What is Websphere Application Server?

Objectives

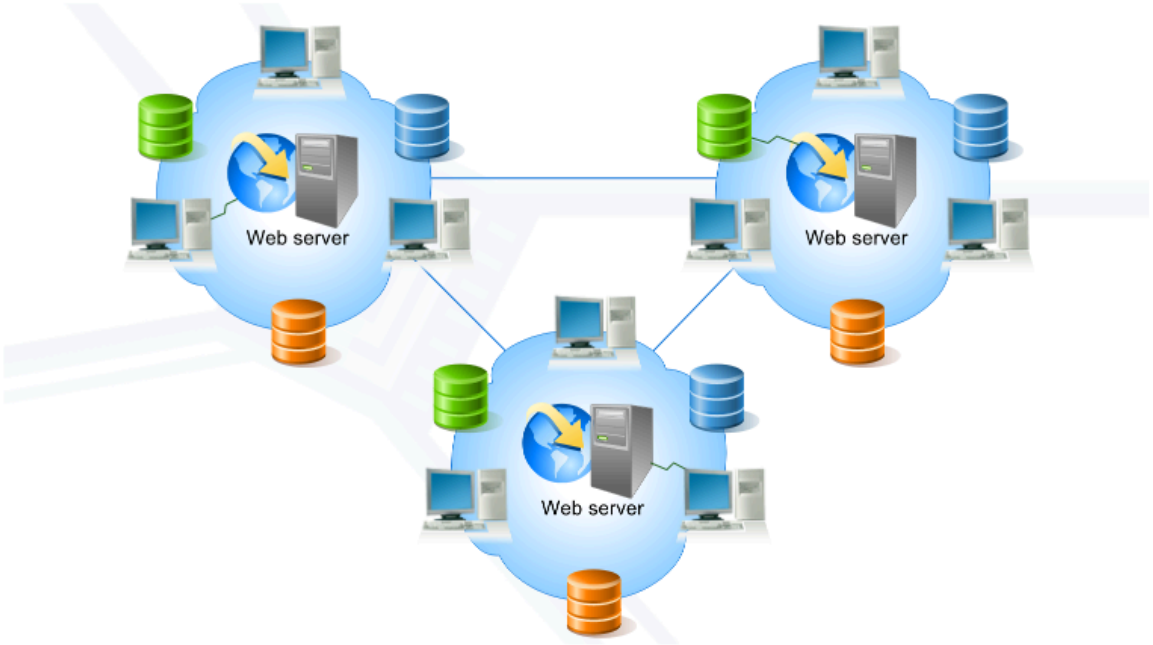
What is Websphere Application Server?

In this module, you will be introduced to WebSphere Application Server, the complex and multifaceted middleware used by large organizations as well as small businesses.

You will also take a brief look at the Java 2 Platform, Enterprise Edition (J2EE) Application Model, a platform-independent environment for developing, building, and deploying Web-based enterprise applications online.

After completing this module, you will be able to:

- Define WebSphere Application Server
- Recognize the Qualities of the J2EE Application Model



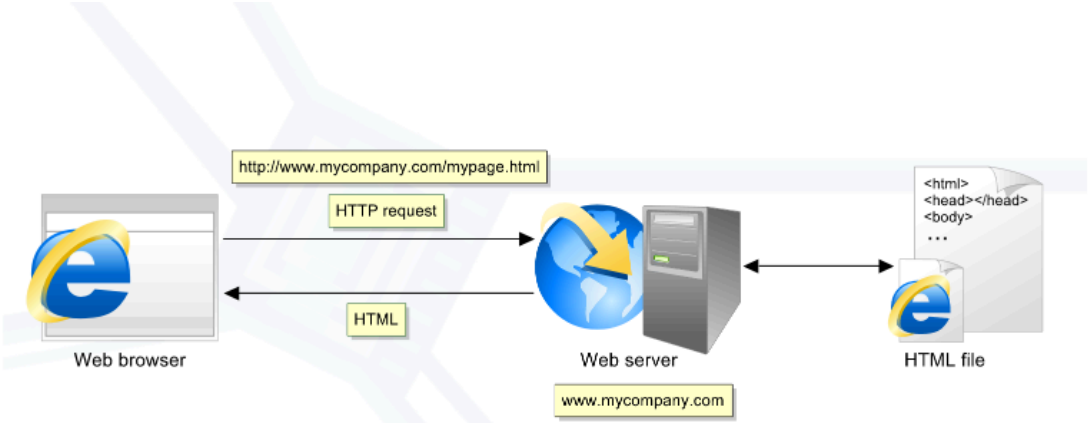
The World Wide Web (WWW), a system of interlinked, hypertext documents that are accessed through the Internet, evolved from a small network that enabled a few people to easily share documents to a huge network of millions of computers.

Not only is it an enormous repository of information, it is the home for e-commerce, which is business transacted electronically over the Internet.

Despite the global popularity of the Web, most corporate data still resides on mainframe computers.

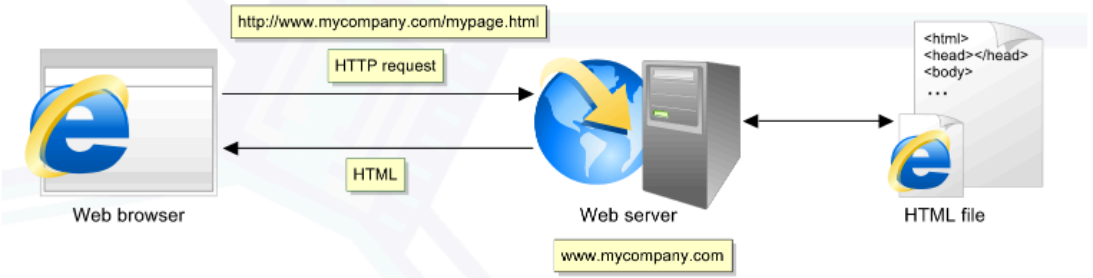
Large organizations have invested substantially in mainframe computers so they are loathe to incur the cost of redeveloping their applications for the new Internet-distributed environment, which is comprised of UNIX midrange and Windows NT servers.

To Web-enable their existing applications, these organizations can use WebSphere Application Server on z/OS.



A client using a Web browser, such as Microsoft Internet Explorer or Mozilla, makes a Hypertext Transfer Protocol (HTTP) request to an HTTP server, which is also known as a Web server, by entering a Uniform Resource Locator (URL). A URL is an addressing scheme that uniquely identifies a protocol, server, port, and file or document.

The HTTP protocol is a simple communications protocol that is used to transfer data from the Web server to the browser. It is a stateless protocol, which means there is no context between "get" transactions; each one is an independent event.

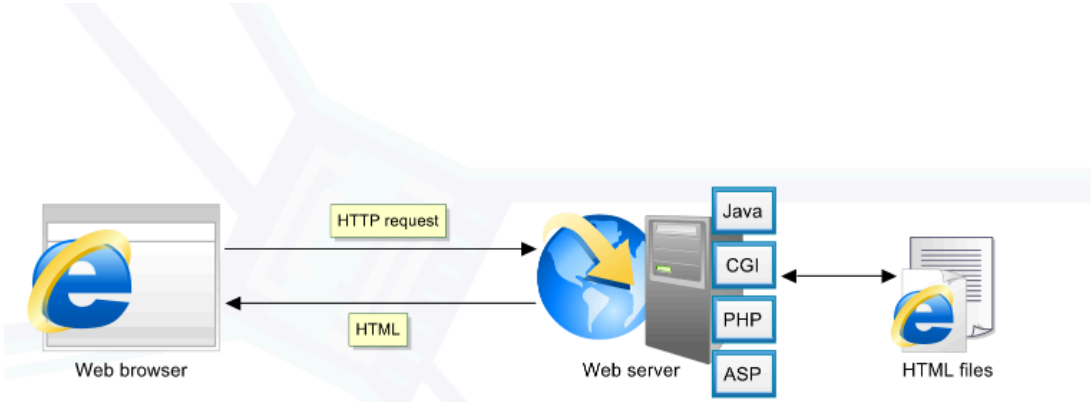


The object of retrieval is usually a Hypertext Markup Language (HTML) page. HTML defines how areas of textual information should be formatted, and uses directives known as tags to define the layout. It can also include graphical images.

HTML enables hypertext links between documents and within a document. These links enable you to retrieve further documents by clicking your mouse button on them. This is how you "surf the Net".

z/OS HTTP Server provides this simple service on z/OS.

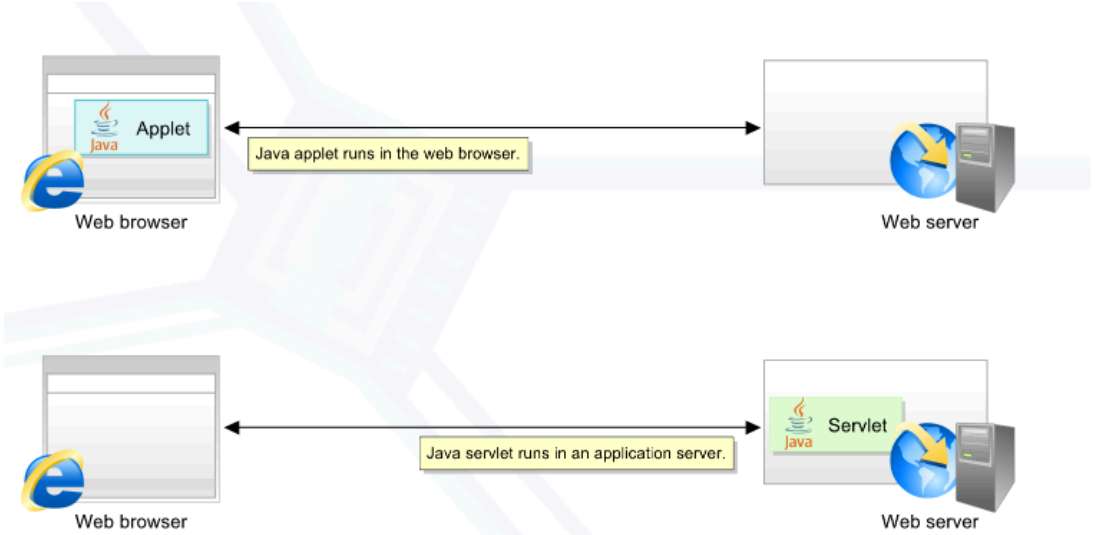
Today, the World Wide Web is about business: business to business and business to customer. This is why an application server framework is so important.



Although many Web pages use a simple HTML interface, this has many drawbacks. HTML pages by themselves are static: they are always the same. However sophisticated web pages, like internet banking, need pages that can change. So every customer sees their own individual account details. The pages need to be dynamic.

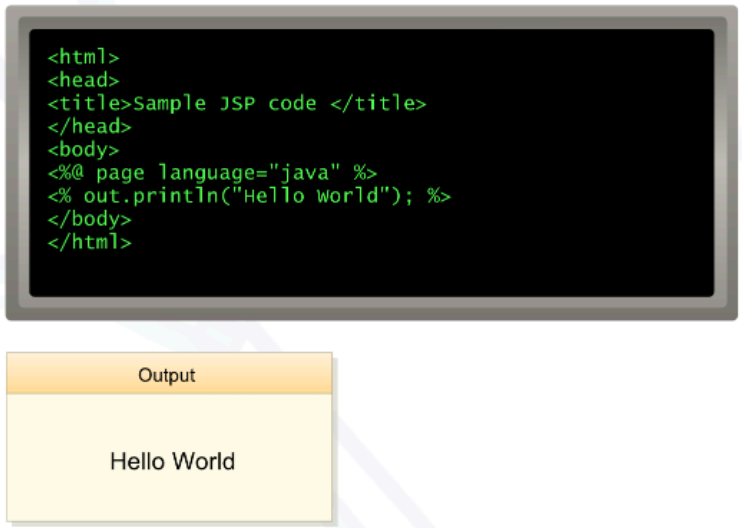
There are many ways to make dynamic web pages, including PHP, ASP, and CGI programs. However one of the most popular ways is to use Java.

Java is one of the most popular ways of making web pages dynamic. Java is a language that can be used on all computer platforms: from Windows and UNIX to z/OS.

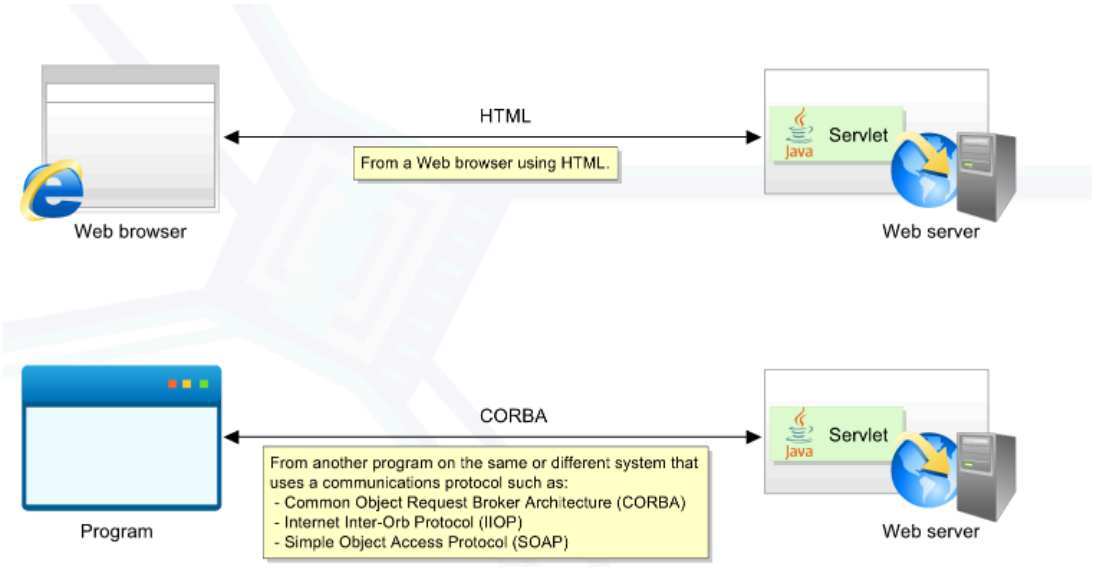


Java code can run in the web browser (a Java applet), or in the web server (a Java servlet).

A web server that runs Java servlets becomes an application server.



Java code can even be directly embedded into an HTML file, and compiled and run when the HTML file is requested. These are known as Java Server Pages (JSP).



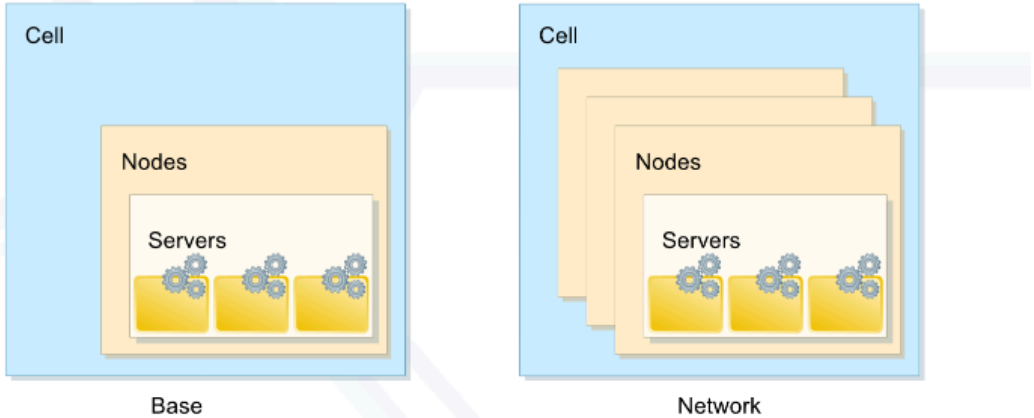
A server running Java servlets adheres to a specification called Java Enterprise Edition. This specification has changed over the past few years. Today servers use the Java 2 Enterprise Edition, or J2EE.

J2EE servers can service requests from web browsers, as well as other requestors.

WebSphere Application Server for z/OS is a comprehensive Java 2 Enterprise Edition (J2EE) and Web services application system.

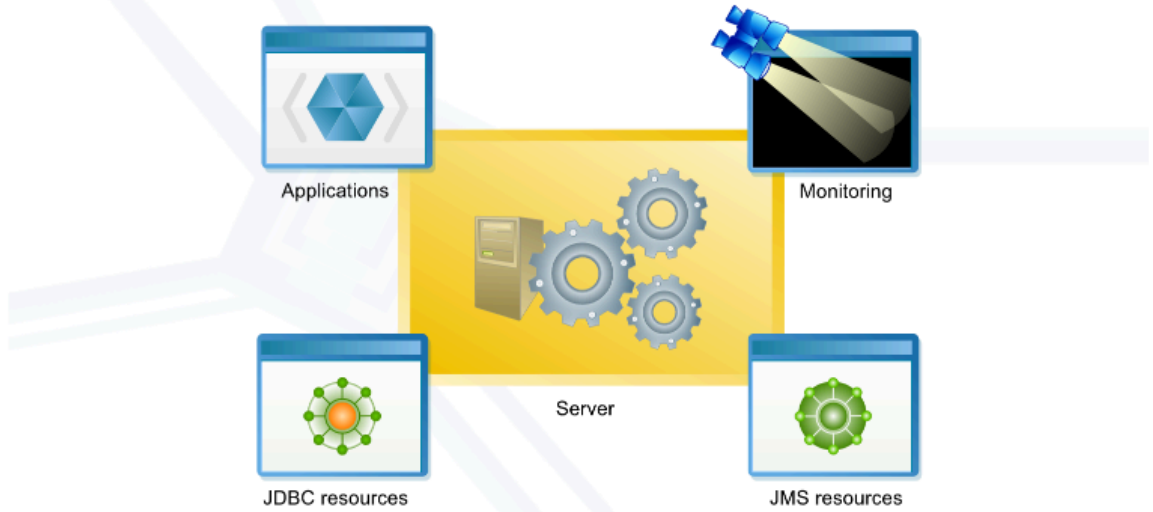
It is built on open standards-based technology, such as CORBA, HTML, HTTP, IIOP, and J2EE-compliant Java technology standards for servlets, JSP, and Enterprise JavaBeans (EJB). It also supports all Java APIs needed for J2EE compliance.

The application server is highly integrated with z/OS and all z/OS subsystems, providing a superior interface to interact with z/OS and manage data, such as CICS, IMS, and DB2, and deliver it to the Web.



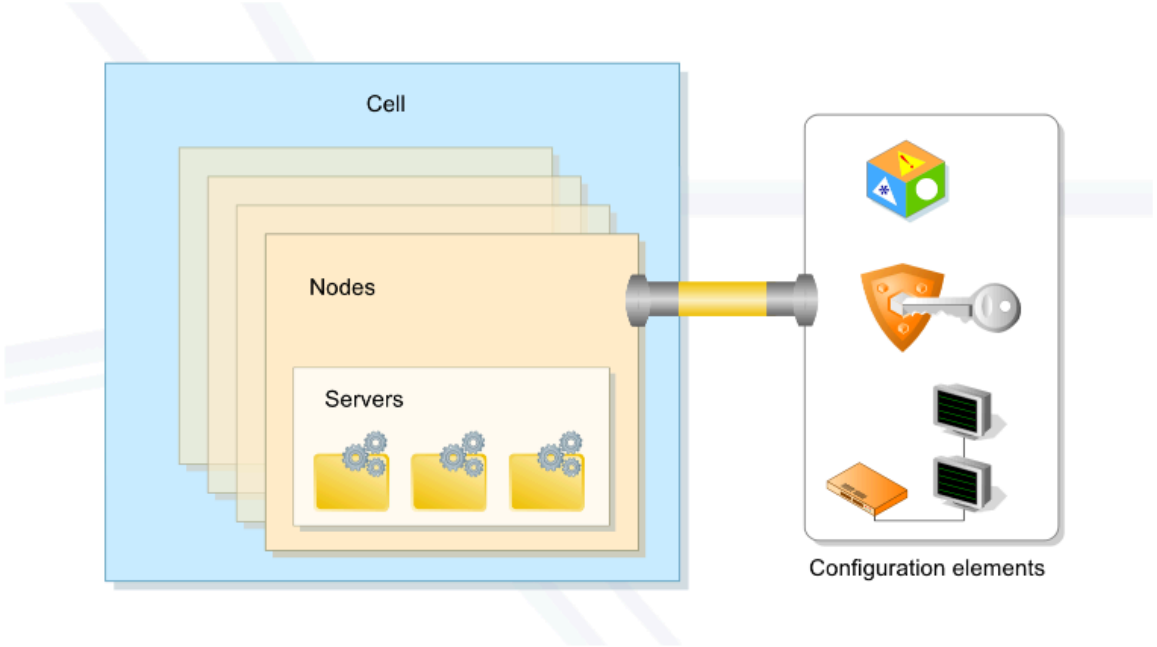
WebSphere Application Server on z/OS can be set up in two different types of deployments, base and network.

Each deployment uses a hierarchy of servers, nodes, and cells, but nodes and cells are only important in a network deployment.

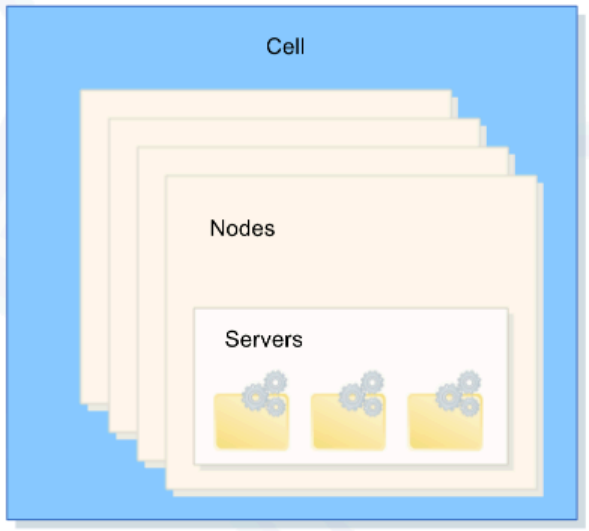


A server is the primary run-time component; this is where your application executes.

The server provides containers and services enabling the execution of specific Java application components. Each server runs in its own Java Virtual Machine (JVM).



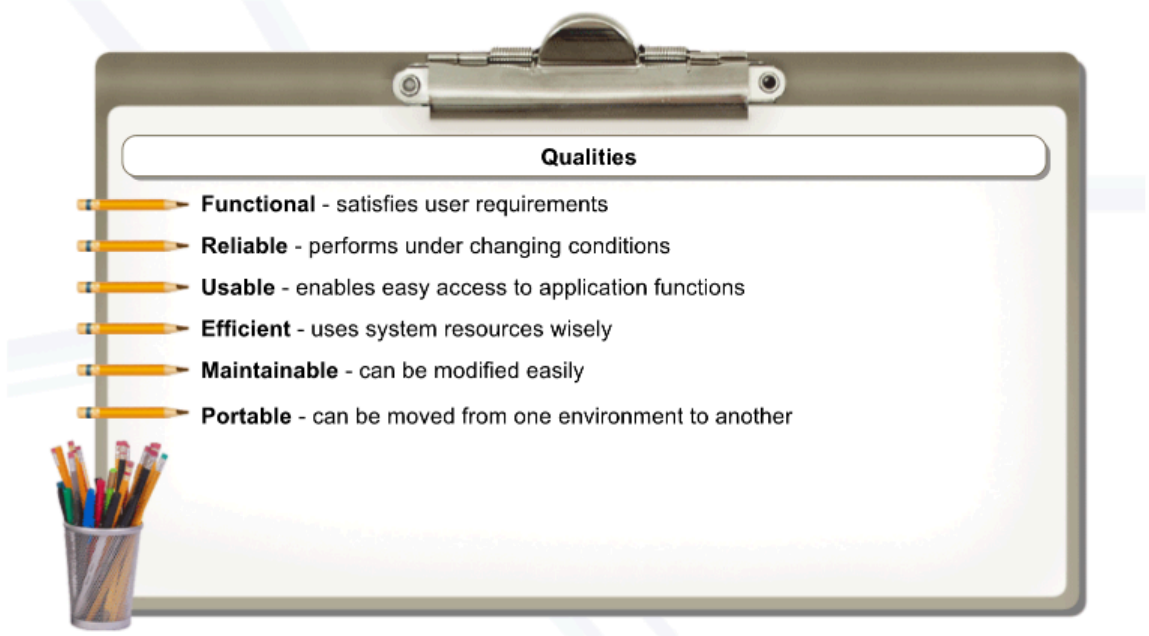
A node is a logical grouping of server processes. They share a common configuration and operational control. A node is usually associated with a physical installation of the WebSphere Application Server.



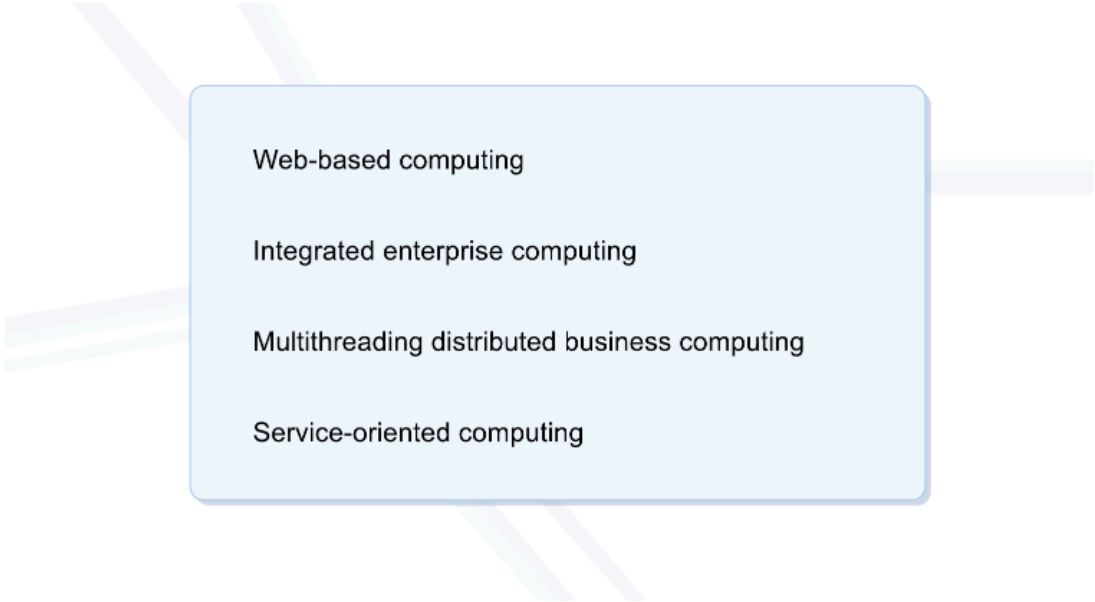
A cell is a grouping of nodes into a single administrative domain.

In the base configuration, a cell contains only one node, which may have multiple servers.

In the network deployment configuration, a cell can consist of multiple nodes that are administered together.



The J2EE application model is implemented on many platforms. In WebSphere Application Server for z/OS, it follows the specification and exhibits the qualities listed above.



WebSphere Application Server on z/OS supports the four major models of application design listed above.

These models focus on separating the application logic from the underlying infrastructure. With z/OS data, this means that Web-based applications can be designed and built to access data and logic held in, for example, CICS, DB2, or IMS.

Summary

What is Websphere Application Server?

In this module, you were introduced to WebSphere Application Server. You also looked briefly at the J2EE Application Model.

You should now be able to:

- Define WebSphere Application Server
- Recognize the Qualities of the J2EE Application Model