Table of Contents

1. Networks
   - Internet
   - Resources
   - Client–Server Model

2. World Wide Web
   - Overview
   - HTTP Protocol
     - HTTP Commands
     - HTTP Status Codes
Open Systems Interconnection Model

- **Conceptual model**
  - not realized directly
  - various implementations

- **Layered structure**

- **Each layer**
  - uses below layers
  - serves above layers
  - has well-defined function
Internet Protocol Suite

TCP/IP model

Application layer
Transport layer
Internet layer
Network Interface layer

TCP/IP protocol suite

Telnet   FTP    SMTP   DNS    RIP    SNMP
TCP      UDP    IGMP   ICMP
IP       IPSEC
Ethernet  Token Ring  Frame Relay  ATM
Transmission Control Protocol / IP

- packet-switching
- reliable
- ordered
- error–checking

Applications
- World Wide Web (HTTP)
- email (POP, IMAP, SMTP)
- FTP
- SSH
- DHCP
- DNS
Resource

Static content
- Text
- HTML
- image
- audio / video
- binary file (pdf, doc, etc.)
- ...

Dynamic content
- Service
  - SSH
  - FTP
  - HTTP
- Generated content
  - CGI
  - PHP, Python
  - JSP, ASP
  - REST / JSON
Unified Resource Identifier

scheme://[user:password@]host[:port]/[/]path[?query][#fragment]

Unified Resource Locator
- Location based identification
- Fits to computer networks
- Usage
  - web address
  - file transfer
  - remote login
  - database access

Unified Resource Name
- Name based identification
  - ISBN books
  - ISSN journals
  - ISAN audiovisual works
URL Examples

- http://www.origo.hu/index.html
- http://www.w3schools.com/
- imap://imap.iit.uni-miskolc.hu:143
- imaps://imap.iit.uni-miskolc.hu:993
- smtp://smtp.iit.uni-miskolc.hu:25
- ftp://ftp.iit.uni-miskolc.hu
- jdbc:mysql://localhost:3306/sakila?profileSQL=true
Hosts Address

IPv4
- 32bit = 4 byte
- xxx.xxx.xxx.xxx
- Unique address (NAT)

IPv6
- Compatibility with IPv4
- 128bit = 16byte
- Unique address

Subnets
- Net Mask
- 32 bit
- 255.255.255.0
- 192.168.2.1/24

Network Address Translation
- Hide subnets
- Request from the Internet ✓
- Service to the Internet X
Domain Name System

- IP address ↔ domain name
- Hierarchical structure
- "." is the root
- world–wide
- decentralized
- DNS Servers
- Hosts belongs to one domain
- Domains contains multiple hosts

Internet Assigned Numbers Agency

Top Level Domain
- Generic TLD
  - gov, edu, net,
  - com, org, mil
- Country Code TLD
  - hu, es, ru,
  - au, gb, us

www.iit.uni-miskolc.hu.
www.abc.net.au.
Client–Server Model

- Simple
- Centralized
- Server
  - provides service or resource
  - runs on a publicly available computer
  - waits for requests from arbitrary clients
- Client
  - requests resources from server
  - runs locally
Client–Server Model

Database

Server

Firewall

Internet

Clients

LAN
Client

Thin client

- Simple application
  - Browser
- Representation, visualization
- No business logic
  - Check values
- Low hardware requirements
- Web applications

Thick (fat, rich) clients

- Complex applications
- Lot of functionalities
- Independent from the server
- Higher hardware requirements
- Off-line work
- Installation
  - for each clients
  - ¿ Versioning?
Server – Application

- Application **not** computer!
- Waits for requests
- Specific purpose
  - Web servers
  - Mail servers
  - Application servers
  - Authentication servers
  - FTP servers
  - etc.
- a computer – many server applications
Server – Computer

- Executes server application
- Fix IP Address
- High performance
  - Expensive hardware
  - High upkeep
  - Custom OS
- One or multiple computers
  - GRID
  - Cloud Computing
- Server room, data centers

As a developer,

- you will run a web/application server and a database server locally.
- you will **use** various servers (SCM, CI, Issue tracker, etc.)
- you may deploy your application to **test** and **demo** servers for **presentation** purposes.
- you will **never** want to operate server. That is the job of **system administrators**!
Table of Contents

1. Networks
   - Internet
   - Resources
   - Client–Server Model

2. World Wide Web
   - Overview
   - HTTP Protocol
     - HTTP Commands
     - HTTP Status Codes
History

- Hypertext system
- 1990-91 First web server
- Expansion (’91-’95)
- Commercialization (’95-2000)
- Dot–com bubble (’97-2000)
- Present (since 2000)
  - Ubiquitous
  - Social network
  - Multimedia
  - Semantic web
Architecture

Client

Browser

HTTP

TPC/IP

Server

Apache

HTTP

TPC/IP
Web Server

- Application
- Static IP
- Listen on port (80, 8080, ...)
- Communicate via HTTP

Major providers
Market share of active sites (Netcraft Sept 2015)

1. Apache (≈50.5%)
2. nginx (≈14.3%)
3. Microsoft (≈10.2%)
4. Google (≈7.8%)

Zsolt Tóth (University of Miskolc)
HyperText Transfer Protocol

- Application Layer
- Information Systems
- Requests and responses
- Defines
  - Methods / verbs
  - Status codes
- Designed to exchange HTML documents

HTTPS
- is not a standalone protocol
- uses secure communication
  - Secure Socket Layer
  - Transport Layer Security
HTML

- Plain Text
- Tags
- Standard
- Browsers
  - visualize
  - interpret

```
1 <html>
   <head>
   <!-- Meta data -->
   </head>
3 <!-- Content -->
5 <body>
   <!-- Content -->
7 </body>
</html>
```
Browser

Challenges

- Browsers visualize the same HTML standardized differently.
- So sites have to be optimized to browser (IE)
- Customers have special needs:
  - we want to include videos so please use HTML 5
  - the site has to be compatible with Internet Explorer 7
Browser

Browser Usage (May 2016)

- Chrome: 71.4%
- IE: 5.7%
- Firefox: 16.9%
- Safari: 3.6%
- W3Schools: 1.2%
- Opera: 0.0%
HTTP Protocol

- Application protocol
- Data communication
- Versions
  - HTTP/0.9
  - HTTP/1.0
  - **HTTP/1.1**
  - HTTP 2.0

- Defines
  - Methods
  - Status codes

- Typical clients
  - Browsers
  - Web robots
  - Mobil applications
HTTP Methods

- HEAD
- GET
- POST
- PUT
- DELETE
- TRACE
- OPTIONS
- CONNECT
- PATCH
GET, HEAD

GET
- safe
- idempotent
- Request a page
- URL encoding

HEAD
- safe
- idempotent
- Request the head part only
- Retrieve meta–information

GET path HTTP/1.1

HEAD path HTTP/1.1
POST, PUT, DELETE, PATCH

POST
- Request a page
- Parameters are encoded in the request
- Can send more data than GET
  - Form data
  - File upload

PUT
- Store an entity.
- Idempotent
- File upload

DELETE
- Removes an entity.
- Idempotent

PATCH
- Partial modification of an entity.
CONNECT
- use tunnel such as SSL

OPTIONS
- Query HTTP methods supported.

TRACE
- Echo HTTP request.

Web servers not always supports all of these methods!
You can check them with `telnet` or `netcat`. 
HTTP Status Codes

- Part of the response
- Sent by the server
- Clients computes
- Important for AJAX, REST

Categories

- 1XX Informational
- 2XX Successful
- 3XX Redirection
- 4XX Bad Request
- 5XX Sever Error
1XX

- HTTP/1.1
  - to switch from HTTP/1.0 to HTTP/1.1
  - Unknown 1XX → Ignore

100 Continue
101 Switching Protocols
  HTTP/1.0
  ↓
  HTTP/1.1
102 Processing
2XX

- Request
  - received
  - understood
  - accepted
  - processed

- 200 OK
  - GET
  - HEAD
  - POST
  - TRACE

- 201 Created
- 202 Accepted
- 203 Non-Authoritative Information
- 204 No Content
- 205 Reset content
- 206 Partial content
3XX

- for the User Agent
  - Browser modifies the request automatically.
  - Caused by
    - site moved.
    - usage of proxy
    - redirection

- 300 Multiple choices
- 301 Moved Permanently
- 302 Found
- 303 See Other
- 304 Not Modified
- 305 Use Proxy
- 306 Unused
- 307 Temporary Redirect
4XX

403

405
4XX

- Bad Request
- Client–side problem
- Any request can cause it.
- Examples:
  - bad web address
  - authentication required

- 400 Bad Request
- 401 Unauthorized
- 402 Payment Required
- 403 Forbidden
- 404
- 405 Method Not Allowed
- 406 Not Acceptable
- …
Server Error
Can be caused by any request.
There is a problem at the server side.
All users, who use the system, will get the error code.
For example,
  - exception do not handled
  - database unavailable
It can show the poor job of the developers.

501  Not Implemented
502  Bad Gateway
503  Service Unavailable
504  Gateway Timeout
505  HTTP Version Not Supported