Web Application Development

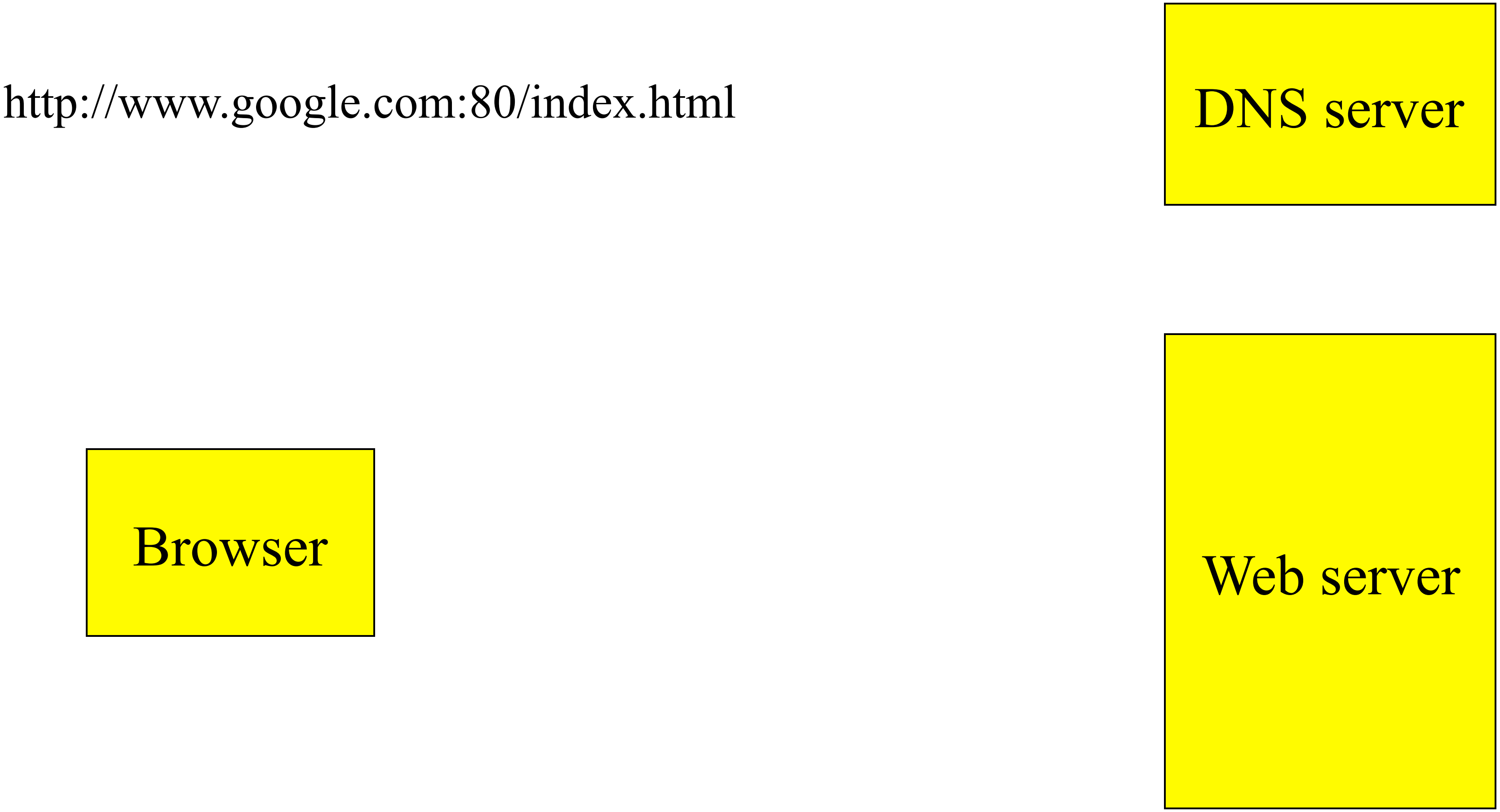
HTTP: Hypertext Transfer Protocol

Charlie Garrod

School of Computer Science

Carnegie Mellon University

The web connection



http://www.google.com:80/index.html

DNS server

Browser

Web server

Standard application network ports

- Telnet (23), SMTP (25), MySQL (3306)
- For web servers:
  - Deployment: HTTP (80), HTTPS (443)
  - Development: HTTP (8000 or 8080), HTTPS (8443)

HTTP requests and responses

- Request:

```
<METHOD>   <URI>   <VERSION>
<HEADER LINE>
<HEADER LINE>
…
<HEADER LINE>
<BLANK LINE>
<OPTIONAL MESSAGE BODY>
```

- Response:

```
<VERSION>   <STATUS CODE>   <STATUS MESSAGE>
<HEADER LINE>
<HEADER LINE>
…
<HEADER LINE>
<BLANK LINE>
<OPTIONAL MESSAGE BODY>
```

HTTP request methods

- Safe methods are not supposed to have side-effects:
  - GET, HEAD, TRACE, and OPTIONS
- Idempotent methods can have side-effects:
  - PUT and DELETE
- Update method:
  - POST

To prepare for class:

- Know the basic purpose of each HTTP/1.1 method
- Understand GET and POST in detail
  - What data is contained in a typical request line? Header lines? The message body?
    - Understand *query parameters* and how they are transmitted
  - What are GET and POST typically used for?
  - How do browsers treat GET and POST requests differently?
- Know how to interact with servers manually using telnet

Some resources:

- 15-213 Web Services lecture
  - http://www.cs.cmu.edu/afs/cs/academic/class/15213-s16/www/lectures/22-netprog2.pdf
- The World Wide Web Consortium (W3C)
  - http://www.w3.org/Protocols/rfc2616/rfc2616.html