Glossary

access control The methods by which interactions with resources are limited to collections of users or programs for the purpose of enforcing integrity, confidentiality, or availability constraints.

ACID The acronym for the four properties guaranteed by transactions: atomicity, consistency, isolation, and durability.

activation The process of transferring an enterprise bean from secondary storage to memory. See **passivation**.

applet A component that typically executes in a Web browser, but can execute in a variety of other applications or devices that support the applet programming model.

applet container A container that includes support for the applet programming model.

application component provider A vendor that provides the Java classes that implement components' methods, JSP page definitions, and any required deployment descriptors.

application assembler A person who combines components and modules into deployable application units.

application client A first-tier client component that executes in its own Java virtual machine. Application clients have access to some J2EE platform APIs (JNDI, JDBC, RMI-IIOP, JMS).

application client container A container that supports application client components.

application client module A software unit that consists of one or more classes and an application client deployment descriptor.

authentication The process by which an entity proves to another entity that it is acting on behalf of a specific identity. The J2EE platform requires three types of authentication: basic, form-based, and mutual, and it supports digest authentication.

authorization *See* **access control**.

authorization constraint An authorization rule that determines who is permitted to access a Web resource collection.

basic authentication An authentication mechanism in which a Web server authenticates an entity with a user name and password obtained using the Web client's built-in authentication mechanism.

bean-managed persistence Data transfer between an entity bean's variables and a resource manager managed by the entity bean.

bean-managed transaction A transaction whose boundaries are defined by an enterprise bean.

business logic The code that implements the functionality of an application. In the Enterprise JavaBeans model, this logic is implemented by the methods of an enterprise bean.

business method A method of an enterprise bean that implements the business logic or rules of an application.

callback methods Methods in a component called by the container to notify the component of important events in its life cycle.

caller Same as **caller principal**.

caller principal The principal that identifies the invoker of the enterprise bean method.

canonical data model A data model independent of any application that is used for EAI purposes.

client certificate authentication An authentication mechanism in which a client uses a X.509 certificate to establish its identity.

commit The point in a transaction when all updates to any resources involved in the transaction are made permanent.

component An application-level software unit supported by a container. Components are configurable at deployment time. The J2EE platform defines four types of components: enterprise beans, Web components, applets, and application clients.

component contract The contract between a component and its container. The contract includes life cycle management of the component, a context interface that the instance uses to obtain various information and services from its container, and a list of services that every container must provide for its components.

connection *See* **resource manager connection**.

connection factory *See* **resource manager connection factory**.

connector A standard extension mechanism for containers to provide connectivity to enterprise information systems. A connector is specific to an enterprise information system and consists of a resource adapter and application development tools for enterprise information system connectivity. The resource adapter is plugged into a container through its support for system-level contracts defined in the connector architecture.

Connector architecture An architecture for integration of J2EE products with enterprise information systems. There are two parts to this architecture: a resource adapter provided by an enterprise information system vendor and the J2EE product that allows this resource adapter to plug in. This architecture defines a set of contracts that a resource adapter has to support to plug in to a J2EE product, for example, transactions, security, and resource management.

container An entity that provides life cycle management, security, deployment, and runtime services to components. Each type of container (EJB, Web, JSP, servlet, applet, and application client) also provides component-specific services.

container-managed persistence Data transfer between an entity bean's variables and a resource manager managed by the entity bean's container.

container-managed transaction A transaction whose boundaries are defined by an EJB container. An entity bean must use container-managed transactions.

context attribute An object bound into the context associated with a servlet.

conversational state The field values of a session bean plus the transitive closure of the objects reachable from the bean's fields. The transitive closure of a bean is defined in terms of the serialization protocol for the Java programming language, that is, the fields that would be stored by serializing the bean instance.

CORBA Common Object Request Broker Architecture. A language-independent, distributed object model specified by the Object Management Group.

create method A method defined in the home interface and invoked by a client to create an enterprise bean.

credentials The information describing the security attributes of a principal.

CTS Compatibility Test Suite. A suite of compatibility tests for verifying that a J2EE product complies with the J2EE platform specification.

data origin authentication A corroboration that the source of received data is as claimed.

data origin authentication service A security service that verifies that the identity of the original source of some data received by another participant is the identity as claimed. No association is required between the sender and receiver.

delegation An act whereby one principal authorizes another principal to use its identity or privileges with some restrictions.

demilitarized zone (DMZ) A small subnetwork that buffers the trusted internal network, such as a corporate private LAN, from an untrusted external network, such as the public Internet.

deployer A person who installs modules and J2EE applications into an operational environment.

deployment The process whereby software is installed into an operational environment.

deployment descriptor An XML file provided with each module and application that describes how they should be deployed. The deployment descriptor directs a deployment tool to deploy a module or application with specific container options and describes specific configuration requirements that a deployer must resolve.

digest authentication An authentication mechanism in which a Web client authenticates to a Web server by sending the server a message digest along its HTTP request message. The digest is computed by employing a one-way hash algorithm to a concatenation of the HTTP request message and the client's

password. The digest is typically much smaller than the HTTP request and doesn't contain the password.

distributed application An application made up of distinct components running in separate runtime environments, usually on different platforms connected via a network. Typical distributed applications are two-tier (client-server), three-tier (client-middleware-server), and multi-tier (client-multiple, middleware-multiple servers).

DOM Document Object Model. A tree of objects with interfaces for traversing the tree and writing an XML version of it, as defined by the W3Cspecification.

DTD Document Type Definition. A description of the structure and properties of a class of XML files.

EAR file A JAR archive that contains a J2EE application.

EJB™ *See* **Enterprise JavaBeans**.

EJB container A container that implements the EJB component contract of the J2EE architecture. This contract specifies a runtime environment for enterprise beans that includes security, concurrency, life cycle management, transaction, deployment, naming, and other services. An EJB container is provided by an EJB or J2EE server.

EJB container provider A vendor that supplies an EJB container.

EJB context An object that allows an enterprise bean to invoke services provided by the container and to obtain the information about the caller of a client-invoked method.

EJB home object An object that provides the life cycle operations (create, remove, find) for an enterprise bean. The class for the EJB home object is generated by the container's deployment tools. The EJB home object implements the enterprise bean's home interface. The client references an EJB home object to perform life cycle operations on an EJB object. The client uses JNDI to locate an EJB home object.

EJB JAR file A JAR archive that contains an EJB module.

EJB module A software unit that consists of one or more enterprise beans and an EJB deployment descriptor.

EJB object An object whose class implements the enterprise bean's remote interface. A client never references an enterprise bean instance directly; a client always references an EJB object. The class of an EJB object is generated by the container's deployment tools.

EJB server Software provides services to an EJB container. For example, an EJB container typically relies on a transaction manager that is part of the EJB server to perform the two-phase commit across all the participating resource managers. The J2EE architecture assumes that an EJB container is hosted by an EJB server from the same vendor, so it does not specify the contract between these two entities. An EJB server may host one or more EJB containers.

EJB server provider A vendor that supplies an EJB server.

enterprise bean A component that implements a business task or business entity and resides in an EJB container; either an entity bean or a session bean.

enterprise information system The applications that comprise an enterprise's existing system for handling company-wide information. These applications provide an information infrastructure for an enterprise. An enterprise information system offers a well-defined set of services to its clients. These services are exposed to clients as local and/or remote interfaces. Examples of enterprise information systems include enterprise resource planning systems, mainframe transaction processing systems, and legacy database systems.

enterprise information system resource An entity that provides enterprise information system-specific functionality to its clients. Examples are a record or set of records in a database system, a business object in an enterprise resource planning system, and a transaction program in a transaction processing system.

enterprise bean provider An application programmer who produces enterprise bean classes, remote and home interfaces, and deployment descriptor files, and packages them in an EJB .jar file.

Enterprise JavaBeans™ (EJB™) A component architecture for the development and deployment of object-oriented, distributed, enterprise-level applications. Applications written using the Enterprise JavaBeans architecture are scalable, transactional, and secure.

entity bean An enterprise bean that represents persistent data maintained in a database. An entity bean can manage its own persistence or it can delegate this function to its container. An entity bean is identified by a primary key. If the container in which an entity bean is hosted crashes, the entity bean, its primary key, and any remote references survive the crash.

finder method A method defined in the home interface and invoked by a client to locate an entity bean.

form-based authentication An authentication mechanism in which a Web container provides an application-specific form for logging in.

group A collection of principals within a given security policy domain.

handle An object that identifies an enterprise bean. A client may serialize the handle and then later deserialize it to obtain a reference to the enterprise bean.

home interface One of two interfaces for an enterprise bean. The home interface defines zero or more methods for creating and removing an enterprise bean. For session beans, the home interface defines create and remove methods, while for entity beans, the home interface defines create, finder, and remove methods.

home handle An object that can be used to obtain a reference of the home interface. A home handle can be serialized and written to stable storage and deserialized to obtain the reference.

HTML HyperText Markup Language. A markup language for hypertext documents on the Internet. HTML enables the embedding of images, sounds, video streams, form fields, references to other objects with URLs, and basic text formatting.

HTTP HyperText Transfer Protocol. The Internet protocol used to fetch hypertext objects from remote hosts. HTTP messages consist of requests from client to server and responses from server to client.

HTTPS HTTP layered over the SSL protocol.

impersonation An act whereby one entity assumes the identity and privileges of another entity without restrictions and without any indication visible to the recipients of the impersonator's calls that delegation has taken place. Impersonation is a case of simple delegation.

integrity mechanisms Mechanisms that ensure that outside parties cannot tamper with communication between entities. Integrity mechanisms prevent outside parties from intercepting and modifying communication between entities, and they ensure that messages can be used only once.

IDL Interface Definition Language. A language used to define interfaces to remote CORBA objects. The interfaces are independent of operating systems and programming languages.

IIOP Internet Inter-ORB Protocol. A protocol used for communication between CORBA object request brokers.

initialization parameter A parameter that initializes the context associated with a servlet.

ISV Independent Software Vendor.

J2EE™ Java 2, Enterprise Edition.

J2ME™ Java 2, Micro Edition.

J2SE™ Java 2, Standard Edition.

J2EE application Any deployable unit of J2EE functionality. This can be a single module or a group of modules packaged into an `.ear` file with a J2EE application deployment descriptor. J2EE applications are typically engineered to be distributed across multiple computing tiers.

J2EE product An implementation that conforms to the J2EE platform specification.

J2EE product provider A vendor that supplies a J2EE product.

J2EE server The runtime portion of a J2EE product. A J2EE server provides Web and/or EJB containers.

JAR Java ARchive. A platform-independent file format that permits many files to be aggregated into one file.

Java™ 2 Platform, Standard Edition (J2SE platform) The core Java technology platform.

Java™ 2 Platform, Enterprise Edition (J2EE platform) An environment for developing and deploying enterprise applications. The J2EE platform consists of a set of services, application programming interfaces (APIs), and protocols

that provide the functionality for developing multitiered, Web-based applications.

Java™ 2 SDK, Enterprise Edition (**J2EE SDK**) Sun's implementation of the J2EE platform. This implementation provides an operational definition of the J2EE platform.

Java IDL A technology that provides CORBA interoperability and connectivity capabilities for the J2EE platform. These capabilities enable J2EE applications to invoke operations on remote network services using the OMG IDL and IIOP.

JavaMail™ An API for sending and receiving e-mail.

Java™ Message Service (JMS) An API for using enterprise messaging systems such as IBM MQ Series, TIBCO Rendezvous, and so on.

Java Naming and Directory Interface™ (JNDI) An API that provides naming and directory functionality.

Java™ Transaction API (JTA) An API that allows applications and J2EE servers to access transactions.

Java™ Transaction Service (JTS) Specifies the implementation of a transaction manager that supports JTA and implements the Java mapping of the OMG Object Transaction Service (OTS) 1.1 specification at the level below the API.

JavaBeans™ component A Java class that can be manipulated in a visual builder tool and composed into applications. A JavaBeans component must adhere to certain property and event interface conventions.

JavaServer Pages™ (JSP) An extensible Web technology that uses template data, custom elements, scripting languages, and server-side Java objects to return dynamic content to a client. Typically the template data is HTML or XML elements, and in many cases the client is a Web browser.

JDBC™ An API for database-independent connectivity between the J2EE platform and a wide range of data sources.

JMS *See* **Java Message Service**.

JNDI *See* **Java Naming and Directory Interface**.

JSP *See* **JavaServer Pages**.

JSP action A JSP element that can act on implicit objects and other server-side objects or can define new scripting variables. Actions follow the XML syntax for elements with a start tag, a body, and an end tag; if the body is empty it can also use the empty tag syntax. The tag must use a prefix.

JSP action, custom An action described in a portable manner by a tag library descriptor and a collection of Java classes and imported into a JSP page by a `taglib` directive. A custom action is invoked when a JSP page uses a *custom tag*.

JSP action, standard An action that is defined in the JSP specification and is always available to a JSP file without being imported.

JSP application A stand-alone Web application, written using the JavaServer Pages technology, that can contain JSP pages, servlets, HTML files, images, applets, and JavaBeans components.

JSP container A container that provides the same services as a servlet container and an engine that interprets and processes JSP pages into a servlet.

JSP container, distributed A JSP container that can run a Web application that is tagged as distributable and is spread across multiple Java virtual machines that might be running on different hosts.

JSP declaration A JSP scripting element that declares methods, variables, or both in a JSP file.

JSP directive A JSP element that gives an instruction to the JSP container and is interpreted at translation time.

JSP element A portion of a JSP page that is recognized by a JSP translator. An element can be a directive, an action, or a scripting element.

JSP expression A scripting element that contains a valid scripting language expression that is evaluated, converted to a `String`, and placed into the implicit `out` object.

JSP file A file that contains a JSP page. In the Servlet 2.2 specification, a JSP file must have a `.jsp` extension.

JSP page A text-based document using fixed template data and JSP elements that describes how to process a request to create a response.

JSP scripting element A JSP declaration, scriptlet, or expression whose tag syntax is defined by the JSP specification and whose content is written according to the scripting language used in the JSP page. The JSP specification describes the syntax and semantics for the case where the language page attribute is "java."

JSP scriptlet A JSP scripting element containing any code fragment that is valid in the scripting language used in the JSP page. The JSP specification describes what is a valid scriptlet for the case where the language page attribute is "java."

JSP tag A piece of text between a left angle bracket and a right angle bracket that is used in a JSP file as part of a JSP element. The tag is distinguishable as markup, as opposed to data, because it is surrounded by angle brackets.

JSP tag library A collection of custom tags identifying custom actions described via a tag library descriptor and Java classes.

JTA *See* **Java Transaction API**.

JTS *See* **Java Transaction Service**.

message signature A means to ensure message integrity. A message signature, which is attached to a message, is a signed (that is, cryptographically enciphered using a public key mechanism) digest of the message contents calcuated using a one-way hash algorithm. A message signature ensures that the receiver can detect any unauthorized modification of the message by anyone other than the message sender.

method permission An authorization rule that determines who is permitted to execute one or more enterprise bean methods.

module A software unit that consists of one or more J2EE components of the same container type and one deployment descriptor of that type. There are three types of modules: EJB, Web, and application client. Modules can be deployed as stand-alone units or assembled into an application.

mutual authentication An authentication mechanism employed by two parties for the purpose of proving each other's identity to one another.

namespace A set of unique names defined for a particular context and which conform to rules specific for the namespace. XML schemas define namespaces.

naming context A set of associations between distinct, atomic, people-friendly identifiers and objects.

naming environment A mechanism that allows a component to be customized without the need to access or change the component's source code. A container implements the component's naming environment and provides it to the component as a JNDI naming context. Each component names and accesses its environment entries using the `java:comp/env` JNDI context. The environment entries are declaratively specified in the component's deployment descriptor.

ORB Object Request Broker. A library than enables CORBA objects to locate and communicate with one another.

OS principal A principal native to the operating system on which the J2EE platform is executing.

passivation The process of transferring an enterprise bean from memory to secondary storage. See **activation**.

peer entity authentication A corroboration that the peer entity in an association is as claimed.

peer entity authentication service A security service that verifies an identity claimed by or for a system entity in an association, thus preventing against a masquerade by the first entity. This service requires an association to exist between the two entities, and the corroboration is valid only at the time that the service is provided.

persistence The protocol for transferring the state of an entity bean between its instance variables and an underlying database.

POA Portable Object Adapter. A CORBA standard for building server-side applications that are portable across heterogeneous ORBs.

principal The identity assigned to a user as a result of authentication.

privilege A security attribute that does not have the property of uniqueness and that may be shared by many principals.

primary key An object that uniquely identifies an entity bean within a home.

QName A QName represents an XML qualified name consisting of a prefix and a local part.

realm *See* **security policy domain**. Also, a string, passed as part of an HTTP request during basic authentication, that defines a protection space. The protected resources on a server can be partitioned into a set of protection spaces, each with its own authentication scheme and/or authorization database.

Reference Implementation *See* **Java 2 SDK, Enterprise Edition**.

remote interface One of two interfaces for an enterprise bean. The remote interface defines the business methods callable by a client.

remove method Method defined in the home interface and invoked by a client to destroy an enterprise bean.

resource adapter A system-level software driver that is used by an EJB container or an application client to connect to an enterprise information system. A resource adapter is typically specific to an enterprise information system. It is available as a library and is used within the address space of the server or client using it. A resource adapter plugs into a container. The application components deployed on the container then use the client API (exposed by adapter) or tool-generated high-level abstractions to access the underlying enterprise information system. The resource adapter and EJB container collaborate to provide the underlying mechanisms—transactions, security, and connection pooling—for connectivity to the enterprise information system.

resource manager Provides access to a set of shared resources. A resource manager participates in transactions that are externally controlled and coordinated by a transaction manager. A resource manager is typically in a different address space or on a different machine from the clients that access it. **Note:** An enterprise information system is referred to as resource manager when it is mentioned in the context of resource and transaction management.

resource manager connection An object that represents a session with a resource manager.

resource manager connection factory An object used for creating a resource manager connection.

RMI Remote Method Invocation. A technology that allows an object running in one Java virtual machine to invoke methods on an object running in a different Java virtual machine.

RMI-IIOP A version of RMI implemented to use the CORBA IIOP protocol. RMI over IIOP provides interoperability with CORBA objects implemented in any language if all the remote interfaces are originally defined as RMI interfaces.

role (development) The function performed by a party in the development and deployment phases of an application developed using J2EE technology. The roles are: Application component provider, application assembler, deployer, J2EE product provider, EJB container provider, EJB server provider, Web container provider, Web server provider, tool provider, and system administrator.

role (security) An abstract logical grouping of users that is defined by the application assembler. When an application is deployed, the roles are mapped to security identities, such as principals or groups, in the operational environment.

role mapping The process of associating the groups and/or principals recognized by the container to security roles specified in the deployment descriptor. Security roles have to be mapped by the deployer before the component is installed in the server.

rollback The point in a transaction when all updates to any resources involved in the transaction are reversed.

SAX Simple API for XML. An event-driven, serial-access mechanism for accessing XML documents.

schema A set of tags and tag structure that may be allowed or are expected in an XML document.

screen scraping A technique for accessing a legacy information system by simulating user interaction with the legacy system's user interface.

security attributes A set of properties associated with a principal. Security attributes can be associated with a principal by an authentication protocol and/or by a J2EE product provider.

security constraint A declarative way to annotate the intended protection of Web content. A security constraint consists of a Web resource collection, an authorization constraint, and a user data constraint.

security context An object that encapsulates the shared-state information regarding security between two entities.

security domain An environment in which entities are presumed to trust one another.

security permission A mechanism, defined by J2SE, used by the J2EE platform to express the programming restrictions imposed on application component providers.

security permission set The minimum set of security permissions that a J2EE product provider must provide for the execution of each component type.

security policy domain A scope over which security policies are defined and enforced by a security administrator. A security policy domain has a collection of users (or principals), uses a well defined authentication protocol(s) for authenticating users (or principals), and may have groups to simplify the setting of security policies.

security role *See* **role (security)**.

security technology domain A scope over which the same security mechanism is used to enforce a security policy. Multiple security policy domains can exist within a single technology domain.

security view The set of security roles defined by the application assembler.

server principal The OS principal that the server is executing as.

service-oriented architecture A set of components that can be invoked and whose interface descriptions can be published and discovered.

servlet A Java program that extends the functionality of a Web server, generating dynamic content and interacting with Web clients using a request-response paradigm.

servlet container A container that provides the network services over which requests and responses are sent, decodes requests, and formats responses. All servlet containers must support HTTP as a protocol for requests and responses, but may also support additional request-response protocols, such as HTTPS.

servlet container, distributed A servlet container that can run a Web application that is tagged as distributable and that executes across multiple Java virtual machines running on the same host or on different hosts.

servlet context An object that contains a servlet's view of the Web application within which the servlet is running. Using the context, a servlet can log events, obtain URL references to resources, and set and store attributes that other servlets in the context can use.

servlet mapping Defines an association between a URL pattern and a servlet. The mapping is used to map requests to servlets.

session An object used by a servlet to track a user's interaction with a Web application across multiple HTTP requests.

session bean An enterprise bean that is created by a client and that usually exists only for the duration of a single client-server session. A session bean performs operations, such as calculations or accessing a database, for the client. While a session bean may be transactional, it is not recoverable should a system crash occur. Session bean objects either can be stateless or can maintain conversational state across methods and transactions. If a session bean maintains state, then the EJB container manages this state if the object must be removed from memory. However, the session bean object itself must manage its own persistent data.

Simple Object Access Protocol (SOAP). A text-based protocol that uses an XML-based data encoding format and HTTP/SMTP to transport messages.

SSL Secure Socket Layer. A security protocol that provides privacy over the Internet. The protocol allows client-server applications to communicate in a way that cannot be eavesdropped or tampered with. Servers are always authenticated and clients are optionally authenticated.

SQL Structured Query Language. The standardized relational database language for defining database objects and manipulating data.

SQL/J A set of standards that includes specifications for embedding SQL statements in methods in the Java programming language and specifications for calling Java static methods as SQL stored procedures and user-defined functions. An SQL checker can detect errors in static SQL statements at program development time, rather than at execution time as with a JDBC driver.

stateful session bean A session bean with a conversational state.

stateless session bean A session bean with no conversational state. All instances of a stateless session bean are identical.

system administrator The person responsible for configuring and administering the enterprise's computers, networks, and software systems.

transaction An atomic unit of work that modifies data. A transaction encloses one or more program statements, all of which either complete or roll back. Transactions enable multiple users to access the same data concurrently.

transaction attribute A value specified in an enterprise bean's deployment descriptor that is used by the EJB container to control the transaction scope when the enterprise bean's methods are invoked. A transaction attribute can have the following values: `Required`, `RequiresNew`, `Supports`, `NotSupported`, `Mandatory`, `Never`.

transaction isolation level The degree to which the intermediate state of the data being modified by a transaction is visible to other concurrent transactions and data being modified by other transactions is visible to it.

transaction manager Provides the services and management functions required to support transaction demarcation, transactional resource management, synchronization, and transaction context propagation.

tool provider An organization or software vendor that provides tools used for the development, packaging, and deployment of J2EE applications.

Universal Description, Discovery and Integration (UDDI) A standards-based specification for Web service registration, description, and discovery. Providers register their Web services in a UDDI registry and requestors use the registry to find services.

URI Uniform Resource Identifier. A compact string of characters for identifying an abstract or physical resource. A URI is either a URL or a URN. URLs and URNs are concrete entities that actually exist. A URI is an abstract superclass.

URL Uniform Resource Locator. A standard for writing a textual reference to an arbitrary piece of data in the World Wide Web. A URL looks like "protocol:// host/localinfo" where "protocol" specifies a protocol for fetching the object (such as HTTP or FTP), "host" specifies the Internet name of the targeted

host, and “localinfo” is a string (often a file name) passed to the protocol handler on the remote host.

URL path The URL passed by a HTTP request to invoke a servlet. The URL consists of the Context Path + Servlet Path + PathInfo, where Context Path is the path prefix associated with a servlet context of which this servlet is a part. If this context is the default context rooted at the base of the Web server’s URL namespace, the path prefix will be an empty string. Otherwise, the path prefix starts with a / character but does not end with a / character. Servlet Path is the path section that directly corresponds to the mapping that activated this request. This path starts with a / character. PathInfo is the part of the request path that follows the Servlet Path but precedes the query string.

URN Uniform Resource Name. A unique identifier that identifies an entity but doesn’t tell where it is located. A system can use a URN to look up an entity locally before trying to find it on the Web. It also allows the Web location to change while still allowing the entity to be found.

user data constraint Indicates how data between a client and a Web container should be protected. The protection can be the prevention of tampering with the data or prevention of eavesdropping on the data.

WAR file A JAR archive that contains a Web module.

Web application An application written for the Internet, including those built with Java technologies such as JavaServer Pages and servlets, as well as those built with non-Java technologies such as CGI and Perl.

Web application, distributable A Web application that uses J2EE technology written so that it can be deployed in a Web container distributed across multiple Java virtual machines running on the same host or different hosts. The deployment descriptor for such an application uses the `distributable` element.

Web component A component that provides services in response to requests; either a servlet or a JSP page.

Web container An entity that implements the Web component contract of the J2EE architecture. This contract specifies a runtime environment for Web components that includes security, concurrency, life cycle management, transaction, deployment, and other services. A Web container provides the same

services as a JSP container and a federated view of the J2EE platform APIs. A Web container is provided by a Web or J2EE server.

Web container, distributed A Web container that can run a Web application that is tagged as distributable and that executes across multiple Java virtual machines running on the same host or on different hosts.

Web container provider A vendor that supplies a Web container.

Web module A unit that consists of one or more Web components and a Web deployment descriptor.

Web resource collection A list of URL patterns and HTTP methods that describe a set of resources to be protected.

Web server Software that provides services to access the Internet, an intranet, or an extranet. A Web server hosts Web sites, provides support for HTTP and other protocols, and executes server-side programs (such as CGI scripts or servlets) that perform certain functions. In the J2EE architecture, a Web server provides services to a Web container. For example, a Web container typically relies on a Web server to provide HTTP message handling. The J2EE architecture assumes that a Web container is hosted by a Web server from the same vendor, so it does not specify the contract between these two entities. A Web server may host one or more Web containers.

Web server provider A vendor that supplies a Web server.

Web Services Description Language A general-purpose XML schema used to specify details of Web service interfaces, bindings, and other deployment details.

XML eXtensible Markup Language. A markup language that allows you to define the tags (markup) needed to identify the data and text in XML documents. J2EE deployment descriptors are expressed in XML.

XSD XML Schema Definition. An XML schema language standardized by W3C to describe XML documents.