

Basic Deployment

At the end of this module you will be able to:

- ✓ Describe the Web server capabilities of WebLogic Server
- ✓ Use static and dynamic deployment
- ✓ Work with the built-in WebLogic Server servlets
- ✓ Define and work with enterprise applications

Road Map



1. Web Servers

- Web Servers Defined
- HTTP
- Static and Dynamic Content

2. Web Applications

3. EJB Applications

4. Enterprise Applications

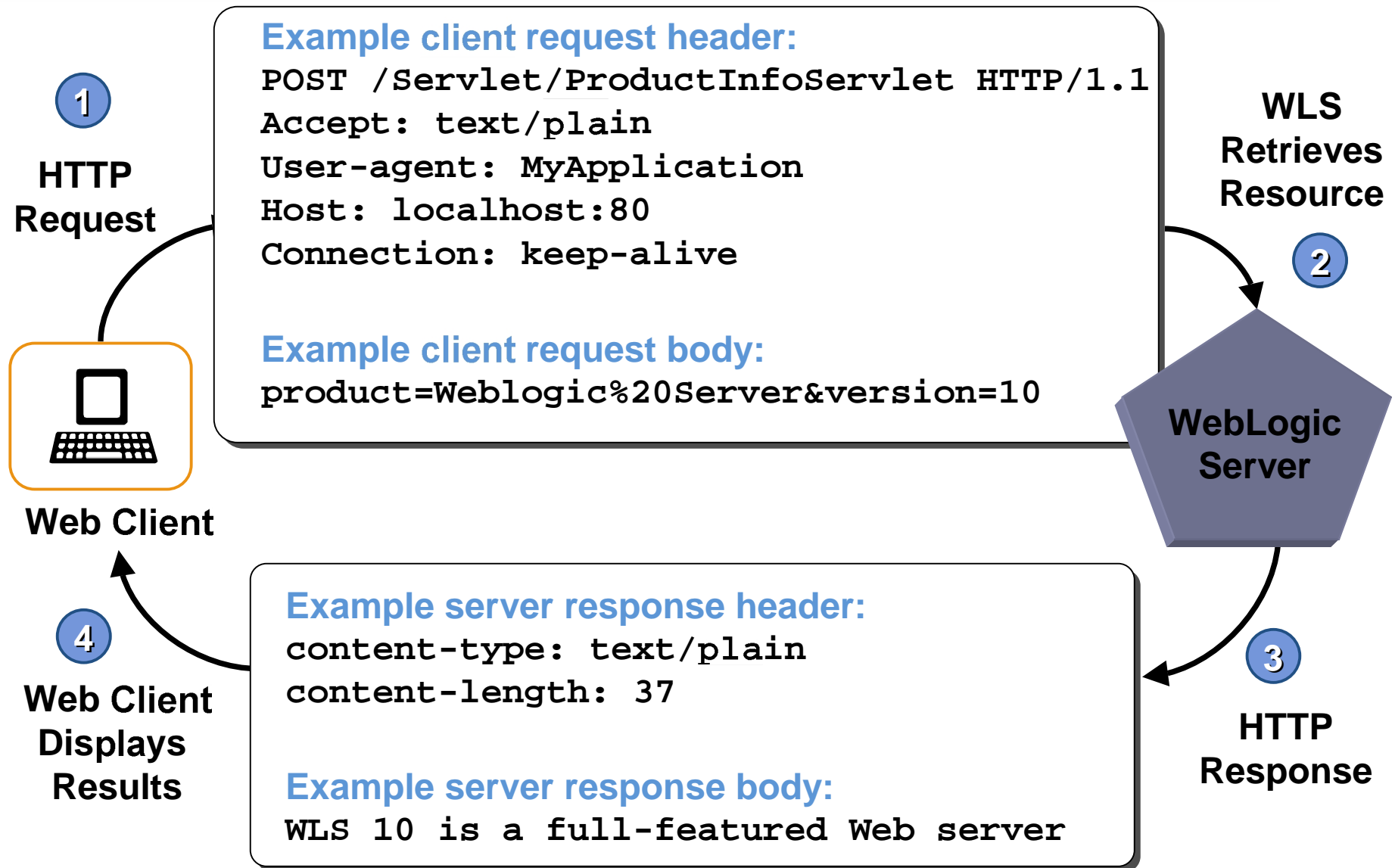
5. Deployment

The Role of Web Servers



- ▶ Web servers are responsible for handling HTTP requests from clients.
- ▶ Web servers typically return:
 - Static content (HTML pages, graphics, ...)
 - Dynamic content (Servlet, JSPs, CGIs, ...)

A Typical Web Interaction



- ▶ Multipurpose Internet Mail Extensions (MIME) is a protocol for identifying and encoding binary data.
- ▶ All HTTP response data is encoded with a MIME content type.
- ▶ Browsers interpret HTTP response data differently depending upon the MIME type of the data:
 - HTML pages are parsed and displayed.
 - PDF documents can be sent to Adobe Acrobat.
 - Application code can be directly executed.

HTTP Status Codes



▶ HTTP status codes:

- Indicate to the client whether or not the request was successful
- Provide the client a reason for a failed request
- Are used by clients to provide alternate behavior

Indicating success:

The default status code is 200, which indicates success.

Reason for failure:

A status code of 404 tells the client the requested resource was not found.

Providing alternate behavior:

If a browser receives a 401 status code, the browser can prompt the user for an ID and password to login. WLS 10 is a full-featured Web server.

- ▶ Static content documents are predefined on the server and do not change.
- ▶ WebLogic Server can be used to serve static content such as:
 - HTML documents
 - Images
 - PDF documents
- ▶ WebLogic Server can serve static documents:
 - Over standard HTTP
 - Through SSL using HTTPS

Dynamic Content



- ▶ Dynamic content documents may change based on the client's request.
- ▶ HTML documents can be created by using:
 - Servlets
 - JSPs
 - Common Gateway Interface (CGI) programs

Section Review



In this section we discussed:

- ✓ The role of Web servers
- ✓ HTTP requests, responses, MIME types, status codes
- ✓ Serving static HTML, images and files
- ✓ Serving JSP and servlet requests



Road Map



1. Web Servers
2. **Web Applications**
 - Web Applications
 - Directory Structure and Deployment Descriptors
 - Using the Console to Deploy Web Applications
 - Monitoring Web Applications
3. EJB Applications
4. Enterprise Applications
5. Deployment

What Is a Web Application?



- ▶ A *Web application* is a group of server-side resources that create an interactive online application.
- ▶ Server-side resources include:
 - Servlets (small server-side applications)
 - JavaServer Pages (dynamic content)
 - Static documents (HTML, images)
 - Server-side classes
 - Client-side applets and beans

Packaging Web Applications







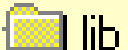


- ▶ Before deploying an application package and registering it with a WLS server, follow these steps to package a Web App:

1. Arrange resources in a prescribed directory structure.
2. Develop `web.xml` Deployment Descriptor (or copy as required).
3. Develop `weblogic.xml` Deployment Descriptor (WLS-Specific).
4. Archive Web App into `.war` file using `jar`.
5. Deploy Web App onto WLS.
6. Configure Web App with WLS Administration Console.

Web Application Structure



- ▶ The structure of Web applications is defined by the Servlet specification.
- ▶ A Web application can be either:
 - An archived file (.war file)
 - An expanded directory structure

Directory/File	Description
 MyWebApplication	Document root of Web application
 META-INF	Information for archive tools (manifest)
 WEB-INF	Private files that will not be served to clients
 classes	Server-side classes such as servlets and applet
 lib	.jar files used by Web app
 web.xml	Web app deployment descriptor
 weblogic.xml	WLS-specific deployment descriptor

Configuring Web Applications



- ▶ Web applications are configured through *deployment descriptors* `web.xml` and `weblogic.xml` which:
 - Define run-time environment
 - Map URLs to servlets and JSPs
 - Define application defaults such as welcome and error pages
 - Specify J2EE security constraints
 - Define work managers for applications
 - Set the context-root for the application

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/908127142025006074>