Distributed Systems

1 Introduction

Organization, Overview, Introduction

April-20-2009

Summer Term 2009

System Architecture Group

Roadmap of Today

- Organization
- Literature
- Motivation
- Introduction
- Examples
- Problem Analysis

Organization

Online Information

Intended Schedule

Related Courses

Involved People

Voluntary Assignments

Intended Course Schedule

- Introduction
- Architectures
- Processes
- Communication
- Naming
- Synchronization
- Consistency & Replication
- Examples

  Distributed File Systems

  Clustered File Systems

Similar to content of our main textbook:
Andy Tanenbaum & Marten van Steen:
"Distributed Systems"

Online Information

- *Everything* will be on the Web
- **http://i30www.ira.uka.de**
  - Slides available in the previous week
  - Assignments
  - Additional Literature
  - Links
  - Forum

Copyright hint:

I'm using slides from the authors of the recommended textbooks and from some of my colleagues around the world. Feel free to use these slides for educational (and non commercial) purpose as well.

19 Lectures + 8 Tutorials

	Mo: 20. 4.	We: 22. 4.	Mo: 27. 4.	**We: 29.4.**
	Mo: 04. 5.	We: 06. 5.	Mo: 11. 5.	**We: 13.5.**
	Mo: 18.5.	We: 20. 5	Mo: 25.5.	**We: 27.5.**
Pentecost	Mo: 01. 6.	**We: 03. 6.**	Mo: 08. 6.	We: 10. 6.
	Mo: 15. 6.	**We: 17. 6.**	Mo: 22. 6.	We: 24. 6.
	Mo: 29. 6	**We: 01. 7.**	Mo: 06. 7.	We: 08. 7
	Mo: 13. 7.	**We: 15. 7.**	Mo: 20. 7.	**We: 22. 7**

Monday	11:30 – 13:00 in HS -102	(lectures)
Wednesday	09:45 – 11:15 in HS -102	(tutorials or lectures)

Involved People

Lecturer:	Gerd Liefländer
E-mail:	lief@ira.uka.de
Phone:	608-3837
Meeting Times:	Tuesday 15:45 – 17:15
Office:	160, 1st Floor, Informatik-Neubau (50.34)
Tutor:	Philipp Kupferschmied
E-mail:	pkupfer@ibds.uka.de
Phone:	608-3836
Meeting Times:	Tuesday 15:45 – 17:15
Office:	163, 1st Floor, Informatik-Neubau (50.34)

Assignments

- Non-programming assignments:
  - Questions similar to those used for topics of system architecture, e.g.

    “discuss the pros and cons of ...”
- Assignment 1 published this week
- First tutorial: Wednesday 29. 4. 2009

Why Assignments?

- Assignments will
  - help you to understand ("learning by doing")
  - train you to **reason appropriately**
  - train you to apply in practice what you've learned
- Discuss the topics with your colleagues & Philipp
- Train yourself to be able to explain
  - principal ideas and general paradigms of DS
  - special concepts, and their pros & cons
  - typical applications of a specific DS concept

Examination[1]

[1] Oral examinations either in German and/or in English

Veranstaltungstyp

DOS = Veranstaltung (3 + 1) aus dem Vertiefungsfach (Systemarchitektur)Betriebssysteme

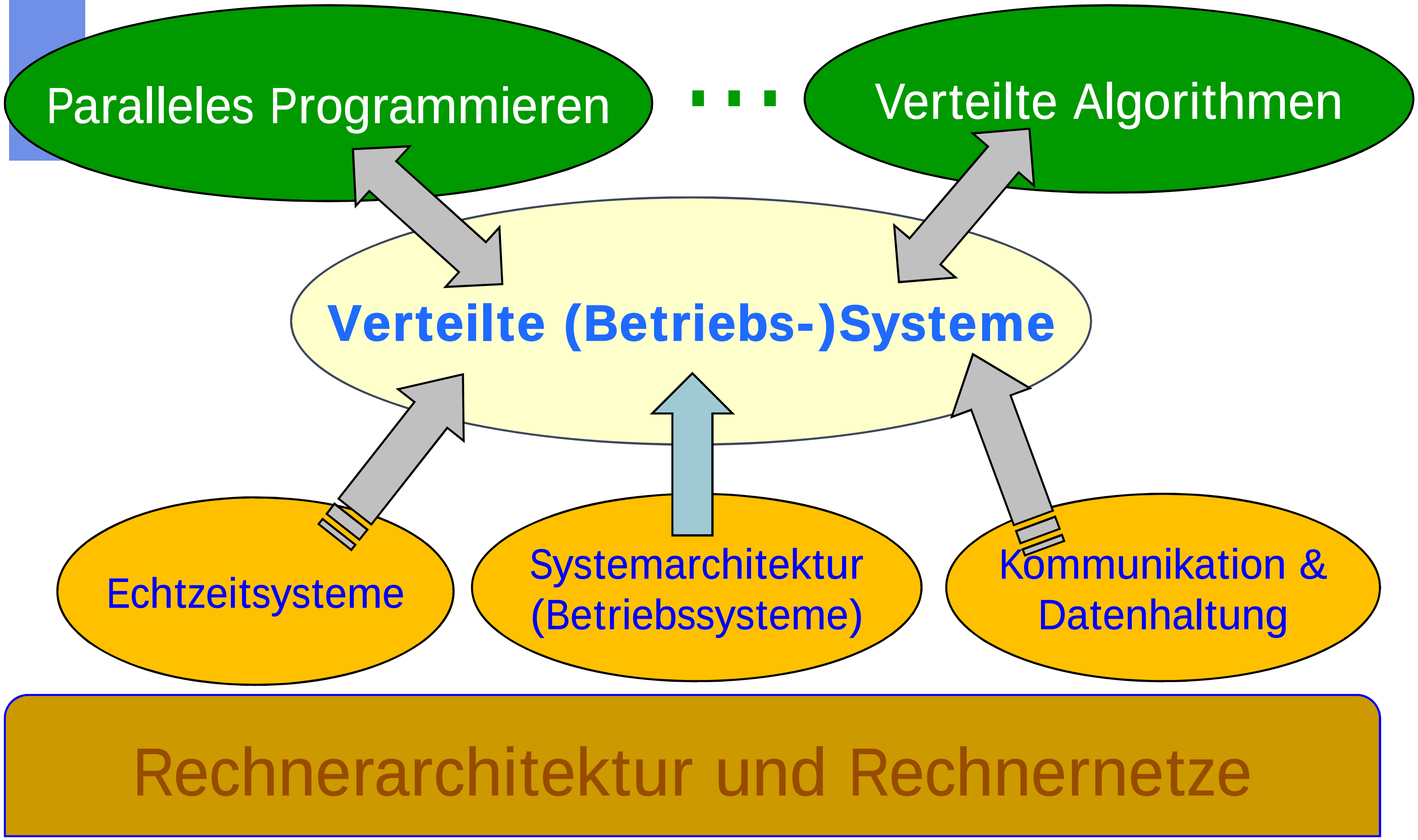
ECTS Points: 4,5 (Tutorial inclusive)

Voraussetzung:

- Grundkenntnisse aus Systemarchitektur

Nur mündlich abprüfbar

Einordnung des Stoffgebiets



Paralleles Programmieren

...

Verteilte Algorithmen

Verteilte (Betriebs-)Systeme

Echtzeitsysteme

Systemarchitektur (Betriebssysteme)

Kommunikation & Datenhaltung

Rechnerarchitektur und Rechnernetze

Related KIT Courses: ST 2009

- T. Dreier: “Internetrecht”
- S. Abeck et al.: “Advanced Web Applications”
- E. Buchman: “Datenschutz und Privatheit in vernetzten Informationssystemen”
- U.D. Hanebeck:”Lokalisierung mobiler Agenten”
- O. Waldhorst et al: “Mobilkommunikation”
- M. Schöller: “Netzsicherheit: Architekturen und Protokolle”
- B. Katz:”Algorithmen für Ad-Hoc- und Sensornetze”
- R. Bless et al: “Next Generation Internet”
- W. Juling: “Parallelrechner & -Programmierung”
- H. Hartenstein et al: “Simulation von Rechnernetzen”

Literature

Recommended Reading

Additional Literature

Similar Online Courses

Major Conferences

Recommended Reading

Andrew S. Tanenbaum, Maarten van Steen: "**Distributed Systems: Principles and Paradigms**", Prentice-Hall, 2nd Edition, 2006

Non plus ultra

Jean Bacon, Tim Harris: "Operating Systems: Concurrent & Distributed Software Design", Addison-Wesley, 2003

George Coulouris, Jean Dollimore, Tim Kindberg: "Distributed Systems: Concepts and Design", 4. Edition Addison-Wesley, 2005

Paul Verissimo, Luis Rodrigues: "Distributed Systems for System Architects", Kluwer Academic Publ., 2nd Ed. 2004, www-online !!!

Additional Literature (1)

Bengel, G.: “Verteilte Systeme”, Vieweg, 2004

Chow, R., Johnson, T.: “Distributed OSs & Algorithms”,
Addison Wesley, 1997

D. Galli: “Distributed Operating Systems”, Prentice Hall, 2000

Kopetz, H.: “Real-Time Systems: Design Principles for Distributed
Embedded Applications”, Kluwer, 1997

Lynch, N.: “Distributed Algorithms”, Morgan Kaufmann, 1996

Mullender, S.: “Distributed Systems”, 3. Ed. , Addison-Wesley, 1995

Sinha, P.:“Distributed OSs”, IEEE Comp. Soc. Press, 1997

Additional Literature (2)

Singhal, M; “Advanced Concepts in Operating Systems”,
Shivaratri, N.: Mc-Graw Hill, 1994

Zhao, F., “Wireless Sensor Networks”,
Guibas, L. : Morgan Kaufmann, 2004

<u>Network Literature:</u>

Stallings, W.: “Local and Metropolitan Area Networks”,
Prentice Hall 2000

Peterson, L., Davie, B.: “Computer Networks”,
Morgan Kaufmann, 2003

Tanenbaum, A.: “Computer Networks”, Prentice Hall, 2003

Similar Courses

- I. Kuz:"Distributed Systems", UNSW, Australia, http://www.cse.unsw.edu.au/~cs9243/lectures/
- S. Fischer: "Verteilte Systeme" TU Braunschweig, Folien + Videos http://www.ibr.cs.tu-bs.de/courses/ws0203/vs/
- G. Mühl: "Verteilte Systeme", Uni Mannheim bzw. SS 2006 TU Berlin http://pi3.informatik.uni-mannheim.de/~schiele/distsys/
- K. Irmscher: Skriptum "Verteilte Syteme", Uni Leipzig www.informatik.uni-leipzig.de/~irmscher/lehre/skripte/
- W. Schröder-Preikschat: "Verteilte Systeme", Uni-Erlangen http://www4.informatik.uni-erlangen.de/Lehre/SS05/V_VS/
- Gunnar Teege: "Betriebssysteme II (VS)", Bundeswehrhochschule München, Skriptum 2004
- Oliver Theel: "Verteilte BS", WS 2004 Uni Oldenburg http://www.svs.informatik.uni-oldenburg.de/teaching/courses/vbs/2004
- ... and many others

Major Conferences

HotOS

SOSP (Symposium on OS Principles)

(IC)DCS

PODCS (Principles of Distributed Computing Systems)

PDCS (Parallel and Distributed Systems)

SPDP Symposium on Parallel and Distr. Processing)

ICCDS (Int. Conf. On Configurable Distributed Systems)

NSDI (Network System Design and Implementation)

...

Motivation

DS – what and why?

Characteristics of DS

Some DS Researchers

Course Coals

Preliminary Definition

- A distributed system is a collection of independent computers that are used jointly to perform a single application task/process or to provide a singly service via aserver task/process

First Examples of DS

- Collection of Web servers: distributed data base of hypertext and multimedia documents
- Distributed file system on a LAN
- Domain Name Service (DNS)
- Cray XT5 & CLE (massive multiprocessor)

Find some more widely known examples of distributed systems

Why Distributed Systems?

What's the alternative? Mega Mainframe"?

Some apparent advantages of DS:

Using commodity hardware

- Cost: Better price/performance
- Performance: Beyond scope of a mainframe

Add components when needed

- Scalability: Even incrementally scalable
- Reliability: HW- and SW-failures don't have such a disastrous impact
- Inherent distribution: Some applications (like the Web) are naturally distributed

Apparent Disadvantages of DS

- New component “network“: networks are needed to connect the independent computer nodes
- (Unreliable) network might limit the performance of a DS significantly
- Security: it is easier to compromise a DS
- Software complexity: Software for DS is more complex and harder to develop than conventional software, i.e. it is harder to develop and get it right
- Corollary: DS are hard to build and understand

Why Lecture about DS?

- ∃ many per se distributed applications
  i.e. airline-reservation systems
  ⇒ offer appropriate synchronization facilities, e.g. don’t book a window seat twice
- ∃ no common shared physical memory anymore
  ⇒ need explicit IPC by communicating messages or an “emulation via distributed shared memory”
- Systems can grow significantly
  ⇒ face new problems of system architecture: “quantity can become quality”
- Failures in DS are quite probable
  ⇒ tolerate failures and errors
- System components are heterogeneous
  ⇒ integrate heterogeneous hardware + software
  ⇒ need standards

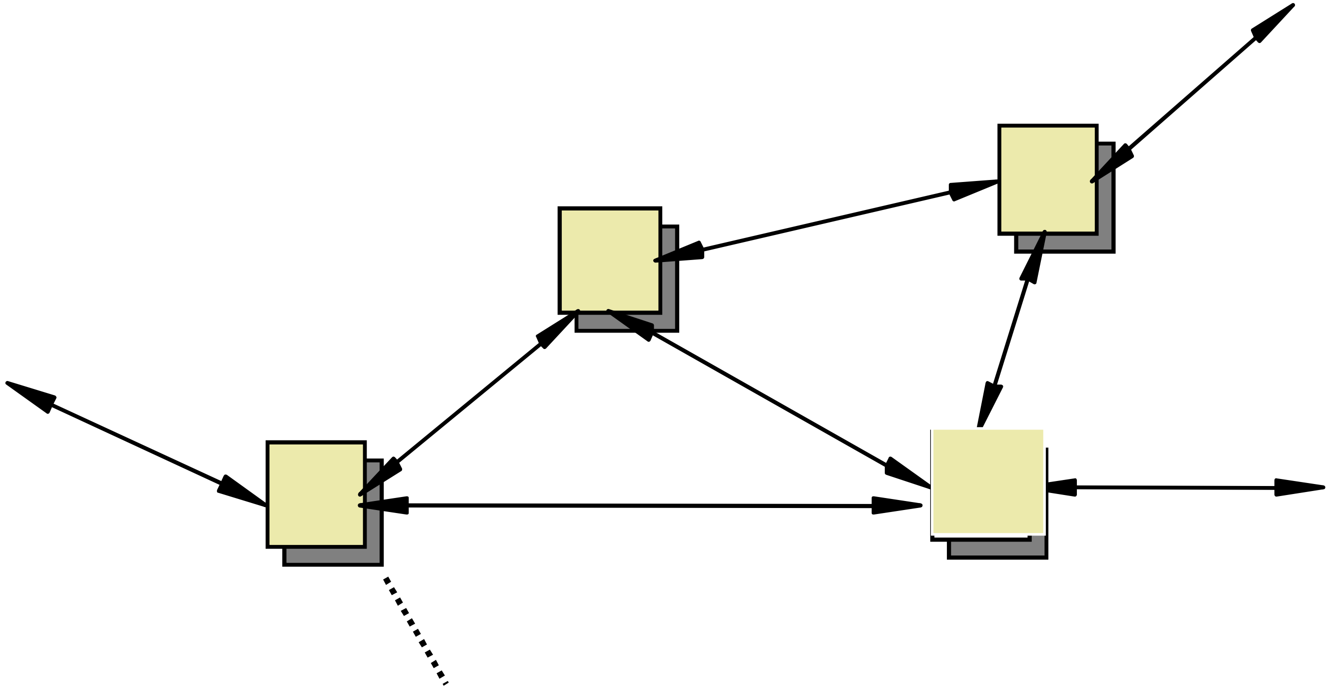
Introduction

Abstract DS

Course Goals

History of DS

Review: View on Local Systems

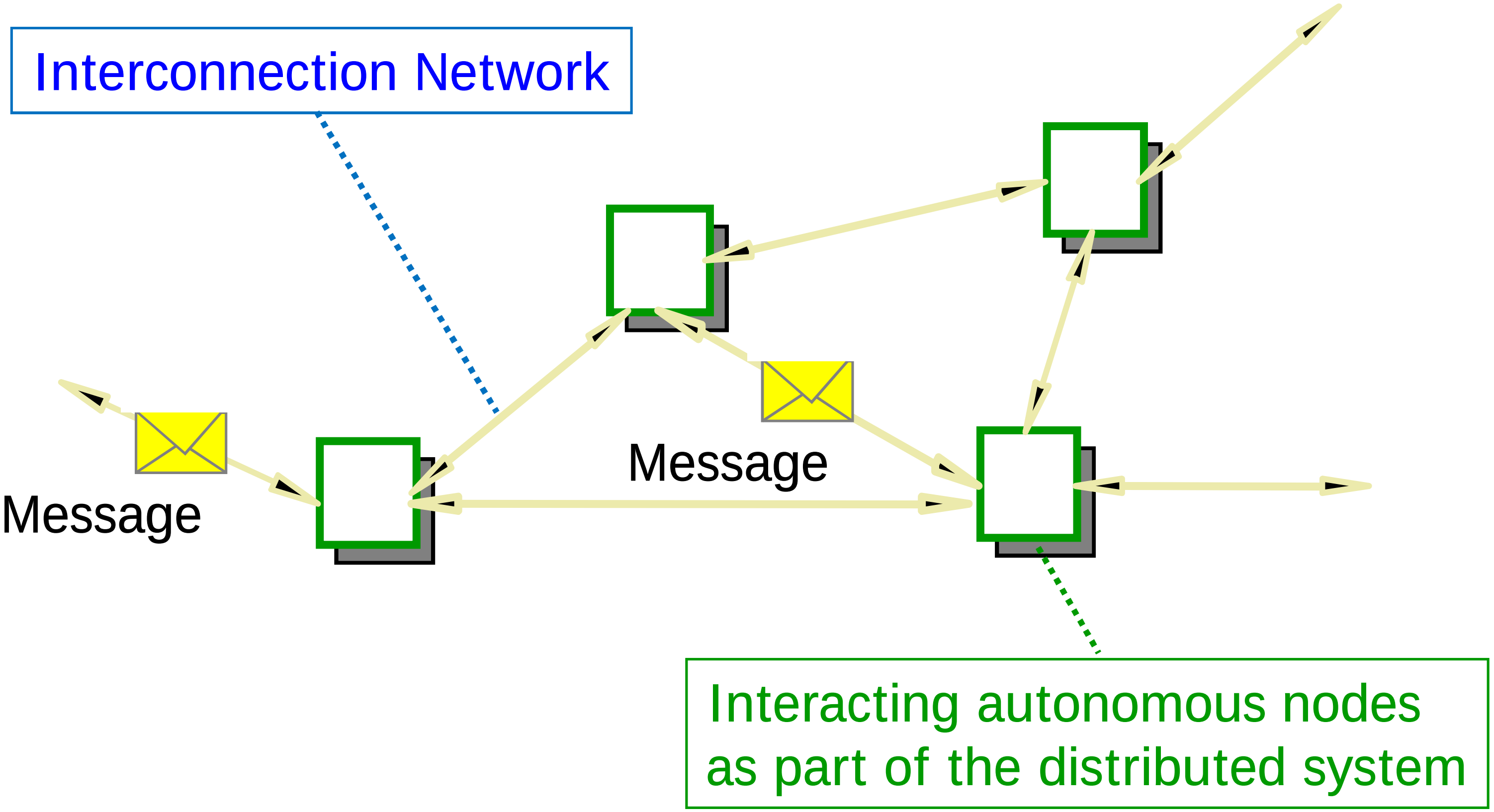


System Components

1. Interacting system components as part of a local system, relying on the concept of common main memory

2. System components in separate address spaces might have to interact via IPC

Abstract View: Distributed Systems


Interconnection Network
Message
Message
Interacting autonomous nodes as part of the distributed system

Data transfer rate varying between

- some Kbps … Gbps

First Summary

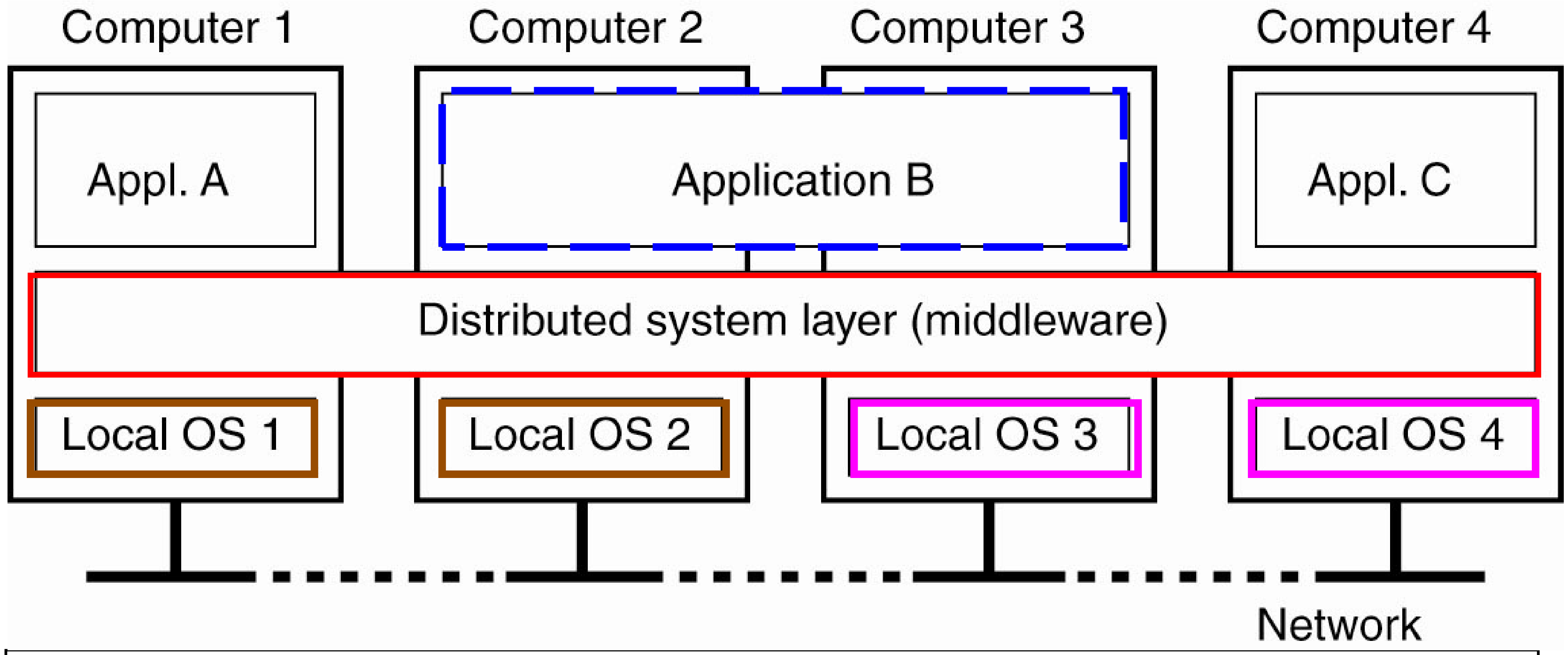
A DS is a

- collection of autonomous computer nodes
- connected by a network

HW

- supported by system & application software that makes it appear as a single system image (SSI)

A Distributed System



- DS can be designed as a middleware layer
- Distributed application
- Heterogeneous OS kernels

Our Course Goals

- Major system design problems of a DS
  - *What problems have to be solved to allow* ***fast & reliable interactions of applications*** *on different nodes?*
    - Know your RPC/RMI or IPC in full detail
  - *When, why and how to use replication?*
  - *What additional security problems?*
- Major DS design principles and concepts
- No middleware or mobile systems in details, e.g.
  - No agent, no ad hoc systems etc.
  - see other KIT courses

AGAIN: Specific Goals of Course

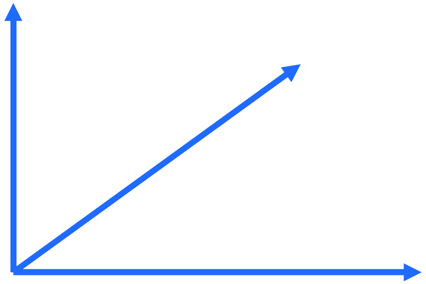
- Decompose a DS into its major system components
- Applying principles and concepts
- Studying orthogonal design parameters of
  - DS as a whole
  - Individual system components

Examples:

Various communication facilities

Handling of failures

Transparency degrees



Again, never forget!

History (1)

http://www.spinroot.com/gerard/hist.html

Nice history of “long distance communication”:

“The Early History of Data Networks”

by Gerard J. Holzmann, Björn Pehrson

Content: From torches, pigeons, flags, beacons, telegraphs, finally to the *Internet*

History (2)

In the 80ies two major advances in IT technology enabled transition from local to distributed systems:

- Cheap µ-processors leading to PCs
- High-speed LANs (e.g. Ethernet)

Since then, system architects had another challenge:

Establish a DS with a performance similar to a fast mainframe (+ some additional features).

History (3)

- Computer-Computer Communication
  - Remote access (DFÜ)
  - No decentralized computing
  - ⇒ Mostly master/slave-relation (RJE, Terminals)

- ARPA Network (Prototype of WAN)
  - Peer to peer communication
  - Internet protocol family (TCP/IP)
  - Standards (ISO/OSI)

History (4)

- Net of Workstations (LAN)
    - Xerox-Parc WS
    - Ethernet, RPC, distributed file systems
    - Today standard for PC-applications.
        - Communication via LAN (resource-sharing)
        - Software for team working (email, ftp, ...)
- Projects
    - Reservation systems, banking, credit cards
    - Joint authoring, teleconferencing, distant learning

Impetus

“First driving force behind the trend towards DSs has been economics.”

A. Tanenbaum

2 different starting-points for DSs:

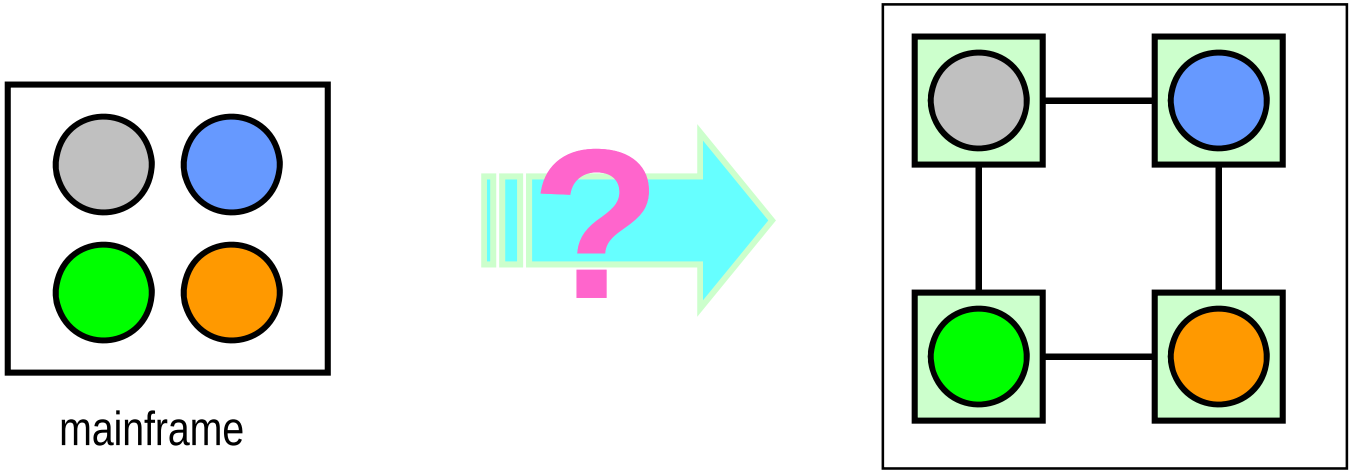
- distribute
- connect

Distribution Problem

Given: an expensive mainframe

1. How to distribute applications onto cheaper PCs or WSs?

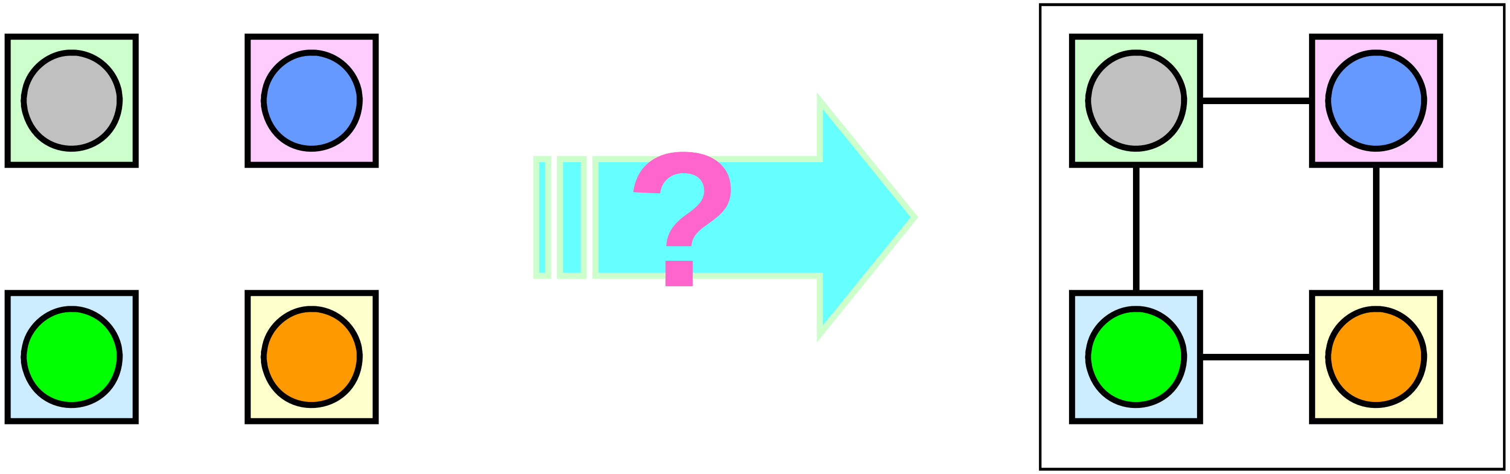
2. How to distribute services of an OS amongst the nodes?



Connection Problem

Given: n specialized PCs spread all over the world

1. *How to connect systems to get good remote service?*
2. *How to support this heterogeneity and how to handle platform-dependent formats?*



<u>Hope:</u> To end the „*tyranny of geography*".

Examples & Applications

Intranet

Embedded Systems

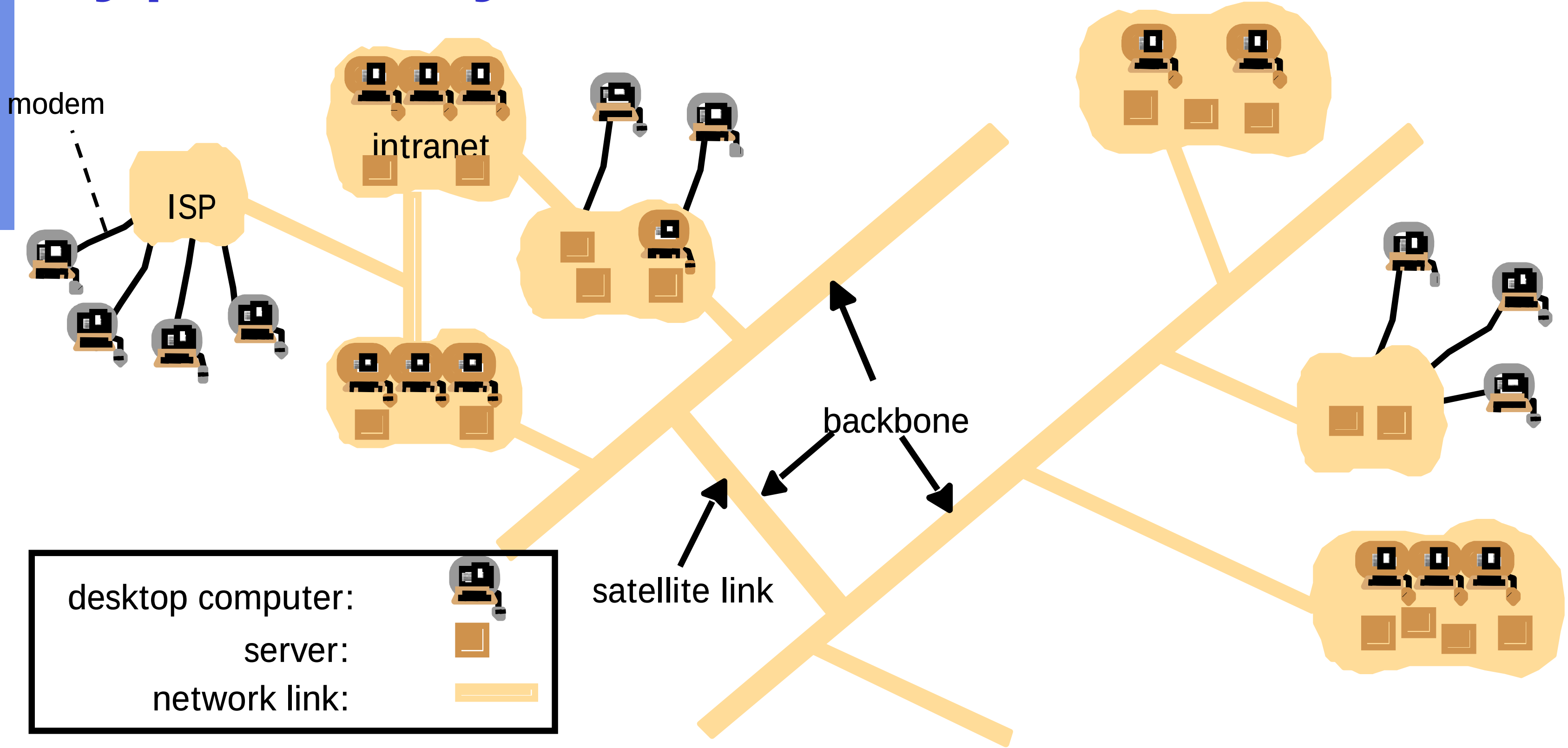
Mobile Systems

Current State of the Art[1]

- Typical examples
  - Internet
  - Intranet of an enterprise with a distributed FS
    - Compute farm (CPU farm)
    - Automatic teller machines
    - Production & Assembly lines
    - Cars
    - Military defense/offense systems
- *Can we identify specific DS related problem areas ?*
- *What are the main challenges of current DS?*

[1] Some of the following slides are from W. Schröder-Preikschat's lecture DS, ST 2005, others are from Coulouris' textbook

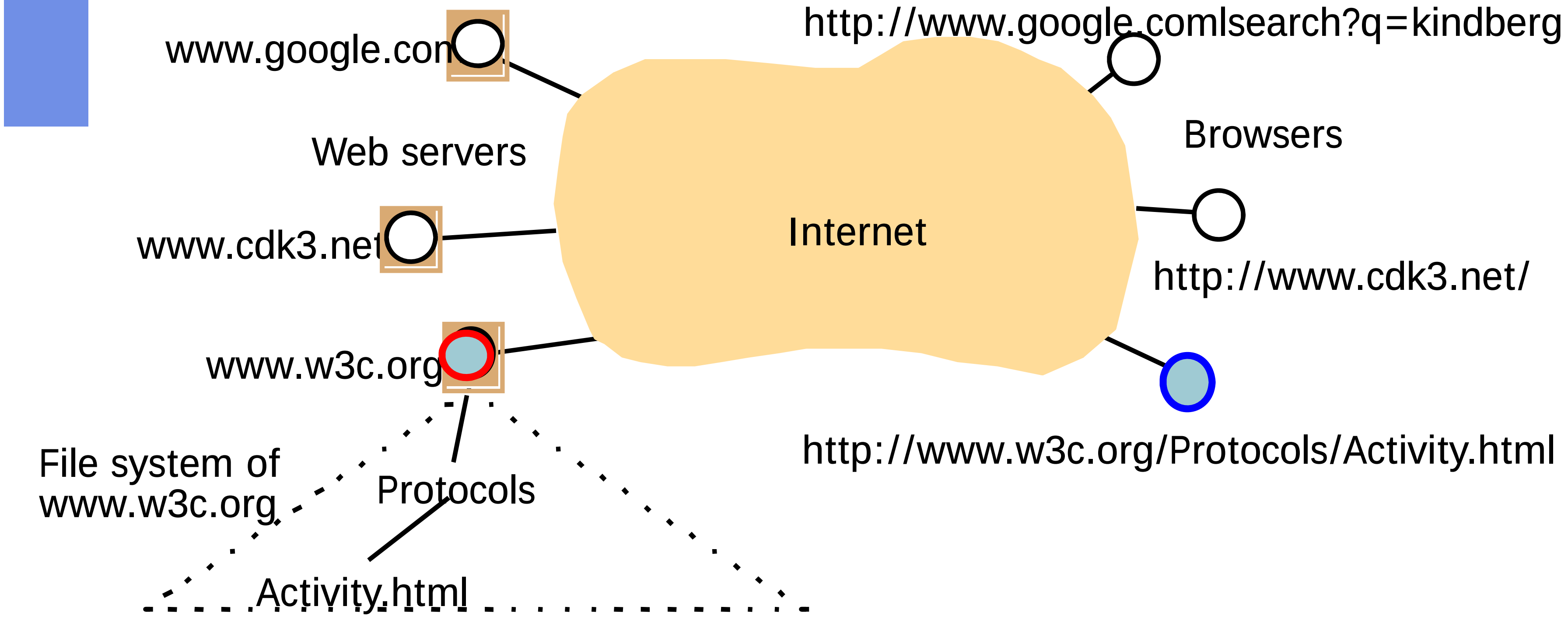
Typical Layout of the Internet



Quite heterogeneous nodes and *interconnections* and *services*, e.g.

- www
- email
- ftp

Web Server and Browser[1]



[1] In Coulouris chapter 1.3 you can read in full detail how a web browser interacts with a web server to publish a web site at the clients desktop

Computers in the Internet

Date	*Computers*	*Web servers*
1979, Dec.	188	0
1989, July	130,000	0
1999, July	56,218,000	5,560,866
? →		

Computers vs. Web Servers

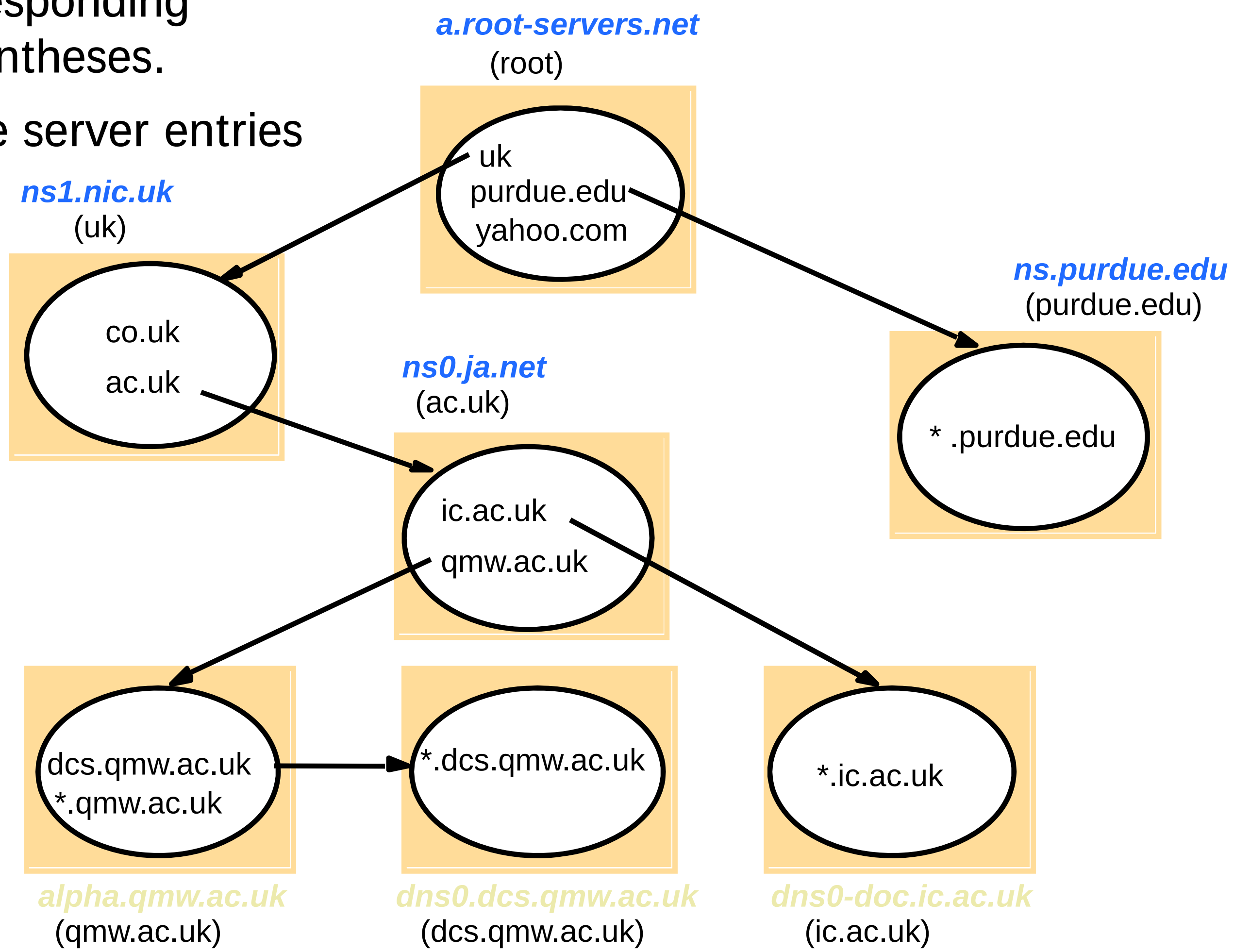
Date	*Computers*	*Web servers*	*Percentage*
1993, July	1,776,000	130	0.008
1995, July	6,642,000	23,500	0.4
1997, July	19,540,000	1,203,096	6
1999, July	56,218,000	6,598,697	12

Both slides indicate a need for scalable DS

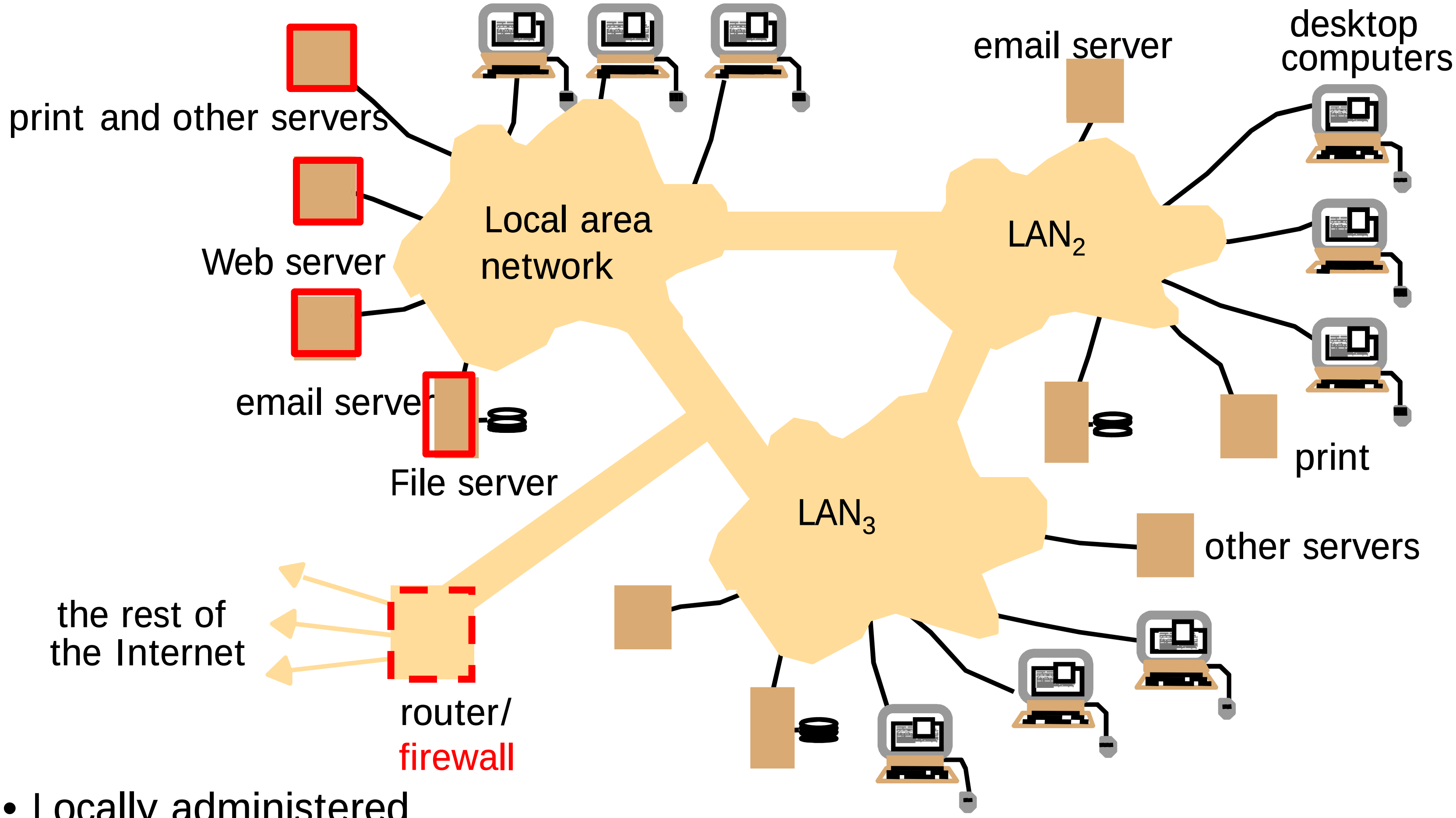
Domain Name Service

Note: *Name server names are in italics*, and the corresponding domains are in parentheses.

Arrows denote name server entries

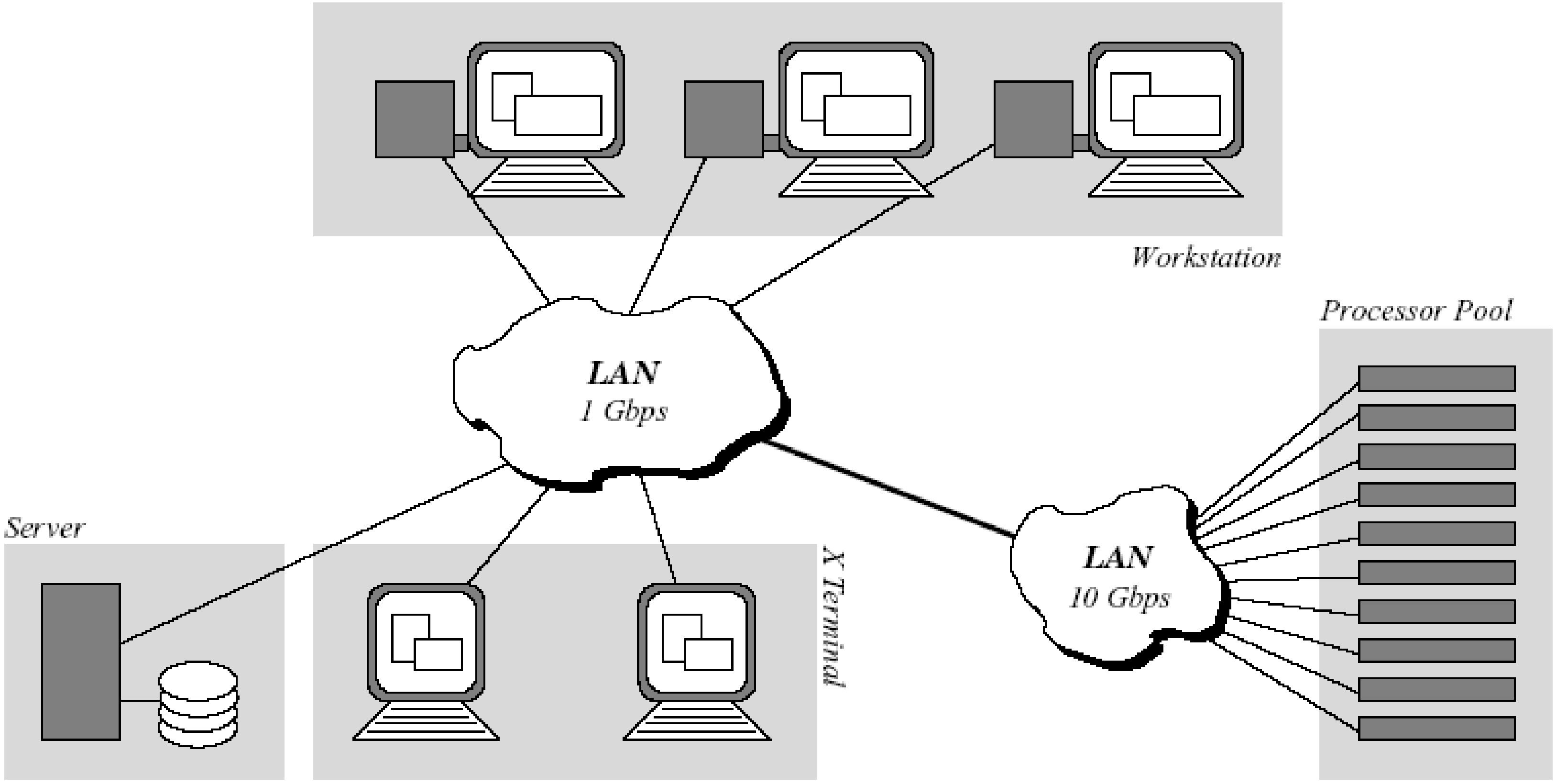


Typical Layout of an Intranet

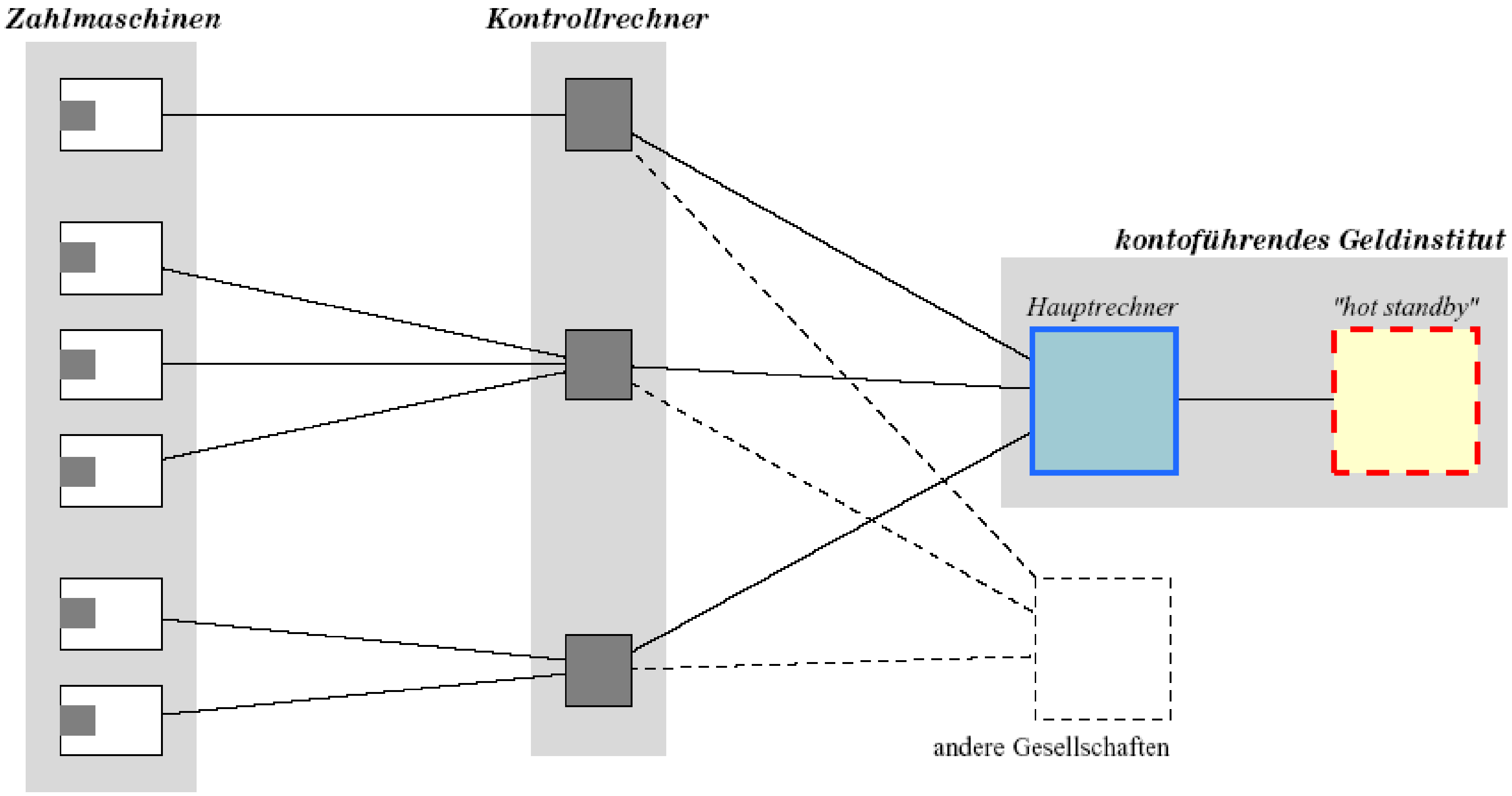


- Locally administered
  - Usually proprietary
  - Interfaces to the Internet via firewalls
- Provides services internally and externally

Compute Farms

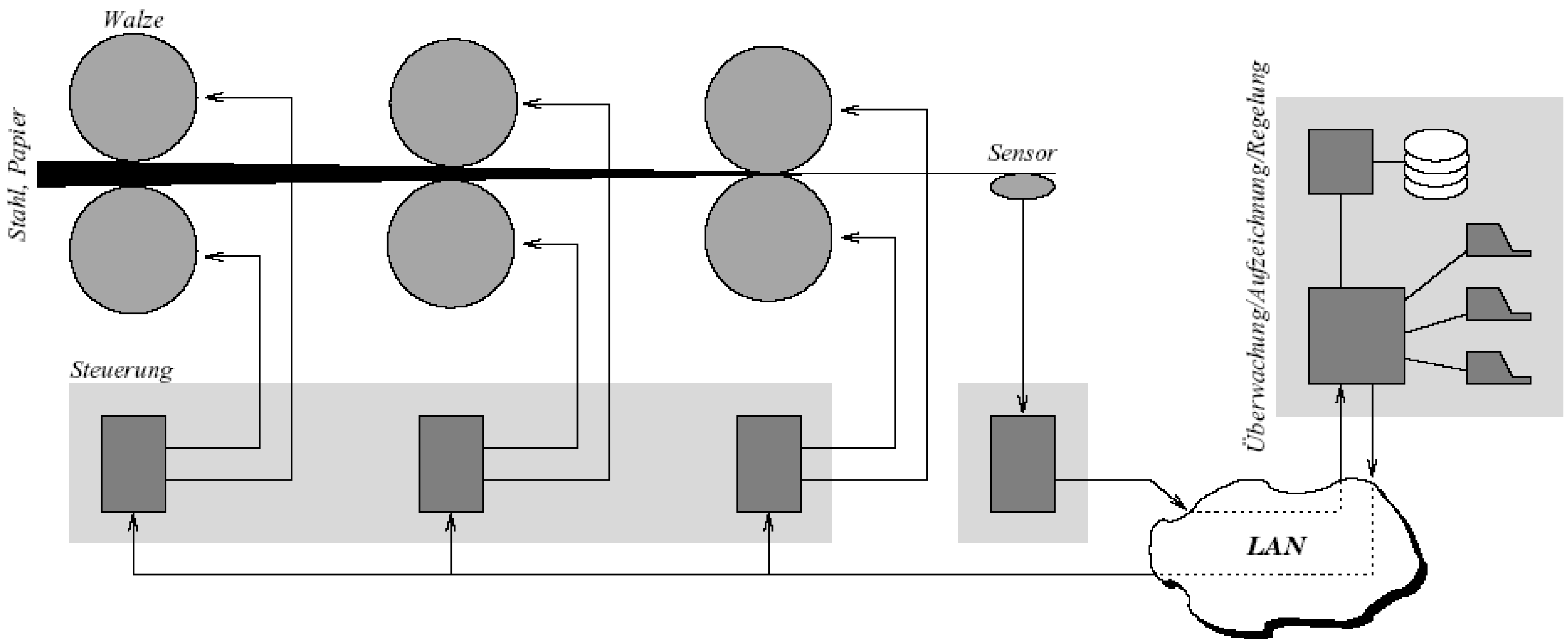


Automatic Teller Machines

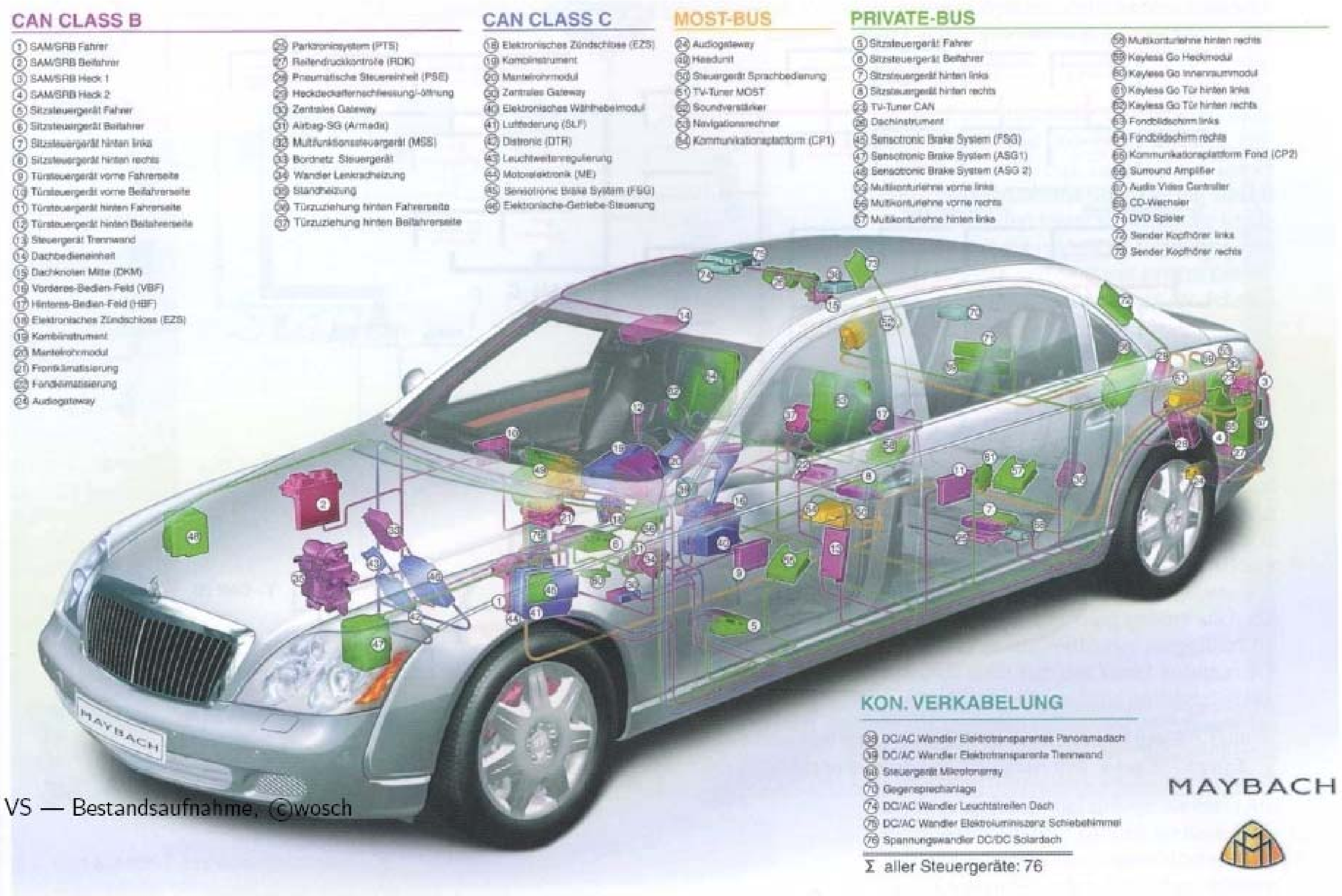


- We need distributed transactions

Production Line (Rolling Mill)



Cars



CAN CLASS B

1 SAM/SRB Fahrer
2 SAM/SRB Beifahrer
3 SAM/SRB Heck 1
4 SAM/SRB Heck 2
5 Sitzsteuergerät Fahrer
6 Sitzsteuergerät Beifahrer
7 Sitzsteuergerät hinten links
8 Sitzsteuergerät hinten rechts
9 Türsteuergerät vorne Fahrerseite
10 Türsteuergerät vorne Beifahrerseite
11 Türsteuergerät hinten Fahrerseite
12 Türsteuergerät hinten Beifahrerseite
13 Steuergerät Trennwand
14 Dachbedieneinheit
15 Dachknoten Mitte (DKM)
16 Vorderes-Bedien-Feld (VBF)
17 Hinteres-Bedien-Feld (HBF)
18 Elektronisches Zündschloss (EZS)
19 Kombiinstrument
20 Mantelrohrmodul
21 Frontklimatisierung
22 Fondklimatisierung
24 Audiogateway
26 Parktronicsystem (PTS)
27 Reifendruckkontrolle (RDK)
28 Pneumatische Steuereinheit (PSE)
29 Heckdeckelfernschliessung/-öffnung
30 Zentrales Gateway
31 Airbag-SG (Armada)
32 Multifunktionssteuergerät (MSS)
33 Bordnetz Steuergerät
34 Wandler Lenkradheizung
35 Standheizung
36 Türzuziehung hinten Fahrerseite
37 Türzuziehung hinten Beifahrerseite

CAN CLASS C

18 Elektronisches Zündschloss (EZS)
19 Kombiinstrument
20 Mantelrohrmodul
30 Zentrales Gateway
40 Elektronisches Wählhebelmodul
41 Luftfederung (SLF)
42 Distronic (DTR)
43 Leuchtweitenregulierung
44 Motorelektronik (ME)
45 Sensotronic Brake System (FSG)
46 Elektronische-Getriebe-Steuerung

MOST-BUS

24 Audiogateway
49 Headunit
50 Steuergerät Sprachbedienung
51 TV-Tuner MOST
52 Soundverstärker
53 Navigationsrechner
54 Kommunikationsplattform (CP1)

PRIVATE-BUS

5 Sitzsteuergerät Fahrer
6 Sitzsteuergerät Beifahrer
7 Sitzsteuergerät hinten links
8 Sitzsteuergerät hinten rechts
23 TV-Tuner CAN
25 Dachinstrument
45 Sensotronic Brake System (FSG)
47 Sensotronic Brake System (ASG 1)
48 Sensotronic Brake System (ASG 2)
55 Multikonturlehne vorne links
56 Multikonturlehne vorne rechts
57 Multikonturlehne hinten links
58 Multikonturlehne hinten rechts
59 Keyless Go Heckmodul
60 Keyless Go Innenraummodul
61 Keyless Go Tür hinten links
62 Keyless Go Tür hinten rechts
63 Fondbildschirm links
64 Fondbildschirm rechts
65 Kommunikationsplattform Fond (CP2)
66 Surround Amplifier
67 Audio Video Controller
68 CD-Wechsler
71 DVD Spieler
72 Sender Kopfhörer links
73 Sender Kopfhörer rechts

KON. VERKABELUNG

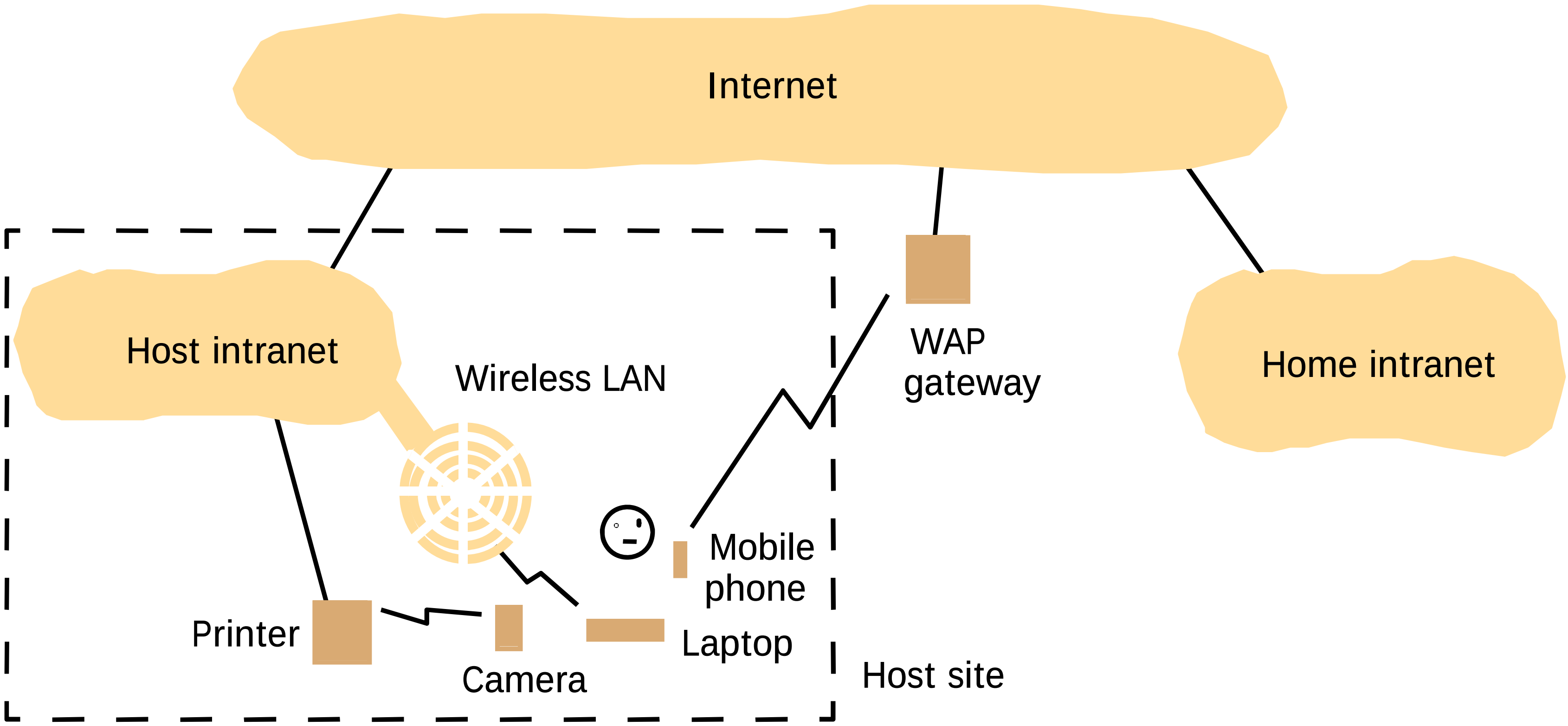
38 DC/AC Wandler Elektrotransparentes Panoramadach
39 DC/AC Wandler Elektrotransparente Trennwand
69 Steuergerät Mikrofonarray
70 Gegensprechanlage
74 DC/AC Wandler Leuchtstreifen Dach
75 DC/AC Wandler Elektrolumineszenz Schiebehimmel
76 Spannungswandler DC/DC Solardach

Σ aller Steuergeräte: 76

VS — Bestandsaufnahme, ©wosch

DaimlerChrysler AG. Der neue Maybach. ATZ/MTZ Sonderheft, page 125, Sept. 2002.

Portable & Handheld Devices in a DS



Scenarios & Applications of DS?

- Multiple users at distinct locations
  - Improving efficiency
  - Load balancing
  - Availability
- Examples
  - Multi-user games
  - Virtual worlds
  - Chat, video conferences, e-commerce

Typical Applications of a DS?

- Collaborative work around the world
  - Email
  - Internet
  - Web
  - Mobile and ubiquitous computing
  - Worldwide design of a new product
    - Crossing borders
    - Time-zones

Reasons for DS

- Functional distribution: computers (PC, WS, ...) have different functional capabilities
  - Client/server
  - Host/terminal
  - Data gathering/data processing
  - Sharing of resources
- Inherent distribution by an application, e.g.
  - Cash register and inventory systems for a supermarket
  - Computer supported collaborative work

Problem Analysis

Rough Analysis of DS

What is different compared with a local system?

- Local → remote
  - remote interaction induces more failures
- Direct → indirect binding
  - Configuring needs linkage support at run-time
- Sequential (or concurrent) → parallel
  - Instead of pseudo-parallelism on single-processors we have to face real parallelism in a DS and need appropriate concurrency control, that can not rely on a shared main memory

Rough Analysis of DS (2)

- Synchronous → asynchronous
  - Delays in remote interaction often require asynchronous communication and pipelining
- Homogeneous → heterogeneous
  - Remote interaction requires standardized common data representation
- Single instance → replicated group
  - Replication is used to enhance availability & dependability, however, requires additional efforts to maintain consistency

Rough Analysis of DS (3)

- Fixed location → migration
  - Location of remote objects & interfaces can change during run time
- Uniform → compound name space
  - Name resolution has to reflect the crossing of domain boundaries
- Shared memory → distributed memory
  - Mechanisms based on shared memory concept are hard to apply in a DS

Current *Hot* Topics of DS

- Massive scale
  - Internet consists of hundred of millions of nodes
  - Potential number of users is incredibly large
  - CNN.com got 30 000 hits/s on Sept. 11, 2001

⇒ tackle scalability

Current *Hot* Topics of DS

- Self-organization and decentralization
  - No central authority (administrator) managing, or organizing, or deploying the system
    - Gnutella nodes discover each other through broadcasting advertisements
  - Any part of the network can be taken down, the rest will survive

⇒ tackle P2P and related approaches

Current *Hot* Topics of DS

- Robustness and fault tolerance
  - Novel systems deployed on well-maintained, well-configured HW in an “engineered” environment
  - Systems must tolerate unprecedented degrees of heterogeneity and rate of failure

⇒ tackle robustness & fault tolerance

Ideal System Architecture of DS?

- We don’t know yet
- We must deal with far too many design parameters due to the various distributed applications
- Hopefully, we can identify some promising approaches for specialized DS
- As with OS, we still need a better understanding of the interrelationships of HW & SW components

Preview

- Characteristics
- Goals & Challenges
- Types of DS
- Examples

Recommended reading
Chapter 1 of Tanenbaum or Chapter 2 of Coulouris

B. Clifford Neuman: “Scale in DS”, IEEE Comp.Soc. Press, 1994

Bradley N. Miller, Joseph A. Konstan, John Riedl: “PocketLens: Toward a Personal Recommender System”, ACM Transaction On Information Systems, July 2004