Contents

About This Document

What You Need to Know ............................................................... vii
e-docs Web Site ........................................................................ viii
How to Print the Document ....................................................... viii
Related Information ................................................................. viii
Contact Us! ............................................................................... ix
Documentation Conventions ................................................... ix

1. EJB Deployment Descriptor Reference

About Deployment Descriptors .................................................. 1-2
  EJB Roles and Deployment Descriptors .................................. 1-2
File Formats ........................................................................... 1-3
DOCTYPE Declaration ............................................................ 1-3
Summary of Deployment Descriptor Elements ...................... 1-3
Description of Deployment Descriptor Elements .................. 1-8
  assembly-descriptor ......................................................... 1-8
  cmp-field .................................................................... 1-8
  container-transaction ..................................................... 1-8
  description ................................................................. 1-8
  display-name ............................................................... 1-9
  ejb-class .................................................................... 1-9
  ejb-jar ........................................................................ 1-10
  ejb-link ...................................................................... 1-10
  ejb-name .................................................................... 1-10
  ejb-ref ........................................................................ 1-10
  ejb-ref-name .............................................................. 1-11
  ejb-ref-type ............................................................... 1-11

EJB XML Reference iii
<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>enterprise-beans</td>
<td>1-11</td>
</tr>
<tr>
<td>entity</td>
<td>1-12</td>
</tr>
<tr>
<td>env-entry</td>
<td>1-12</td>
</tr>
<tr>
<td>env-entry-name</td>
<td>1-12</td>
</tr>
<tr>
<td>env-entry-type</td>
<td>1-13</td>
</tr>
<tr>
<td>env-entry-value</td>
<td>1-13</td>
</tr>
<tr>
<td>field-name</td>
<td>1-13</td>
</tr>
<tr>
<td>home</td>
<td>1-14</td>
</tr>
<tr>
<td>large-icon</td>
<td>1-14</td>
</tr>
<tr>
<td>method</td>
<td>1-14</td>
</tr>
<tr>
<td>method-intf</td>
<td>1-14</td>
</tr>
<tr>
<td>method-name</td>
<td>1-14</td>
</tr>
<tr>
<td>method-param</td>
<td>1-15</td>
</tr>
<tr>
<td>method-params</td>
<td>1-15</td>
</tr>
<tr>
<td>method-permission</td>
<td>1-16</td>
</tr>
<tr>
<td>persistence-type</td>
<td>1-16</td>
</tr>
<tr>
<td>prim-key-class</td>
<td>1-16</td>
</tr>
<tr>
<td>primkey-field</td>
<td>1-17</td>
</tr>
<tr>
<td>reentrant</td>
<td>1-17</td>
</tr>
<tr>
<td>remote</td>
<td>1-17</td>
</tr>
<tr>
<td>res-auth</td>
<td>1-18</td>
</tr>
<tr>
<td>res-ref-name</td>
<td>1-18</td>
</tr>
<tr>
<td>res-type</td>
<td>1-18</td>
</tr>
<tr>
<td>resource-ref</td>
<td>1-18</td>
</tr>
<tr>
<td>role-link</td>
<td>1-19</td>
</tr>
<tr>
<td>role-name</td>
<td>1-19</td>
</tr>
<tr>
<td>security-role</td>
<td>1-19</td>
</tr>
<tr>
<td>security-role-ref</td>
<td>1-19</td>
</tr>
<tr>
<td>session-type</td>
<td>1-20</td>
</tr>
<tr>
<td>session</td>
<td>1-20</td>
</tr>
<tr>
<td>small-icon</td>
<td>1-20</td>
</tr>
<tr>
<td>transaction-type</td>
<td>1-21</td>
</tr>
<tr>
<td>trans-attribute</td>
<td>1-21</td>
</tr>
<tr>
<td>ID</td>
<td>1-22</td>
</tr>
</tbody>
</table>
2. WebLogic Enterprise EJB Extensions Reference

About the WebLogic Enterprise EJB Extensions ........................................... 2-2
DOCTYPE Declaration ..................................................................................... 2-2
Summary of WebLogic Enterprise EJB Extensions ...................................... 2-3
Description of WebLogic Enterprise EJB Extension Elements ................. 2-6
  attribute-map ............................................................................................ 2-6
  attribute-map-entry .................................................................................. 2-6
  bean-field-name ...................................................................................... 2-6
  finder-descriptor ...................................................................................... 2-7
  is-cacheable ............................................................................................ 2-8
  is-modified-method-name ...................................................................... 2-9
  jndi-name ............................................................................................ 2-10
  module-initializer-class ........................................................................ 2-11
  module-initializer-class-name ............................................................. 2-11
  persistence-store-descriptor ............................................................... 2-12
  persistence-store-directory-root .......................................................... 2-13
  persistence-store-file ............................................................................. 2-14
  persistence-store-jdbc ........................................................................... 2-14
  pool-name .............................................................................................. 2-15
  query-grammar ....................................................................................... 2-15
  table-column-name ................................................................................ 2-17
  table-name .............................................................................................. 2-17
  trans-timeout-seconds .......................................................................... 2-17
  weblogic-deployment-params ............................................................... 2-17
  weblogic-ejb-extensions ....................................................................... 2-18
  weblogic-enterprise-bean ..................................................................... 2-18
  weblogic-version .................................................................................... 2-18

Index
About This Document

This document provides XML reference information for configuring Enterprise JavaBean (EJB) applications that run in the BEA WebLogic Enterprise™ (WLE) environment.

This document includes the following topics:

- Chapter 1, “EJB Deployment Descriptor Reference,” describes the XML DTD for the EJB deployment descriptor, as specified in the Enterprise JavaBeans Specification 1.1 published by Sun Microsystems, Inc.

- Chapter 2, “WebLogic Enterprise EJB Extensions Reference,” describes the WebLogic Enterprise XML extensions for EJB applications. WebLogic Enterprise extends the ability to configure EJBs by providing additional, optional XML elements that can be associated with EJBs. These extensions are defined in a separate file (weblogic-ejb-extensions.xml), in accordance with the Enterprise JavaBeans Specification 1.1.

What You Need to Know

This document is intended for Bean Providers, Application Assemblers, and Deployers who need to review or change the configuration for EJBs in EJB applications. It assumes that you are knowledgeable about EJB programming and familiar with the Enterprise JavaBeans Specification 1.1.
e-docs Web Site

The BEA WebLogic Enterprise product documentation is available on the BEA Systems, Inc. corporate Web site. From the BEA Home page, click the Product Documentation button or go directly to the “e-docs” Product Documentation page at http://e-docs.bea.com.

How to Print the Document

You can print a copy of this document from a Web browser, one file at a time, by using the File—>Print option on your Web browser.

A PDF version of this document is available on the WebLogic Enterprise documentation Home page on the e-docs Web site (and also on the documentation CD). You can open the PDF in Adobe Acrobat Reader and print the entire document (or a portion of it) in book format. To access the PDFs, open the WebLogic Enterprise documentation Home page, click the PDF Files button, and select the document you want to print.

If you do not have Adobe Acrobat Reader installed, you can download it for free from the Adobe Web site at http://www.adobe.com/.

Related Information

For more information about CORBA, Java 2 Enterprise Edition (J2EE), BEA Tuxedo®, distributed object computing, transaction processing, C++ programming, and Java programming, see the WebLogic Enterprise Bibliography in the WebLogic Enterprise online documentation.
Contact Us!

Your feedback on the BEA WebLogic Enterprise documentation is important to us. Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA Systems, Inc. professionals who create and update the WebLogic Enterprise documentation.

In your e-mail message, please indicate that you are using the documentation for the BEA WebLogic Enterprise 5.1 release.

If you have any questions about this version of BEA WebLogic Enterprise, or if you have problems installing and running BEA WebLogic Enterprise, contact BEA Customer Support through BEA WebSUPPORT at www.bea.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Documentation Conventions

The following documentation conventions are used throughout this document.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface text</strong></td>
<td>Indicates terms defined in the glossary.</td>
</tr>
<tr>
<td>Ctrl+Tab</td>
<td>Indicates that you must press two or more keys simultaneously.</td>
</tr>
<tr>
<td>Convention</td>
<td>Item</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td><em>italics</em></td>
<td>Indicates emphasis or book titles.</td>
</tr>
</tbody>
</table>
| *monospace*      | Indicates code samples, commands and their options, data structures and their members, data types, directories, and filenames and their extensions. Monospace text also indicates text that you must enter from the keyboard.  
*Examples:*  
#include <iostream.h> void main ( ) the pointer psz  
chmod u+w *  
\tux\data\ap  
.doc  
tux.doc  
BITMAP  
float |
| *monospace*      | Identifies significant words in code.                               |
| *boldface*       | Example:                                                            |
| *italic*         | void commit ( )                                                     |
| *monospace*      | Identifies variables in code.                                       |
| *italic*         | Example:                                                            |
| *text*           | String expr                                                         |
| **UPPERCASE**    | Indicates device names, environment variables, and logical operators. |
| **TEXT**         | Example:                                                            |
|                  | LPT1                                                               |
|                  | SIGNON                                                             |
|                  | OR                                                                 |
|                  | ( ) Indicates a set of choices in a syntax line. The braces themselves should never be typed. |
|                  | [ ] Indicates optional items in a syntax line. The brackets themselves should never be typed.  
*Example:*  
buldobjclient [-v] [-o name ] [-f file-list]...  
[-l file-list]... |
### Documentation Conventions

<table>
<thead>
<tr>
<th>Convention</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.</td>
</tr>
<tr>
<td>...</td>
<td>Indicates one of the following in a command line:</td>
</tr>
<tr>
<td></td>
<td>- That an argument can be repeated several times in a command line</td>
</tr>
<tr>
<td></td>
<td>- That the statement omits additional optional arguments</td>
</tr>
<tr>
<td></td>
<td>- That you can enter additional parameters, values, or other information</td>
</tr>
<tr>
<td></td>
<td>The ellipsis itself should never be typed.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td><code>buildobjclient [v] [o name ] [f file-list]... [l file-list]...</code></td>
</tr>
<tr>
<td>.</td>
<td>Indicates the omission of items from a code example or from a syntax line.</td>
</tr>
<tr>
<td>.</td>
<td>The vertical ellipsis itself should never be typed.</td>
</tr>
</tbody>
</table>
This topic describes the XML DTD for the EJB deployment descriptor, as defined in the Enterprise JavaBeans Specification 1.1 published by Sun Microsystems, Inc.

This topic includes the following sections:

- About Deployment Descriptors
- DOCTYPE Declaration
- Summary of Deployment Descriptor Elements
- Description of Deployment Descriptor Elements

In addition to the standard EJB deployment descriptor provided by Sun Microsystems, Inc., WebLogic Enterprise (WLE) provides the WebLogic Enterprise EJB Extensions, which are XML elements used to further customize EJBs for the WebLogic Enterprise environment. For more information, see Chapter 2, “WebLogic Enterprise EJB Extensions Reference.” For more information about deploying EJBs in the WebLogic Enterprise environment, see “Developing WebLogic Enterprise EJB Applications” in Getting Started.

Note: Material in this topic was republished with permission from Sun Microsystems, Inc.
About Deployment Descriptors

A deployment descriptor is a file that defines the following kinds of information:

- **EJB structural information**, such as the EJB name, class, home and remote interfaces, bean type (session or entity), environment entries, resource factory references, EJB references, security role references, as well as additional information based on the bean type.

- **Application assembly information**, such as EJB references, security roles, security role references, method permissions, and container transaction attributes. Specifying assembly descriptor information is an optional task that an Application Assembler performs.


EJB Roles and Deployment Descriptors

Specifying deployment descriptor information is a required task that a Bean Provider performs. The Bean Provider creates a standard EJB deployment descriptor file using the XML markup conventions in accordance with the syntax described in the Enterprise JavaBeans Specification 1.1. Multiple EJBs can be defined in a single deployment descriptor. For more information about creating deployment descriptors, see the following documents:

- “Developing WebLogic Enterprise EJB Applications” in Getting Started
- Using the WebLogic EJB Deployer

An Application Assembler or Deployer may subsequently modify only certain information (such as the EJB name, values of environment entries, and description entries) in the deployment descriptor. The Deployer may use the WebLogic EJB Deployer to update certain settings in the deployment descriptor.

Thereafter, the Application Assembler creates a deployable EJB JAR file using the ejbc command-line tool, specifying the -l parameter and passing this filename (ejb-jar.xml). For more information about the ejbc command-line tool, see the Commands, Systems Processes, and MIB Reference.
Deployment descriptors are stored in the EJB JAR file. The WebLogic EJB Deployer accepts standard, extended, and deployable EJB JAR files.

**Summary of Deployment Descriptor Elements**

Table 1-1 describes the standard deployment descriptor elements.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>EJB Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>assembly-descriptor</td>
<td>Contains application-assembly information.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>cmp-field</td>
<td>Describes a container-managed field.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>container-transaction</td>
<td>Specifies how the container must manage transaction scopes for the EJB’s method invocations.</td>
<td>Application Assembler</td>
</tr>
</tbody>
</table>
Table 1-1 List of Standard Deployment Descriptor Elements (Continued)

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>EJB Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>Provides text describing the parent element.</td>
<td>Bean Provider or Application Assembler</td>
</tr>
<tr>
<td>display-name</td>
<td>Specifies a short name that is intended to be displayed by tools.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>ejb-class</td>
<td>Specifies the fully qualified name of the EJB’s class.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>ejb-jar</td>
<td>Root element of the EJB deployment descriptor.</td>
<td>All</td>
</tr>
<tr>
<td>ejb-link</td>
<td>Used in the ejb-ref element to specify that an EJB reference is linked to another EJB in the EJB JAR file.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>ejb-name</td>
<td>Specifies an EJB’s name.</td>
<td>Bean Provider or Application Assembler</td>
</tr>
<tr>
<td>ejb-ref</td>
<td>Declares a reference to another EJB’s home.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>ejb-ref-name</td>
<td>Specifies the name of an EJB reference, which is an entry in the EJB’s environment.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>ejb-ref-type</td>
<td>Specifies the expected type (entity or session) of the referenced EJB.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>enterprise-beans</td>
<td>Declares one or more EJBs.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>entity</td>
<td>Declares an entity bean.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>env-entry</td>
<td>Declares an EJB’s environment entries.</td>
<td>Bean Provider or Application Assembler</td>
</tr>
<tr>
<td>env-entry-name</td>
<td>Specifies the name of an EJB’s environment entry.</td>
<td>Bean Provider or Application Assembler</td>
</tr>
<tr>
<td>env-entry-type</td>
<td>Specifies the fully qualified Java type of the environment entry value that is expected by the EJB’s code.</td>
<td>Bean Provider or Application Assembler</td>
</tr>
<tr>
<td>Element</td>
<td>Description</td>
<td>EJB Role</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>env-entry-value</td>
<td>Specifies the value of an EJB’s environment entry.</td>
<td>Bean Provider or Application Assembler</td>
</tr>
<tr>
<td>field-name</td>
<td>Specifies the name of a container-managed field.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>home</td>
<td>Specifies the fully qualified name of the EJB’s home interface.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>large-icon</td>
<td>Specifies the name of a file containing a large (32 x 32) icon image. The filename is a relative path within the EJB JAR file.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>method</td>
<td>Denotes a method of an EJB’s home or remote interface, or a set of methods.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>method-intf</td>
<td>Allows a method element to differentiate between the methods, with the same name and signature, that are defined in both the remote and home interfaces.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>method-name</td>
<td>Specifies the name of an EJB method, or the asterisk (*) character, which is used when the element denotes all the methods of an EJB’s remote and home interfaces.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>method-param</td>
<td>Specifies the fully qualified Java type name of a method parameter.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>method-params</td>
<td>Contains a list of the fully qualified Java type names of the method parameters.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>method-permission</td>
<td>Specifies one or more security roles that are allowed to invoke one or more EJB methods.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>persistence-type</td>
<td>Specifies an entity bean’s persistence management type (bean- or container-managed).</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>prim-key-class</td>
<td>Specifies the fully qualified name of an entity bean’s primary key class.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>primkey-field</td>
<td>Specifies the name of the primary key field for an entity with container-managed persistence.</td>
<td>Bean Provider</td>
</tr>
</tbody>
</table>
### Table 1-1  List of Standard Deployment Descriptor Elements (Continued)

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>EJB Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>reentrant</td>
<td>Specifies whether an entity bean is reentrant (True) or not (False).</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>remote</td>
<td>Specifies the fully qualified name of the EJB’s remote interface.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>res-auth</td>
<td>Specifies whether the EJB code signs on programmatically to the resource manager, or whether the Container will sign on to the resource manager on behalf of the EJB. In the latter case, the Container uses information that is supplied by the Deployer.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>res-ref-name</td>
<td>Specifies the name of a resource factory reference.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>res-type</td>
<td>Specifies the type of the data source, which is the Java interface (or class) expected to be implemented by the data source.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>resource-ref</td>
<td>Declares the EJB’s reference to an external resource.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>role-link</td>
<td>Links a security role reference to a defined security role.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>role-name</td>
<td>Specifies the name of a security role.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>security-role</td>
<td>Defines a security role.</td>
<td>Application Assembler</td>
</tr>
<tr>
<td>security-role-ref</td>
<td>Declares a security role reference in the EJB’s code.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>session-type</td>
<td>Specifies the type of a session bean (stateful or stateless).</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>session</td>
<td>Declares a session bean.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>small-icon</td>
<td>Specifies the name of a file containing a small (16 x 16) icon image. The filename is relative path within the EJB JAR file.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>transaction-type</td>
<td>Specifies the transaction management type of an EJB (bean- or container-managed).</td>
<td>Bean Provider</td>
</tr>
</tbody>
</table>
Table 1-1  List of Standard Deployment Descriptor Elements  (Continued)

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>EJB Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>trans-attribute</td>
<td>Specifies how the container must manage the transaction boundaries when delegating a method invocation to an EJB’s business method.</td>
<td>Application Assembler</td>
</tr>
</tbody>
</table>

Summary of Deployment Descriptor Elements
Description of Deployment Descriptor Elements

**assembly-descriptor**

- **Synopsis**: Contains application-assembly information.
- **Syntax**: `<!ELEMENT assembly-descriptor (security-role*, method-permission*, container-transaction*)>`
- **EJB Role**: Application Assembler
- **Used In**: `ejb-jar`

**cmp-field**

- **Synopsis**: Describes a container-managed field.
- **Syntax**: `<!ELEMENT cmp-field (description?, field-name)>`
- **EJB Role**: Bean Provider
- **Used In**: `entity`

**container-transaction**

- **Synopsis**: Specifies how the container must manage transaction scopes for the EJB’s method invocations.
- **Syntax**: `<!ELEMENT container-transaction (description?, method+, trans-attribute)>`
- **EJB Role**: Application Assembler
- **Used In**: `assembly-descriptor`

**description**

- **Synopsis**: Provides text describing the parent element.
- **Syntax**: `<!ELEMENT description (#PCDATA)>`
Description of Deployment Descriptor Elements

**EJB Role**
Bean Provider or Application Assembler

**Used In**
- cmp-field
- container-transaction
- ejb-jar
- entity
- env-entry
- ejb-ref
- method
- method-permission
- resource-ref
- security-role
- security-role-ref
- session

**display-name**

**Synopsis**
Specifies a short name that is intended to be displayed by tools.

**Syntax**
```xml
<!ELEMENT display-name (#PCDATA)>
```

**EJB Role**
Bean Provider

**Used In**
- ejb-jar
- session
- entity

**ejb-class**

**Synopsis**
Specifies the fully qualified name of the EJB’s class.

**Syntax**
```xml
<!ELEMENT ejb-class (#PCDATA)>
```
1 EJB Deployment Descriptor Reference

**ejb-jar**

**Synopsis** Root element of the EJB deployment descriptor.

**Syntax**

```
<!ELEMENT ejb-jar (description?, display-name?, small-icon?,
large-icon?, enterprise-beans, assembly-descriptor?)>
```

**EJB Role** All

**ejb-link**

**Synopsis** Used in the ejb-ref element to specify that an EJB reference is linked to another EJB in the EJB JAR file.

**Syntax**

```
<!ELEMENT ejb-link (#PCDATA)>
```

**Valid Values** The value of the ejb-link element must be the ejb-name of an EJB in the same EJB JAR file, or in another EJB JAR file in the same J2EE application unit.

**EJB Role** Application Assembler

**Used In** ejb-ref

**ejb-name**

**Synopsis** Specifies an EJB’s name.

**Syntax**

```
<!ELEMENT ejb-name (#PCDATA)>
```

**Valid Values**

- The name must be unique among the names of the EJBs in the same EJB JAR file.
- The name must conform to the lexical rules for an NMTOKEN.

**EJB Role** Bean Provider or Application Assembler

**Used In**

- entity
- method
- session

**ejb-ref**

**Synopsis** Declares a reference to another EJB’s home.
Description of Deployment Descriptor Elements

Syntax

```
<!ELEMENT ejb-ref (description?, ejb-ref-name, ejb-ref-type, home, remote, ejb-link?)>
```

**EJB Role**

Bean Provider

**Used In**

- entity
- session

**ejb-ref-name**

**Synopsis**

Specifies the name of an EJB reference, which is an entry in the EJB’s environment.

**Syntax**

```
<!ELEMENT ejb-ref-name (#PCDATA)>
```

**Valid Values**

It is recommended that name is prefixed with ejb/.

**EJB Role**

Bean Provider

**Used In**

ejb-ref

**ejb-ref-type**

**Synopsis**

Specifies the expected type (entity or session) of the referenced EJB.

**Syntax**

```
<!ELEMENT ejb-ref-type (#PCDATA)>
```

**Valid Values**

One of the following values:

- Entity
- Session

**EJB Role**

Bean Provider

**Used In**

ejb-ref

**enterprise-beans**

**Synopsis**

Declares one or more EJBs.

**Syntax**

```
<!ELEMENT enterprise-beans (session | entity)+>
```

**EJB Role**

Bean Provider
entity

Synopsis
Declares an entity bean.

Syntax
<!ELEMENT entity (description?, display-name?, small-icon?,
large-icon?, ejb-name, home, remote, ejb-class, persistence-type,
prim-key-class, reentrant, cmp-field*, primkey-field?, env-entry*,
ejb-ref*, security-role-ref*, resource-ref*)>

Valid Values
- The optional primkey-field may be present in the descriptor if the entity’s
  persistence-type is Container.
- At least one cmp-field element must be present in the descriptor if the entity’s
  persistency-type is Container, and none must not be present if the entity’s
  persistence-type is Bean.

EJB Role  Bean Provider
Used In   enterprise-beans

env-entry

Synopsis
Declares an EJB’s environment entries.

Syntax
<!ELEMENT env-entry (description?, env-entry-name, env-entry-type,
env-entry-value?)>

EJB Role  Bean Provider or Application Assembler
Used In   entity
          session

env-entry-name

Specifies the name of an EJB’s environment entry.

Syntax
<!ELEMENT env-entry-name (#PCDATA)>

EJB Role  Bean Provider or Application Assembler
Used In   env-entry
Description of Deployment Descriptor Elements

**env-entry-type**

**Synopsis**
Specifies the fully qualified Java type of the environment entry value that is expected by the EJB’s code.

**Syntax**
```xml
<!ELEMENT env-entry-type (#PCDATA)>
```

**Valid Values**
One of the following values:
- `java.lang.Boolean`
- `java.lang.String`
- `java.lang.Integer`
- `java.lang.Double`
- `java.lang.Byte`
- `java.lang.Short`
- `java.lang.Long`
- `java.lang.Float`

**EJB Role**
Bean Provider or Application Assembler

**Used In**
`env-entry`

**env-entry-value**

**Synopsis**
Specifies the value of an EJB’s environment entry.

**Syntax**
```xml
<!ELEMENT env-entry-value (#PCDATA)>
```

**EJB Role**
Bean Provider or Application Assembler

**Used In**
`env-entry`

**field-name**

**Synopsis**
Specifies the name of a container-managed field.

**Syntax**
```xml
<!ELEMENT field-name (#PCDATA)>
```

**Valid Values**
The name must be a public field of the EJB class or one of its superclasses.

**EJB Role**
Bean Provider

**Used In**
cmp-field
home

Synopsis  Specifies the fully qualified name of the EJB’s home interface.
Syntax  <!ELEMENT home (#PCDATA)> 
EJB Role  Bean Provider
Used In  ■  ejb-ref
         ■  entity
         ■  session

large-icon

Synopsis  Specifies the name of a file containing a large (32 x 32) icon image. The filename is a relative path within the EJB JAR file.
Syntax  <!ELEMENT large-icon (#PCDATA)> 
Valid Values  ■  The image must be either in the JPEG or GIF format.
              ■  The filename must end with the suffix .jpg or .gif, respectively.
EJB Role  Bean Provider

method

Synopsis  Denotes a method of an EJB’s home or remote interface, or a set of methods.
Syntax  <!ELEMENT method (description?, ejb-name, method-intf?, method-name, method-params?)>
Valid Values  The ejb-name element must be the name of one of the EJBs declared in the deployment descriptor.
EJB Role  Application Assembler
Used In  ■  method-permission
         ■  container-transaction

method-intf

Synopsis  Allows a method element to differentiate between the methods, with the same name and signature, that are defined in both the remote and home interfaces.
Description of Deployment Descriptor Elements

Syntax

```xml
<!ELEMENT method-intf (#PCDATA)>
```

Valid Values

- Home
- Remote

EJB Role  Application Assembler

Used In  method

**method-name**

Synopsis

Specifies the name of an EJB method, or the asterisk (*) character, which is used when the element denotes all the methods of an EJB’s remote and home interfaces.

Syntax

```xml
<!ELEMENT method-name (#PCDATA)>
```

EJB Role  Application Assembler

Used In  method

**method-param**

Synopsis

Specifies the fully qualified Java type name of a method parameter.

Syntax

```xml
<!ELEMENT method-param (#PCDATA)>
```

EJB Role  Application Assembler

Used In  method-params

**method-params**

Synopsis

Contains a list of the fully qualified Java type names of the method parameters.

Syntax

```xml
<!ELEMENT method-params (method-param*)>
```

EJB Role  Application Assembler

Used In  method

EJB XML Reference  1-15
method-permission

Synopsis Specifies one or more security roles that are allowed to invoke one or more EJB methods.

Syntax `<!ELEMENT method-permission (description?, role-name+, method+)>

Valid Values
- The security roles used in the method-permission element must be defined in the `security-role` element of the deployment descriptor.
- The methods must be methods defined in the EJB’s remote and/or home interfaces.

EJB Role Application Assembler
Used In assembly-descriptor

persistence-type

Synopsis Specifies an entity bean’s persistence management type (bean- or container-managed).

Syntax `<!ELEMENT persistence-type (#PCDATA)>

Valid Values One of the following values:
- Bean
- Container

EJB Role Bean Provider
Used In entity

prim-key-class

Synopsis Specifies the fully qualified name of an entity bean’s primary key class.

Syntax `<!ELEMENT prim-key-class (#PCDATA)>

Valid Values If the definition of the primary key class is deferred to deployment time, the `prim-key-class` element should specify `java.lang.Object`.

EJB Role Bean Provider
Used In entity
**primkey-field**

**Synopsis**
Specifies the name of the primary key field for an entity with container-managed persistence.

**Syntax**
```xml
<!ELEMENT primkey-field (#PCDATA)>
```

**Valid Values**
- The `primkey-field` must be one of the fields declared in the `cmp-field` element.
- The type of the field must be the same as the primary key type.

**EJB Role**
Bean Provider

**Used In**
- entity

---

**reentrant**

**Synopsis**
Specifies whether an entity bean is reentrant (True) or not (False).

**Syntax**
```xml
<!ELEMENT reentrant (#PCDATA)>
```

**Valid Values**
One of the following values:
- True
- False

**EJB Role**
Bean Provider

**Used In**
- entity

---

**remote**

**Synopsis**
Specifies the fully qualified name of the EJB’s remote interface.

**Syntax**
```xml
<!ELEMENT remote (#PCDATA)>
```

**EJB Role**
Bean Provider

**Used In**
- ejb-ref
- entity
- session
1 EJB Deployment Descriptor Reference

res-auth

Synopsis Specifies whether the EJB code signs on programmatically to the resource manager, or whether the Container will sign on to the resource manager on behalf of the EJB. In the latter case, the Container uses information that is supplied by the Deployer.

Syntax <!ELEMENT res-auth (#PCDATA)>

Valid Values One of the following values:
- Application
- Container

EJB Role Bean Provider

res-ref-name

Synopsis Specifies the name of a resource manager connection factory reference.

Syntax <!ELEMENT res-ref-name (#PCDATA)>

EJB Role Bean Provider

Used In resource-ref

res-type

Synopsis Specifies the type of the data source, which is the Java interface (or class) expected to be implemented by the data source.

Syntax <!ELEMENT res-type (#PCDATA)>

EJB Role Bean Provider

Used In resource-ref

resource-ref

Synopsis Declares the EJB’s reference to an external resource.

Syntax <!ELEMENT resource-ref (description?, res-ref-name, res-type, res-auth)>

EJB Role Bean Provider

1-18 EJB XML Reference
Description of Deployment Descriptor Elements

<table>
<thead>
<tr>
<th>Used In</th>
<th>entity</th>
<th>session</th>
</tr>
</thead>
</table>

**role-link**

**Synopsis**
Links a security role reference to a defined security role.

**Syntax**
<!ELEMENT role-link (#PCDATA)>

**Valid Values**
The `role-link` element must contain the name of one of the security roles defined in the `security-role` elements.

**EJB Role**
Application Assembler

**Used In**
security-role-ref

**role-name**

**Synopsis**
Specifies the name of a security role.

**Syntax**
<!ELEMENT role-name (#PCDATA)>

**Valid Values**
The name must conform to the lexical rules for an NMTOKEN.

**EJB Role**
Application Assembler

**Used In**
method-permission

<table>
<thead>
<tr>
<th>security-role</th>
</tr>
</thead>
</table>

**Synopsis**
Defines a security role.

**Syntax**
<!ELEMENT security-role (description?, role-name)>

**EJB Role**
Application Assembler

**Used In**
assembly-descriptor

**security-role-ref**

**Synopsis**
Declares a security role reference in the EJB’s code.
1  EJB Deployment Descriptor Reference

Syntax  <!ELEMENT security-role-ref (description?, role-name, role-link?)>

Valid Values  ■ The value of the role-name element must be the string used as the parameter to the EJBCcontext.isCallerInRole(String roleName) method.
■ The value of the role-link element must be the name of one of the security roles defined in the security-role elements.

EJB Role  Bean Provider

Used In  entity
■ session

**session-type**

Synopsis  Specifies the type of a session bean (stateful or stateless).

Syntax  <!ELEMENT session-type (#PCDATA)>

Valid Values  One of the following values:
■ Stateful
■ Stateless

EJB Role  Bean Provider

**session**

Synopsis  Declares a session bean.

Syntax  <!ELEMENT session (description?, display-name?, small-icon?, large-icon?, ejb-name, home, remote, ejb-class, session-type, transaction-type, env-entry*, ejb-ref*, security-role-ref*, resource-ref*)>

EJB Role  Bean Provider

Used In  enterprise-beans

**small-icon**

Synopsis  Specifies the name of a file containing a small (16 x 16) icon image. The filename is relative path within the EJB JAR file.
Description of Deployment Descriptor Elements

Syntax

```
<!ELEMENT small-icon (#PCDATA)>
```

Valid Values
- The image must be either in the JPEG or GIF format.
- The filename must end with the suffix .jpg or .gif, respectively.

**EJB Role** Bean Provider

**transaction-type**

**Synopsis**
Specifies the transaction management type of an EJB (bean- or container-managed).

**Syntax**
```
<!ELEMENT transaction-type (#PCDATA)>
```

**Valid Values**
One of the following values:
- Bean
- Container

**EJB Role** Bean Provider

**Used In** session

**trans-attribute**

**Synopsis**
Specifies how the container must manage the transaction boundaries when delegating a method invocation to an EJB’s business method.

**Syntax**
```
<!ELEMENT trans-attribute (#PCDATA)>
```

**Valid Values**
One of the following values:
- NotSupported
- Supports
- Required
- RequiresNew
- Mandatory
- Never

**EJB Role** Application Assembler
ID

Synopsis
The ID mechanism allows tools that produce additional deployment information (such as information beyond the standard EJB deployment descriptor information) to store the nonstandard information in a separate file, and easily refer from these tools-specific files to the information in the standard deployment descriptor.

The EJB architecture does not allow the tools to add the nonstandard information into the EJB deployment descriptor.

Syntax

```xml
<!ATTLIST assembly-descriptor id ID #IMPLIED>
<!ATTLIST cmp-field id ID #IMPLIED>
<!ATTLIST container-transaction id ID #IMPLIED>
<!ATTLIST description id ID #IMPLIED>
<!ATTLIST display-name id ID #IMPLIED>
<!ATTLIST ejb-class id ID #IMPLIED>
<!ATTLIST ejb-jar id ID #IMPLIED>
<!ATTLIST ejb-link id ID #IMPLIED>
<!ATTLIST ejb-name id ID #IMPLIED>
<!ATTLIST ejb-ref id ID #IMPLIED>
<!ATTLIST ejb-ref-name id ID #IMPLIED>
<!ATTLIST ejb-ref-type id ID #IMPLIED>
<!ATTLIST enterprise-beans id ID #IMPLIED>
<!ATTLIST entity id ID #IMPLIED>
<!ATTLIST env-entry id ID #IMPLIED>
<!ATTLIST env-entry-name id ID #IMPLIED>
<!ATTLIST env-entry-type id ID #IMPLIED>
<!ATTLIST env-entry-value id ID #IMPLIED>
<!ATTLIST field-name id ID #IMPLIED>
<!ATTLIST home id ID #IMPLIED>
```
Description of Deployment Descriptor Elements

<!ATTLIST large-icon id ID #IMPLIED>
<!ATTLIST method id ID #IMPLIED>
<!ATTLIST method-intf id ID #IMPLIED>
<!ATTLIST method-name id ID #IMPLIED>
<!ATTLIST method-param id ID #IMPLIED>
<!ATTLIST method-params id ID #IMPLIED>
<!ATTLIST method-permission id ID #IMPLIED>
<!ATTLIST persistence-type id ID #IMPLIED>
<!ATTLIST prim-key-class id ID #IMPLIED>
<!ATTLIST primkey-field id ID #IMPLIED>
<!ATTLIST reentrant id ID #IMPLIED>
<!ATTLIST remote id ID #IMPLIED>
<!ATTLIST res-auth id ID #IMPLIED>
<!ATTLIST res-ref-name id ID #IMPLIED>
<!ATTLIST res-type id ID #IMPLIED>
<!ATTLIST resource-ref id ID #IMPLIED>
<!ATTLIST role-link id ID #IMPLIED>
<!ATTLIST role-name id ID #IMPLIED>
<!ATTLIST security-role id ID #IMPLIED>
<!ATTLIST security-role-ref id ID #IMPLIED>
<!ATTLIST session-type id ID #IMPLIED>
<!ATTLIST session id ID #IMPLIED>
<!ATTLIST small-icon id ID #IMPLIED>
<!ATTLIST transaction-type id ID #IMPLIED>
<!ATTLIST trans-attribute id ID #IMPLIED>
EJB Deployment Descriptor Reference
This topic describes the WebLogic EJB Extensions, which are XML elements used to configure EJBs used in applications that run in the WebLogic Enterprise (WLE) environment.

This topic includes the following sections:

- About the WebLogic Enterprise EJB Extensions
- DOCTYPE Declaration
- Summary of WebLogic Enterprise EJB Extensions
- Description of WebLogic Enterprise EJB Extension Elements

For more information about deploying EJBs in the WebLogic Enterprise environment, see “Developing WebLogic Enterprise EJB Applications” in Getting Started.
About the WebLogic Enterprise EJB Extensions

The Enterprise JavaBeans Specification 1.1, published by Sun Microsystems, Inc., defines a standard DTD for describing EJB deployment information. For a list of the standard XML elements, see Chapter 1, “EJB Deployment Descriptor Reference.”

WebLogic Enterprise extends the ability to configure EJBs by providing additional, optional XML elements that can be associated with EJBs. The Enterprise JavaBeans Specification 1.1 mandates that these extensions be specified in a separate file. The WebLogic Enterprise EJB Extensions are defined in a file named weblogic-ejb-extensions.xml.

The Bean Developer is responsible for creating this file, if needed, and the Application Assembler is responsible for including it in a deployable EJB JAR file. To add this extensions file to an EJB JAR file, an Application Assembler uses the ejbc command-line tool, specifying the -x parameter and passing the weblogic-ejb-extensions.xml filename. For more information about the ejbc command-line tool, see the Commands, Systems Processes, and MIB Reference.

DOCTYPE Declaration

All valid file named weblogic-ejb-extensions.xml files must contain the following DOCTYPE declaration:

<!DOCTYPE weblogic-ejb-extensions SYSTEM "weblogic-ejb-extensions.dtd" >
Summary of WebLogic Enterprise EJ B Extensions

Table 2-1 describes the WebLogic Enterprise XML extension elements.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>EJB Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>attribute-map</td>
<td>Specifies an attribute map containing one or more attribute map entries, which define a mapping between an EJB field (a container-managed attribute) and a table column name.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>attribute-map-entry</td>
<td>Specifies the mapping between an EJB field and a table column.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>bean-field-name</td>
<td>Specifies the name of an EJB field in an attribute map entry.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>finder-descriptor</td>
<td>Specifies the finder descriptor for finding one or more EJBs.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>is-cacheable</td>
<td>Specifies whether or not an entity bean or stateful session bean can be cached.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>is-modified-method-name</td>
<td>Specifies the name of an EJB method that tells the WebLogic Enterprise EJB container whether the bean data has been modified.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>jndi-name</td>
<td>Specifies the name for the EJB home that is registered in the Global WebLogic Enterprise JNDI namespace.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>module-initializer-class</td>
<td>Specifies the server implementation class for handling the initialize and release methods.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>Element</td>
<td>Description</td>
<td>EJB Role</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>module-initializer-class-name</td>
<td>Specifies the name of the server implementation class for handling the initialize and release methods.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>persistence-store-descriptor</td>
<td>Specifies the type of persistence storage (in a flat file or a JDBC-managed database) for container-managed persistence.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>persistence-store-directory-root</td>
<td>If persistent storage is kept in a flat file, specifies the directory root for storing all instances of an EJB. Each ejb-name element is converted to a directory name.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>persistence-store-file</td>
<td>Specifies persistent storage in a flat file.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>persistence-store-jdbc</td>
<td>Specifies persistent storage in a database that is managed by a WebLogic Enterprise JDBC connection pool.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>pool-name</td>
<td>Specifies the name of the JDBC connection pool associated with the database used to store EJB state information.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>query-grammar</td>
<td>Specifies the query expression for a finder method.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>table-column-name</td>
<td>Specifies the name of a table column in an attribute map entry.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>table-name</td>
<td>Specifies the name of the table used to store EJB state information.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>trans-timeout-seconds</td>
<td>Specifies the default timeout, in seconds, for container-initiated transactions for this bean. If a transaction duration exceeds this value, the transaction will be rolled back.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>weblogic-deployment-params</td>
<td>Specifies deployment options for the BEA WebLogic Server™.</td>
<td>Bean Provider</td>
</tr>
</tbody>
</table>
Summary of WebLogic Enterprise EJB Extensions

Table 2-1  WebLogic Enterprise XML Extension Elements  (Continued)

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>EJB Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>weblogic-ejb-extensions</td>
<td>Root element of the WebLogic Enterprise XML extensions to the deployment descriptor.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>weblogic-enterprise-bean</td>
<td>Specifies deployment information for a bean that is specific to WebLogic Enterprise.</td>
<td>Bean Provider</td>
</tr>
<tr>
<td>weblogic-version</td>
<td>Specifies the WebLogic Enterprise version.</td>
<td>Bean Provider</td>
</tr>
</tbody>
</table>
Description of WebLogic Enterprise EJB Extension Elements

attribute-map

Synopsis Specifies an attribute map containing one or more attribute map entries, which define a mapping between an EJB field (a container-managed attribute) and a table column name.

Syntax <!ELEMENT attribute-map (description?, attribute-map-entry+)>

EJB Role Bean Provider

Used In persistence-store-jdbc

attribute-map-entry

Synopsis Specifies the mapping between an EJB field and a table column.

Syntax <!ELEMENT attribute-map-entry (bean-field-name, table-column-name)>

Description The specified mapping must be between equivalent types, because no automatic type conversion between types is performed on primary key columns.

EJB Role Bean Provider

Used In attribute-map

bean-field-name

Synopsis Specifies the name of an EJB field in an attribute map entry.

Syntax <!ELEMENT bean-field-name (#PCDATA)>

EJB Role Bean Provider

Used In attribute-map-entry
Description of WebLogic Enterprise EJB Extension Elements

finder-descriptor

Synopsis Specifies the finder descriptor for finding one or more EJBs.

Syntax 

```
<!ELEMENT finder-descriptor (description?, method?, query-grammar?)>
```

Valid Values

- Method names must always start with the prefix find.
- If you use a nonprimitive data type in a finder descriptor, you must specify a fully qualified name, such as `java.sql.Timestamp`.
- Finder descriptors always use EJB field names, not column names, except for the ordering of the returned beans (the `OrderBy` expression).

Description Finder descriptors are the WebLogic Enterprise implementation of the EJB find methods. If an EJB uses container-managed JDBC persistence, you can specify finder methods of the form `findMethod()` to find either an individual EJB or a collection of EJBs.

The `finder-descriptor` elements are pairs of method signatures and expressions. You specify a method signature in the EJB Home interface and specify the method's expression in the deployment descriptor using the `query-grammar` element. Expressions contain method parameters and EJB attributes. The EJB container automatically maps the attributes to the appropriate columns in the persistent store.

For complex SQL lookups, such as a dynamically set `WHERE` clause, you need to use bean-managed persistence and write custom finders.

EJB Role Bean Provider

Used In `persistence-store-jdbc`

Example The `findBigAccounts` method has the following interface:

```java
public interface AccountHome extends EJBHome {
    // ...
    public Enumeration findBigAccounts(double balanceGreaterThan)
        throws FinderException, RemoteException;
    // ...
}
```

The following sample expression associated with the `findBigAccounts` method is defined in the deployment descriptor:

```
(> balance $balanceGreaterThan)
```
where balance is an attribute (field) of the EJB and $balanceGreaterThan is the symbol for the method parameter balanceGreaterThan.

A call in the client application, such as myEJBean.findBigAccounts(amount), will return a list of all EJBs whose balance attribute is greater than the value of amount.

**is-cacheable**

**Synopsis**
Specifies whether or not an entity bean or stateful session bean can be cached.

**Syntax**
```xml
<!ELEMENT is-cacheable (#PCDATA)>
```

**Valid Values**
One of the following values:
- True
- False

**Description**
The information supplied for the `is-cacheable` element is used by the EJB container to determine whether a bean can be cached. The `is-cacheable` element applies to entity beans and stateful session beans, but not to stateless session beans.

The setting for the `is-cacheable` element determines the EJB's caching behavior:

- If set to True (the default), the WebLogic Enterprise EJB container will cache the bean and passivate it based on the caching parameters that are set in the `UBBCONFIG` file. For more information, see the description for the following JavaServer parameters, MAXEJBCACHE and EJBCACHE_FLUSH, in “Creating a Configuration File” in the Administration Guide.

- If set to False, then caching is disabled for this bean.

**Note:** If caching is disabled in the `UBBCONFIG` file, then setting the `is-cacheable` element to True has no effect. Also, entity bean or stateful session bean are cached by default, so explicitly setting the `is-cacheable` element to True is not necessary.

In general, you specify the `is-cacheable` element only if you want to turn caching off for a specific bean on a server that supports caching. If `is-cacheable` is set to True for a bean that resides on a server with caching turned off, the bean is not cached.

**EJB Role**
Bean Provider

**Used In**
weblogic-deployment-params
Description of WebLogic Enterprise EJB Extension Elements

is-modified-method-name

Synopsis
Specifies the name of an EJB method that tells the WebLogic Enterprise EJB container whether the bean data has been modified or not.

Syntax
<!ELEMENT is-modified-method-name (#PCDATA)>

Description
The is-modified-method-name element points to the method that is called before the EJB is stored. The specified method must return a Boolean value, where True indicates that the EJB has been modified, and False indicates that it has not been modified and should not be saved. The is-modified-method-name element applies to stateful session beans or entity beans using container managed persistence, but not to stateless session beans.

When using the is-modified-method-name element:

- For stateful session beans, if the specified method returns False, the WebLogic Enterprise EJB container does not attempt to update the stored state of the bean when the ejbPassivate method is called.

- For entity beans using container-managed persistence, if the specified method returns False, the WebLogic Enterprise EJB container does not attempt to update the stored state of the bean when the ejbStore method is called.

- For entity beans using bean-managed persistence, the is-modified-method-name element should not be specified in the deployment descriptor.

If the is-modified-method-name element is not specified, WebLogic Enterprise always assumes that the EJB has been modified and saves it.

Providing a method and setting the method appropriately improves performance by eliminating unnecessary writing of the bean to the database or file system. When you are using container-managed persistence for entity beans or stateful session beans, the WebLogic Enterprise EJB container must write the contents of the EJB. The is-modified-method-name element provides a mechanism for telling the container when the bean has and has not been changed. Otherwise, the container has no way of knowing that the EJB’s contents have not been modified during the usage of the EJB.

However, this check for a modified EJB assumes that no other client has modified the entity either from WebLogic Enterprise or directly to the database. The extension should be used only as an optimizer for specific situations; otherwise, a loss of data integrity can result.
This optimization works by taking advantage of what Bean Providers know about the logic of the business methods to reduce the frequency with which beans are passivated. If all the methods of your bean change the value of the bean (and therefore the container should save the contents after each invocation), this optimization is not appropriate. There are other circumstances (such as shared databases) in which you also should not use this optimization.

Using this optimization will not degrade the portability of your EJB, because it simply adds a method and a transient value that other EJB containers ignore.

This is an advanced use of EJBs. Only programmers who have an in-depth understanding of EJB and databases should use this optimization.

<table>
<thead>
<tr>
<th>EJ B Role</th>
<th>Bean Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used In</td>
<td>persistence-store-descriptor</td>
</tr>
</tbody>
</table>

**jndi-name**

**Synopsis**
Specifies the name for the EJB home that is registered in the Global WebLogic Enterprise JNDI namespace.

**Syntax**
```xml
<!ELEMENT jndi-name (#PCDATA)>
```

**Description**
To allow Java clients to perform a lookup on the JNDI name for the EJB home and gain access to the object, the name of the EJB home must be registered in the Global WebLogic Enterprise JNDI namespace.

This name can differ from the ejb-name element specified in the standard EJB XML. Whereas the ejb-name element must be unique only among the names of the EJBs in the same EJB JAR file, however, the JNDI name must be unique among all global factory names in a WebLogic Enterprise domain, including EJB homes, CORBA factories, and RMI-named objects.

<table>
<thead>
<tr>
<th>EJ B Role</th>
<th>Bean Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used In</td>
<td>weblogic-deployment-params</td>
</tr>
</tbody>
</table>

**Example**
```xml
<weblogic-ejb-extensions>
  <weblogic-enterprise-bean>
    <ejb-name>Trader</ejb-name>
  </weblogic-enterprise-bean>
</weblogic-ejb-extensions>
```
**module-initializer-class**

**Synopsis**
Specifies the server implementation class for handling the `initialize` and `release` methods.

**Syntax**
```xml
<!ELEMENT module-initializer-class
  (module-initializer-class-name)>  
```

**Description**
The syntax for specifying a Server implementation class for handling the `initialize` and `release` methods is specified in the EJB XML deployment descriptor extensions. These XML elements are processed by the `ejbc` command. The WebLogic Enterprise EJB framework parses the XML at run time and performs the startup/shutdown processing.

For a complete description of startup and shutdown handling in WebLogic Enterprise, see “Starting and Shutting Down Applications” in the Administration Guide.

**EJB Role**
Bean Provider

**Used In**
`weblogic-ejb-extensions`

**Example**
```xml
<weblogic-ejb-extensions>
...

  <module-initializer-class>
    <module-initializer-class-name>ServerImpl
  </module-initializer-class>
</weblogic-ejb-extensions>
```
persistence-store-descriptor

Synopsis  Specifies the type of persistence storage (in a flat file or a JDBC-managed database) for container-managed persistence.

Syntax  

```
<!ELEMENT persistence-store-descriptor (description?,
is-modified-method-name?,
(persistence-store-file | persistence-store-jdbc)?)+>
```

Description  For container-managed persistence, WebLogic Enterprise EJB container supports storing an EJB in either of the following formats:

- a flat file (the `persistence-store-file` element).
- a database that is managed with a WebLogic Enterprise JDBC connection pool (the `persistence-store-jdbc` element).

Note:  Database persistence is **required** for an EJB to fully cooperate in a WebLogic Enterprise global transaction. File persistence is intended primarily for development and prototyping.

The standard EJB XML descriptor provides XML elements for specifying which fields in the entity bean are to be persisted (the `cmp-field` element), and information about the primary key. The WebLogic Enterprise XML extensions provide additional information for mapping to the persistent store.

The code for implementing the persistence is generated by the EJB compiler, based on the bean’s deployment descriptors.

EJB Role  Bean Provider

Used In  `weblogic-deployment-params`

Example  Using JDBC persistence, the following sample XML code shows how to define the automatic saving of two fields (`accountId`, `balance`) in an entity bean to a database table (`ejbAccounts`) using a connection pool (`ejbPool`):

```xml
<weblogic-ejb-extensions>
    <weblogic-enterprise-bean>
        ...
    </weblogic-enterprise-bean>
</weblogic-ejb-extensions>
```
Description of WebLogic Enterprise EJB Extension Elements

<persistence-store-directory-root>

Synopsis If persistent storage is kept in a flat file, specifies the directory root for storing all instances of an EJB. Each <ejb-name> element is converted to a directory name.

Syntax  

```xml
<!ELEMENT persistence-store-directory-root (#PCDATA)>```

EJB XML Reference 2-13
For EJBs with container-managed persistence, stateful session beans are stored using file persistence when they are passivated. The `persistence-store-directory-root` element defines the storage location for these instances. By default, the root directory is `pstore`, which WebLogic Enterprise creates in the directory where the JavaServer is run.

The default filename for a passivated instance is the fully qualified name of the bean, with underscores replacing the periods, plus additional information to ensure uniqueness. For example, if `persistence-store-directory-root` is `c:\weblogic\mystore`, all instances of the EJB `examples.ejb.AccountBean` might be stored in:

```
c:\weblogic\mystore\examples_ejb_AccountBean\SRVGRP_GROUPX_SRVID_Y_Z_examples.ejb.AccountBean_someBigNumber.db
```

**EJB Role**  
Bean Provider

**Used In**  
persistence-store-file

### persistence-store-file

**Synopsis**  
Specifies persistent storage in a flat file.

**Syntax**  
```xml
<!ELEMENT persistence-store-file (description?, persistence-store-directory-root)>
```

**Description**  
File persistence for EJBs is intended primarily for development and prototyping. For an EJB to fully cooperate in a WebLogic Enterprise global transaction, use `persistence-store-jdbc` instead.

**EJB Role**  
Bean Provider

**Used In**  
persistence-store-descriptor

### persistence-store-jdbc

**Synopsis**  
Specifies persistent storage in a database that is managed by a WebLogic Enterprise JDBC connection pool.

**Syntax**  
```xml
<!ELEMENT persistence-store-jdbc (description?, pool-name, table-name, attribute-map, finder-descriptor*)>
```
Description of WebLogic Enterprise EJB Extension Elements

Description
EJB instances are stored in a database that has been previously set up with the JDBC connection pool specified in the pool-name element. The table-name and attribute-map elements map the EJB fields to the appropriate table columns in the database.

EJB Role
Bean Provider

Used In
persistence-store-descriptor

pool-name

Synopsis
Specifies the name of the JDBC connection pool associated with the database used to store EJB state information.

Syntax
<!ELEMENT pool-name (#PCDATA)>

EJB Role
Bean Provider

Used In
persistence-store-jdbc

query-grammar

Synopsis
Specifies the query expression for a finder method.

Syntax
<!ELEMENT query-grammar (#PCDATA)>

Description
The query grammar syntax has an expression prefix notation of:

[operator operand1 operand2]

Valid operators include:
- ( ) = < > <= >= / ! & |
- like
- isNull
- isNotNull
- orderBy
- ! is the Boolean not
- & is the Boolean and
- | is the Boolean or
Valid operands include:
- Another expression
- An EJB field
- A finder method parameter (when preceded by $)

EJB Role
Bean Provider

Examples
The following examples show sample query expressions in finder methods.

The following expression determines whether balance exceeds amount:

"(> balance $amount)"

The following example shows the use of compound expressions. Note that single quotes are used to denote strings:

"{& (> bal $amount) {! (= accountType 'checking')}"

The following example is a method signature and an expression that is converted to a findBigAccounts method in the generated persistent storage class. The method returns an enumeration of beans of this class that satisfy this expression:

"findBigAccounts(double amount)" "(> balance $amount)"

The following example finds all the EJ Bs in a table:

"findAll()" "(= 1 1)"

The following example finds all EJ Bs whose lastName fields starts with M:

"findNamesStartingWithM()" "(like lastName M%)"

The following example finds all EJ Bs whose firstName field is null:

"findNullFirstNames()" "(isNull firstName)"

The following example finds all EJ Bs whose firstName field is not null:

"findNullFirstNames()" "(isNotNull firstName)"

The following expression finds all EJ Bs in which the value of the balance field exceeds balanceGreaterThan and returns the beans ordered by the id database column. Notice that this expression should all be contained in a single line and that no column mapping is performed. The database column must be specified directly:
"findOrderedAccounts(double balanceGreaterThan)"
"(orderBy 'id' (> balance $balanceGreaterThan))"

The following expression is identical to the previous one except that it returns the beans in descending order:

"findOrderedAccounts(double balanceGreaterThan)"
"(orderBy 'id desc' (> balance $balanceGreaterThan))"

**table-column-name**

**Synopsis**
Specifies the name of a table column in an attribute map entry.

**Syntax**
```xml
<!ELEMENT table-column-name (#PCDATA)>
```

**EJB Role**
Bean Provider

**Used In**
attribute-map-entry

**table-name**

**Synopsis**
Specifies the name of the table used to store EJB state information.

**Syntax**
```xml
<!ELEMENT table-name (#PCDATA)>
```

**EJB Role**
Bean Provider

**Used In**
persistence-store-jdbc

**trans-timeout-seconds**

**Synopsis**
Sets the default timeout, in seconds, for container-initiated transactions for this bean. If a transaction duration exceeds this value, the transaction will be rolled back.

**Syntax**
```xml
<!ELEMENT trans-timeout-seconds (#PCDATA)>
```

**EJB Role**
Bean Provider

**Used In**
weblogic-deployment-params

**weblogic-deployment-params**

**Synopsis**
Specifies deployment options for the WebLogic Server.
weblogic-deployment-params

Syntax
<!ELEMENT weblogic-deployment-params (is-cacheable?,
trans-timeout-seconds?, jndi-name?,
persistence-store-descriptor?)>

EJB Role  Bean Provider

Used In  weblogic-enterprise-bean

weblogic-ejb-extensions

Synopsis  Root element of the WebLogic Enterprise XML extensions.

Syntax
<!ELEMENT weblogic-ejb-extensions (description?,
weblogic-version?, weblogic-enterprise-bean*,
module-initializer-class*)>

EJB Role  Bean Provider

weblogic-enterprise-bean

Synopsis  Specifies deployment information for a bean that is specific to WebLogic Enterprise.

Syntax
<!ELEMENT weblogic-enterprise-bean (ejb-name,
weblogic-deployment-params?)>

EJB Role  Bean Provider

Used In  weblogic-ejb-extensions

weblogic-version

Synopsis  Specifies the WebLogic Enterprise version.

Syntax
<!ELEMENT weblogic-version (#PCDATA)>

EJB Role  Bean Provider

Used In  weblogic-ejb-extensions
Index

A
assembly-descriptor element 1-8
attribute-map element 2-6
attribute-map-entry element 2-6

B
bean-field-name element 2-6

cmp-field element 1-8
container-transaction element 1-8
customer support contact information ix

deployment descriptor
  about deployment descriptors 1-2
  assembly-descriptor element 1-8
  attribute-map element 2-6
  attribute-map-entry element 2-6
  bean-field-name element 2-6
  cmp-field element 1-8
  container-transaction element 1-8
  description element 1-8
display-name element 1-9
DOCTYPE declaration 1-3
EJB roles 1-2
ejb-class element 1-9
ejb-jar element 1-10
ejb-link element 1-10
ejb-name element 1-10
ejb-ref element 1-10
ejb-ref-name element 1-11
ejb-ref-type element 1-11
enterprise-beans element 1-11
entity element 1-12
ev-entry element 1-12
ev-entry-name element 1-12
ev-entry-type element 1-13
ev-entry-value element 1-13
extensions (see extensions) 2-2
field-name element 1-13
file formats 1-3
finder-descriptor element 2-7
home element 1-14
ID mechanism 1-22
is-cacheable element 2-8
is-modified-method-name element 2-9
jndi-name element 2-10
large-icon element 1-14
method element 1-14
method-intf element 1-14
method-name element 1-15
method-param element 1-15
method-params element 1-15
method-permission element 1-16
module-initializer-class element 2-11
module-initializer-class-name element 2-11
persistence-store element 2-12
persistence-store-directory-root element

EJB XML Reference 1
EJB roles 1-2

description element 1-8
display-name element 1-9
DOCTYPE declaration
deployment descriptor 1-3
extensions 2-2
documentation, where to find it viii

EJB roles 1-2
field-name element 1-13
file formats 1-3
finder-descriptor element 2-7

home element 1-14

ID mechanism 1-22
is-cacheable element 2-8
is-modified-method-name element 2-9

jndi-name element 2-10

large-icon element 1-14

method element 1-14
method-intf element 1-14
method-name element 1-15
method-param element 1-15
method-params element 1-15
method-permission element 1-16
module-initializer-class element 2-11
module-initializer-class-name element 2-11

prim-key-class element 1-16
primkey-field element 1-17
printing product documentation viii

query-grammar element 2-15

reentrant element 1-17
related information viii
remote element 1-17
res-auth element 1-18
resource-ref element 1-18
res-ref-name element 1-18
res-type element 1-18
role-link element 1-19
role-name element 1-19

security-role element 1-19
security-role-ref element 1-19
session element 1-20
session-type element 1-20
small-icon element 1-20
support
   technical ix

table-column-name element 2-17
table-name element 2-17
transaction-type element 1-21
trans-attribute element 1-21
trans-timeout-seconds element 2-17

weblogic-deployment-params element 2-17
weblogic-ejb-extensions element 2-18
weblogic-enterprise-bean element 2-18
weblogic-version element 2-18

X
XML extensions 2-2