Service.
- 211 **Extensibility** - the JCA Bindings provide an extensibility mechanism that allows further customization of
- 212 the bindings with the vendor specific attributes or elements using extensibility element in the schema as
- 213 follows:
- 214 • `<anyAttribute namespace="##any" processContents="lax" />`
- 215 • `<any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>`

216

4 Binding Properties

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

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

223

```
224 <reference name="EISHelloWorldReference">  
225   <binding.jca uri=" java:comp/env/eis/EISMCF"  
226     connectionInfo="JCA_Services">  
227     </binding.jca>  
228 </reference>
```

229

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

```
231 <definitions targetNamespace="http://acme.com"  
232   xmlns="http://docs.oasis-open.org/ns/opencsa/sca/200712">  
233   <binding.jca name="JCA_Services">  
234     <jca.outbound.interaction >  
235       <connectionSpec name="FAConnectionSpec">  
236         <property name="group">GROUP1</property>  
237         <property name="userid">SYSAD</property>  
238         <property name="password">SYSAD</property>  
239       </connectionSpec>  
240       ...  
241     </jca.outbound.interaction>  
242   </binding.jca>  
243 </definitions>
```

244

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

249

```
250 <jca.outbound.interaction >  
251   <connectionSpec  
252     name="connector.file.outbound.FAConnectionSpec">  
253     <property name="group">GROUP1</property>  
254     <property name="userid">SYSAD</property>  
255     <property name="password" source="$password"/>  
256   </connectionSpec>  
257 </jca.outbound.interaction>
```

258

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

260

```
261 <reference name="JCAHelloWorldReference">  
262   <binding.jca uri=" java:comp/env/eis/MCF "  
263     connectionInfo="JCA_Services">  
264     <property name="password">SYSAD</property>  
265   </binding.jca>  
266 </reference>
```

267

268 The properties can also be specified by the composite, in that case the reference or service would contain
269 the source attribute pointing to the property of the composite:

270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286

```
<composite xmlns="http://docs.oasis-pen.org/ns/opencsa/sca/200712"
  name="EISHelloworld">
  <reference name="EISHelloWorldReference">
    <binding.jca uri=" java:comp/env/eis/EISMCF"
      connectionInfo="JCA_Services">
      <property name="userid" source="$UID"/>
    </binding.jca>
  </reference>

  <property name="UID">SYSAD</property>
</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.

287

5 Additional Binding Configuration Data

288

289

290

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.

291 6 Examples

292 6.1 Minimal JCA Binding

293 The minimal JCA Binding only contains the binding's uri attribute with JNDI name of the connection
294 factory. It allows to obtain the Connection to execute request against EIS using adapter. Since no
295 interaction properties are specified, it is assumed that Resource Adapter accepts the null values for the
296 invocation methods.
297
298

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

303

304 6.2 Existing resources

305 The sample reference with the JCA Binding, the binding's uri attribute specifies the existing resource - the
306 JNDI name under which the connection factory object is located. The interaction properties are specified
307 explicitly in the inlined jca.outbound.interaction element.
308

```
309 <reference name="EISHelloWorldReference">  
310   <binding.jca uri="java:comp/env/eis/EISMCF">  
311     <jca.outbound.interaction>  
312  
313       <connectionSpec name="FAConnectionSpec">  
314         <property name="userid">SYSAD</property>  
315       </connectionSpec>  
316       <interactionSpec name="FAInteractionSpec">  
317       </interactionSpec>  
318  
319       <operation name="hello">  
320         <interactionSpec>  
321           <property name="dir">temp</property>  
322           <property name="fileMode">read</property>  
323         </interactionSpec>  
324       </operation>  
325     </jca.outbound.interaction>  
326   </binding.jca>  
327 </reference>
```

329

330 6.3 Resource Creation

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

```
333 <reference name="JCAHelloWorldReference">  
334   <binding.jca>  
335     <jca.outbound.connection managed="true">  
336       <resourceAdapter  
337         name="connector.file.FAResourceAdapter">  
338         <property name="logDrive">D</property>  
339       </resourceAdapter>
```

340
341
342
343
344
345
346
347
348

```
<connection name="FAManagedConnectionFactory"
  create="always">
  <property name="host">localhost</property>
  <property name="drive">C</property>
</connection>
</jca.outbound.connection>
</binding.jca>
</reference>
```

349 6.4 Existing Resources specified in the definition file

350 This sample shows the resources specified in the definitions file and referred to by the binding elements.
351 The definitions file contains the following

352
353
354
355
356
357
358
359
360
361
362
363
364
365
366

```
<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>
```

367

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

370
371
372
373
374
375
376
377
378
379

```
<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>
```

380

7 Conformance

381

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.

382

383

A. JCA Binding Schema

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

```

<?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"
  xmlns: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>
  <simpleType name="CreateResource">
    <restriction base="string">
      <enumeration value="always" />
      <enumeration value="never" />
      <enumeration value="ifnotexist" />
    </restriction>
  </simpleType>
  <simpleType name="ResAuth">
    <restriction base="string">
      <enumeration value="Container" />
      <enumeration value="Application" />
    </restriction>
  </simpleType>
  <complexType name="JCAOutboundConnection">
    <sequence>
      <element name="resourceAdapter" type="sca:ResourceAdapter"

```

```

438         minOccurs="0" />
439         <element name="connection" type="sca:Connection" />
440         <element name="resAuth" type="sca:ResAuth" minOccurs="0" />
441         <any namespace="##other"
442             processContents="lax" minOccurs="0"
443             maxOccurs="unbounded" />
444     </sequence>
445     <attribute name="managed" type="boolean" use="optional"
446         default="true" />
447     <anyAttribute namespace="##any" processContents="lax" />
448 </complexType>
449 <complexType name="JCAInboundConnection">
450     <sequence>
451         <element name="resourceAdapter" type="sca:ResourceAdapter" />
452         <element name="activationSpec" type="sca:ActivationSpec" />
453         <any namespace="##other" processContents="lax" minOccurs="0"
454             maxOccurs="unbounded" />
455     </sequence>
456 </complexType>
457 <complexType name="JCAOutboundInteraction">
458     <sequence>
459         <element name="connectionSpec" type="sca:ConnectionSpec"
460             minOccurs="0" />
461         <element name="interactionSpec" type="sca:InteractionSpec"
462             minOccurs="0" />
463         <element name="operation" type="sca:Operation" minOccurs="0" />
464         <any namespace="##other" processContents="lax" minOccurs="0"
465             maxOccurs="unbounded" />
466     </sequence>
467 </complexType>
468 <complexType name="JCAInboundInteraction">
469     <sequence>
470         <element name="listener" type="string" minOccurs="0" />
471         <element name="inboundOperation" type="sca:InboundOperation"
472             minOccurs="0" maxOccurs="unbounded" />
473         <any namespace="##other" processContents="lax" minOccurs="0"
474             maxOccurs="unbounded" />
475     </sequence>
476 </complexType>
477 <complexType name="ResourceAdapter">
478     <sequence>
479         <element name="property" type="sca:Property" minOccurs="0"
480             maxOccurs="unbounded" />
481         <any namespace="##other" processContents="lax" minOccurs="0"
482             maxOccurs="unbounded" />
483     </sequence>
484     <attribute name="name" type="NMTOKEN" use="optional" />
485     <attribute name="type" type="NMTOKEN" use="required" />
486     <anyAttribute namespace="##any" processContents="lax" />
487 </complexType>
488 <complexType name="Connection">
489     <sequence>
490         <element name="property" type="sca:Property" minOccurs="0"
491             maxOccurs="unbounded" />
492         <any namespace="##other" processContents="lax" minOccurs="0"
493             maxOccurs="unbounded" />
494     </sequence>
495     <attribute name="name" type="NMTOKEN" use="optional" />

```

```

496     <attribute name="type" type="NMTOKEN" use="required" />
497     <attribute name="create" type="sca:CreateResource" use="optional"
498         default="ifnotexist" />
499     <anyAttribute namespace="##any" processContents="lax" />
500 </complexType>
501 <complexType name="ActivationSpec">
502     <sequence>
503         <element name="property" type="sca:Property" minOccurs="0"
504             maxOccurs="unbounded" />
505         <any namespace="##other" processContents="lax" minOccurs="0"
506             maxOccurs="unbounded" />
507     </sequence>
508     <attribute name="name" type="NMTOKEN" use="optional" />
509     <attribute name="type" type="NMTOKEN" use="required" />
510     <attribute name="create" type="sca:CreateResource" use="optional"
511         default="ifnotexist" />
512     <anyAttribute namespace="##any" processContents="lax" />
513 </complexType>
514 <complexType name="Operation">
515     <sequence>
516         <element name="interactionSpec" type="sca:InteractionSpec"
517             minOccurs="0" />
518         <any namespace="##other" processContents="lax" minOccurs="0"
519             maxOccurs="unbounded" />
520     </sequence>
521     <attribute name="name" type="NMTOKEN" use="required" />
522     <anyAttribute namespace="##any" processContents="lax" />
523 </complexType>
524 <complexType name="InboundOperation">
525     <sequence>
526         <any namespace="##other" processContents="lax" minOccurs="0"
527             maxOccurs="unbounded" />
528     </sequence>
529     <attribute name="name" type="NMTOKEN" use="required" />
530     <attribute name="nativeOperation" type="string" use="required" />
531     <anyAttribute namespace="##any" processContents="lax" />
532 </complexType>
533 <complexType name="ConnectionSpec">
534     <sequence>
535         <element name="property" type="sca:Property" minOccurs="0"
536             maxOccurs="unbounded" />
537         <any namespace="##other" processContents="lax" minOccurs="0"
538             maxOccurs="unbounded" />
539     </sequence>
540     <attribute name="type" type="NMTOKEN" use="required" />
541     <anyAttribute namespace="##any" processContents="lax" />
542 </complexType>
543 <complexType name="InteractionSpec">
544     <sequence>
545         <element name="property" type="sca:Property" minOccurs="0"
546             maxOccurs="unbounded" />
547         <any namespace="##other" processContents="lax" minOccurs="0"
548             maxOccurs="unbounded" />
549     </sequence>
550     <attribute name="type" type="NMTOKEN" use="required" />
551     <anyAttribute namespace="##any" processContents="lax" />
552 </complexType>
553 <element name="binding.jca" type="sca:JCABinding"

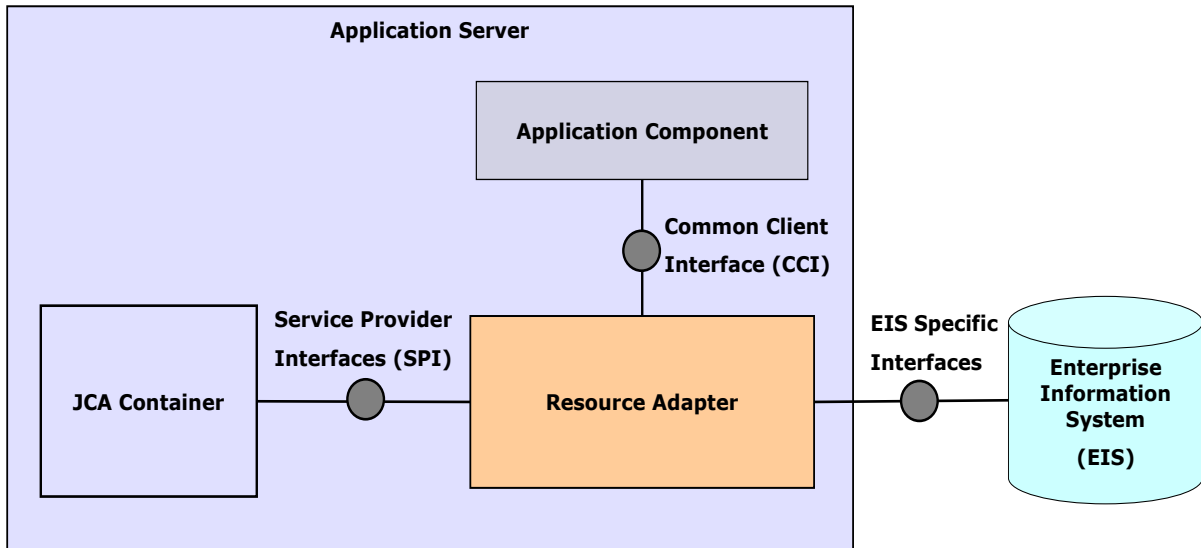
```

```
554 substitutionGroup="sca:binding" />
555 </schema>
556
```

557 B. Java EE Connector Architecture

558 B.1 Introduction

559 The connector architecture specification defines set of contracts that allow interoperability of the resource
560 adapters and application server environments. The specification also defines set of client interfaces that
561 can be optionally supported by the adapter and allow the use of adapter functionality by the application
562 clients. The following figure illustrates the relationships of these interfaces.
563



564
565 The SPI defines the following management contracts that give adapter consistent view of the
566 infrastructure provided by the server and give sever consistent view of all the adapters thus helping with
567 integration of adapters and servers.

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

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

586 B.2 Selected JCA CCI Interfaces

587 Record

```
588     public interface javax.resource.cci.Record
589         extends java.lang.Cloneable, java.io.Serializable {
590
591     public String getRecordName();
592         public void setRecordName(String name);
593     public void setRecordShortDescription(String description);
594     public String getRecordShortDescription();
595         public boolean equals(Object other);
596         public int hashCode();
597         public Object clone() throws CloneNotSupportedException;
598     }
```

599

600 **Interaction**

```
601
602     public interface javax.resource.cci.Interaction {
603
604         public Connection getConnection();
605         public void close() throws ResourceException;
606         public boolean execute(InteractionSpec ispec,
607             Record input, Record output) throws ResourceException;
608         public Record execute(InteractionSpec ispec,
609             Record input) throws ResourceException;
610
611     }
```

612 **MessageListener**

```
613
614     interface javax.resource.cci.MessageListener {
615
616         Record onMessage(Record inputData) throws ResourceException;
617     }
```

618

619

620 **C. Acknowledgements**

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

623 **Participants:**

624 [Participant Name, Affiliation | Individual Member]

625 [Participant Name, Affiliation | Individual Member]

626

628

E. Revision History

629

[optional; should not be included in OASIS Standards]

630

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

631

632

633