

# Verteilte Systeme

Sommersemester 1999  
Universität Trier  
Peter Sturm

## Abstract

In einem verteilten System kommunizieren mehrere Prozesse durch den Austausch von Nachrichten. Ausführungsplattform sind dabei viele, über ein Kommunikationsnetz verbundene Computer. Die Vorteile einer verteilten Realisierung von Systemsoftware und Anwendungsprogrammen sind:

- Inkrementelle Erweiterbarkeit
- Leistungssteigerung durch Parallelarbeit
- Erhöhung der Ausfallsicherheit durch Redundanz

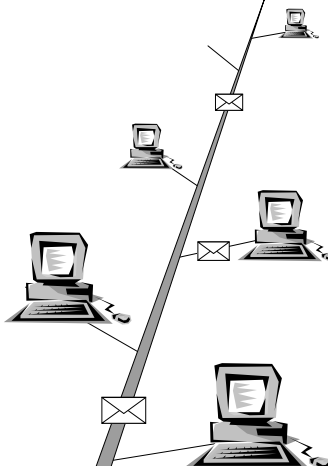
Die Realisierung dieser Systeme wird jedoch durch eine Reihe prinzipieller Beschränkungen wie z.B. dem Fehlen eines globalen Zeitbegriffs und einer globalen Systemsicht erschwert. In dieser Vorlesung werden die Möglichkeiten und Grenzen verteilter Systeme vorgestellt.

Vorlesungsinhalte:

- Eigenschaften von Computernetzen
- Kommunikationsprotokolle (IPv4, IPv6, ...)
- Verteilte Algorithmen
- Fehlertoleranz
- Lastverteilung
- RPC und Client/Server-Systeme
- Verteilte Ausführungsplattformen (PVM, MPE, DCOM, CORBA, ...)
- Middleware
- Alternative Ansätze (Verteilter gemeinsamer Speicher, ...)

### Was ist ein verteiltes System?

- Rechnernetz
  - Mono- und Multiprozessorsysteme
  - Kommunikationsnetzwerk
- Kein gemeinsamer Speicher !
- Senden und Empfangen von Daten
  - Nachrichten
- Eigenschaften
  - Inkrementelle Erweiterbarkeit
  - Hohe Nominalleistung



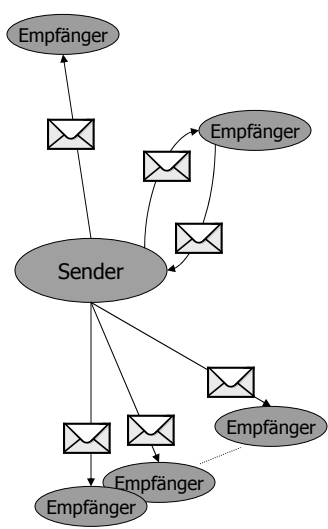
Wieviele Rechner mit jeweils 100 MIPS bilden ein Rechnernetz mit einer scheinbaren Instruktionszeit von 1 Pikosekunde ( $10^{-12}$  s)?

Was würde ein Monoprozessor mit einer solchen Taktfrequenz kosten?

Verteilte Systeme, Sommersemester 1999 Folie 3

### Nachrichtenbasierte Kommunikation

- „Größe“ von Sender und Empfänger
  - Prozeß
  - Thread
  - Komponente (DCOM, Corba, JavaBean, ...)
  - Objekt
- Nachricht
  - Untypisiert (n Bytes)
  - Enthält Typinformation
  - Enthält ausschließlich Daten (Data Shipping)
  - Enthält auch Code (Data Shipping)
- Vielfältige Interaktionsmuster
  - Mitteilung, Auftrag
  - Synchron, Asynchron
  - Direkt oder Indirekt (Port, Mailbox, ...)
  - Unicast, Multicast, Broadcast
  - ...



Verteilte Systeme, Sommersemester 1999 Folie 4

## Vorteile und herausfordernde Probleme

### Vorteile

- Leistungssteigerung durch Parallelarbeit
- Erhöhte Verfügbarkeit
- Inhärent verteilte Anwendungen

### Probleme

- Konsistente globale Zustände
- Fehlen einer globalen Uhr
- Komplexität

### Eigenschaften

- Übertragungsgeschwindigkeit ist endlich
- Nachrichten sind bei der Ankunft bereits veraltet



Verteilte Systeme, Sommersemester 1999

Folie 5

## Vorteil 1: Leistungssteigerung durch Parallelarbeit

### Erzielbarer Leistungsgewinn

- Rechenzeit auf Monoprozessor:  $T_1(n)$
- Rechenzeit auf k Monoprozessoren:  $T_k(n)$
- Maximaler Speedup:  $\frac{T_1(n)}{T_k(n)} \leq k$

Wie definiert man die Rechenzeit bei k Monoprozessoren?

Warum ist der Speedup meist ein gutes Stück kleiner als k?

### Partitionierung der Aufgabe

- Möglichst hoher Rechenbedarf innerhalb einer Partition
- Möglichst geringer Kommunikationsbedarf zwischen Partitionen

### Genügend Rechenlast

Verteilte Systeme, Sommersemester 1999

Folie 6

### Beispiel: Verteiltes Raytracing

---

Identische Szenenbeschreibung auf allen Rechnern

- u.U. umfangreiche Daten
- Zentraler Ursprung
- Effiziente Verteilung

Sinnvolle Partitionierungsansätze?

Erreichbarer Speedup?

Szene

Abbildungsfläche

Verteilte Systeme, Sommersemester 1999 Folie 7

### Partitionierung entscheidet über Speedup

---

Vergleichbarer Rechenaufwand in jedem Abschnitt

Speedup  $k$  prinzipiell erreichbar

Wie ermittelt man eine geeignete Partitionierung?

Rechner 1 	Rechner 2 
Rechner 3 	Rechner 4 

Starkes Ungleichgewicht beim Rechenaufwand

Unter Umständen Speedup  $< 1$

- Zusätzlicher Initialisierungsaufwand
- Zusammenfassen der Einzelresultate


Verteilte Systeme, Sommersemester 1999 Folie 8

### Vorteil 2: Erhöhte Verfügbarkeit (Zuverlässigkeit)

Verteiltes System = Redundanz

- Rechnern
- Kommunikationsnetzen

Grundidee

- Einzelne Komponente  $k$  fällt mit einer Wahrscheinlichkeit  $p$  aus
- Ausfallwahrscheinlichkeit bei  $n$  Replikaten von  $k$ :

$$p^n < p$$

Voraussetzung für  $p^n < p$ ?

Verteilte Systeme, Sommersemester 1999 Folie 9

### Beispiel: Zuverlässiger Bank-Server

Zustandsändernde Transaktionen

Realisierung 1:  
Wir vertrauen der Software!

Realisierung 2:  
Toleranz gegenüber Software-Fehlern?

Verteilte Systeme, Sommersemester 1999 Folie 10

### Vorteil 3: Inhärent verteilte Anwendungen

---

Räumlich verteilte Partner

Trivial: Asymmetrische Lösung

- Zentraler Server
- „Dump Clients“

Vor- und Nachteile?

Symmetrische Lösung

- Zustände a priori verteilt
- Nebenläufige Kontrollflüsse isolierbar

Vor- und Nachteile?

Das Diagramm zeigt zwei Szenarien. Oben ist eine asymmetrische Lösung dargestellt: Ein zentraler Server (Server) ist mit einem Netzwerk (Netzwerk) verbunden, das wiederum mit mehreren Clients (Client) verbunden ist. Unten ist eine symmetrische Lösung dargestellt: Ein Netzwerk (Netzwerk) ist mit mehreren Peers (Peer) verbunden, die untereinander ebenfalls vernetzt sind.

Verteilte Systeme, Sommersemester 1999 Folie 11

### Probleme

---

Endliche Ausdehnungsgeschwindigkeit der Information

Das Diagramm zeigt zwei horizontale Linien, die durch eine diagonale Linie verbunden sind, die von links oben nach rechts unten verläuft. Die horizontale Achse ist mit 'Zeit' beschriftet. Unter der Achse sind zwei Zeitpunkte markiert:  $t_{\text{Send}}$  und  $t_{\text{Receive}}$ , wobei  $t_{\text{Send}} < t_{\text{Receive}}$  gilt.

- Zeitsynchronisation schwierig
- Veraltete Information

Eigenschaften des Kommunikationsnetzwerkes

- Schwankungen in der Latenz
- Verluste, Vertauschungen, Duplikate

Implizite und explizite Abhängigkeiten

Verteilte Systeme, Sommersemester 1999 Folie 12

### Problem: Konsistenz globaler Zustände

Beispiel: Terminierung in einem verteilten Aktorensystem

Aktoren kommunizieren nur durch Nachrichten

Zustände eines Aktoren

- Aktiv
- Passiv

Was heißt "Terminierung des verteilten Aktorensystems"?

Lösungsvorschlag?

Verteilte Systeme, Sommersemester 1999 Folie 13

### Problem: Fehlen einer globalen Zeit

Wie schon Einstein sagte: „Alles ist relativ“

Bedeutung der Zeit

- Vergleichbarkeit und Ordnung von Ereignissen
- Vielfach versteckte Abhängigkeiten

Gleichzeitigkeit

Konsequenzen

- Naiv: Neue Physik muß her
- Pragmatisch: Hinreichend präzise Uhrensynchronisation
- Theoretisch: Schwächere Zeitbegriffe (Kausalität)

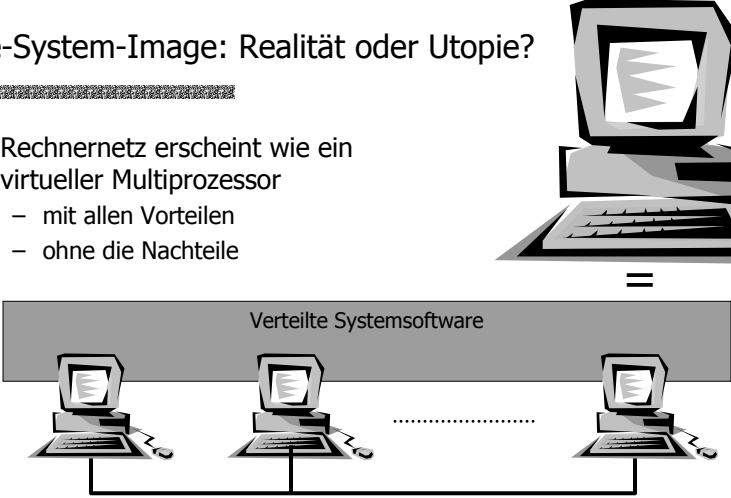
Warum würde es bei einer gemeinsamen globalen Zeit z.B. die Problematik konsistenter verteilter Zustände nicht geben?

Verteilte Systeme, Sommersemester 1999 Folie 14

### Single-System-Image: Realität oder Utopie?

Rechnernetz erscheint wie ein virtueller Multiprozessor

- mit allen Vorteilen
- ohne die Nachteile




Leistungen der verteilten Systemsoftware?

Verteilte Systeme, Sommersemester 1999 Folie 15

### Aktuelle Tendenzen

Nachrichtenlaufzeit war bisher dominant

- Signifikante Verzögerungen der beteiligten Prozesse
- Leistungssteigerungen schwierig
- Probleme verteilter Systeme unmittelbar spürbar



Ultraschnelle Netze

Der Hauptspeicher des Nachbarrechners ist näher als meine lokale Platte?

- Probleme verteilter Systeme bis zu einer gewissen Ausdehnung vernachlässigbar
- Cluster

Component-Ware

- DCOM, Corba, ...

Verteilte Systeme, Sommersemester 1999 Folie 16



## Zusammenfassung

### Gute Gründe dafür

- Leistungssteigerung
- Zuverlässigkeit
- Natürliches Modell

### „Dagegen“-Gründe sind lösbar

- Konzepte
- Pragmatische Ansätze
- Unterstützung
- Vereinfachende Modelle

### Große Zukunft



## Begleitende Literatur

*G. Coulouris, J. Dollimore, T. Kindberg*  
Distributed Systems - Concepts and Design  
2. Auflage, Addison-Wesley, 1994

*A. Silberschatz, P.B. Galvin*  
Operating System Concepts  
5. Auflage, Addison-Wesley, 1997

*A.S. Tanenbaum*  
Distributed Operating Systems  
Prentice Hall, 1995

Spezialliteratur im Verlauf der Vorlesung