

X ADMINISTRATIVES

- X.1 Kontakt
- X.2 Struktur der Veranstaltung
- X.3 Organisation: Vorlesung + Übung + Klausur
- X.4 Organisation: Moodle
- X.5 Vorlesung: Inhalt
- X.6 Literatur

Prof. Dr. Rainer Mueller

Raum: O 204

Sprechstunde: Nach Vereinbarung (→ einfache E-Mail genügt)

E-Mail: rainer.mueller@htwg-konstanz.de

STRUKTUR

Studienordnung	WIN SPO2/SPO3
Semester	4-7
Vorlesung	2 SWS
Prüfung	Benotete Modulteilprüfung (Klausur)
Übung	2 SWS
Prüfung	Unbenotete Modulteilprüfung (Schein)

ÜBUNG

- Gruppenarbeit (max. 2) oder Einzelarbeit je nach Studentenzahl
- Übungen und Aufgaben in Java auf Windows-Plattform

Vorlesung (2 SWS)	
Zeitpunkt	Mittwochs, 09:45-11:15 Uhr
Raum	O 102
Termine	Jede Woche (Ausnahme: evtl. 19.04.)
Prüfungsform	Klausur K90

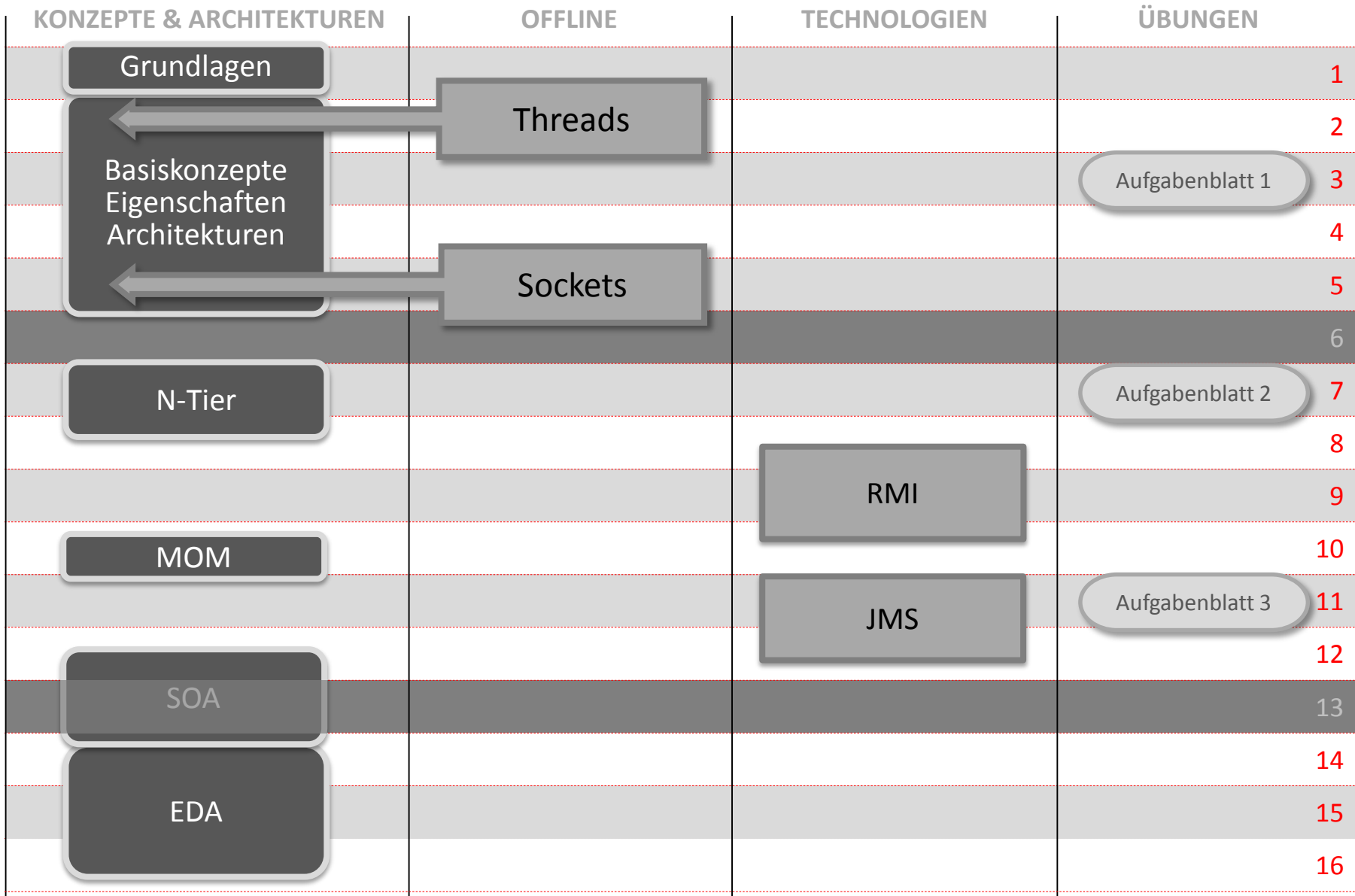
Übung (2 SWS)	
Zeiten	Gruppe 1: Mittwochs, 14:00-15:30 Uhr, Gruppe 2: Mittwochs, 19:00-20:30 Uhr (Alt.: 17:30-19:00 Uhr zusammen mit BESY)
Raum	O 008
Termine	Jede Woche, ab 28.03.
Gruppeneinteilung	Abstimmung zur Übungsteilnahme in Moodle: bis 19.03. (Achtung Maximalzahl pro Gruppe: 15) https://moodle.htwg-konstanz.de/moodle/mod/choice/view.php?id=91763 <i>Einschreibeschlüssel: 120317_1430</i>
Prüfungsform	Übungsschein (Gruppenarbeit in Zweiertteams möglich)

Klausur

Zeitpunkt	01.07.-26.07. (<i>Genauer Zeitpunkt wird bekannt gegeben</i>)
Raum	<i>(wird bekannt gegeben)</i>
Dauer	90 min
Themen	Vorlesung

Moodle

URL	https://moodle.htwg-konstanz.de/moodle/course/view.php?id=2992 <i>Einschreibeschlüssel: 120317_1430</i>
Kurs	„VSYS Verteilte Systeme SS 2017“
Inhalte	Folien, Übungsblätter, organisatorische Informationen



OUT OF SCOPE – THEMEN, DIE NICHT EXPLIZIT BEHANDELT WERDEN

- CORBA (**C**ommon **O**bject **R**quest **B**roker **A**rchitecture)
- JINI
- DCOM (**D**istributed **C**omponent **O**bject **M**odel)
- Virtualisierung
- Mobile Computing
- Ubiquitous Computing
- Cloud Computing
- SAAS (**S**oftware **A**s **A** Service), IAAS, PAAS
- ...

- G. Coulouris, J. Dollimore, T. Kindberg; Verteilte Systeme: Konzepte und Design; Addison-Wesley; 2002
→ *Bib ibe/2/40*
Der „Stroustrup“ od. „Kernighan/Ritchie“ für verteilte Systeme 
- O. Haase; Kommunikation in verteilten Anwendungen; Oldenbourg-Verlag; 2008
→ *Bib ibe/72*
Sehr praxisnahes, recht neues Buch aus eigenem Haus 
- A. S. Tanenbaum, M. van Steen; Verteilte Systeme: Grundlagen und Paradigmen; Prentice Hall; 2003
→ *Bib ibe/2/44c*
Das Standardwerk
- J. Dunkel, A. Eberhart, S. Fischer, C. Kleiner, A. Koschel; System-Architekturen für verteilte Anwendungen; Hanser-Verlag; 2008
→ *Bib ibe/2/53*
Sehr modernes, architektur-bezogenes Buch 
- A. Puder, K. Römer, F. Pilhofer; Distributed Systems Architecture; Morgan Kaufmann by Elsevier; 2006
→ *Bib ibe/2/51*
Der Middleware-Ansatz
- R. Oechsle; Parallele und verteilte Anwendungen in Java; Hanser-Verlag; 2011
→ *Bib ibe/72/286c*
Praxisnahes, java-orientiertes und neues Buch 



X ADMINISTRATIVES



- X.1 Kontakt



- X.2 Struktur der Veranstaltung



- X.3 Organisation: Vorlesung + Übung + Klausur



- X.4 Organisation: Moodle

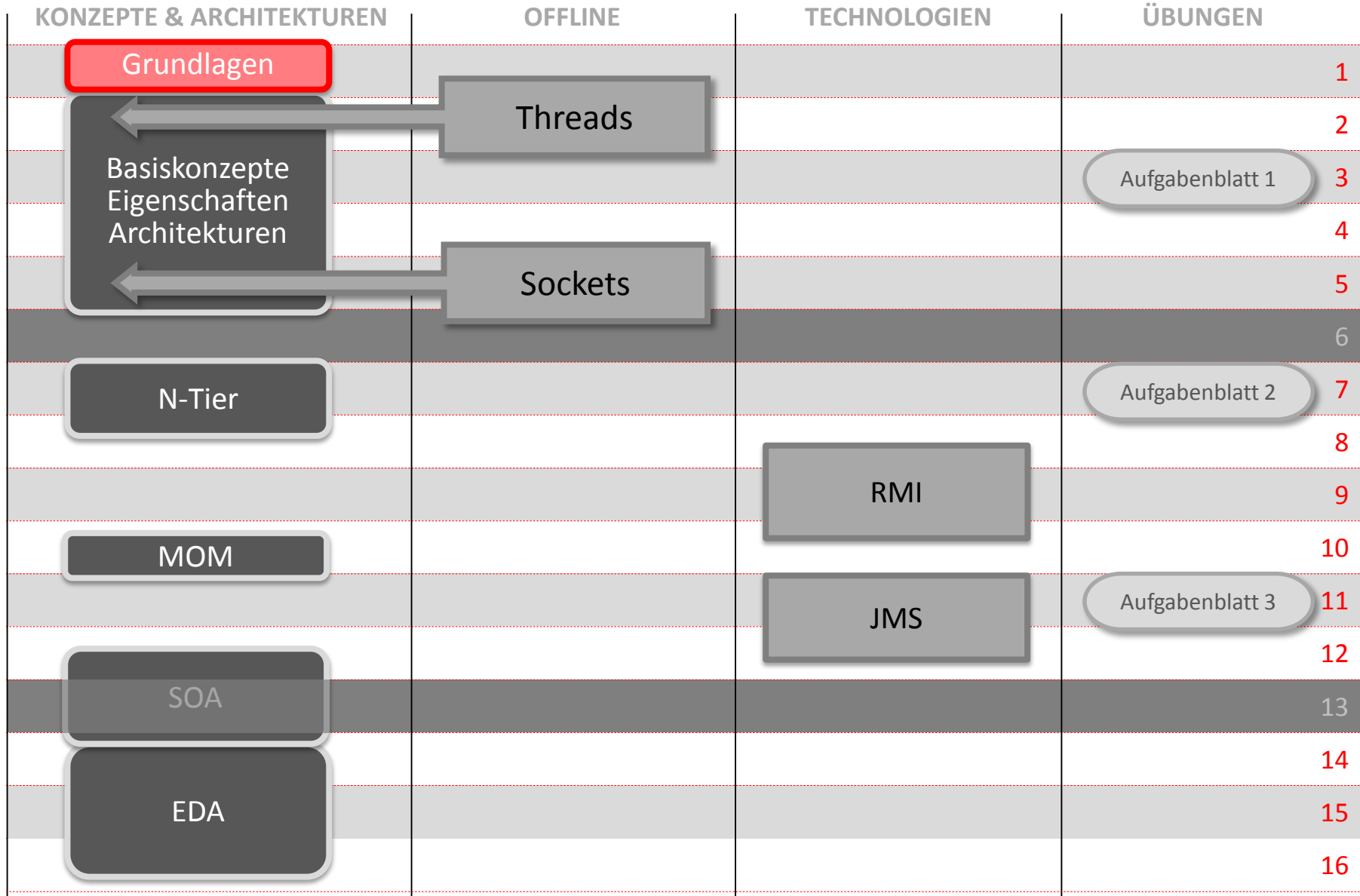


- X.5 Vorlesung: Inhalt



- X.6 Literatur

Vorlesung: Übersicht



KAPITEL 1

Grundlagen

1.1 MOTIVATION

- 1.1.1 Lokale Schranken
- 1.1.2 Netzausbau
- 1.1.3 Verteiltere Welt
- 1.1.4 Technologischer Fortschritt

1.2 DEFINITION

- 1.2.1 Zitate
- 1.2.2 Verteilt vs. zentralisiert

1.3 KONSEQUENZEN, VORTEILE UND NACHTEILE

- 1.3.1 Konsequenzen aus der Definition
- 1.3.2 Unterstützung verteilter Aufgabenstellungen
- 1.3.3 Robustheit, Zuverlässigkeit, Verfügbarkeit
- 1.3.4 Unterstützung von Spezial-Hardware
- 1.3.5 Ausnutzung freier Rechenressourcen
- 1.3.6 Kosten für Kommunikation
- 1.3.7 Komplexität bei Entwicklung
- 1.3.8 Komplexität bei Betrieb

FLASCHENHALS: ISOLIERTES LOKALES SYSTEM

- Zentrale Systemfunktionen und zentrale Datenstrukturen bilden Schranken

Zentrale Systemfunktionen und zentrale Datenstrukturen sind ein Flaschenhals in verteilten Systemen. Hohe Verfügbarkeit und Knotenautonomie sind neben dem Leistungsengpass weitere Gründe, die gegen zentrale Dienste sprechen.

Quelle: J. Anton Illik, Verteilte Systeme und Software-Architekturen

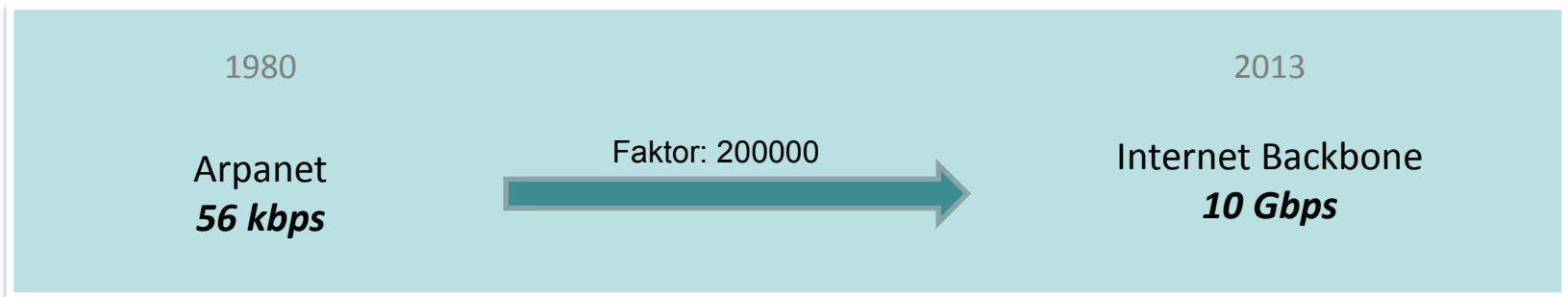
- Nicht das Netz ist der Flaschenhals

Nicht das Netz, sondern der Computer ist oftmals der Flaschenhals. Bandbreite lässt sich nahezu beliebig steigern, für Wartezeiten (Latency) gibt es Untergrenzen: Nachricht komplett optisch über 20000 km mit Latenz 70 ms.

Quelle: Oliver Haase, Verteilte Systeme

- **Fazit zu zentralisierten Diensten**
 - Wenig performant und fehleranfällig
 - Netz ist robust und zunehmend performanter

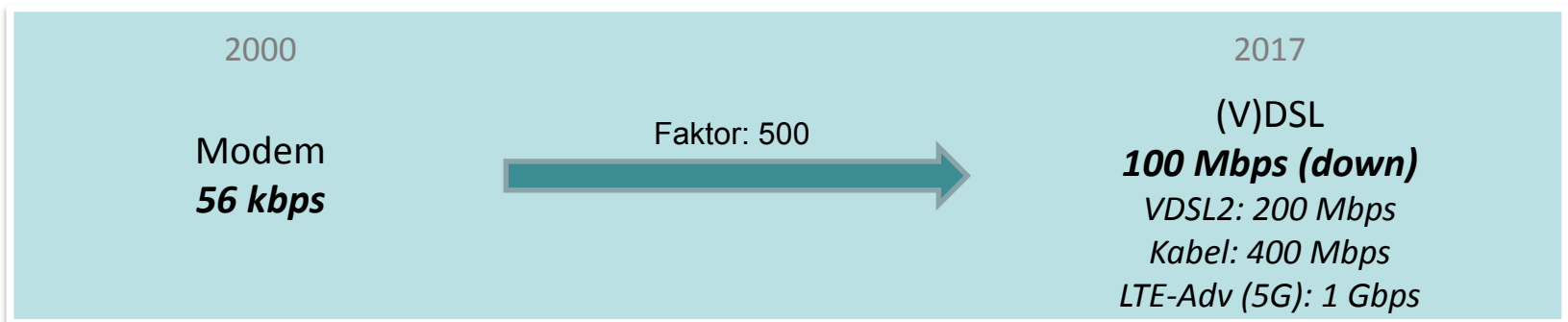
BANDBREITE INTERNET BACKBONE



FEHLERRATE



BANDBREITE ZUGANGSNETZ



GLOBALISIERUNG

- **Zunehmende Team-Verteilung und -Distanz:** Unterschiedlichere Standorte; unterschiedlichere Zeitzone
- **Heterogenere Projektwelt:** Projekt-orientierteres Geschäftsfeld; Heterogenität in Bezug auf Budget, Laufzeit und Fachgebiet; viele parallele Projekte mit unterschiedlichen Mitarbeitern
- **Ansteigende Mobilität:** Kundenentfernung; Entfernung zum Arbeitsort

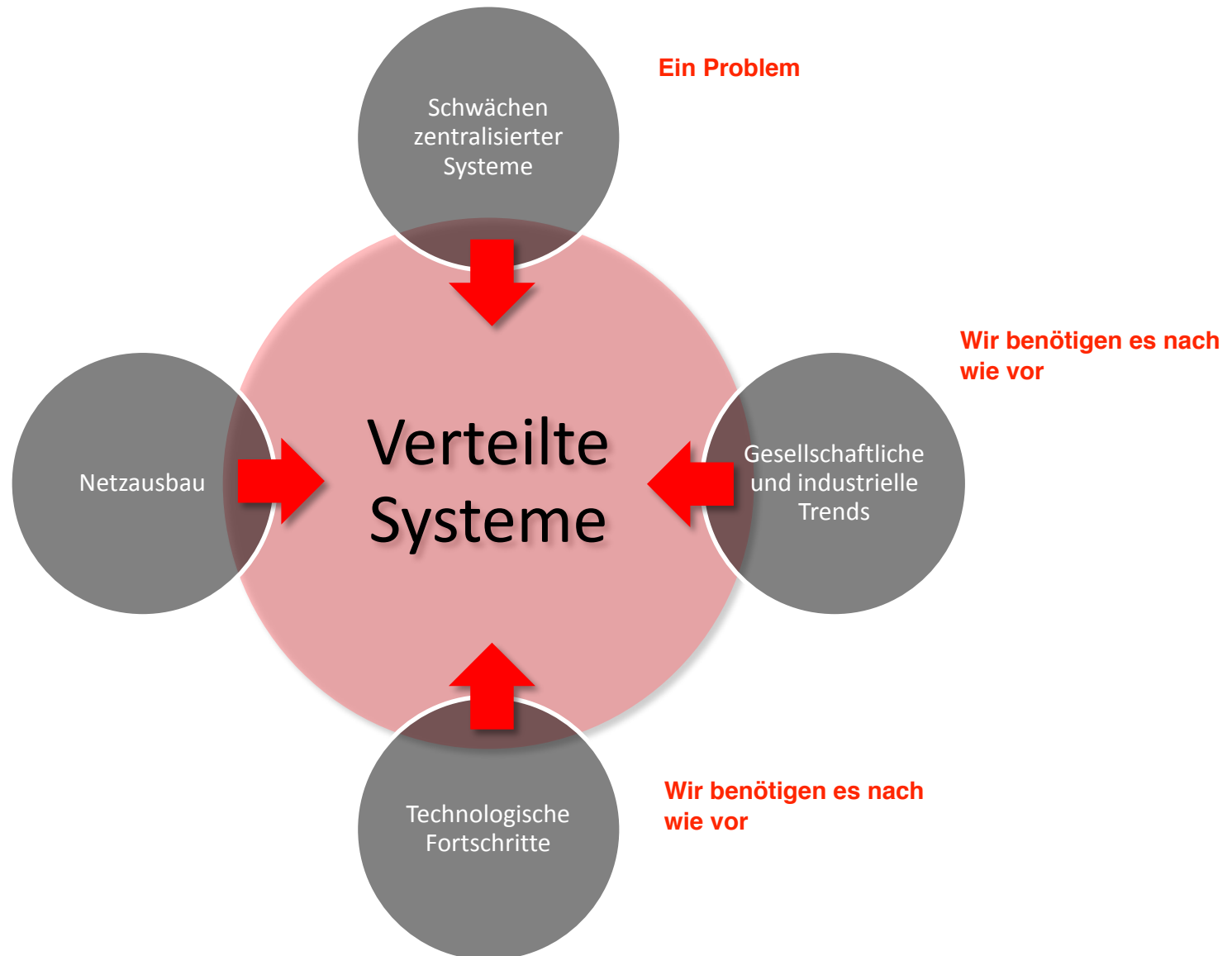
ALLGEWÄRTIGES INTERNET/WEB

- Internet und Web im Wandel **vom Informationsnetz zur Plattform für alles**
 - Web-Applikationen ersetzen lokale Anwendungsinstallationen
 - Web-Browser ersetzen lokale Betriebssysteme

TREND GEGEN PLATTFORMABHÄNGIGKEIT

- Kosten für Software-Hersteller: Portierung auf verschiedene Plattformen
- Kosten für Software-Anwender: Installation, Schulung und Wartung auf verschiedene Plattformen
 - Plattformunabhängigkeit durch Web-Clients
 - Interaktivität (Look&Feel) durch Rich-Client-Prinzip → Bsp.: GWT, PrimeFaces, Angular

FAZIT: TECHNOLOGISCHE/GESELLSCHAFTLICHE ENTWICKLUNG PASSEND ZUM ANGEBOT



✓✓✓✓✓ 1.1 MOTIVATION

- 1.1.1 Lokale Schranken
- 1.1.2 Netzausbau
- 1.1.3 Verteiltere Welt
- 1.1.4 Technologischer Fortschritt

1.2 DEFINITION

- 1.2.1 Zitate
- 1.2.2 Verteilt vs. zentralisiert

1.3 KONSEQUENZEN, VORTEILE UND NACHTEILE

- 1.3.1 Konsequenzen aus der Definition
- 1.3.2 Unterstützung verteilter Aufgabenstellungen
- 1.3.3 Robustheit, Zuverlässigkeit, Verfügbarkeit
- 1.3.4 Unterstützung von Spezial-Hardware
- 1.3.5 Ausnutzung freier Rechenressourcen
- 1.3.6 Kosten für Kommunikation
- 1.3.7 Komplexität bei Entwicklung
- 1.3.8 Komplexität bei Betrieb

- *A collection of independent computers that appears to its users as a single coherent system.*

Quelle: Andrew S Tanenbaum, Verteilte Systeme, Grundlagen u. Paradigmen

- *A distributed system is one in which components located at networked computers communicate and coordinate their actions only by passing messages.*

Quelle: Coulouris et al., Distributed Systems – Concepts and Design

- *A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable.*

Quelle: Leslie Lamport

Wir wollen verteilte Systeme die man nicht merkt -> laut Müller nicht möglich, da diese sonst keinen Sinn machen.

	Zentrale Anwendung	Verteilte Anwendung
Prozesse	Einer oder mehrere	Mehrere
Vernetzte Rechner	Einer	Mehrere
Prozesse/Rechner	Mehrere (ggf. auch nur einer)	Einer (ggf. auch mehrere)

Der Unterschied ist, dass zentrale Anwendungen auf einem Rechner verlaufen, verteilte Anwendungen aber auf mehrere

✓ 1.1 MOTIVATION

- ✓ • 1.1.1 Lokale Schranken
- ✓ • 1.1.2 Netzausbau
- ✓ • 1.1.3 Verteiltere Welt
- ✓ • 1.1.4 Technologischer Fortschritt

✓ 1.2 DEFINITION

- ✓ • 1.2.1 Zitate
- ✓ • 1.2.2 Verteilt vs. zentralisiert

1.3 KONSEQUENZEN, VORTEILE UND NACHTEILE

- 1.3.1 Konsequenzen aus der Definition
- 1.3.2 Unterstützung verteilter Aufgabenstellungen
- 1.3.3 Robustheit, Zuverlässigkeit, Verfügbarkeit
- 1.3.4 Unterstützung von Spezial-Hardware
- 1.3.5 Ausnutzung freier Rechenressourcen
- 1.3.6 Kosten für Kommunikation
- 1.3.7 Komplexität bei Entwicklung
- 1.3.8 Komplexität bei Betrieb

Essenz aus der Definition „Verteiltes System“: **Definition Verteilte Systeme!**

Komponenten auf vernetzten Rechnern kommunizieren (ausschließlich über Nachrichten)

NEBENLÄUFIGKEIT

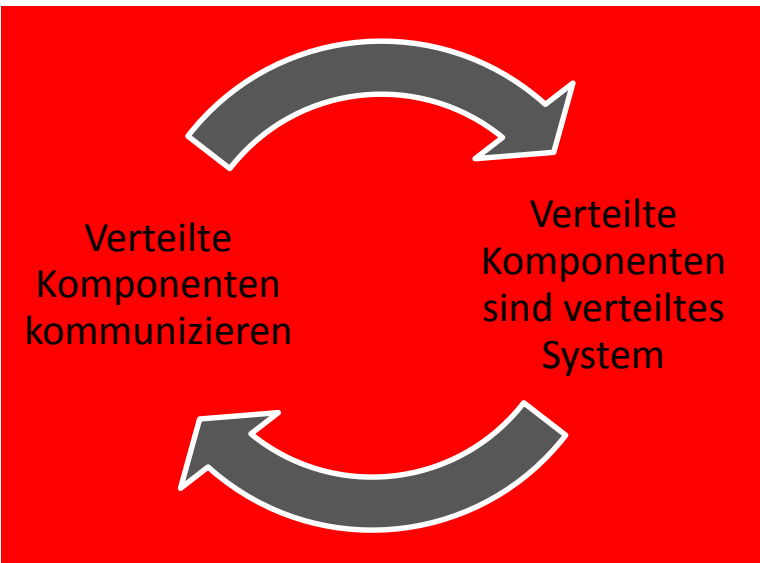
- Nebenläufige Ausführung von Komponenten mit geteiltem Ressourcenzugriff
 - Synchronisierung der Komponenten erforderlich
 - Zusätzliche Ressourcen steigern Gesamtkapazität des verteilten Systems

FEHLENDE GLOBALE UHR

kaum möglich da Zeitpunkte nicht exakt
bestimmbar wegen Übertragungsver-
zögerung

- Kein globales Zeitkonzept: Synchronisierung der Uhren auf Teilnehmersystemen schwierig
 - Koordination von Komponenten nur über Nachrichteninhalte, nicht über Nachrichtenzeitpunkt
 - Keine Koordination von Komponenten ohne Nachrichten (also z.B. nur durch Zeit)

nur durch Kommunikation möglich
z.B. durch kooperierenden Threads



Ende 15.3 KOMPONENTENAUSFALL

Vortsetzung Folie

2

- Komponenten fallen unabhängig aus: Andere Komponenten funktionieren weiter
 - Komponenten bemerken Ausfall nicht und reagieren nicht darauf
 - Einzelkomponenten isoliert
 - Gesamtsystem wird langsamer oder fällt aus

warten dauert zu lang und
kommt gar nicht vor