

INF3190 Group lecture

Lecture #2

Jan Anders Bremer

Math-linux.com

25.01.2013

- OSI Model vs Internet Model
- Protocols and services
- A program using sockets

Contact information

Phone: 91125994

Mail: janabr@ifi.uio.no

IRC: janabr at irc.ifi.uio.no (currently in [#ping.uio.no](#) and [#cyb](#), but you can always `/msg me`)

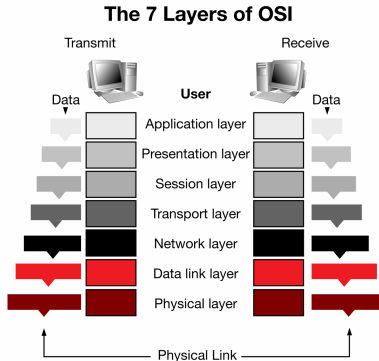
Feel free to send questions, suggestions and feedback.

OSI Model 1/6

- Used to properly separate different tasks in the network stack
- Each layer offers services to the layer above and uses the layer beneath
- Every layer adds their own header.

Source (image):

<http://realitypod.com/2012/03/osi-model/>



OSI Model 2/6 - L1 Physical

- Lowest level - concerned with transferring over the a physical medium between two machines *mostly*.
- Multiplexing
- Devices operating on this level *exclusively*: repeaters, hubs.

OSI Model 3/6 - L2 Data Link

- Polls L1 for data
- Concerned with delivering content on the local network.
- Offers more advanced services: error detection (and correction), ARP (mac-to-IP-translation), Quality of Service, flow control.

OSI Model 4/6 - L3 Network

- Polls L2 for data.
- Packet forwarding/routing (hierarchical)
- Connectionless communication

OSI Model 5/6 - L4 Transport

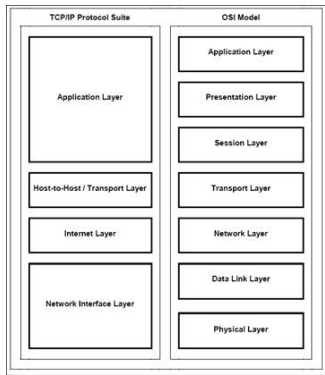
- Polls L3 for data
- Provides (maybe!) reliability, flow control and multiplexing and many other services.
- Most known protocols: UDP and TCP
- You can choose what protocol to use in an application.

OSI Model 6/6 - Remaining layers

- L5 - Session
- L6 - Presentation
- L7 - Application
- L7 is what most programs "see".
- The layers are described in detail on multiple pages on the internet.

Internet Model (Internet protocol suite)

- Each corresponding section is responsible for the same things.
- Source: http://www.sqa.org.uk/e-learning/NetInf101CD/page_15.htm



Protocols and services

- A layer offer services to the layer above it and uses services from the layer beneath them.
- Protocols are used between the same layers to provide these services.
- Example: UDP or TCP.

Sample program: chat

- Server-client system
- Demo: simple implementation first (bad!)
- Select-based (better)
- Write the server first as the client can be replaced by netcat.

- `https://github.com/TZer0/INF3190-groupcode`
- To get the source code run the following command in a terminal: `git clone`
`git://github.com/TZer0/INF3190-groupcode.git`
- If this fails, install `git-svn`.