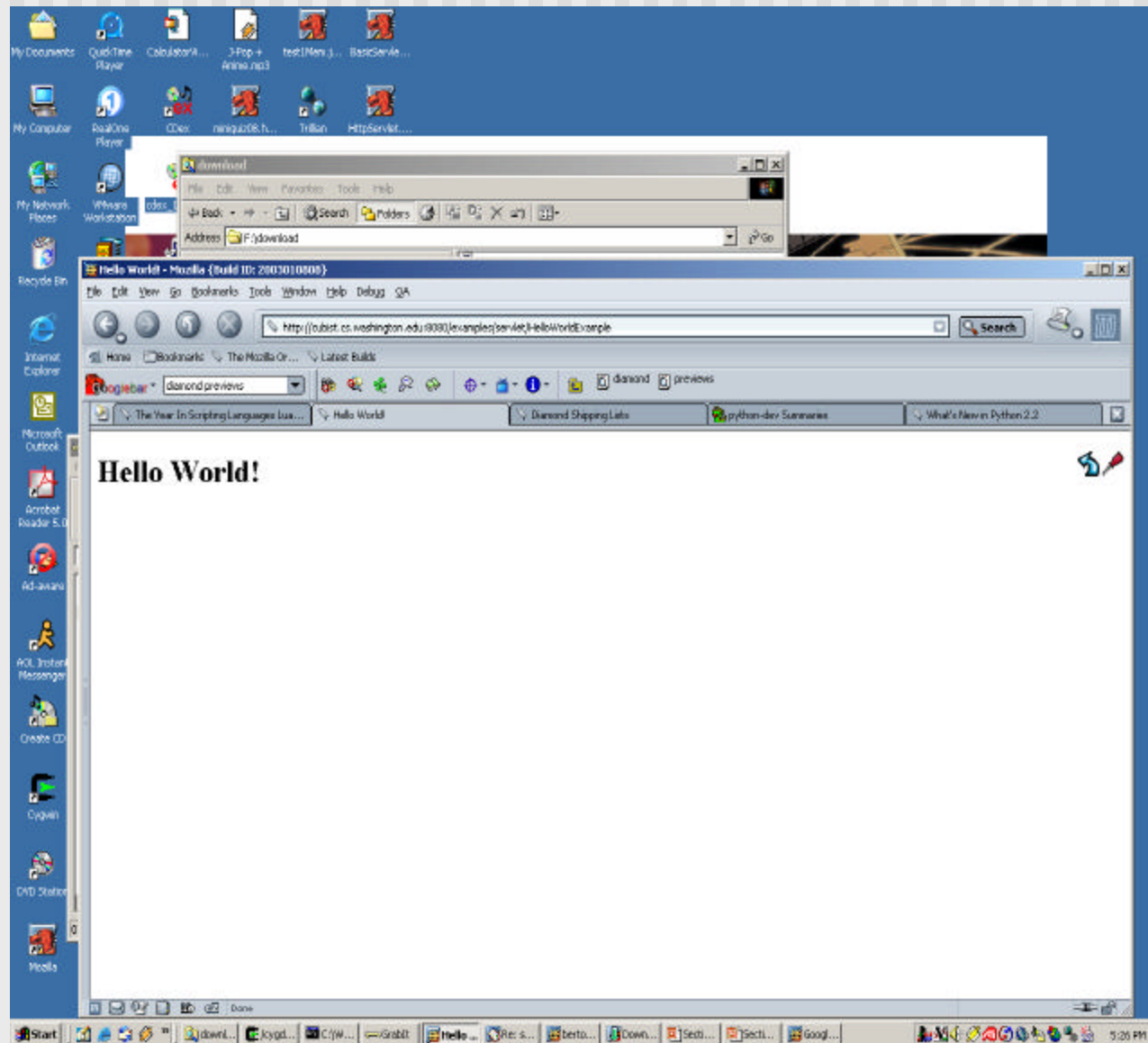# CSE 403

## The Mythical Servlet

# Goals

- Introduce basic servlet terminology
- Formally introduce homework 2
- Walkthrough of build & deployment
- Group discussion of *MMM*

# What is a Servlet?

- Server-side Java Program
- Equivalent to a CGI app
- Efficient
- Portable
- Powerful
- Stand-alone or extension
  - We're using Tomcat on cubist.cs.washington.edu

# Demo Example

# Where do I find this stuff?

```
/*For generic , protocol independent
servlet functionality*/

import javax.servlet.*;

/*For servlets that will use the http
protocol (WWW Based)*/

import javax.servlet.http.*;
```

Tutorial @

http://java.sun.com/webservices/docs/1.0/tutorial/doc/Servlets.html

API @

http://java.sun.com/j2ee/sdk_1.3/techdocs/api/javax/servlet/package-summary.html
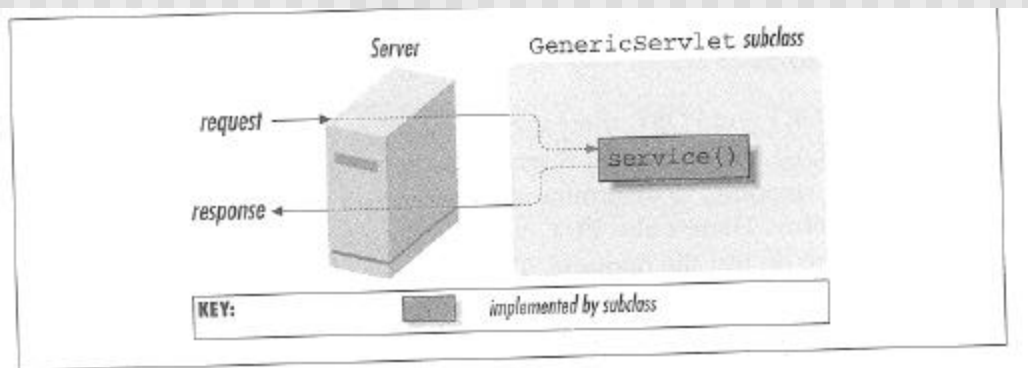
# GenericServlet



Figure 2-1. A generic servlet handling a request

Image from *Java Servlet Programming* by Jason Hunter

- Protocol Independent
- Override `service()` when extended
- Must write one's own means of handling requests

# HttpServlet

- Extends `GenericServlet()`
- Overide `doGet(…)` & `doPost(…)`
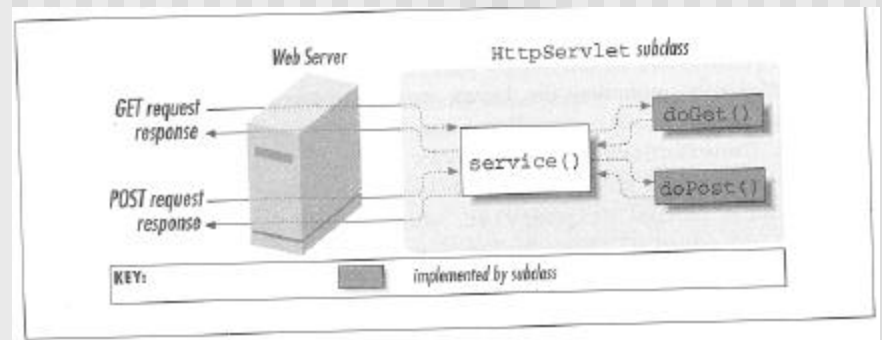- Already understands http
- Majority of servlets extend



Figure 2-2. An HTTP servlet handling GET and POST requests

Image from *Java Servlet Programming* by Jason Hunter

# HttpServlet Skeleton

```java
import javax.servlet.*;

import javax.servlet.http.*;


public class ServletSkeleton extends HttpServlet {
    public void doGet(HttpServletRequest req,
                            HttpServletResponse res)
            throws ServletException, IOException {}

    public void doPost(HttpServletRequest req,
                            HttpServletResponse res)
            throws ServletException, IOException {}
}
```
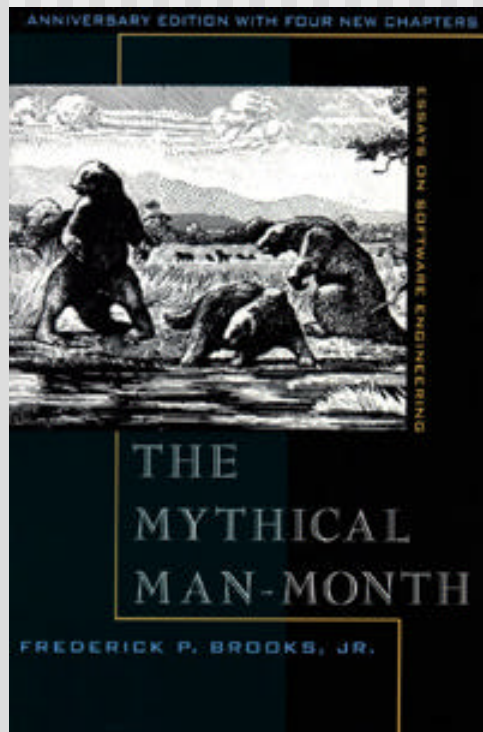
# GET vs POST

## GET

- retrieving information
- html/images
- parameters listed in URL

## POST

- send information
- uploads/forms
- parameters hidden within header

# HW2 Prep Walkthrough

# Did you do your reading?



What project inspired Brooks to write this book?

# Programming :) / :(

What joys does computer programming have?

What woes?

# Project Failure

What are some common reasons
for project failure?

What is Brooks'?

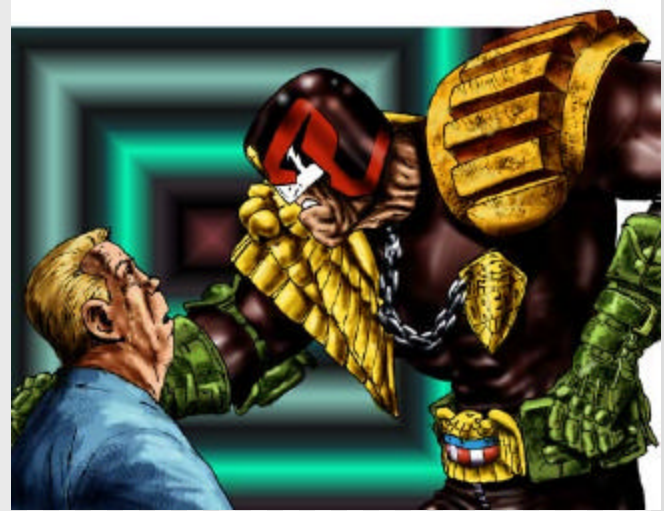All programmers are optimists:

"All will go well"

# Scheduling

1/3 design

1/6 coding

1/4 component testing

1/4 system testing

# Brooks' Law



**Adding manpower to a late software project will make it later.**

Why?

# The Surgeon Model



Fig. 3.1 Communication patterns in 10-man programming teams