SOEN 387 Web-based Enterprise Application Design

Stuart Thiel

Concordia University
Department of Computer & Software Engineering

Fall, 2015

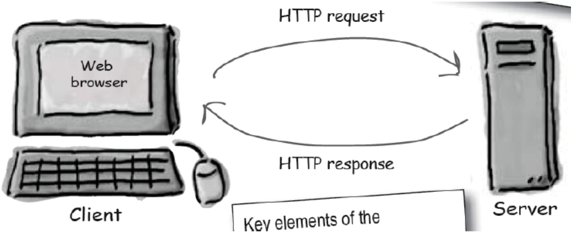
Outline

Review of HTTP

Servlets

Contexts

Talking to Web Servers



[1]

- clients send requests
- servers send responses
- all with the HTTP protocol

[1] "Head First Servlets & JSP", p.10

Web Requests

- header information
  - method
  - cookies
  - location
  - language
  - ...
- body
  - extended parameters/data
  - only used with certain request methods
  - used for the actual web-page response

Request Methods

- **GET**
  - Request info
- **POST**
  - Ask server to **post** the submitted data. . .
- HEAD
  - Like GET, but just asking for the headers, not the body
  - e.g. get last time changed for large file was changed
- TRACE
  - Ask server to repeat what you just asked (debugging)
- PUT
  - Put the submitted data at the given URL
- DELETE
  - Delete whatever is at the requested URL
- OPTIONS
  - Specify HTTP Methods that this URL responds to
- CONNECT
  - Stay connected for commet/tunneling/weird stuff

Web Servers

- Web Servers or HTTP Servers serve static content
- Dynamic content too
- A Web Server can serve another server's content
- We want to serve up Servlets

Servlets

- The business-end of this course
- A type of web application
    - Other types?

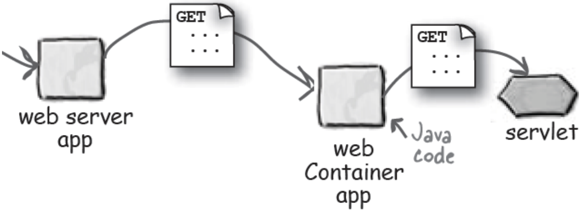
What Are Servlets

- Java classes
- The entry-point follows a specific interface
- Some configuration to smooth stuff out

Where Do Servlets Live?

- Inside a Servlet Container
- Containers take care of housekeeping
- Life-cycle management
  - Initialization
  - Request Setup
  - Session Management
  - Decomissioning

Servlets in Context



[2]

- Tomcat will be our Servlet Container
- Reference implementation

[2] "Head First Servlets & JSP", p.39

Servlets and Java Server Pages

- Using servlets and POJOs, we will make web apps
- We will also use Java Server Pages (JSPs)
  - Like PHP, but for Java
  - Actually compiled to Java servlets at runtime

First Tutorial

- Hello World
- Eclipse 4.x + WTP and Tomcat 7 or 8
- Learn to set up eclipse
- Learn to set up tomcat
- Learn to deploy
- Learn to forward to a JSP

Contexts

- When you talk to someone, there are contexts
- When someone asks you a simple question?
  - What do you need to know only for question and response?
  - What do you keep track of for whole conversation?
  - What do you keep track of between conversations?
  - What do you know irrespective of the conversation?

Contexts in Servlets

- Application
- Session
- Request

Context Boundaries

- Application: Start and Stop
- Session: ?
- Request: Send Request, Get Response

Session Boundaries

- Starts when you access the Servlet
- When does it stop?

Security and Sessions

- Can you abuse a session to hack someone?
- Protecting Login
- Protecting Logout

Sanitize EVERYTHING

- Sanitize Sessions on Login
- Keep some pre-auth data explicitly

A Quick Note on Concurrency

- Does the request context ever share? No.
- How about the Application? Of course.
- And the Session?

What Does A Servlet Look Like File-wise?

- https://tomcat.apache.org/tomcat-8.0-doc/appdev/deployment.html
- *.html, *.jsp, etc.
- /WEB-INF/web.xml
- /WEB-INF/classes/
- /WEB-INF/lib/

Some Suggestions for file-size?

- `https://tomcat.apache.org/tomcat-8.0-doc/appdev/source.html`
- web/ webapp document root
    - under which you find WEB-INF, etc
- src/ keeps the source for your webapp nearby
- docs/ keep any documentation nearby
- Put src/DOCs in SVN (repo)!
- Put document root, save classes in SVN!

How to access JSPs

- I put JSPs where they cannot be directly accessed
  - inside WEB-INF/JSP
  - We'll talk about why this can be useful

Finish with a Look at Servlet API

- `https://tomcat.apache.org/tomcat-8.0-doc/servletapi/`
- look at HttpServlet