

# Web Architecture, 3-Tier Apps

PDBM 15.1, 15.6

Dr. Chris Mayfield

Department of Computer Science  
James Madison University

Mar 22, 2022



# Definitions

What is **The Internet**?

A global system of interconnected **computer networks** that use the Internet protocol suite (TCP/IP) to link billions of devices worldwide.

<https://en.wikipedia.org/wiki/Internet>

What is the **World Wide Web**?

An open source **information space** where documents and other web resources are identified by URLs, interlinked by hypertext links, ...

[https://en.wikipedia.org/wiki/World\\_Wide\\_Web](https://en.wikipedia.org/wiki/World_Wide_Web)

# How the web works

In short:

- ▶ Web servers “store” documents/etc
- ▶ Web clients (browsers) request them
- ▶ The Internet provides the file transfer

[https://en.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol)

Request:

```
GET /index.html HTTP/1.1  
Host: www.example.com  
...
```

Response:

```
HTTP/1.1 200 OK  
Date: Mon, 23 May 2005 22:38:34  
...
```

# How addresses work

See [https://en.wikipedia.org/wiki/Uniform\\_resource\\_locator](https://en.wikipedia.org/wiki/Uniform_resource_locator)

<https://www.google.com/search?q=hello&ie=UTF-8>

protocol      host      path      query

Notes about the *query string*:

- ▶ Key=value pairs
- ▶ Begins with ?
- ▶ Separated by &
- ▶ URL encoding

# HTTP is stateless

When you browse the web:

- ▶ Client asks for URL
- ▶ Server returns a file
- ▶ Otherwise disconnected

How to *simulate* an application?

- ▶ URL parameters (via GET/POST)
- ▶ Server-side sessions (via cookies)

(we will mainly focus on parameters)



# HTML forms

```
<form action="query.jsp" method="get">  
  <input type="text" name="title">  
  <input type="submit" name="ins" value="Insert" />  
  <br>  
  <a href="query.jsp?del=%7BGo+Dukes!%7D">Delete</a>  
</form>
```

## Action

- ▶ Where to submit the form

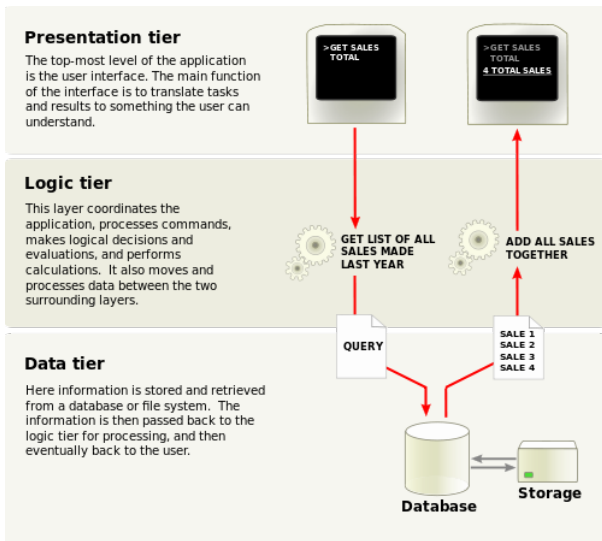
## Method

- ▶ GET = parameters visible in URL query string
- ▶ POST = parameters in HTTP request headers

## Demo: Chrome DevTools

<https://developers.google.com/web/tools/chrome-devtools>

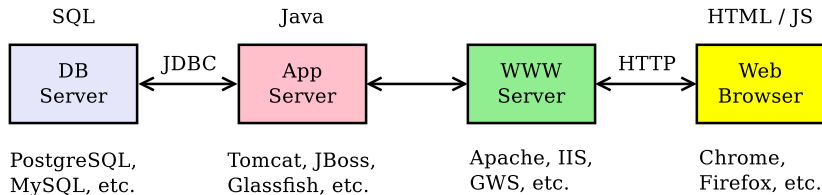
# Three-tier architecture



Source: [https://en.wikipedia.org/wiki/Multitier\\_architecture](https://en.wikipedia.org/wiki/Multitier_architecture)



# Languages and software



## Design principles

- ▶ Separation of concerns (data, logic, gui)
- ▶ Development: all tiers run on same machine
- ▶ Deployment: multiple machines for each tier

# Example: Java

JSP = template for Java servlet

- ▶ See [https://en.wikipedia.org/wiki/Java\\_Servlet](https://en.wikipedia.org/wiki/Java_Servlet)
- ▶ Goal: generate dynamic HTML response

```
<html>
<body>
Hello! The time is now <%= new java.util.Date() %>
<%
    // this block of code is called a scriptlet
    out.print("<br>Your machine's address is ");
    out.println(request.getRemoteHost());
%>
</body>
</html>
```

# Example: PHP

PHP stands for “PHP Hypertext Preprocessor”

- ▶ [https://en.wikipedia.org/wiki/Recursive\\_acronym](https://en.wikipedia.org/wiki/Recursive_acronym)

Scripting language for generating HTML pages

```
<html>  
<body>  
Hello, today is <?php print(Date("l F d, Y")); ?>.  
</body>  
</html>
```

Language Reference:

- ▶ <https://www.php.net/manual/en/langref.php>

## Example: Python



# Flask

web development,  
one drop at a time

Hello world application:

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'
```

# Rendering Templates

Hello with a name:

```
from flask import render_template

@app.route('/hello/')
@app.route('/hello/<name>')
def hello(name=None):
    return render_template('hello.html', name=name)
```

hello.html template:

```
<!doctype html>
<title>Hello from Flask</title>
{% if name %}
    <h1>Hello {{ name }}!</h1>
{% else %}
    <h1>Hello, World!</h1>
{% endif %}
```

## Other frameworks



<https://www.djangoproject.com>



<https://nodejs.org>



<http://rubyonrails.org>