Introduction to Web Applications

Jeff Offutt

http://www.cs.gmu.edu/~offutt/

SWE 432
**Design and Implementation of**
Software for the Web

Web Applications

- A web application uses enabling technologies to
  1. make web site contents dynamic
  2. allow users of the system to implement business logic on the server
- Web applications let users affect state on the server

A web application is a program deployed on the web

An enabling technology makes web pages interactive and responsive to user input

Web Applications
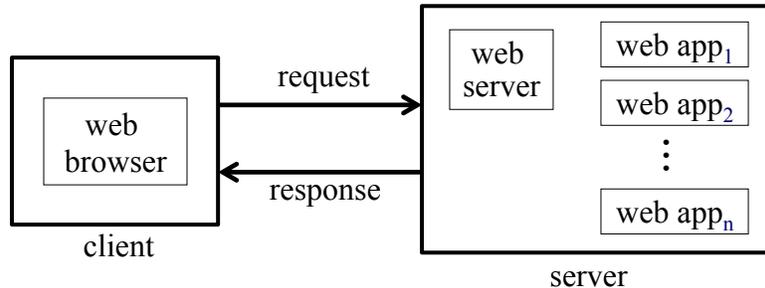
	request		web server	web app$_1$
web browser				web app$_2$
	response			$\vdots$
				web app$_n$

client

server

Web applications are constructed from diverse, distributed, and dynamically generated web components

Web Application Failures

Web applications have many problems

- Oct 2004: Paypal waived transaction fees for an entire day because of a service outage after an upgrade
- Aug 2006 and July 2008: Amazon failure caused businesses to lose information and revenue
- Sep 2011: 2 hours downtime of Target's web site and its 1 day intermittently functioning delayed and cancelled customers' orders
- Mar 2012: The BodyShop BOGO turned to extra discount
- September 2012: Bank of America, JPMorgan Chase, Citigroup, US Bank, Wells Fargo, and PNC suffered online attacks

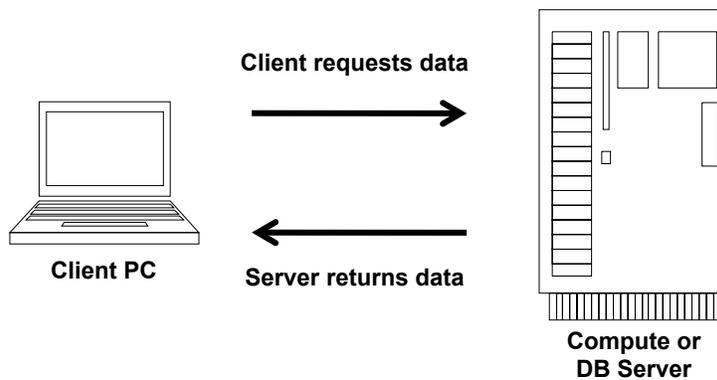# Traditional Computing Use

A user works with software on her computer

Client – Server Computing

A user works with software or data on a separate computer called a *server*

Client requests data

Client PC

Server returns data

Compute or DB Server
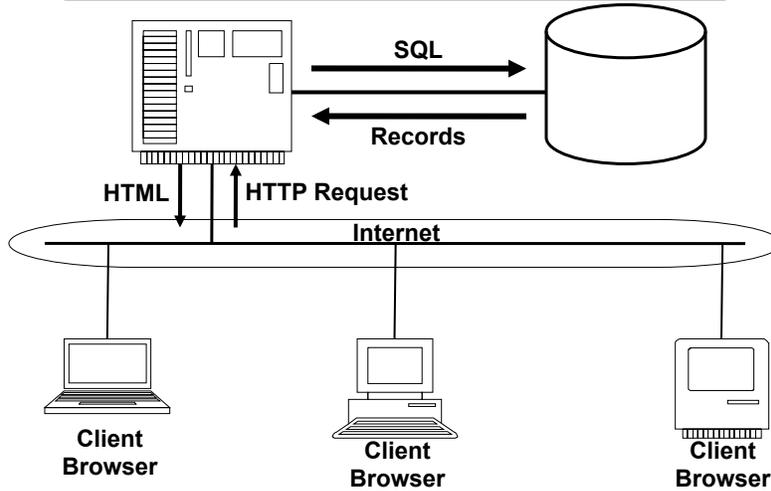
Web Applications

Many users work with servers and databases that can be reached through the Internet with the HyperText Transfer Protocol
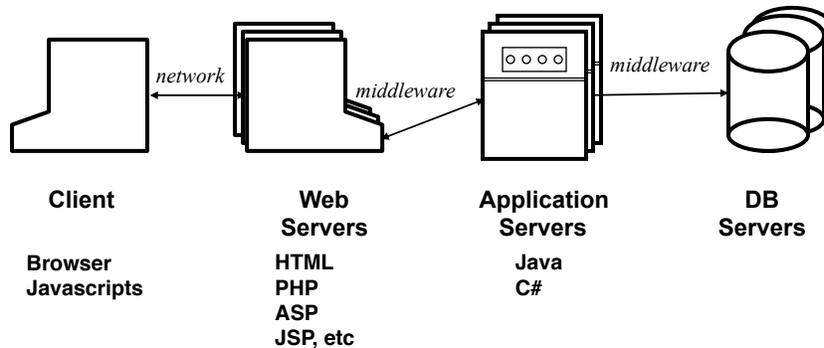
SQL

Records

HTML   HTTP Request

Internet

Client Browser

Client Browser

Client Browser

© Offutt, 2011
7

N-Tier Web Architecture

Large web applications run on many computers that have to coordinate with each other.

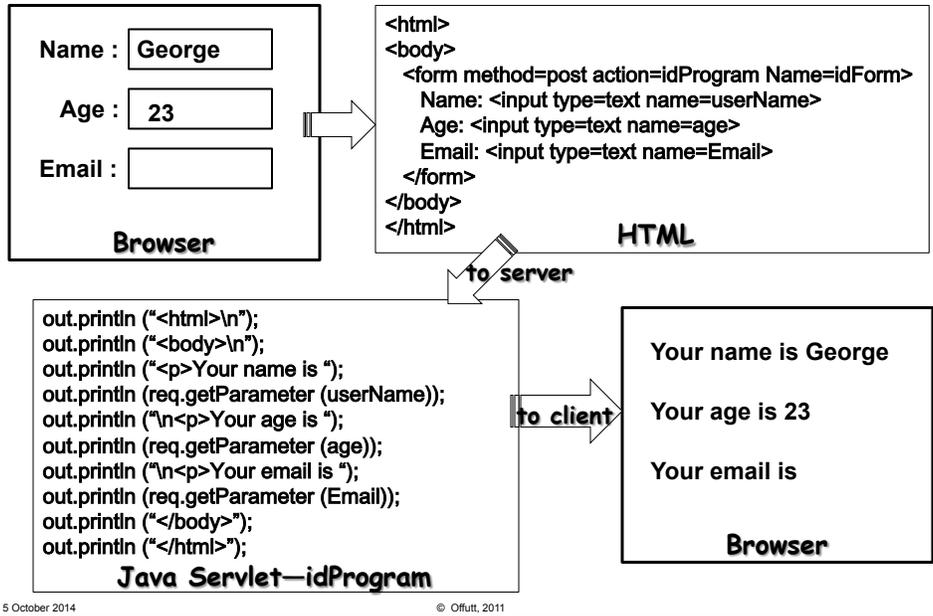**Amazon and Netflix have *thousands* of servers.**

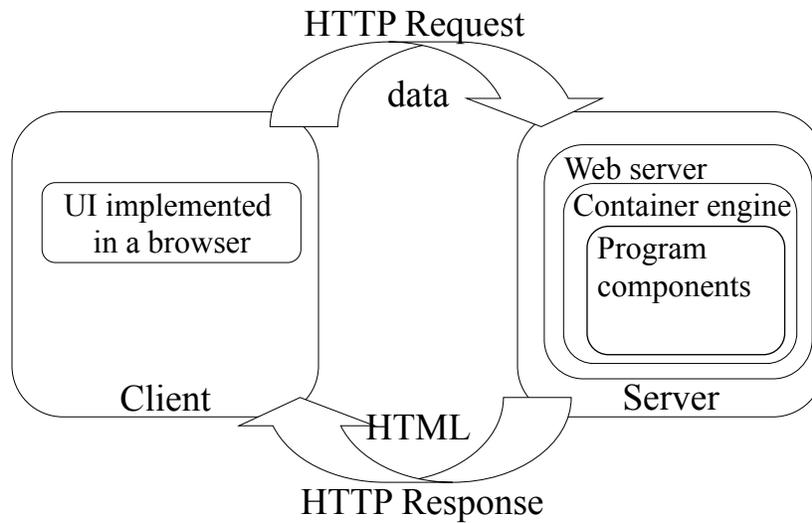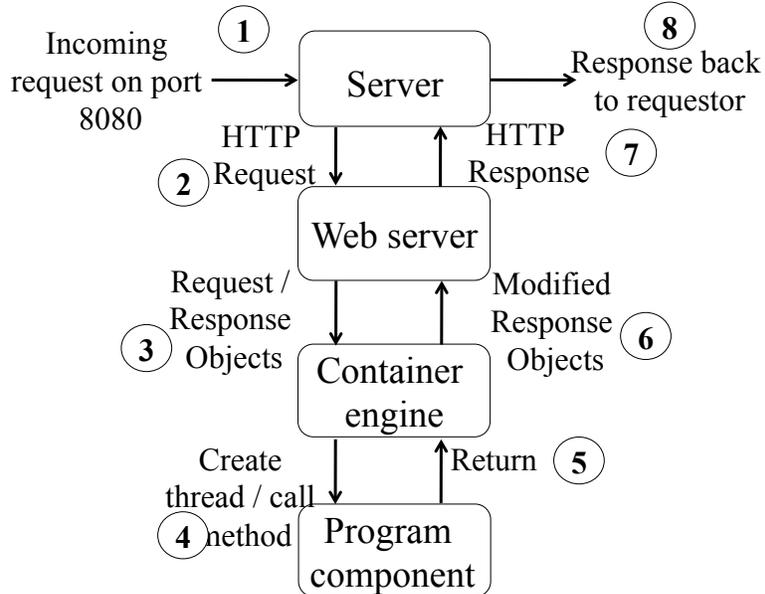*network*   *middleware*   *middleware*

Client

Web Servers

Application Servers

DB Servers

Browser Javascripts

HTML PHP ASP JSP, etc

Java C#

© Offutt, 2011
8

4

How the Software Works

Name : George

Age : 23

Email :

Browser

```
<html>
<body>
  <form method=post action=idProgram Name=idForm>
    Name: <input type=text name=userName>
    Age: <input type=text name=age>
    Email: <input type=text name=Email>
  </form>
</body>
</html>
```
HTML

to server

```
out.println ("<html>\n");
out.println ("<body>\n");
out.println ("<p>Your name is ");
out.println (req.getParameter (userName));
out.println ("\n<p>Your age is ");
out.println (req.getParameter (age));
out.println ("\n<p>Your email is ");
out.println (req.getParameter (Email));
out.println ("</body>");
out.println ("</html>");
```
Java Servlet—idProgram

to client

Your name is George

Your age is 23

Your email is

Browser

Server Side Processing

HTTP Request

data

UI implemented
in a browser

Web server

Container engine

Program
components

Client

Server

HTML

HTTP Response

Execution Overview

Incoming request on port 8080 ① → **Server** → ⑧ Response back to requestor

② HTTP Request

Web server

⑦ HTTP Response

③ Request / Response Objects

Container engine

⑥ Modified Response Objects

④ Create thread / call method

Program component

⑤ Return

Web Software Container Engine
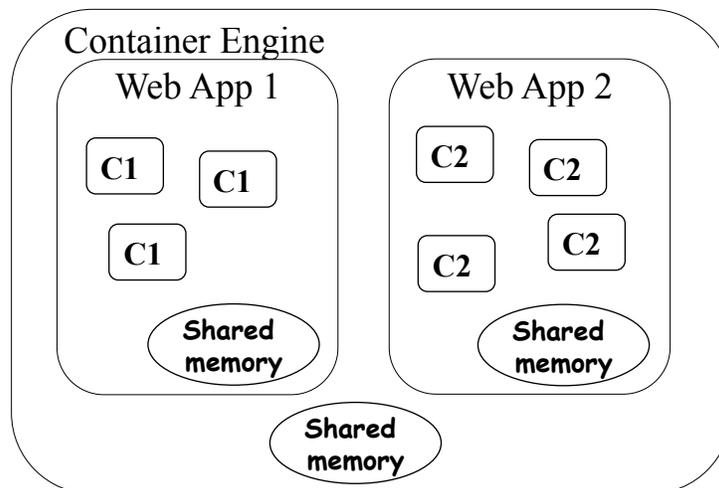
Container Engine

Web App 1

C1   C1   C1

Shared memory

Web App 2

C2   C2   C2   C2

Shared memory

Shared memory

Web applications are constructed from diverse, distributed, and dynamically generated web components
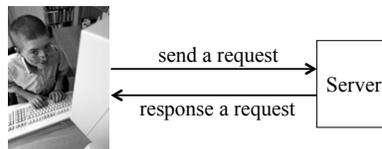
Session Management

HTTP client-server communication is *connnectionless*

- As soon as the request is made and fulfilled, the connection is terminated
- Communication is simple and resistant to network problems

send a request

response a request

Server

Session Management

How can servers keep track of state of different clients?

1. *Session* : A single coherent use of the system by the same user
   - Example : shopping carts
2. *Cookies* : A string of characters that a web server places on a browser's client to keep track of a session
   - Usually used as an index into a table (*dictionary*) on the server
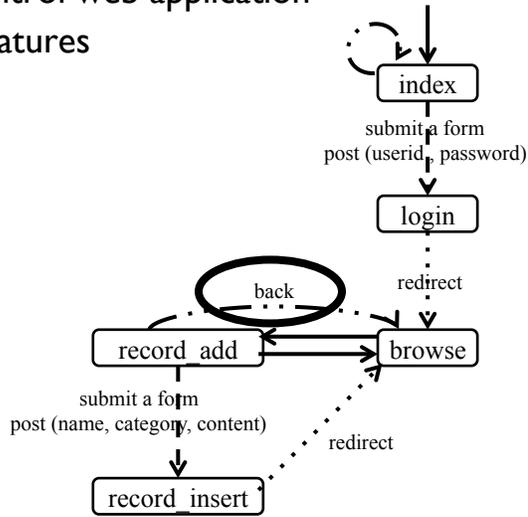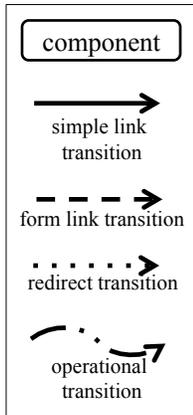   - Most dictionaries expire after a period of time (15 to 30 minutes)

Additional mechanisms

- URL rewriting
- Hidden form control

We will come back to this later …

Additional Web Features

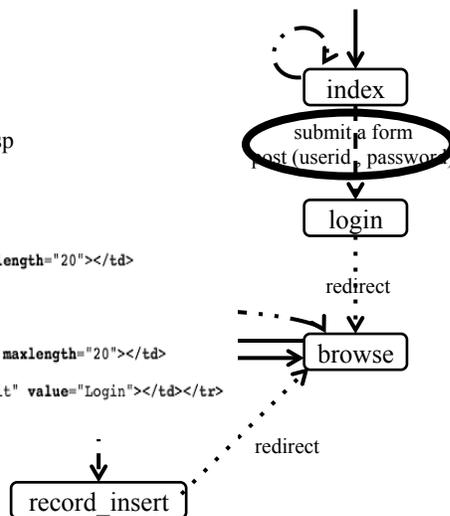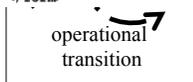User's ability to control web application
via web browser features

component

→ simple link transition

--→ form link transition

⋯⋯→ redirect transition

operational transition

index

submit a form
post (userid, password)

login

back

redirect

record_add → browse

submit a form
post (name, category, content)

redirect

record_insert

15

Additional Web Features

Identifying web application resources with URLs

```
20  <!-- Form for displaying login info -->     userlogin.jsp
21  <form method="post" action="login.jsp">
22  <!-- Table -->
23  <table cellspacing="0" cellpadding="3" border="0">
24  <!-- Displays username fields -->
25  <tr>
26    <td>Username:</td>
27    <td><input type="text" name="userid" size="10" maxlength="20"></td>
28  </tr>
29  <!-- Displays password fields -->
30  <tr>
31    <td>Password:</td>
32    <td><input type="passwd" name="password" size="10" maxlength="20"></td>
33  </tr>
34  <tr><td colspan="2" align="center"><input type="submit" value="Login"></td></tr>
35  </table>
36  </form>
```
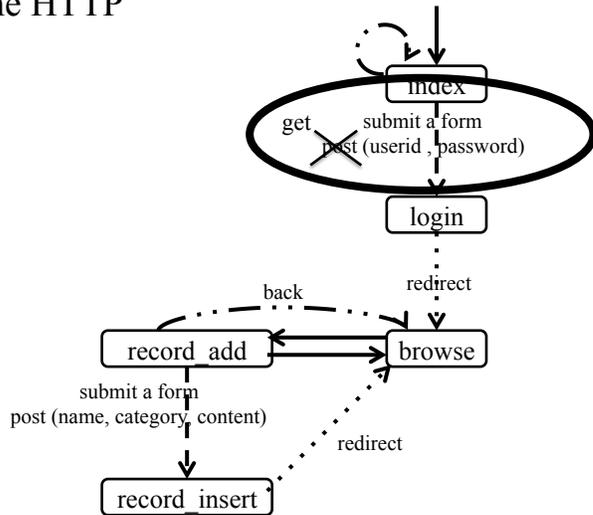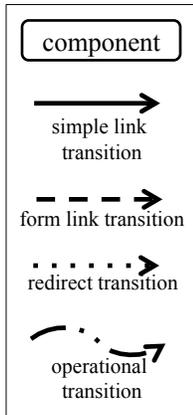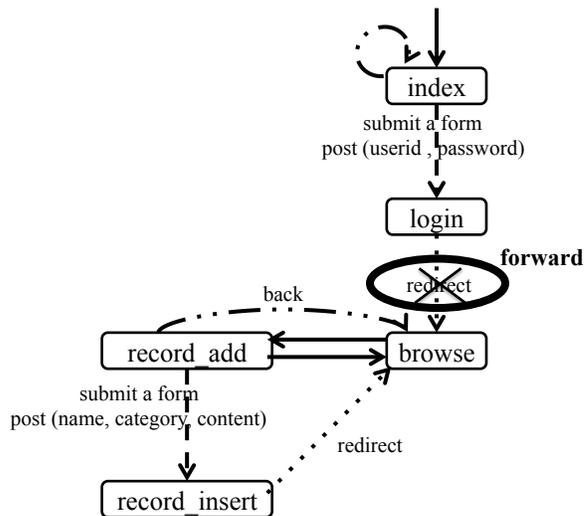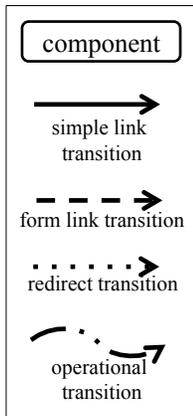
operational transition

index

submit a form
post (userid, password)

login

redirect

browse

redirect

record_insert

16

8

Additional Web Features

Communication among web components depending on requests through the HTTP

component

simple link transition

form link transition

redirect transition

operational transition

index

get   submit a form
post (userid , password)

login

redirect

back

record_add → browse

submit a form
post (name, category, content)

redirect

record_insert

Additional Web Features

Control connections: forward, include, redirect

component

simple link transition

form link transition

redirect transition

operational transition

index

submit a form
post (userid , password)

login

forward

redirect

back

record_add → browse

submit a form
post (name, category, content)

redirect

record_insert

Additional Web Features

State scope of web components
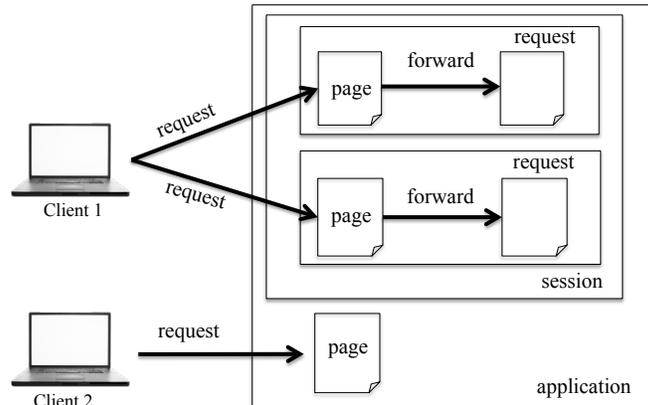
Scope and Handling State in JSPs, slides from SWE 642, reproduced with permission from J. Offutt

---

Summary Web Programming

- The major difference is deployment
  - Software is deployed across the Web using HTTP
  - Other deployment methods include bundling, shrink-wrapping, embedding, and contracting
- New software technologies
- New conceptual language constructs for programming
  - Integration
  - Data management
  - Control connections

> These differences affects every aspect of how to engineer high quality software