

Java™ 2 Platform, Enterprise Edition Management Specification

JSR-77

Hans Hrasna

Sun Microsystems, Inc.

<http://jcp.org/jsr/detail/77.jsp>
jsr-77-comments@jcp.org

Specification: JSR-000077 J2EE(tm) Management ("Specification")

Version: 1.1

Status: Maintenance Release

Release: 8 May 2006

Copyright 2006 SUN MICROSYSTEMS, INC.
4150 Network Circle, Santa Clara, California 95054, U.S.A
All rights reserved.

LIMITED LICENSE GRANTS

1. License for Evaluation Purposes. Sun hereby grants you a fully-paid, non-exclusive, non-transferable, worldwide, limited license (without the right to sublicense), under Sun's applicable intellectual property rights to view, download, use and reproduce the Specification only for the purpose of internal evaluation. This includes (i) developing applications intended to run on an implementation of the Specification, provided that such applications do not themselves implement any portion(s) of the Specification, and (ii) discussing the Specification with any third party; and (iii) excerpting brief portions of the Specification in oral or written communications which discuss the Specification provided that such excerpts do not in the aggregate constitute a significant portion of the Specification.

2. License for the Distribution of Compliant Implementations. Sun also grants you a perpetual, non-exclusive, non-transferable, worldwide, fully paid-up, royalty free, limited license (without the right to sublicense) under any applicable copyrights or, subject to the provisions of subsection 4 below, patent rights it may have covering the Specification to create and/or distribute an Independent Implementation of the Specification that: (a) fully implements the Specification including all its required interfaces and functionality; (b) does not modify, subset, superset or otherwise extend the Licensor Name Space, or include any public or protected packages, classes, Java interfaces, fields or methods within the Licensor Name Space other than those required/authorized by the Specification or Specifications being implemented; and (c) passes the Technology Compatibility Kit (including satisfying the requirements of the applicable TCK Users Guide) for such Specification ("Compliant Implementation"). In addition, the foregoing license is expressly conditioned on your not acting outside its scope. No license is granted hereunder for any other purpose (including, for example, modifying the Specification, other than to the extent of your fair use rights, or distributing the Specification to third parties). Also, no right, title, or interest in or to any trademarks, service marks, or trade names of Sun or Sun's licensors is granted hereunder. Java, and Java-related logos, marks and names are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

3. Pass-through Conditions. You need not include limitations (a)-(c) from the previous paragraph or any other particular "pass through" requirements in any license You grant concerning the use of your Independent Implementation or products derived from it. However, except with respect to Independent Implementations (and products derived from them) that satisfy limitations (a)-(c) from the previous paragraph, You may neither: (a) grant or otherwise pass through to your licensees any licenses under Sun's applicable intellectual property rights; nor (b) authorize your licensees to make any claims concerning their implementation's compliance with the Specification in question.

4. Reciprocity Concerning Patent Licenses.

a. With respect to any patent claims covered by the license granted under subparagraph 2 above that would be infringed by all technically feasible implementations of the Specification, such license is conditioned upon your offering on fair, reasonable and non-discriminatory terms, to any party seeking it from You, a perpetual, non-exclusive, non-transferable, worldwide license under Your patent rights which are or would be infringed by all technically feasible implementations of the Specification to develop, distribute and use a Compliant Implementation.

b. With respect to any patent claims owned by Sun and covered by the license granted under subparagraph 2, whether or not their infringement can be avoided in a technically feasible manner when implementing the Specification, such license shall terminate with respect to such claims if You initiate a claim against Sun that it has, in the course of performing its responsibilities as the Specification Lead, induced any other entity to infringe Your patent rights.

c. Also with respect to any patent claims owned by Sun and covered by the license granted under subparagraph 2 above, where the infringement of such claims can be avoided in a technically feasible manner when implementing the Specification such license, with respect to such claims, shall terminate if You initiate a claim against Sun that its making, having made, using, offering to sell, selling or importing a Compliant Implementation infringes Your patent rights.

5. Definitions. For the purposes of this Agreement: "Independent Implementation" shall mean an implementation of the Specification that neither derives from any of Sun's source code or binary code materials nor, except with an appropriate and separate license from Sun, includes any of Sun's source code or binary code materials; "Licensor Name Space" shall mean the public class or interface declarations whose names begin with "java", "javax", "com.sun" or their equivalents in any subsequent naming convention adopted by Sun through the Java Community Process, or any recognized successors or replacements thereof; and "Technology Compatibility Kit" or "TCK" shall mean the test suite and accompanying TCK User's Guide provided by Sun which corresponds to the Specification and that was available either (i) from Sun's 120 days before the first release of Your Independent Implementation that allows its use for commercial purposes, or (ii) more recently than 120 days from such release but against which You elect to test Your implementation of the Specification.

This Agreement will terminate immediately without notice from Sun if you breach the Agreement or act outside the scope of the licenses granted above.

DISCLAIMER OF WARRANTIES

THE SPECIFICATION IS PROVIDED "AS IS". SUN MAKES NO REPRESENTATIONS OR WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT (INCLUDING AS A CONSEQUENCE OF ANY PRACTICE OR IMPLEMENTATION OF THE SPECIFICATION), OR THAT THE CONTENTS OF THE SPECIFICATION ARE SUITABLE FOR ANY PURPOSE. This document does not represent any commitment to release or implement any portion of the Specification in any product. In addition, the Specification could include technical inaccuracies or typographical errors.

LIMITATION OF LIABILITY

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION, LOST REVENUE, PROFITS OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED IN ANY WAY TO YOUR HAVING, IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION, EVEN IF SUN AND/OR ITS LICENSORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You will indemnify, hold harmless, and defend Sun and its licensors from any claims arising or resulting from: (i) your use of the Specification; (ii) the use or distribution of your Java application, applet and/or implementation; and/or (iii) any claims that later versions or releases of any Specification furnished to you are incompatible with the Specification provided to you under this license.

RESTRICTED RIGHTS LEGEND

U.S. Government: If this Specification is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in the Software and accompanying documentation shall be only as set forth in this license; this is in accordance with 48 C.F.R. 227.7201 through 227.7202-4 (for Department of Defense (DoD) acquisitions) and with 48 C.F.R. 2.101 and 12.212 (for non-DoD acquisitions).

REPORT

If you provide Sun with any comments or suggestions concerning the Specification ("Feedback"), you hereby: (i) agree that such Feedback is provided on a non-proprietary and non-confidential basis, and (ii) grant Sun a perpetual, non-exclusive, worldwide, fully paid-up, irrevocable license, with the right to sublicense through multiple levels of sublicensees, to incorporate, disclose, and use without limitation the Feedback for any purpose.

GENERAL TERMS

Any action related to this Agreement will be governed by California law and controlling U.S. federal law. The U.N. Convention for the International Sale of Goods and the choice of law rules of any jurisdiction will not apply.

The Specification is subject to U.S. export control laws and may be subject to export or import regulations in other countries. Licensee agrees to comply strictly with all such laws and regulations and acknowledges that it has the responsibility to obtain such licenses to export, re-export or import as may be required after delivery to Licensee.

This Agreement is the parties' entire agreement relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, conditions, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification to this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

Contents

JSR77.1 Introduction	9
JSR77.1.1 Overview	9
JSR77.1.2 Scope	10
JSR77.1.3 Organization	10
JSR77.1.4 Acknowledgments	11
JSR77.2 J2EE Management Model	13
JSR77.2.1 Overview	13
JSR77.2.2 Elements of the J2EE Management Model	14
JSR77.2.3 Interpreting the UML Diagrams	14
JSR77.2.3.1 Supplier Cardinality	15
JSR77.2.3.2 J2EEApplication managed object example	16
JSR77.2.4 J2EE Product Provider's Responsibilities	18
JSR77.2.5 Extending the managed object models	18
JSR77.2.5.1 Vendor-specific attributes and operations	18
JSR77.2.5.2 Vendor-specific managed object types	18
JSR77.3 Managed Objects	19
JSR77.3.1 J2EEManagedObject	20
JSR77.3.1.1 Attribute Detail	21
JSR77.3.2 J2EEDomain extends J2EEManagedObject	26
JSR77.3.2.1 Attribute Detail	26
JSR77.3.3 J2EEServer extends J2EEManagedObject	27
JSR77.3.3.1 Attribute Detail	28
JSR77.3.4 JVM extends J2EEManagedObject	30
JSR77.3.4.1 Attribute Detail	30
JSR77.3.5 J2EEDeployedObject extends J2EEManagedObject	31
JSR77.3.6 J2EEApplication extends J2EEDeployedObject	31
JSR77.3.6.1 Attribute Detail	32
JSR77.3.7 J2EEModule extends J2EEDeployedObject	32
JSR77.3.7.1 Attribute Detail	33
JSR77.3.8 AppClientModule extends J2EEModule	33
JSR77.3.9 EJBModule extends J2EEModule	33

JSR77.3.9.1 Attribute Detail	33
JSR77.3.10EJB extends J2EEManagedObject	34
JSR77.3.11MessageDrivenBean extends EJB	34
JSR77.3.12EntityBean extends EJB	34
JSR77.3.13SessionBean extends EJB	34
JSR77.3.14StatefulSessionBean extends SessionBean	35
JSR77.3.15StatelessSessionBean extends SessionBean	35
JSR77.3.16WebModule extends J2EEModule	35
JSR77.3.16.1 Attribute Detail	35
JSR77.3.17Servlet extends J2EEManagedObject	35
JSR77.3.18ResourceAdapterModule extends J2EEModule	35
JSR77.3.18.1 Attribute Detail	36
JSR77.3.19ResourceAdapter extends J2EEManagedObject	36
JSR77.3.19.1 Attribute Detail	36
JSR77.3.20J2EEResource extends J2EEManagedObject	37
JSR77.3.21JavaMailResource extends J2EEResource	37
JSR77.3.22JCAResource extends J2EEResource	38
JSR77.3.22.1 Attribute Detail	38
JSR77.3.23JCAConnectionFactory extends J2EEManagedObject	38
JSR77.3.23.1 Attribute Detail	39
JSR77.3.24JCAManagedConnectionFactory extends J2EEManagedObject	39
JSR77.3.25JDBCResource extends J2EEResource	39
JSR77.3.25.1 Attribute Detail	40
JSR77.3.26JDBCDataSource extends J2EEManagedObject	40
JSR77.3.26.1 Attribute Detail	40
JSR77.3.27JDBCDriver extends J2EEManagedObject	40
JSR77.3.28JMSResource extends J2EEResource	41
JSR77.3.29JNDIResource extends J2EEResource	41
JSR77.3.30JTAResource extends J2EEResource	41
JSR77.3.31RMI_IIOResource extends J2EEResource	41
JSR77.3.32URLResource extends J2EEResource	41
JSR77.4 Events	43
JSR77.4.1 EventProvider	43
JSR77.4.1.1 Attribute Detail	43
JSR77.4.2 Notification	44

JSR77.4.2.1 Attribute Detail	45
JSR77.4.3 Event Notification.	47
JSR77.4.3.1 javax.management.Notification.	47
JSR77.5 State Management	49
JSR77.5.1 StateManageable.	49
JSR77.5.1.1 Attribute Detail	50
JSR77.5.1.2 Operation Detail	51
JSR77.6 Performance Monitoring	53
JSR77.6.1 Performance Data Framework	53
JSR77.6.2 StatisticsProvider	53
JSR77.6.2.1 Attribute Detail	54
JSR77.6.3 package javax.management.j2ee.statistics	55
JSR77.6.4 Statistic	55
JSR77.6.4.1 Operation Detail	56
JSR77.6.5 CountStatistic extends Statistic.	57
JSR77.6.5.1 Operation Detail	57
JSR77.6.6 TimeStatistic extends Statistic	57
JSR77.6.6.1 Operation Detail	57
JSR77.6.7 RangeStatistic extends Statistic	58
JSR77.6.7.1 Operation Detail	58
JSR77.6.8 BoundaryStatistic extends Statistic.	59
JSR77.6.8.1 Operation Detail	59
JSR77.6.9 BoundedRangeStatistic extends RangeStatistic, BoundaryStatistic	60
JSR77.6.9.1 Operation Detail	60
JSR77.6.10Stats.	61
JSR77.6.10.1 Operation Detail	61
JSR77.6.11EJBStats extends Stats	62
JSR77.6.11.1 Operation Detail	62
JSR77.6.12EntityBeanStats extends EJBStats	63
JSR77.6.12.1 Operation Detail	63
JSR77.6.13MessageDrivenBeanStats extends EJBStats.	63
JSR77.6.13.1 Operation Detail	63
JSR77.6.14SessionBeanStats extends EJBStats	64
JSR77.6.14.1 Operation Detail	64
JSR77.6.15StatefulSessionBeanStats extends SessionBeanStats	64

JSR77.6.15.1 Operation Detail	64
JSR77.6.16StatelessSessionBeanStats extends SessionBeanStats	64
JSR77.6.17JavaMailStats extends Stats	65
JSR77.6.17.1 Operation Detail	65
JSR77.6.18JCAStats extends Stats	66
JSR77.6.18.1 Operation Detail	66
JSR77.6.19JCAConnectionStats extends Stats	67
JSR77.6.19.1 Operation Detail	67
JSR77.6.20JCAConnectionPoolStats extends JCAConnectionStats	68
JSR77.6.20.1 Operation Detail	68
JSR77.6.21JDBCStats extends Stats	69
JSR77.6.21.1 Operation Detail	69
JSR77.6.22JDBCConnectionStats extends Stats	70
JSR77.6.22.1 Operation Detail	70
JSR77.6.23JDBCConnectionPoolStats extends JDBCConnectionStats	71
JSR77.6.23.1 Operation Detail	71
JSR77.6.24JMSStats extends Stats	72
JSR77.6.24.1 Operation Detail	72
JSR77.6.25JMSSessionStats extends Stats	73
JSR77.6.25.1 Operation Detail	73
JSR77.6.26JMSSessionStats extends Stats	73
JSR77.6.26.1 Operation Detail	73
JSR77.6.27JMSEndpointStats extends Stats	75
JSR77.6.27.1 Operation Detail	75
JSR77.6.28JMSProducerStats extends JMSEndpointStats	76
JSR77.6.28.1 Operation Detail	76
JSR77.6.29JMSConsumerStats extends JMSEndpointStats	76
JSR77.6.29.1 Operation Detail	76
JSR77.6.30JTASStats extends Stats	77
JSR77.6.30.1 Operation Detail	77
JSR77.6.31JVMStats extends Stats	78
JSR77.6.31.1 Operation Detail	78
JSR77.6.32ServletStats extends Stats	79
JSR77.6.32.1 Operation Detail	79
JSR77.6.33URLStats extends Stats	79
JSR77.6.33.1 Operation Detail	79

JSR77.7 J2EE Management EJB Component	81
JSR77.7.1 MEJB Implementation requirements	81
JSR77.7.1.1 JMX requirements	82
JSR77.7.1.2 Remote Interface	82
JSR77.7.1.3 Home Interface	82
JSR77.7.1.4 Naming Requirements	82
JSR77.7.1.5 Event Listener Requirements	83
JSR77.7.2 Using the Management EJB component	84
JSR77.7.2.1 Creating a MEJB Session Object	84
JSR77.7.2.2 Finding managed objects of a certain type	85
JSR77.7.2.3 Retrieving attribute information	85
JSR77.7.2.4 Invoking methods	86
JSR77.7.2.5 Registering an event notification listener	86
JSR77.7.3 package javax.management.j2ee	87
JSR77.7.3.1 Interface ListenerRegistration	87
JSR77.7.3.2 Interface Management	90
JSR77.7.3.3 Interface ManagementHome	98
JSR77.8 J2EE Management SNMP	99
JSR77.8.1 Introduction	99
JSR77.8.2 The SNMP Management Framework	100
JSR77.8.3 Overview	101
JSR77.8.4 The Structure of the MIB	102
JSR77.8.5 Definitions	103
JSR77.8.6 IANA Considerations	181
JSR77.8.7 Security Considerations	181
JSR77.8.8 Intellectual Property	182
JSR77.8.9 Acknowledgments	182
JSR77.8.10 References	183
JSR77.9 J2EE Management CIM	185
JSR77.9.1 Introduction	185
JSR77.9.2 Managed Objects	185
JSR77.9.2.1 J2EEDomain	185
JSR77.9.2.2 J2EEServer	186
JSR77.9.3 Events	187
JSR77.9.3.2 Modeling J2EE Events in CIM	188
JSR77.9.3.3 Handling subscriptions	188
JSR77.9.4 State Management	189

JSR77.9.5 Acknowledgments	189
JSR77.9.6 Appendix	190

CHAPTER JSR77.1

Introduction

The increasing popularity of the Java™ 2 Enterprise Edition (J2EE™) platform has mandated the specification of a common framework for the delivery of Java enterprise management and monitoring services. The goal of JSR77 is to abstract the fundamental manageable aspects of the J2EE architecture to provide a well defined model for implementing instrumentation and information access. In addition, this specification defines a standardized API for interoperating with J2EE™ components that participate in the monitoring and control of the platform's resources.

JSR77.1.1 Overview

This specification defines a management information model for the J2EE platform, the J2EE Management Model. This specification also includes standard mappings of the model to the Common Information Model (CIM), an SNMP Management Information Base (MIB), and to a Java API as a server resident Enterprise JavaBeans™ (EJB™) component, the J2EE Management EJB (MEJB) component. The MEJB component provides interoperable remote access to the model from any standard J2EE application.

JSR77.1.2 Scope

The scope of this version of the specification is limited to the management of J2EE servers, application components that are deployed onto a J2EE server and resources which provide the J2EE Standard Services. The scope of the information and operations required by the specification is limited to that required to enable basic functionality for the following:

- Discovery - discovering and navigating the managed objects on a managed system
- Events - receiving notifications of important events that occur on managed objects
- State - observing and controlling the running state of managed objects
- Performance - monitoring basic performance statistics of managed objects

The following management categories are also considered important but were deemed out of scope for this version of the specification:

- Security management
- Accounting
- Configuration management
- Fault Management/Recovery

JSR77.1.3 Organization

- Chapter 2 introduces the J2EE Management Model and provides definitions of the terms used to describe the model.
- Chapter 3 specifies the structure, format, and semantics of managed objects required by this specification.
- Chapter 4 specifies a standard extensible event model for the J2EE Management Model.
- Chapter 5 specifies a standard state management model for the J2EE Management Model.

- Chapter 6 specifies the performance data requirements of the J2EE Management Model.
- Chapter 7 specifies the J2EE Management EJB component (MEJB).
- Chapter 8 specifies the SNMP MIB mapping of the J2EE Management Model.
- Chapter 9 specifies the CIM mapping of the J2EE Management Model.

JSR77.1.4 Acknowledgments

This specification was developed under the Java Community Process 2.0 as JSR-077 and includes contributions from numerous groups at Sun and at partner companies. We would like to thank the members of the JSR-077 Expert Group in particular for their contributions:

- Altaworks Corporation - Ralph Beck
- BEA Systems - Vadim Draluk, Steve Fanshier
- Borland Software Corporation - Michael Stern, Charlton Barreto
- Flashline - Gary Baney
- Gemstone Systems, Inc. - Chris Jacobson
- IBM - Leigh Williamson
- IONA Technologies - Brian Joyce
- Informix Software - Tony Daniel
- JBoss group - Marc Fluery, Andreas Schaefer
- Laminar Software - Anand Byrappagari
- Lutris Technologies - David Simons
- Pramati Technologies - Ramesh Loganathan
- Sitraka - Alan Armstrong, Doug Doe
- Sun Microsystems Inc. - Rebecca Searls



CHAPTER JSR77.2

J2EE Management Model

This chapter provides an overview of the J2EE Management Model and explains how it is applied to managed systems.

JSR77.2.1 Overview

The J2EE Management Model is a specification of the attributes, operations and architecture of the managed objects required by compliant J2EE platform implementations. The model is designed to be interoperable with a variety of industry standard management systems and protocols.

The Management EJB (MEJB) component (Chapter JSR77.7, “J2EE Management EJB Component”) provides interoperable remote access of the model to J2EE application components. All compliant implementations must support the MEJB component.

This specification also provides standard mappings of the model to the CIM/WBEM Common Information Model (CIM) (Chapter JSR77.9, “J2EE Management CIM”) and the SNMP Management Information Base (MIB) (Chapter JSR77.8, “J2EE Management SNMP”). Implementation support for SNMP and CIM/WBEM is optional.

JSR77.2.2 Elements of the J2EE Management Model

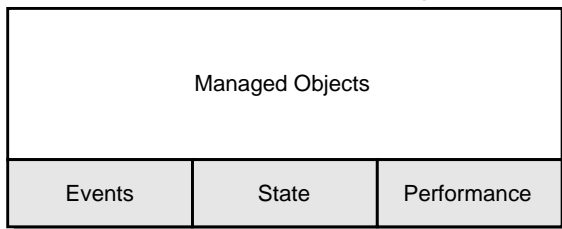


Figure JSR77.2.1 J2EE Management Model

Figure JSR77.2.1 illustrates the elements of the J2EE Management Model. The shaded sections represent standard extensions of the basic required models that each managed object may support. The requirements of the J2EE Management Model are specified by the following chapters:

- Chapter JSR77.3, “Managed Objects” specifies the managed object types required by the J2EE Management Model. The J2EE product provider must implement the managed objects described in this chapter.
- Chapter JSR77.4, “Events” specifies the EventProvider model which specifies the requirements of managed objects that support event notification.
- Chapter JSR77.5, “State Management” specifies the StateManageable model which specifies the requirements of managed objects that support state manipulation.
- Chapter JSR77.6, “Performance Monitoring” specifies the StatisticProvider model which specifies the requirements of managed objects that provide performance data.

The specification of the models includes the UML diagrams as well as the textual specification of the metadata.

JSR77.2.3 Interpreting the UML Diagrams

The elements specified by the managed object UML diagrams consist of:

- A name which identifies the managed object types and supertypes.

The use of supertypes in the managed object model does not imply the capability of polymorphism. Inheritance in the models is used simply as a conve-

nient way to express the attributes and operation requirements shared by a manageable type. The set of standard J2EE managed object types is limited to the sub-types specified by the model at the end of the hierarchy of inheritance of a particular model. The required attributes and operations for each type includes all of the attributes and operations in the super-types in the hierarchy. Table 3-1 on page 23 lists the standard J2EE managed object types.

- Attributes which contain data about the managed object.
 - All specified attributes must implemented as readable and may also be implemented as writable. It is implementation specific which attributes are writable.
 - Attributes which identify managed objects are specified to be of the OBJECT_NAME type. The type identifier OBJECT_NAME refers to a formatted string whose syntax is defined by the specification of the `objectName` attribute of the `J2EEManagedObject` model (see section JSR77.3.1.1.1 on page 21). The OBJECT_NAME string type provides a unique identifier for a managed object within a J2EE management domain. Every managed object has a name attribute whose value complies with the OBJECT_NAME syntax.
- Operations which describe the functions required by a managed object type. All operations specified for a managed object type must be invocable by a management client.

JSR77.2.3.1 Supplier Cardinality

In each of the attributes in the model that provide a list of managed objects there is also a specification of supplier cardinality. Supplier cardinality specifies the minimum and maximum number of items that can be expected to exist in a list. Supplier cardinality is specified as:

min..max

Where *min* can be any whole number and *max* can be any whole number or an asterisk '*', which indicates that there is no maximum limit.

JSR77.2.3.2 J2EEApplication managed object example

The following example illustrates how the J2EE Management Model specification of the J2EEApplication type translates to accessing a J2EEApplication managed object instance through an implementation of the J2EE Management EJB(MEJB) component.

The specification for the J2EEApplication managed object type (section JSR77.3.6 on page 31) indicates that J2EEApplication extends J2EEDeployedObject which extends J2EEManagedObject (section JSR77.3.1 on page 20).

This indicates that a managed object instance of the J2EEApplication type must support at least the following attributes:

- **OBJECT_NAME** `objectName`
- **boolean** `stateManageable`
- **boolean** `statisticsProvider`
- **boolean** `eventProvider`
- **String** `deploymentDescriptor`
- **OBJECT_NAME** `server`
- **OBJECT_NAME[]** `modules`

If any of the boolean attributes is true it indicates that this J2EEApplication managed object also includes the additional required attributes and operations of the respective models `StateManageable`, `StatisticsProvider`, or `EventProvider`.

Code Example 2.2 shows how a J2EE component accesses the attributes of a managed object of the J2EEApplication type through the MEJB component.

Code Example 2.2

```
// Note: applicationName is the javax.management.ObjectName for the application
//
// get the value of an attribute which is a primitive type
// in this case, the deployment descriptor for the application
String deploymentDescriptor = (String) mejb.getAttribute(applicationName,
"deploymentDescriptor");

// get the OBJECT_NAMEs of the modules for this application
String [] modules = (String[])mejb.getAttribute(applicationName, "modules");
```

After the component has the array of OBJECT_NAME strings for all of the J2EEModule managed objects associated with this application, they can be used to access the attributes and operations of the modules managed objects the same way they were used to accessed the application managed object:

Code Example 2.3

```
// get the deployment descriptors for the modules
String[] moduleDeploymentDescriptors = new String[modules.length];
for (int i=0; i < modules.length; i++) {
    try {
        ObjectName moduleName = new ObjectName(modules[i]);
        moduleDeploymentDescriptors[i] = (String) mejb.getAttribute(moduleName,
"deploymentDescriptor");
    } catch (MalformedObjectNameException e) {
    }
}
```

JSR77.2.4 J2EE Product Provider's Responsibilities

The J2EE Product Provider has the following responsibilities:

- Provide the instrumentation required by the J2EE Management Model. The J2EE Product Provider must provide a J2EE product that provides an accurate representation of the metadata required by this specification.
- Provide an implementation of the J2EE Management EJB (MEJB) component. The J2EE Product Provider must provide a compliant implementation of the MEJB component. The MEJB component may be automatically deployed during server installation. A compliant J2EE Product must deploy an MEJB component before installation of that product can be considered complete.

JSR77.2.5 Extending the managed object models

JSR77.2.5.1 Vendor-specific attributes and operations

J2EE Product Providers may extend the functionality of any of the models specified by the J2EE Management Model by adding vendor-specific attributes, operations and associations. The attributes, operations and associations specified by the models must always be supported exactly as specified.

JSR77.2.5.2 Vendor-specific managed object types

In addition to the standard specified managed object types, J2EE Product Providers may include vendor-specific managed object types in the management system implementation. Vendor-specific managed object type models must extend the `J2EEManagedObject` type and support the attributes required by the `J2EEManagedObject` model. The standard managed object types must always be supported exactly as specified.

CHAPTER JSR77.3

Managed Objects

The term *managed object* refers to the definition of a unit of management information. Management instrumentation provides the “glue” which takes the information available on an entity to be managed and makes it appear as a collection of managed objects.

This chapter contains the models and metamodels that specify the format, semantics and relationship of the managed objects required by all compliant implementations of this specification.

Although the diagrams and textual descriptions that specify the managed object types closely resemble Java classes, they are not specifications of Java class types or Java class inheritance hierarchies and do not represent requirements of the class names or class hierarchies of a particular implementation.

JSR77.3.1 J2EEManagedObject

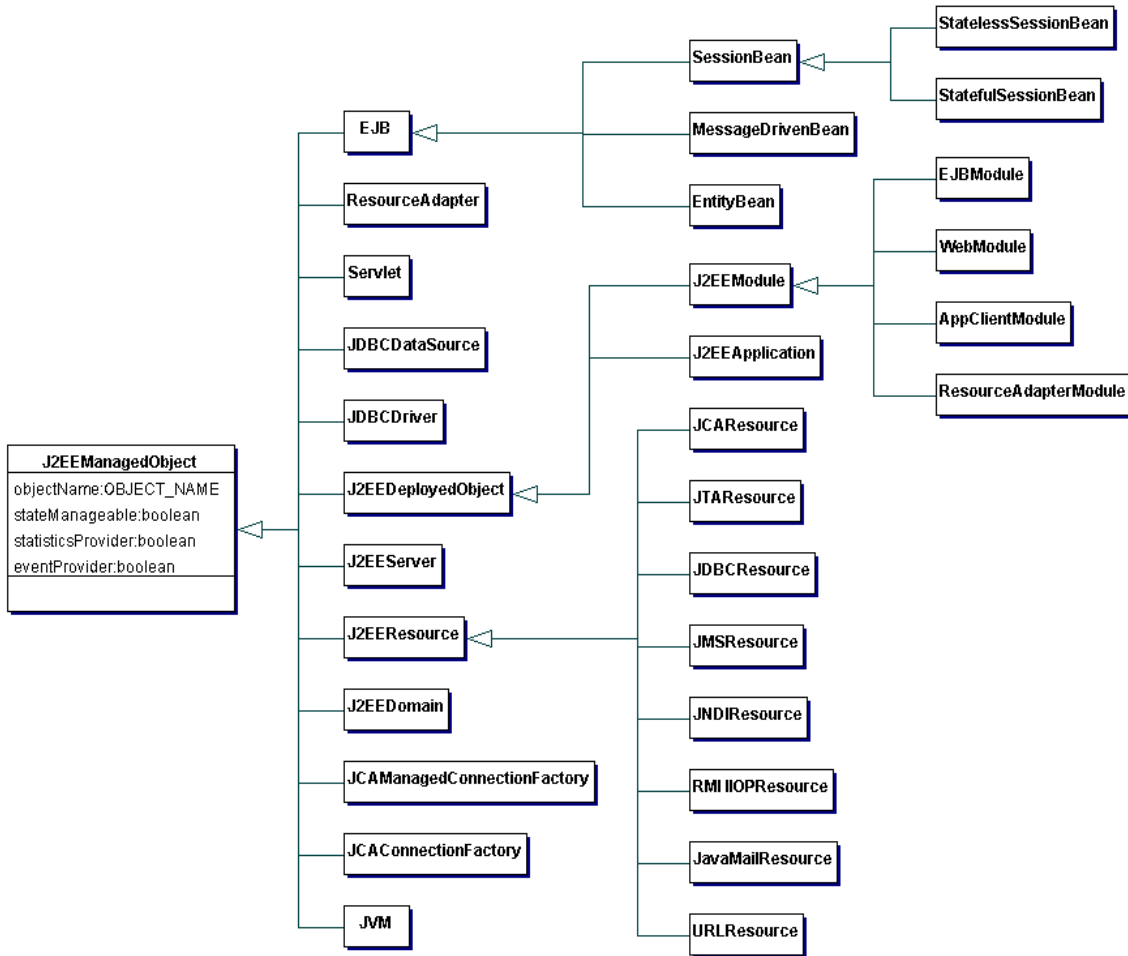


Figure JSR77.3.1 J2EEManagedObject detail with inheritance overview

The J2EEManagedObject model is the base model of all managed objects in the J2EE Management Model. All managed objects in the J2EE platform must include the attributes of the J2EEManagedObject model. All managed objects must have a unique name.

Any managed object may optionally implement the metadata specified by the StateManageable model (see “StateManageable” on page 49), which enables the managed object to provide state management operations.

Any managed object may optionally implement the metadata specified by the `StatisticsProvider` model (see “`StatisticsProvider`” on page 53), which provides access to the performance statistics for the managed object.

Any managed object may optionally implement the `EventProvider` model, which indicates it emits events that implement the `Notification` model (see “`EventProvider`” on page 43).

JSR77.3.1.1 Attribute Detail

JSR77.3.1.1.1 objectName

OBJECT_NAME objectName

The object name of the managed object. The `objectName` attribute is of the type `OBJECT_NAME` which is a string that complies with the syntax specified for a `J2EEManagedObject` name below. The `objectName` attribute must not be null. The value of `objectName` must be unique within the management domain. Management applications use this value to identify managed objects, for example identifying the source of events.

The `J2EEManagedObject` `objectName` consists of two parts:

- A domain name
- An unordered set of key properties, which must include the `j2eeType`, `name` and `<parent-j2eeType>` key properties.

The `J2EEManagedObject` name has the following syntax:

```
[domainName];j2eeType=value,name=value,<parent-j2eeType>[,property=value]*
```

JSR77.3.1.1.1.1 Domain Name

The domain name is a case-sensitive string. It provides a structure for the naming space within a J2EE management domain. The manner in which the domain name is structured is application-dependent. The domain name string may contain any characters, including spaces, except those which are object name separators or wildcards, namely the colon (:), comma (,), equal sign (=), asterisk (*) or question mark (?). The domain name is treated as a whole, therefore any semantic sub-definitions within the string are opaque to a J2EE Management implementation.

JSR77.3.1.1.1.2 Key Property List

The key property list enables you to assign unique names to the managed objects of a given domain. A key property is a property-value pair, where the property does not need to correspond to an actual attribute of a managed object. The key property list must contain at least the mandatory key properties. It may contain any number of optional key properties whose order is not significant. Keys and values are subject to the same character restrictions as domain names.

JSR77.3.1.1.1.3 Mandatory Key Properties

The following key properties are required in the key property list for every managed object name:

- **j2eeType** - specifies the J2EE type of the managed object. The value for `j2eeType` must be a valid standard J2EEManagedObject type. Table 3-1 on page 23 lists the standard types. Managed objects with vendor-specific types are permitted to coexist in the management system. The capabilities of managed objects with vendor-specific `j2eeType` values are implementation specific and are not governed by this specification.
- **name** - specifies the name of the J2EEManagedObject. (for example, `j2eeType=Servlet,name=My Cool Servlet`). The value of the name key property is arbitrary to the vendors implementation with the following exception:

The value of the name key property for managed object instances of the J2EEDomain type must be equivalent to the domain name of the domain it manages.

- **<parent-j2eeType>** - specifies key properties which are derived from the valid managed object types defined in Table 3-1 on page 23. The value of a parent-j2eeType key properties is equal to the value of the name key properties of the OBJECT_NAME of the parents of this managed object. These standard key properties provide a simple mechanism to define a hierarchy of managed object names which will be unique and identifiable. The standard key properties are defined as follows:

`<parent-j2eeType>=<parent J2EEManagedObject name>`

For example, for a management domain named “FirstEverBank”, the management system would have a J2EEDomain object whose `objectName` is:

```
FirstEverBank:j2eeType=J2EEDomain,name=FirstEverBank
```

A J2EEServer in this domain could be named as follows:

```
FirstEverBank:j2eeType=J2EEServer,name=BankServer1
```

When an application is deployed into the server, the J2EEApplication could be named as follows:

```
FirstEverBank:j2eeType=J2EEApplication,name=AccountsController,J2EEServer=BankServer1
```

An EJBModule in the J2EEApplication could be named as follows:

```
FirstEverBank:j2eeType=EJBModule,name=BankAccount,J2EEApplication=AccountsController,J2EEServer=BankServer1
```

An Entity Bean in the EJBModule could be named as follows:

```
FirstEverBank:j2eeType=EntityBean,name=Account,EJBModule=BankAccount,J2EEApplication=AccountsController,J2EEServer=BankServer1
```

Note – The value of the J2EEApplication `<parent-j2eeType>` key must be the string “null” if the module or component which name the key is part of was deployed standalone, and not as part of an application EAR file.

Table JSR77.3-1 lists the required `<parent-j2eeType>` key properties for each managed object type.

Table JSR77.3-1 Managed Object j2eeTypes and required <parent-j2eeType> keys

Managed Object j2eeType	required parent-j2eeType keys
J2EEDomain	none
J2EEServer	none
J2EEApplication	J2EEServer
AppClientModule	J2EEServer, J2EEApplication
EJBModule	J2EEServer, J2EEApplication

**Table JSR77.3-1 Managed Object `j2eeTypes` and required
<parent-`j2eeType`> keys**

Managed Object <code>j2eeType</code>	required parent- <code>j2eeType</code> keys
WebModule	J2EEServer, J2EEApplication
ResourceAdapterModule	J2EEServer, J2EEApplication
EntityBean	EJBModule, J2EEApplication, J2EEServer
StatefulSessionBean	EJBModule, J2EEApplication, J2EEServer
StatelessSessionBean	EJBModule, J2EEApplication, J2EEServer
MessageDrivenBean	EJBModule, J2EEApplication, J2EEServer
Servlet	WebModule, J2EEApplication, J2EEServer
ResourceAdapter	ResourceAdapterModule, J2EEApplication, J2EEServer
JavaMailResource	J2EEServer
JCAResource	J2EEServer, ResourceAdapter
JCAConnectionFactory	JCAResource, J2EEServer
JCAManagedConnectionFactory	J2EEServer
JDBCResource	J2EEServer
JDBCDataSource	JDBCResource, J2EEServer
JBCDriver	J2EEServer
JMSResource	J2EEServer
JNDIResource	J2EEServer
JTAResource	J2EEServer
RMI_IIOPResource	J2EEServer
URLResource	J2EEServer
JVM	J2EEServer

JSR77.3.1.1.2 stateManageable

boolean stateManageable

If true, indicates that this managed object implements the StateManageable model and is state manageable by the specification of Chapter JSR77.5, “State Management”. If false, the managed object does not support state management.

JSR77.3.1.1.3 statisticsProvider**boolean statisticsProvider**

If true, indicates that the managed object supports the generation of performance statistics and implements the StatisticsProvider model (see “StatisticsProvider” on page 53).

If false, the J2EEManagedObject does not support performance statistics.

JSR77.3.1.1.4 eventProvider**boolean eventProvider**

If true, indicates that the managed object implements the EventProvider model (see “EventProvider” on page 43) and provides notification about events that occur on that object.

The semantic of the eventProvider attribute in the context of a J2EEDomain managed object is a special case. If its value is true, it indicates that the domain supports event notification and that the J2EEDomain managed object is a central registration point for event listeners that wish to receive event notification from all event providers in the domain. When the eventProvider attribute is true, the J2EEDomain managed object must emit all events from all event providers in the domain.

In addition, when the value of the eventProvider attribute of a J2EEDomain is true, the J2EEDomain managed object must emit the `j2ee.object.created` and `j2ee.object.deleted` event types (see “type” on page 45).

JSR77.3.2 J2EEDomain extends J2EEManagedObject

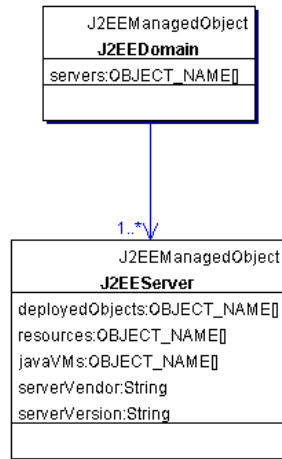


Figure JSR77.3.2 J2EEDomain

The J2EEDomain managed object type represents a management domain. There must be one managed object that implements the J2EEDomain model per management domain. All servers and applications associated with the same domain must be accessible from the J2EEDomain managed object.

JSR77.3.2.1 Attribute Detail

JSR77.3.2.1.1 servers

OBJECT_NAME[] servers

- supplier cardinality: 1..*

A list of all J2EE Servers in this domain. For each J2EE Server running in the domain, there must be one J2EEServer OBJECT_NAME in the `servers` list that identifies it.

JSR77.3.3 J2EEServer extends J2EEManagedObject

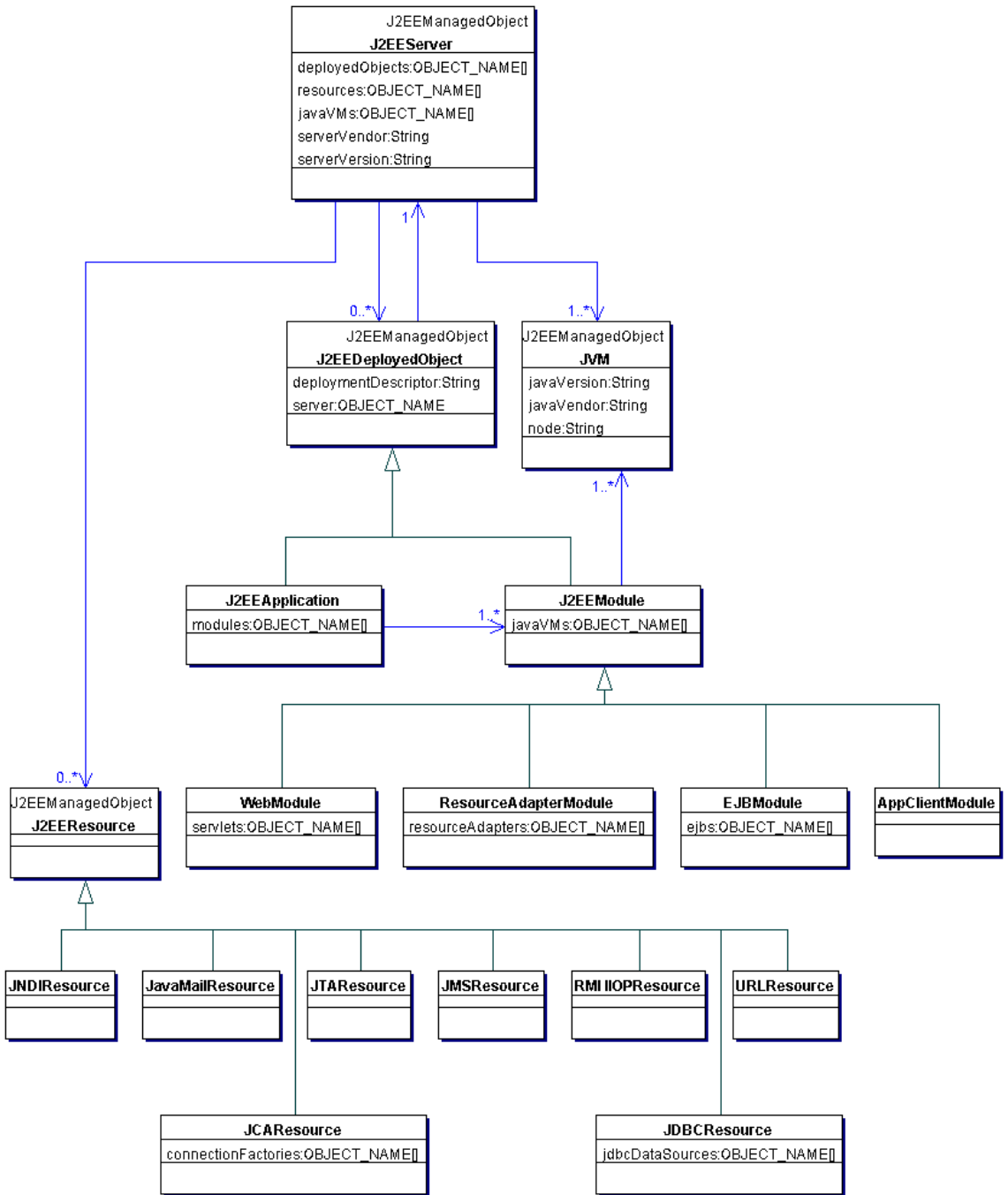


Figure JSR77.3.3 J2EEServer and associated managed objects

The J2EEServer model specifies the management information for a single J2EE server core implementation. The J2EE server core identifies the server core of one instance of a J2EE platform product as described in the Java 2 Enterprise Edition Platform specification section 2.1, Architecture.

JSR77.3.3.1 Attribute Detail

JSR77.3.3.1.1 deployedObjects

OBJECT_NAME[] deployedObjects

- supplier cardinality: 0..*

A list of all of the J2EEApplication and J2EEModule types deployed on this J2EEServer.

For each J2EE application (EAR) deployed on this server, there must be one J2EEApplication OBJECT_NAME in the `deployedObjects` list that identifies it.

For each Application client module deployed on this server, there must be one AppClientModule OBJECT_NAME in the `deployedObjects` list that identifies it.

For each Resource Adapter module (RAR) deployed on this server, there must be one ResourceAdapterModule OBJECT_NAME in the `deployedObjects` list that identifies it.

For each EJB module deployed on this server, there must be one EJBModule OBJECT_NAME in the `deployedObjects` list that identifies it.

For each Web module (WAR) deployed on this server, there must be one WebModule OBJECT_NAME in the `deployedObjects` list that identifies it.

JSR77.3.3.1.2 resources

OBJECT_NAME[] resources

- supplier cardinality: 0..*

A list of resources available to this server.

For each JCA resource associated with this server, there must be one JCAResource OBJECT_NAME in the `resources` list that identifies it.

For each JavaMailResource resource associated with this server, there must be one JavaMailResource OBJECT_NAME in the `resources` list that identifies it.

For each JDBCResource resource associated with this server, there must be one JDBCResource OBJECT_NAME in the resources list that identifies it.

For each JMS resource associated with this server, there must be one JMSResource OBJECT_NAME in the resources list that identifies it.

For each JNDI resource associated with this server, there must be one JNDIResource OBJECT_NAME in the resources list that identifies it.

For each JTA resource associated with this server, there must be one JTAResource OBJECT_NAME in the resources list that identifies it.

For each RMI_IOP resource associated with this server, there must be one RMI_IOPResource OBJECT_NAME in the resources list that identifies it.

For each URL resource associated with this server, there must be one URLResource OBJECT_NAME in the resources list that identifies it.

JSR77.3.3.1.3 javaVMs

OBJECT_NAME [] javaVMs

- supplier cardinality 1..*

A list of all Java virtual machines on which this J2EEServer has running threads. For each Java virtual machine this server utilizes, there must be one JVM OBJECT_NAME in the javaVMs list that identifies it.

JSR77.3.3.1.4 serverVendor

String serverVendor

Identifies the J2EE platform vendor of this J2EEServer. The value of serverVendor is specified by the server vendor.

JSR77.3.3.1.5 serverVersion

String serverVersion

Identifies the J2EE implementation version of this J2EEServer. The value of serverVersion is specified by the server vendor.

JSR77.3.4 JVM extends J2EEManagedObject

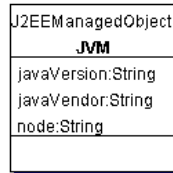


Figure JSR77.3.4 JVM type

Identifies a Java VM being utilized by a server. For each Java VM which is running threads associated with the J2EE server, it's containers or it's resources, there must be one managed object that implements the JVM model. A JVM managed object must be removed when the Java VM it manages is no longer running.

JSR77.3.4.1 Attribute Detail

JSR77.3.4.1.1 *javaVersion*

String javaVersion

Identifies the Java Runtime Environment version of this Java VM. The value of `javaVersion` must be identical to the value of the system property `java.version`.

JSR77.3.4.1.2 *javaVendor*

String javaVendor

Identifies the Java Runtime Environment vendor of this Java VM. The value of `javaVendor` must be identical to the value of the system property `java.vendor`.

JSR77.3.4.1.3 *node*

String node

Identifies the node (machine) this JVM is running on. The value of the `node` attribute must be the fully qualified hostname of the node the JVM is running on.

JSR77.3.5 J2EEDeployedObject extends J2EEManagedObject

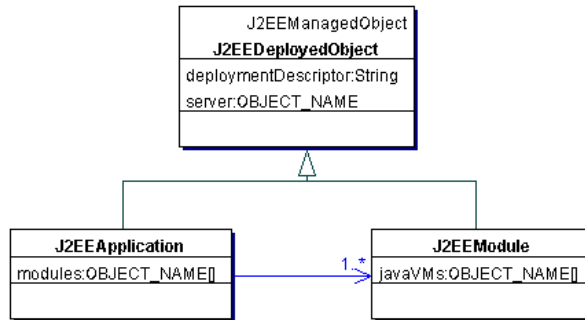


Figure JSR77.3.5 J2EEDeployedObject types

The base model for J2EEApplication and J2EEModule. All J2EEDeployedObject managed objects contain the original XML deployment descriptor that was created for the application or module during the deployment process.

JSR77.3.5.0.1 deploymentDescriptor

String deploymentDescriptor

The deploymentDescriptor string must contain the original XML deployment descriptor that was created for this module during the deployment process. The deploymentDescriptor attribute must provide a full deployment descriptor based on any partial deployment descriptor plus deployment annotations.

JSR77.3.5.0.2 server

OBJECT_NAME server

- supplier cardinality: 1

The J2EE server the application or module is deployed on.

JSR77.3.6 J2EEApplication extends J2EEDeployedObject

Identifies a J2EE application EAR that has been deployed.

JSR77.3.6.1 Attribute Detail

JSR77.3.6.1.1 *modules*

OBJECT_NAME[] *modules*

- supplier cardinality: 1..*

A list of J2EEModules that comprise this application. For each J2EE module that is utilized by this application, there must be one J2EEModule OBJECT_NAME in the `modules` list that identifies it.

JSR77.3.7 J2EEModule extends J2EEDeployedObject

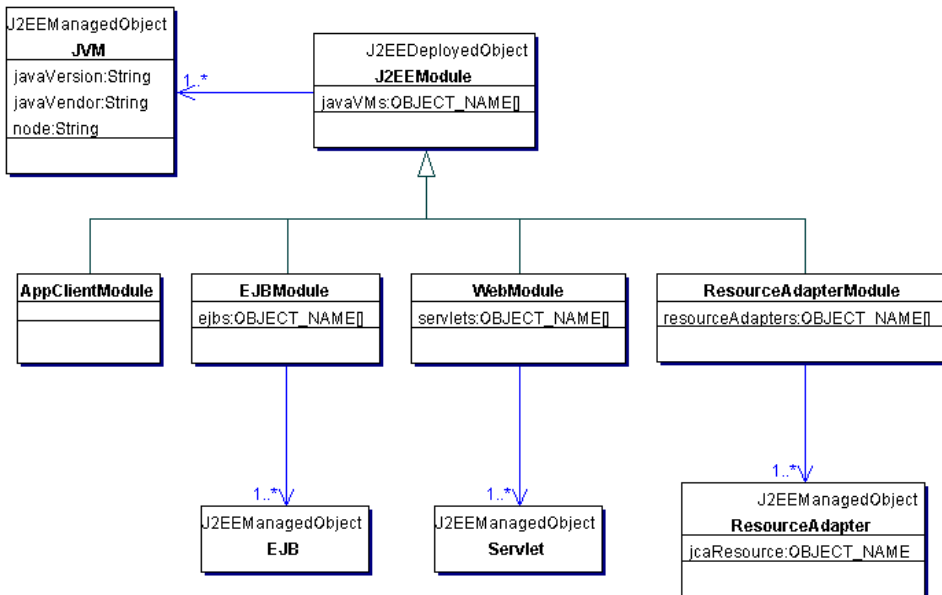


Figure JSR77.3.6 J2EEModule types

The J2EEModule model is the base model for all of the J2EE Module types. Managed objects that implement the J2EEModule model represent EAR, JAR, WAR and RAR files that have been deployed.

JSR77.3.7.1 Attribute Detail

JSR77.3.7.1.1 javaVMs

OBJECT_NAME[] javaVMs

- supplier cardinality: 1..*

Identifies the Java virtual machines on which this module is running. For each JVM on which this module has running threads there must be one JVM OBJECT_NAME in the `javaVMs` list that identifies it.

Each OBJECT_NAME in the J2EEModule `javaVMs` list must match one of the Java VM names in the `javaVMs` attribute of the J2EEServer on which this module is deployed.

JSR77.3.8 AppClientModule extends J2EEModule

Identifies a deployed Application Client module.

JSR77.3.9 EJBModule extends J2EEModule

Identifies a deployed EJB JAR module.

JSR77.3.9.1 Attribute Detail

JSR77.3.9.1.1 ejbs

OBJECT_NAME [] ejbs

- supplier cardinality: 1..*

A list of EJB components contained in the deployed EJB JAR module. For each EJB component contained in the deployed EJB JAR there must be one EJB OBJECT_NAME in the `ejbs` list that identifies it.

JSR77.3.10 EJB extends J2EEManagedObject

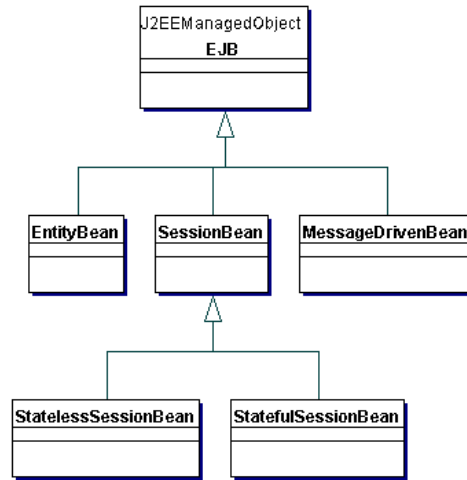


Figure JSR77.3.7 EJB types

The EJB model is the base model for the specific types of Enterprise JavaBeans components.

JSR77.3.11 MessageDrivenBean extends EJB

Identifies an individual message-driven bean. For each deployed message-driven bean, there must be one managed object that implements the MessageDrivenBean model.

JSR77.3.12 EntityBean extends EJB

Identifies a deployed entity bean. For each deployed entity bean, there must be one managed object that implements the EntityBean model.

JSR77.3.13 SessionBean extends EJB

Identifies a deployed session bean. The SessionBean model is always extended by one of the sub-models StatefulSessionBean or StatelessSessionBean.

JSR77.3.14 StatefulSessionBean extends SessionBean

Identifies a deployed stateful session bean.

JSR77.3.15 StatelessSessionBean extends SessionBean

Identifies a deployed stateless session bean.

JSR77.3.16 WebModule extends J2EEModule

The WebModule model identifies a deployed WAR module.

JSR77.3.16.1 Attribute Detail

JSR77.3.16.1.1 servlets

OBJECT_NAME [] servlets

- supplier cardinality: 1..*

A list of servlets contained in the deployed WAR module. For each servlet contained in the deployed WAR module there must be one Servlet OBJECT_NAME in the `servlets` list that identifies it.

JSR77.3.17 Servlet extends J2EEManagedObject

Identifies a deployed servlet component.

JSR77.3.18 ResourceAdapterModule extends J2EEModule

The ResourceAdapterModule model identifies a deployed resource adapter archive (RAR).

JSR77.3.18.1 Attribute Detail

JSR77.3.18.1.1 resourceAdapters

OBJECT_NAME [] resourceAdapters

- supplier cardinality: 1..*

A list of resource adapters contained in this resource adapter module. For each resource adapter contained in the deployed RAR module, there must be one ResourceAdapter OBJECT_NAME in the `resourceAdapters` list that identifies it.

JSR77.3.19 ResourceAdapter extends J2EEManagedObject

Identifies a deployed resource adapter.

JSR77.3.19.1 Attribute Detail

JSR77.3.19.1.1 jcaResource

OBJECT_NAME jcaResource

- supplier cardinality: 1

The value of `jcaResource` must be a JCAResource OBJECT_NAME that identifies the JCA connector resource implemented by this ResourceAdapter.

JSR77.3.20 J2EEResource extends J2EEManagedObject

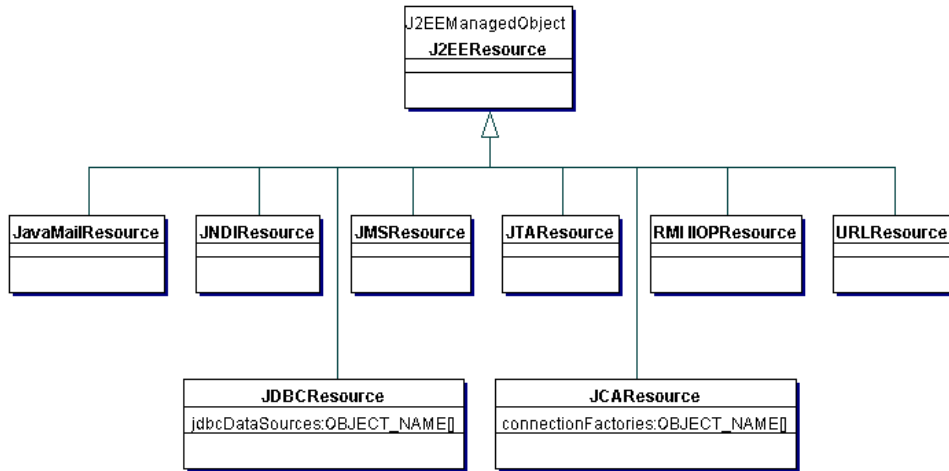


Figure JSR77.3.8 J2EEResource

J2EEResource is the base model for all managed object types that represent J2EE resources. J2EE resources are resources utilized by the J2EE core server to provide the J2EE standard services required by the J2EE platform architecture. For each J2EE standard service that a server provides, there must be one managed object that implements the J2EEResource model of the appropriate type.

JSR77.3.21 JavaMailResource extends J2EEResource

Identifies a JavaMail™ resource. For each JavaMail™ resource available to a server there must be one managed object that implements the JavaMail™ model.

JSR77.3.22 JCAResource extends J2EEResource

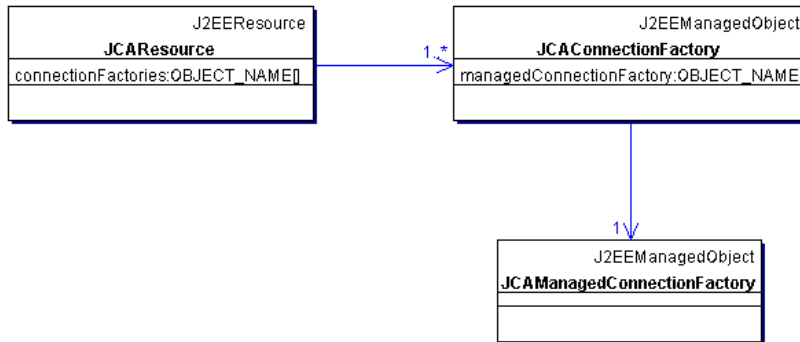


Figure JSR77.3.9 JCAResource

Identifies a JCA resource. A JCAResource object manages one or more connection factories. For each JCA resource provided on a server, there must be one JCAResource OBJECT_NAME in the servers resources list that identifies it.

JSR77.3.22.1 Attribute Detail

JSR77.3.22.1.1 connectionFactories

OBJECT_NAME[] connectionFactories

- supplier cardinality: 1..*

A list of the connection factories available on the corresponding JCAResource object. For each connection factory available to this JCAResource there must be one JCAConnectionFactory OBJECT_NAME in the `connectionFactories` list that identifies it.

JSR77.3.23 JCAConnectionFactory extends J2EEManagedObject

Identifies a connection factory. For each connection factory available to a server, there must be one managed object that implements the JCAConnectionFactory model.

JSR77.3.23.1 Attribute Detail

JSR77.3.23.1.1 managedConnectionFactory

OBJECT_NAME managedConnectionFactory

- supplier cardinality: 1

The value of managedConnectionFactory must be a JCAManagedConnectionFactory OBJECT_NAME that identifies the managed connection factory associated with the corresponding connection factory.

JSR77.3.24 JCAManagedConnectionFactory extends J2EEManagedObject

Identifies a JCA managed connection factory. For each JCA managed connection factory available to a JCAResource, there must be one managed object that implements the JCAManagedConnectionFactory model.

JSR77.3.25 JDBCResource extends J2EEResource

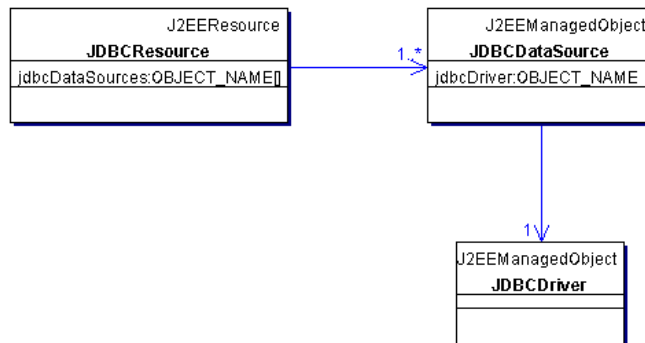


Figure JSR77.3.10 JDBCResource

The JDBCResource model identifies a JDBC resource. A JDBC resource manages one or more JDBC data sources. For each JDBC resource provided on a server, there must be one JDBCResource OBJECT_NAME in the servers resources list that identifies it.

JSR77.3.25.1 Attribute Detail

JSR77.3.25.1.1 jdbcDataSource

OBJECT_NAME[] jdbcDataSouces

- supplier cardinality: 1..*

Identifies the JDBC data sources available on the corresponding JDBC resource. For each JDBC data source available on this JDBC resource there must be one JDBCDataSource OBJECT_NAME in the jdbcDataSources list.

JSR77.3.26 JDBCDataSource extends J2EEManagedObject

Identifies a physical JDBC data source. For each JDBC data source available to a server there must be one managed object that implements the JDBCDataSource model.

JSR77.3.26.1 Attribute Detail

JSR77.3.26.1.1 jdbcDriver

OBJECT_NAME jdbcDriver

- supplier cardinality: 1

The value of jdbcDriver must be an JDBCDriver OBJECT_NAME that identifies the JDBC driver for the corresponding JDBC data source.

JSR77.3.27 JDBCDriver extends J2EEManagedObject

Identifies an individual JDBC driver. For each JDBC driver available to a JDBC resource there must be one managed object that implements the JDBCDriver model.

JSR77.3.28 JMSResource extends J2EEResource

The JMSResource model identifies a Java Message Service resource. For each JMS resource provided on a server, there must be one managed object that implements the JMSResource model.

JSR77.3.29 JNDIResource extends J2EEResource

Identifies a Java Naming and Directory Interface™ (JNDI) resource. For each JNDI resource available to a server there must be one managed object that implements the JNDIResource model.

JSR77.3.30 JTAResource extends J2EEResource

Identifies a Java Transaction API (JTA) resource. For each JTA resource available to a server there must be one managed object that implements the JTAResource model.

JSR77.3.31 RMI_IIOPResource extends J2EEResource

Identifies an RMI_IIOP resource. For each RMI_IIOP resource available to a server there must be one managed object that implements the RMI_IIOPResource model.

JSR77.3.32 URLResource extends J2EEResource

Identifies a managed URL resource. For each managed URL resource provided by a server there should be one managed object that implements the URLResource model. It is specific to a server implementation which URL resources are exposed as manageable and there are no specific requirements as to which URL resources provided by a server are exposed as managed objects.



CHAPTER JSR77.4

Events

This chapter specifies a standard extensible event model for the J2EE Management Model. Events notify management applications that need to react to state changes and specific conditions that occur in the platform containers and services.

JSR77.4.1 EventProvider

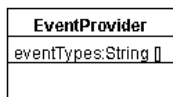


Figure JSR77.4.1 EventProvider

The EventProvider model specifies the `eventTypes` attribute, which must be implemented by all managed objects that emit events.

JSR77.4.1.1 Attribute Detail

JSR77.4.1.1.1 eventTypes

String [] eventTypes

A list of the types of events the managed object emits. The contents of the list are type strings, as specified in “type” on page 45.

JSR77.4.2 Notification

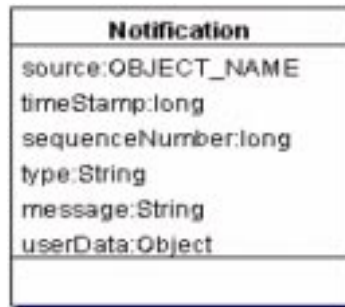


Figure JSR77.4.2 Notification

The Notification model specifies the attributes that must be included in every event generated by the management system. A Notification includes:

- The name of the source managed object
- A timestamp
- a sequence number
- a string that identifies its type
- a message that provides additional information about the event
- optional vendor specific user data.

All management systems that support event notification must support all of the attributes of the Notification model. All events that are generated on behalf of managed objects that are part of or extend the J2EE Management Model must provide valid data for each of the attributes specified by the Notification model.

JSR77.4.2.1 Attribute Detail

JSR77.4.2.1.1 source

OBJECT_NAME source

The name of the managed object that generated this event. The value of the source attribute must be equal to the OBJECT_NAME of the managed object from which the event originated.

Note – Components which deliver events as Java object instances, such as in the case of the MEJB component's ListenerRegistration interface, must emit event objects which are instances of the class `javax.management.Notification` as specified by the Java Management Extensions Instrumentation and Agent Specification (JSR003). The return type of the source attribute accessor of `javax.management.Notification` is a `java.lang.Object` which is actually the `javax.management.ObjectName` instance that encapsulates the OBJECT_NAME of the source managed object.

JSR77.4.2.1.2 timeStamp

long timeStamp

The time of the event represented as a long, whose value is the number of milliseconds since January 1, 1970, 00:00:00.

JSR77.4.2.1.3 sequenceNumber

long sequenceNumber

The sequence number of the event. Identifies the position of the event in a stream of events. The sequence number provides a means of determining the order of sequential events that occurred with the same timestamp within the minimum supported unit of time. The scope of the sequence number is the source of the event.

JSR77.4.2.1.4 type

String type

The type of an event, not to be confused with its Java class, is the characterization of a generic event object. The type is assigned by the source object and conveys the semantic meaning of a particular event. The type is given as a string, which is interpreted as any number of dot-separated components. This allows some structure in the naming of event types.

All strings prefixed by “j2ee.” are reserved for the standard events defined by this specification. Otherwise, event sources are free to define all types they wish to use when naming the events they emit. All of the event types emitted by an event provider must be listed in the eventTypes attribute.

Standard event types are:

- `j2ee.object.created`- a new managed object was created.
- `j2ee.object.deleted` - a managed object was deleted.
- `j2ee.state.starting` - a state manageable object entered the starting state.
- `j2ee.state.running` - a state manageable object entered the running state.
- `j2ee.state.stopping` - a state manageable object entered the stopping state.
- `j2ee.state.stopped` - a state manageable object entered the stopped state.
- `j2ee.state.failed` - a state manageable object entered the failed state.
- `j2ee.attribute.changed` - an attribute has change value

JSR77.4.2.1.5 message

String message

An informational message about the event.

JSR77.4.2.1.6 userData

Object userData

Optional data the notification broadcaster wishes to communicate to listeners. The content of the data is vendor-specific. The userData attribute may be null.

JSR77.4.3 Event Notification

Most management systems specify an event notification mechanism. Implementations of the J2EE Management Model by management systems that support event notification must deliver events that implement the Notification model.

Management tools that interoperate with these management systems and their protocols receive events through the delivery mechanisms specified by the management system being incorporated.

JSR77.4.3.1 `javax.management.Notification`

All implementations of this specification, which deliver a Java object as an instance of an event notification, must emit event objects which are instances of the class `javax.management.Notification` as specified by the Java Management Extensions Instrumentation and Agent Specification (JSR003). The the value of the attributes in any instance of `javax.management.Notification` emitted by a managed object type specified by the J2EE Management Model must be exactly as specified in this chapter.



CHAPTER JSR77.5

State Management

This chapter specifies a standard state management model for the J2EE Management Model. State management refers to the management facilities that are provided by compliant J2EE platforms to manage the state of a J2EE platform and the components that comprise it. The management facilities allow Management Applications to get the current state of the platform and its components, find out how long the platform and components have been running, and start and stop the platform components.

JSR77.5.1 StateManageable

StateManageable
state:int startTime:long
start:void startRecursive:void stop:void

Figure JSR77.5.1 StateManageable

The StateManageable model specifies the operations and attributes that must be implemented by a managed object that supports state management. A managed object that implements the StateManageable model is termed a State Manageable Object (SMO). An SMO generates events when its state changes.

JSR77.5.1.1 Attribute Detail

JSR77.5.1.1.1 *state*

int state

The current state of this SMO.

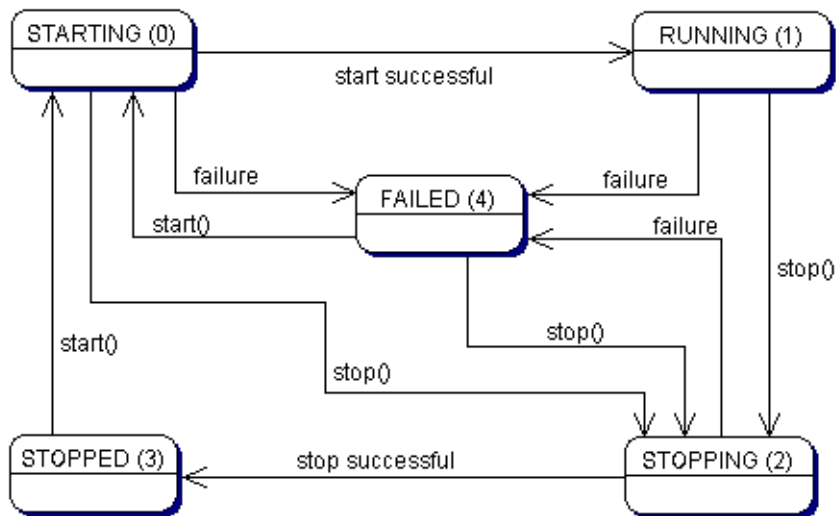


Figure JSR77.5.2 StateManageable object state diagram

The SMO can be in one of the following states (see the Event Model for details of the events):

STARTING (0)

This state indicates that the SMO has been requested to start, and is in the process of starting. On entering this state an SMO may generate an event whose type value is “`j2ee.state.starting`”. Event notification of the STARTING state is optional for all managed objects that implement StateManageable.

RUNNING (1)

This is the normal running state for an SMO. This state indicates that the SMO is operational. On entering this state an SMO must generate an event whose type value is “`j2ee.state.running`”. Event notification of the RUNNING state is required for all managed objects that implement StateManageable.

STOPPING (2)

This state indicates that the SMO has been requested to stop, and is in the process of stopping. On entering this state an SMO may generate an event whose type value is “`j2ee.state.stopping`”. Event notification of the STOPPING state is optional for all managed objects that implement StateManageable.

STOPPED (3)

This state indicates that the StateManageable Object has stopped and can be restarted. On entering this state an SMO must generate an event whose type value is “`j2ee.state.stopped`”. Event notification of the STOPPED state is required by all managed objects that implement StateManageable.

FAILED (4)

This state indicates that the StateManageable Object has unexpectedly stopped. On entering this state an SMO must generate an event whose type value is “`j2ee.state.failed`”. Event notification of the FAILED state is required by all managed objects that implement StateManageable. It may be possible to start an SMO that has entered the FAILED state by calling the `start()` operation. It may be possible to stop an SMO that has entered the FAILED state by calling the `stop()` operation. An SMO that cannot successfully reach the RUNNING or STOPPED state from the FAILED state must reenter the FAILED state after the attempt to reach the RUNNING or STOPPED state fails and emit another “`j2ee.state.failed`” event. An SMO that cannot successfully reach the RUNNING or STOPPED state from the FAILED state requires intervention, such as redeployment, to restore the managed object to the desired state.

JSR77.5.1.1.2 startTime

long startTime

The time that the managed object was started (i.e. entered the RUNNING state) represented as a long, which value is the number of milliseconds since January 1, 1970, 00:00:00.

JSR77.5.1.2 Operation Detail

JSR77.5.1.2.1 start

void start()

Starts the SMO. This operation can be invoked only when the SMO is in the STOPPED or FAILED state. It causes the SMO to go into the STARTING state initially, and if it completes successfully, the SMO will be in the RUNNING state. Note that `start()` is not called on any of the child SMOs that are registered with this SMO; it is the responsibility of the calling application to start the child SMO if this is required.

JSR77.5.1.2.2 startRecursive

void startRecursive()

Starts the SMO. This operation can only be invoked when the SMO is in the STOPPED or FAILED state. It causes the SMO to go into the STARTING state initially, and if it completes successfully, the SMO will be in the RUNNING state. The operation `startRecursive()` is called on all the child SMOs registered with this SMO that are in the STOPPED or FAILED state.

JSR77.5.1.2.3 stop

void stop()

Stops the SMO. This operation can only be invoked when the SMO is in the RUNNING or STARTING state. It causes `stop()` to be called on all the child SMOs registered with this SMO that are in the RUNNING or STARTING state. It is mandatory if an SMO is in the STOPPED or FAILED state, that all its child SMOs must also be in the STOPPED or FAILED state, therefore there is no `stopRecursive()` operation. Invoking `stop()` causes the SMO to go into the STOPPING state initially, and if it completes successfully, the SMO and all the child SMOs will be in the STOPPED state.

CHAPTER JSR77.6

Performance Monitoring

This chapter describes the Performance Data Framework, which specifies a performance data model as well as performance data requirements of the J2EE Management Model.

JSR77.6.1 Performance Data Framework

The Performance Data Framework consists of the `StatisticsProvider` model, which any managed object may implement, the `Stats` interfaces, which specify standard performance attribute semantics for each managed object type, and the `Statistic` interfaces which provide specific interfaces for representing the common performance data types.

JSR77.6.2 StatisticsProvider

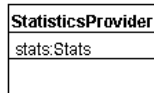


Figure JSR77.6.1 StatisticsProvider

The `StatisticsProvider` model must be implemented by all managed objects that provide performance data. A managed object that implements the `StatisticsProvider` model must have its `statisticProvider` attribute set to "true".

The `stats` attribute references the specific Stats interface that corresponds to the managed object type that is providing Statistics. For example, an EntityBean managed object that implements StatisticsProvider will have a reference in the `stats` attribute to an object that implements the EntityBeanStats interface. The detail for the `stats` attribute includes a table of the appropriate Stats interface that each managed object must implement if it provides performance data.

JSR77.6.2.1 Attribute Detail

JSR77.6.2.1.1 *stats*

Stats stats

Provides access to the implementation of the specific Stats interface that this managed object is required to support if it implements the StatisticProvider model. Table JSR77.6-1 shows the Stats interface required by each managed object type. See “Stats” on page 61 for details about support for specific statistics.

Table JSR77.6-1 Stats interface requirements

J2EEManagedObject	Required Stats interface
EJB	EJBStats
EntityBean	EntityBeanStats
JavaMail	JavaMailStats
JCAResource	JCAStats
JDBCResource	JDBCStats
JMSResource	JMSStats
JTAResource	JTAStats
JVM	JVMStats
MessageDrivenBean	MessageDrivenBeanStats
Servlet	ServletStats
SessionBean	SessionBeanStats
StatefulSessionBean	StatefulSessionBeanStats
StatelessSessionBean	StatelessSessionBeanStats

Table JSR77.6-1 Stats interface requirements

J2EEManagedObject	Required Stats interface
URLResource	URLStats

JSR77.6.3 package javax.management.j2ee.statistics

All of the following interfaces are part of the javax.management.j2ee.statistics package which provides the standard interfaces for Java implementations of the Statistic and Stats interfaces and subinterfaces. All implementations that provide statistics as Java object instances must implement the appropriate interfaces from the javax.management.j2ee.statistics package.

JSR77.6.4 Statistic

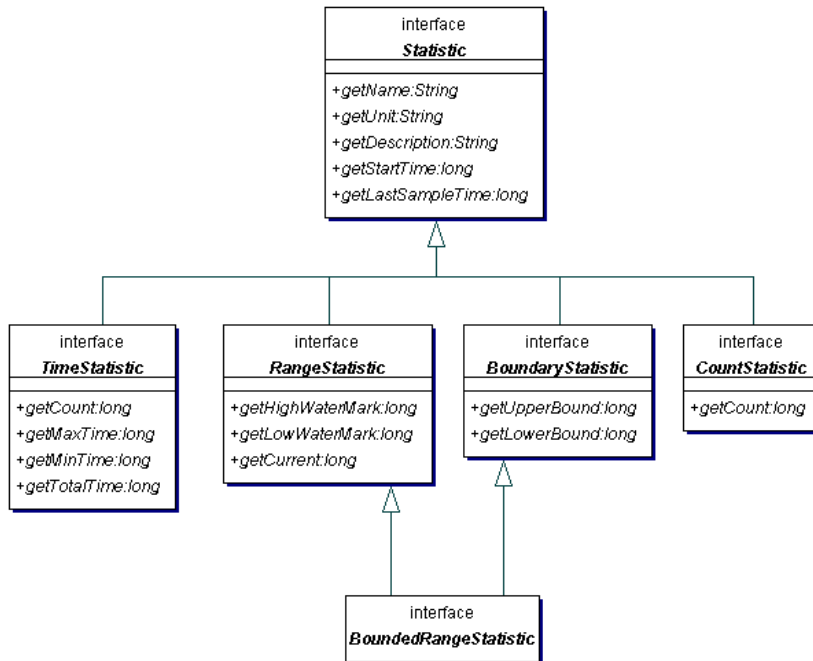


Figure JSR77.6.2 Statistic

The `Statistic` interface and its subinterfaces specify the required accessors which provide the performance data described by the specific attributes in the `Stats` interfaces. The `Statistic` subinterfaces specify accessors which provide statistical data about count, time, and both bounded and unbounded ranges.

JSR77.6.4.1 Operation Detail

JSR77.6.4.1.1 getName

String `getName()`

Returns the name of this `Statistic`. The name must always correspond to the name of the `Stats` accessor that is providing the data for this statistic.

JSR77.6.4.1.2 getUnit

String `getUnit()`

Returns the unit of measurement for this `Statistic`.

Valid values for `TimeStatistic` measurements are “`HOURL`”, “`MINUTE`”, “`SECOND`”, “`MILLISECOND`”, “`MICROSECOND`” and “`NANOSECOND`”.

JSR77.6.4.1.3 getDescription

String `getDescription()`

Returns a human-readable description of the `Statistic`.

JSR77.6.4.1.4 getStartTime

long `getStartTime()`

Returns the time the first measurement was taken represented as a `long`, whose value is the number of milliseconds since January 1, 1970, 00:00:00.

JSR77.6.4.1.5 getLastSampleTime

long `getLastSampleTime()`

Returns the time the most recent measurement was taken represented as a `long`, whose value is the number of milliseconds since January 1, 1970, 00:00:00.

JSR77.6.5 **CountStatistic** extends Statistic

The CountStatistic interface specifies standard count measurements.

JSR77.6.5.1 **Operation Detail**

JSR77.6.5.1.1 *getCount*

long getCount()

Returns the count since the measurement started.

JSR77.6.6 **TimeStatistic** extends Statistic

Specifies standard timing measurements for a given operation.

JSR77.6.6.1 **Operation Detail**

JSR77.6.6.1.1 *getCount*

long getCount()

Returns the number of times the operation was invoked since the beginning of this measurement.

JSR77.6.6.1.2 *getMaxTime*

long getMaxTime()

Returns the maximum amount of time taken to complete one invocation of this operation since the beginning of this measurement.

JSR77.6.6.1.3 *getMinTime*

long getMinTime()

Returns the minimum amount of time taken to complete one invocation of this operation since the beginning of this measurement.

JSR77.6.6.1.4 *getTotalTime***long getTotalTime()**

Returns the sum total of time taken to complete every invocation of this operation since the beginning of this measurement. Dividing `totalTime` by `count` will give you the average execution time for this operation.

JSR77.6.7 *RangeStatistic* extends *Statistic*

Specifies standard measurements of the lowest and highest values an attribute has held as well as its current value.

JSR77.6.7.1 *Operation Detail***JSR77.6.7.1.1 *getHighWaterMark*****long getHighWaterMark()**

Returns the highest value this attribute has held since the beginning of the measurement.

JSR77.6.7.1.2 *getLowWaterMark***long getLowWaterMark()**

Returns the lowest value this attribute has held since the beginning of the measurement.

JSR77.6.7.1.3 *getCurrent()***long getCurrent()**

Returns the current value of this attribute.

JSR77.6.8 BoundaryStatistic extends Statistic

The BoundaryStatistic interface specifies standard measurements of the upper and lower limits of the value of an attribute.

JSR77.6.8.1 Operation Detail

JSR77.6.8.1.1 getUpperBound

long getUpperBound()

Returns the upper limit of the value of this attribute.

JSR77.6.8.1.2 getLowerBound

long getLowerBound()

Returns the lower limit of the value of this attribute.

JSR77.6.9 BoundedRangeStatistic extends RangeStatistic, BoundaryStatistic

The BoundedRangeStatistic interface extends the RangeStatistic and BoundaryStatistic interfaces and provides standard measurements of a range that has fixed limits.

JSR77.6.9.1 Operation Detail

JSR77.6.9.1.1 getHighWaterMark

long getHighWaterMark()

Returns the highest value this attribute has held since the beginning of the measurement. The value of `getHighWaterMark()` must be less than or equal to the return value of `getUpperBound()`.

JSR77.6.9.1.2 getLowWaterMark

long getLowWaterMark()

Returns the lowest value this attribute has held since the beginning of the measurement. The return value of `getLowWaterMark()` must be greater than or equal to the return value of `getLowerBound()`.

JSR77.6.9.1.3 getCurrent

long getCurrent()

Returns the current value of this attribute.

JSR77.6.9.1.4 getUpperBound

long getUpperBound()

Returns the upper limit of the value of this attribute.

JSR77.6.9.1.5 getLowerBound

long getLowerBound()

Returns the lower limit of the value of this attribute.

JSR77.6.10 Stats

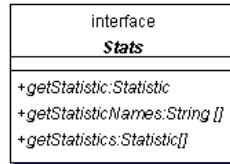


Figure JSR77.6.3 Stats hierarchy

The Stats interface and its subinterfaces specify performance data accessors for each of the specific managed object types. The data required by the interfaces is commonly available on all platform implementations. Managed objects that support statistics are permitted to provide support for a subset of the accessors in the Stats interfaces. Managed objects indicate which of the specified accessors return valid data by including only the names of the supported statistics in the statisticNames list. The data provided by a supported statistic must be exactly as specified by the corresponding Stats interface. The Stats interfaces may be extended to provide vendor specific performance statistics. Vendor specific performance statistics must implement or extend one of the standard Statistics interfaces.

JSR77.6.10.1 Operation Detail

JSR77.6.10.1.1 getStatisticNames

String [] getStatisticNames()

Returns a list of the names of the attributes from the specific Stats interface that this object supports. Attributes named in the list must correspond to the names of operations in the Stats interface that will return Statistics object of the appropriate type which contains valid data. Each operation in a Stats interface is an accessor which follows the pattern *getAttributeName*. The *AttributeName* portion of the operation name is the value that is returned as the name in the StatisticNames list.

The value of attributes whose names are not included in the StatisticNames list is undefined and must be considered invalid. For each attribute name in the StatisticNames list that returns a Statistic there must be one Statistic object with the same name in the statistics list.

JSR77.6.10.1.2 getStatistics

Statistic [] getStatistics()

Returns a list of all the Statistic objects supported by this Stats object..

JSR77.6.10.1.3 getStatistic

Statistic getStatistic(String statisticName)

Gets a Statistic by name.

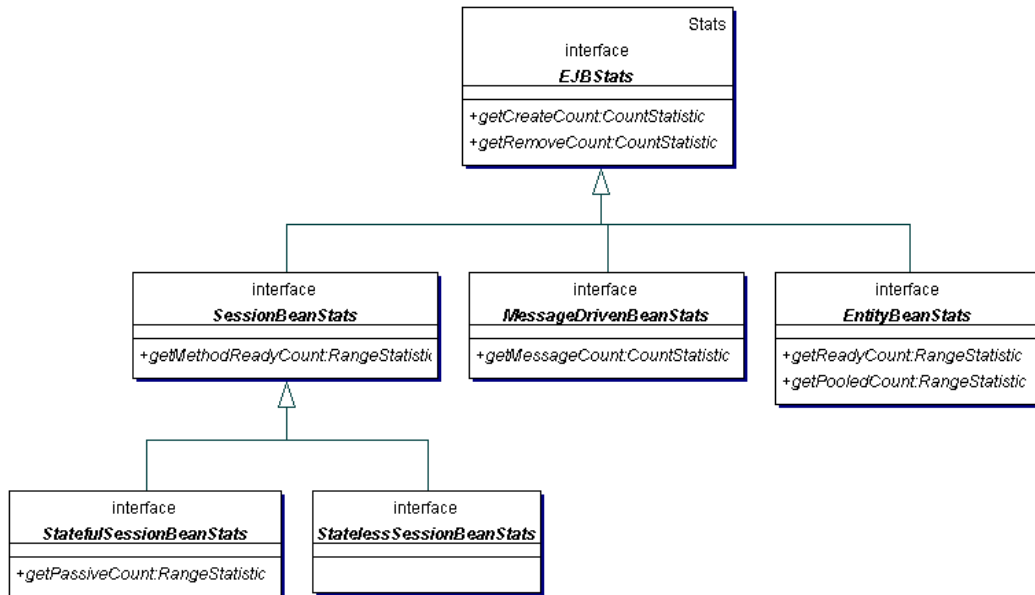
JSR77.6.11 EJBStats extends Stats

Figure JSR77.6.4 EJBStats

The EJBStats interface specifies statistics provided by all EJB component types.

JSR77.6.11.1 Operation Detail**JSR77.6.11.1.1 getCreateCount**

CountStatistic getCreateCount()

Returns the number of times the beans create method was called.

JSR77.6.11.1.2 getRemoveCount

CountStatistic getRemoveCount()

Returns the number of times the beans remove method was called.

JSR77.6.12 EntityBeanStats extends EJBStats

Specifies statistics provided by entity beans.

JSR77.6.12.1 Operation Detail

JSR77.6.12.1.1 getReadyCount

RangeStatistic getReadyCount()

Returns the number of bean instances in the ready state.

JSR77.6.12.1.2 getPooledCount

RangeStatistic getPooledCount()

Returns the number of bean instances in the pooled state.

JSR77.6.13 MessageDrivenBeanStats extends EJBStats

Specifies the statistics provided by a message driven bean.

JSR77.6.13.1 Operation Detail

JSR77.6.13.1.1 getMessageCount

CountStatistic getMessageCount()

Returns the number of messages received.

JSR77.6.14 SessionBeanStats extends EJBStats

Specifies the statistics provided by session beans of both stateful and stateless types.

JSR77.6.14.1 Operation Detail***JSR77.6.14.1.1 getMethodReadyCount***

RangeStatistic `getMethodReadyCount()`

Returns the number of beans in the method-ready state.

JSR77.6.15 StatefulSessionBeanStats extends SessionBeanStats

Specifies the statistics provided by a stateful session bean.

JSR77.6.15.1 Operation Detail***JSR77.6.15.1.1 getPassiveCount***

RangeStatistic `getPassiveCount()`

Returns the number of beans that are in the passivated state.

JSR77.6.16 StatelessSessionBeanStats extends SessionBeanStats

Specifies the statistics provided by a stateless session bean.

JSR77.6.17 JavaMailStats extends Stats

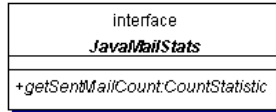


Figure JSR77.6.5 JavaMailStats

Specifies the statistics provided by a JavaMail resource.

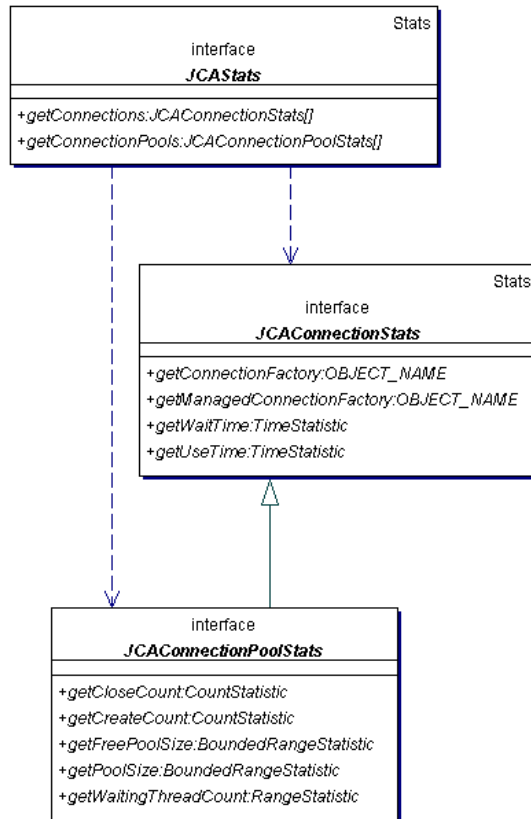
JSR77.6.17.1 Operation Detail

JSR77.6.17.1.1 getSentMailCount

CountStatistic `getSentMailCount()`

Returns the number of mail messages sent.

JSR77.6.18 JCAStats extends Stats



The JCAStats interface specifies the statistics provided by a JCA resource.

JSR77.6.18.1 Operation Detail

JSR77.6.18.1.1 *getConnections*

JCAConnectionStats[] `getConnections()`

Returns a list of JCAConnectionStats that provide statistics about the non-pooled connections associated with the referencing JCA resource statistics.

JSR77.6.18.1.2 getConnectionPools

JCAConnectionPoolStats[] getConnectionPools()

Returns a list of JCAConnectionPoolStats that provide statistics about the connection pools associated with the referencing JCA resource statistics.

JSR77.6.19 JCAConnectionStats extends Stats

The JCAConnectionStats interface specifies the statistics provided by a JCA connection.

JSR77.6.19.1 Operation Detail

JSR77.6.19.1.1 getConnectionFactory

OBJECT_NAME getConnectionFactory()

Returns the JCAConnectionFactory OBJECT_NAME of the managed object that identifies the connector's connection factory for this connection.

JSR77.6.19.1.2 getManagedConnectionFactory

OBJECT_NAME getManagedConnectionFactory()

Returns the JCAManagedConnectionFactory OBJECT_NAME of the managed object that identifies the connector's managed connection factory for this connection.

JSR77.6.19.1.3 getWaitTime

TimeStatistic getWaitTime()

Returns time spent waiting for a connection to be available.

JSR77.6.19.1.4 getUseTime

TimeStatistic getUseTime()

Returns the time spent using a connection.

JSR77.6.20 JCAConnectionPoolStats extends JCAConnectionStats

The JCAConnectionPoolStats interface specifies the statistics provided by a JCA connection pool.

JSR77.6.20.1 Operation Detail

JSR77.6.20.1.1 getCloseCount

CountStatistic getCloseCount()

Returns the number of connections closed.

JSR77.6.20.1.2 getCreateCount

CountStatistic getCreateCount()

Returns the number of connections created.

JSR77.6.20.1.3 getFreePoolSize

BoundedRangeStatistic getFreePoolSize()

Returns the number of free connections in the pool.

JSR77.6.20.1.4 getPoolSize

BoundedRangeStatistic getPoolSize()

Returns the size of the connection pool.

JSR77.6.20.1.5 getWaitingThreadCount

BoundedRangeStatistic getWaitingThreadCount()

Returns the number of threads waiting for a connection.

JSR77.6.21 JDBCStats extends Stats

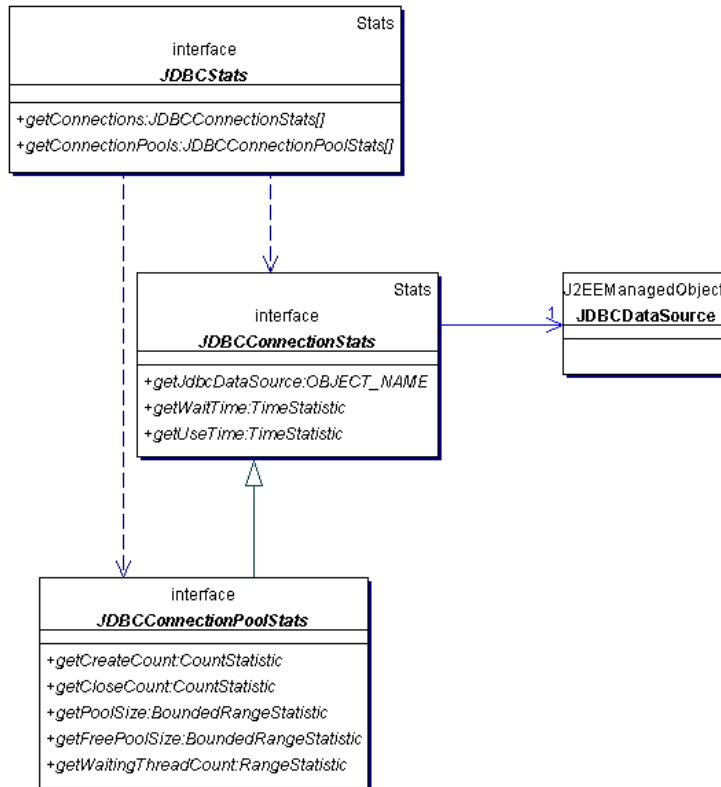


Figure JSR77.6.6 JDBCStats, JDBCConnectionStats, JDBCConnectionPoolStats

The JDBCStats type specifies the statistics provided by a JDBC resource.

JSR77.6.21.1 Operation Detail

JSR77.6.21.1.1 getConnections

JDBCConnectionStats[] getConnections

Returns a list of JDBCConnectionStats that provide statistics about the non-pooled connections associated with the referencing JDBC resource statistics.

JSR77.6.21.1.2 getConnectionPools

JDBCConnectionPoolStats[] getConnectionPools()

Returns a list of `JDBCConnectionPoolStats` that provide statistics about the connection pools associated with the referencing JDBC resource statistics.

JSR77.6.22 JDBCConnectionStats extends Stats

Specifies the statistics provided by a JDBC connection.

JSR77.6.22.1 Operation Detail

JSR77.6.22.1.1 getJdbcDataSource

OBJECT_NAME `getJdbcDataSource()`

Returns the name of the managed object that identifies the JDBC data source for this connection.

JSR77.6.22.1.2 getWaitTime

TimeStatistic `getWaitTime()`

Returns the time spent waiting for a connection to be available.

JSR77.6.22.1.3 getUseTime

TimeStatistic `getUseTime()`

Returns the time spent using a connection.

JSR77.6.23 JDBCConnectionPoolStats extends JDBCConnectionStats

Specifies the statistics provided by a JDBC connection pool.

JSR77.6.23.1 Operation Detail

JSR77.6.23.1.1 getCloseCount

CountStatistic getCloseCount()

Returns the number of connections closed.

JSR77.6.23.1.2 getCreateCount

CountStatistic getCreateCount()

Returns the number of connections created.

JSR77.6.23.1.3 getFreePoolSize

BoundedRangeStatistic getFreePoolSize()

Returns the number of free connections in the pool.

JSR77.6.23.1.4 getPoolSize

BoundedRangeStatistic getPoolSize()

Returns the size of the connection pool.

JSR77.6.23.1.5 getWaitingThreadCount

RangeStatistic getWaitingThreadCount()

Returns the number of threads waiting for a connection.

JSR77.6.24 JMSStats extends Stats

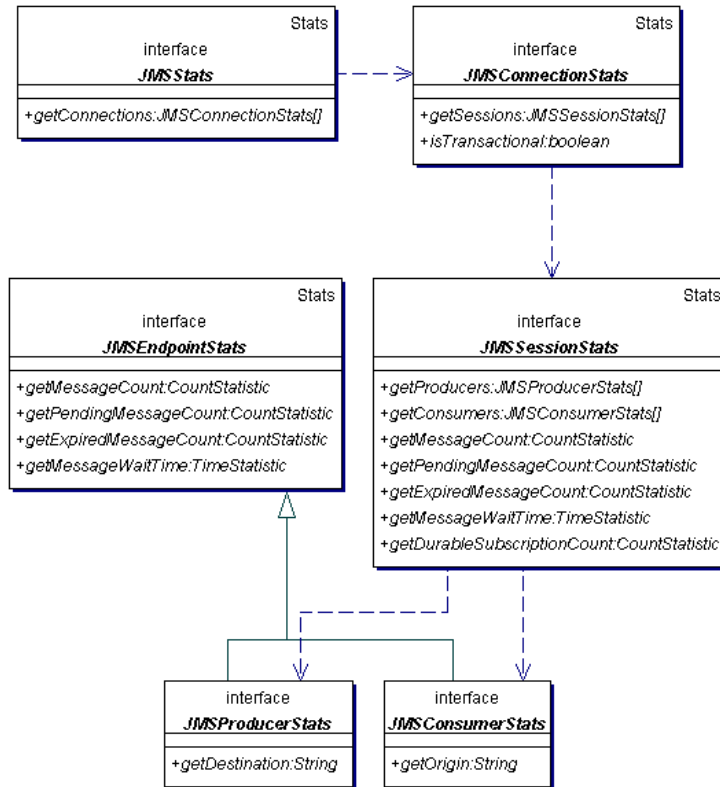


Figure JSR77.6.7 JMSStats, JMSConnectionStats, JMSSessionStats, JMSProducerStats, JMSConsumerStats

The JMSStats interface specifies the statistics provided by a JMS resource.

JSR77.6.24.1 Operation Detail

JSR77.6.24.1.1 getConnections

```
JMSConnectionStats[] getConnections()
```

Returns a list of JMSConnectionStats that provide statistics about the connections associated with the referencing JMS resource.

JSR77.6.25 JMSConnectionStats extends Stats

Specifies the statistics provided by a JMS connection.

JSR77.6.25.1 Operation Detail

JSR77.6.25.1.1 getSessions

JMSSessionStats[] getSessions()

Returns a list of JMSSessionStats that provide statistics about the sessions associated with the referencing JMSConnectionStats.

JSR77.6.25.1.2 isTransactional

boolean isTransactional()

Returns the transactional state of this JMS connection. If true, indicates that this JMS connection is transactional.

JSR77.6.26 JMSSessionStats extends Stats

Specifies the statistics provided by a JMS session.

JSR77.6.26.1 Operation Detail

JSR77.6.26.1.1 getProducers

JMSProducerStats[] getProducers()

Returns a list of JMSProducerStats that provide statistics about the message producers associated with the referencing JMS session statistics.

JSR77.6.26.1.2 getConsumers

JMSConsumerStats[] getConsumers()

Returns a list of JMSConsumerStats that provide statistics about the message consumers associated with the referencing JMS session statistics.

JSR77.6.26.1.3 getDurableSubscriptionCount**CountStatistic getDurableSubscriptionCount()**

Returns the number of durable subscriptions.

JSR77.6.26.1.4 getExpiredMessageCount**CountStatistic getExpiredMessageCount()**

Returns the number of expired messages.

JSR77.6.26.1.5 getMessageCount**CountStatistic getMessageCount()**

Returns the number of messages exchanged.

JSR77.6.26.1.6 getMessageWaitTime**TimeStatistic getMessageWaitTime()**

Returns the time spent by a message before being delivered.

JSR77.6.26.1.7 getPendingMessageCount**CountStatistic getPendingMessageCount()**

Returns the number of pending messages.

JSR77.6.27 JMSEndpointStats extends Stats

Specifies the base interface for the statistics provided by a JMS message producer or a JMS message consumer.

JSR77.6.27.1 Operation Detail

JSR77.6.27.1.1 getExpiredMessageCount

CountStatistic getExpiredMessageCount()

Returns the number of messages that expired before delivery.

JSR77.6.27.1.2 messageCount

CountStatistic getMessageCount()

Returns the number of messages sent or received.

JSR77.6.27.1.3 getMessageWaitTime

TimeStatistic getMessageWaitTime()

Returns the time spent by a message before being delivered.

JSR77.6.27.1.4 getPendingMessageCount

CountStatistic getPendingMessageCount()

Returns the number of pending messages.

JSR77.6.28 JMSProducerStats extends JMSEndpointStats

Specifies the statistics provided by a JMS message producer.

JSR77.6.28.1 Operation Detail***JSR77.6.28.1.1 getDestination***

String getDestination()

Returns a string that encapsulates the identity of a message destination.

JSR77.6.29 JMSConsumerStats extends JMSEndpointStats

Specifies the statistics provided by a JMS message consumer.

JSR77.6.29.1 Operation Detail***JSR77.6.29.1.1 getOrigin***

String getOrigin()

Returns a string that encapsulates the identity of a message origin.

JSR77.6.30 JTAStats extends Stats

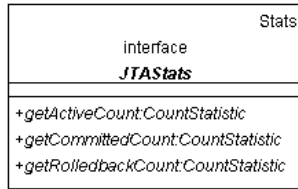


Figure JSR77.6.8 JTAStats

Specifies the statistics provided by a JTA resource.

JSR77.6.30.1 Operation Detail

JSR77.6.30.1.1 getActiveCount

CountStatistic getActiveCount()

Returns the number of active transactions.

JSR77.6.30.1.2 getCommittedCount

CountStatistic getCommittedCount()

Returns the number of committed transactions.

JSR77.6.30.1.3 getRolledbackCount

CountStatistic getRolledbackCount()

Returns the number of rolled-back transactions.

JSR77.6.31 JVMStats extends Stats

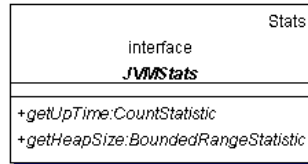


Figure JSR77.6.9 JVMStats

Specifies the statistics provided by a Java VM.

JSR77.6.31.1 Operation Detail

JSR77.6.31.1.1 getHeapSize

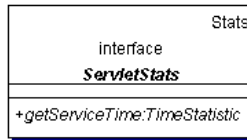
BoundedRangeStatistic getHeapSize()

Returns the size of the JVM's heap.

JSR77.6.31.1.2 getUpTime

CountStatistic getUpTime()

Returns the amount of time the JVM has been running.

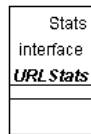
JSR77.6.32 ServletStats extends Stats**Figure JSR77.6.10** ServletStats

Specifies the statistics provided by a Servlet component.

JSR77.6.32.1 Operation Detail**JSR77.6.32.1.1 getServiceTime**

TimeStatistic getServiceTime()

Returns the execution time of the servlets service method.

JSR77.6.33 URLStats extends Stats**Figure JSR77.6.11** URLStats

Specifies the statistics provided by a URL resource.

JSR77.6.33.1 Operation Detail

There are no standard required URL Statistics defined by this specification. The URLStats interface may be extended to provide vendor specific performance statistics.



CHAPTER JSR77.7

J2EE Management EJB Component

The J2EE Management EJB component (MEJB) provides interoperable access to the J2EE Management Model from any J2EE component on all platforms that implement the J2EE Management specification. The MEJB component incorporates the Java Management Extensions (JMX) API, a standard framework for Java object instrumentation.

The MEJB component exposes the managed objects on any J2EE platform as JMX manageable resources as defined by the Java Management Extensions Instrumentation and Agent Specification (JSR003). The MEJB component provides local and remote access of the platform's manageable resources through the EJB interoperability protocol.

JSR77.7.1 MEJB Implementation requirements

All compliant J2EE products must provide an implementation of an Enterprise Session bean component which implements the interfaces specified in this chapter. The MEJB component may be automatically deployed during server installation. A compliant J2EE product must deploy an MEJB component before installation of that product can be considered complete.

The MEJB component provides access to the managed object instances of all the available managed objects in one or more management domains. Compliant implementations of the MEJB component must provide access to all managed object instances required by the J2EE Management Model. All attributes required

by the J2EE Management Model for a standard J2EE managed object type must be accessible from the MEJB component. All operations required by the J2EE Management Model for a standard J2EE managed object type must be invocable from the MEJB component.

JSR77.7.1.1 JMX requirements

The MEJB component exposes the manageable resources on a J2EE platform as JMX Managed Beans (MBeans) and requires an implementation of the JMX public APIs specified by the Java Management Extensions Instrumentation and Agent Specification, v1.1.

JSR77.7.1.2 Remote Interface

The interface `javax.management.j2ee.Management` is the standard remote interface for the MEJB component. The remote interface of the MEJB component implementation must be `javax.management.j2ee.Management` or a subinterface of `javax.management.j2ee.Management`.

JSR77.7.1.3 Home Interface

The interface `javax.management.j2ee.ManagementHome` is the standard home interface for the MEJB component. The home interface of the MEJB component implementation must be `javax.management.j2ee.ManagementHome` or a subinterface of `javax.management.j2ee.ManagementHome`. The `create()` method specified by the `ManagementHome` interface must return a session object that provides access to the entire managed object hierarchy.

JSR77.7.1.4 Naming Requirements

It is recommended that the home interface for the MEJB component be named in the `ejb/mgmt` subcontext. It is recommended that a default name of `ejb/mgmt/MEJB` be used whenever possible.

JSR77.7.1.5 Event Listener Requirements

Management applications that wish to receive events must register an event listener object that implements the `javax.management.NotificationListener` interface specified in the Java Management Extensions Instrumentation and Agent Specification (JSR003).

It is not explicitly required that all event listeners communicate with remote objects, since internal listeners may operate in the same Java VM. In most cases, the event listener registered by the management application will need to receive events from a remote object in order to interoperate with the managed system.

It is the responsibility of the implementation of the `ListenerRegistration` interface returned by the MEJB `getListenerRegistry()` method to implement the appropriate mechanism to receive events from the management system and deliver them to the event listener registered by the client.

The implementation of the `ListenerRegistration` interface is a system value class produced by the J2EE product provider and must obey the interoperability requirements for value classes as specified in the Enterprise JavaBeans specification, “Support for Distribution and Interoperability” section.

Figure JSR77.7.1 illustrates the event delivery paradigm. The listener connector and listener proxy are presented as conceptual types for the proprietary event delivery mechanism supported by a hypothetical implementation of an MEJB. The transport mechanism for remote delivery is also implementation-specific and may use services provided by the J2EE platform such as JMS or RMI.

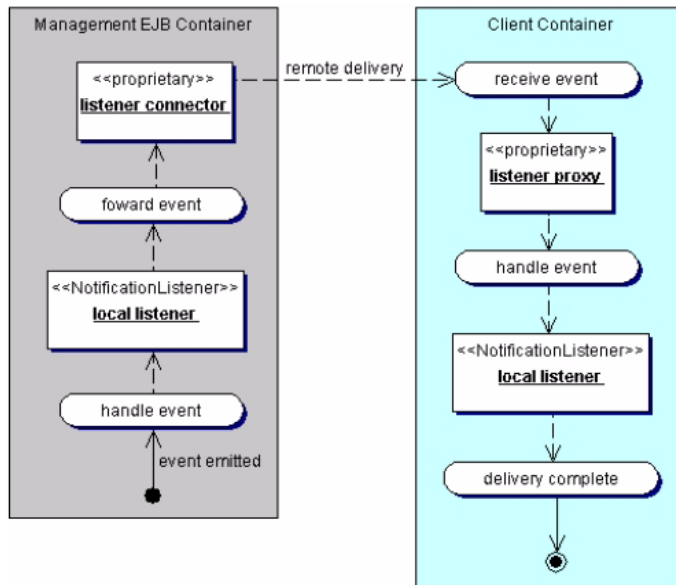


Figure JSR77.7.1 Event delivery

JSR77.7.2 Using the Management EJB component

This section provides code examples of a J2EE management component using the Management EJB component.

JSR77.7.2.1 Creating a MEJB Session Object

The code fragment in Code Example 7.2 illustrates how a J2EE component creates the MEJB session object.

```
Context ic = new InitialContext();
java.lang.Object objref = ic.lookup(ejbName);
ManagementHome home =
    (ManagementHome)PortableRemoteObject.narrow(objref,ManagementHome.class);
Management mejb = home.create();
```

Code Example 7.2 creating a MEJB session object from a J2EE application

JSR77.7.2.2 Finding managed objects of a certain type

The code fragment in Code Example 7.3 illustrates how the queryNames method can be used to find managed objects of a certain type.

```
/* Find all J2EEApplication managed objects on this system */
ObjectName searchpattern = new ObjectName("*:j2eeType=J2EEApplication,*");
Set managed_object_set = mejb.queryNames(searchpattern, null);
System.out.println("found " + managed_object_set.size() + " matching objects.");
```

Code Example 7.3 finding all J2EEApplication types

JSR77.7.2.3 Retrieving attribute information

The code fragment in Code Example 7.4 illustrates how the attributes of a managed object can be retrieved and their values determined.

```
Iterator managed_objects = managed_object_set.iterator();
while (managed_objects.hasNext()) {
    ObjectName objectname = (ObjectName)managed_objects.next();
    System.out.println(objectname);
    /* get MBeanInfo and print the info */
    MBeanInfo moi = mejb.getMBeanInfo(objectname);
    MBeanAttributeInfo[] atts = moi.getAttributes();
    for (int i = 0; i < atts.length; i++) {
        System.out.println("**** Attribute Info ****");
        System.out.println("Name " + atts[i].getName());
        System.out.println("Type " + atts[i].getType());
        System.out.println("isIs " + atts[i].isIs());
        System.out.println("isReadable " + atts[i].isReadable());
        System.out.println("isWritable " + atts[i].isWritable());
        System.out.println("value " + mejb.getAttribute(objectname, atts[i].getName()));
    }
}
```

Code Example 7.4 retrieving attribute information from a set of managed objects

JSR77.7.2.4 Invoking methods

The code fragment in Code Example 7.5 illustrates how a method with the signature `void test()` can be invoked.

```
mejb.invoke(appname, "test", new Object [] {}, new String [] {});
```

Code Example 7.5 invoking a method which takes no arguments

The code fragment in Code Example 7.6 illustrates how a method with the signature `String test(String, int)` can be invoked.

```
String sig1 = String.class.getName();
String sig2 = Integer.TYPE.getName();
String [] signature = new String [] {sig1, sig2};
Object [] params = new Object [] {"This is a Test", new Integer(77)};
String s = (String) mejb.invoke(appname, "test", params, signature);
```

Code Example 7.6 invoking a method with two arguments (String, int)

JSR77.7.2.5 Registering an event notification listener

The code fragment in Code Example 7.7 illustrates how the client registers an event listener.

```
/* register an event listener */
objectname = new ObjectName(domainname + ":type=J2EEDomain,name=Hans' J2EE Domain");
System.out.println("addNotificationListener( " + listener + " ) to " + objectname);
ListenerRegistration lr = mejb.getListenerRegistry();
lr.addNotificationListener(objectname, listener, null, "MEJBTester");
```

Code Example 7.7 registering an event listener

JSR77.7.3 package javax.management.j2ee

This section includes the complete API documentation for the J2EE Management EJB component interfaces.

All Interfaces Summary	
ListenerRegistration	Provides the methods to add and remove event listeners
Management	The remote interface for the MEJB
ManagementHome	The Home interface for the MEJB

JSR77.7.3.1 Interface *ListenerRegistration*

package: javax.management.j2ee

[public class ListenerRegistration implements java.io.Serializable](#)

Provides the methods to add and remove event listeners to the MEJB component.

Method Summary	
public void	addNotificationListener (javax.management.ObjectName name, javax.management.NotificationListener listener, javax.management.NotificationFilter filter, Object handback) Add a listener to a registered managed object.
public void	removeNotificationListener (javax.management.ObjectName name, javax.management.NotificationListener listener) Remove a listener from a managed object.

JSR77.7.3.1.1 Method Detail

addNotificationListener

```
public void addNotificationListener(javax.management.ObjectName name,  
    javax.management.NotificationListener listener, javax.manage-  
    ment.NotificationFilter filter, Object handback)
```

Add a listener to a registered managed object.

Throws:

javax.management.InstanceNotFoundException
java.rmi.RemoteException

Parameters:

`name` - The name of the managed object on which the listener should be added.

`listener` - The listener object which will handle the events emitted by the registered managed object.

`filter` - The filter object. If filter is null, no filtering will be performed before handling events.

`handback` - An opaque object to be sent back to the listener when a notification is emitted which helps the listener to associate information regarding the MBean emitter. This object cannot be used by the Notification broadcaster object. It should be resent unchanged with the notification to the listener.

Exception doc:

`javax.management.InstanceNotFoundException` - The managed object name provided does not match any of the registered managed objects.

`java.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

removeNotificationListener

```
public void removeNotificationListener(javax.management.ObjectName name,  
    javax.management.NotificationListener listener)
```

Enables to remove a listener from a registered managed object.

Throws:

- javax.management.InstanceNotFoundException
- javax.management.ListenerNotFoundException
- java.rmi.RemoteException

Parameters:

name - The name of the managed object on which the listener should be removed.

listener - The listener object which will handle the events emitted by the registered managed object. This method will remove all the information related to this listener.

Exception doc:

javax.management.InstanceNotFoundException - The managed object name provided does not match any of the registered managed objects.

javax.management.ListenerNotFoundException - The listener is not registered in the managed object.

java.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

JSR77.7.3.2 Interface Management

package: javax.management.j2ee

public interface Management extends javax.ejb.EJBObject

The Management interface provides the APIs to navigate and manipulate managed objects. The Management EJB component must implement this interface as its remote interface.

Method Summary	
public Object	getAttribute (javax.management.ObjectName name, String attribute) Gets the value of a specific attribute of a named managed object.
public AttributeList	getAttributes (javax.management.ObjectName name, String[] attributes) Enables the values of several attributes of a named managed object.
public String	getDefaultDomain () Returns the default domain name of this MEJB.
public Integer	getMBeanCount () Returns the number of managed objects registered in the MEJB.
public MBeanInfo	getMBeanInfo (javax.management.ObjectName name) Returns an object that allows the retrieval of the attributes and operations that a managed object exposes for management.
public Object	invoke (javax.management.ObjectName name, String operationName, Object[] params, String[] signature) Invokes an operation on a managed object.
public boolean	isRegistered (javax.management.ObjectName name) Checks whether a managed object, identified by its object name, is already registered with the MEJB.
public Set	queryNames (javax.management.ObjectName name, javax.management.QueryExp query) Gets the names of managed objects controlled by the MEJB.
public void	setAttribute (javax.management.ObjectName name, javax.management.Attribute attribute) Sets the value of a specific attribute of a named managed object.
public AttributeList	setAttributes (javax.management.ObjectName name, javax.management.AttributeList attributes) Sets the values of several attributes of a named managed object.

public <code>ListenerRegistration</code>	<code>getListenerRegistry()</code> Returns the <code>ListenerRegistration</code> implementation for this MEJB component.
--	---

JSR77.7.3.2.1 Method Detail**getAttribute**

```
public Object getAttribute(javax.management.ObjectName name, String attribute)
```

Gets the value of a specific attribute of a named managed object. The managed object is identified by its object name.

Throws:

`javax.management.MBeanException`
`javax.management.AttributeNotFoundException`
`javax.management.InstanceNotFoundException`
`javax.management.ReflectionException`
`java.rmi.RemoteException`

Parameters:

`name` - The object name of the managed object from which the attribute is to be retrieved.
`attribute` - A String specifying the name of the attribute to be retrieved.

Returns:

The value of the retrieved attribute.

Exception doc:

`javax.management.AttributeNotFoundException` - The attribute specified is not accessible in the managed object.
`javax.management.MBeanException` - Wraps an exception thrown by the managed object's getter.
`javax.management.InstanceNotFoundException` - The managed object specified is not registered in the MEJB.
`javax.management.ReflectionException` - An exception occurred when trying to invoke the `getAttribute` method of a Dynamic MBean.
`java.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

getAttributes

```
public javax.management.AttributeList getAttributes(javax.management.ObjectName name, String[] attributes)
```

Gets the values of several attributes of a named managed object. The managed object is identified by its object name.

Throws:

- javax.management.InstanceNotFoundException
- javax.management.ReflectionException
- java.rmi.RemoteException

Parameters:

- `name` - The object name of the managed object from which the attributes are retrieved.
- `attributes` - A list of the attributes to be retrieved.

Returns:

An instance of `javax.management.AttributeList` which contains a list of the retrieved attributes as `javax.management.Attribute` instances.

Exception doc:

- `javax.management.InstanceNotFoundException` - The managed object specified is not registered in the MEJB.
- `javax.management.ReflectionException` - An exception occurred when trying to invoke the `getAttributes` method of a Dynamic MBean.
- `java.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

getDefaultDomain

```
public String getDefaultDomain()
```

Returns the default domain name of this MEJB.

Throws:

- java.rmi.RemoteException

Exception Doc:

- `java.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

getMBeanCount

```
public Integer getMBeanCount()
```

Returns the number of managed objects registered in the MEJB.

Throws:

java.rmi.RemoteException

Exception doc:

javax.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

getMBeanInfo

```
public javax.management.MBeanInfo getMBeanInfo(javax.management.Object-  
Name name)
```

This method discovers the attributes and operations that a managed object exposes for management.

Throws:

javax.management.IntrospectionException

javax.management.InstanceNotFoundException

javax.management.ReflectionException

java.rmi.RemoteException

Parameters:

name - The object name of the managed object to analyze.

Returns:

An instance of `javax.management.MBeanInfo` allowing the retrieval of all attributes and operations of this managed object.

Exception doc:

javax.management.IntrospectionException An exception occurred during introspection.

javax.management.InstanceNotFoundException The managed object specified was not found.

javax.management.ReflectionException An exception occurred when trying to perform reflection on the managed object.

javax.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

invoke

```
public Object invoke(javax.management.ObjectName name, String operation-
    Name, Object[] params, String[] signature)
```

Invokes an operation on a managed object.

Throws:

- javax.management.InstanceNotFoundException
- javax.management.MBeanException
- javax.management.ReflectionException
- java.rmi.RemoteException

Parameters:

- `name` - The object name of the managed object on which the method is to be invoked.
- `operationName` - The name of the operation to be invoked.
- `params` - An array containing the parameters to be set when the operation is invoked
- `signature` - An array containing the signature of the operation. Each element of the array contains a fully-qualified name of the entity (class, interface, array class, primitive type) that corresponds with a parameter type in the method's signature. The format of the strings must be as specified by `java.lang.Class.getName()`. The class objects will be loaded using the same class loader as the one used for loading the managed object on which the operation was invoked.

Returns:

The object returned by the operation, which represents the result of invoking the operation on the managed object specified.

Exception doc:

- `javax.management.InstanceNotFoundException` - The managed object specified is not registered in the MEJB.
- `javax.management.MBeanException` - Wraps an exception thrown by the managed object's invoked method.
- `javax.management.ReflectionException` - Wraps a `java.lang.Exception` thrown while trying to invoke the method.
- `java.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

isRegistered

```
public boolean isRegistered(javax.management.ObjectName name)
```

Checks whether a managed object, identified by its object name, is already registered with the MEJB.

Throws:

java.rmi.RemoteException

Parameters:

name - The object name of the managed object to be checked.

Returns:

True if the managed object is already registered in the MEJB, false otherwise.

Exception doc:

javax.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

queryNames

```
public java.util.Set queryNames(javax.management.ObjectName name, javax.management.QueryExp query)
```

Gets the names of managed objects controlled by the MEJB. This method enables any of the following to be obtained: The names of all managed objects, the names of a set of managed objects specified by pattern matching on the ObjectName and/or a query expression, a specific managed object name (equivalent to testing whether a managed object is registered). When the object name is null or no domain and key properties are specified, all objects are selected. It returns the set of ObjectNames for the managed objects selected.

Throws:

java.rmi.RemoteException

Parameters:

name - The object name pattern identifying the managed objects to be retrieved. If null or no domain and key properties are specified, all the managed objects registered will be retrieved.

query - The query expression to be applied for selecting managed objects. If null no query expression will be applied for selecting managed objects.

Returns:

A set containing the ObjectNames for the managed objects selected. If no managed object satisfies the query, an empty set is returned.

Exception doc:

`javax.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

setAttribute

```
public void setAttribute(javax.management.ObjectName name, javax.management.Attribute attribute)
```

Sets the value of a specific attribute of a named managed object. The managed object is identified by its object name.

Throws:

`javax.management.InstanceNotFoundException`
`javax.management.AttributeNotFoundException`
`javax.management.InvalidAttributeValueException`
`javax.management.MBeanException`
`javax.management.ReflectionException`
`java.rmi.RemoteException`

Parameters:

`name` - The name of the managed object within which the attribute is to be set.
`attribute` - The identification of the attribute to be set and the value it is to be set to.

Returns:

The value of the attribute that has been set.

Exception doc:

`javax.management.InstanceNotFoundException` - The managed object specified is not registered in the MEJB.
`javax.management.AttributeNotFoundException` - The attribute specified is not accessible in the managed object.
`javax.management.InvalidAttributeValueException` - The value specified for the attribute is not valid.
`javax.management.MBeanException` - Wraps an exception thrown by the managed object's setter.
`javax.management.ReflectionException` - An exception occurred when trying to invoke the `setAttribute` method of a Dynamic MBean.
`javax.rmi.RemoteException` - a communication exception occurred during the execution of a remote method call.

setAttributes

```
public javax.management.AttributeList setAttributes(javax.management.ObjectName name, javax.management.AttributeList attributes)
```

Sets the values of several attributes of a named managed object. The managed object is identified by its object name.

Throws:

- javax.management.InstanceNotFoundException
- javax.management.ReflectionException
- java.rmi.RemoteException

Parameters:

- name - The object name of the managed object within which the attributes are to be set.
- attributes - A list of attributes: The identification of the attributes to be set and the values they are to be set to.

Returns:

The list of attributes that were set, with their new values.

Exception doc:

- javax.management.InstanceNotFoundException - The managed object specified is not registered in the MEJB.
- javax.management.ReflectionException - An exception occurred when trying to invoke the setAttributes method of a Dynamic MBean.
- javax.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

getListenerRegistry

```
public ListenerRegistration getListenerRegistry()
```

Returns the ListenerRegistration implementation for the MEJB component implementation which allows the client to register a event notification listener.

Throws:

- java.rmi.RemoteException

Returns:

An implementation of ListenerRegistration.

Exception doc:

- javax.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

JSR77.7.3.3 Interface *ManagementHome*

package: javax.management.j2ee

public interface *ManagementHome* extends javax.ejb.EJBHome

The required home interface for the Management EJB component. The interface may be extended by a proprietary interface to include additional create methods that take initialization arguments. A J2EE client must be able to create a compliant session object using the specified create method which takes no arguments.

Method Summary	
public <i>Management</i>	create() Create an MEJB session object.

JSR77.7.3.3.1 Method Detail

create

public *Management* create()

Throws:

javax.ejb.CreateException
java.rmi.RemoteException

Creates an MEJB session object which provides access to the J2EE Management Model.

Returns:

An session object which implements
javax.management.j2ee.Management.

Exception doc:

javax.ejb.CreateException - indicates a failure to create an EJB Object.
java.rmi.RemoteException - a communication exception occurred during the execution of a remote method call.

CHAPTER JSR77.8

J2EE Management SNMP

This chapter defines a portion of the Management Information Base (MIB) for use with network and systems management protocols in the Internet community. More specifically, it describes a set of managed objects that allow for the monitoring of the Java 2 Enterprise Edition Platform.

JSR77.8.1 Introduction

As part of the Java Community Process (see <http://www.jcp.org/>), JSR 77 was formed to define a management information model for the Java 2 Enterprise Edition Platform, the J2EE Management Model. The J2EE Management Model is a UML model which describes the structure, format, and semantics of management data for the J2EE Platform.

This chapter defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community which is consistent with that model as well as with the Structure of Management Information defined for the SNMP Management Framework.

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 RFC 2119 [RFC2119].

JSR77.8.2 The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- An overall architecture, described in RFC 2571 [RFC2571].
- Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIV1 and described in STD 16, RFC 1155 [RFC1155], STD 16, RFC 1212 [RFC1212] and RFC 1215 [RFC1215]. The second version, called SMIV2, is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].
- Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in STD 15, RFC 1157 [RFC1157]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [RFC1901] and RFC 1906 [RFC1906]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [RFC1906], RFC 2572 [RFC2572] and RFC 2574 [RFC2574].
- Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in STD 15, RFC 1157 [RFC1157]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [RFC1905].
- A set of fundamental applications described in RFC 2573 [RFC2573] and the view-based access control mechanism described in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMPv3 Management Framework can be found in RFC 2570 [RFC2570]. A framework, termed the AgentX framework, which extends the SNMP Management Framework and can accommodate environments wherein multiple components might need to have a shared mechanism for sending and receiving SNMP protocol messages, is described in RFC 2741 [RFC2741].

In SNMP, managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This chapter specifies a MIB module that is compliant to the SMIV2. A MIB conforming to the SMIV1 can be produced through the appropriate translations. Some machine readable information in SMIV2 will be converted into textual descriptions in SMIV1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

JSR77.8.3 Overview

The J2EE MIB module defined in this chapter can be used to discover and monitor details about J2EE servers. The status of the server and its components is reported as MIB objects which can be polled, and MAY also be reported as traps. OPTIONAL performance statistics are also defined.

The J2EE management model allows multiple servers on a single node as well as single servers spanning multiple nodes. The MIB is defined and indexed in support of this, and permits many but not all alternatives in agent and instrumentation architectures; these MAY be based on either monolithic or AgentX-like SNMP frameworks. All of the following are supported:

- Components of different multi-node servers may coexist on a node, from one or more vendors.
- The agent (or AgentX master agent, or, less likely, the AgentX subagent) might reside on a node which does not host any component of a server.
- A J2EE management domain (part of the J2EE management model) can span multiple servers and therefore multiple nodes.
- Servers and domains may share nodes with other SNMP-managed components.
- An AgentX-like master agent might be present, with sessions to one or more subagents having access to a server's instrumentation.

An SNMP agent (or agentx-like subagent) supporting this MIB MUST assemble and present an integrated view of any server, whether single node or multi-node, for which it has created a row in the server table (j2eeSrvTable).

JSR77.8.4 The Structure of the MIB

The following guidelines were followed for the MIB:

- Managed object types from the model become conceptual tables
- Some attribute lists for certain managed object types also become conceptual tables
- Attributes inherited from supertypes become conceptual columns in tables for all managed object types inheriting that attribute
- Key property lists used to assign unique names to managed objects become the indices into a conceptual table entry
- Events are mapped into traps
- State management is embedded into the individual tables
- The Performance Data Framework become augmentations of conceptual tables (or were embedded into the tables directly in cases where the table was specific to a particular managed object); those augmentations (or their embedded equivalents) are organized into object groups which are referenced in module compliance macros in the definition of what subsets of the MIB may be implemented while still conforming to this definition.
- Truth values indicate on a per-table basis at runtime whether an optional feature (events, state management, performance statistics) is supported.
- Only a subset of J2EE management events are issued as traps. No creation, deletion, or attribute changes are reported as traps. Only “core” (see J2eeSMState) state changes are issued, and only for servers, resources, entity beans, and message-driven beans.

Note – All strings defined in the MIB are up to 255 characters in length; no such limitation has been specified in the attributes for the UML model.

Note – It is considered out-of-scope for this MIB to include objects which could facilitate the discovery of an authoritative source for MIB information about multi-node servers. Such a mapping is considered an SNMP client concern.

JSR77.8.5 Definitions

Note – The MIB is included in this section for reference only. The actual MIB file is available from <http://java.sun.com/j2ee/management>

```
J2EE-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,  
Integer32, Unsigned32, TimeTicks, enterprises  
FROM SNMPv2-SMI
```

```
TruthValue, DateAndTime, TAddress  
FROM SNMPv2-TC
```

```
MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP  
FROM SNMPv2-CONF
```

```
SnmpAdminString  
FROM SNMP-FRAMEWORK-MIB;
```

```
sun OBJECT IDENTIFIER ::= { enterprises 42 }
```

```
-- To be allocated through SMPO
```

```
--
```

```
jcp OBJECT IDENTIFIER ::= { sun 2 9999 }
```

```
j2ee OBJECT IDENTIFIER ::= { jcp 1 }
```

```
j2eeMIB MODULE-IDENTITY
```

```
LAST-UPDATED "200203080000Z"
```

```
ORGANIZATION "JSR 77 Expert Group"
```

```
CONTACT-INFO
```

```
"WG EMail: http://groups.yahoo.com/group/eemgmt-experts
```

```
See also: http://www.jcp.org/jsr/detail/77.jsp
```

Author: Anthony Daniel
 EMail: jsr77work@yahoo.com"

DESCRIPTION

"This MIB module defines a set of objects for monitoring the J2EE platform."

REVISION "200107020000Z" -- July 2, 2001

DESCRIPTION

"Initial version, for expert group review."

REVISION "200112170000Z" -- December 17, 2001

DESCRIPTION

"Revised version, for public review.

Updates include the following:

- Added domains, nodes, JVMs.
- Dropped more detailed tables for resources whose instrumentation was eliminated from the model.
- Clarified that state-manageable objects which are stopped can be restarted.
- Introduced a textual convention for managed object names.
- MIB was made private to JCP.org; it can be re-rooted when published as RFC.
- Section 5 added.
- Introductory sections fleshed out.
- Added Enterprise numbers for Domains, Servers, JVMs, to make vendor checks more authoritative for JSR77 clients.
- Added NOTIFICATION-TYPEs (but left OBJECT list empty).
- Some generic managed object and state manageable attributes left unexpanded.
- Still missing JVM, JDBC, and JMS performance statistics; left inconsistent the fact that sometimes augmentation is used, sometimes the objects are just embedded in the table.
- Still missing conformance and compliance information.

"

REVISION "200203050000Z" -- March 5, 2002

DESCRIPTION

"Revised version, corrected for syntax checking.

- Removed all references to Nodes.
- Filled all replicated OBJECT-TYPES corresponding to J2EEManagedObject attributes.
- Added conformance statements.
- Filled OBJECTS list in NOTIFICATION-TYPES
- Replaced AUGMENTS clauses by INDEX - as the intended purpose seemed to define a 1-[0..1] relationship rather than a 1-1 relationship.

- Slightly reorganized the OID tree structure.
- All Statistics now defined as separate tables.
- Added some Revisit: tags when in doubt.

Revised by Daniel.Fuchs@sun.com

"

REVISION "200203080000Z" -- March 8, 2002

DESCRIPTION "

- Corrected the NOTIFICATION-TYPE OIDs to bring them in conformance to RFC 2576 (coexistence draft).

"

REVISION "200204160000Z" -- April 16, 2002

DESCRIPTION

"Revised version, included missing performance statistics.

- Separated EJB stats as different tables as per EJB type. In future will be useful in extending the EJB stats and avoid redundant information.
- Added JCA connection and connection pool stat tables.
- Added JDBC connection and connection pool stat tables.
- Added JMS sessions, producer, consumer stat tables.
- Added JVM stat table.
- Removed j2eeDomIndex(redundant data) from all tables except j2eeDomTable and j2eeSrvTable.
- Changed j2eeSrvVersion access from read-create to read-only
- Added j2eeJVMNode to j2eeJVMTTable
- Changed syntax of 'Enterprise' attribute in tables from Integer32 to OBJECT IDENTIFIER, incline to SNMP standard.
- Bug fix: j2eeSrvVersion added to the J2eeSrvEntry sequence

Revision introduced by hyther@adventnet.com

"

::= { j2ee 1 } -- OID assigned by JCP PMO

 --
 -- Textual Conventions
 --

J2eeMoName ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION "

A textual identifier used for names of J2EE Managed Objects.

To facilitate internationalization, a J2eeMoName is represented using the ISO/IEC IS 10646-1 character set, encoded as an octet string using the UTF-8 transformation format described in [RFC2279].

A J2eeMoName cannot be null, and cannot contain a colon (':'), comma (','), equal sign ('='), asterisk (*), or question mark ('?').

"

SYNTAX OCTET STRING (SIZE (1..255))

J2eeSMState ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION "

The operational status of a 'StateManageable' object within the JSR77 model. Six states are possible. The three core states are RUNNING, STOPPED, and FAILED. Extended states are STARTING, STOPPING, and OTHER. Core states MUST be supported; extended states SHOULD be supported.

RUNNING is the normal operational status of a StateManageable object. It indicates that the StateManageable object is fully operational from the perspective of the JSR77 model. Thus requests on MIB objects associated with this StateManageable object will return results as defined by the JSR77 model.

STOPPED indicates that the associated StateManageable object is quiescent, inactive, or nonexistent. It also indicates that the

associated object can be restarted. Requests on MIB objects associated with this StateManageable object will return results as defined by the JSR77 model.

FAILED is exactly the same as STOPPED except that the StateManageable object has become stopped for unexpected reasons, and some form of intervention, perhaps out-of-band, is required.

STARTING indicates that the associated StateManageable object has begun the process of becoming fully operational. This state indicates that the StateManageable object is in a setup or initialization phase, and consequently has undefined behavior from the perspective of the JSR77 model. Requests on MIB objects associated with this StateManageable object may return valid results, invalid results, or errors.

STOPPING indicates that the associated StateManageable object has begun the process of becoming quiescent, inactive, or nonexistent. Requests on MIB objects associated with this StateManageable object may return valid results, invalid results, or errors.

OTHER means none of the above apply.

"

```
SYNTAX  INTEGER { other(1),
                 failed(2), stopped(3), running(4),
                 starting(5), stopping(6) }
```

J2eeSMStartTime ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION "

A sequence of octets, following the SNMPv2-TC DateAndTime format,

representing the date and time, to the nearest hundredth of a second, at which the Managed Object has most recently entered the running(4) state.

"

SYNTAX DateAndTime

```

--
-- The groups defined within this MIB module:
--
-----
--
-- Object Identifiers
--
-----

j2eeObjects    OBJECT IDENTIFIER ::= { j2eeMIB 1 1 }
j2eeStatistics OBJECT IDENTIFIER ::= { j2eeMIB 1 2 }
j2eeNotifications OBJECT IDENTIFIER ::= { j2eeMIB 2 }
j2eeConformance OBJECT IDENTIFIER ::= { j2eeMIB 3 }

-----
--
-- J2EE Managed Object Group
--
-----

-- Revisit: OBJECT-IDENTITY might be preferred here
--
j2eeMoGroup    OBJECT IDENTIFIER ::= { j2eeObjects 1 }

-----
--
-- J2EE J2EEDomain Managed Objects
--

```

j2eeDomTable OBJECT-TYPE
 SYNTAX SEQUENCE OF J2eeDomEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table lists the J2EE Management Domains which have a
 presence on a system."
 ::= { j2eeMoGroup 1 }

j2eeDomEntry OBJECT-TYPE
 SYNTAX J2eeDomEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "An entry describing a J2EE Management Domain."
 INDEX { j2eeDomIndex }
 ::= { j2eeDomTable 1 }

J2eeDomEntry ::= SEQUENCE {
 j2eeDomIndex Integer32,
 j2eeDomMoName J2eeMoName,
 j2eeDomEnterprise OBJECT IDENTIFIER,
 j2eeDomMoStateManaged TruthValue,
 j2eeDomMoStatProv TruthValue,
 j2eeDomMoEventProv TruthValue,
 j2eeDomSMState J2eeSMState,
 j2eeDomSMStartTime J2eeSMStartTime
 }

j2eeDomIndex OBJECT-TYPE
 SYNTAX Integer32 (1..2147483647)
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "The locally arbitrary, but unique identifier associated
 with each J2EE management domain entry.

The value is expected to remain constant at least from one
 re-initialization of the entity's management system

to the next re-initialization."
 ::= { j2eeDomEntry 1 }

j2eeDomMoName OBJECT-TYPE

SYNTAX J2eeMoName

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The name of the domain. This attribute is required because the JSR77 model requires servers to be managed objects, and all managed objects have names."

::= { j2eeDomEntry 2 }

j2eeDomEnterprise OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The SMI Network Management Private Enterprise Code assigned by IANA to the vendor providing the J2EE Management software for this domain. The value should be .1.3.6.1.4.1.x , where x is the enterprise number"

::= { j2eeDomEntry 3 }

j2eeDomMoStateManaged OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if the domain is state-manageable as defined by the JSR77 model for all managed objects."

::= { j2eeDomEntry 4 }

j2eeDomMoStatProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if the domain provides statistics as

defined by the JSR77 model for managed objects."
 ::= { j2eeDomEntry 5 }

j2eeDomMoEventProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if the domain supports event notification and is currently configured to send traps for all events defined and available from all event providers in the domain."

::= { j2eeDomEntry 6 }

j2eeDomSMState OBJECT-TYPE

SYNTAX J2eeSMState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of the domain, which is based on the state of all of the servers within the domain. If all servers are in the running(4) state, the domain's state is running(4). Otherwise, the domain's state is the least functional state that one or more of its servers is at. The states, ordered from most to least functional are: running(4), starting(5), other(1), stopping(6), stopped(3), failed(2)."

::= { j2eeDomEntry 7 }

j2eeDomSMStartTime OBJECT-TYPE

SYNTAX J2eeSMStartTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A sequence of octets, following the SNMPv2-TC DateAndTime format, representing the date and time, to the nearest hundredth of a second, at which the domain has most recently entered the running(4) state."

::= { j2eeDomEntry 8 }

```

--
-- J2EE J2EE Server Managed Objects
--
-----

j2eeSrvTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF J2eeSrvEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "This table lists the J2EE Servers present on a system.
        The J2EE resources and applications has parent relationship
        with the J2EE server. Since j2eeSrvIndex is an arbitrary
        unique identifier for any given domain, it is enough
        to use j2eeSrvIndex as external index to the J2EE resources
        and application tables and need not use j2eeDomIndex as
        external index.
        "
    ::= { j2eeMoGroup 2 }

-- Revisit: Do we have a 1-N relationship between Domains and Servers?
--     Are all servers in at least and at most 1 domain?
--     If not we have to introduce a j2eeDomainServerRelationTable,
--     and suppress the j2eeDomIndex from all tables in which
--     it appears.
-- Solution : Since j2eeSrvIndex is an arbitrary
--     unique identifier for any given domain, it is enough
--     to use j2eeSrvIndex as external index to the J2EE resources
--     and application tables and need not use j2eeDomIndex as
--     external index.

j2eeSrvEntry OBJECT-TYPE
    SYNTAX     J2eeSrvEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "An entry describing a J2EE Server in a J2EE Management Domain."
    INDEX { j2eeDomIndex, j2eeSrvIndex }
    ::= { j2eeSrvTable 1 }

```



```

J2eeSrvEntry ::= SEQUENCE {
    j2eeSrvIndex      Integer32,
    j2eeSrvMoName     J2eeMoName,
    j2eeSrvEnterprise OBJECT IDENTIFIER,
    j2eeSrvVendor     SnmpAdminString,
    j2eeSrvVersion    SnmpAdminString,
    j2eeSrvMoStateManaged TruthValue,
    j2eeSrvMoStatProv TruthValue,
    j2eeSrvMoEventProv TruthValue,
    j2eeSrvSMState    J2eeSMState,
    j2eeSrvSMStartTime J2eeSMStartTime
}

```

j2eeSrvIndex OBJECT-TYPE
 SYNTAX Integer32 (1..2147483647)
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "The locally arbitrary, but unique identifier associated
 with each J2EE server entry.

The value is expected to remain constant at least from one
 re-initialization of the entity's management system
 to the next re-initialization."

```
 ::= { j2eeSrvEntry 1 }
```

j2eeSrvMoName OBJECT-TYPE
 SYNTAX J2eeMoName
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "The name of the server. This attribute is required because
 the JSR77 model requires servers to be managed objects,
 and all managed objects have names."

```
 ::= { j2eeSrvEntry 2 }
```

j2eeSrvEnterprise OBJECT-TYPE
 SYNTAX OBJECT IDENTIFIER
 MAX-ACCESS read-only
 STATUS current

DESCRIPTION

"The SMI Network Management Private Enterprise Code assigned by IANA to the J2EE platform vendor for the server. The value should be .1.3.6.1.4.1.x , where x is the enterprise number"

::= { j2eeSrvEntry 3 }

j2eeSrvVendor OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A human-readable string identifying the J2EE platform vendor for the server."

::= { j2eeSrvEntry 5 }

j2eeSrvVersion OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A human-readable string identifying the version of the Server"

::= { j2eeSrvEntry 6 }

j2eeSrvMoStateManaged OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if the server is state-manageable as defined by the JSR77 model for all managed objects."

::= { j2eeSrvEntry 7 }

j2eeSrvMoStatProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if the server provides statistics as defined by the JSR77 model for managed objects."

::= { j2eeSrvEntry 8 }

j2eeSrvMoEventProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if the server provides events as defined by the JSR77 model for managed objects."

::= { j2eeSrvEntry 9 }

j2eeSrvSMState OBJECT-TYPE

SYNTAX J2eeSMState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of the server, following state-manageable object semantics."

::= { j2eeSrvEntry 10 }

j2eeSrvSMStartTime OBJECT-TYPE

SYNTAX J2eeSMStartTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A sequence of octets, following the SNMPv2-TC DateAndTime format, representing the date and time, to the nearest hundredth of a second, at which the server has most recently entered the running(4) state."

::= { j2eeSrvEntry 11 }

 --
 -- J2EE JVM Managed Objects
 --

j2eeJVMTTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeJVMTEntry

MAX-ACCESS not-accessible

```

STATUS    current
DESCRIPTION
    "This table lists all JVMs instances that a particular
    server utilizes."
 ::= { j2eeMoGroup 3 }

j2eeJVMEEntry OBJECT-TYPE
SYNTAX    J2eeJVMEEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "An entry for a JVM utilized by a particular J2EE server."
INDEX { j2eeSrvIndex, j2eeJVMIIndex }
 ::= { j2eeJVMTTable 1 }

J2eeJVMEEntry ::= SEQUENCE {
    j2eeJVMIIndex      Integer32,
    j2eeJVMEVersion    SnmpAdminString,
    j2eeJVMEVendor     SnmpAdminString,
    j2eeJVMEEnterprise OBJECT IDENTIFIER,
    j2eeJVMENode       SnmpAdminString

-- The following are implied by the J2EE management model
--
-- Revisit: Shouldn't this information be left anyway for coherency?
--     If the name of the JVM appear in some other traces
--     coming form the Object Model, for instance?
--
-- j2eeJVMMoName      J2eeMoName, -- JVM don't need no stinkin' name
-- j2eeJVMMoStateManaged TruthValue,
-- j2eeJVMMoStatProv  TruthValue,
-- j2eeJVMMoEventProv TruthValue,
-- j2eeJVMSMState     J2eeSMState,
-- j2eeJVMSMStartTime J2eeSMStartTime
}

j2eeJVMIIndex OBJECT-TYPE
SYNTAX    Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS    current

```

DESCRIPTION

"The locally arbitrary, but unique identifier associated with each J2EE JVM entry. Note that the existence of this index permits a server to make use of multiple JVMs on a single node.

The value is expected to remain constant at least from one re-initialization of the entity's management system to the next re-initialization."

::= { j2eeJVMEEntry 1 }

j2eeJVMVersion OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A human-readable string identifying the Java Runtime Environment version of the JVM. It must be identical to the value of the system property java.version."

::= { j2eeJVMEEntry 2 }

j2eeJVMVendor OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A human-readable string identifying the Java Runtime Environment vendor of the JVM. It must be identical to the value of the system property java.vendor."

::= { j2eeJVMEEntry 3 }

j2eeJVMEnterprise OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The SMI Network Management Private Enterprise Code assigned by IANA to the Java Runtime Environment vendor of the JVM. The value should be .1.3.6.1.4.1.x , where x is the enterprise

```

        number"
 ::= { j2eeJVMEEntry 5 }

j2eeJVMEEntry OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Identifies the node (machine) this JVM is running on. The value
    of the node attribute must be fully qualified hostname of the node
    the JVM is running on."
 ::= { j2eeJVMEEntry 6 }

-----
--
-- J2EE J2EEApplication Managed Objects
--
-----

j2eeAppTable OBJECT-TYPE
SYNTAX      SEQUENCE OF J2eeAppEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "This table lists the J2EE Applications present on a system."
 ::= { j2eeMoGroup 4}

j2eeAppEntry OBJECT-TYPE
SYNTAX      J2eeAppEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "An entry describing a J2EE Application in a J2EE
    Management Domain."
INDEX { j2eeSrvIndex, j2eeAppIndex }
 ::= { j2eeAppTable 1 }

J2eeAppEntry ::= SEQUENCE {
    j2eeAppIndex      Integer32,
    j2eeAppMoName     J2eeMoName,

```

```

j2eeAppMoStateManaged TruthValue,
j2eeAppMoStatProv TruthValue,
j2eeAppMoEventProv TruthValue,
j2eeAppSMState J2eeSMState,
j2eeAppSMStartTime J2eeSMStartTime
}

```

```

j2eeAppIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "The locally arbitrary, but unique identifier associated
    with each J2EE application entry.

```

The value is expected to remain constant at least from one re-initialization of the entity's management system to the next re-initialization."

```
 ::= { j2eeAppEntry 1 }
```

```

j2eeAppMoName OBJECT-TYPE -- expanded verbiage to be filled in later
SYNTAX J2eeMoName
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "The name of the application. This attribute is required because
    the JSR77 model requires servers to be managed objects,
    and all managed objects have names."
 ::= { j2eeAppEntry 2 }

```

```

j2eeAppMoStateManaged OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This flag is True if the application is state-manageable
    as defined by the JSR77 model for all managed objects."
 ::= { j2eeAppEntry 3 }

```

j2eeAppMoStatProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object provides statistics as defined by the JSR77 model for managed objects."

::= { j2eeAppEntry 4 }

j2eeAppMoEventProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object provides events as defined by the JSR77 model for managed objects."

::= { j2eeAppEntry 5 }

j2eeAppSMState OBJECT-TYPE

SYNTAX J2eeSMState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of this object, following state-manageable object semantics."

::= { j2eeAppEntry 6 }

j2eeAppSMStartTime OBJECT-TYPE

SYNTAX J2eeSMStartTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A sequence of octets, following the SNMPv2-TC DateAndTime format, representing the date and time, to the nearest hundredth of a second, at which this object has most recently entered the running(4) state."

::= { j2eeAppEntry 7 }

--

-- J2EE Module Managed Objects

--

j2eeModTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeModEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table lists the modules in all applications deployed in
in a domain."

::= { j2eeMoGroup 5 }

j2eeModEntry OBJECT-TYPE

SYNTAX J2eeModEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry for each module in each application in a J2EE Management
Domain. The entry is indexed by the J2EE application index as well
as a unique ID per module."

INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex }

::= { j2eeModTable 1 }

J2eeModEntry ::= SEQUENCE {

j2eeModIndex Integer32,

j2eeModMoName J2eeMoName,

j2eeModType INTEGER,

j2eeModJVMIIndex Integer32,

j2eeModMoStateManaged TruthValue,

j2eeModMoStatProv TruthValue,

j2eeModMoEventProv TruthValue,

j2eeModSMState J2eeSMState,

j2eeModSMStartTime J2eeSMStartTime

}

j2eeModIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS not-accessible

STATUS current
 DESCRIPTION
 "The locally arbitrary, but unique identifier associated
 with each J2EE module.

The value is expected to remain constant at least from one
 re-initialization of the entity's management system
 to the next re-initialization."

::= { j2eeModEntry 1 }

j2eeModMoName OBJECT-TYPE

SYNTAX J2eeMoName

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The name of the module. This attribute is required because
 the JSR77 model requires modules to be managed objects,
 and all managed objects have names."

::= { j2eeModEntry 2 }

j2eeModType OBJECT-TYPE

SYNTAX INTEGER {
 other(1), ejb(2), web(3), resourceAdapter(4),
 applient(5)
 }

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

The type of this J2EE Module.

"

::= { j2eeModEntry 3 }

-- Revisit: Do we have a 1-1 relationship between MIB and Modules?
 -- If not we have to introduce a j2eeModuleJVMRelationTable
 --

j2eeModJVIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS read-only

STATUS current
 DESCRIPTION
 "The arbitrary, but unique identifier for the JVM on which each J2EE module is running.

The value MUST be able to be used to query the j2eeJVMTTable for details about the JVM."

::= { j2eeModEntry 4 }

j2eeModMoStateManaged OBJECT-TYPE
 SYNTAX TruthValue
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "This flag is True if this object is state-manageable as defined by the JSR77 model for all managed objects."
 ::= { j2eeModEntry 5 }

j2eeModMoStatProv OBJECT-TYPE
 SYNTAX TruthValue
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "This flag is True if this object provides statistics as defined by the JSR77 model for managed objects."
 ::= { j2eeModEntry 6 }

j2eeModMoEventProv OBJECT-TYPE
 SYNTAX TruthValue
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "This flag is True if this object provides events as defined by the JSR77 model for managed objects."
 ::= { j2eeModEntry 7 }

j2eeModSMState OBJECT-TYPE
 SYNTAX J2eeSMState
 MAX-ACCESS read-only

```

STATUS    current
DESCRIPTION
    "The status of this object, following state-manageable object
    semantics."
 ::= { j2eeModEntry 8 }

```

```

j2eeModSMStartTime OBJECT-TYPE
SYNTAX    J2eeSMStartTime
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
    "A sequence of octets, following the SNMPv2-TC DateAndTime format,
    representing the date and time, to the nearest hundredth of a
    second, at which this object has most recently entered the
    running(4) state."
 ::= { j2eeModEntry 9 }

```

```

-----
--
-- J2EE EJB Managed Objects
--
-----

```

```

j2eeBeanTable OBJECT-TYPE
SYNTAX    SEQUENCE OF J2eeBeanEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "This table lists the EJBs in all applications deployed in
    a domain."
 ::= { j2eeMoGroup 6 }

```

```

j2eeBeanEntry OBJECT-TYPE
SYNTAX    J2eeBeanEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "An entry for each bean in each application in a J2EE Management
    Domain. The entry is indexed by the J2EE application index, the
    module index, and a unique ID per bean."

```

```
INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeBeanIndex }
::= { j2eeBeanTable 1 }
```

```
J2eeBeanEntry ::= SEQUENCE {
    j2eeBeanIndex      Integer32,
    j2eeBeanMoName    J2eeMoName,
    j2eeBeanType      Integer32,
    j2eeBeanMoStateManaged TruthValue,
    j2eeBeanMoStatProv TruthValue,
    j2eeBeanMoEventProv TruthValue,
    j2eeBeanSMState   J2eeSMState,
    j2eeBeanSMStartTime J2eeSMStartTime
}
```

```
j2eeBeanIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The locally arbitrary, but unique identifier associated
        with each J2EE Bean entry.
```

The value is expected to remain constant at least from one re-initialization of the entity's management system to the next re-initialization."

```
::= { j2eeBeanEntry 1 }
```

```
j2eeBeanMoName OBJECT-TYPE
    SYNTAX      J2eeMoName
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The name of the bean. This attribute is required because
        the JSR77 model requires servers to be managed objects,
        and all managed objects have names."
```

```
::= { j2eeBeanEntry 2 }
```

```
j2eeBeanType OBJECT-TYPE
    SYNTAX      Integer32 {
```

```

        other(1), entity(2), messageDriven(3),
        stateful(4), stateless(5)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    The type of Enterprise Java Bean this object represents.
"
 ::= { j2eeBeanEntry 3 }

```

j2eeBeanMoStateManaged OBJECT-TYPE

```

SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This flag is True if this object is state-manageable as defined
    by the JSR77 model for all managed objects."
 ::= { j2eeBeanEntry 4 }

```

j2eeBeanMoStatProv OBJECT-TYPE

```

SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This flag is True if this object provides statistics as
    defined by the JSR77 model for managed objects."
 ::= { j2eeBeanEntry 5 }

```

j2eeBeanMoEventProv OBJECT-TYPE

```

SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This flag is True if this object provides events as
    defined by the JSR77 model for managed objects."
 ::= { j2eeBeanEntry 8 }

```

j2eeBeanSMState OBJECT-TYPE

```

SYNTAX J2eeSMState
MAX-ACCESS read-only

```

STATUS current
 DESCRIPTION
 "The status of this object, following state-manageable object semantics."
 ::= { j2eeBeanEntry 9 }

j2eeBeanSMStartTime OBJECT-TYPE

SYNTAX J2eeSMStartTime
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "A sequence of octets, following the SNMPv2-TC DateAndTime format, representing the date and time, to the nearest hundredth of a second, at which this object has most recently entered the running(4) state."
 ::= { j2eeBeanEntry 10 }

 --
 -- J2EE Servlet Managed Objects
 --

j2eeSletTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeSletEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table lists the servlets in all applications deployed in a domain."
 ::= { j2eeMoGroup 7 }

j2eeSletEntry OBJECT-TYPE

SYNTAX J2eeSletEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "An entry for each servlet in each application in a J2EE Management Domain. The entry is indexed by the J2EE application index, the

module index, and a unique ID per servlet."
 INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeSletIndex }
 ::= { j2eeSletTable 1 }

```
J2eeSletEntry ::= SEQUENCE {
  j2eeSletIndex      Integer32,
  j2eeSletMoName     J2eeMoName,
  j2eeSletMoStateManaged TruthValue,
  j2eeSletMoStatProv TruthValue,
  j2eeSletMoEventProv TruthValue,
  j2eeSletSMState    J2eeSMState,
  j2eeSletSMStartTime J2eeSMStartTime
}
```

j2eeSletIndex OBJECT-TYPE
 SYNTAX Integer32 (1..2147483647)
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "The locally arbitrary, but unique identifier associated
 with each J2EE servlet entry.

The value is expected to remain constant at least from one
 re-initialization of the entity's management system
 to the next re-initialization."

```
::= { j2eeSletEntry 1 }
```

j2eeSletMoName OBJECT-TYPE
 SYNTAX J2eeMoName
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "The name of the servlet. This attribute is required because
 the JSR77 model requires servlet to be managed objects,
 and all managed objects have names."

```
::= { j2eeSletEntry 2 }
```

j2eeSletMoStateManaged OBJECT-TYPE
 SYNTAX TruthValue

MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object is state-manageable as defined
by the JSR77 model for all managed objects."
::= { j2eeSletEntry 3 }

j2eeSletMoStatProv OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object provides statistics as
defined by the JSR77 model for managed objects."
::= { j2eeSletEntry 4 }

j2eeSletMoEventProv OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object provides events as
defined by the JSR77 model for managed objects."
::= { j2eeSletEntry 5 }

j2eeSletSMState OBJECT-TYPE
SYNTAX J2eeSMState
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The status of this object, following state-manageable object
semantics."
::= { j2eeSletEntry 6 }

j2eeSletSMStartTime OBJECT-TYPE
SYNTAX J2eeSMStartTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A sequence of octets, following the SNMPv2-TC DateAndTime format,

representing the date and time, to the nearest hundredth of a second, at which this object has most recently entered the running(4) state."
 ::= { j2eeSletEntry 7 }

 --
 -- J2EE J2EEApplication Managed Objects
 --

j2eeAdapTable OBJECT-TYPE
 SYNTAX SEQUENCE OF J2eeAdapEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table lists the resource adapters in all applications deployed in a domain."
 ::= { j2eeMoGroup 8 }

j2eeAdapEntry OBJECT-TYPE
 SYNTAX J2eeAdapEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "An entry for each resource adapter in each application in a J2EE Management Domain. The entry is indexed by the J2EE application index, the module index, and a unique ID per adapter."
 INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeAdapIndex }
 ::= { j2eeAdapTable 1 }

J2eeAdapEntry ::= SEQUENCE {
 j2eeAdapIndex Integer32,
 j2eeAdapMoName J2eeMoName,
 j2eeAdapMoStateManaged TruthValue,
 j2eeAdapMoStatProv TruthValue,
 j2eeAdapMoEventProv TruthValue,
 j2eeAdapSMState J2eeSMState,
 j2eeAdapSMStartTime J2eeSMStartTime
 }

j2eeAdapIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The locally arbitrary, but unique identifier associated with each J2EE resource adapter entry.

The value is expected to remain constant at least from one re-initialization of the entity's management system to the next re-initialization."

::= { j2eeAdapEntry 1 }

j2eeAdapMoName OBJECT-TYPE

SYNTAX J2eeMoName

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The name of the resource adapter. This attribute is required because the JSR77 model requires resource adapters to be managed objects, and all managed objects have names."

::= { j2eeAdapEntry 2 }

j2eeAdapMoStateManaged OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object is state-manageable as defined by the JSR77 model for all managed objects."

::= { j2eeAdapEntry 3 }

j2eeAdapMoStatProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object provides statistics as defined by the JSR77 model for managed objects."
 ::= { j2eeAdapEntry 4 }

j2eeAdapMoEventProv OBJECT-TYPE
 SYNTAX TruthValue
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "This flag is True if this object provides events as defined by the JSR77 model for managed objects."
 ::= { j2eeAdapEntry 5 }

j2eeAdapSMState OBJECT-TYPE
 SYNTAX J2eeSMState
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "The status of this object, following state-manageable object semantics."
 ::= { j2eeAdapEntry 6 }

j2eeAdapSMStartTime OBJECT-TYPE
 SYNTAX J2eeSMStartTime
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "A sequence of octets, following the SNMPv2-TC DateAndTime format, representing the date and time, to the nearest hundredth of a second, at which this object has most recently entered the running(4) state."
 ::= { j2eeAdapEntry 7 }

 --
 -- J2EE J2EEResource Managed Objects
 --

j2eeRsrcTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeRsrcEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table lists the resources used by a server to provide J2EE standard services."

::= { j2eeMoGroup 9 }

j2eeRsrcEntry OBJECT-TYPE

SYNTAX J2eeRsrcEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry for each resource used in each server in a J2EE Management Domain. These resources are used by the J2EE core sever to provide the J2EE standard services required by the J2EE platform architecture.

The entry is indexed by the J2EE server and a unique ID per resource."

INDEX { j2eeSrvIndex, j2eeRsrcIndex }

::= { j2eeRsrcTable 1 }

J2eeRsrcEntry ::= SEQUENCE {

j2eeRsrcIndex Integer32,

j2eeRsrcMoName J2eeMoName,

j2eeRsrcType Integer32,

j2eeRsrcMoStateManaged TruthValue,

j2eeRsrcMoStatProv TruthValue,

j2eeRsrcMoEventProv TruthValue,

j2eeRsrcSMState J2eeSMState,

j2eeRsrcSMStartTime J2eeSMStartTime

}

j2eeRsrcIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The locally arbitrary, but unique identifier associated with each J2EE resource.

The value is expected to remain constant at least from one re-initialization of the entity's management system to the next re-initialization."

::= { j2eeRsrcEntry 1 }

j2eeRsrcMoName OBJECT-TYPE

SYNTAX J2eeMoName

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The name of the resource. This attribute is required because the JSR77 model requires resources to be managed objects, and all managed objects have names."

::= { j2eeRsrcEntry 2 }

j2eeRsrcType OBJECT-TYPE

SYNTAX Integer32 {

other(1), javamail(2), jdbc(3), jms(4),

jndi(5), jta(6), rmiiop(7), url(8) , jca(9)

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The type of resource:

javamail(2) JavaMail(tm)

jdbc(3) Java Database Connectivity

jms(4) Java Message Service

jndi(5) Java Naming and Directory Interface(tm)

jta(6) Java Transaction API

rmiiop(7) Remote Method Invocation over Internet
Inter-ORB Protocol

url(8) URL

jca(9) J2EE Connector Architecture"

::= { j2eeRsrcEntry 3 }

j2eeRsrcMoStateManaged OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object is state-manageable as defined
by the JSR77 model for all managed objects."
::= { j2eeRsrcEntry 4 }

j2eeRsrcMoStatProv OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object provides statistics as
defined by the JSR77 model for managed objects."
::= { j2eeRsrcEntry 5 }

j2eeRsrcMoEventProv OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object provides events as
defined by the JSR77 model for managed objects."
::= { j2eeRsrcEntry 6 }

j2eeRsrcSMState OBJECT-TYPE
SYNTAX J2eeSMState
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The status of this object, following state-manageable object
semantics."
::= { j2eeRsrcEntry 7 }

j2eeRsrcSMStartTime OBJECT-TYPE
SYNTAX J2eeSMStartTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A sequence of octets, following the SNMPv2-TC DateAndTime format,

representing the date and time, to the nearest hundredth of a second, at which this object has most recently entered the running(4) state."
 ::= { j2eeRsrcEntry 8 }

 --
 -- J2EE JCA Objects
 --

j2eeJCATable OBJECT-TYPE
 SYNTAX SEQUENCE OF J2eeJCAEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table lists the JCA Managed Connection Factories and
 Connection Factories used by a server to provide J2EE
 standard services."
 ::= { j2eeMoGroup 10 }

j2eeJCAEntry OBJECT-TYPE
 SYNTAX J2eeJCAEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "An entry for each ConnectionFactory and
 ManagedConnectionFactory pair used in each server
 in a J2EE Management Domain. The entry is
 indexed by the J2EE server and a unique ID per pair."
 INDEX { j2eeSrvIndex, j2eeJCAIndex }
 ::= { j2eeJCATable 1 }

-- Revisit: This implies that the state/statistics/etc... are
 -- in fact applicable to the pair, and not to each
 -- individual object in the pair. Is this correct?
 --
 -- I am also assuming here that the "most significant" object
 -- in the pair (the handle?) is the ConnectionFactory - as the UML


```
-- model shows that the ConnectionFactory holds a pointer to the
-- ManagedConnectionFactory
```

```
J2eeJCAEntry ::= SEQUENCE {
    j2eeJCAIndex          Integer32,
    j2eeJCAMoConnectionFactoryName J2eeMoName,
    j2eeJCAMoManagedConnectionFactoryName J2eeMoName,
    j2eeJCAMoStateManaged TruthValue,
    j2eeJCAMoStatProv      TruthValue,
    j2eeJCAMoEventProv     TruthValue,
    j2eeJCASMState         J2eeSMState,
    j2eeJCASMStartTime     J2eeSMStartTime
}
```

```
j2eeJCAIndex OBJECT-TYPE
    SYNTAX Integer32 (1..2147483647)
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "The locally arbitrary, but unique identifier associated
        with each J2EE resource. The value is expected to remain
        constant at least from one re-initialization of the
        entity's management system to the next re-initialization."
    ::= { j2eeJCAEntry 1 }
```

```
j2eeJCAMoConnectionFactoryName OBJECT-TYPE
    SYNTAX J2eeMoName
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The name of the JCA ConnectionFactory. This attribute is
        required because the JSR77 model requires resources to
        be managed objects, and all managed objects have names."
    ::= { j2eeJCAEntry 2 }
```

```
j2eeJCAMoManagedConnectionFactoryName OBJECT-TYPE
    SYNTAX J2eeMoName
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
```

"The name of the JCA ManagedConnectionFactory. This attribute is required because the JSR77 model requires resources to be managed objects, and all managed objects have names."

::= { j2eeJCAEntry 3 }

j2eeJCAMoStateManaged OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object is state-manageable as defined by the JSR77 model for all managed objects."

::= { j2eeJCAEntry 4 }

j2eeJCAMoStatProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object provides statistics as defined by the JSR77 model for managed objects."

::= { j2eeJCAEntry 5 }

j2eeJCAMoEventProv OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This flag is True if this object provides events as defined by the JSR77 model for managed objects."

::= { j2eeJCAEntry 6 }

j2eeJCASMState OBJECT-TYPE

SYNTAX J2eeSMState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of this object, following state-manageable object semantics."

```
::= { j2eeJCAEntry 7 }
```

```
j2eeJCASMSStartTime OBJECT-TYPE
```

```
SYNTAX J2eeSMStartTime
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

"A sequence of octets, following the SNMPv2-TC DateAndTime format, representing the date and time, to the nearest hundredth of a second, at which this object has most recently entered the running(4) state."

```
::= { j2eeJCAEntry 8 }
```

```
-----
--
-- J2EE JDBC Managed Objects
--
-----
```

```
j2eeJDBCTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF J2eeJDBCEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

"This table lists the JDBC Drivers and Data Sources used by a server to provide J2EE standard services."

```
::= { j2eeMoGroup 11 }
```

```
j2eeJDBCEntry OBJECT-TYPE
```

```
SYNTAX J2eeJDBCEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

"An entry for each JDBC Driver and Data Source pair used in each server in a J2EE Management Domain. The entry is indexed by the J2EE server and a unique ID per pair."

```
INDEX { j2eeSrvIndex, j2eeJDBCIndex }
```

```
::= { j2eeJDBCTable 1 }
```

```
-- Revisit: This implies that the state/statistics/etc... are
--   in fact applicable to the pair, and not to each
--   individual object in the pair. Is this correct?
--
-- I am also assuming here that the "most significant" object
-- in the pair (the handle?) is the DataSource - as the UML
-- model shows that the DataSource holds a pointer to the Driver
```

```
J2eeJDBCEntry ::= SEQUENCE {
    j2eeJDBCIndex      Integer32,
    j2eeJDBCMoSourceName  J2eeMoName,
    j2eeJDBCMoDriverName  J2eeMoName,
    j2eeJDBCMoStateManaged  TruthValue,
    j2eeJDBCMoStatProv      TruthValue,
    j2eeJDBCMoEventProv     TruthValue,
    j2eeJDBCSMState        J2eeSMState,
    j2eeJDBCSMStartTime    J2eeSMStartTime
}
```

```
j2eeJDBCIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The locally arbitrary, but unique identifier associated
        with each J2EE JDBC object.
```

The value is expected to remain constant at least from one re-initialization of the entity's management system to the next re-initialization."

```
::= { j2eeJDBCEntry 1 }
```

```
j2eeJDBCMoSourceName OBJECT-TYPE
    SYNTAX      J2eeMoName
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The name of the JDBC Data Source. This attribute is
```

required because the JSR77 model requires resources to be managed objects, and all managed objects have names."
::= { j2eeJDBCEntry 2 }

j2eeJDBCMoDriverName OBJECT-TYPE
SYNTAX J2eeMoName
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The name of the JDBC Driver. This attribute is required because the JSR77 model requires resources to be managed objects, and all managed objects have names."
::= { j2eeJDBCEntry 3 }

j2eeJDBCMoStateManaged OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object is state-manageable as defined by the JSR77 model for all managed objects."
::= { j2eeJDBCEntry 4 }

j2eeJDBCMoStatProv OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object provides statistics as defined by the JSR77 model for managed objects."
::= { j2eeJDBCEntry 5 }

j2eeJDBCMoEventProv OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This flag is True if this object provides events as defined by the JSR77 model for managed objects."
::= { j2eeJDBCEntry 6 }

```

j2eeJDBCSMState    OBJECT-TYPE
SYNTAX    J2eeSMSState
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
    "The status of this object, following state-manageable object
    semantics."
 ::= { j2eeJDBCEntry 7 }

j2eeJDBCSMStartTime OBJECT-TYPE
SYNTAX    J2eeSMStartTime
MAX-ACCESS read-only
STATUS    current
DESCRIPTION
    "A sequence of octets, following the SNMPv2-TC DateAndTime format,
    representing the date and time, to the nearest hundredth of a
    second, at which this object has most recently entered the
    running(4) state."
 ::= { j2eeJDBCEntry 8 }

-----
--
-- J2EE Servlet Statistics
--
-----

-- Revisit: OBJECT-IDENTITY might be preferred here
--
j2eeServletStatGroup    OBJECT IDENTIFIER ::= { j2eeStatistics 1 }

j2eeServletStatTable OBJECT-TYPE
SYNTAX    SEQUENCE OF J2eeServletStatEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "This table contains a row for each servlet for which
    statistics are provided."

```

```
::= { j2eeServletStatGroup 1 }
```

j2eeServletStatEntry OBJECT-TYPE

SYNTAX J2eeServletStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry providing statistics for a given servlet.

For each entry in this table there MUST be a corresponding entry in the j2eeSletTable.

The servlet should have its j2eeSletMoStatProv flag set to true.

"

INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeSletIndex }

```
::= { j2eeServletStatTable 1 }
```

J2eeServletStatEntry ::= SEQUENCE {

j2eeSletStatServiceCount Counter32,

j2eeSletStatServiceMaxTime Gauge32,

j2eeSletStatServiceMinTime Gauge32,

j2eeSletStatServiceTotal Counter32

}

j2eeSletStatServiceCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Returns the number of times the Servlet getService operation was invoked.

"

```
::= { j2eeServletStatEntry 1 }
```

j2eeSletStatServiceMaxTime OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Returns the maximum amount of time taken to complete one invocation of the Servlet getService operation.

"

```
::= { j2eeServletStatEntry 2 }
```

```
j2eeSletStatServiceMinTime OBJECT-TYPE
```

```
SYNTAX Gauge32
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION "
```

```
    Returns the minimum amount of time taken to complete
    one invocation of the Servlet getService operation.
```

```
"
```

```
::= { j2eeServletStatEntry 3 }
```

```
j2eeSletStatServiceTotal OBJECT-TYPE
```

```
SYNTAX Counter32
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION "
```

```
    Returns the sum total of time taken to complete
    every invocation of the Servlet getService operation.
    Dividing j2eeSletStatServiceTotal by
    j2eeSletStatServiceCount will give you the average
    execution time for this operation.
```

```
"
```

```
::= { j2eeServletStatEntry 4 }
```

```
-----
--
-- J2EE EJB Statistics
--
-----
```

```
-- Revisit: OBJECT-IDENTITY might be preferred here
```

```
--
```

```
j2eeEjbStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 2 }
```

```
j2eeEjbEntityStatTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF J2eeEjbEntityStatEntry
```

```
MAX-ACCESS not-accessible
```


STATUS current
 DESCRIPTION
 "This table extends the J2EE Bean table with performance data framework statistics objects.
 This table contains a row for each Entity EJB for which statistics are provided.
 "
 ::= { j2eeEjbStatGroup 1 }

j2eeEjbEntityStatEntry OBJECT-TYPE
 SYNTAX J2eeEjbEntityStatEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "The performance-extended version of the J2EE Entity EJB table. This entry provides specific performance statistics for statistics-enabled EJBs.
 For each row in this table there MUST be a corresponding row in the j2eeBeanTable.
 The corresponding Bean should have its j2eeBeanMoStatProv flag set to true.
 "
 INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeBeanIndex }
 ::= { j2eeEjbEntityStatTable 1 }

J2eeEjbEntityStatEntry ::= SEQUENCE {
 j2eeEjbEntityStatCreateCount Counter32,
 j2eeEjbEntityStatRemoveCount Counter32,
 j2eeEjbEntityStatReadyCount Gauge32,
 j2eeEjbEntityStatPooledCount Gauge32
 }

j2eeEjbEntityStatCreateCount OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION "
 Corresponds to EjbEntityStats.getCreateCount().
 Returns the number of times the beans create method was called.

```

"
 ::= { j2eeEjbEntityStatEntry 1 }

j2eeEjbEntityStatRemoveCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Corresponds to EjbEntityStats.getRemoveCount().
    Returns the number of times the beans remove method
    was called.
"
 ::= { j2eeEjbEntityStatEntry 2 }

j2eeEjbEntityStatReadyCount OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    This number corresponds to EntityBeanStats.getReadyCount()
    and returns the number of bean instances in the ready state.
"
 ::= { j2eeEjbEntityStatEntry 3 }

j2eeEjbEntityStatPooledCount OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    This number corresponds to EntityBeanStats.getPoolCount() and
    returns the number of bean instances in the pooled state.
"
 ::= { j2eeEjbEntityStatEntry 4 }

j2eeEjbStatelessStatTable OBJECT-TYPE
SYNTAX SEQUENCE OF J2eeEjbStatelessStatEntry
MAX-ACCESS not-accessible
STATUS current

```

DESCRIPTION

"This table extends the J2EE Bean table with performance data framework statistics objects.

This table contains a row for each Stateless Session EJB for which statistics are provided.

"

::= { j2eeEjbStatGroup 2 }

j2eeEjbStatelessStatEntry OBJECT-TYPE

SYNTAX J2eeEjbStatelessStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The performance-extended version of the J2EE EJB table. This entry provides specific performance statistics for statistics-enabled Stateless Session EJBs.

For each row in this table there MUST be a corresponding row in the j2eeBeanTable.

The corresponding Bean should have its j2eeBeanMoStatProv flag set to true.

"

INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeBeanIndex }

::= { j2eeEjbStatelessStatTable 1 }

J2eeEjbStatelessStatEntry ::= SEQUENCE {

j2eeEjbStatelessStatCreateCount Counter32,

j2eeEjbStatelessStatRemoveCount Counter32,

j2eeEjbStatelessStatReadyCount Gauge32

}

j2eeEjbStatelessStatCreateCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION "

Corresponds to EjbStatelessStats.getCreateCount().

Returns the number of times the beans create method was called.

"

::= { j2eeEjbStatelessStatEntry 1 }

j2eeEjbStatelessStatRemoveCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Corresponds to EjbStatelessStats.getRemoveCount().
Returns the number of times the beans remove method
was called.

"

::= { j2eeEjbStatelessStatEntry 2 }

j2eeEjbStatelessStatReadyCount OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

This number corresponds to
SessionBeanStats.getMethodReadyCount(),
and it returns the number of beans that are in the
in the method-ready state.

"

::= { j2eeEjbStatelessStatEntry 3 }

j2eeEjbStatefulStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeEjbStatefulStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table extends the J2EE Bean table with performance
data framework statistics objects.

This table contains a row for each Stateful Session EJB
for which statistics are provided. Some statistics will
have different interpretations depending on the type of
bean being for which they are provided.

"

::= { j2eeEjbStatGroup 3 }

j2eeEjbStatefulStatEntry OBJECT-TYPE
 SYNTAX J2eeEjbStatefulStatEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "The performance-extended version of the J2EE Stateful Session
 EJB table. This entry provides specific performance statistics
 for statistics-enabled EJBs.
 For each row in this table there MUST be a corresponding row
 in the j2eeBeanTable.
 The corresponding Bean should have its j2eeBeanMoStatProv
 flag set to true.
 "
 INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeBeanIndex }
 ::= { j2eeEjbStatefulStatTable 1 }

J2eeEjbStatefulStatEntry ::= SEQUENCE {
 j2eeEjbStatefulStatCreateCount Counter32,
 j2eeEjbStatefulStatRemoveCount Counter32,
 j2eeEjbStatefulStatReadyCount Gauge32,
 j2eeEjbStatefulStatPassiveCount Gauge32
 }

j2eeEjbStatefulStatCreateCount OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION "
 Corresponds to EjbStatefulStats.getCreateCount().
 Returns the number of times the beans create method
 was called.
 "
 ::= { j2eeEjbStatefulStatEntry 1 }

j2eeEjbStatefulStatRemoveCount OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION "
 Corresponds to EjbStatefulStats.getRemoveCount().

Returns the number of times the beans remove method was called.

"

::= { j2eeEjbStatefulStatEntry 2 }

j2eeEjbStatefulStatReadyCount OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

This number corresponds to
SessionBeanStats.getMethodReadyCount(),
and it returns the number of beans that are in the
in the method-ready state.

"

::= { j2eeEjbStatefulStatEntry 3 }

j2eeEjbStatefulStatPassiveCount OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

This number corresponds to
StatefulSessionBeanStats.getPassiveCount(),
and it returns the number of beans that are in the
passivated state.

"

::= { j2eeEjbStatefulStatEntry 4 }

j2eeEjbMessageDrivenStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeEjbMessageDrivenStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table extends the J2EE Bean table with performance data framework statistics objects.

This table contains a row for each Message Driven EJB for which statistics are provided. Some statistics will have different interpretations depending on the type of bean being for which they are provided.

```
"
 ::= { j2eeEjbStatGroup 4 }
```

j2eeEjbMessageDrivenStatEntry OBJECT-TYPE

SYNTAX J2eeEjbMessageDrivenStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The performance-extended version of the J2EE Message Driven EJB table. This entry provides specific performance statistics for statistics-enabled EJBs.

For each row in this table there MUST be a corresponding row in the j2eeBeanTable.

The corresponding Bean should have its j2eeBeanMoStatProv flag set to true.

```
"
INDEX { j2eeSrvIndex, j2eeAppIndex, j2eeModIndex, j2eeBeanIndex }
 ::= { j2eeEjbMessageDrivenStatTable 1 }
```

```
J2eeEjbMessageDrivenStatEntry ::= SEQUENCE {
    j2eeEjbMessageDrivenStatCreateCount Counter32,
    j2eeEjbMessageDrivenStatRemoveCount Counter32,
    j2eeEjbMessageDrivenStatMessageCount Counter32
}
```

j2eeEjbMessageDrivenStatCreateCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION "

Corresponds to EjbMessageDrivenStats.getCreateCount().

Returns the number of times the beans create method was called.

```
"
 ::= { j2eeEjbMessageDrivenStatEntry 1 }
```

j2eeEjbMessageDrivenStatRemoveCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

```

STATUS    current
DESCRIPTION "
    Corresponds to EjbMessageDrivenStats.getRemoveCount().
    Returns the number of times the beans remove method
    was called.
"
 ::= { j2eeEjbMessageDrivenStatEntry 2 }

```

```

j2eeEjbMessageDrivenStatMessageCount OBJECT-TYPE
SYNTAX    Counter32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "
    Returns the number of messages received.
"
 ::= { j2eeEjbMessageDrivenStatEntry 3 }

```

```

-----
--
-- J2EE JavaMail Statistics
--
-----

```

```

-- Revisit: OBJECT-IDENTITY might be preferred here
--
j2eeJavaMailStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 3 }

```

```

j2eeJavaMailStatTable OBJECT-TYPE
SYNTAX    SEQUENCE OF J2eeJavaMailStatEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "This table extends the J2EE Resource table with performance
    data framework statistics objects for JavaMail resources.
    This table contains a row for each JavaMail resource for which
    statistics are provided.
"
 ::= { j2eeJavaMailStatGroup 1 }

```



```

j2eeJavaMailStatEntry OBJECT-TYPE
    SYNTAX      J2eeJavaMailStatEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The performance-extended, JavaMail-specific version of the J2EE
        Resource table.
        This entry extends provides JavaMail-specific performance
        statistics for statistics-enabled JavaMail resources.
        Note that this extension only applies to rows in the J2EE
        resource table where j2eeRsrcType is javamail(2);
        attempts to access the corresponding instance of any of
        these objects when j2eeRsrcType is not javamail(2) may
        result in either noSuchName (SNMPv1) or noSuchInstance (SNMPv2)
        being returned by the agent.
        For each row in this table there MUST be a corresponding
        row of type javamail(2) in the j2eeRsrcTable.
        The resource should have its j2eeRsrcMoStatProv flag set to true.
        "
    INDEX { j2eeSrvIndex, j2eeRsrcIndex }
    ::= { j2eeJavaMailStatTable 1 }

J2eeJavaMailStatEntry ::= SEQUENCE {
    j2eeJavaMailStatSentCount  Counter32
}

j2eeJavaMailStatSentCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "
        Correspond to JavaMailStats.getSentMailCount().
        Returns the number of mail messages sent.
        "
    ::= { j2eeJavaMailStatEntry 1 }

-----
--
-- J2EE JTA Resource Statistics
--

```

```
-- Revisit: OBJECT-IDENTITY might be preferred here
--
j2eeJtaStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 4 }
```

```
j2eeJtaStatTable OBJECT-TYPE
SYNTAX SEQUENCE OF J2eeJtaStatEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "This table extends the Resource table with performance
    data framework statistics objects for JTA resources."
 ::= { j2eeJtaStatGroup 1 }
```

```
j2eeJtaStatEntry OBJECT-TYPE
SYNTAX J2eeJtaStatEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "The performance-extended, JTA-specific version of the J2EE
    Resource table. This entry provides JTA-specific performance
    statistics for statistics-enabled JTA Resources.
    Note that this extension only applies to rows in the J2EE
    resource table where j2eeRsrcType is jta(6);
    attempts to access the corresponding instance of any of
    these objects when j2eeRsrcType is not jta(6) may
    result in either noSuchName (SNMPv1) or noSuchInstance (SNMPv2)
    being returned by the agent.
    For each row in this table there MUST be a corresponding
    row of type jta(6) in the j2eeRsrcTable.
    The resource should have its j2eeRsrcMoStatProv flag set to true.
    "
INDEX { j2eeSrvIndex, j2eeRsrcIndex }
 ::= { j2eeJtaStatTable 1 }
```

```
J2eeJtaStatEntry ::= SEQUENCE {
    j2eeJtaStatActiveCount Gauge32,
    j2eeJtaStatCommittedCount Counter32,
    j2eeJtaStatRolledBackCount Counter32
```

```
}

```

```
j2eeJtaStatActiveCount OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JTAStats.getActiveCount().
    Returns the number of active transactions.
    "
 ::= { j2eeJtaStatEntry 1 }
```

```
j2eeJtaStatCommittedCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JTAStats.getCommittedCount().
    Returns the number of comitted transactions.
    "
 ::= { j2eeJtaStatEntry 2 }
```

```
j2eeJtaStatRolledBackCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JTAStats.getRolledbackCount().
    Returns the number of rolled-back transactions.
    "
 ::= { j2eeJtaStatEntry 3 }
```

```
-----
--
-- J2EE JCA Resource Statistics
--
-----
```

```
j2eeJcaStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 5 }
```

j2eeJcaConnStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeJcaConnStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table provides statistics of non-pooled connections associated with the referencing JCA resource, connection factory.

"

::= { j2eeJcaStatGroup 1 }

j2eeJcaConnStatEntry OBJECT-TYPE

SYNTAX J2eeJcaConnStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This entry provides the non-pooled connections associated with JCA connection factories. It extends the JCA Connection Factory table. The resource should have its j2eeJCAMoStatProv flag set to true.

"

INDEX { j2eeSrvIndex, j2eeJCAIndex, j2eeJcaConnStatIndex }

::= { j2eeJcaConnStatTable 1 }

J2eeJcaConnStatEntry ::= SEQUENCE {

j2eeJcaConnStatIndex Integer32,

j2eeJcaConnStatWaitTime TimeTicks,

j2eeJcaConnStatUseTime TimeTicks

}

j2eeJcaConnStatIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The locally arbitrary, but unique identifier associated with each {j2eeServer, j2eeJCAFactory, j2eeJCAConnection} entry

"

```
::= { j2eeJcaConnStatEntry 1 }
```

j2eeJcaConnStatWaitTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Correspond to JcaConnectionStats.getWaitTime().

Returns time spent waiting for a connection to be available.

"

```
::= { j2eeJcaConnStatEntry 2 }
```

j2eeJcaConnStatUseTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Correspond to JcaConnectionStats.getUseTime().

Returns the time spent using a connection.

"

```
::= { j2eeJcaConnStatEntry 3 }
```

j2eeJcaConnPoolStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeJcaConnPoolStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table provides statistics of connection pools associated with the referencing JCA resource, connection factory.

"

```
::= { j2eeJcaStatGroup 2 }
```

j2eeJcaConnPoolStatEntry OBJECT-TYPE

SYNTAX J2eeJcaConnPoolStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This entry provides the connection pools associated with JCA connection factories. It extends the JCA Connection Factory table. The resource should have its j2eeJCAMoStatProv flag set to true.

"

```
INDEX { j2eeSrvIndex, j2eeJCAIndex, j2eeJcaConnPoolStatIndex }
 ::= { j2eeJcaConnPoolStatTable 1 }
```

```
J2eeJcaConnPoolStatEntry ::= SEQUENCE {
    j2eeJcaConnPoolStatIndex      Integer32,
    j2eeJcaConnPoolStatWaitTime   TimeTicks,
    j2eeJcaConnPoolStatUseTime    TimeTicks,
    j2eeJcaConnPoolStatCreateCount Counter32,
    j2eeJcaConnPoolStatCloseCount Counter32,
    j2eeJcaConnPoolStatPoolSize   Gauge32,
    j2eeJcaConnPoolStatFreePoolSize Gauge32,
    j2eeJcaConnPoolStatWaitingThreadCount Gauge32
}
```

```
j2eeJcaConnPoolStatIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The locally arbitrary, but unique identifier associated
        with each {j2eeServer, j2eeJCAFactory, j2eeJCAConnectionPool}
        entry
        "
    ::= { j2eeJcaConnPoolStatEntry 1 }
```

```
j2eeJcaConnPoolStatWaitTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "
        Correspond to JcaConnectionPoolStats.getWaitTime().
        Returns time spent waiting for a connection to be
        available.
        "
    ::= { j2eeJcaConnPoolStatEntry 2 }
```

```

j2eeJcaConnPoolStatUseTime OBJECT-TYPE
SYNTAX    TimeTicks
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "
    Correspond to JcaConnectionPoolStats.getUseTime().
    Returns the time spent using a connection.
    "
 ::= { j2eeJcaConnPoolStatEntry 3 }

j2eeJcaConnPoolStatCreateCount OBJECT-TYPE
SYNTAX    Counter32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "
    Correspond to JcaConnectionPoolStats.getCreateCount().
    Returns the number of connections created
    "
 ::= { j2eeJcaConnPoolStatEntry 4 }

j2eeJcaConnPoolStatCloseCount OBJECT-TYPE
SYNTAX    Counter32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "
    Correspond to JcaConnectionPoolStats.getCloseCount().
    Returns the number of connections closed
    "
 ::= { j2eeJcaConnPoolStatEntry 5 }

j2eeJcaConnPoolStatPoolSize OBJECT-TYPE
SYNTAX    Gauge32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "
    Correspond to JcaConnectionPoolStats.getPoolSize().
    Returns the size of the connection pool.
    "
 ::= { j2eeJcaConnPoolStatEntry 6 }

j2eeJcaConnPoolStatFreePoolSize OBJECT-TYPE
SYNTAX    Gauge32
MAX-ACCESS read-only

```

```

STATUS    current
DESCRIPTION "
    Correspond to JcaConnectionPoolStats.getFreePoolSize().
    Returns the number of free connections in the pool.
"
 ::= { j2eeJcaConnPoolStatEntry 7 }

j2eeJcaConnPoolStatWaitingThreadCount OBJECT-TYPE
SYNTAX    Gauge32
MAX-ACCESS read-only
STATUS    current
DESCRIPTION "
    Correspond to JcaConnectionPoolStats.getWaitingThreadCount().
    Returns the number of threads waiting for a connection.
"
 ::= { j2eeJcaConnPoolStatEntry 8 }

```

```

-----
--
-- J2EE JDBC Resource Statistics
--
-----

```

```

j2eeJdbcStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 6 }

```

```

j2eeJdbcConnStatTable OBJECT-TYPE
SYNTAX    SEQUENCE OF J2eeJdbcConnStatEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "This table provides statistics of non-pooled
    connections associated with the referencing JDBC resource,
    connection factory.
"
 ::= { j2eeJdbcStatGroup 1 }

```

```

j2eeJdbcConnStatEntry OBJECT-TYPE
SYNTAX    J2eeJdbcConnStatEntry
MAX-ACCESS not-accessible
STATUS    current

```


DESCRIPTION

"This entry provides the non-pooled connections associated with JDBC connection factories. It extends the JDBC Connection Factory table. The resource should have its j2eeJDBCMoStatProv flag set to true.

INDEX { j2eeSrvIndex, j2eeJDBCIndex, j2eeJdbcConnStatIndex }
 ::= { j2eeJdbcConnStatTable 1 }

J2eeJdbcConnStatEntry ::= SEQUENCE {
 j2eeJdbcConnStatIndex Integer32,
 j2eeJdbcConnStatWaitTime TimeTicks,
 j2eeJdbcConnStatUseTime TimeTicks
 }

j2eeJdbcConnStatIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The locally arbitrary, but unique identifier associated with each {j2eeServer, j2eeJDBCFactory, j2eeJDBCConnection} entry

::= { j2eeJdbcConnStatEntry 1 }

j2eeJdbcConnStatWaitTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

 Correspond to JdbcConnectionStats.getWaitTime().
 Returns time spent waiting for a connection to be available.

::= { j2eeJdbcConnStatEntry 2 }

j2eeJdbcConnStatUseTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

```

STATUS    current
DESCRIPTION "
    Correspond to JdbcConnectionStats.getUseTime().
    Returns the time spent using a connection.
"
 ::= { j2eeJdbcConnStatEntry 3 }

```

```

j2eeJdbcConnPoolStatTable OBJECT-TYPE
SYNTAX    SEQUENCE OF J2eeJdbcConnPoolStatEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "This table provides statistics of connection pools
    associated with the referencing JDBC resource,
    connection factory.
"
 ::= { j2eeJdbcStatGroup 2 }

```

```

j2eeJdbcConnPoolStatEntry OBJECT-TYPE
SYNTAX    J2eeJdbcConnPoolStatEntry
MAX-ACCESS not-accessible
STATUS    current
DESCRIPTION
    "This entry provides the connection pools associated
    with JDBC connection factories. It extends the JDBC Connection
    Factory table. The resource should have its j2eeJDBCMoStatProv
    flag set to true.
"
INDEX { j2eeSrvIndex, j2eeJDBCIndex, j2eeJdbcConnPoolStatIndex }
 ::= { j2eeJdbcConnPoolStatTable 1 }

```

```

J2eeJdbcConnPoolStatEntry ::= SEQUENCE {
    j2eeJdbcConnPoolStatIndex      Integer32,
    j2eeJdbcConnPoolStatWaitTime   TimeTicks,
    j2eeJdbcConnPoolStatUseTime     TimeTicks,
    j2eeJdbcConnPoolStatCreateCount Counter32,
    j2eeJdbcConnPoolStatCloseCount Counter32,
    j2eeJdbcConnPoolStatPoolSize   Gauge32,
    j2eeJdbcConnPoolStatFreePoolSize Gauge32,
    j2eeJdbcConnPoolStatWaitingThreadCount Gauge32
}

```

}

j2eeJdbcConnPoolStatIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The locally arbitrary, but unique identifier associated
with each {j2eeServer, j2eeJDBCFactory, j2eeJDBCConnectionPool}
entry
"

::= { j2eeJdbcConnPoolStatEntry 1 }

j2eeJdbcConnPoolStatWaitTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Correspond to JdbcConnectionPoolStats.getWaitTime().
Returns time spent waiting for a connection to be
available.
"

::= { j2eeJdbcConnPoolStatEntry 2 }

j2eeJdbcConnPoolStatUseTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Correspond to JdbcConnectionPoolStats.getUseTime().
Returns the time spent using a connection.
"

::= { j2eeJdbcConnPoolStatEntry 3 }

j2eeJdbcConnPoolStatCreateCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Correspond to JdbcConnectionPoolStats.getCreateCount().
Returns the number of connections created

```

"
 ::= { j2eeJdbcConnPoolStatEntry 4 }

j2eeJdbcConnPoolStatCloseCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JdbcConnectionPoolStats.getCloseCount().
    Returns the number of connections closed
"
 ::= { j2eeJdbcConnPoolStatEntry 5 }

j2eeJdbcConnPoolStatPoolSize OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JdbcConnectionPoolStats.getPoolSize().
    Returns the size of the connection pool.
"
 ::= { j2eeJdbcConnPoolStatEntry 6 }

j2eeJdbcConnPoolStatFreePoolSize OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JdbcConnectionPoolStats.getFreePoolSize().
    Returns the number of free connections in the pool.
"
 ::= { j2eeJdbcConnPoolStatEntry 7 }

j2eeJdbcConnPoolStatWaitingThreadCount OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JdbcConnectionPoolStats.getWaitingThreadCount().
    Returns the number of threads waiting for a connection.
"
 ::= { j2eeJdbcConnPoolStatEntry 8 }

```

```

--
-- J2EE JMS Resource Statistics
--
-----

j2eeJmsStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 7 }

j2eeJmsConnSessionStatTable OBJECT-TYPE
    SYNTAX SEQUENCE OF J2eeJmsConnSessionStatEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "This table extends the Resource table with performance
        data framework statistics objects for JMS connection session.
        "
    ::= { j2eeJmsStatGroup 1 }

j2eeJmsConnSessionStatEntry OBJECT-TYPE
    SYNTAX J2eeJmsConnSessionStatEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "The performance-extended, JMS-specific version of the J2EE
        Resource table. This entry provides JMS-specific performance
        statistics for statistics-enabled JMS connection session.
        Note that this extension only applies to rows in the J2EE
        resource table where j2eeRsrcType is jms(4);
        attempts to access the corresponding instance of any of
        these objects when j2eeRsrcType is not jms(4) may
        result in either noSuchName (SNMPv1) or noSuchInstance (SNMPv2)
        being returned by the agent.
        For each row in this table there MUST be a corresponding
        row of type jms(4) in the j2eeRsrcTable.
        The resource should have its j2eeRsrcMoStatProv flag set to true.
        "
    INDEX { j2eeSrvIndex, j2eeRsrcIndex, j2eeJmsConnIndex, j2eeJmsSessionIndex }
    ::= { j2eeJmsConnSessionStatTable 1 }

J2eeJmsConnSessionStatEntry ::= SEQUENCE {
    j2eeJmsConnIndex Integer32,

```

```

    j2eeJmsSessionIndex          Integer32,
    j2eeJmsConnSessionStatMessageCount Counter32,
    j2eeJmsConnSessionStatPendingMessageCount Counter32,
    j2eeJmsConnSessionStatExpiredMessageCount Counter32,
    j2eeJmsConnSessionStatMessageWaitTime TimeTicks,
    j2eeJmsConnSessionStatDurableSubCount Counter32
}

```

```

j2eeJmsConnIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "The locally arbitrary, but unique identifier associated
    with each {j2eeServer, j2eeJmsResource, j2eeJmsConnection}
    entry
    "
 ::= { j2eeJmsConnSessionStatEntry 1 }

```

```

j2eeJmsSessionIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "The locally arbitrary, but unique identifier associated
    with each {j2eeServer, j2eeJmsResource, j2eeJmsConnection,
    j2eeJmsSession} entry
    "
 ::= { j2eeJmsConnSessionStatEntry 2 }

```

```

j2eeJmsConnSessionStatMessageCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsSessionStats.getMessageCount().
    Returns the number of messages exchanged.
    "
 ::= { j2eeJmsConnSessionStatEntry 3 }

```

```

j2eeJmsConnSessionStatPendingMessageCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current

```

```

DESCRIPTION "
    Correspond to JmsSessionStats.getPendingMessageCount().
    Returns the number of pending messages.
    "
::= { j2eeJmsConnSessionStatEntry 4 }

j2eeJmsConnSessionStatExpiredMessageCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsSessionStats.getExpiredMessageCount().
    Returns the number of expired messages.
    "
::= { j2eeJmsConnSessionStatEntry 5 }

j2eeJmsConnSessionStatMessageWaitTime OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsSessionStats.getMessageWaitTime().
    Returns the time spent by a message before being delivered.
    "
::= { j2eeJmsConnSessionStatEntry 6 }

j2eeJmsConnSessionStatDurableSubCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsSessionStats.getDurableSubscriptionCount().
    Returns the number of durable subscriptions.
    "
::= { j2eeJmsConnSessionStatEntry 7 }

j2eeJmsProducerStatTable OBJECT-TYPE
SYNTAX SEQUENCE OF J2eeJmsProducerStatEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "This table extends the Resource table with performance

```



```

    j2eeJmsSession, j2eeJmsProducer} entry
    "
    ::= { j2eeJmsProducerStatEntry 1 }

j2eeJmsProducerStatMessageCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "
        Correspond to JmsProducerStats.getMessageCount().
        Returns the number of messages sent.
    "
    ::= { j2eeJmsProducerStatEntry 2 }

j2eeJmsProducerStatPendingMessageCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "
        Correspond to JmsProducerStats.getPendingMessageCount().
        Returns the number of pending messages.
    "
    ::= { j2eeJmsProducerStatEntry 3 }

j2eeJmsProducerStatExpiredMessageCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "
        Correspond to JmsProducerStats.getExpiredMessageCount().
        Returns the number of messages that expired
        before delivery.
    "
    ::= { j2eeJmsProducerStatEntry 4 }

j2eeJmsProducerStatMessageWaitTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "
        Correspond to JmsProducerStats.getMessageWaitTime().
        Returns the time spent by a message before being delivered.
    "
    ::= { j2eeJmsProducerStatEntry 5 }

```

j2eeJmsProducerStatDestination OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Correspond to JmsProducerStats.getDestination().

Returns a string that encapsulates the identity of a message destination.

"

::= { j2eeJmsProducerStatEntry 6 }

j2eeJmsConsumerStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeJmsConsumerStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table extends the Resource table with performance data framework statistics objects for JMS consumer.

"

::= { j2eeJmsStatGroup 3 }

j2eeJmsConsumerStatEntry OBJECT-TYPE

SYNTAX J2eeJmsConsumerStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The performance-extended, JMS-specific version of the J2EE Resource table. This entry provides JMS-specific performance statistics for statistics-enabled JMS consumer.

Note that this extension only applies to rows in the J2EE resource table where j2eeRsrcType is jms(4); attempts to access the corresponding instance of any of these objects when j2eeRsrcType is not jms(4) may result in either noSuchName (SNMPv1) or noSuchInstance (SNMPv2) being returned by the agent.

For each row in this table there MUST be a corresponding row of type jms(4) in the j2eeRsrcTable.

The resource should have its j2eeRsrcMoStatProv flag set to true.

"

INDEX { j2eeSrvIndex, j2eeRsrcIndex, j2eeJmsConnIndex,
j2eeJmsSessionIndex, j2eeJmsConsumerIndex }

```
::= { j2eeJmsConsumerStatTable 1 }
```

```
J2eeJmsConsumerStatEntry ::= SEQUENCE {
    j2eeJmsConsumerIndex          Integer32,
    j2eeJmsConsumerStatMessageCount Counter32,
    j2eeJmsConsumerStatPendingMessageCount Counter32,
    j2eeJmsConsumerStatExpiredMessageCount Counter32,
    j2eeJmsConsumerStatMessageWaitTime TimeTicks,
    j2eeJmsConsumerStatOrigin      SnmpAdminString
}
```

```
j2eeJmsConsumerIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "The locally arbitrary, but unique identifier associated
    with each {j2eeServer, j2eeJmsResource, j2eeJmsConnection,
    j2eeJmsSession, j2eeJmsConsumer} entry
    "
::= { j2eeJmsConsumerStatEntry 1 }
```

```
j2eeJmsConsumerStatMessageCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsConsumerStats.getMessageCount().
    Returns the number of messages received.
    "
::= { j2eeJmsConsumerStatEntry 2 }
```

```
j2eeJmsConsumerStatPendingMessageCount OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsConsumerStats.getPendingMessageCount().
    Returns the number of pending messages.
    "
::= { j2eeJmsConsumerStatEntry 3 }
```

```
j2eeJmsConsumerStatExpiredMessageCount OBJECT-TYPE
```

```

SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsConsumerStats.getExpiredMessageCount().
    Returns the number of messages that expired
    before delivery.
    "
 ::= { j2eeJmsConsumerStatEntry 4 }

```

j2eeJmsConsumerStatMessageWaitTime OBJECT-TYPE

```

SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsConsumerStats.getMessageWaitTime().
    Returns the time spent by a message before being delivered.
    "
 ::= { j2eeJmsConsumerStatEntry 5 }

```

j2eeJmsConsumerStatOrigin OBJECT-TYPE

```

SYNTAX SnmpAdminString
MAX-ACCESS read-only
STATUS current
DESCRIPTION "
    Correspond to JmsConsumerStats.getOrigin().
    Returns a string that encapsulates the identity of
    a message origin.
    "
 ::= { j2eeJmsConsumerStatEntry 6 }

```

```

-----
--
-- J2EE JVM Statistics
--
-----

```

```

j2eeJvmStatGroup OBJECT IDENTIFIER ::= { j2eeStatistics 8 }

```

j2eeJvmStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF J2eeJvmStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table contains a row for each JVM for which statistics are provided."

::= { j2eeJvmStatGroup 1 }

j2eeJvmStatEntry OBJECT-TYPE

SYNTAX J2eeJvmStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry providing statistics for a given JVM.

For each entry in this table there MUST be a corresponding entry in the j2eeJVMTTable.

"

INDEX { j2eeSrvIndex, j2eeJVMTIndex }

::= { j2eeJvmStatTable 1 }

J2eeJvmStatEntry ::= SEQUENCE {

j2eeJvmStatUpTime TimeTicks,

j2eeJvmStatHeapSize Gauge32

}

j2eeJvmStatUpTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Returns the amount of time the JVM has been running.

"

::= { j2eeJvmStatEntry 1 }

j2eeJvmStatHeapSize OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "

Returns the size of the JVM's heap.

```

"
 ::= { j2eeJvmStatEntry 2 }

-----
--
-- J2EE Notifications
--
-----

j2eeStateChangeNotifications OBJECT IDENTIFIER ::= { j2eeNotifications 1 }

j2eeServerStateChange NOTIFICATION-TYPE
  OBJECTS { j2eeSrvMoName, j2eeSrvSMState }
  STATUS current
  DESCRIPTION "
    A server entered RUNNING, STOPPED, or FAILED state.
  "
 ::= { j2eeStateChangeNotifications 0 1 }

j2eeEntityBeanStateChange NOTIFICATION-TYPE
  OBJECTS { j2eeBeanMoName, j2eeBeanSMState, j2eeBeanType }
  STATUS current
  DESCRIPTION "
    An entity bean entered RUNNING, STOPPED, or FAILED state.
  "
 ::= { j2eeStateChangeNotifications 0 2 }

j2eeMessageDrivenBeanStateChange NOTIFICATION-TYPE
  OBJECTS { j2eeBeanMoName, j2eeBeanSMState, j2eeBeanType }
  STATUS current
  DESCRIPTION "
    A message-driven bean entered RUNNING, STOPPED, or
    FAILED state.
  "
 ::= { j2eeStateChangeNotifications 0 3 }

j2eeResourceStateChange NOTIFICATION-TYPE

```

```

OBJECTS { j2eeRsrcMoName, j2eeRsrcSMState, j2eeRsrcType }
STATUS current
DESCRIPTION "
    A resource entered RUNNING, STOPPED, or FAILED state.
"
::= { j2eeStateChangeNotifications 0 4 }

```

```

j2eeJDBCStateChange NOTIFICATION-TYPE
OBJECTS { j2eeJDBCMoSourceName, j2eeJDBCMoDriverName,
    j2eeJDBCSMState }
STATUS current
DESCRIPTION "
    A JDBC driver or data source entered RUNNING,
    STOPPED, or FAILED state.
"
::= { j2eeStateChangeNotifications 0 5 }

```

```

j2eeJCAStateChange NOTIFICATION-TYPE
OBJECTS { j2eeJCAMoConnectionFactoryName,
    j2eeJCAMoManagedConnectionFactoryName,
    j2eeJCASMState }
STATUS current
DESCRIPTION "
    A JCA connection factory or managed connection
    factory entered RUNNING, STOPPED, or FAILED state.
"
::= { j2eeStateChangeNotifications 0 6 }

```

```

-- Revisit:
-- missing NOTIFICATION TYPES for:
-- Domain, Server, Servlet, resourceAdapter, Application, Module, (JVM)
-- Is this intended?

```

```

-----
--
-- J2EE Conformance
--
-----

```

```

j2eeCompliances    OBJECT IDENTIFIER ::= { j2eeConformance 1 }
j2eeObjectGroups  OBJECT IDENTIFIER ::= { j2eeConformance 2 }
j2eeNotificationGroups OBJECT IDENTIFIER ::= { j2eeConformance 3 }

```

```
-- compliance statements
```

```

j2eeBasicCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "The compliance statement for SNMPv2 entities which
    implement this MIB."
  MODULE -- this module
    MANDATORY-GROUPS { j2eeManagedObjectGroup,
                      j2eeStateChangeNotificationGroup }
  -- optional/conditional groups
  GROUP j2eeStatisticsGroup
  DESCRIPTION
    "This group is present only if the agent provides an
    instrumentation for gathering statistics data.
    "

 ::= { j2eeCompliances 2 }

```

```
-- units of conformance
```

```

j2eeManagedObjectGroup OBJECT-GROUP
  OBJECTS {
    j2eeDomIndex,
    j2eeDomMoName,
    j2eeDomEnterprise,
    j2eeDomMoStateManaged,
    j2eeDomMoStatProv,
    j2eeDomMoEventProv,
    j2eeDomSMSState,
    j2eeDomSMSStartTime,
    j2eeSrvIndex,
    j2eeSrvMoName,
    j2eeSrvEnterprise,

```


j2eeSrvVendor,
j2eeSrvVersion,
j2eeSrvMoStateManaged,
j2eeSrvMoStatProv,
j2eeSrvMoEventProv,
j2eeSrvSMState,
j2eeSrvSMStartTime,
j2eeJVMIndex,
j2eeJVMVersion,
j2eeJVMVendor,
j2eeJVMEnterprise,
j2eeJVMNode,
j2eeAppIndex,
j2eeAppMoName,
j2eeAppMoStateManaged,
j2eeAppMoStatProv,
j2eeAppMoEventProv,
j2eeAppSMState,
j2eeAppSMStartTime,
j2eeModIndex,
j2eeModMoName,
j2eeModType,
j2eeModJVMIndex,
j2eeModMoStateManaged,
j2eeModMoStatProv,
j2eeModMoEventProv,
j2eeModSMState,
j2eeModSMStartTime,
j2eeBeanIndex,
j2eeBeanMoName,
j2eeBeanType,
j2eeBeanMoStateManaged,
j2eeBeanMoStatProv,
j2eeBeanMoEventProv,
j2eeBeanSMState,
j2eeBeanSMStartTime,
j2eeSletIndex,
j2eeSletMoName,
j2eeSletMoStateManaged,
j2eeSletMoStatProv,
j2eeSletMoEventProv,
j2eeSletSMState,
j2eeSletSMStartTime,
j2eeAdapIndex,

```

j2eeAdapMoName,
j2eeAdapMoStateManaged,
j2eeAdapMoStatProv,
j2eeAdapMoEventProv,
j2eeAdapSMState,
j2eeAdapSMStartTime,
j2eeRsrcIndex,
j2eeRsrcMoName,
j2eeRsrcType,
j2eeRsrcMoStateManaged,
j2eeRsrcMoStatProv,
j2eeRsrcMoEventProv,
j2eeRsrcSMState,
j2eeRsrcSMStartTime,
j2eeJCAIndex,
j2eeJCAMoConnectionFactoryName,
j2eeJCAMoManagedConnectionFactoryName,
j2eeJCAMoStateManaged,
j2eeJCAMoStatProv,
j2eeJCAMoEventProv,
j2eeJCASMSState,
j2eeJCASMSStartTime,
j2eeJDBCIndex,
j2eeJDBCMoSourceName,
j2eeJDBCMoDriverName,
j2eeJDBCMoStateManaged,
j2eeJDBCMoStatProv,
j2eeJDBCMoEventProv,
j2eeJDBCSMState,
j2eeJDBCSMStartTime
}
STATUS current
DESCRIPTION
    "A collection of attributes implemented by J2EE Managed Objects."
 ::= { j2eeObjectGroups 1 }

```

j2eeStatisticsGroup OBJECT-GROUP

```

OBJECTS {
    j2eeSletStatServiceCount,
    j2eeSletStatServiceMaxTime,
    j2eeSletStatServiceMinTime,
    j2eeSletStatServiceTotal,
    j2eeEjbEntityStatCreateCount,

```

j2eeEjbEntityStatRemoveCount,
j2eeEjbEntityStatReadyCount,
j2eeEjbEntityStatPooledCount,
j2eeEjbStatelessStatCreateCount,
j2eeEjbStatelessStatRemoveCount,
j2eeEjbStatelessStatReadyCount,
j2eeEjbStatefulStatCreateCount,
j2eeEjbStatefulStatRemoveCount,
j2eeEjbStatefulStatReadyCount,
j2eeEjbStatefulStatPassCount,
j2eeEjbMessageDrivenStatCreateCount,
j2eeEjbMessageDrivenStatRemoveCount,
j2eeEjbMessageDrivenStatMessageCount,
j2eeJavaMailStatSentCount,
j2eeJtaStatActiveCount,
j2eeJtaStatCommittedCount,
j2eeJtaStatRolledBackCount,
j2eeJcaConnStatIndex,
 j2eeJcaConnStatWaitTime,
 j2eeJcaConnStatUseTime,
 j2eeJcaConnPoolStatIndex,
 j2eeJcaConnPoolStatWaitTime,
 j2eeJcaConnPoolStatUseTime,
 j2eeJcaConnPoolStatCreateCount,
 j2eeJcaConnPoolStatCloseCount,
 j2eeJcaConnPoolStatPoolSize,
 j2eeJcaConnPoolStatFreePoolSize,
 j2eeJcaConnPoolStatWaitingThreadCount,
 j2eeJdbcConnStatIndex,
 j2eeJdbcConnStatWaitTime,
 j2eeJdbcConnStatUseTime,
 j2eeJdbcConnPoolStatIndex,
 j2eeJdbcConnPoolStatWaitTime,
 j2eeJdbcConnPoolStatUseTime,
 j2eeJdbcConnPoolStatCreateCount,
 j2eeJdbcConnPoolStatCloseCount,
 j2eeJdbcConnPoolStatPoolSize,
 j2eeJdbcConnPoolStatFreePoolSize,
 j2eeJdbcConnPoolStatWaitingThreadCount,
 j2eeJmsConnIndex,
 j2eeJmsSessionIndex,
 j2eeJmsConnSessionStatMessageCount,
 j2eeJmsConnSessionStatPendingMessageCount,
 j2eeJmsConnSessionStatExpiredMessageCount,

```

    j2eeJmsConnSessionStatMessageWaitTime,
    j2eeJmsConnSessionStatDurableSubCount,
    j2eeJmsProducerIndex,
    j2eeJmsProducerStatMessageCount,
    j2eeJmsProducerStatPendingMessageCount,
    j2eeJmsProducerStatExpiredMessageCount,
    j2eeJmsProducerStatMessageWaitTime,
    j2eeJmsProducerStatDestination,
    j2eeJmsConsumerIndex,
    j2eeJmsConsumerStatMessageCount,
    j2eeJmsConsumerStatPendingMessageCount,
    j2eeJmsConsumerStatExpiredMessageCount,
    j2eeJmsConsumerStatMessageWaitTime,
    j2eeJmsConsumerStatOrigin,
    j2eeJvmStatUpTime,
    j2eeJvmStatHeapSize
}
STATUS current
DESCRIPTION
    "A collection of tables providing statistics about managed
    objects."
::= { j2eeObjectGroups 2 }

j2eeStateChangeNotificationGroup NOTIFICATION-GROUP
NOTIFICATIONS {
    j2eeServerStateChange,
    j2eeEntityBeanStateChange,
    j2eeMessageDrivenBeanStateChange,
    j2eeResourceStateChange,
    j2eeJDBCStateChange,
    j2eeJCAStateChange
}
STATUS current
DESCRIPTION
    "The list of State Change Notifications that an agent should
    emit."
::= { j2eeNotificationGroups 1 }

END

```

JSR77.8.6 IANA Considerations

The Internet Assigned Numbers Authority (IANA) is responsible for maintaining a MIB module which provides OID registrations for well-known languages. The IANA language registry is intended to reduce interoperability problems by providing a single list of well-known languages. However, it is of course still possible to register languages in private OID spaces. Registering languages in private spaces is especially attractive if a language is used for experimentation or if a language is only used in environments where the distribution of MIB modules with the language registration does not cause any maintenance problems.

Any additions or changes to the list of languages registered via IANA require Designated Expert Review as defined in the IANA guidelines [RFC2434]. The Designated Expert will be selected by the IESG Area Director for the IETF Operations and Management Area.

JSR77.8.7 Security Considerations

SNMPv1 by itself is not a secure environment. Even if the network itself is secure (for example by using IPsec), there is no control as to who on the secure network is allowed to access and GET (read) the objects in this MIB.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model RFC 2574 [RFC2574] and the View-based Access Control Model RFC 2575 [RFC2575] is recommended.

It is a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET them.

There are no management objects defined in this MIB that have a MAX-ACCESS clause of read-write and/or read-create. If this MIB is implemented correctly, then there is no risk that an intruder can alter or create any management objects of this MIB via direct SNMP SET operations.

It is possible that some information, particularly in the `j2eeRsrcTable`, might be considered sensitive. It is thus important to control even GET access to these objects and possibly to even encrypt the values of these object when sending them over the network via SNMP. Not all implementations of SNMP provide features for such a secure environment.

JSR77.8.8 Intellectual Property

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this chapter or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

JSR77.8.9 Acknowledgments

The J2EE management model this MIB is based on was developed under the Java Community Process (version 2.0) as JSR-077 and includes contributions from the members of the JSR-077 Expert Group (including Hans Hrasna, the spec lead), Sun (as represented by Hans), and other companies. This MIB was written while the author was a member of the JSR-077 Expert Group. Special thanks to Tony Daniel - Informix/IBM, Daniel Fuchs - Sun Microsystems, and Hyther Nizam - AdventNet for their contributions to the MIB.

See <http://www.jcp.org/jsr/detail/77.jsp> for further information.

JSR77.8.10 References

All RFCs can be found at <http://www.ietf.org/rfc/rfcNNNN.txt>, replacing NNNN with the appropriate number.



CHAPTER JSR77.9

J2EE Management CIM

JSR77.9.1 Introduction

The CIM model for J2EE management is a standard specification of the DTMF CIM model known as CIM_Application-J2eeAppServer. This chapter describes the CIM model for J2EE Managed Objects (applications, modules, servers etc), events, state management and statistics. It is optional for J2EE platform implementations to provide a CIM/WBEM implementation of this specification. Implementations that support CIM/WBEM must comply with the CIM model specified in this chapter.

The Managed Object Format (MOF) and UML representation of the model are available at <http://www.dmtf.org/standards/cim>.

JSR77.9.2 Managed Objects

This section deals with modelling basic managed objects such as J2EE servers, applications, modules, resources.

JSR77.9.2.1 J2EEDomain

A domain identifies the scope of the J2EE managed object. As defined in the specification, an instance of J2EEDomain type must be equivalent to the name of the domain it manages. CIM_J2eeDomain is a subclass of CIM_AdminDomain, which inherits from CIM_System.

JSR77.9.2.2 J2EEServer

The J2EE Server is a complex system of applications, modules, resources etc. It can be modelled as a subclass of CIM_System called CIM_J2eeServer.

A J2EE server is associated with multiple Java Virtual Machines, nodes, applications, modules and resources.

JSR77.9.2.2.1 J2EEApplication

A J2EE application is a software entity that is installed on a J2EEServer. An application has two aspects to it - one is the installation/deployment aspect (static) and the second is the runtime aspect.

The installed/deployment aspect is not addressed in this JSR 077, but will be included as part of JSR 088, J2EE Application Deployment.

The runtime/deployed aspect of a J2EE application is modeled in CIM as a J2eeApplication, which is a subclass of the abstract J2EE_DeployedObject class. The J2EE_DeployedObject class is a subclass of CIM_Service and is provided to allow for both J2eeApplication objects as well as J2eeModule objects. An application can be decomposed into the underlying modules that comprise the application. The modules themselves are subclasses of J2EE_DeployedObject because they can be deployed independent of an J2EE application and associated with multiple J2EE applications. A J2eeApplication is comprised of a number of J2eeModule(s) and this fact is captured by the CIM_ServiceComponet association. An example of a module would be the EJBModule which could be represented as the leaf level subclass of J2eeModule.

JSR77.9.2.2.2 J2EEResource

A resource is utilized by a server to provide standard services. A J2EEResource is represented by class CIM_J2eeResource which is a subclass of CIM_Service. This is an abstract class which has concrete subclasses CIM_J2eeJMSResource, CIM_J2eeJNDIResource, CIM_J2eeJDBCResource etc.

These resource classes are associated with a J2eeServer through an aggregated association, J2EEResourceOnServer.

JSR77.9.3 Events

J2EE.9.3.1 Overview of CIM/WBEM Events

The CIM meta model allows special ‘Indication’ types to represent events that may be delivered to subscribers. All indications inherit from the CIM indication class, `CIM_Indication`. The CIM Indications are split into Life Cycle Indications and Process Indications.

‘Life Cycle Indications’ which are created when CIM classes are created, modified and deleted as well as when instances of CIM classes are created, modified, deleted, read and have methods invoked on them. Process Indications are used to represent indications that cannot be expressed as Life Cycle Indications. The CIM indication class that represents these indications is referred to as `CIM_ProcessIndication`. The DMTF has defined SNMP traps as sub classes of `CIM_ProcessIndication`.

The subscription process in WBEM consists of associating an instance of `CIM_IndicationFilter` (which specifies the indications of interest) with a `CIM_IndicationHandler` (which represents the destination). A subscriber may create instances of a filter and handler or may choose to instances which already exists. In CIM associating two or more instances is performed by creating an instance of the appropriate association class. In this case, an instance of a `CIM_IndicationSubscription` association is created to associate the filter and destination.

More details of the CIM Event model can be found at <http://www.dmtf.org/education/whitepapers.php>

JSR77.9.3.2 Modeling J2EE Events in CIM

In order to model J2EE Events, we introduce a subclass of `CIM_ProcessIndication` called `CIM_J2eeNotification`. This subclass has the properties defined in J2EE Management Event with the same semantics.

Note – Since the `J2EEManagedObject` classes are modeled as CIM classes, it is highly recommended that J2EE instrumentation which maps these classes into CIM should take care of Life Cycle indications on these classes.

JSR77.9.3.3 Handling subscriptions

As described above, a subscription is activated when the `CIM_IndicationFilter` instance is associated to the `CIM_IndicationHandler` instance. The `Query` string property within `CIM_IndicationFilter` specifies what particular indications are of interest. A query in the WQL query language would typically consist of boolean conditions on the properties of the `J2eeNotification` class as well as the properties of the referenced `J2eeManagedObject` (which is stored in the `SourceInstance` property as per the MOF). Following is a concrete example:

```
select SourceInstance from J2EE_Indication where J2eeNotification.Type = 'STATE' and
J2eeNotification.SourceInstance.Name = 'EJB125'
```

The above filter requests the source instance from J2EE Notifications whose type is `STATE` (i.e. these are state manageable events) from the `J2eeManagedObject` named `EJB125`. An implementation mapping CIM to J2EE events may add an Event Listener to the appropriate managed object. Once the `J2EEEvent` is received by the listener can be mapped to a `J2EE_Indication` and delivered to the destination specified by the `CIM_IndicationHandler` instance.

JSR77.9.4 Acknowledgments

The JSR-77 expert group would like to thank Guru Bhat and Richard Spellman from Sun Microsystems and Reinhold Kautzleben from SAP for the development of the original CIM mapping of the J2EE Management Model.

The JSR-77 expert group would like to thank the DMTF Application Server subgroup of the DMTF Applications work group for producing the official version of the `CIM_ApplicationServer` model which was introduced in CIM 2.8.