

SS 16

Luke Hain

14. April 2016

Inhaltsverzeichnis

I	Computer Networks	3
1	Vorlesung	4
1.1	Einführung	4
1.2	Bitübertragungsschicht	5
1.2.1	Nachrichtentechnische Kanäle	5
1.2.2	Übertragungsmedien	6
1.2.3	Mehrfachnutzung von Kanälen	8
1.2.4	Datenübertragung	9
1.2.5	Beispieltechnologien	10
1.2.6	Digitaler Netzzugang über Kabelmodem	10
2	Übung	11
2.1	Einführung	11
2.1.1	11
2.1.2	12
2.1.3	12
2.1.4	13
II	Theoretical Informatic and Logic	15
3	Vorlesung	16
3.1	Prädikatenlogik erster Stufe	16
3.2	Prädikatenlogik erster Stufe	17
3.3	Prädikatenlogik erster Stufe	18
3.3.1	Komposition von Substitutionen	18
3.3.2	Beschränkung von Substitutionen	18
3.3.3	Anwendung von Substitutionen auf Formeln	19
3.3.4	Substitutionen und Formeln	19
3.3.5	Satz 4.18	19
3.3.6	Beweis Hilfsaussage aus Satz 4.18	21
3.3.7	Varianten	21

4 Übung	22
4.1 Prädikatenlogik	22
III Computer Architecture	23
5 Vorlesung	24
5.1 Einführung	24
5.1.1 Big Data	24
5.2 Vorlesung	24
5.2.1 ZIH	24
5.2.2 Begriffe und Definitionen	24
6 Übung	26
6.1 Einführung	26
IV Database	27
7 Vorlesung	28
7.1 Einführung	28
7.2 Konzeptueller Entwurf	29
7.2.1 Drei Phasen des Datenbank-Entwurfs (4, ff.)	29
7.2.2 Lebenszyklus einer Datenbank	29
7.2.3 Prinzip eines Datenmodells (16)	29
7.2.4 Entity-Relationship-Modell	30
8 Übung	32
8.1 Einführung	32
V Hardware Internship	33
VI C++4CG	34

Teil I

Computer Networks

Kapitel 1

Vorlesung

1.1 Einführung

- Anwendungsfelder Rechnernetze (1.4)
 - Geschäftsanwendungen - gemeinsame Nutzung von Ressourcen
 - Privatbereich - Informationszugriff (z.B. WWW, IM)
 - Mobile Benutzer - Textnachrichten, ...
 - Gesellschaftliche Aspekte - Copyright, Profile, ...
- Client Server Modell (1.5)
- Peer-to-Peer Communication (1.6)
- Basis-Netzstruktur (1.7)
 - Übertragungsmodi
 - * Verbindungsorientiert
 - * Verbindungslos (z.B. IP)
 - * Leitungsvermittelt
 - * Paketvermittelt (flexibler, ressourcenschonend)
- Schichtenarchitektur - ISO/OSI Referenzmodell (1.8)
 - International Organization for Standardization
 - Open Systems Interconnection
 - Schichtenübersicht auf 1.8 ff.
- Integriertes Referenzmodell (Tanenbaum) (1.11)
 - Protokollimplementierung oft abweichend vom Referenzmodell

- Beispiel Datenübertragung (1.12)
- Schichteneffizienz (1.13)
- Dienste - Begriffsklärung (1.14)
 - Beispiel Ablaufdiagramm (1.15)
- Netzkopplung - Basis-Topologien
 - Punkt-zu-Punkt-Kanäle (Unicast)
 - Rundsendekanäle (Broadcast)
 - Klassifizierung nach Ausdehnung (1.17)
 - * Pan - Personal Area Network
 - * LAN - Local Area Network
 - * MAN - Metropolitan Area Network
 - * WAN - Wide Area Network (1.18)
 - Mobilität || Leistung (1.19)
 - Konzepte - Layer-N-Gateway(1.20)
 - Beispiel (1.21)
- Internet(1.22 ff)
 - Internet
 - * Geschichte des Internet (1.24 ff)
 - * Normen (1.26)
 - Intranet (1.22)

1.2 Bitübertragungsschicht

1.2.1 Nachrichtentechnische Kanäle

- Aufgabe: Physikalische Bitübertragung mittels Transformation in elektromagnetisches Signal
- Daten \rightarrow Kanal \rightsquigarrow Störeinflüsse \rightarrow Daten

Kenngößen (2.4 ff)

- Bandbreite B: Breite des Frequenzbereichs eines Kanals, in dem ohne größere Dämpfung übertragen wird
- Baudrate
- Bitrate
- Nyquist Theorem $b < 2 \cdot B \cdot \lg(S)$
 - * Erweiterung durch Shannon $b < B \cdot \lg(1 + SNR)$
 - * Kombination $b < \min(2 \cdot B \cdot \lg(S) ; B \cdot \lg(1 + SNR))$

Leitungscode

- Wie soll Folge von 0en und 1en übertragen werden?
- NRZ: **N**on-**R**eturn-to-**Z**ero (2.6)
- Manchester-Codierung
- NRZI: NRZ-**I**nverted (2.7)
 - * Signaländerung bei 1, keine Signaländerung bei 0
 - * Vorteil: hohe Netto-Datenrate
 - * Nachteil: Probleme bei langer Folge von Nullen
 - * Lösung: 4B/5B Code
 - jeweils 4 Bits Daten werden auf 5-Bit-Muster abgebildet $\rightarrow 25$
 - durch 4B/5B-Code treten niemals mehr als 3 Nullen nacheinander auf

1.2.2 Übertragungsmedien

Elektrische Leitungen

- Twisted Pair (2.8)
 - isolierte Kupferdrähte von 0,4 bis 1mm Stärke
 - Paarweise verdreht \rightarrow Reduzierung von Störungen
 - Üblicherweise 4 Paare pro Kabel
 - Mehrere Kilometer Reichweite, mehrere MBit/s, preiswert
 - Signal aus Spannungsdifferenz zwischen den 2 Kabeln übertragen
 - Cat 3
 - Cat 5
 - Cat 6

- Cat 7
- Koaxialkabel (2.9)
 - mehrere km, mehrere MBit/s, T-stecker oder rTap
 - 50-Ohm-Kabel: für digitale Übertragung
 - 75-Ohm-Kabel: für analoge Übertragung und Kabelfernsehen
 - Kabelfernsehen → Breitband-Koaxialkabel, häufig mit analoger Übertragung bis ca. 1 GHz, bidirektionaler Ausbau für Internet-Zugang via Kabel

Optische Leitungen und Sichtverbindung

- Optische Leitungen
 - Lichtwellenleiter (LWL) / "Glasfaser"
 - * bis TBit/s-Bereich, über viele km Entfernung
 - * Monomodefaser: nur eine ausbreitungsfähige Wellenform
 - * Multimodefaser: verschiedene ausbreitungsfähige Wellenformen
 - * Gradientenfaser: schrittweise Änderung des Brechungsindex

Sichtverbindung

- Infrarotverbindung
- Richtfunkstrecken

Satelliten / Zellularfunk (2.11)

- Satelliten
 - Getrennte Aufwärts-/Abwärtsbänder
 - Bandbreite von 500MHz, z.B. in mehrere 50 MBit/s - Kanäle oder 800 digitale Sprachkanäle mit 64 kBit/s
 - Zuordnung kurzer Zeitabschnitte zu einzelnen Kanälen (Zeitmultiplex)
 - Lange Laufzeiten (ca. 250 bis 300ms)
- Zellularfunk
 - Aufteilung eines geographischen Bereichs in Funkzellen mit spezifischen Frequenzbändern
 - Beispiel: GSM (Global System for Mobile Communication)

Strukturierte Verkabelung (2.12)

- Ziel: Systematische, gut wartbare und erweiterbare Kabelinfrastruktur
- Trennung in drei wesentliche Bereiche (jeweils sternförmig hierarchisch)
 - Primärebene
 - Sekundärebene
 - Tertiärebene

1.2.3 Mehrfachnutzung von Kanälen

Frequenzmultiplex (2.13)

- getrennte Frequenzbänder (mit z.B. 3000 Hz) und zwischengeschaltete Sperrbänder (mit z.B. 500 Hz)

Orthogonales Frequenzmultiplex (Orthogonal FDM, OFDM)

- Überlagerung der Kanäle ohne Sperrbänder → effizienter
- Empfänger: Trennung der Signale mehrerer Bänder durch schnelle Fouriertransformation
- Einsatz: Wlan, Kabelnetze, 4G Mobilfunk, LTE, ...

Zeitmultiplex (2.14)

- Zyklische Kanalzuteilung

Statistisches Zeitmultiplex

- flexible Zuteilung nach Bedarf

Codemultiplex (CDM, 2.15)

- Didizierte (Kodierungs-)Codes pro Teilnehmerpaar

Wellenlängenmultiplex (WDM)

- Variation von Frequenzmultiplex, indem direkte optische Einkopplung mehrerer Lichtwellenleiter (mit Licht unterschiedlicher Wellenlängen) in einen besonders leistungsfähigen Lichtwellenleiter erfolgt
- entsprechende Wiederauskopplung im Zielsystem

1.2.4 Datenübertragung

Signalklassen (2.16)

- Wert/Zeit kontinuierlich \leftrightarrow Wert/Zeit diskret
- Beispiele (2.17)
 - Wert- und zeitkontinuierlich: analoges Telefon
 - Wertkontinuierlich, zeitdiskret: Prozesssteuerung mit periodischen Messpunkten
 - Wertdiskret, zeitkontinuierlich: digitale Temperaturanzeige
 - Wert- und zeitdiskret: digitale Übertragung mit isochronem Taktmuster; z.B. Sprachübertragung über digitale Kanäle

Beispiel: Telefonsystem (2.18)

Sprachübertragung über digitale Kanäle (2.19)

- Analoge Eingangssignale (Sprache) vor Übertragung im Kernnetz zu digitalisieren: Codec (Coder-Decoder)
- Basis: Abtasttheorem nach Shannon $f(A) > 2 \cdot f(G)$
- PCM: Pulse Code Modulation
 - Bsp.: Grenzfrequenz (Telefon) : 3400 z; Abtastfrequenz: 8000 Hz
 - logarithmische Quantisierungsintervalle \rightarrow Quantisierungsfehler begrenzen

Datenübertragung über analoge Kanäle

- Modem: Übertragung digitaler Signale über analoge (2.20) Telefonverbindung
 - Problem: Nicht direkt möglich wegen kapazitiver und induktiver Einflüsse
- Amplitudenabtastung
- Periodenabtastung
- Phasenabtastung
 - Ziel: Deutlich höhere Übertragungsleistung durch gleichzeitige Anwendung mehrerer Modulationsverfahren (2.21)
 - Beispiele
 - * QPSK
 - * QAM 16
 - * QAM 64

1.2.5 Beispieltechnologien

Digital Subscriber Line (DSL, 2.22)

- digitaler Netzzugang über herkömmliche Telefonleitungen
- Datenübertragung und Telefondienst gleichzeitig nutzbar
- Realisierung durch Nutzung höherer Frequenzbereiche
- hohe Datenraten, meist asymmetrisch (ADSL) bzgl. Up-/Downlink
- weitere Varianten:
 - VDSL (Very High Bitrate) : nur über kurze Entfernungen
 - SDSL (Symmetric): GLEICHE dATENRATE AUF Up-/Downlink
- Signaltrennung (Telefon/Daten) und Modulation (basierend auf QAM, 2.23)
 - CAS (Carrierless Amplitude / Phase System)
 - DMT (Discrete Multitone)

1.2.6 Digitaler Netzzugang über Kabelmodem

- Signaltrennung zwischen Kabelfernsehen und Daten:
 - Umwidmung einzelner TV-Kanäle in Datenkanäle
 - Rückkanalfähige Verstärker erforderlich
 - Datenraten theoretisch bis ca. 36 MBit/s, aber "SShared Medium", d.h. abhängig von der Zahl der Teilnehmer geringere Datenrate

Kapitel 2

Übung

2.1 Einführung

timo.schick@tu-dresden.de

2.1.1

- 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) (1) $l(n) = n$ bei Sterntopologie
(2) $l(n) = \sum \dots = (n * (n - 1)) / 2$ bei vollvermaschter Topologie
- d) (1) LAN
 - Reichweite: 10m
 - Reaktionszeit: niedrig
 - Datenrate: hoch
 - Topologien: Sterntopologie
- (2) MAN
 - Reichweite: 10km
 - Reaktionszeit: mittel
 - Datenrate: mittel
 - Topologien: hierarchische Topologie
- (3) WAN
 - Reichweite: 100km - 10.000km
 - Reaktionszeit: hoch
 - Datenrate: niedrig
 - Topologien: Vollvermaschte Topologie

2.1.2

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

2.1.3

- a)
- siehe Folie 1.15;
 - Initiator (Prozess A), ...
 - Responder (Prozess B), ...

b) (1) Zustände bestimmen

- idle
- connected
- prepare(Initiator)
- prepare(Responder)

(2) 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) (1) Ablaufdiagramm

- c1) + zeitlicher Ablauf
- c2) - es werden n Diagramme benötigt
- c3) -

(2) Zustandsdiagramm

- c1) -
- c2) + alle Abläufe in einem Diagramm darstellbar
- c3) +

2.1.4

a) siehe Folie 1.10

$$(1) PDU(N) = SDU(N - 1)$$

$$(2) IDU(N) = ICI(N) + SDU(N)$$

b) Seitenaufruf: <http://www.heise.de/software>

(1) httpRequest

- i. GET/software/http/1.1
- ii. Host: www.heise.de

(2) ICI

- i. ip: 193.99.144.85 port:80

(3) SDU

- i. GET/software/http/1.1
- ii. Host: www.heise.de

(4) IDU

- i. ICI
- ii. SDU

(5) TCP-PDU

- i. src:80, dest:80,...
- ii. SDU
- iii. Data

c)

$$b_0 = 125 \frac{\text{Mbit}}{\text{s}}$$

$$b_1 = b_0 \cdot 0,8$$

$$b_2 = b_1 \frac{(55 + 99)0,01}{2}$$

$$b_3 = b_2 \frac{(57 + 99)0,01}{2}$$

$$b_4 = b_3 \frac{(23 + 99)0,01}{2} = 36,4 \frac{\text{Mbit}}{\text{s}}$$

$$b_4 = b_{\text{goodput}}$$

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

Teil II

Theoretical Informatic and Logic

Kapitel 3

Vorlesung

3.1 Prädikatenlogik erster Stufe

- Syntax
 - Ein Alphabet der Prädikatenlogik besteht aus ... (2)
 - forall heist universeller Quantor, exists heißt existenzieller Quantor
 - Funktions- und Relationssymbolen ist eine Stelligkeit $n \in \mathbb{N}$
 - Nullstellige Funktionssymbole werden als ... (3)
- Terme
 - Definition 4.2 prädikatenlogische Terme (4)
 - Ein Term ist abgeschlossen oder grundinstanziiert, wenn in ihm keine Variablen vorkommen
 - Die Menge der abgeschlossenen Terme wird mit $T(F)$ bezeichnet
- Prädikatenlogische Atome (5)
- Prädikatenlogische Formeln (6)
 - prädikatenlogische Formeln
- Strukturelle Rekursion
 - Rekursionssätze lassen sich für $T(F, V)$ und $L(R, F, V)$ formulieren
 - Es gibt genau eine Funktion foo die die folgenden Bedingungen erfüllt: (7)
 - * Rekursionsanfang
 - * Rekursionsschritt
 - Beispiele (8/9)

3.2 Prädikatenlogik erster Stufe

- Strukturelle Induktion
 - Induktionssätze lassen sich für $T(F, V)$ und $L(R, F, V)$ formulieren
 - jeder Term besitzt die Eigenschaft E , wenn: (10)
 - analog für prädikatenlogische Formeln
- Aufgabe (11)
 - Beweisen Sie, dass $\forall F \in L(R, F, V)$ die Aussage $l'(m(F)) \geq l(F)$ gilt
- Teilterme und Teilformeln (12)
 - Die Def. 3.8 lässt sich auf Terme und Formeln übertragen
 - Beispiel
- Freie und gebundene Vorkommen einer Variablen (13)
 - Def. 4.5 Die **freien Vorkommen einer Variablen** in einer prädikatenlogischen Formel sind wie folgt definiert: (13)
- Abgeschlossene Terme und Formeln (14)
 - nach Def. 4.2: Ein abgeschlossener Term ist ein Term, in dem keine Variable vorkommt
 - Def. 4.6 Eine abgeschlossene Formel (oder kurz ein Satz) der Sprache $L(R, F, V)$ ist eine Formel der Sprache $L(R, F, V)$, in der jedes Vorkommen einer Variablen gebunden ist
- Substitutionen (19)
 - Def. 4.7: Eine **Substitution** ist eine Abbildung $\sigma : V \rightarrow T(F, V)$, die bis auf endlich viele Stellen mit der Identitätsabbildung übereinstimmt
 - Beispiel
- Instanzen
 - Statt $\sigma(X)$ schreiben wir in der Folge $X\sigma$
 - Def. 4.8: Sei σ eine Substitution $\sigma : V \rightarrow T(F, V)$ kann wie folgt zu einer Abbildung $\sigma_{dach} : T(F, V) \rightarrow T(F, V)$ erweitert werden: (25)
 - Grundinstanz
 - Proposition
- Komposition von Substitutionen

- Def. 4.10: Seien σ und θ zwei Substitutionen Die Komposition $\sigma\theta$ von σ und θ ist die Substitution: (30)
- Aufgaben

3.3 Prädikatenlogik erster Stufe

3.3.1 Komposition von Substitutionen

Korollar 4.11

Für jede Substitution σ gilt $\epsilon\sigma = \sigma = \sigma\epsilon$

Proposition 4.12

Seien σ und θ Substitutionen. Für jeden term t gilt $t(\hat{\sigma}\theta) = (t\hat{\sigma})\theta$
Beweis Strukturelle Induktion über $t \rightarrow$ Übung

Proposition 4.13

Sei $t \in T(F, V)$ und seien σ, θ sowie λ Substitutionen. Dann gilt:

- $t((\sigma\theta)\lambda)$
- $\sigma\theta\lambda = \sigma(\theta\lambda)$

Beweis siehe Folien (19)

3.3.2 Beschränkung von Substitutionen

Definition 4.14

Sei σ eine Substitution. Dann ist

$$\sigma_x = \begin{cases} \sigma & \text{wenn } X \notin \text{dom}(\sigma) \\ \sigma/\{X \mapsto t\} & \text{wenn } X \mapsto t \in \sigma \end{cases}$$

Proposition 4.15

Sei σ eine Substitution und t ein Term, in dem die Variable X nicht vorkommt.
Dann gilt: $t\sigma = t\sigma_x$

3.3.3 Anwendung von Substitutionen auf Formeln

Definition 4.16

Die Anwendung einer Substitution σ auf eine Formel ist induktiv über den Aufbau prädikatenlogische Formel wie folgt definiert:

- $p(t_1, \dots, t_n)\sigma = p(t_1\sigma, \dots, t_n\sigma)$
- $(\neg F)\sigma = \neg(F\sigma)$
- $(F \circ G)\sigma = (F\sigma \circ G\sigma)$ für jeden binären Junktor \circ
- $((QX)F)\sigma = (QX)(F\sigma_X)$ für jeden Quantor Q

Beobachtung

Bei der Anwendung einer Substitution auf eine Formel werden nur frei vorkommende Variablen ersetzt

Beweis: Übung

3.3.4 Substitutionen und Formeln

Definition 4.17

Eine Substitution σ ist genau dann frei für eine prädikatenlogische Formel F , wenn sie sich gemäß der folgenden Bedingungen als frei erweist:

- σ ist frei für F , wenn F ein Atom ist
- σ ist frei für $\neg F$ gdw σ ist frei für F
- σ ist frei für $(F \circ G)$ gdw σ ist frei für F und σ ist frei für G
- σ ist frei für $(QY)F$ gdw σ_Y ist frei für F und für jede von Y verschiedene und in F frei vorkommende Variable X gilt: Y kommt in $X\sigma$ nicht vor

3.3.5 Satz 4.18

Satz 4.18

Wenn die Substitution σ frei für die prädikatenlogische Formel F und die Substitution θ frei für $F\sigma$ ist, dann gilt: $F(\sigma\theta) = (F\sigma)\theta$

Beweis Satz 4.18

Strukturelle Induktion über F

- **IA** F ist Atom der Form $p(t_1, \dots, t_n)$

$$\begin{aligned}
 & p(t_1, \dots, t_n)(\sigma\theta) \\
 &= p(t_1(\sigma\theta), \dots, t_n(\sigma\theta)) && \text{Def 4.16} \\
 &= p((t_1\sigma)\theta, \dots, (t_n\sigma)\theta) && \text{Prop 4.12} \\
 &= p(t_1\sigma, \dots, t_n\sigma)\theta && \text{Def 4.16} \\
 &= p(t_1, \dots, t_n)\sigma\theta && \text{Def 4.16}
 \end{aligned}$$

- **IH** Das Resultat gilt für F

- **IS**

- **Fall** $\neg F$

Sei σ frei für $\neg F$ und θ frei für $(\neg F)\sigma$

Da σ frei für $\neg F$ ist, ist σ auch frei für F

Da θ frei für $(\neg F)\sigma$ und $(\neg F)\sigma = \neg(F\sigma)$ ist, ist θ auch frei für $F\sigma$

$$((\neg F)\sigma)\theta = (\neg(F\sigma))\theta = \neg((F\sigma)\theta) =_{(IH)} \neg(F(\sigma\theta)) = (\neg F)\sigma\theta$$

- **Fall** $(F \circ G) \rightsquigarrow$ Übung

- **Fall** $(\forall X)F$

Sei σ frei für $(\forall X)F$ und θ frei für $((\forall X)F)\sigma$

Da σ frei für $(\forall X)F$ ist, ist σ_X frei für F

Da θ frei für $((\forall X)F)\sigma = (\forall X)(F\sigma_X)$ ist, ist θ_X frei für $F\sigma_X$

Hilfsaussage $F(\sigma_X\theta_X) = F(\sigma\theta)_X$

Dann gilt:

$$\begin{aligned}
 & (((\forall X)F)\sigma)\theta \\
 &= ((\forall X)(F\sigma_X))\theta && \text{Def 4.16} \\
 &= (\forall X)((F\sigma_X)\theta_X) && \text{Def 4.16} \\
 &= (\forall X)(F(\sigma_X\theta_X)) && \text{IH} \\
 &= (\forall X)(F(\sigma\theta)_X) && \text{Hilfsaussage} \\
 &= ((\forall X)F)(\sigma\theta) && \text{Def 4.16}
 \end{aligned}$$

- **Fall** $\exists X)F \rightsquigarrow$ Übung

3.3.6 Beweis Hilfsaussage aus Satz 4.18

Unter den genannten Bedingungen gilt $F(\sigma_X\theta_X) = F(\sigma\theta)_X$

Beweis Da in F nur frei vorkommende Variablen ersetzt werden, genügt es zu zeigen, dass für jede frei in F vorkommende Variable Y gilt: $Y(\sigma_X\theta_X) = Y(\sigma\theta)_X$

- **Fall** $Y = X$

$$Y(\sigma_X\theta_X) = Y = Y(\sigma\theta)_X$$

- **Fall** $Y \neq X$

$$Y\sigma = Y\sigma_X \text{ und } Y(\sigma\theta) = Y(\sigma\theta)_X$$

Da σ frei für $(\forall X)F$ ist, kommt die Variable X in $Y\sigma$ nicht vor

Deshalb ist $(Y\sigma)\theta = (Y\sigma)\theta_X$

Dann gilt:

$$\begin{aligned} Y(\sigma_X\theta_X) &= (Y\sigma_X)\theta_X && \text{Prop 4.12} \\ &= (Y\sigma)\theta_X && (X \neq Y) \\ &= (Y\sigma)\theta && X \text{ kommt in } Y \text{ nicht vor} \\ &= Y(\sigma\theta) && \text{Prop 4.12} \\ &= Y(\sigma\theta)_X && (X \neq Y) \end{aligned}$$

3.3.7 Varianten

Definition 4.19

Seien E_1 und E_2 zwei Terme oder zwei prädikatenlogische Formeln. E_1 und E_2 heißen Varianten, wenn es Substitutionen σ und θ gibt, so dass $E_1 = E_2\sigma$ und $E_2 = E_1\theta$. In diesem Fall wollen wir E_1 auch als Variante von E_2 und E_2 als Variante von E_1 bezeichnen.

Wenn E_1 und E_2 Varianten sind und die in E_2 vorkommenden Variablen im bisherigen Kontext nicht verwendet wurden, dann ist E_2 eine neue Variante von E_1 .

Kapitel 4

Übung

4.1 Prädikatenlogik

Teil III

Computer Architecture

Kapitel 5

Vorlesung

5.1 Einführung

5.1.1 Big Data

"Big Data hat die Chance die geistige Mittelschicht in Hartz IV zu bringen"

5.2 Vorlesung

5.2.1 ZIH

- HAEC
- CRESTA Performance optimization
- MPI correctness checking: MUST
- Architecture of the new system (HRSK-II)

5.2.2 Begriffe und Definitionen

- Der Begriff Rechnerarchitektur wurde von dem englischsprachigen Begriff computer architecture abgeleitet
- Computer architecture ist eine Teildisziplin des Wissenschaftsgebietes computer engineering, welches die überwiegend ingeniermäßige Herangehensweise beim Entwurf und der Optimierung von Rechnersystemen deutlich zum Ausdruck bringt.
- Zwei Deutungen des englischen Begriffs "Architecture"
- Zur Definition der Rechnerarchitektur

- Architektur: Ausdruck insbesondere der Möglichkeiten der Programmierschnittstelle
 - * Maschinenbefehlssatz
 - * Registerstruktur
 - * Adressierungsmodi
 - * Unterbrechungsbehandlung
 - * Ein- und Ausgabe-Funktionalität
- Komponenten / Begriffsbildung
 - * Hardwarestruktur
 - * Informationsstruktur (Maschinendatentypen)
 - * Steuerungsstruktur
 - * Operationsprinzip
- Taxonomie
- Dreiphasenmodell zum Entwurf eines REchnersystems
 - Bottom-up (Realisierung → Implementierung → Rechnerarchitektur)
 - Top-down (Rechnerarchitektur → Implementierung → Realisierung)
 - Rückwirkungen durch den technologischen Stand

Kapitel 6

Übung

6.1 Einführung

Teil IV

Database

Kapitel 7

Vorlesung

7.1 Einführung

Gründe für DBS-Einsatz:

- Effizienz und Skalierbarkeit
- Fehlerbehandlung und Fehlertoleranz
- Mehrbenutzersynchronisation

ANSI - Database

- Standard siehe 1VL

Geschichte der Datenbanktechnologie

- siehe 1VL(28 ff.)

Databases vs Information Retrieval

- Information Retrieval 1VL(44)
 - Suche nach Dokumenten
 - Nimmt ständig zu
 - In welchem Datenbestand wird gesucht? etc...

Databases vs Big Data

- Big Data 1VL(47)

7.2 Konzeptueller Entwurf

7.2.1 Drei Phasen des Datenbank-Entwurfs (4, ff.)

Phasen der SW-Entwicklung

- Anforderungs-analyse → Vorstudie
- Fachentwurf → Fachknozept
- IT-Entwurf → IT-Konzept
- Implementierung → Module/Klassen/DB-Tabellen

Phasen des DB-Entwurfs

- nach Fachentwurf: fachliche Anforderungen an Datenstrukturen → Konzeptueller DB-Entwurf → Konzeptuelles Schema (ER-D, UML, etc.)
- nach IT-Entwurf: Entscheidung für logisches (Implementierungs-)Modell → Logischer DB-Entwurf → Logisches Schema (relational, OO, etc.)
- nach Implementierung: Umsetzung in konkretem System → Physischer DB-Entwurf → Physisches Schema (konkretes DBS)
- Datenbank = Schema + Daten

Datenbank = Schema + Daten

7.2.2 Lebenszyklus einer Datenbank

- Konzeptioneller Entwurf (12)
- Logischer Entwurf (13)
- Physischer Entwurf (14)
- Wartung, Modifikationen, Erweiterungen (14)
- Beispiel (15)

7.2.3 Prinzip eines Datenmodells (16)

- Grundlegendes Prinzip
- Leistung: Beschreibung
- Bestandteile
- Skizze (17)

7.2.4 Entity-Relationship-Modell

Entitäten (20)

- Definition
 - Existiert in der realen Welt, unterscheidet sich von anderen Entitäten
 - Eine Entität ist ein Objekt der realen oder der Vorstellungswelt, über das Informationen gespeichert werden sollen
 - Es ist im Sinne der Anwendung eindeutig beschreibbar und von anderen unterscheidbar
 - Gleichartige Entitäten werden zu Entitätstypen (Entitätsmengen) zusammengefasst
- Anmerkung
 - Welche Entitäten zusammengehören, ist von Semantik der Anwendung abhängig
- Merkmale von Entitätstypen (21)
 - Nur für die Anwendung relevante Merkmale werden modelliert
 - Beschreiben eine charakteristische Eigenschaft eines Entitätstyps
 - Werte eines Attributes aus Wertebereichen wie INTEGER, REAL, STRING
- Schlüsselattribut(e)
 - Ein Attribut oder eine Menge von Attributen, anhand deren Entitäten eines Entitätstyps unterscheiden lassen
 - Werden durch Unterstreichung gekennzeichnet
 - Beispiel: die ISBN-Nummer identifiziert das Buch

Beziehungen / Relationships (22)

- Abbildung von Zusammenhängen zwischen Entitäten
- Homogene Menge von Beziehungen wird zu Beziehungstyp zusammengefasst
- binär / n-när
- Kardinalitäten Titel \leftrightarrow Exemplar
- Bemerkungen
 - Ein Entitätstyp darf in einem Beziehungstyp mehrfach vorkommen
 - Mehr als zweistellige Beziehungstypen dürfen vorkommen
 - Beziehungstypen können auch Attribute besitzen

Beispiel eines ER-Diagramms (23)

Beispiel Funktionalitäten (24)

Funktionalität von Beziehungstypen (25)

- Beispiele (26 ff.)

Besonderheiten (32 ff.)

- Rolle
 - Anfrage an DB: "Gib mir alle Angestellten, die mehr verdienen als ihr Chef"
- Extended-ER
 - Weak Entities
 - * ID nur im Kontext eindeutig (Bsp.: Stuhlnummer in Hörsaal 003 \leftrightarrow Stuhlnummer in Hörsaal 004)
 - Strukturierte Attribute
 - * Min-Max Beziehung (35 ff.)

Entwurf eines ER Diagramms (38 ff.)

Varianten für mehrstellige Beziehungstypen (40)

Kapitel 8

Übung

8.1 Einführung

Teil V

Hardware Internship

Teil VI

C++4CG