

Block B (Software-Architektur): Systemarchitektur 16.11.2004 1

Vorlesung „Methoden des Software Engineering“

Block B „Software Architektur“  
**Systemarchitektur**

Dr. Harald Störrle

Einheit B.2, 16.11.2004

Methoden des Software Engineering (c) 2004, Dr. Harald Störrle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 2

### Gliederung Block B

Rahmen	A Anforderungen	Methodischer Entwurf
	B Architektur	Systemarchitektur
	C Formale Methoden	Middleware
	D Test, Validation, Verifikation	Muster & Komponenten
	E Prozeß	Formale Architektur-Beschreibungssprachen
	F Web-Engineering	

Methoden des Software Engineering (c) 2004, Dr. Harald Störrle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 3

### Gliederung / Themen heute

- **Systemarchitektur**
  - Was ist das?
  - Was hat das mit *Softwarearchitektur* zu tun?
  - Welche Arten von Systemarchitektur gab es historisch?
  - Wieso hat sich das geändert?
  - Wo geht die Reise hin?
- **Schichtenarchitektur**
  - Schichtung als Konstruktionsprinzip

Methoden des Software Engineering (c) 2004, Dr. Harald Störrle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 4

### Epochen der Systemarchitektur

1940: 1950: 1960: 1970: 1980: 1990: 2000: 2003: z900

1965: S/360-20 erster Großrechner 15.000 Stück verkauft 0,02 MIPS

1960: Minicomputer VAX, CDC, AS/400

1970: Host: optimale Ressourcenausnutzung

1980: GIS Skalierbarkeit, Leistung, GUI

1990: Thin Client TCO

2000: SOA Flexibilität

2003: z900 aktuell größte Maschine 8..10<sup>10</sup> MIPS

Methoden des Software Engineering (c) 2004, Dr. Harald Störrle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 5

### Softwaretechnische Gliederung Schichten („Layer“)

Layers (from top to bottom):

- Dialogdarstellung
- Dialogsteuerung
- Präsentation
- Fachlogik
- Persistenz
- Datenzugriff
- Datenhaltung
- GePros
- Fachobj. & -funktionen

Methoden des Software Engineering (c) 2004, Dr. Harald Störrle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 6

### Notationen

Notations:

- Knoten (Node)
- Prozess (Process)
- Komponente (Component)

Network Architecture Diagram:

- Zugangs Zone (Access Zone): Firewall 1, Switch, Web Server, Apache SSL Accelerator
- Dienste Zone (Service Zone): Firewall 2, Switch, Dialog Server, Applikations Server
- Backend Zone: Firewall 3, Switch, DB-DB2, Batch Server, Datenbank mit DB2 Server

Methoden des Software Engineering (c) 2004, Dr. Harald Störrle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 7

## Systemtechnische Gliederung Ebenen („Tier“)

ab 1965 Host- zentriert	ab 1980'er PC-Lösung	ab 1990'er Fat-Client	Klassische 3-tier Client-Server	moderne n-tier Thin-Client

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 8

## Moore's Gesetz

Die Integrationsdichte von ICs verdoppelt sich alle 12-18 Monate.

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 9

## Klassische Host-zentrierte Systemarchitektur

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 10

## Vom Host zu Client/Server: Hin...

- **Kosten**
  - Lizenzen
  - Skalierung
  - kostengünstige gemäßigte Ausfallsicherheit
- Der PC ermöglichte den Abteilungen größere Autonomie.
- Die Informatiker suchten die Herausforderung.
- C/S war schicker.
- IBM hat eine ungeschickte Politik betrieben.

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 11

## Skalierbarkeit

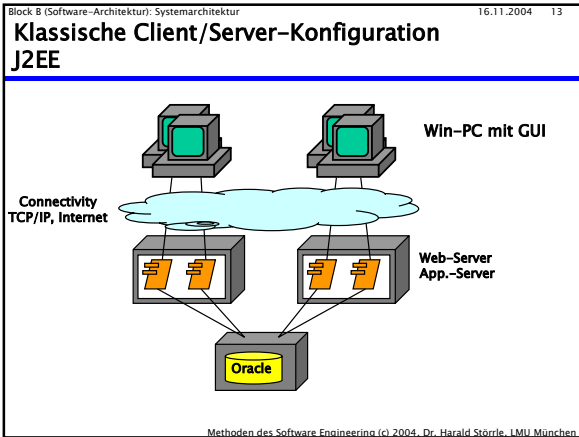
- Eine definierte Aufgabe kann von einer Recheneinheit des Typs R in k Zeiteinheiten bearbeitet werden.
- Die gleiche Aufgabe kann von n Recheneinheiten des Typs R in m Zeiteinheiten bearbeitet werden. Die Beschleunigung ist also k/m.
- Das Optimum ist k/n, wird aber wegen Fixkosten („Overhead“) und Artefakten (z.B. max. Zahl von DB-Verbindungen u.ä.) praktisch nicht erreicht. Es ist durchaus möglich, daß m > n.

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 12

## Klassische Client/Server-Konfiguration lokale Fat-Clients

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München



Block B (Software-Architektur): Systemarchitektur 16.11.2004 14

### Ausfallsicherheit – Grundlagen

- Seien  $X$  und  $Y$  zwei stochastisch voneinander unabhängige Zufallsvariablen, die die Verfügbarkeit eines Rechners in Prozent (bzw. als reelle Zahl zwischen 0 und 1) angeben.
- Wahrscheinlichkeit, daß...
  - beide verfügbar sind:  $X \cdot Y$
  - beide ausfallen:  $(1-X) \cdot (1-Y)$
  - genau einer ausfällt:  $(1-X) \cdot Y + X \cdot (1-Y)$

	$Y$	$\bar{Y}$
$X$		
$\bar{X}$		

Summe aller Felder: 1

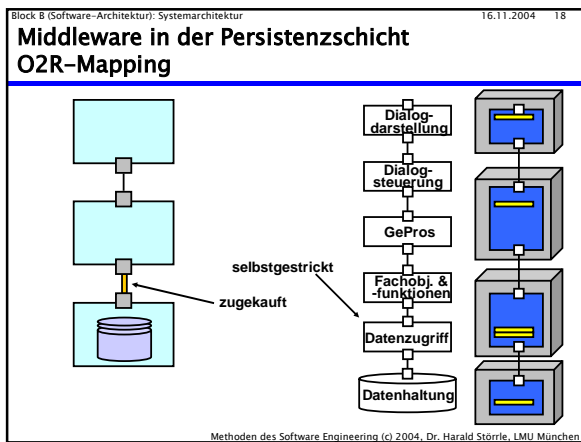
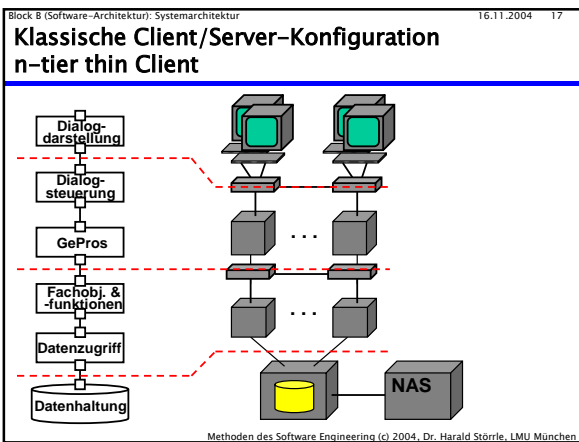
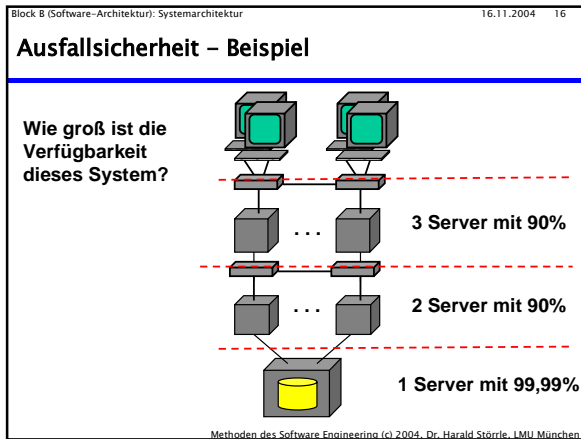
Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 15

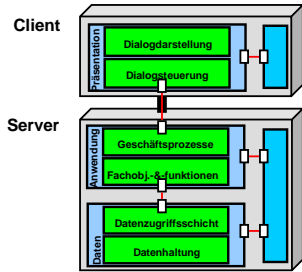
### Ausfallsicherheit – Anwendung

- Gestaltung eines Gesamtsystems
- 1) Redundante Verschaltung
  - Für Dienstbringung reicht ein funktionierender Server aus.
- 2) Schichtung
  - Für Dienstbringung müssen beide Server laufen

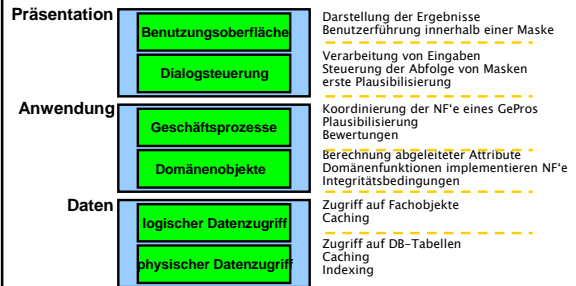
Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München



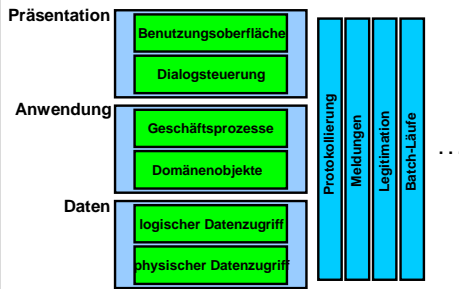
### Kapselung von Middleware



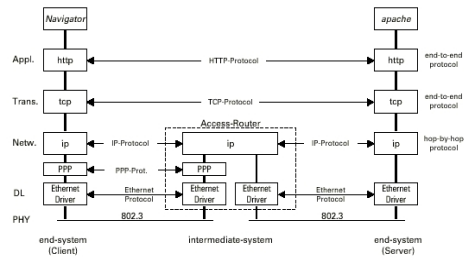
### Schichtenarchitektur Aufgaben der einzelnen Schichten



### Schichtenarchitektur Übergreifende Dienste

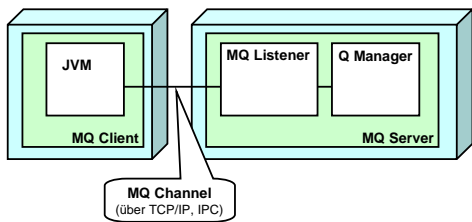


### Beispiel: ISO/OSI-7-Schichten-Modell



### MQSeries

- Paradigmatisches Beispiel für Message-Oriented Middleware
  - 85 % Marktanteil
  - auf 35 Plattformen verfügbar
  - heißt neuerdings WebSphere MQ



### Vom Host zu Client/Server: ...und zurück

- Kosten**
  - Stark steigende Betriebskosten rücken „Total Cost of Ownership“ (TCO) in den Mittelpunkt.
- IBM hat sich nach der Krise Mitte der 90'er gewandelt.**
- Im Internet-Zeitalter zählt extreme Ausfallsicherheit wieder**
  - Im 24-Stunden-Betrieb gibt es kein Batch-Fenster
  - Im 7-Tage-Betrieb gibt es kein Wartungswochenende
  - Im Online-Betrieb kostet jeder Ausfall Gewinn

Block B (Software-Architektur): Systemarchitektur 16.11.2004 25

## Total Cost of Ownership (TCO)

- Die Kosten eines Systems über seine gesamte Lebensdauer setzen sich aus verschiedenen Faktoren zusammen.

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 26

## Relevante Themen im Betrieb von Systemen

- Ausfallsicherheit / Verfügbarkeit
- Durchsatz / Leistung / Skalierbarkeit
- Security / Safety
- Backup / Speicherlösungen
- Stapelverarbeitung („Batch-Betrieb“)
- Quality-of-Service (QoS)

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 27

## Betriebskostenanteil vs. Lebensdauer

Anteil der Betriebskosten an den gesamten Kosten [%]

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 28

## Bedeutung von Großrechnern heute

### Verbreitung

- 90 % der weltweit größten Unternehmen setzten S/390 als zentralen Server ein.
  - [Karlheinz Kronauer, Marketing Manager Software AG in CZ 21.7.2003]
- ca. 2/3 aller weltweit relevanten wirtschaftlichen Daten sind in EBCDIC-Format gespeichert
- 60 % aller aus dem Web aufrufbaren Daten befinden sich auf Hosts
- 16000 Unternehmen setzen CICS als Transaktionsmonitor ein
- führende Beratungsunternehmen wie z. B. Gartner Group bescheinigen der S/390-Hardware eine Spitzenstellung unter den Rechnern

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 29

## Bedeutung von Großrechnern heute

### Die Meinung der Gartner Group

Hersteller Rechner Betriebssystem	IBM S/390 OS/390	SUN Exxk Solaris	HP 9000 HP-UX	Compaq Alpha Tru64	Proliant NT 4.0
System Performance	15	15	15	12	6
Clustering Performance	5	2	2	3	1
Single System Availability	15	9	9	9	3
Multiple Systems Availability	15	9	12	12	3
Workload Management	15	6	9	6	3
Partitioning	10	6	2	2	2
Systems Management	10	6	8	6	6
<b>Summe</b>	<b>85</b>	<b>53</b>	<b>57</b>	<b>50</b>	<b>24</b>

Studie der Gartner-Group, 1999 Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 30

## Gegenüberstellung der Technik

### Host / Unix

	Host	Unix
Ursprung	EnterpriseComputing	WorkgroupComputing
Batch-Verarbeitung	JCL und RZ-Controllsysteme	Kaum, Script-Lösungen (SHELL oder PERL)
Online	TM IMS und CICS	Demonprogrammierung 1 Benutzer 1 Prozess Eigensprogrammierung Kaum TM (Fuzeol)
Oberflächen	Zeichenorientiert (3270) Browser	Zeichenorientiert (VT100) GUI (Windows) Browser
Programmiersprachen	PLI, COBOL, Assembler C/C++ u. Java (sehr wenig)	C/C++ u. Java
Systemcalls	Strikte Trennung von System- und Anwendungsprogrammierung	Mischung aus beiden
Workloadmangement	Integriert in OS/390 homogener Rechnerverbund (Systech)	Häufig heterogene Clusterlösungen
Verfügbarkeit	> 99,9 %	< 99
Datenbanken	IMS/DB, DB2	Oracle, Informix, DB2, ...
Präsentation	3270-Terminals	graphische Fenstermanager
Ablaufumgebung	CICS, IMS	unüblich (Fuzeol, App-Server)

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 31

## Vom Host zu Client/Server: ...und weiter Richtung SOA

- **SOA = Service Oriented Architecture**
- **Host-Architektur**
  - ist leicht zu betreiben,
  - aber ziemlich starr und nicht gerade billig
- **klassische C/S-Architektur**
  - ist sehr flexibel und skalierbar
  - aber sehr aufwändig in Herstellung, Betrieb und Wartung
- **Service Oriented Architecture**
  - flexibel und skalierbar
  - und abstrahiert von Plattform
    - Integration/Konsolidierung nach Fusion, Kooperationen
    - Controlling der Betriebskosten, Outsourcing
  - aber großer Overhead und hohe Umstrukturierungskosten und unsicherer QoS

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 32

## Web Services

The diagram illustrates the Web Services architecture. On the left, a network of yellow ovals represents various service providers or consumers. On the right, a stack of components is shown: 'Vermittlung' (mediation) at the top, followed by 'Web-Service' components. A 'UDDI' registry is connected to the stack, likely for service discovery. Red arrows indicate the flow of data or service requests between the network and the service stack.

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München

Block B (Software-Architektur): Systemarchitektur 16.11.2004 33

The diagram shows the relationship between software engineering phases and architectural components. On the left, a vertical stack of phases is labeled 'Rahmen' (Framework) and includes:
 

- A Anforderungen (Requirements)
- B Architektur (Architecture)
- C Formale Methoden (Formal Methods)
- D Test, Validation, Verifikation (Testing, Validation, Verification)
- E Prozeß (Process)
- F Web-Engineering (Web Engineering)

 On the right, a horizontal stack of components is shown:
 

- Methodischer Entwurf (Methodical Design)
- Systemarchitektur (System Architecture)
- Middleware
- Muster & Komponenten (Patterns & Components)
- Formale Architektur-Beschreibungssprachen (Formal Architecture Description Languages)

 Green arrows indicate the flow of information or influence from the phases to the components.

Methoden des Software Engineering (c) 2004, Dr. Harald Störle, LMU München