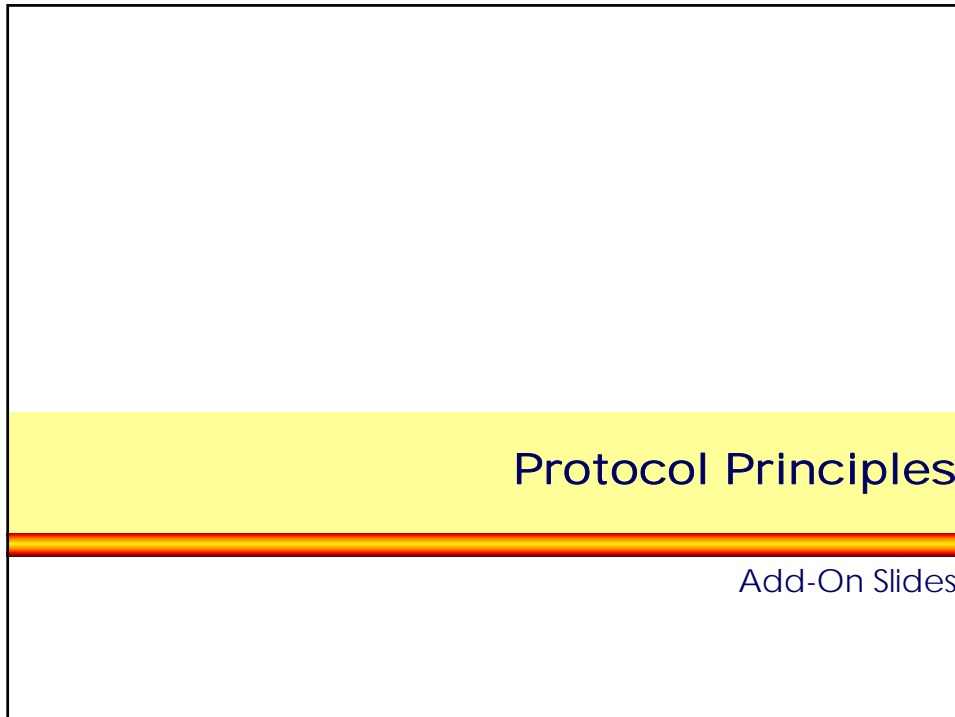# Protocol Principles

Add-On Slides
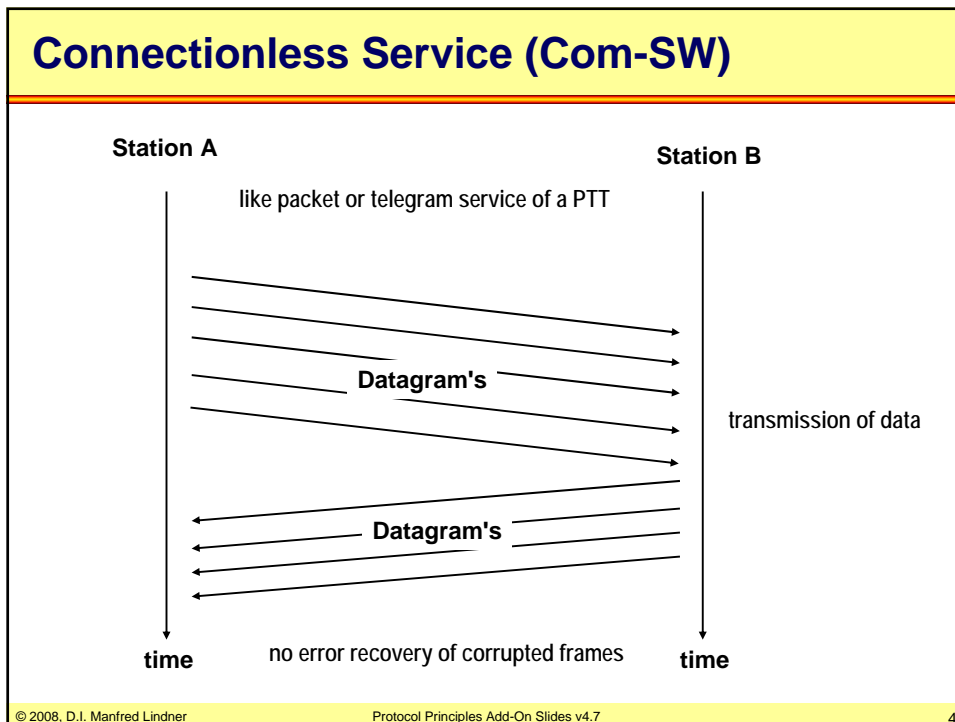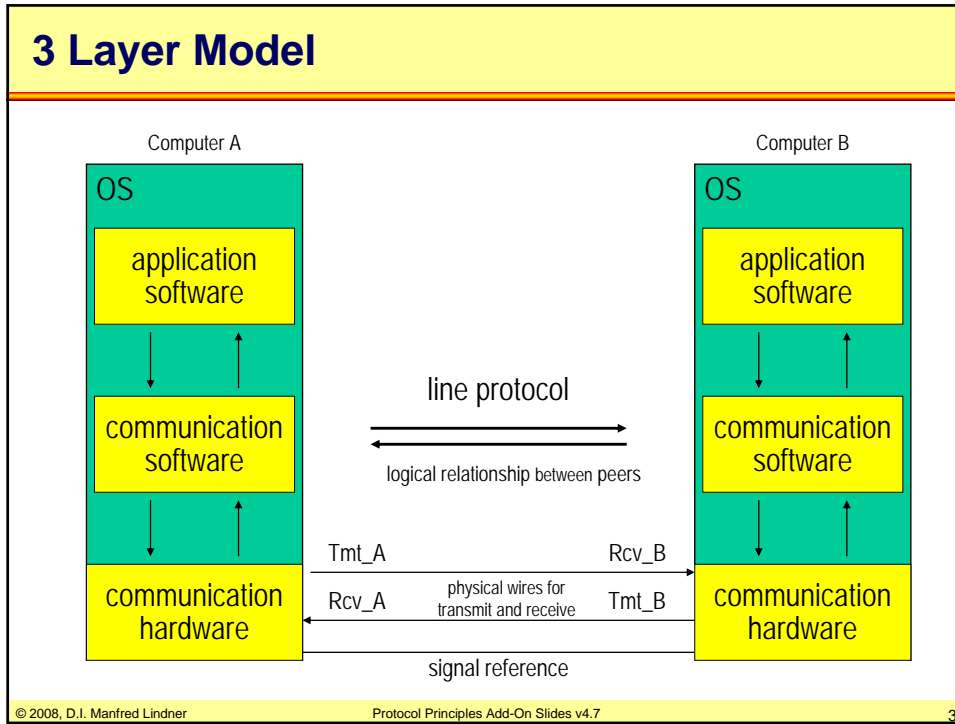
Summary Protocol Principles Day 1

- **Two important principles of data communication**
  - Layering
    - Structuring the complex task of data communication into smaller pieces by usage of "layers"
    - A layer is built by the resources of the corresponding protocol peer entities and by the protocol procedures performed between them
      - protocol standards define fields of the control field of a frame (bits seen on the wire) and the communication behavior of the peers receiving and sending frames
    - A layer is using the services of the lower layers to provide a enhanced service to the upper layer
      - The application layer can access the lower layer (the protocol stack) via API (application programming interface)
      - The communication layer can access the lower layer via network-card driver
  - Connectionless versus connection-oriented service

© 2008, D.I. Manfred Lindner

3 Layer Model

Computer A   Computer B

OS   OS

application software   line protocol   application software

communication software   logical relationship between peers   communication software

Tmt_A   Rcv_B

communication hardware   Rcv_A   physical wires for transmit and receive   Tmt_B   communication hardware

signal reference

Protocol Principles Add-On Slides v4.7
3

Connectionless Service (Com-SW)

Station A   Station B

like packet or telegram service of a PTT

Datagram's

transmission of data

Datagram's

time   no error recovery of corrupted frames   time

Protocol Principles Add-On Slides v4.7
4

Connection-Oriented Service (Com-SW)

Station A   like telephone call of Telecom   Station B

Connection Request

Connection Acknowledgement

connection
establishment

DATA

transmission of data

Disconnection Request

Disconnected Acknowledgment

clearing
of connection

time   error recovery of corrupted frames is possible   **time**
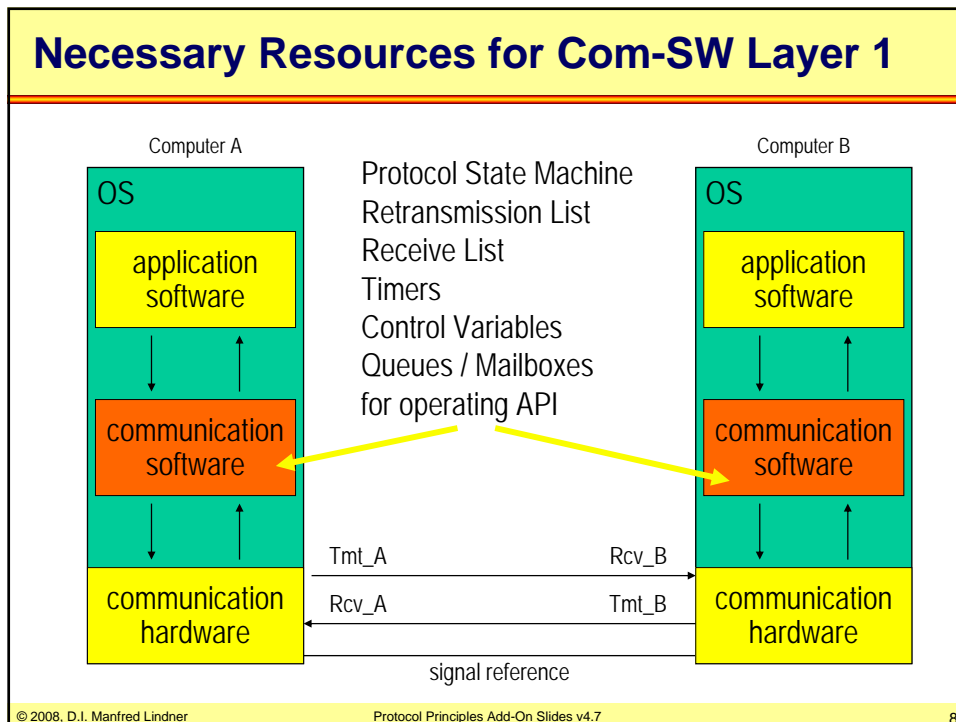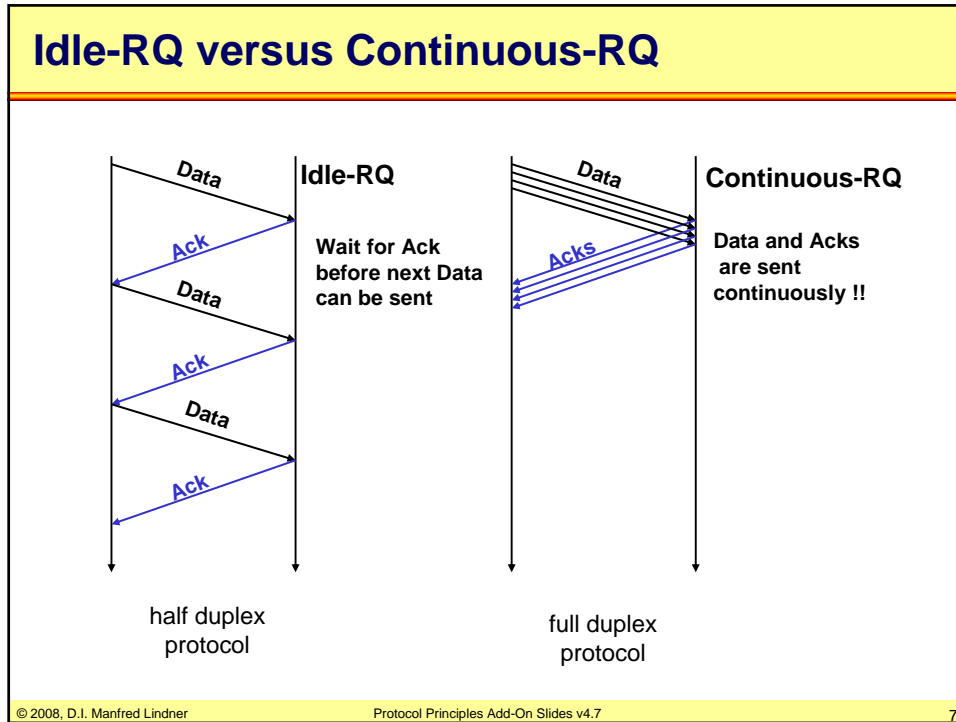by usage of ARQ techniques

Summary Protocol Principles Day 2

- **Connection-oriented service**
  - allows error recovery by feedback error control
  - ARQ techniques
    - Receiver acknowledges correct receipt of data frame
    - Idle-RQ
      - Easy to implement (few resources necessary) but inefficient concerning usage of bandwidth in case of full-duplex line
    - Continous-RQ
      - Several methods
        - » Selective ACK (e.g. TCP optional SACK procedure)
        - » GoBackN (e.g. HDLC basic REJ procedure)
        - » Positive Acknowledgement (e.g. TCP basic procedure)
        - » Selective Reject (HDLC optional SREJ procedure)
      - More complex to implement (much more resources necessary) , more efficient concerning usage of bandwidth in case of full-duplex line

L02 - Protocol Principles Add-On Slides

Idle-RQ versus Continuous-RQ

Idle-RQ

Wait for Ack before next Data can be sent

Continuous-RQ

Data and Acks are sent continuously !!

half duplex protocol

full duplex protocol

Necessary Resources for Com-SW Layer 1

Computer A

OS

application software

communication software

communication hardware

Protocol State Machine
Retransmission List
Receive List
Timers
Control Variables
Queues / Mailboxes
for operating API

Tmt_A   Rcv_B

Rcv_A   Tmt_B

signal reference

Computer B

OS

application software

communication software

communication hardware

© 2008, D.I. Manfred Lindner

Page 02 Add On - 4

Necessary Resources for Com-SW Layer 2

Computer A

OS

application software

Frame Types:
Connect request, ack
Disconnect request, ack
I, ACK, NACK, SREJ
Identifiers

communication software

line protocol

communication hardware

Tmt_A   Rcv_B

Rcv_A   Tmt_B

signal reference

Computer B

OS

application software

communication software

communication hardware

Generic Frame Format

frame header   frame trailer

payload

| SYNC | SD | Control | DATA | FCS | ED |

control information
= line protocol header

checksum

Frame Type Field: Connect Request, I, ACK, NACK, …
Identifier Field: N, N+1, …

© 2008, D.I. Manfred Lindner