

Feldbusse und Sensornetze

Jörg Kaiser

Institut für Verteilte Systeme (IVS)
Eingebettete Systeme und Betriebssysteme (EOS)

Sommersemester 2005

Organisatorisches

Vorlesung

Prof. Dr. Jörg Kaiser
Institut für Verteilte Systeme (IVS)
Arbeitsgruppe Eingebettete Systeme und Betriebssysteme

kaiser@ivs.cs.uni-magdeburg.de

Übungen

Thomas Kiebel
Institut für Verteilte Systeme (IVS)
Arbeitsgruppe Eingebettete Systeme und Betriebssysteme

kiebel@ivs.cs.uni-magdeburg.de

Organisatorisches

Vorlesung: Mittwoch 11:00 Uhr c.t. Raum G22A-111
Übungen: Dienstag 15.00 Uhr c.t. Raum G29-334

Voraussetzungen: Vordiplom, VL Betriebssysteme 1,
VL Technische Informatik II.

Creditpoints: 6 ECTS

Erfolgreiche Teilnahme: Übungen, Prüfung

Einordnung: Informatik III: Spezialvorlesung

Organisatorisches

- **Übungen: Information auf dem web.**
- **Vorlesungsfolien stehen nach der Vorlesung auf dem web zur Verfügung**

<http://ivs.cs.uni-magdeburg.de/eos/lehre/SS2005/>

- **auch verfügbar über UNIVIS**

Teilnahme erfordert Registrierung auf der web-Seite :

https://bode.cs.uni-magdeburg.de/eos/anmeldung/form_in.php

Feldbusse und Sensornetze

- o **Einführung**
- o **Zuverlässigkeit und Fehlertoleranz**
- o **Uhrensynchronisation**
- o **Die physische Übertragungsschicht**
- o **Protokolle für zeitbeschränkte, zuverlässige Kommunikation**
- o **Sensornetze**

Literatur

Paulo Veríssimo, Luís Rodrigues:

Distributed Systems for System Architects

Kluwer Academic Publishers, Boston, January 2001

Hermann Kopetz:

Distributed Real-Time Systems

Kluwer Academic Publishers, 1997

Konrad Etschberger:

CAN - Controller Area Network, Grundlagen, Protokolle, Bausteine, Anwendungen

Carl Hanser Verlag, München, Wien, 1994

Sape Mullender (Hrsg.):

Distributed Systems

ACM Press, 1989

Weitere Literatur, insbesondere in Bezug auf existierende Standards und Protokolle wird in der Vorlesung bekanntgegeben.

On-line Dokumatation:

CAN: <http://www.can-cia.de>
Profibus: <http://profibus.com/downloads.html>
FIP: <http://worldfip.org/downloads>
LON: <http://echelon.com>

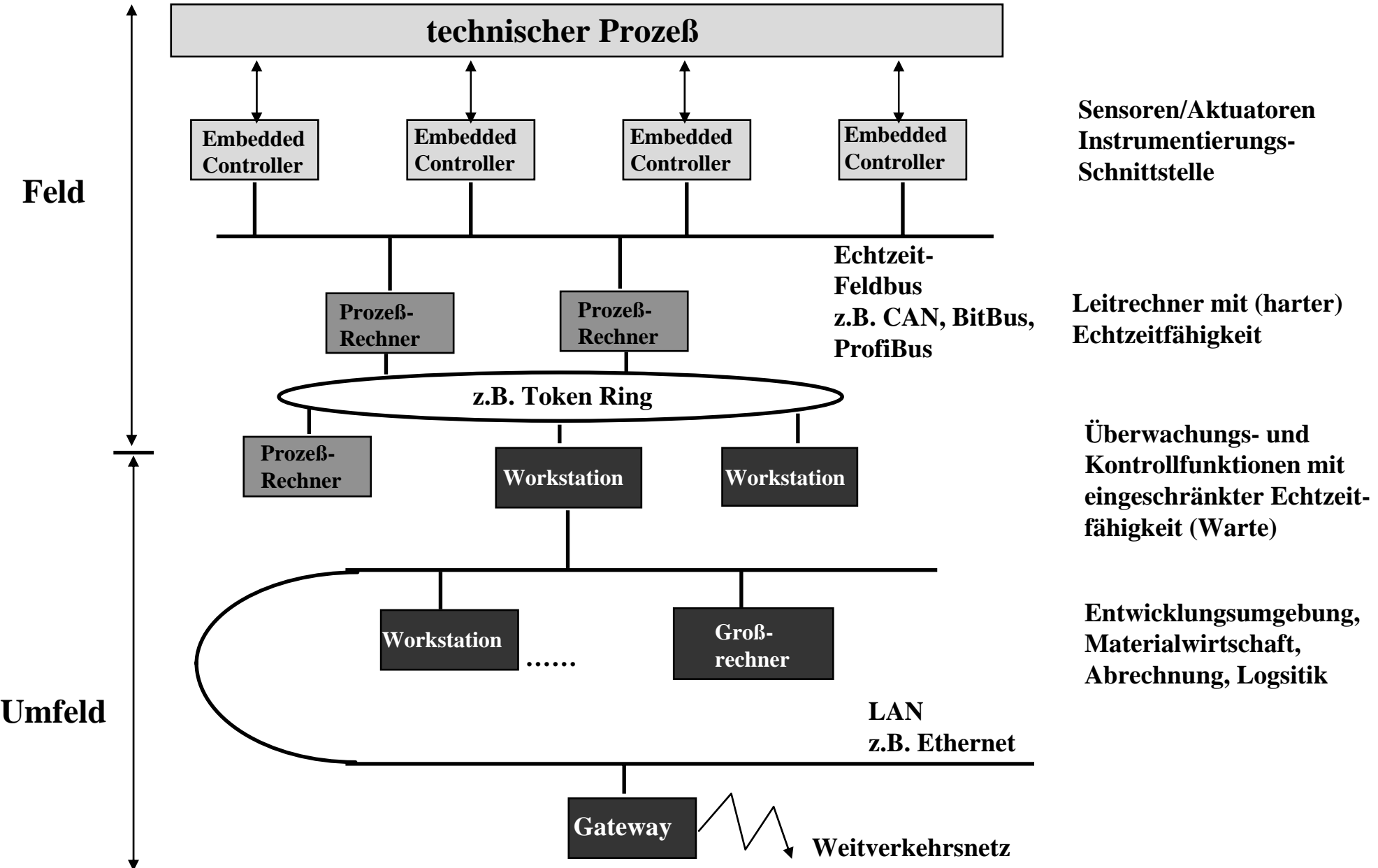
**Feldbusse und Sensornetze
oder**

**Kommunikationnetze zur
Erfassung und Kontrolle
der physischen Umgebung**

Einführung

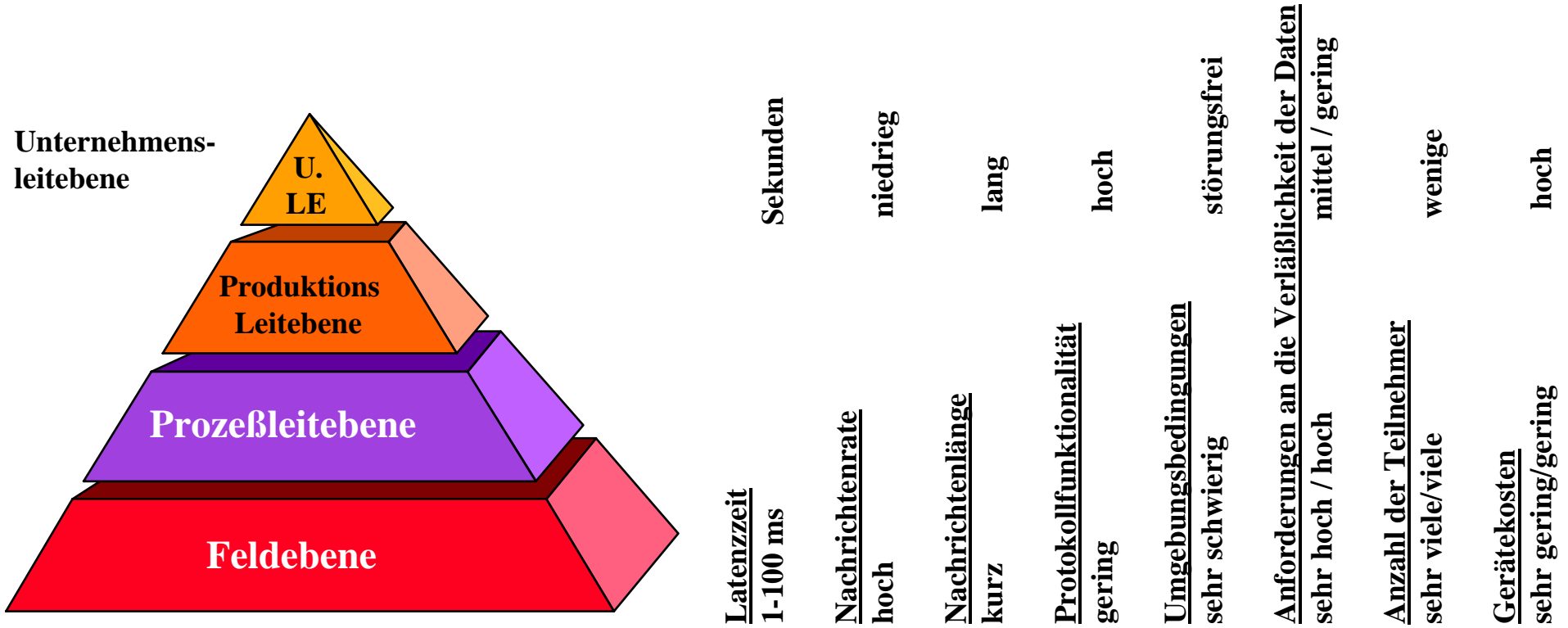
Feldbusse

Einbettung von Echtzeitsystemen in eine CIM-Umgebung



Anforderungen an das Kommunikationssystem bezogen auf die spezifische Ebene

(K. Etschberger(Hrsg.) CAN)

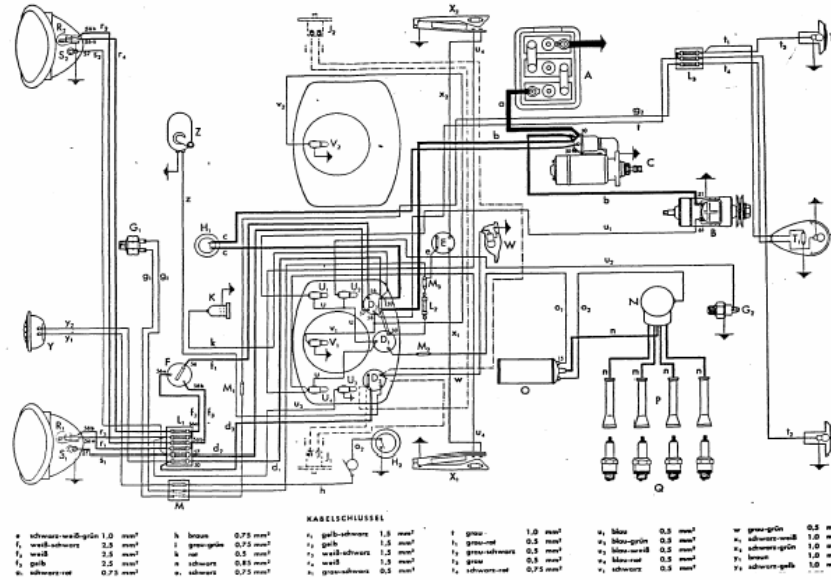


Steuerung und Kontrolle im Auto

Gestern



Elektrischer Schaltplan (Volkswagen)

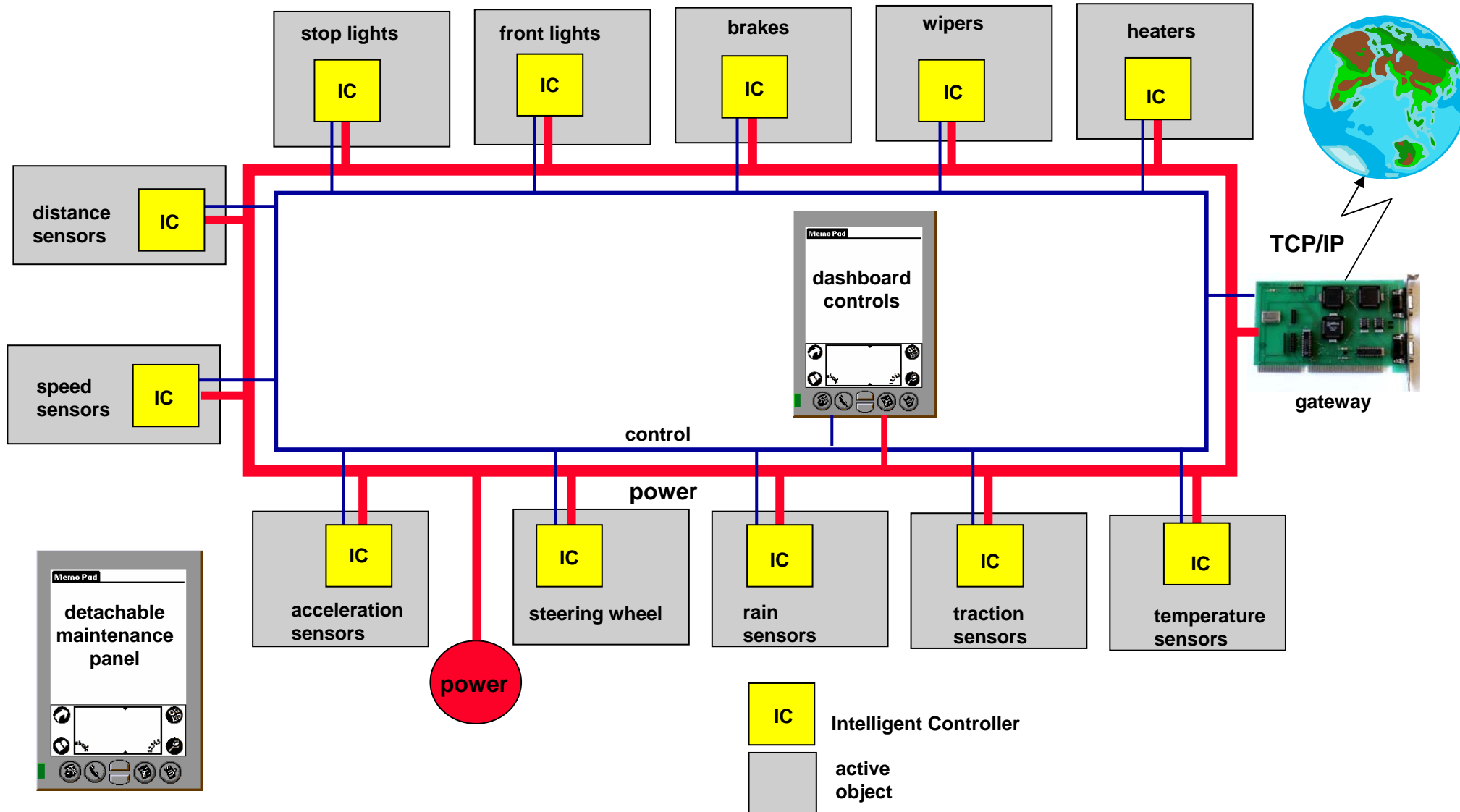


Heute



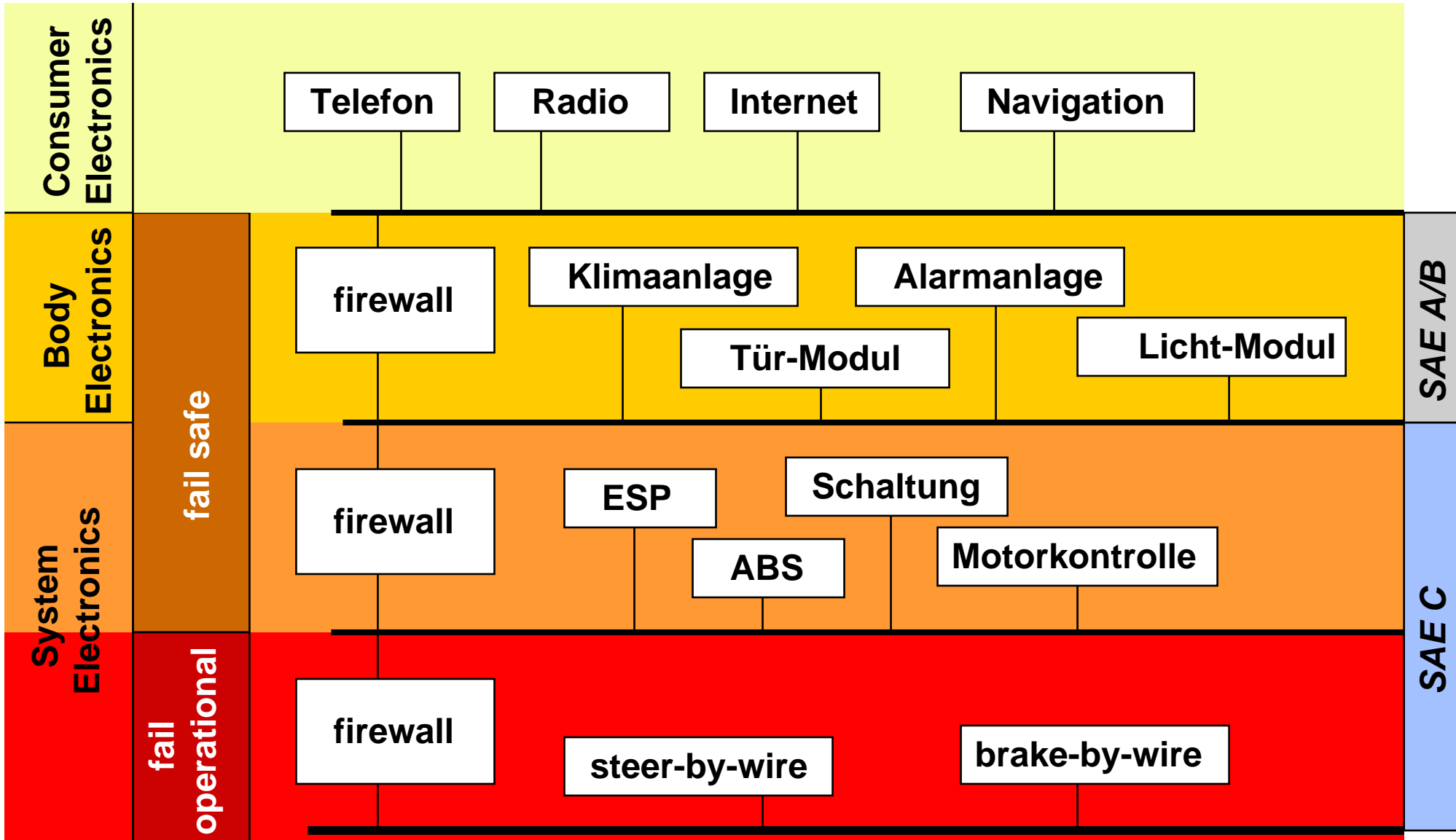
- 11.136 electrical parts
- 61 ECUs
- Optical bus for information and entertainment
- Sub networks based on proprietary serial bus
- 35 ECUs connected to 3 CAN-Busses
- 2500 signals in 250 CAN messages

In Zukunft: Verteilte kooperative Kontrolle



Ebenen der Kommunikation im Auto

Nach: T. Führer, B. Müller, W. Dieterle, F. Hartwich, R. Hugel, M. Walther:
 „Time Triggered Communication on CAN“





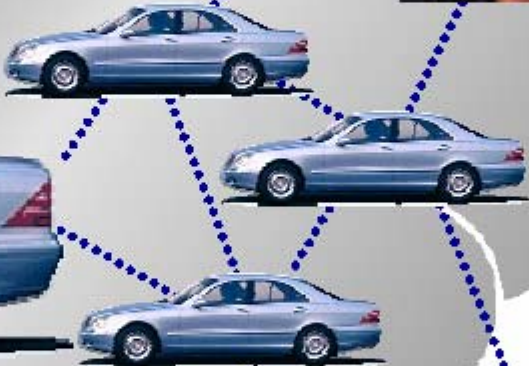
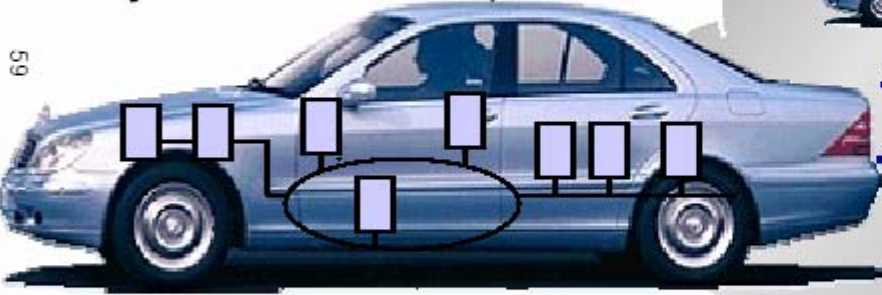
Vehicles become a means of communication with ...

... connectivity within vehicle internal domains (engine, body, telematics,...)

... connectivity to service centers

... connectivity to other vehicles

... connectivity to (public) infrastructure



69

Anwendungsbereiche für Feldbus-Kommunikationssysteme

- **Produktionsautomation**
- **Kraftfahrzeuge**
- **Gebäude**
- **Maschinen und Systeme**

Was sind die Anforderungen an ein Echtzeit-Kommunikationssystem ?

Ziel für ein Kommunikationssystem:

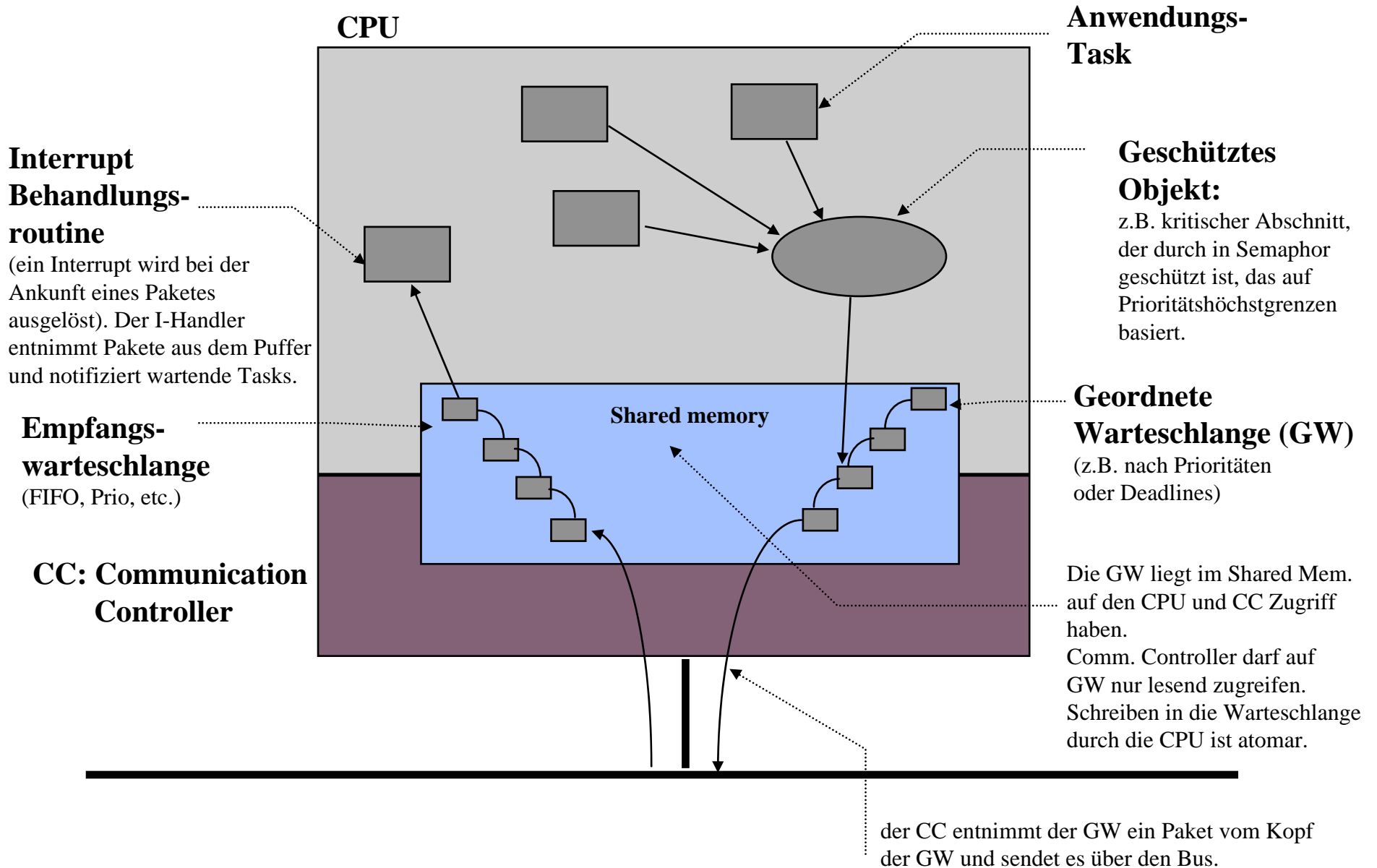
- NRT-Kommunikation: höchstmöglicher Durchsatz

d.h. die Datenmenge, die in einem Zeitintervall über das Netz übertragen werden kann soll unter gegebenen technischen Gegebenheiten und Kostenbedingungen maximal sein.

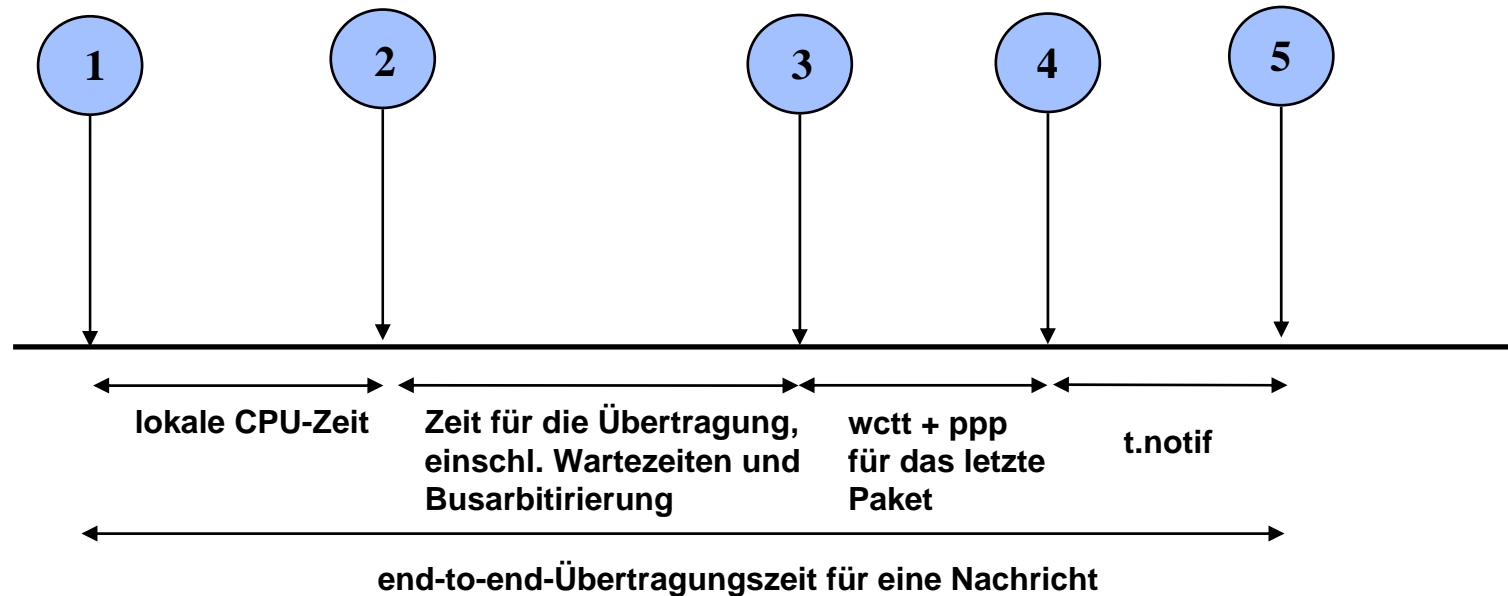
- RT-Kommunikation: Vorhersagbarkeit der Kommunikation

d.h. Schwergewicht liegt auf der Bestimmung der Wahrscheinlichkeit, dass eine Nachricht zu einem bestimmten Zeitpunkt, der Deadline im (in den) Empfänger(n) ausgeliefert werden kann.

End-to-End Kommunikationskosten

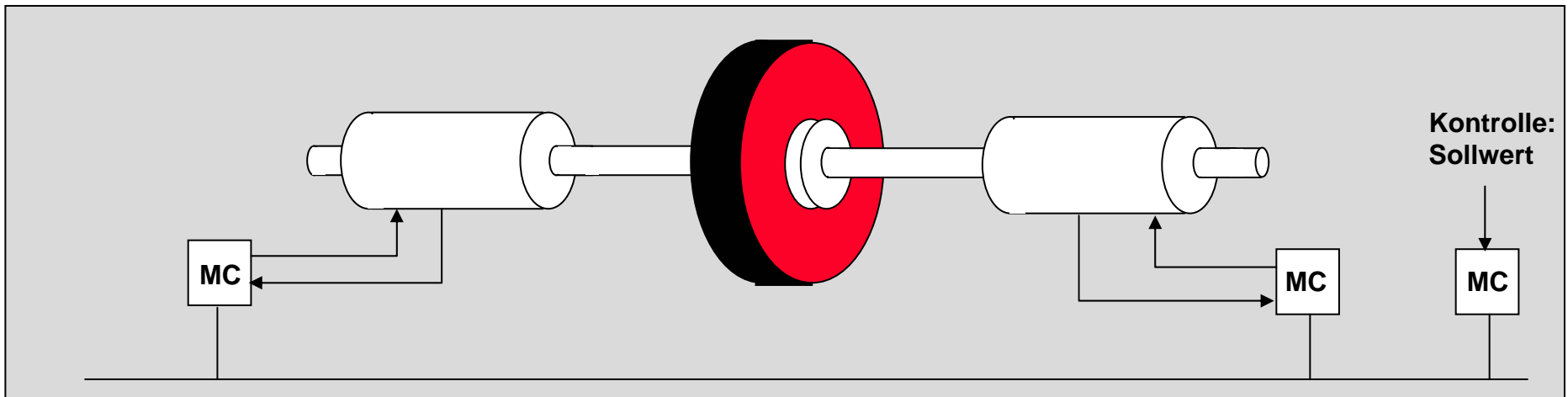
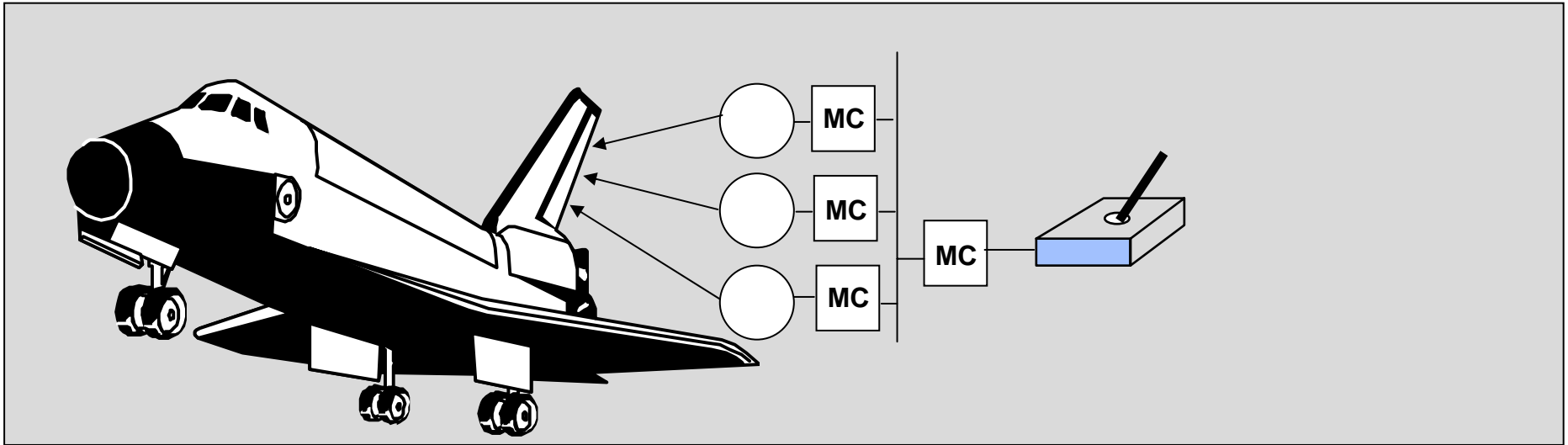


End-to-End Kommunikationskosten



1. Sendetask wird bereit
2. Spätester Zeitpunkt, an dem die Nachricht m in die GW eingeordnet ist
3. Alle Pakete der Nachricht m wurden aus der GW gelesen und entfernt.
Die Übertragung des letzten Paketes beginnt.
wctt: worst case transmit time
ppp: physical propagation delay
4. Letztes Paket der Nachricht erreicht den CC des Empfängers
5. Der "Paket empfangen" Interrupt wird ausgelöst.
t.notif: worst case Verzögerung zwischen dem abgeschlossenen Empfang eines Pakets durch den CC und dem Zeitpunkt, an dem der CC die CPU notifiziert.
Zu diesem Zeitpunkt wird die Empfangstask bereit.

Realzeit-Szenarien, die Konsens über das **Was** und **Wann** benötigen!



Ebenen in einem Kommunikationssystem

**Kommunikations-
ebene**

Netzebene

**1-zu-1,
1-zu-n**

**zeitbeschränkter, zuverlässiger
Nachrichtenaustausch**

**Kooperations-
ebene**

Systemebene

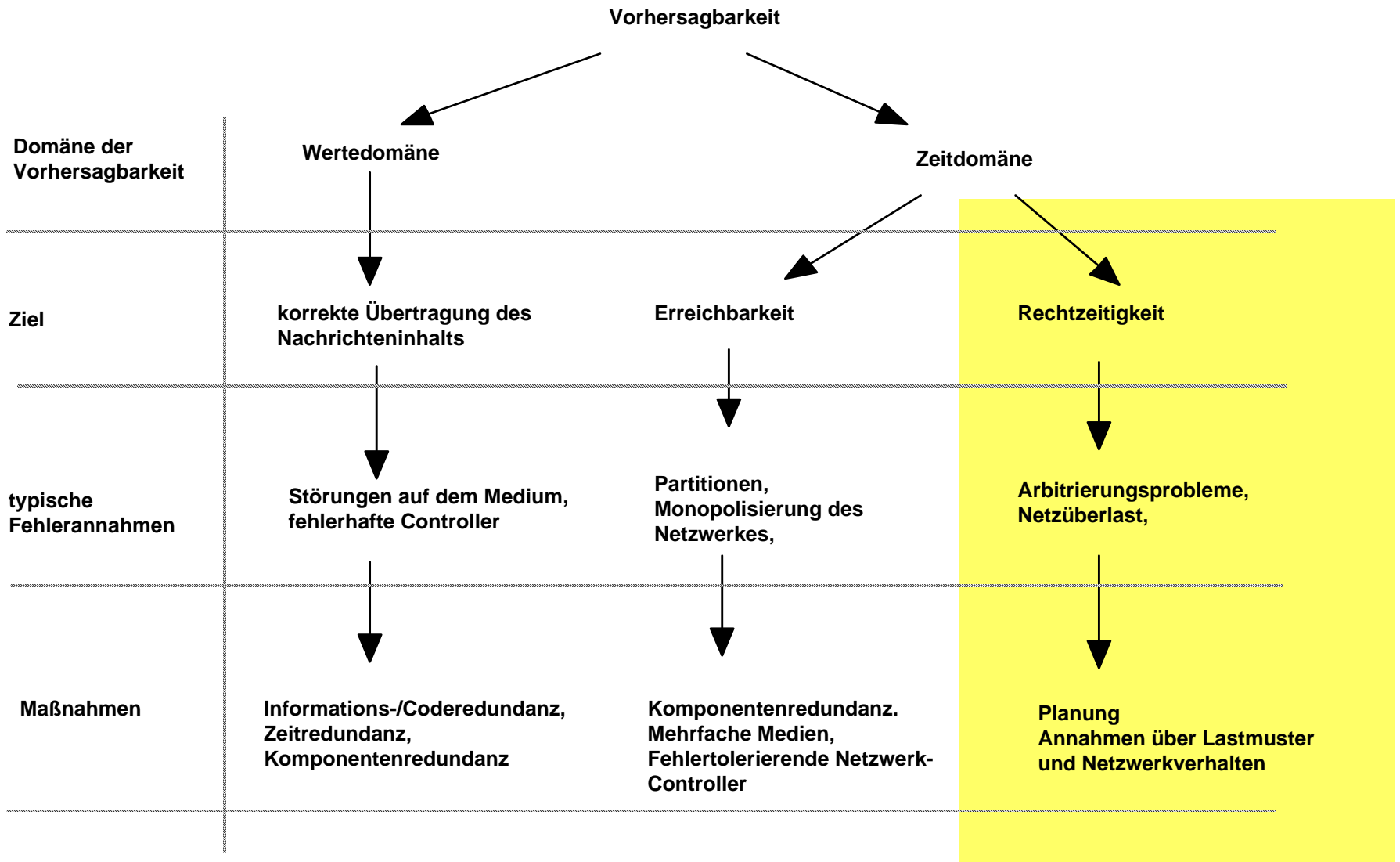
n-zu-m

Konsensus über:

- **die Ordnung der Nachrichten**
- **Mitgliedschaft in einer Gruppe**
- **Zeitpunkt des Ausliefern der
Nachricht**

Aspekte der Vorhersagbarkeit

- **Beschränkte Nachrichtenlaufzeit**
- **Synchronität (Synchrony)**
- **Vorrangregelung (Priorität)**
- **Reaktionsfähigkeit, d.h. eine wichtige Nachricht soll von weniger wichtigen nicht verzögert werden. Daraus folgt der Zielkonflikt zwischen Durchsatz und Reaktionsfähigkeit eines Echtzeitkommunikationssystems**



Metriken der Synchronität

Wie gut ist ein Kommunikationsmedium für Echtzeitanwendungen geeignet ?
Was sind die Kriterien anhand man die Eignung feststellen kann?

Problem 1:

Wie groß ist der Unterschied der Laufzeiten **einer** Nachricht zu **verschiedenen** Knoten ?

Problem 2:

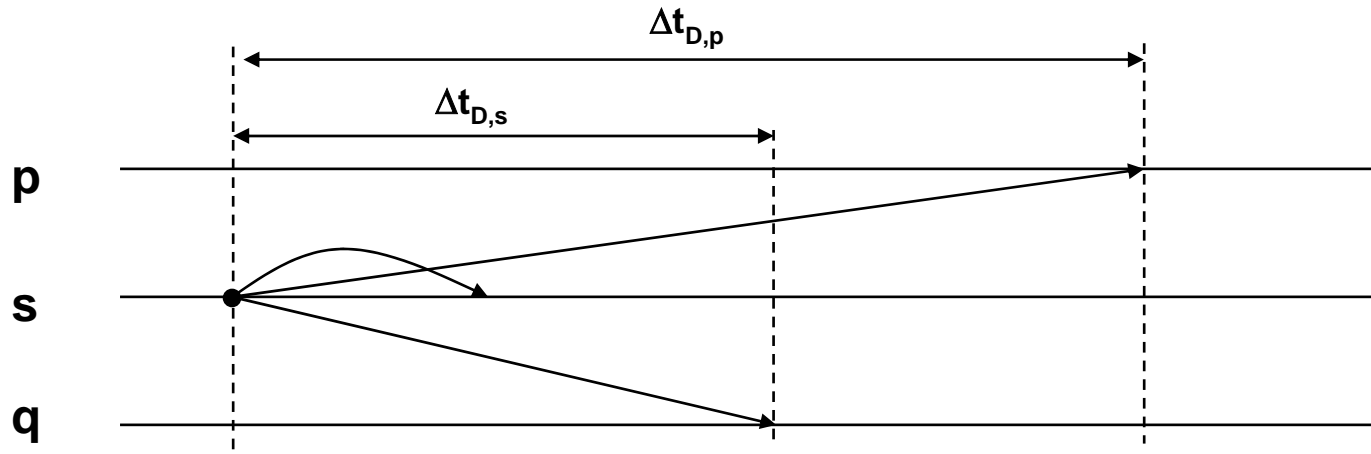
Wie groß ist der Unterschied der Laufzeiten **verschiedener** Nachrichten zu **einem** Knoten ?

Metriken der Synchronität

Definition: Auslieferungszeit einer Nachricht (delivery time)

$$\Delta t_{D,p} = t(\text{deliver}_p(m)) - t(\text{send}(m))$$

$\Delta t_{D,p}$: Intervall zwischen dem Versenden einer Nachricht m und ihrer Auslieferung an Prozeß p



Metriken der Synchronität

Tightness τ

Definition: Tightness

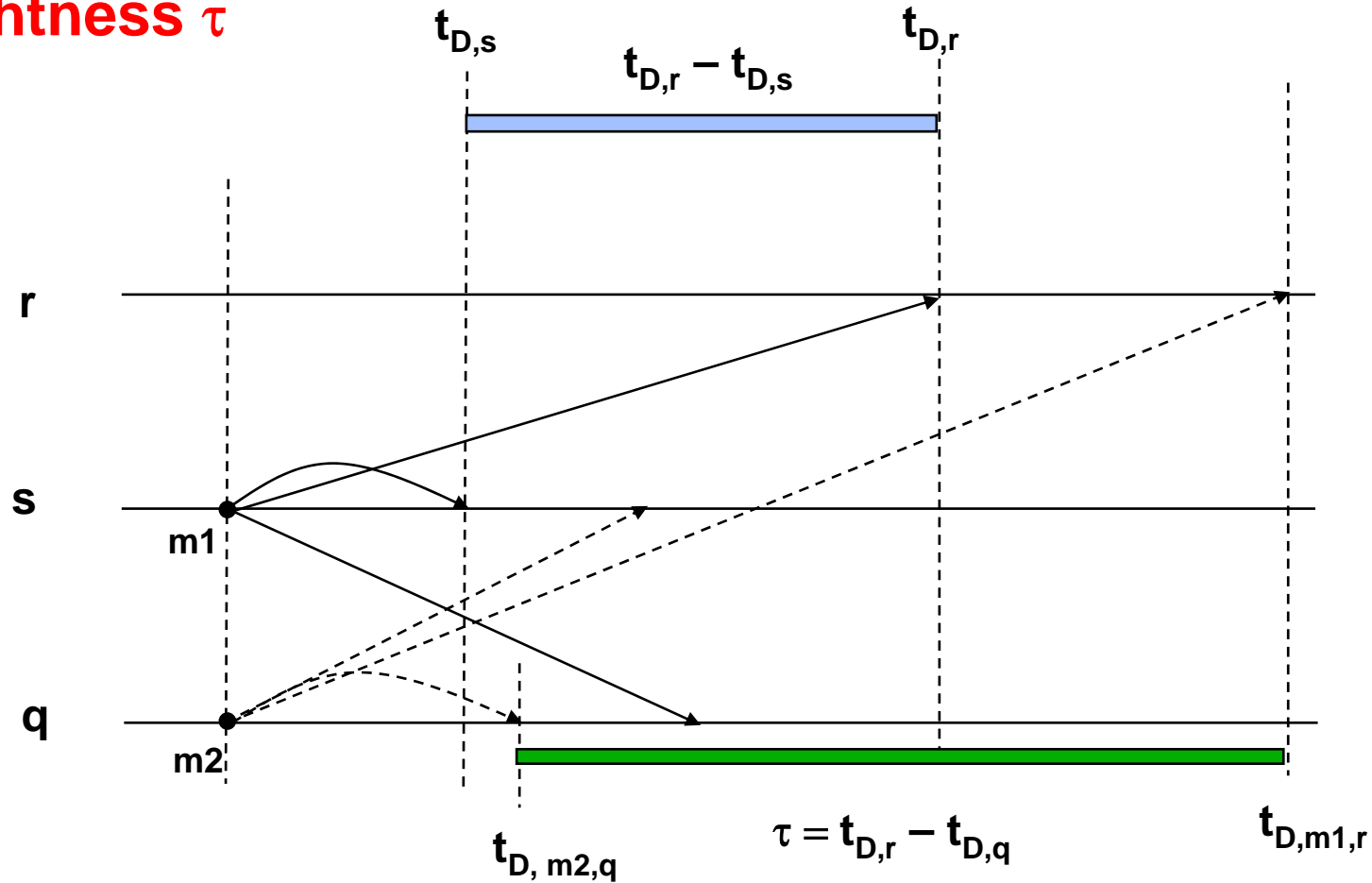
$$\tau = \max_{m,r,q} (t_{D,r} - t_{D,q})$$

τ ist - für jede einzelne Nachricht m - die größte Differenz, die zwischen den Auslieferungszeiten dieser Nachricht bezogen auf die beliebigen Teilnehmer r und q jemals auftritt.

Tightness ist ein Maß für den Unterschied der Laufzeiten **einer** Nachrichten zu **verschiedenen** Knoten ?

Metriken der Synchronität

Tightness τ



$$\tau = \max ((t_{D,r} - t_{D,s}), (t_{D,r} - t_{D,q}))$$

Metriken der Synchronität

steadyness σ

Definition: Steadyness

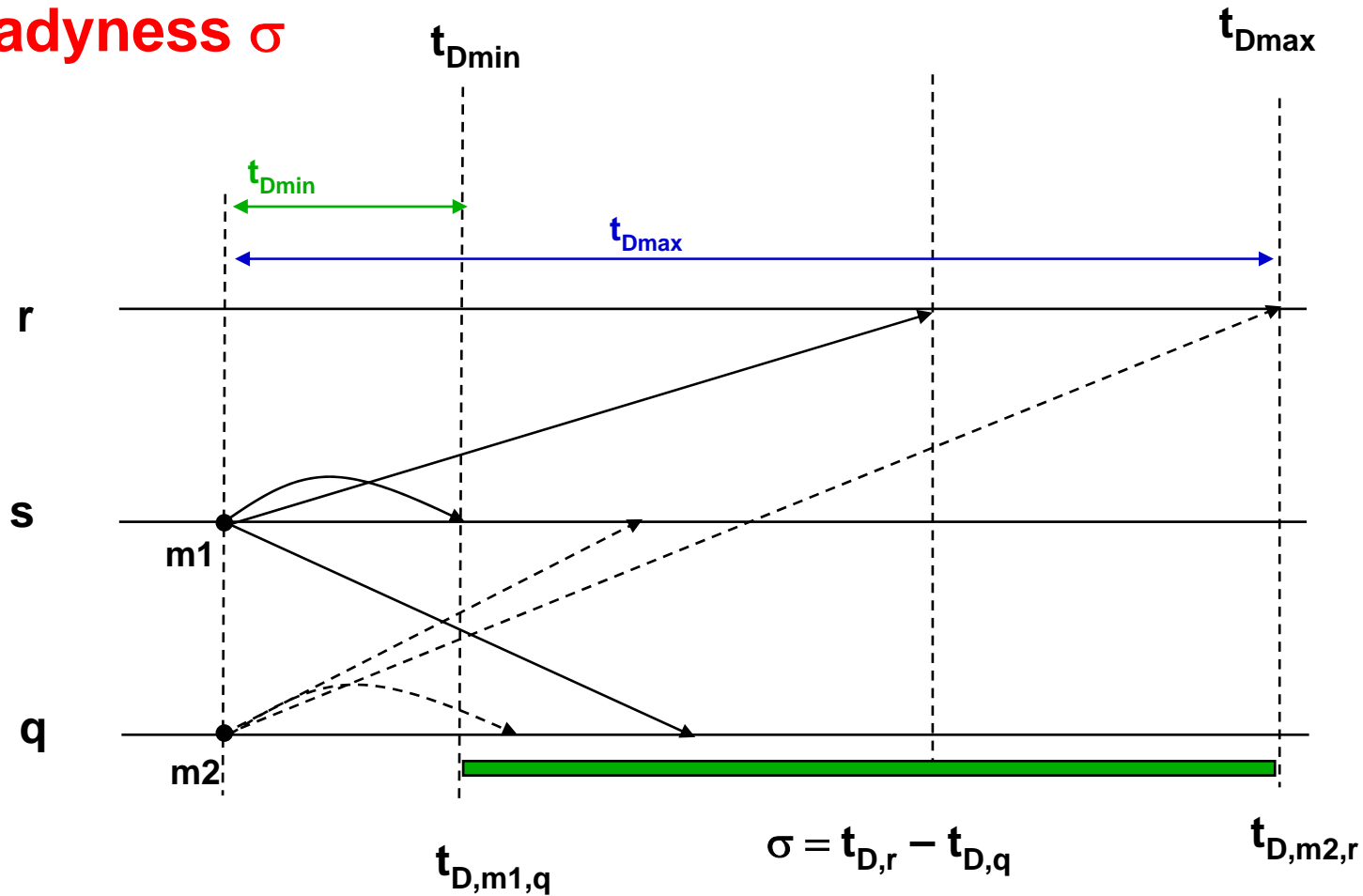
$$\sigma = \max_p (t_{D_{\max}} - t_{D_{\min}})$$

σ ist die größte Differenz, die zwischen dem Maximum $t_{D_{\max}}$ und dem Minimum $t_{D_{\min}}$ der Auslieferungszeiten, die bei beliebigen Teilnehmern beobachtet werden kann.

Steadyness ist ein Maß für den maximalen Unterschied der Laufzeiten **verschiedener** Nachrichten (zu verschiedenen Knoten).

Metriken der Synchronität

Steadyness σ



$$\sigma = \max ((t_{D,r} - t_{D,s}), (t_{D,r} - t_{D,s}))$$

Quelle für Unvorhersagbarkeiten ?

Quelle für Unvorhersagbarkeiten ?

Kommunikationsnetz ist gemeinsam genutztes Medium

→ Arbitrierung, Kollisionen

Sender und Empfänger müssen sich synchronisieren

→ beschränkte Puffer, verlorene Nachrichten

Fehler bei der Übertragung

→ re-send

Generelle Anforderungen an ein Kommunikationsprotokoll für Realzeitsysteme

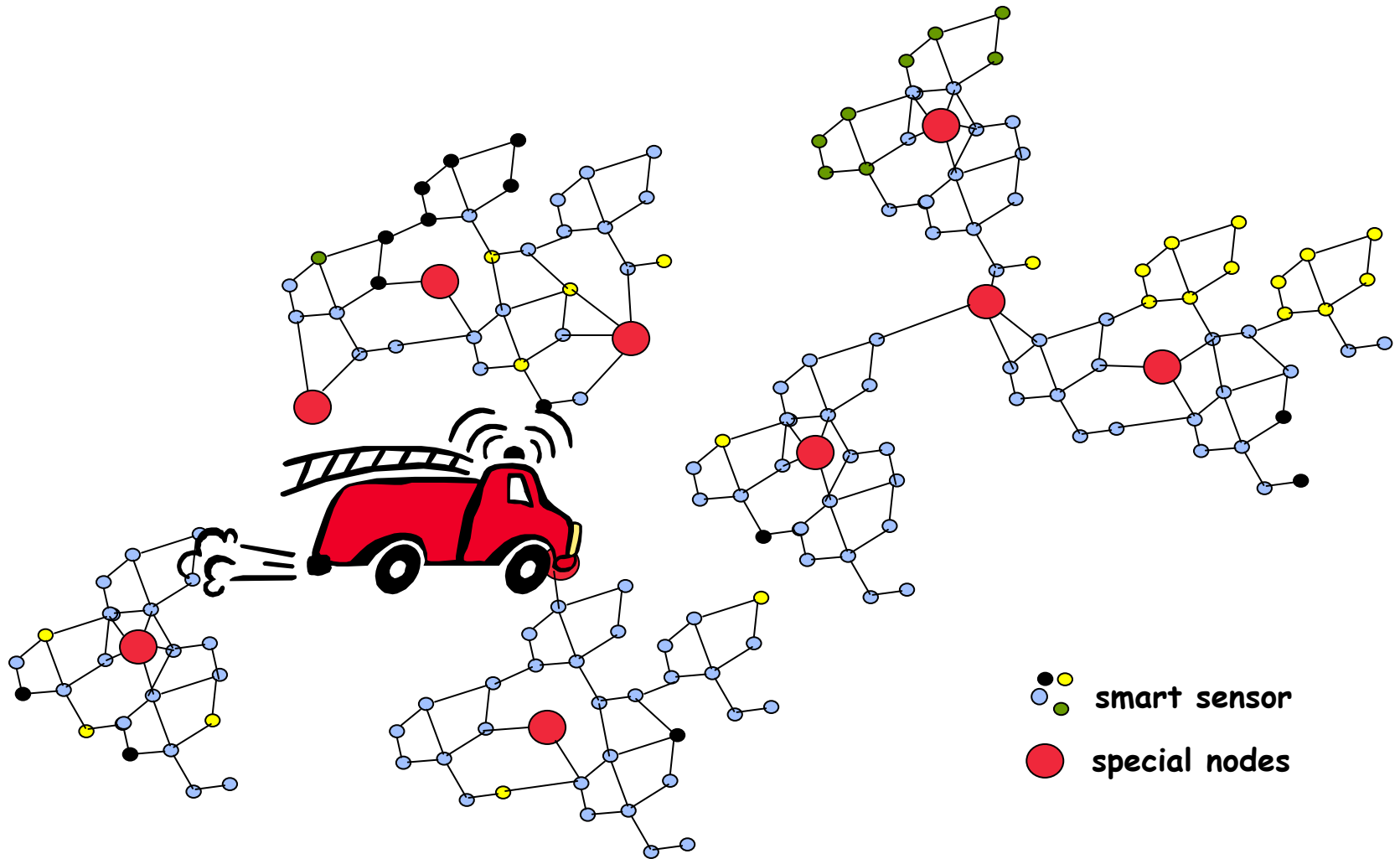
- **Beschränkte, vorhersagbare Nachrichtenlaufzeit**
- **Maximale Ausführungszeit für das Protokoll ist klein**
- **Variation der Ausführungszeit (Delay Jitter) ist klein**
- **Fehlererkennung im Empfänger und Sender**
- **Fehlererkennung mit kurzer Latenz**
- **Kein Thrashing unter Spitzenlast (konstanter Durchsatz)**
- **Unterstützung für Multicast-Kommunikation**
- **Unterstützung von Many-to-Many -Kommunikation**
- **Zusammensetzbarkeit (Composability)**

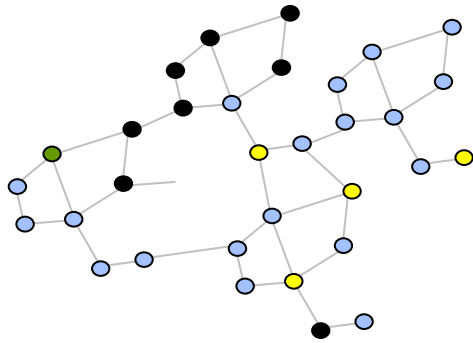
Sensornetze

Komponenten

Protokolle

Eine "vernetzte" physische Welt





Komponenten:

- heterogene Sensoren
- stationäre und mobile Einheiten
- sehr hohe Anzahl der Komponenten
- Wegwerfprodukte (im wahrsten Sinne des Wortes)
- Lebensdauer = Batterielebensdauer
- beschränkte Rechen- und Speicherressourcen

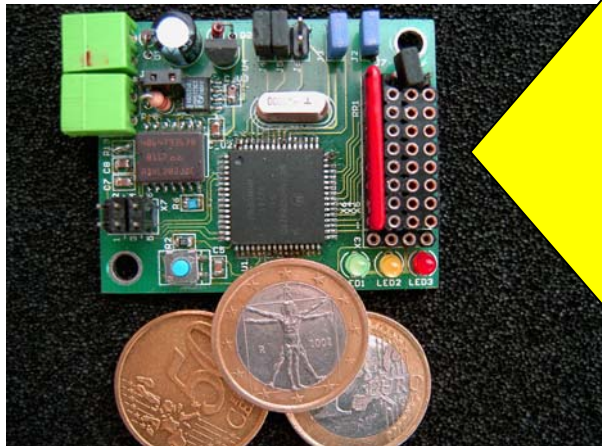
Verhalten:

- spontanes Verhalten
- nicht immer aktiv
- Arbeitsteilung erforderlich

Netzwerk:

- beschränkte Bandbreite der Kommunikation
- Multi-hop
- Alterung der Information
- Qualität der Verbreitung

Hardware für Sensornetze "Smart Dust"



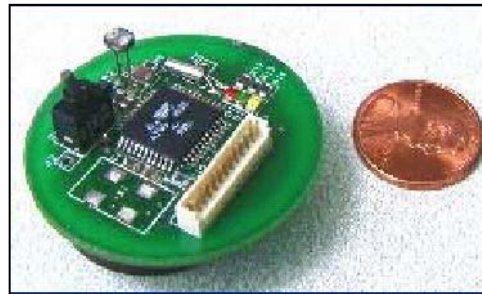
Developed Sensors at CORE

- infrared motion detector
- infrared distance sensor
- acceleration sensor
- embedded gyro
- weather station
- magnetic field detector
- in-house location system



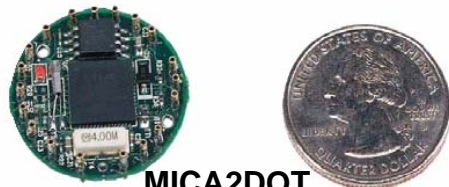
68HC11 CAN-Sensor Boards, CORE, Ulm

tiny-board, CORE, Ulm



WeC „Smart Rock“ UCB

a mica mote,
Berkeley, Crossbow

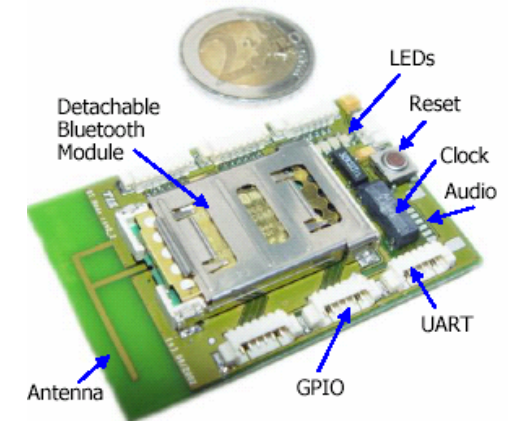


MICA2DOT



The EYES prototype

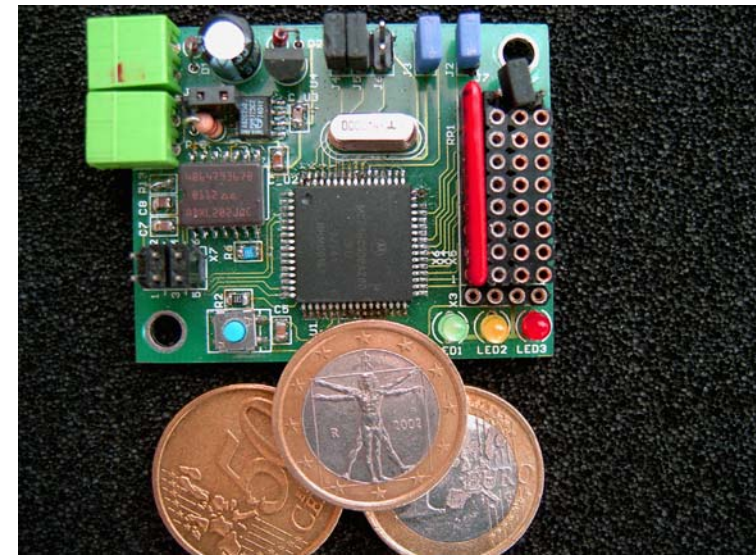
Smart-its: ETH Zurich,



cpu, memory
on back side

Tiny Properties

**Designed for experimentation:
Basic Board + Piggyback extension**



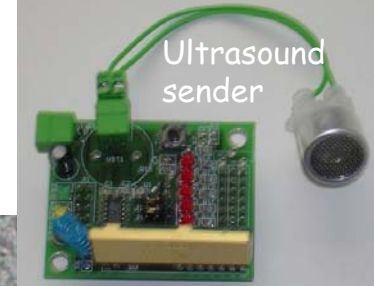
Basic board:

- Processor 68HC908AZ60 (60k Flash, 2k RAM)
- Power regulator (linear or switched) 6-14 V
- LEDs for checks, configuration jumpers
- CAN-Bus Network Interface
- Sockets for AD, C&C, digital I/O
- Sockets for asynch.and synch serial comms.

Power consumption:

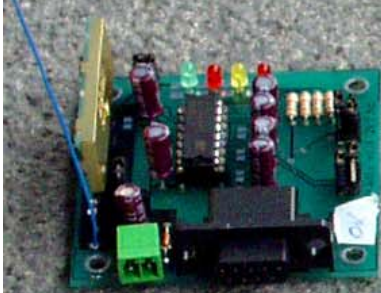
- Processor ~ 250 mW @ 16MHz
- Radio link (Easy Radio, 19kbit/sec): ~150mW(transmit), ~75mW(idle)
- 9V Block (565 mAh): ~ 8h@continuous operation, ~30 days@10ms/sec

the TINY family



Magnetic field detector

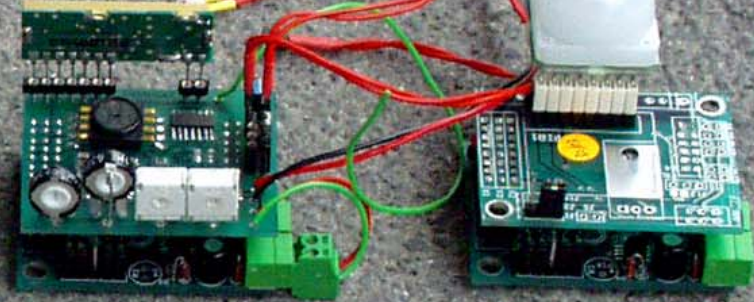
weather station:
pressure, temperature,
humidity, light intensity



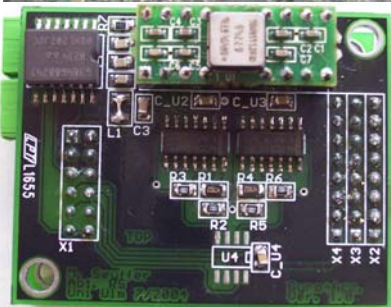
PIR: Passive InfraRed
motion detector



TINY with on-board
acceleration/
inclination sensor



Wireless modem for
PC serial line



gyro



TINY basic board: 68HC908 AZ60: 60 k Flash, 2 k RAM,
CAN, serial, extension ports for piggy-back boards.



Low Power Wireless
comm. board.

TINY Family: Motorola 68HC908AZ60A, CAN-Bus, Serial Line, Port for Extension Boards, phys. size: 50x40 mm, power consumption: approx. 55 mA @ 16MHz, Power input: 6-12 V, linear (< 100mA) or switched power regulator. Available sensor/actuator extension boards: PIR, weather, distance, acceleration, DC motor control, radio transceiver

Netze

drahtlose Kommunikationsnetze geringer Bandbreite

(noch) keine Standards (mit Einschränkungen: ZigBee)

Wechsel von Schlaf- und Wachphasen problematisch für MAC-Layer Protokolle

Inhärent Multi-hop

Adress-, inhalts- und ortsbasiertes Routing

Feldbusse und Sensornetze

Gemeinsamkeiten:

- ➔ **transferieren Informationen zur Erfassung und Kontrolle der physischen Umgebung,**
- ➔ **transferieren Informationen, die der Alterung unterworfen sind,**
- ➔ **individuelle zeitliche Beschränkungen sind eher Grundlagen der Entwurfsentscheidungen als der Durchsatz,**
- ➔ **berücksichtigen Zielkonflikte bezüglich des Energieverbrauchs, der Bandbreite, der Zuverlässigkeit und der Priorisierung des Nachrichtenverkehrs.**

Unterschiede:

	Feldbusse	Sensornetze
Anzahl der Knoten	wenige bis mittel	sehr viele (theoretisch)
Prozess-sicherheit	sehr hoch bis mittel	niedrig
Vorhersagbarkeit	sehr hoch	mittel bis niedrig
Anzahl der hops	ein bis wenige	viele
indiv. Ausfallsicherheit	sehr hoch bis hoch	sehr niedrig

Feldbusse und Sensornetze

- o **Einführung**
- o **Zuverlässigkeit und Fehlertoleranz**
 - * Attribute und Maße der Zuverlässigkeit
 - * grundlegende Techniken der Fehlertoleranz
- o **Uhrensynchronisation**
- o **Die physische Übertragungsschicht**
- o **Protokolle für zeitbeschränkte zuverlässige Kommunikation**
 - * Einführung und Problemanalyse
 - * Time Triggered Protokoll (TTP/C)
 - * CAN, ProfiBus, WorldFip, Lon (Echelon)
 - * Echtzeitfähige CSMA-Netze (VTCSMA)
 - * Token Protokolle
- o **Sensornetze**
 - * Anforderungen an Protokolle für Sensornetze
 - * Protokolle zur drahtlosen Kommunikation
 - * Energie-effiziente MAC-Protokolle
 - * Routing