Special Course in Computer Science:
Local Networks
(456400.8)

Lecture 1
14.3.2012
Aim of the course

• Motivation
  • Network: communicating computers
  • LANs (local-area networks) are wide-spread
    • “gates” towards Internet communication
    • the means of sharing data, hardware, and software
    • the suitable choice for some industrial processing
    • the fast backbone connecting slower networks
  
• To understand the fundamentals of LANs
LANs at home

- Appliances in the home (will be) capable of communicating
  - TVs, DVDs, phones, cameras, clock radios, thermostats, utility meters
- Different properties than normal LANs
  - Very easy to install
  - Foolproof in operation
  - Low price
  - Scalability: start with one-two devices and be possible to add later
  - Security and reliability
- Wired, wireless, power-line networks
Practicalities

• When and where
  • Wednesdays and Fridays 10-12, in Catbert (B3028), ICT-house

• Components
  • Lectures (about 28h), exercises (about 8h), essay writing and examination

• Registration
  • at https://minplan.abo.fi/minplan/

• Webpage: http://www.users.abo.fi/lpetre/localnet12/

• Teaching
  • Luigia Petre and Maryam Kamali
Materials

  - Copies in the library (2-3)
  - About 10 lectures based on this book
- Research papers on local networks
  - About 4 lectures based on these
- Exercises from the book
- Papers on local networks
Part 1: Local Networks fundamentals
- 2 lectures are followed by an exercise session
- We have about 4-5 such little modules
  - Preliminaries (telecom fundamentals)
  - Bases of LANs (LAN protocols)
  - LANs today (major protocols and technologies)
  - Further LAN insights (connecting LANs, LAN management)
  - Other issues (LAN performance)

Part 2: Advanced topics in Local Networks
- 4-5 lectures
- Topics can include: sensor networks, Bluetooth, ÅA’s network, network-on-chip, etc
What is a network?

- Nodes connected by links *through a single technology*
  - Connected nodes: able to exchange information
- Nodes
  - Computer
  - Printer
  - Any device capable of sending/receiving data generated by other network nodes
- Links = communication channels
Types of networks

- Depending on size, ownership, covered distance, physical architecture, transmission technology

- Local area network (LAN)
- Metropolitan area network (MAN)
- Wide area network (WAN)
Local Area Networks

- Privately owned, size limited to few kilometers
- Links the devices in a single office, building, campus
Metropolitan Area Networks

- Designed to extend over an entire city
- Single network (cable TV) or a connection of several LANs into one network
Wide Area Networks

- Provides long-distance transmission of data, voice, image, video information over large geographical areas

![Diagram of Wide Area Networks](image-url)
Inter-networks

- Two or more networks connected by using specialized devices (routers, gateways)
LAN definition

“Data communicating system allowing a number of independent devices to communicate directly with each other, within a moderately sized geographical area over a physical communication channel of moderate data rate”

(IEEE)
LAN Components - Hardware

- Stations (nodes)
  - Must be capable of connecting to network
  - NIC (network interface card) is ensuring the necessary circuitry (chips) to perform network functions

- Transmission media: guided or unguided

- Connecting devices
  - Transceivers: connect media to stations
  - Repeaters and bridges: connect network segments
LAN Components - Software

- **Network operating system (NOS)**
  - Allows the logical connection of stations and devices to the network
  - Enables users to communicate and share resources

- **Application programs**
  - Allow users to solve specific problems
  - Not specific to LANs
LAN Models

- **Client/Server LAN**
  - Some stations (the servers) provide services to other stations (the clients)
  - NOS specialized for server or client
  - Either a general server or several dedicated servers

- **Peer-to-peer LAN**
  - A station can be both a client and a server
Dedicated servers in a Client/Server LAN

- In a large network there can be servers dedicated to various tasks
One general server in a Client/Server LAN

- In a small network there may be only one server, responsible for all services typically required from a server.
Peer-to-peer LAN

- No station is designated as server or client
- Each station runs server programs and also client program if needed
LAN Applications

- **Office networks**
  - For sharing, interoffice communication, and Internet communication

- **Industry networks**
  - Some LAN architectures suitable for automated manufacturing and production

- **Backbone networks**
  - When a high-speed LAN is used as backbone to connect low-speed networks (e.g. campus)
Data Communication Models

- Real communication
  - between one application program (AP) at one end system and another AP on another end system
- Need for harmony
  - APs should understand each other
  - End systems should coordinate
  - Intermediate system should facilitate information routing
  - Transmission media should provide desired rate and accuracy
3 models

- Layered architecture
  - Abstract and most general
- OSI model
  - A standard
  - Can be seen as an application of the above
- TCP/IP model
  - The earliest practical model, still in use
Layered architecture model

- The task of communicating between 2 APs broken into smaller subtasks
- Each subtask attached to a layer
- The subtask of each layer is solved by a protocol
- **PDU** (protocol data unit): unit of data exchanged between corresponding layers
  - N-PDU
Peer-to-peer protocols

- Protocols ensure that layer N of the source **logically** communicates with layer N of the destination.
Encapsulation & Decapsulation

• The real communication (as opposed to logical communication) is realized through layers

• Source: data flows downwards
  • Headers/trailers are added to the upper layer PDU \( \rightarrow \) encapsulation

• Destination: data flows upwards
  • Headers/trailers are removed from the lower layer PDU \( \rightarrow \) de-capsulation
Illustration of E & D

A

(Internet) B

\[(N+1) \text{ PDU} \]
\[N+1 \text{ header} \]

\[N \text{ PDU} \]
\[N \text{ header} \]

\[(N-1) \text{ PDU} \]
\[N-1 \text{ header} \]

\[(N-1) \text{ PDU} \]
\[N \text{ header} \]

\[(N+1) \text{ PDU} \]
\[N+1 \text{ header} \]

a. Encapsulation

b. Decapsulation
OSI Model

- **Open System Interconnection (OSI) model** → layered framework allowing for communication across all types of computer systems
- 7 layers
Physical layer (PhL)

- Responsible for creating a *bit link* (physical connection) between sender and receiver
Concerns in the physical layer

- Representation of bits
- Data rate: \textit{bits/sec}
- Bit synchronization
- Characteristics of interfaces
- Transmission medium
- Transmission mode
Data Link layer (DLL)

- Responsible for the *hop-to-hop* delivery
- Combines the PhL bits into *frames* and delivers the frames to the next hop
Jobs of the DLL

- Framing
- Addressing
- Medium access control
- Flow control
- Error control
Network layer (NL)

- Responsible for source-to-destination delivery of a packet
- Possibly across multiple networks
Jobs of the NL

- Creating a logical end-to-end connection (NL should make a logical network out of many physical networks)
- Hiding the details of the lower layers
- Addressing
- **Routing**
  - Finding the optimum path from the many different paths available
Transport layer (TL)

- Responsible for the end-to-end, error-free delivery of the entire message
Jobs of the TL

- Service-point addressing
  - Handles service-point addresses (port addresses)
- Segmentation and reassembly
- Connection control
- Flow control
- Error control (and correction)
Session layer (SL)

- Network dialog controller
- Establishes, maintains, and synchronizes the interaction between communicating systems
- Responsibilities
  - Half- and full-duplex service
  - Synchronization
  - Atomization
Presentation layer (PL)

- Responsible for solving the different syntaxes of representing data on the communicating systems
- Specific responsibilities
  - Translation
  - Encryption
  - Compression
Application layer (AL)

- Enables the user to access the network
- Provides user interfaces and support for
  - Email
  - Remote file access and transfer
  - Shared database management
  - Various types of distributed information services
Summary of OSI layers

- **Application**: To allow access to network resources
- **Presentation**: To establish, manage, and terminate sessions
- **Session**: To move packets from source to destination; to provide internetworking
- **Transport**: To transmit bits over a medium; to provide mechanical and electrical specifications
- **Network**: To organize bits into frames; to provide hop-to-hop delivery
- **Data link**: To provide reliable process-to-process message delivery and error recovery
- **Physical**: To translate, encrypt, and compress data
TCP/IP Protocol Suite

Application
- SNMP
  Simple Mail Transfer Protocol
- FTP
  File Transfer Protocol
- Terminal Network

Presentation

Session

Transport
- TCP
  Transmission Control Protocol
- UDP
  User Datagram Protocol

Network
- IP
  Internetworking Protocol

Data link
- Protocols defined by the underlying networks

Physical
Layers in TCP/IP

- 2 lower layers, depending on the network
- **Network layer**
- **Transport layer**
- **Application layer**
  - May embed protocols from OSI’s session and presentation layers
Network and transport layers

- **Network (internet(work)) layer**
  - Main protocol: **IP** (*Internetworking Protocol*)
    - Creates packets called **IP datagrams**
    - Sends them to destination
    - Intermediate nodes route the datagrams

- **Transport layer**
  - 2 protocols: **TCP** and **UDP**
  - TCP: Transmission Control Protocol
  - UDP: User Datagram Protocol
IEEE Standards

- Project 802, started in 1985
  - Standards to enable communication between equipment from a variety of manufacturers
  - It details functions of Physical layer and Data Link layer for major LAN protocols
- DLL: LLC and MAC
  - LLC: architecture-independent
  - MAC
    - Contains distinct modules
    - Each module carries proprietary information specific to the LAN being used
    - Contention for shared media
    - Synchronization, flag, flow, error control specifications
    - Protocols specific to Ethernet, token ring, token bus, etc
- PhL: PMD and PMI
  - Depend on the implementation and type of media used
Data Transmission Issues

- Transmitting data through a LAN implies *transforming the data into signals*, to be carried by a medium
- Signals: *digital* or *analog*
- Transmission: *digital* or *analog*
- *Multiplexing* can be used for transmission
- Theoretical *limitations* on the data rate
Digital signals

- Have only a limited number of values (they are discrete)
- Transition between values: instantaneous
Bit Interval and Bit Rate

- **Bit interval**: seconds needed to send one bit.
- **Bit Rate**: number of bit intervals per second = number of bits per second (bps).
- **Bit interval** = $1 / \text{Bit rate}$.
Analog Signals

- Have an infinite number of values (they are *continuous*)
- Transition between values: *smooth*
Periodicity

- Analog signals can be
  - *Periodic*: the signal has a repeated pattern
  - *A-periodic*: there is no repeating pattern
Sine wave

- Most fundamental form of a periodic signal
- Can be fully described by: *amplitude*, *period* or *frequency*, and *phase*
Amplitude

- On a graph: the value of the signal at any point on the wave
Period or frequency

- **Period** ($T$): seconds a signal needs to fulfill one cycle
- **Frequency** ($f$): number of periods per second
- $T = 1/f$
Phase

- The position of the waveform relative to time 0, measured in degrees
- Indicates the status of the first cycle

![Diagram showing phase](image)

- a. 0 degrees
- b. 90 degrees
- c. 180 degrees
More on analog signals

- **Composite signals**: made of several sine waves with different amplitudes, periods (frequencies), and phases
- **Time-domain plot**: changes in signal amplitude with respect to time
- **Frequency-domain plot**: relationship between amplitude and frequency
  - Specifically used to show components of a composite signal
Time and frequency domains

a. Time domain

b. Frequency domain
Plots of different signals

Time domain

Frequency domain

17

10

7

0

6

7

6

10

0

1 second

1 second

1 second

1 second
Frequency spectrum and Bandwidth

• **Frequency spectrum** of a signal: the set of all the component frequencies it contains
  • Shown with a frequency domain

• **Bandwidth of a signal**: the width of the frequency spectrum
  • The difference between the highest and the lowest frequencies

• **Bandwidth**: the range of component frequencies

• **Spectrum**: elements within the range
Illustration of the bandwidth

Bandwidth = 5000 – 1000 = 4000 Hz
Digital transmission

- A digital signal is used to send information from sender to receiver
- If information-to-be-sent is binary
  - we use line coding to encode the bits into a digital signal
- If information-to-be-sent is analog
  - We first sample the continuous quantity to a sequence of bits, then use line coding
Line coding

- The process of transforming binary information (sequence of bits) into a digital signal
- Needed for data coming from a computer and passing through a digital communication system
Line coding aspects

- Signal level vs. data level
- Pulse rate vs. bit rate
- Self-synchronization
Signal level vs. Data Level

- **Signal levels:** levels allowed in a particular signal
- **Data levels:** levels used to represent data

![Diagram](image-url)
Pulse rate vs. bit rate

- **Pulse rate**: number of pulses per second
- **Bit rate**: number of bits per second
- If a pulse defines only one bit => pulse rate = bit rate
- If a pulse defines more than one bit => pulse rate < bit rate
- In general: \( \text{bit rate} = \text{pulse rate} \times \log_2 L \), where \( L \) = number of data levels
Self-synchronization

- Self-synchronizing digital signal
  - Includes timing information in the transmitted data
  - Achieved with transitions in the signal, alerting the receiver to the beginning, middle, or end of a bit interval
We are henceforth assuming only two data levels
- To represent 0 and 1
- Hence: pulse rate = bit rate
Unipolar encoding

- Simple: 1 encoded as positive voltage, 0 encoded as zero voltage
- Primitive: no means of self-synchronization
Polar encoding

- Uses 2 non-zero signal levels: one positive and one negative and is sometimes self synchronized
Non-return to Zero (NRZ)

- Signal level either positive or negative
Return to Zero (RZ)

- A positive/negative level indicates a 1 or a 0 bit
- The signal changes in the middle of the bit interval to remain zero on that interval

These transitions can be used for synchronization.
Biphase

- Best existing solution to synchronization
- The signal changes at the middle of the bit interval to the opposite pole
- Manchester encoding
  - Inversion at half interval used for synchronization and bit representation
- Differential Manchester encoding
  - Inversion at half interval used for synchronization
  - Transition at beginning of interval: binary zero
Manchester and Differential Manchester

Presence of transition at the beginning of bit time means zero.
Bipolar

- Has 3 voltage levels: 0, -, +
- 0: binary zero
- - and +: binary 1
- If the first 1 bit is represented with +, then the following is represented with –, the third with +, and so on
- Alternation occurs independently of the consecutiveness of the 1 bits
Block coding

- Improves line coding using redundancy for synchronization (and error detection)
- Steps
  - Sequence of bits divided into m-bit blocks
  - Each m-bit block is substituted by an n-bit block, \( n > m \)
  - Each n-bit block is line-coded without considering synchronization problems
Illustration of block coding

$mB$ coding

$m$ bits

$0 1 0 .... 1$  
$0 0 0 .... 1$  
$1 1 0 .... 0$

$mB$ to $nB$ conversion

$n$ bits

$0 1 1 .... 111$  
$0 0 0 .... 001$  
$0 1 0 .... 101$

Line coding
Substitution in block coding

- Every \textit{m-bit block} is substituted with an \textit{n-bit block}, denoted \textit{mBnB}.
Pulse code modulation (PCM)

- One of the methods to send continuous data (voice, video) over a digital transmission medium
- Includes
  - Sampling
  - Quantization
  - Assigning binary values
Sampling illustration

a. Analog signal

b. PAM signal
Quantization illustration
## Conversion to binary numbers

<table>
<thead>
<tr>
<th>Value</th>
<th>Binary</th>
<th>Value</th>
<th>Binary</th>
<th>Value</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>024</td>
<td>00011000</td>
<td>015</td>
<td>10001111</td>
<td>125</td>
<td>01111101</td>
</tr>
<tr>
<td>038</td>
<td>00100110</td>
<td>080</td>
<td>11010000</td>
<td>110</td>
<td>01101110</td>
</tr>
<tr>
<td>048</td>
<td>00110000</td>
<td>050</td>
<td>10110010</td>
<td>090</td>
<td>01011010</td>
</tr>
<tr>
<td>039</td>
<td>00100111</td>
<td>052</td>
<td>00110110</td>
<td>088</td>
<td>01011000</td>
</tr>
<tr>
<td>026</td>
<td>00011010</td>
<td>127</td>
<td>01111111</td>
<td>077</td>
<td>01001101</td>
</tr>
</tbody>
</table>

Sign bit
+ is 0, – is 1
Sampling rate

• Accuracy of the digital representation depends on how many samples we take
• According to theoretical results, sampling rate should be at least twice the highest frequency component of the signal
• Hence, a sampling rate of $2x$ ($x$ is the highest frequency) must be done once every $1/2x$ seconds
Illustration of the sampling

Highest frequency = $x$ Hz
Sampling rate = $2x$ samples/second

Sampling interval = $1/2x$
Bit rate

- How many bits per sample?
  - Depends on the required precision
- Bit rate = sampling rate \times \text{number of sample bits}
Analog transmission

- A digital or an analog signal *modulates* a high-frequency analog signal, called *carrier*

- *Modulation*: modification of one or more characteristics of the carrier by an information-bearing signal
Baud Rate vs. Bit Rate

- **Baud**: signal element
  - Transmits a unit of information
- **Baud rate**: number of signal elements per second
- **Bit rate**: number of bits per second
- **Bit rate** = **Baud rate** × \( N \), where \( N \) is the number of bits in each signal element
Modulation schemes for digital signals

- ASK
- FSK
- PSK
- QAM
Amplitude Shift Keying (ASK)

- Amplitude of carrier is varied to represent binary 1 or 0
- Frequency and phase remain constant
- Highly susceptible to noise interferences
Frequency Shift Keying (FSK)

- Frequency of carrier is varied to represent binary 1 or 0
- Frequency during each bit duration is constant (corresponding to 0 or 1)
- Peak amplitude and phase remain constant
- Avoids most of the noise problems of ASK
Phase Shift Keying (PSK)

- Phase of carrier is varied to represent binary 1 or 0
- Peak amplitude and frequency remain constant
- Common to have bit rate > baud rate
Quadrature amplitude modulation (QAM)

- Combines ASK and PSK; most efficient
Multiplexing

- Dividing the available bandwidth of a link between several users

a. No multiplexing

b. Multiplexing
Multiplexing techniques

- Frequency-division multiplexing (FDM)
- Time-division multiplexing (TDM)
  - Synchronous
  - Asynchronous
FDM

- Analog technique
- Applied when the bandwidth of a link > the combined bandwidths of the signals to be transmitted (in hertz)
- Signals generated by sending devices modulate different carrier frequencies
- These modulated signals are combined into a single one transported by the link
FDM concept
FDM time-domain illustration
TDM

- Digital process
- Applied when the bandwidth of the transmission medium > the bandwidth required by the sending and receiving devices (in bps)
- Can be implemented as synchronous TDM or asynchronous TDM
TDM Illustration
Synchronous TDM

- Multiplexer allows *exactly the same time slot to each device at all times*, whereas or not the device has anything to transmit
- Time slots are grouped into frames
- Frame: a complete cycle of time slots
  - One or more slots dedicated to each sending device
  - $n$ input lines $\Rightarrow$ at least $n$ slots per frame
Illustration of synchronous TDM
Framing bits

- Time slot order in a synchronous TDM constant
- Timing inconsistencies can be caused
- One/more synchronization bits (framing bits) are added in the beginning of each frame
- Framing bits follow a pattern to allow the demultiplexer to synchronize with the incoming stream
Illustration of framing bits
Asynchronous TDM (statistical TDM)

- Designed to avoid waste of link capacity
- The total speed of the input lines can be greater than the capacity of the path
  - \( n \) input lines \( \Rightarrow \) at most \( m \) slots per frame, \( m < n \)
- Supports same number of input lines with lower capacity link than synchronous TDM
- Given the same capacity link, it supports more devices than synchronous TDM
Illustration of asynchronous TDM
How Asynchronous TDM works

- $m$ based on statistical analysis
- Each slot available to any of the attached input lines having data to send
- Multiplexer scans input lines, accepts portions of data until a frame is filled and sends
  - If not enough data to fill frame, frame sent incomplete
Addressing and Overhead

- Each slot carries an address telling the demultiplexer how to direct the data
  - Address for local use only
  - Discarded by demultiplexer once read
- Adding addresses => overhead => limits efficiency
  - Usually small
Data Rate

- Upper limit of the data rate for a particular transmission medium is interesting
- Theoretical upper limits
  - Noiseless channel: Nyquist
  - Noisy channel: Shannon
  - Actual data rates are below these theoretical limits
Theoretical data rates

- **Nyquist bit rate**
  - \( \text{BitRate} = 2 \times \text{Bandwidth} \times \log_2 L \), where \( L \) is the data level of the channel

- **Shannon capacity**
  - \( \text{Capacity} = \text{Bandwidth} \times \log_2 (1+\text{SNR}) \), where SNR is the signal-to-noise ratio, capacity is measured in bps