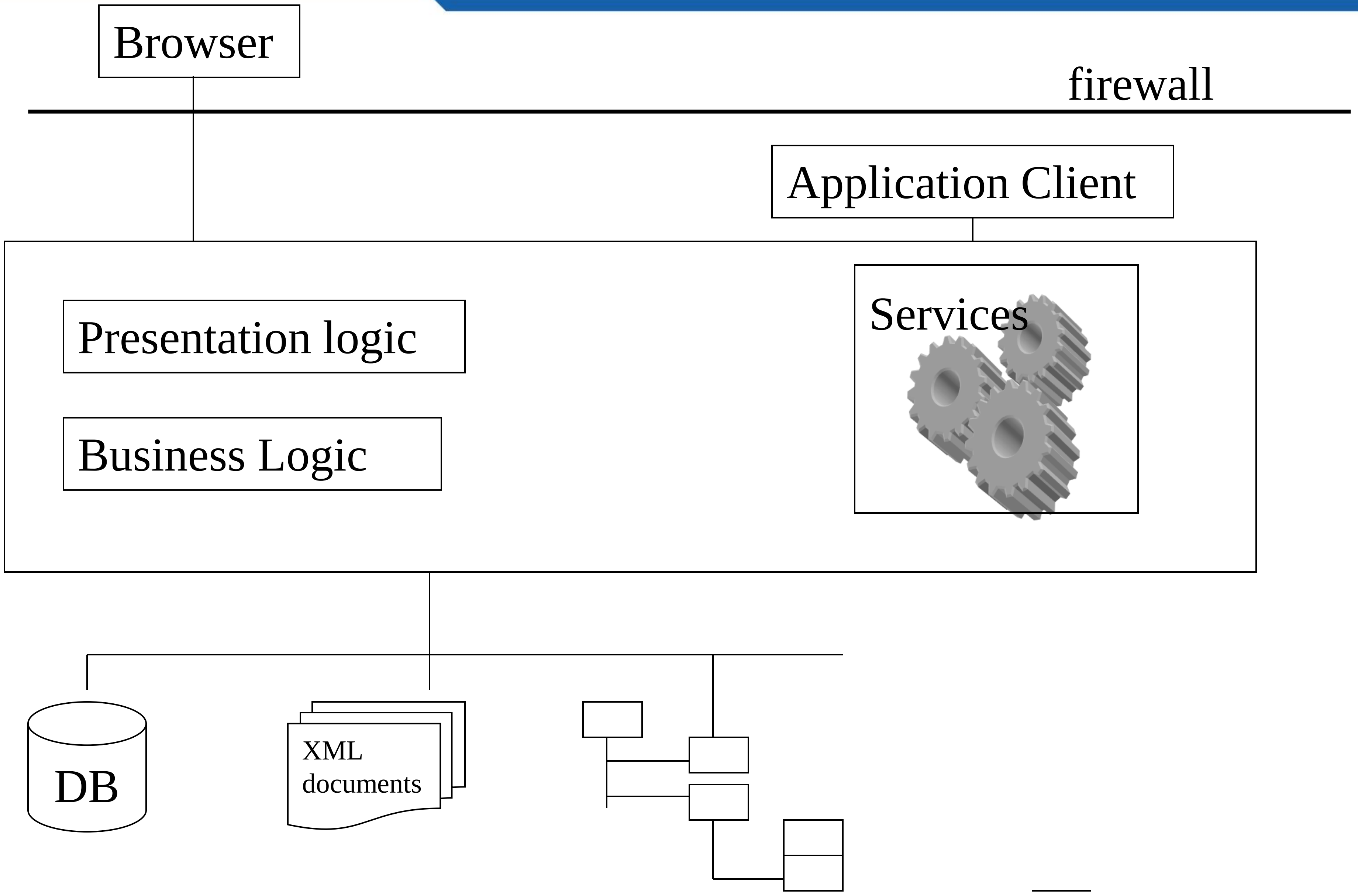
Introduction to JEE

What is an enterprise application?

- Software that performs business functions such as accounting, customer information management, sales and so on
- Aims to improve the organization's performance
- Software application for the enterprises of today…and tomorrow…
    - Need for rapid scalability (both in terms of volume and geographical spread)
    - Need to accommodate multiple types of device access
    - Need to integrate with diverse software applications
    - Need to be very change-friendly
    - High degree of security reliability and performance

Enterprise application n-tier architecture



What is a web application?

- Client accesses applications using a browser.
- Client typically requests for a static or dynamic resource/page from a server connected to the internet. This server is a web server.
- The client typically gets back HTML page as response that can view in the browser. Further interaction with the server can happen using the HTML page.

Static and Dynamic resource/page

- Static pages/resources are the files that are pre-created in a particular path in the web server.
- Dynamic pages are created by the application on the fly.

Web server

- Request for a static resources is met by the web server.
- The web server understands HTTP protocol.
- Example: Apache web server.

Application server

- Application server is an environment where applications can reside and run.
- The definition extended to support web applications as well.
- A web server which can run web application is an application server. In other words, web server is embedded in application server. For instance , Apache web server is embedded in most of the popular application server’s of today like Tomcat, Jboss.
- JEE (Java Platform, Enterprise Edition) application servers support Java based applications and provides API and supporting services.
- Request for a dynamic resource is met by application server.

JEE Application server

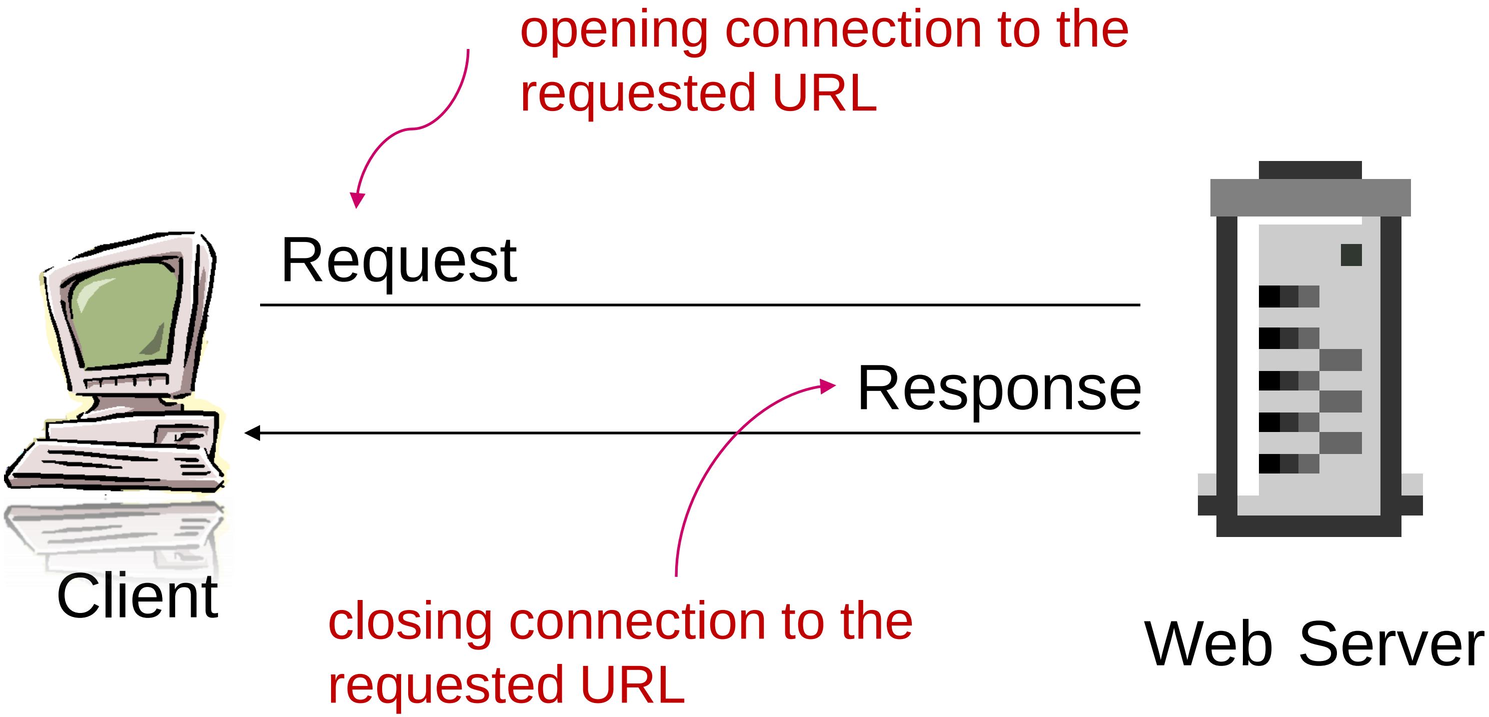
- Apache Tomcat Web Server
- Jboss Application server
- Websphere Application Server from IBM
- Oracle WebLogic Server.

- We will be using Apache Tomcat Web Server 7.0 for our sessions.

HTTP Protocol

- HTTP is an application level protocol (generally implemented over TCP/IP) used to transfer information between the web clients and web applications.
- It provides a set of rules that computers use to communicate over the internet.
- Communication is done by sending packets.
- The packet structure should be is defined by the protocol.
- Packets are request and response packet.
- HTTP 1.1 is stateless implies that after one cycle of request-response the connection between the client and the web server is lost.

HTTP and Statelessness



opening connection to the requested URL

Request

Response

Client

closing connection to the requested URL

Web Server

HTTP Methods

- GET
- POST

Our focus!

- OPTIONS
- HEAD
- PUT
- DELETE
- TRACE
- CONNECT

The GET request method

- Used for accessing static resources such as HTML pages or images.
- Example: requesting for a page www.yahoo.com
- Can also be used to retrieve information that is formed dynamically (example: page generated by the application in a response to a specific user query).
- Form tag assumes GET method when method attribute is not specified.
- Example: **www.mySite.com/index.do?city=Bangalore**
- The convention has been established that the GET (and HEAD methods) SHOULD NOT have the significance of taking an action other than retrieval. These methods ought to be considered "safe".

A sample GET Packet

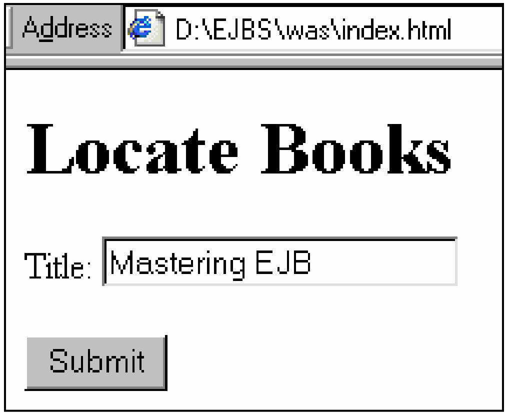
```
GET /mywebdir/index.do?city=Bangalore HTTP/1.1
Host:www.mySite.com
User-Agent:Mozilla/5.0…
Accept: text/xml,application/xml, text/html, image/gif
Accept-Language: en-us
Accept-Encoding: gzip, deflate
Keep-Alive: 300
Connection: keep-alive
```

Request Header

Request Body

```
<html>
<head>
<title>Book list</title>
</head>
<body>
<h1>Locate Books</h1>
<form name=“findBook" method=“get" action="FindBook">
  Title:
  <input name="title" type="text"><br><br>
  <input type="submit" name="Submit" value="Submit">
</form>
</body>
</html>
```

optional



Address D:\EJBS\was\index.html

Locate Books

Title: Mastering EJB

Query String

Request goes to the application as:
http://myWeb/FindBook?title=“Mastering%20EJB”

URL: Uniform Resource Locator

The POST Request

- This method is used to send large amount of data to the server.
- It is commonly used for accessing dynamic resources.
- Data is sent within the body of the request.
- If data such as password is transmitted, the post method is preferred.
- According to W3C, POST is designed to allow a uniform method to cover the following functions:
  - -Annotation of existing resources; -
  - Posting a message to a bulletin board, newsgroup, mailing list, or similar group of articles; -
  - Providing a block of data, such as the result of submitting a form, to a data-handling process; -
  - Extending a database through an append operation.

A sample POST Packet

```
POST /mywebdir/index.do HTTP/1.1

Host:www.mySite.com

User-Agent:Mozilla/5.0…

Accept: text/xml,application/xml, text/html, image/gif

Accept-Language: en-us

Accept-Encoding: gzip, deflate

Keep-Alive: 300

Connection: keep-alive
```

Request Header

```
city=Bangalore&state=Karnataka
```

Request Body

A sample response packet

Status code which indicates successful request acceptance

```
HTTP/1.1 200 OK
Set-Cookie:JSESSIONID=7277272818;Path=/xyz
Content-Type: text/html
Content-Length:230
Date: Sun, 1 Apr 2007 13:45:30 GMT
Server: Apache Server/1.1
Connection:close
```

Response Header

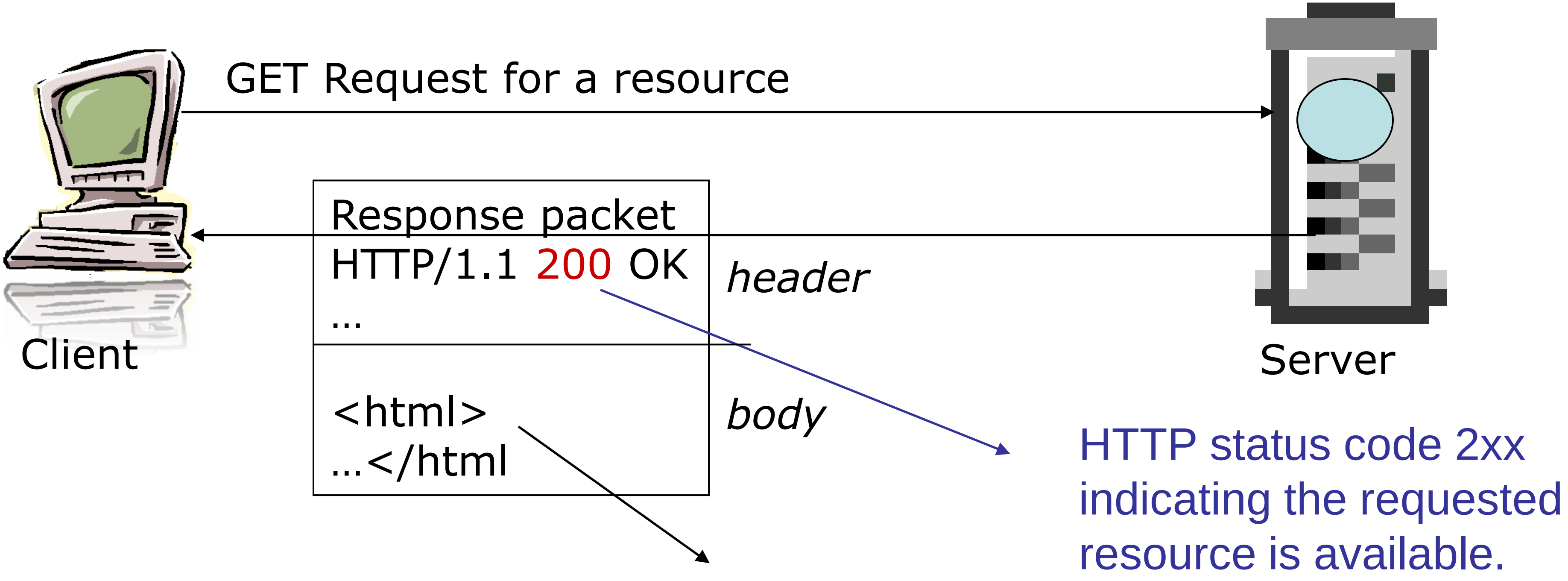
```
<html>
…
</html>
```

Response Body

Status codes

Status Code	Type	Description
1XX	Information	Request received, continuing to process
2XX	Success	The was successfully received and accepted
3XX	Redirection	Further action must be in order to complete the request
4XX	Client Error	Request cannot be fulfilled because of error in client side
5XX	Server Error	Request cannot be fulfilled because of error in server side

Status 2xx



GET Request for a resource

Client

Server

Response packet
HTTP/1.1 200 OK
…

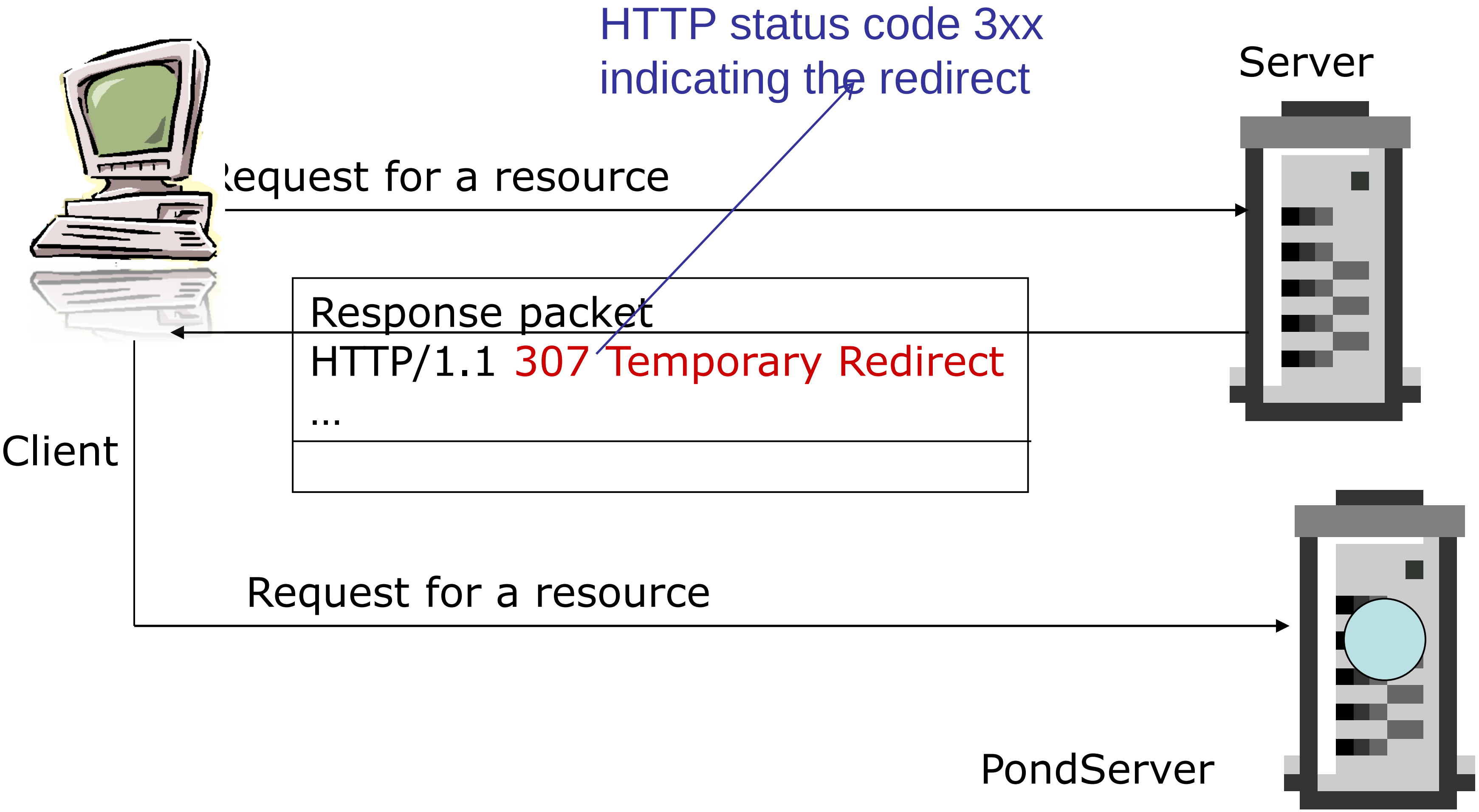
<html>
…</html

header

body

HTTP status code 2xx indicating the requested resource is available.

Status 3xx



HTTP status code 3xx indicating the redirect

Server

Request for a resource

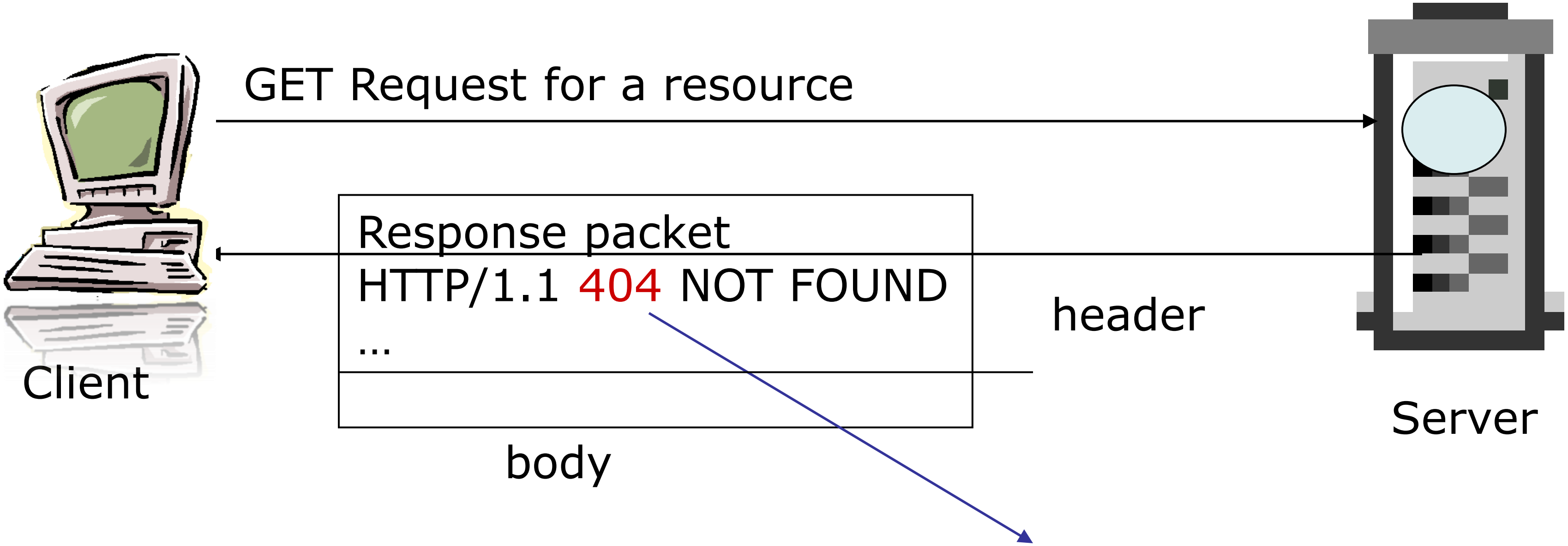
Response packet
HTTP/1.1 307 Temporary Redirect
…

Client

Request for a resource

PondServer

Status 4xx



GET Request for a resource

Response packet
HTTP/1.1 404 NOT FOUND
…

header

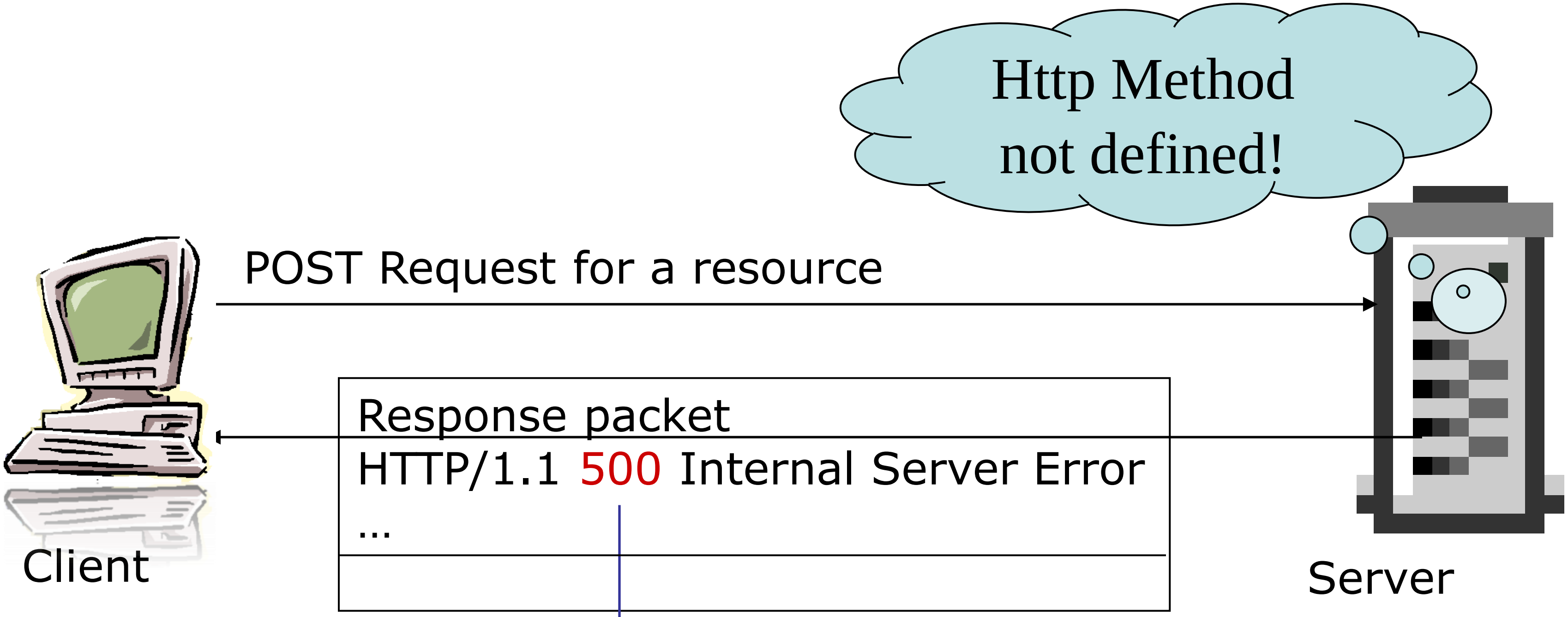
body

Client

Server

HTTP status code 4xx indicating the requested resource is not available.

Status 5xx



Http Method not defined!

POST Request for a resource

Response packet
HTTP/1.1 500 Internal Server Error
…

Client

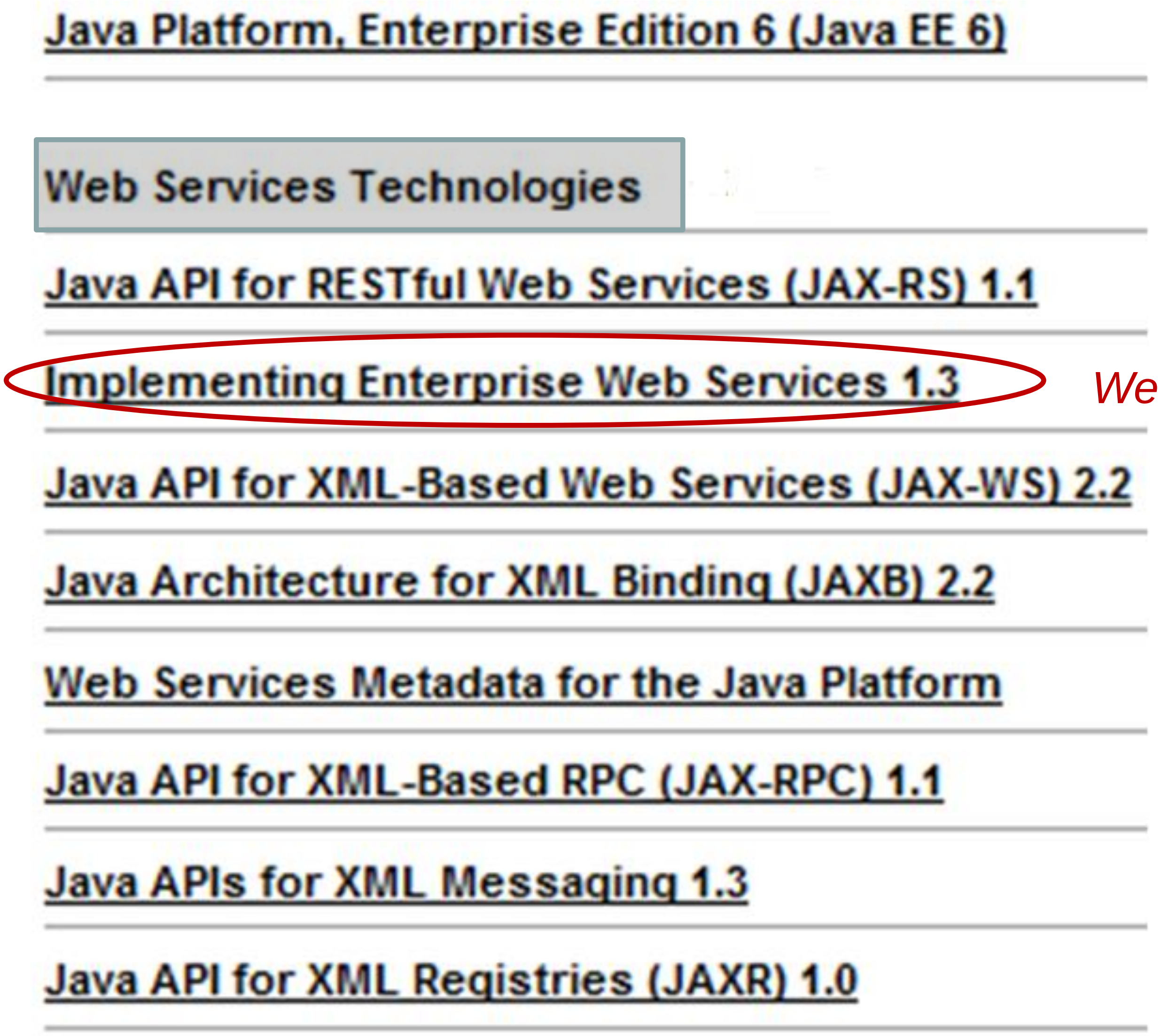
Server

HTTP status code 5xx server error. In this case the request is made for the method not defined in the servlet.

Java Platform, Enterprise Edition

- Java Platform, Enterprise Edition (Java EE) 6 is the industry standard for enterprise Java computing.
- It is a powerful collection of technologies that sit on top of Java Standard Edition (JSE) environment.
- It provides cross-platform compatibility for both application and application server.
- The JEE architecture is managed by Sun Microsystems with the help and support of many active industry partners.
- JEE technology is used to build used to build enterprise application in Java.

Java EE 6 Technologies



Java Platform, Enterprise Edition 6 (Java EE 6)

Web Services Technologies

Java API for RESTful Web Services (JAX-RS) 1.1

Implementing Enterprise Web Services 1.3

Java API for XML-Based Web Services (JAX-WS) 2.2

Java Architecture for XML Binding (JAXB) 2.2

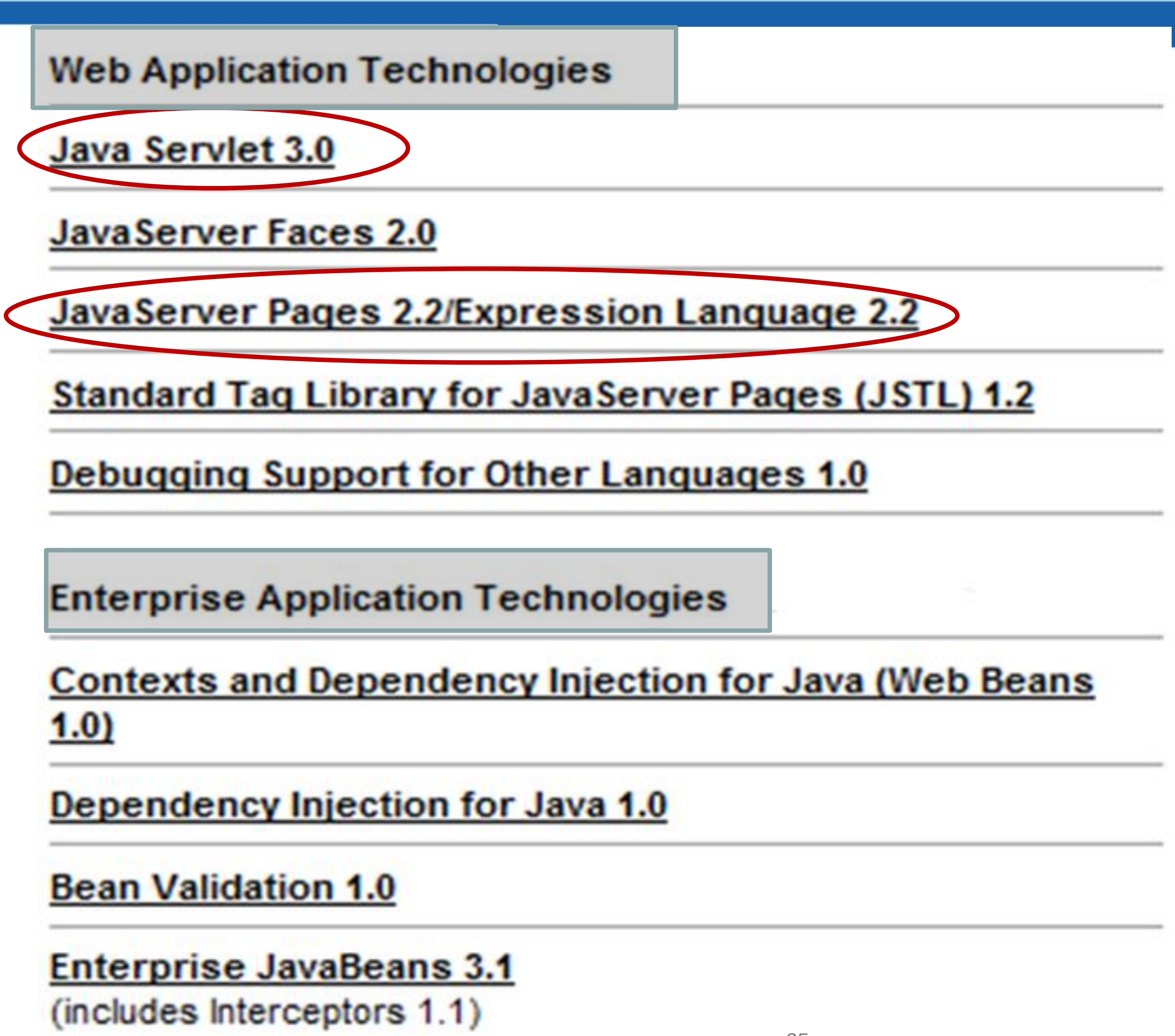
Web Services Metadata for the Java Platform

Java API for XML-Based RPC (JAX-RPC) 1.1

Java APIs for XML Messaging 1.3

Java API for XML Registries (JAXR) 1.0

We will do this



Web Application Technologies

Java Servlet 3.0

JavaServer Faces 2.0

JavaServer Pages 2.2/Expression Language 2.2

Standard Tag Library for JavaServer Pages (JSTL) 1.2

Debugging Support for Other Languages 1.0

Enterprise Application Technologies

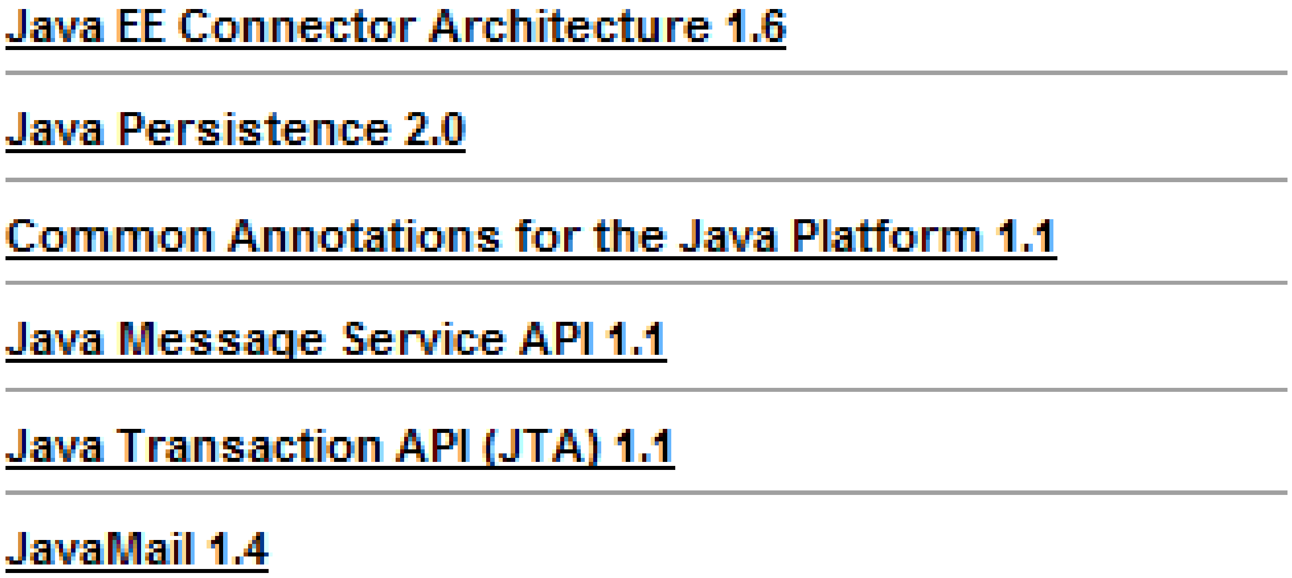
Contexts and Dependency Injection for Java (Web Beans 1.0)

Dependency Injection for Java 1.0

Bean Validation 1.0

Enterprise JavaBeans 3.1
(includes Interceptors 1.1)

We will do these



Java EE Connector Architecture 1.6

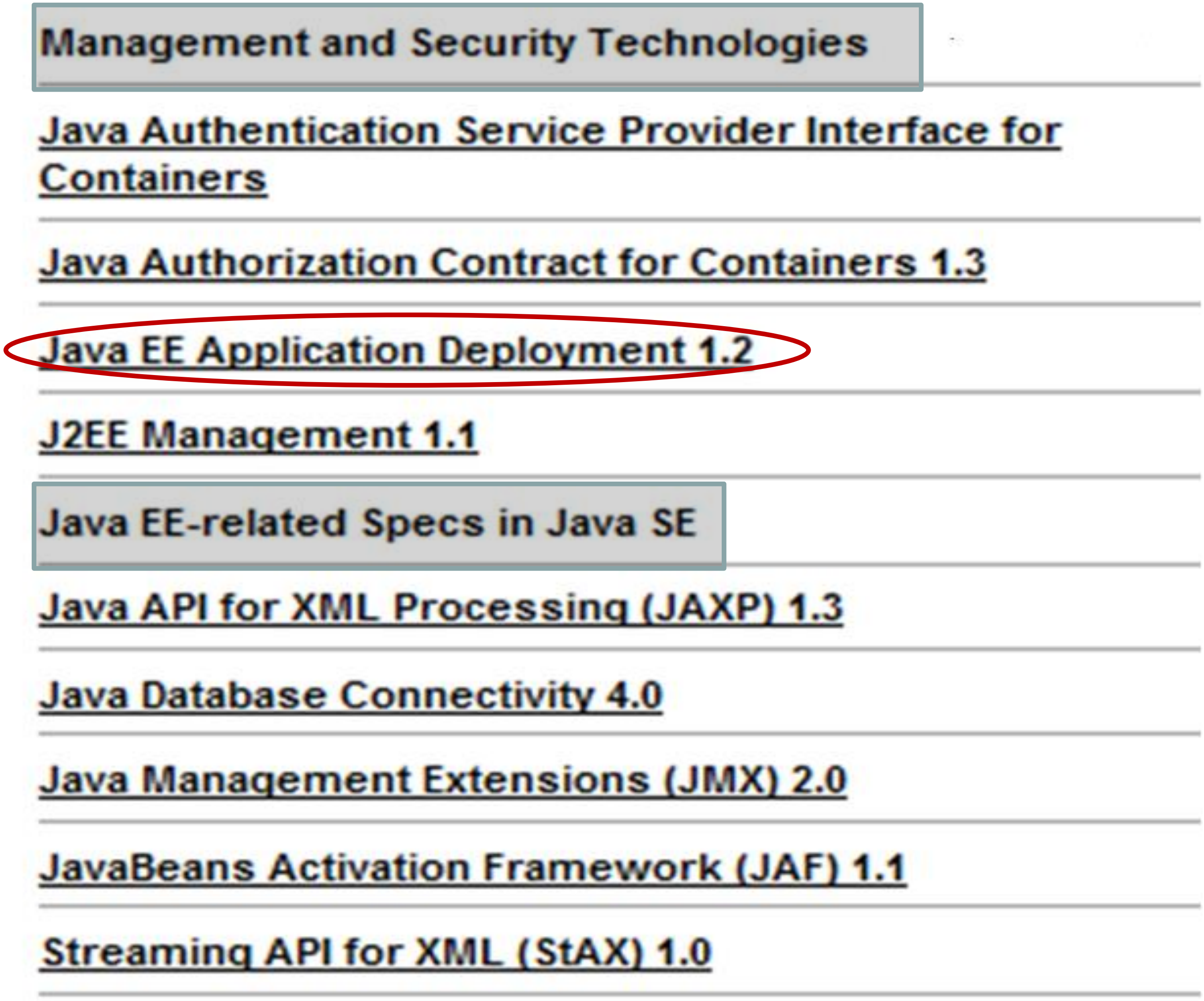
Java Persistence 2.0

Common Annotations for the Java Platform 1.1

Java Message Service API 1.1

Java Transaction API (JTA) 1.1

JavaMail 1.4



Management and Security Technologies

Java Authentication Service Provider Interface for Containers

Java Authorization Contract for Containers 1.3

Java EE Application Deployment 1.2

J2EE Management 1.1

Java EE-related Specs in Java SE

Java API for XML Processing (JAXP) 1.3

Java Database Connectivity 4.0

Java Management Extensions (JMX) 2.0

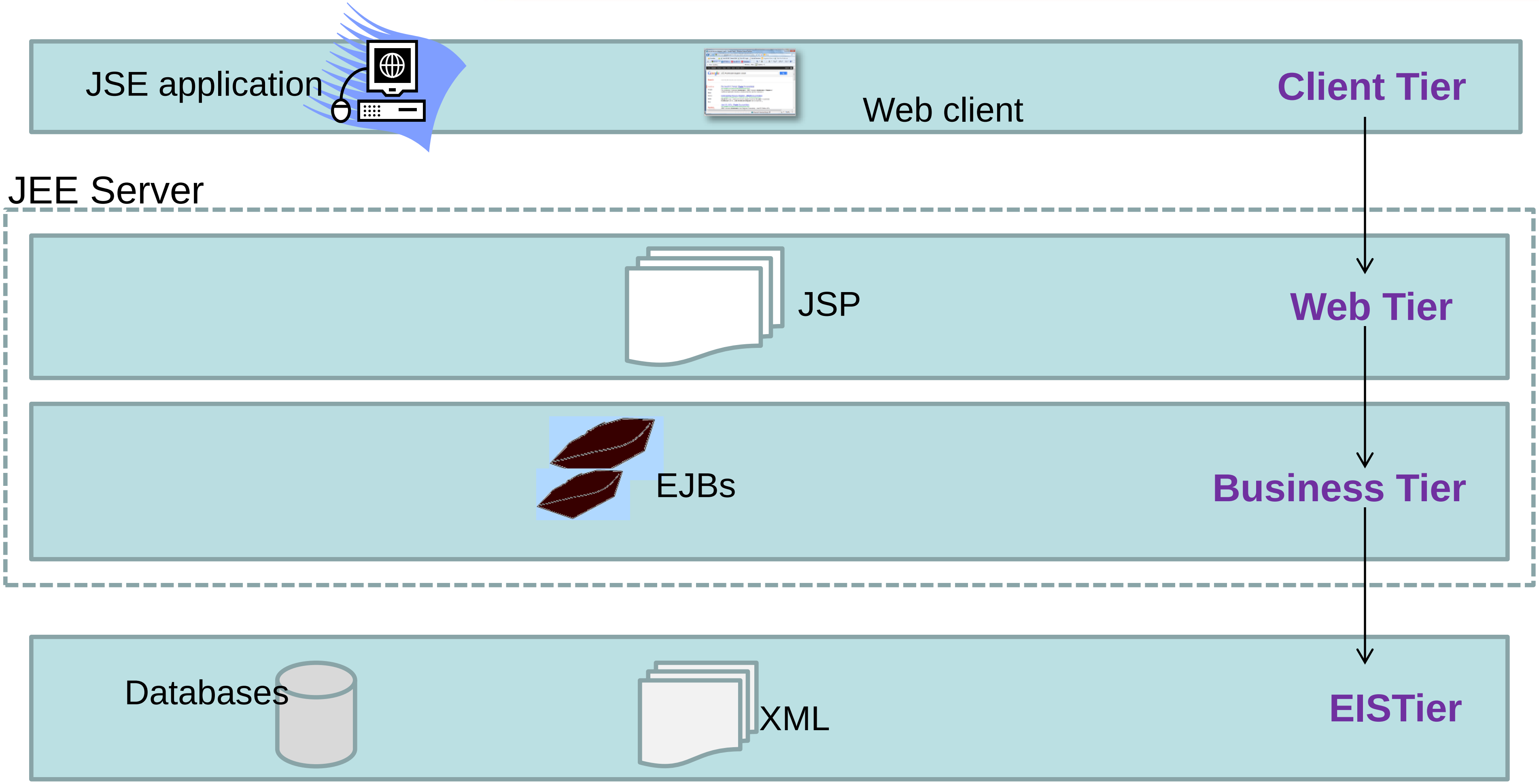
JavaBeans Activation Framework (JAF) 1.1

Streaming API for XML (StAX) 1.0

We will do these

Can you identify the technologies we already covered in this list?

JEE Architecture



JSE application

Web client

Client Tier

JEE Server

JSP

Web Tier

EJBs

Business Tier

Databases

XML

EISTier

Multi-tiered Applications

JEE Container Services

- Security
  - Allows easy configuration of secure features for a web component or enterprise bean so that they are accessed only by authorized users.
- Transaction
  - Transaction model lets allows to configure features for a group of methods which form a single transaction.
- JNDI lookup services
  - common way to look up to application components and other services
- Remote connectivity model
  - Allows clients to communicate to remote enterprise beans as if it were in the same virtual machine.

Java web applications

- Java web applications are made up of JavaServer Pages (JSPs), Servlets and static resources such as HTML pages, images etc
- The Web container is used to host these web components.
- We will start with Servlets