Service Component Architecture EJB Session Bean Binding Specification
Version 1.1

Working Draft 04 + issue 107v3

29 July 2009

Specification URIs:
This Version:
http://docs.oasis-open.org/sca-j/sca-ejbbinding-draft-20070926.html
http://docs.oasis-open.org/sca-j/sca-ejbbinding-draft-20070926.doc

Previous Version:

Latest Version:
http://docs.oasis-open.org/sca-j/sca-ejbbinding-draft-20070926.html
http://docs.oasis-open.org/sca-j/sca-ejbbinding-draft-20070926.doc

Latest Approved Version:

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

Chair(s):
David Booz, IBM
Mark Combellack, Avaya

Editor(s):
David Booz, IBM
Anish Karmarkar, Oracle

Related work:
This specification replaces or supercedes:

• Service Component Architecture EJB Session Bean Binding Specification Version 1.00,
  February 22 2007

This specification is related to:

• Service Component Architecture Assembly Model Specification Version 1.1
• Service Component Architecture Policy Framework Version 1.1

Declared XML Namespace(s):
http://docs.oasis-open.org/ns/opencsa/sca/200903

Abstract:
This document explains the SCA EJB session bean binding. It describes how to integrate a
previously deployed session bean into an SCA assembly, and how to expose SCA services to
clients which use the EJB programming model.
Status:

This document was last revised or approved by the OASIS Service Component Architecture / J (SCA-J) 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-j/.

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-j/ipr.php).

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

1 Introduction ................................................................................................................... 5
   1.1 Terminology .............................................................................................................. 6
   1.2 Normative References .............................................................................................. 6
   1.3 Non-Normative References ....................................................................................... 6
2 Session bean binding schema .......................................................................................... 7
3 Interface Mapping ........................................................................................................... 10
   3.1 EJBOBJECT and EJBLocalObject Interfaces ............................................................... 10
4 Reference Binding .......................................................................................................... 11
   4.1 Exception Handling .................................................................................................... 11
5 Packaging ....................................................................................................................... 12
6 Service Binding .............................................................................................................. 13
   6.1 Handling methods from EJBOBJECT and EJBLocalObject ............................................ 14
A. Use cases ....................................................................................................................... 17
   A.1 Consuming an Existing EJB SOA Service ................................................................. 17
   A.2 Exposing an SCA Service with an EJB SCA Binding .................................................. 17
   A.3 Consuming Existing Local EJB SOA Services ............................................................ 18
   A.4 Exposing an SCA Service with a Local SLSB SCA Binding ......................................... 19
   A.5 Consuming an EJB Service inside a Java EE EAR file .............................................. 20
   A.6 Exposing an SCA Service inside a Java EE EAR file ................................................ 21
B. EJB binding schema ...................................................................................................... 23
C. Acknowledgements ........................................................................................................ 26
D. Non-Normative Text ...................................................................................................... 28
E. Revision History ............................................................................................................. 29
# 1 Introduction

EJB session beans are a common technology used to implement business services. The ability to integrate SCA with session bean based services is useful because it preserves the investment incurred during the creation of those business services, while enabling the enterprise to embrace the newer SCA technology in incremental steps. The simplest form of integration is to simply enable SCA components to invoke session beans as SCA services. There is also a need to expose SCA services such that they are consumable by programmers skilled in the EJB programming model. This enables existing session bean assets to be enhanced to exploit newly deployed SCA services without the EJB programmers having to learn a new programming model.

This document explains the EJB SCA binding. This proposal describes how to integrate a previously deployed stateless session bean into an SCA assembly, and how to expose SCA services to clients which use the EJB programming model.

The EJB Session Bean binding enables:

- SCA developers to treat previously deployed stateless session beans as SCA services, by wiring them into an SCA assembly (SCA reference).
- SCA service deployers to expose a SCA service as a stateless session bean for consumption by Java EE applications.

Stateful session beans are out of scope for this specification. The terms ‘session bean’ and ‘stateless session bean’ are interchangeable for the purpose of this specification.

The use of EJBs and EJB modules as SCA component implementations is beyond the scope of this specification and is described in the Java EE integration specification [SCAJEE]. The following diagram shows the use of the EJB SCA binding on both services and references.

![Figure 1: EJB Binding used on Services and References](image-url)
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].

1.2 Normative References


1.3 Non-Normative References

TBD
2 Session bean binding schema

The EJB session bean binding element is defined by the following pseudo-schema.

```xml
.binding.ejb
    homeInterface="NCName"?
    ejb-link-name="string"?
    ejb-version="EJB2 or EJB3"?
    name="NCName"?
    policySets="sca:listOfQNames"?
    requires="sca:listOfQNames"?
    uri="anyURI"?

    <wireFormat ... />?

    <operationSelector ... />?
</binding.ejb>
```

- **/binding.ejb/@homeInterface : NCName (0..1)** - The homeInterface attribute of the EJB binding is the session bean's home interface, and is used when exposing SCA services as EJB 2.x session beans. For `<binding.ejb/>`, if @ejb-version="EJB2", then `@homeInterface` MUST be specified and MUST have a value that is the fully qualified package name of the Java interface class of the EJB's home interface. [BSB20001]

- **/binding.ejb/@ejb-link-name : string (0..1)** - The ejb-link-name attribute provides a means for integrating EJB reference resolution with SCA. When used on a binding for a reference, it allows a SCA client to bind to an EJB that is packaged in the same Java EE EAR file as the SCA client. When used on a service binding, it exposes an `<ejb-link/>` target for Java EE clients that want to use Java EE assembly to wire to the SCA service. This attribute is functionally equivalent to using the `<ejb-link/>` subelement of the `<ejb-ref/>` element in an EJB deployment descriptor. The value of this attribute is supplied by an application assembler, and is in the form as specified by the Java EE specification [SCAJEE] (i.e. `<jar-name>#<ejb-name>`)

- **/binding.ejb/@ejb-version : VersionValue (0..1)** - The ejb-version attribute is used to indicate the EJB client view exposed by the EJB binding when used on an SCA service. This attribute has no meaning when used on a reference. The value ‘EJB2’ indicates the desire to expose an EJB 2.x client view. The value ‘EJB3’ indicates the desire to expose an EJB 3.0 client view. The default value is ‘EJB3’. When `<binding.ejb/>` applies to a service and the @ejb-version attribute is set to...
'EJB2', the SCA Runtime MUST support invocation of the service using the EJB 2.x client view as specified in the Java EE specification [SCA/EE]. [BSB20004] The value of the @uri attribute MUST take the form of an Object URL as specified in the CORBA Services specification [CORBA]. [BSB20005]

The value of the @uri attribute MUST take the form of an Object URL as specified in the CORBA Services specification [CORBA]. [BSB20005]

• /binding.ejb/@name : NCName (0..1) – As defined in the SCA Assembly Specification [ASSEMBLY]

• /binding.ejb/@requires : QName (0..1) – A list of policy intents as defined in the SCA Policy Framework Specification [POLICY]

• /binding.ejb/@policySets : QName (0..1) – A list of policy sets as defined in the SCA Policy Framework Specification [POLICY]

The base SCA binding schema provides an attribute called uri, that is used to denote the URI of an endpoint. In the context of the SCA EJB binding, the uri attribute is defined as follows:

• /binding.ejb/@uri : anyURI (0..1) – Specifies the URI of a session bean endpoint. For EJB 2.x, this is the endpoint of the session home. For EJB 3.x, this is the endpoint of the session bean. The value of the @uri attribute MUST take the form of an Object URL as specified in the CORBA Services specification [CORBA]. [BSB20006] This is a standard URI form for referring to remotable CORBA objects. Briefly, the corbaname URI format looks like this:

  o corbaname:iip:<hostName>:<port>/<key string>#<path to home>

Typically, a corbaname URI doesn’t include all these components. The following example shows a corbaname URI that uses the default ORB configuration to find an EJB home at ejb/MyHome in the JNDI directory:

  o corbaname:rir:#ejb/MyHome

Other forms of URI specification are admissible when interoperability is of no concern.

• /binding.ejb/wireFormat – As defined in the SCA Assembly Specification [ASSEMBLY]. This specification does not define any new wireFormat elements.

• /binding.ejb/operationSelector – As defined in the SCA Assembly Specification [ASSEMBLY]. This specification does not define any new operationSelector elements.

When <binding.ejb/> applies to an SCA reference, the @uri and @ejb-link-name attributes MUST NOT be specified together in the same binding configuration. [BSB20007]

The <binding.ejb/> element MUST conform to the XML schema defined in the sca-binding-ejb.xsd. [BSB20008]
2.1 Additional binding configuration data

SCA runtime implementations can provide additional metadata that is associated with an EJB binding. This is done by providing extension points in the schema; refer to Appendix B: EJB Binding Schema for the locations of these extension points.
3 Interface Mapping

When used with the EJB binding, an SCA runtime MUST ensure that a service or reference interface is compatible with a session bean interface, according to the rules defined in the section "Interface Mapping". [BSB30001]

- The interface used by an SCA reference is remotable if a remote session bean interface is being accessed, and is local if a local session bean interface is being accessed.
- The interface used by an SCA reference can omit any methods inherited from EJBObject or EJBLocalObject that appear in a session bean interface.
- The interface used by an SCA reference is a compatible subset [ASSEMBLY] of the methods in the interface used by a session bean.
  - Compatibility for an individual method is defined by the SCA Assembly Model Specification [ASSEMBLY], and can be stated simply as compatibility of the signature. That is, the method name, input types, output types, and faults are identical.
    - The order of the input and output types is identical.
    - Except for RemoteExceptions, the set of Faults and Exceptions declared by the methods in the SCA reference interface are the same or a superset of those specified by the EJB interface.
- The interface used by a service or reference can be an SCA business interface or an EJB 3.0 remote or local interface.

3.1 EJBObject and EJBLocalObject Interfaces

The interfaces exposed from EJB 2.X beans inherit from either EJBObject or EJBLocalObject. EJBObject and EJBLocalObject contain methods directed toward the management of bean instances, meaning that the exposed 2.X interfaces mix business and infrastructure methods in a way that makes them poorly suited for use as an SCA business interface. However, EJB 2.X beans developed using the "Business Interface Pattern" will already have an interface that is a suitable SCA business interface. In other cases, a suitable interface can be derived from the SessionBean interface. An EJB 2.X session bean interface itself MUST NOT be used as the interface of a reference binding. [BSB30002]

When <binding ejb/> applies to a service and @ejb-version is set to "EJB2", the binding implementation MUST implement the methods from the EJBObject or EJBLocalObject interface. [BSB30003] Section 6.1 describes the behavior associated with each inherited method.
4 Reference Binding

When used on a reference, the EJB binding specifies the means for connecting an SCA component to a previously deployed or co-deployed session bean.

The reference interface used with the EJB binding can be either a remote or local interface. SCA deployment logic and the binding implementation will introspect the reference interface class to determine whether it is local or remote. If an SCA component needs to access both the local and remote interface of a session bean, then this can be modeled in SCA assembly through two references, one with a local interface and one with a remote interface.

The following example shows a reference binding using a corbaname URI:

```
<reference name="CandidateCheck">
  <interface.java interface="com.app.jobbank.CandidateCheck"/>
  <binding.ejb uri="corbaname:rir:#ejb/CandidateCheckHome"/>
</reference>
```

The specific uri would be supplied prior to the completion of deployment.

The following example is a reference binding using an ejb-link.

```
<reference name="CandidateCheck">
  <interface.java interface="com.app.jobbank.CandidateChk"/>
  <binding.ejb ejb-link-name="candidateEJB.jar#CandidateChk"/>
</reference>
```

4.1 Exception Handling

Exception handling for interactions with session beans has been specified in chapter 14 of the EJB 3 specification [EJB] and in Chapter 18 of the EJB 2.1 specification [EJB]. The EJB [EJB] specifications define non-business exceptions that can be thrown to the EJB client. When <binding.ejb/> applies to a reference, the SCA Runtime MUST wrap non-business exceptions in a ServiceRuntimeException that is thrown to the client [JAVACAA]. [BSB40001].

Formatted: Bullets and Numbering

Deleted: The reference binding for session beans can be imagined to consist of two consecutive invocation paths: ¶
SCA business interface to EJB business interface (if different) ¶
EJB Business interface to session bean instance ¶
For the second invocation path, the rules laid out in the EJB specification apply. For the first invocation path, the following rules apply: ¶
any business exception (see [JAVACAA]) will be re-thrown by the binding implementation. ¶
any other exception will be wrapped in a ServiceRuntimeException which will be thrown by the binding implementation.
5 Packaging

There is no requirement to package the session bean home interface or client stubs with an SCA component that uses the Session bean binding. The EJB Session Bean binding implementation can dynamically lookup, create and invoke the bean without the usual EJB client classes.
6 Service Binding

When used on an SCA service, the EJB SCA binding causes the service to be exposed as a session bean. This enables a client that is using the EJB programming model to call the SCA service using its native programming model.

The /binding.ejb/@homeInterface attribute is used to indicate the Session Home interface that an EJB client will use to bootstrap itself with the SCA service, just as it would with any other session bean. When <binding.ejb/> applies to an SCA service, the Java interface class specified on the @homeInterface attribute MUST have one create method [EJB]. [BSB60001]

The following is an example of a service using the EJB binding.

```
<service name="JobBank">
    <interface.java interface="com.app.jobbank.JobBankService"/>
    <binding.ejb
        uri="corbaname:rir:#ejb/JobBankServiceHome"
        homeInterface="com.app.jobbank.JobBankServiceHome"
        ejb-link-name="jobbankEJB.jar#JobBankComponent"/>
</service>
```

A corresponding local home interface `com.app.jobbank.JobBankServiceHome` looks like this:

```
package com.app.jobbank;

import javax.ejb.CreateException;
import javax.ejb.EJBLocalHome;

public interface JobBankServiceHome extends EJBLocalHome {
    JobBankService create() throws CreateException;
}
```

Similarly, the remote home interface can be formulated by extending `javax.ejb.EJBHome` and making sure to declare a RemoteException:

```
package com.app.jobbank;

import java.rmi.RemoteException;
import javax.ejb.CreateException;
import javax.ejb.EJBHome;

public interface JobBankServiceHome extends EJBHome {
    JobBankService create() throws CreateException, RemoteException;
}
```
In the corbaname used in this example, the first part of the URI (up to the #) would logically be supplied by the target deployment environment. See the SCA Assembly Model Specification [ASSEMBLY] for a discussion of base URIs provided by an SCA domain configuration. The remainder of the name would be provided prior to completion of deployment. The example above shows the URI that a client would use after deployment. Prior to deployment, an assembler or developer can specify only the last portion of the URI (i.e. everything following the #).

The service interface used with the EJB binding can be either a remote or local interface. SCA deployment logic and the binding implementation will introspect the interface class to determine whether it is local or remote. If an SCA component needs to be exposed as both a local and remote session bean, this can be modeled in SCA through two services, one with the local interface and one with the remote interface.

When used on a service binding, ejb-link-name and uri are NOT mutually exclusive. They each provide a means for wiring to the SCA service depending on the locality of the client EJB reference. For example, an SCA service packaged with an JEE EJB application could be exposed for consumption by local EJB clients (using the ejb-link-name element) and remote EJB clients (using the uri).

From the perspective of an EJB client (local and remote), SCA services that are exposed as session beans are not distinguishable from ordinary session beans.

Specifically, this means that a local client will be able to reference the SCA service as a session bean using ejb-(local)-ref declarations in the appropriate locations and by issuing JNDI lookups or relying on dependency injection mechanisms. If the service is exposed as EJB 2.x session bean, by virtue of a home interface specification, the client needs to be aware of the EJB 2.x home interface contract.

Similarly remote EJB clients are expected to be able to consume SCA services that are exposed as session beans just as they are able to consume ordinary session beans.

### 6.1 Handling methods from EJBObject and EJBLocalObject

This section describes the SCA specific behavior of the methods that EJB 2.X service bindings inherit from the EJBObject and EJBLocalObject interfaces.

<table>
<thead>
<tr>
<th>Method</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>isIdentical</td>
<td>Tests whether the service component, which the binding exposes, is the same instance as the one exposed by the specified object.</td>
</tr>
<tr>
<td>getEJBHome</td>
<td>Returns an implementation of the interface specified as /binding.ejb/@homelinkInterface. The instance can be used to create or remove bean instances.</td>
</tr>
<tr>
<td>getEJBLocalHome</td>
<td></td>
</tr>
</tbody>
</table>

Deleted: It should be possible for
Deleted: to
Deleted: session bean
Deleted: then
Deleted: should
Deleted: assembly

Deleted: may
7 Callbacks

The SCA Assembly Model Specification [ASSEMBLY] defines the callback feature which enables asynchronous interactions between two SCA components. This specification does not support the callback feature. However, implementations can choose to support the callback feature, in conjunction with this binding, by creating extensions to this specification.
8 Conformance

The XML schema pointed to by the RDDL document at the namespace URI, defined by this specification, are considered to be authoritative and take precedence over the XML schema defined in the appendix of this document.

There are two categories of artifacts for which this specification defines conformance:

a) SCA EJB Session Bean Binding XML Document

b) SCA Runtime

8.1 SCA EJB Session Bean Binding XML Document

An SCA EJB Session Bean Binding XML document is an SCA Composite Document, or an SCA ComponentType Document, as defined by the SCA Assembly Model Specification, that uses the <binding.ejb> element.

8.2 SCA Runtime

An implementation that claims to conform to the requirements of an SCA Runtime defined in this specification has to meet the following conditions:

1. Error! Reference source not found. [BSB80002]

2. Error! Reference source not found. [BSB80003]

Error! Reference source not found. [BSB80004]
A. Use cases

The following use cases provide some examples of the usage of the SCA EJB Session Bean binding.

A.1 Consuming an Existing EJB SOA Service

An SCA service is developed that needs to call a business service which is already deployed and running in a Java EE server. The SCA service will be deployed into an SCA runtime somewhere in the enterprise that is not necessarily a Java EE runtime. The business service was implemented as a session bean. The SCA service defines a reference to the business service, and the deployer attaches an EJB binding to the reference. In this use case, the EJB remote interface is the business interface.

![Diagram of SCA Reference invoking EJB Session Bean](image)

Figure 2: SCA Reference invoking EJB Session Bean

The reference in the deployed sca.composite file looks like this:

```xml
<reference name="CandidateCheck">
  <interface.java interface="com.app.jobbank.CandidateChk"/>
  <binding.ejb uri="corbaname:rir:#ejb/CandidateChkHome"/>
</reference>
```

A.2 Exposing an SCA Service with an EJB SCA Binding

An SCA service is developed that will be called from a Java EE environment. The Java EE programmer doesn't know the SCA programming model and therefore wants to use the Java EE programming model that he knows in order to invoke the SCA service (i.e. new initialContext(), nc.lookup(), etc.). In this case, the SCA service has to be deployed into a runtime that is capable of supporting the EJB binding. Note that deployment of this service can result in the generation and deployment of a session bean, along with its home interface. This aspect is significantly different from the previous use case.
Since the client will use the standard Java EE programming model, the client needs to know the home interface of the SCA service. The service in the SCA composite file will look like this:

```xml
<service name="CompanyInfo">
  <interface.java interface="com.app.jobbank.CompanyInfo"/>
  <binding.ejb uri="corbaname:rir:#ejb/CompanyInfoHome"
    homeInterface="com.app.jobbank.CompanyInfoHome"
    ejb-version="EJB2"/>
  <reference>CompanyInfoComponent/CompanyInfo</reference>
</service>
```

The client code as per the standard Java EE programming model looks like this:

```java
Context initialContext = new InitialContext(env);
CompanyInfoHome companyInfoHome = (CompanyInfoHome) initialContext.lookup("corbaname:rir:#ejb/CompanyInfoHome");
CompanyInfo companyInfo = companyInfoHome.create();
companyInfo.getCompanyInfo("ACME Corp");
```

A.3 Consuming Existing Local EJB SOA Services
This use case is similar to the use case in section A.1, except that the SCA service is going to be deployed into a Java EE capable JVM, and it is the same JVM as the EJB service. In this use case, the EJB’s local interface is used as the business interface.

Note that the SCA client could also use the EJB remote interface. If an SCA component wanted to access both the local and remote interface, then it would declare 2 references (one with the local interface, one with the remote interface).

---

A.4 Exposing an SCA Service with a Local SLSB SCA Binding

This use case is similar to the use case in section A.2, except that the SCA service is going to be deployed into the same JVM as the client. This use case allows for the possibility that the SCA service is exposed as a local EJB interface. Note that deployment of this service will effectively result in the generation and deployment of a session bean with a local interface and a local home interface.

---

Figure 4: SCA reference consuming a Local EJB service

The example below shows the usage of a local interface in the reference definition.

```xml
<reference name="CandidateCheck">
  <interface.java interface="com.app.jobbank.CandidateCheckLocal"/>
  <binding.ejb uri="corbaname:rir:#ejb/CandidateCheckHome"/>
</reference>
```
The following is an example:

```
<service name="CompanyInfo">
  <interface.java interface="com.app.jobbank.CompanyInfoLocal"/>
  <binding.ejb uri="corbaname:rir#ejb/CompanyInfoHome" homeInterface="com.app.jobbank.CompanyInfoLocalHome"/>
  <reference>CompanyInfoComponent/CompanyInfo</reference>
</service>
```

### A.5 Consuming an EJB Service inside a Java EE EAR file

This use case is similar to sections A.1 and A.3, except that the SCA service is going to be packaged inside a Java EE EAR file. By packaging it in this way, the SCA reference binding can be configured as if it were an `<ejb-ref>` with the `<ejb-link>` subelement.

The following is an example of the SCA reference binding.

```
<reference name="CandidateCheck">
  <interface.java interface="com.app.jobbank.CandidateChk"/>
  <binding.ejb ejb-link-name="candidateEJB.jar#CandidateChk"/>
</reference>
```

The following is an `<ejb-ref/>` snippet that is functionally equivalent to the SCA reference above.
A.6 Exposing an SCA Service inside a Java EE EAR file

This use case is similar to sections A.2 and A.4, except that the SCA service is going to be deployed inside a Java EE EAR file so that it can be referenced by an EJB client, using the EJB assembly model.

The following is an example of the SCA service binding.

```
<service name="CompanyInfo">
    <interface.java interface="com.app.jobbank.CompanyInfo"/>
    <binding.ejb
        homeInterface="com.app.jobbank.CompanyInfoHome"
        ejb-link-name="companyInfoEJB.jar#CompanyInfoComponent"/>
    <reference>CompanyInfoComponent/CompanyInfo</reference>
</service>
```

Figure 6: SCA Service with client within one EAR file
The following is an example of an EJB deployment descriptor created by the client that is wired to the SCA Service binding.

```xml
<ejb-ref>
  <ejb-ref-name>ejb/CompanyInfo</ejb-ref-name>
  <ejb-ref-type>Session</ejb-ref-type>
  <home>com.app.jobbank.CompanyInfoHome</home>
  <remote>com.app.jobbank.CompanyInfo</remote>
  <ejb-link>companyInfoEJB.jar#CompanyInfoComponent</ejb-link>
</ejb-ref>
```

Note: There is a variant of this use case that needs to be considered. If the SCA service is in the same EJB module as the client, then the ejb-link specified by the client does not have to include the EJB module jar name.

Deleted: should
B. EJB binding schema

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright(C) OASIS(R) 2005,2009. All Rights Reserved.  
OASIS trademark, IPR and other policies apply.  -->
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200903"
targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200903"
elementFormDefault="qualified">

<include schemaLocation="sca-core-1.1-cd03.xsd"/>

<element name="binding.ejb" type="sca:EJBSessionBeanBinding"
  substitutionGroup="sca:binding"/>

<simpleType name="VersionValue">
  <restriction base="string">
    <enumeration value="EJB2"/>
    <enumeration value="EJB3"/>
  </restriction>
</simpleType>

<complexType name="EJBSessionBeanBinding">
  <complexContent>
    <extension base="sca:Binding">
      <sequence>
        <any namespace="##other" processContents="lax"
          minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="homeInterface" type="NCName"
        use="optional"/>
      <attribute name="ejb-link-name" type="string"
        use="optional"/>
      <attribute name="ejb-version" type="sca:VersionValue"
        use="optional" default="EJB3"/>
    </extension>
  </complexContent>
</complexType>
</schema>
```
## C. Conformance Items

This section contains a list of conformance items for the SCA EJB Session Bean Binding specification.

<table>
<thead>
<tr>
<th>Conformance ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[BSB20001]</td>
<td>For <code>&lt;binding.ejb/&gt;</code>, if @ejb-version=&quot;EJB2&quot;, then @homeInterface MUST be specified and MUST have a value that is the fully qualified package name of the Java interface class of the EJB's home interface.</td>
</tr>
<tr>
<td>[BSB20002]</td>
<td>When <code>&lt;binding.ejb/&gt;</code> applies to a reference, if @ejb-link-name attribute is specified it MUST contain the value of an <code>&lt;ejb-link/&gt;</code> target packaged within the same Java EE EAR file as the component containing the reference.</td>
</tr>
<tr>
<td>[BSB20003]</td>
<td>When <code>&lt;binding.ejb/&gt;</code> applies to a service, if @ejb-link-name attribute is specified, it MUST contain a value in the form &quot;&lt;jar-name&gt;#&lt;ejb-name&gt;&quot; which MUST be unique amongst the <code>&lt;ejb-link/&gt;</code> targets contained within the same Java EE EAR file as the component containing the service.</td>
</tr>
<tr>
<td>[BSB20004]</td>
<td>When <code>&lt;binding.ejb/&gt;</code> applies to a service and the @ejb-version attribute is set to 'EJB2', the SCA Runtime MUST support invocation of the service using the EJB 2.x client view as specified in the Java EE specification [SCAJEE].</td>
</tr>
<tr>
<td>[BSB20005]</td>
<td>When <code>&lt;binding.ejb/&gt;</code> applies to a service and the @ejb-version attribute is set to 'EJB3', the SCA Runtime MUST support invocation of the service using the EJB 3.x client view as specified in the Java EE specification [SCAJEE].</td>
</tr>
<tr>
<td>[BSB20006]</td>
<td>The value of the @uri attribute MUST take the form of an Object URL as specified in the CORBA Services specification [CORBA].</td>
</tr>
<tr>
<td>[BSB20007]</td>
<td>When <code>&lt;binding.ejb/&gt;</code> applies to an SCA reference, the @uri and @ejb-link-name attributes MUST NOT be specified together in the same binding configuration.</td>
</tr>
<tr>
<td>[BSB20008]</td>
<td>The <code>&lt;binding.ejb/&gt;</code> element MUST conform to the XML schema defined in the sca-binding-ejb.xsd.</td>
</tr>
<tr>
<td>[BSB30001]</td>
<td>When used with the EJB binding, an SCA runtime MUST ensure that a service or reference interface is compatible with a session bean interface, according to the rules defined in the section &quot;Interface Mapping&quot;.</td>
</tr>
<tr>
<td>[BSB30002]</td>
<td>An EJB 2.x session bean interface itself MUST NOT be used as the interface of a reference binding.</td>
</tr>
</tbody>
</table>
| [BSB30003]    | When `<binding.ejb/>` applies to a service and @ejb-version is set to "EJB2", the binding implementation MUST implement the methods.
from the EJBObject or EJBLocalObject interface.

**[BSB40001]**
The EJB [EJB] specifications define non-business exceptions that can
be thrown to the EJB client. When `<binding.ejb/>` applies to a
reference, the SCA Runtime MUST wrap non-business exceptions in
a ServiceRuntimeException that is thrown to the client [JAVACAA].

**[BSB60001]**
When `<binding.ejb/>` applies to an SCA service, the Java interface
class specified on the `@homeInterface` attribute MUST have one
create method [EJB].
## D. Acknowledgements

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

<table>
<thead>
<tr>
<th>Participant Name</th>
<th>Affiliation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bryan Aupperle</td>
<td>IBM</td>
</tr>
<tr>
<td>Ron Barack</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>Michael Beisiegel</td>
<td>IBM</td>
</tr>
<tr>
<td>Henning Blohm</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>David Booz</td>
<td>IBM</td>
</tr>
<tr>
<td>Martin Chapman</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Graham Charters</td>
<td>IBM</td>
</tr>
<tr>
<td>Shih-Chang Chen</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Chris Cheng</td>
<td>Primeton Technologies, Inc.</td>
</tr>
<tr>
<td>Vamsavardhana Reddy Chillakuru</td>
<td>IBM</td>
</tr>
<tr>
<td>Roberto Chinnici</td>
<td>Sun Microsystems</td>
</tr>
<tr>
<td>Pyounguk Cho</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Eric Clairambault</td>
<td>IBM</td>
</tr>
<tr>
<td>Mark Combellack</td>
<td>Avaya, Inc.</td>
</tr>
<tr>
<td>Jean-Sebastien Delfino</td>
<td>IBM</td>
</tr>
<tr>
<td>Mike Edwards</td>
<td>IBM</td>
</tr>
<tr>
<td>Raymond Feng</td>
<td>IBM</td>
</tr>
<tr>
<td>Bo Ji</td>
<td>Primeton Technologies, Inc.</td>
</tr>
<tr>
<td>Uday Joshi</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Anish Karmarkar</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Michael Keith</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Rainer Kerth</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>Meeraj Kunnumpurath</td>
<td>Individual</td>
</tr>
<tr>
<td>Simon Laws</td>
<td>IBM</td>
</tr>
<tr>
<td>Yang Lei</td>
<td>IBM</td>
</tr>
<tr>
<td>Mark Little</td>
<td>Red Hat</td>
</tr>
<tr>
<td>Ashok Malhotra</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Jim Marino</td>
<td>Individual</td>
</tr>
<tr>
<td>Jeff Mischkinsky</td>
<td>Oracle Corporation</td>
</tr>
<tr>
<td>Sriram Narasimhan</td>
<td>TIBCO Software Inc.</td>
</tr>
<tr>
<td>Simon Nash</td>
<td>Individual</td>
</tr>
<tr>
<td>Sanjay Patil</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>Plamen Pavlov</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>Peter Peshev</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>Ramkumar Ramalingam</td>
<td>IBM</td>
</tr>
<tr>
<td>Luciano Resende</td>
<td>IBM</td>
</tr>
<tr>
<td>Michael Rowley</td>
<td>Active Endpoints, Inc.</td>
</tr>
<tr>
<td>Vladimir Savchenko</td>
<td>SAP AG*</td>
</tr>
<tr>
<td>Pradeep Simha</td>
<td>TIBCO Software Inc.</td>
</tr>
<tr>
<td>Raghav Srinivasan</td>
<td>Oracle Corporation</td>
</tr>
</tbody>
</table>
E. Non-Normative Text
### F. Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2007-09-26</td>
<td>Anish Karmarkar</td>
<td>Applied the OASIS template + related changes to the Submission</td>
</tr>
<tr>
<td>2</td>
<td>2007-10-04</td>
<td>David Booz</td>
<td>Issue 5: Ending a conversation should invoke the remove method of EJBOBJECT or EJBLocalObject.</td>
</tr>
<tr>
<td>wd02</td>
<td>2007-11-02</td>
<td>David Booz</td>
<td>Applied OSOA Errata</td>
</tr>
<tr>
<td>wd03</td>
<td>2009-06-04</td>
<td>David Booz</td>
<td>Editorial upgrade of namespaces, attribute descriptions, etc</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Applied Issues 86, 140</td>
</tr>
<tr>
<td>wd04</td>
<td>2009-07-20</td>
<td>David Booz</td>
<td>Applied 24, 122, 118</td>
</tr>
<tr>
<td>ISSUE-107</td>
<td>2009-07-29</td>
<td>David Booz</td>
<td>Issue 107 proposal v3</td>
</tr>
</tbody>
</table>
When `<binidng.ejb/>` applies to a service, if `@ejb-link-name` attribute is specified, it MUST contain a value in the form "<jar-name>#$<ejb-name>" which MUST be unique amongst the `<ejb-link/>` targets contained within the same Java EE EAR file as the component containing the service.

When SCA Services are exposed as EJB 2.X session beans, the exposed interface will inherit from `EJBObject` or `EJBLocalObject`.

However, the session bean interface itself cannot be used as the interface of a reference binding.

I think that there is a need to describe HOW such an interface is derived. There is also a need to say how the derived interface gets used in the presence of the original interface.

At an EJB client MUST interact with the binding using the...