

01 Ü Einführung

11. April 2016

Luke

- 1.1
 - a)
 - * Sterntopologie: Ein zentrales Element(Sternkoppler), jeder Rechner benötigt eine Leitung zu Sternkoppler $\rightarrow 5$
 - b) Jeder mit Jedem $= 4 + 3 + 2 + 1 = 10$
 - c)
 - * $l(n) = n$ bei Sterntopologie
 - * $l(n) = \sum \dots = (n * (n - 1))/2$ bei vollvermaschter Topologie
 - d)
 - * LAN
 - Reichweite: 10m
 - Reaktionszeit: niedrig
 - Datenrate: hoch
 - Topologien: Sterntopologie
 - * MAN
 - Reichweite: 10km
 - Reaktionszeit: mittel
 - Datenrate: mittel
 - Topologien: hierarchische Topologie
 - * WAN
 - Reichweite: 100km - 10.000km
 - Reaktionszeit: hoch
 - Datenrate: niedrig
 - Topologien: Vollvermaschte Topologie

- 1.2

- a) Dienst und Protokoll
 - * siehe Musterlösung
- b) OSI Schichtenmodell
 - * Schichtenmodell siehe Folie 1.8ff
 - * Protokoll:
 - ist eine Sprache zur horizontalen Kommunikation zwischen Prozessen derselben Schicht auf verschiedenen Hosts
 - * Dienst
 - dient der vertikalen Kommunikation zwischen zwei Schichten auf einem Host
 - * Aufteilung des Bitstroms: Schicht 2 Sicherungsschicht
 - * Ende-zu-Ende Kommunikation: Schicht 4 Transportschicht
 - * Wegewahl: Schicht 3 Vermittlungsschicht
- c)
 - * keine inhaltliche Bearbeitung, sondern nur Informationsweiterleitung

- 1.3

- a)
 - * siehe Folie 1.15;
 - * Initiator (Prozess A), ...
 - * Responder (Prozess B), ...
- b)
 - * Zustände bestimmen
 - idle
 - connected
 - prepare(Initiator)
 - prepare(Responder)
 - * Übergänge bestimmen (Knoten, Pfad, Knoten)
 - (idle, conReq, prep(Init))
 - (idle, ConInd, prep(Resp))
 - (prep(Resp), conRsp, connected)
 - (prep(Init), conCnf, connected)
 - (connected, dataRep/dataInd, connected)
 - (prep(Resp)/prep(Init)/connected, disRep/disInd, idle)

- c)
 - * Ablaufdiagramm
 - c1) + zeitlicher Ablauf
 - c2) - es werden n Diagramme benötigt
 - c3) -
 - * Zustandsdiagramm
 - c1) -
 - c2) + alle Abläufe in einem Diagramm darstellbar
 - c3) +
- 1.4
 - a) siehe Folie 1.10
 - * $PDU(N) = SDU(N - 1)$
 - * $IDU(N) = ICI(N) + SDU(N)$
 - b) Seitenaufruf: <http://www.heise.de/software>
 - * httpRequest
 - GET/software/http/1.1
 - Host: www.heise.de
 - * ICI
 - ip: 193.99.144.85 port:80
 - * SDU
 - GET/software/http/1.1
 - Host: www.heise.de
 - * IDU
 - ICI
 - SDU
 - * TCP-PDU
 - src:80, dest:80,...
 - SDU
 - Data
 - c)
 - * $b_0 = 125 \frac{Mbit}{s}$
 - * $b_1 = b_0 * 0,8$
 - * $b_2 = b_1 * \frac{(55+99)*0,01}{2}$
 - * $b_3 = b_2 * \frac{(57+99)*0,01}{2}$
 - * $b_4 = b_3 * \frac{(23+99)*0,01}{2} = 36,4 \frac{Mbit}{s}$

$$* b_4 = b_{goodput}$$

$$* b_{extra} = b_2 * \frac{(23+99)*0,01}{2} = 46,7 \frac{Mbit}{s}$$

- timo.schick@tu-dresden.de

- THE END :) -