http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

# Multimedia-Systems: Resources and Quality of Service (QoS)

Prof. Dr.-Ing. **Ralf Steinmetz**
Prof. Dr. **Max Mühlhäuser**

***MM**: TU Darmstadt - Darmstadt University of Technology,*
*Dept. of of Computer Science*
*TK - Telecooperation, Tel.+49 6151 16-3709,*
*Alexanderstr. 6, D-64283 Darmstadt, Germany, max@informatik.tu-darmstadt.de Fax. +49 6151 16-3052*

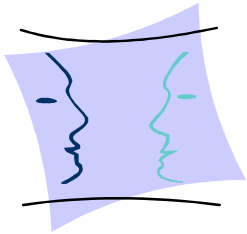
***RS:** TU Darmstadt - Darmstadt University of Technology,*
*Dept. of Electrical Engineering and Information Technology, Dept. of Computer Science*
*KOM - Industrial Process and System Communications, Tel.+49 6151 166151,*
*Merckstr. 25, D-64283 Darmstadt, Germany, Ralf.Steinmetz@KOM.tu-darmstadt.de Fax. +49 6151 166152*
*GMD -German National Research Center for Information Technology*
*httc - Hessian Telemedia Technology Competence-Center e.V*

Scope

Contents

# Scope

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser



Scope

Contents

**Usage**

- Applications
  - Learning & Teaching
  - Design
  - User Interfaces

**Services**

- Content Processing
- Documents
- Security
- ...
- Synchronization
- Group Communications

**Systems**

- Databases
- Programming
- Media-Server
- Operating Systems
- Communications
- Opt. Memories
- Quality of Service
- Networks

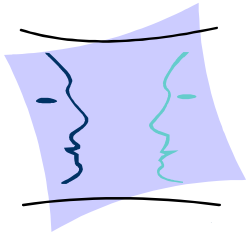
**Basics**

- Computer Architectures
- Compression
  - Image & Graphics
  - Animation
  - Video
  - Audio

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

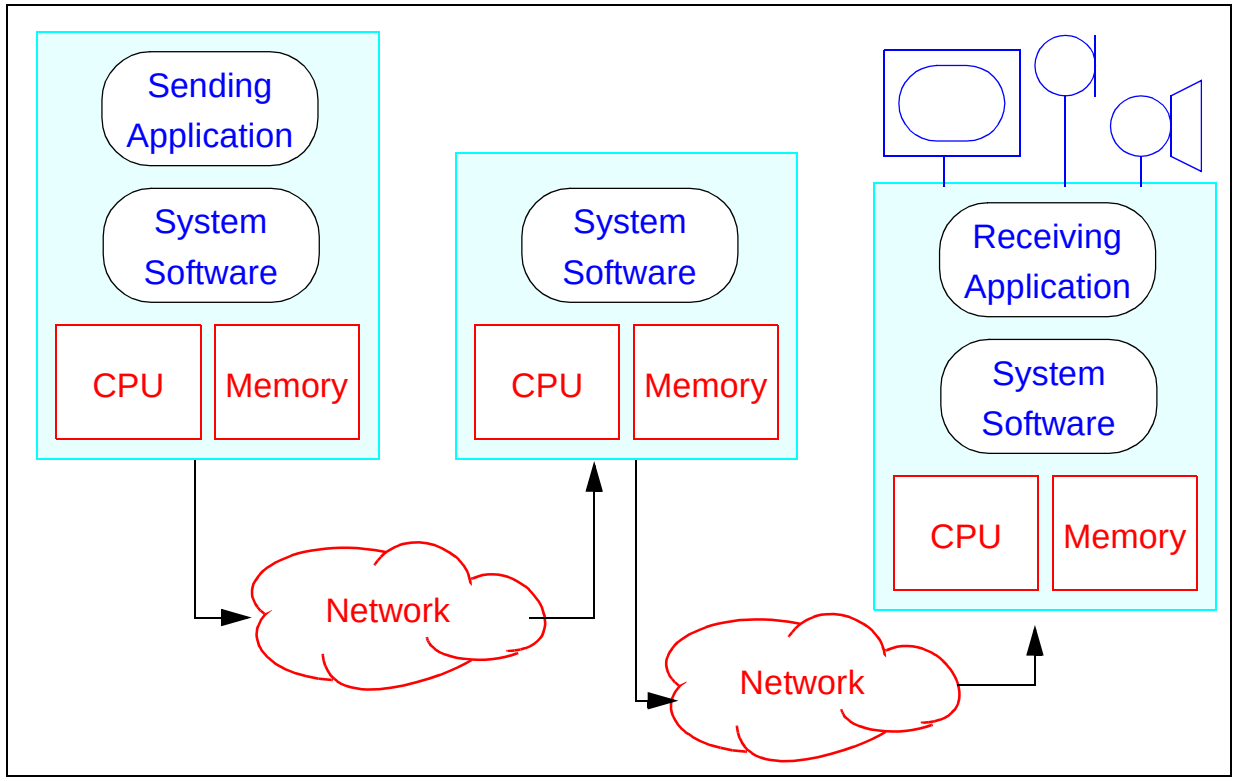
# Contents

1. Motivation
2. Characteristics of Real-time / Multimedia Systems
3. QoS - Definition
4. Resources
5. Providing QoS
   Resource Management Phases
   5.1 QoS Provisioning - Setup Phase
   5.2 QoS Provisioning - Data Processing Phase
6. QoS Architectures
7. Conclusion



# 1. Motivation

**basically, we deal with two types of systems:**




**local:**

- **harddisk recording**
- **interactive DVD**
- **computer based training**

**distributed**

- **conferencing**
- **video on demand**
- **IP-Telephony**

**Basic terminology**

- **Resources**
- **Realtime**
- **Quality of Service**

**What and how much of it do we need and how to describe that?**

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

Scope

Contents

# Motivation (cont.)

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**When we know about the needs, how to fulfill them:**

- **A QoS model and its implications**
  - QoS specification
  - QoS calculation
  - QoS enforcement

**QoS has different implications in different fields:**

- **Operating system / Resource scheduling**
- **File system organization**
- **Compression**
- **Communication system support**
- **Media synchronization**
- **...**
- **User Interface**

Scope

Contents

# 2. Characteristics of Real-time / Multimedia Systems

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Real-time System:**

> ***"A system in which the correctness of a computation depends not only on obtaining the right result, but also upon providing the result on time."***

**Real-time Process:**

> ***"A process which delivers the results of the processing in a given time-span."***

**Real-time Application - examples**

- **Control of temperature in a chemical plant**
  - driven by interrupts from external devices
  - these interrupts occur at irregular and unpredictable intervals
- **Example: Control of a flight simulator**
  - execution at periodic intervals
  - scheduled by timer-service which the application requests from the OS

**Common characteristics:**

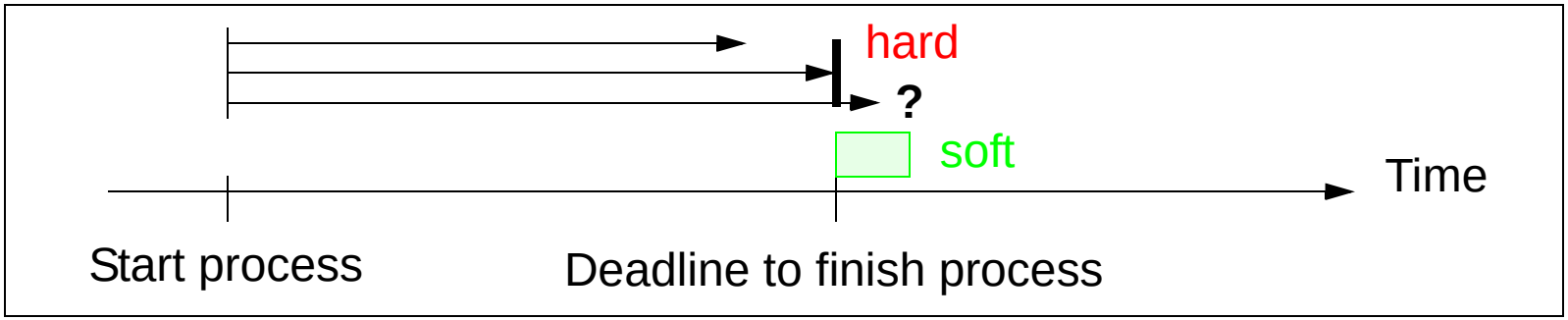
- internal and external events that occur periodically or spontaneous
- correctness also depends on meeting time constraints !

Scope

Contents

# Deadlines in Realtime Systems

**A <u>deadline</u> represents the latest acceptable time to finish an operation, e.g. for the presentation of a processing result**




- **Hard deadlines:**
  - should never be violated
  - result presented too late after deadline has no value for the user
  - violation means severe (potentially catastrophic) system failure
  - Example: Nuclear power plant

- **Soft deadlines:**
  - deadlines are not missed by much
  - in some cases the deadline may be missed, but not too many deadlines are missed
  - presented result has still some value for the user
  - Example: train/plain arrival-departure

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

# Realtime System - Requirements

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Primary goal:**

- **deterministic behaviour according to specification (results in a variety of requirements)**

**mandatory requirements:**

- **Predictable (fast) handling of time-critical events**
- **Adequate schedulability**
- **Stability under overload conditions**

**desirable requirements:**

- **Multi-tasking capabilities**
- **Short interrupt latency**
- **Fast context switching**
- **Control of memory management**
- **Proper scheduling**
- **Fine-granularity of timer services**
- **Rich set of interprocess communication and synchronisation mechanisms**

Scope

Contents

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**New application area for Realtime systems with special characteristics:**

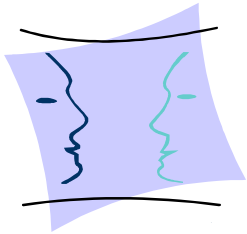
- **Typically soft real-time and not (that) critical**
- **Requirements may often be adapted to ensure proper handling**
  - e.g. Scalability (remember lecture on Compression / Video)

**Note: when reading literature, some distinctions are often not evident**

- **level of enforcement**
  **(guarantee, best-effort, "proper" handling if deadline not met, ...)**
- **"exception" handling (skip on to next data unit? just "know" ...?)**

**Characteristics:**

- **Periodic processing**
  **(increasing importance of non-periodic: interaction, VRML etc.)**
- **Large bandwidth**
- **End-to-End Guarantees**
- **Fault-tolerance**
- **Fairness**
- **Standardization**



Scope

Contents

# Quality of Media vs. Quality of Service

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Quality.... with respect to a Quality "Parameter" ("Measure"?)**

**Example Audio QoM (huge set exists in professional audio):**

- **Frequency Spectrum (linear amplification? ...)**
- **Signal2Noise Ratio SNR (noise, click, ...)**

**Example Video QoM**

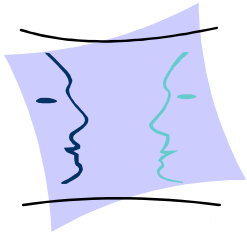
- **spatial / temporal resolution**
- **SNR**

**intuitively spoken:**

- **QoM: something like "HiFi Audio" (16..20k Hz)**
- **QoS: something like "Hi bandwidth" (1Gbps)**

**but: if application delivers HiFi Audio to the user, it does so as a "service"**

**therefore in the remainder: QoS**



Scope

Contents

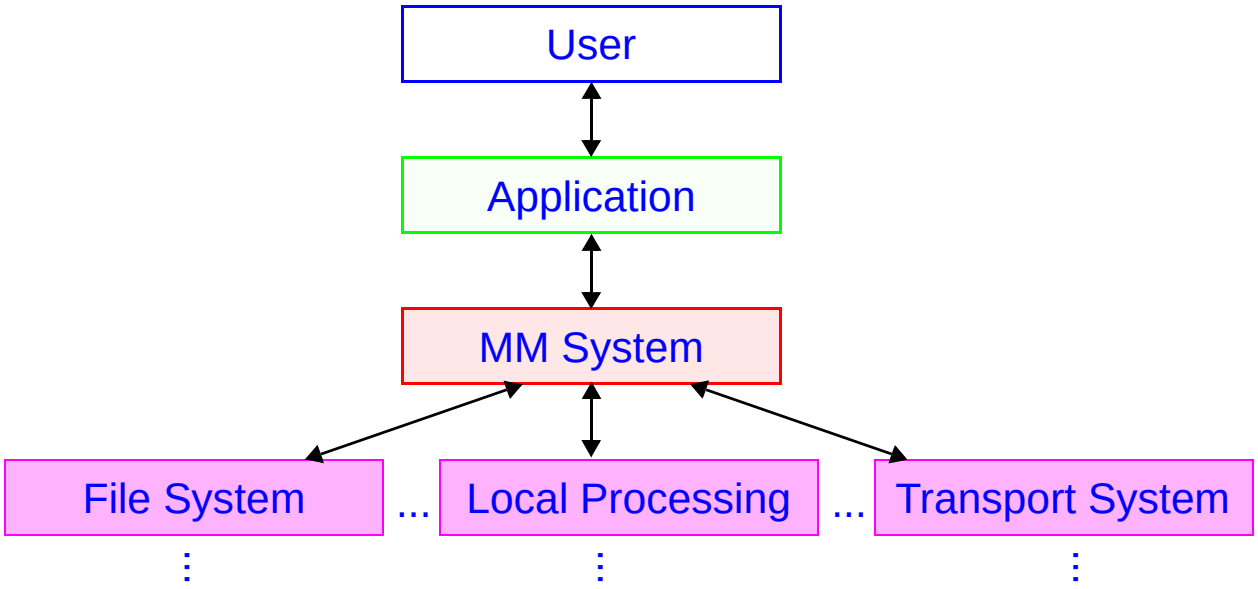
# 3. QoS - Definition

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Quality of Service =**

**„well-defined and controllable behavior of a system according to quantitatively measurable parameters“**

**Layer model:**




**Different Service Objects:**

- **Media, Media Streams --> Packet Streams**
- **Tasks**
- **Memory areas**

Scope

Contents

# QoS - Layer Model

**Examples: both qualitative / quantitative description**

**Perception QoS**
- Tolerable Synchronisation Drift
- Visual Perceptability

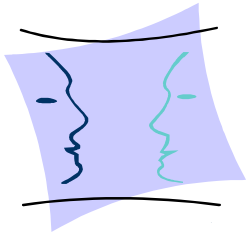
**Application QoS**
- Media Parameters
- Media (Transmission) Characteristics

**System QoS**
- CPU Rate / Usage
- Available Memory

**Communication QoS**
- Packet Size / Rate
- Bandwidth
- End-to-End-Delay

**Device QoS**
- Seek / Data Transfer Rate
- Sample Rate / Resolution

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser



Scope

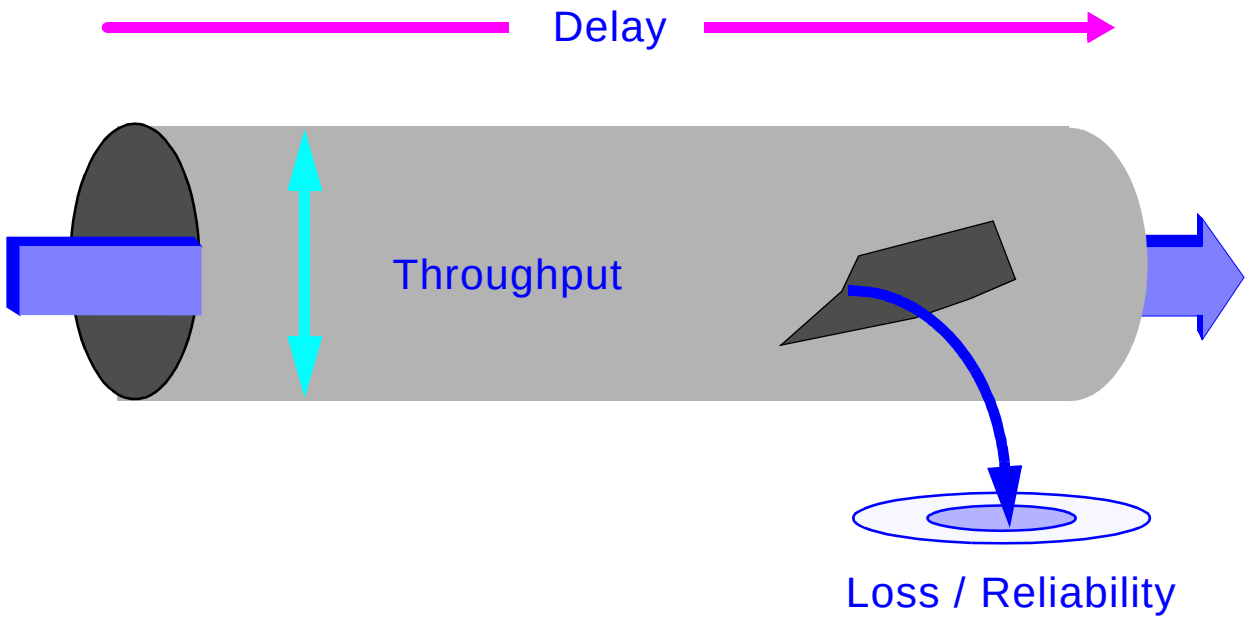
Contents

# QoS Parameters - Example Transport System

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Common parameters concerning the Transport System are:**

- **Throughput**
- **Delay / Jitter**
- **Loss / Reliability**




**but also:**

- **Security**
- **Costs**
- **Stability (Resilience)**

# Example QoS Parameters (cont.)

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Delay:**

- Maximum end-to-end delay for transmission of one packet
- Delay jitter = maximum variance of transmission (-->arrival!) times (math. definition: relative to "expected" arrival time, NOT to arrival time of other packets)

**Throughput:**

- Maximum long-term rate
  = maximum amount of data units transmitted per time interval (e.g. packets or bytes per second)
- Maximum burst size
- Maximum packet size

**Loss:**

- Sensitivity class: ignore / indicate / correct losses
- Loss rate = maximum number of losses per time interval
- Loss size = maximum number of consecutively lost packets

Scope

Contents

# Service Classes

**note: QoS parameters often subject to statistical process**

**---> mean, min, max, distribution, variance, burst-length, ....**

**Guaranteed Service**

- **values or intervals of QoS parameters**
  - deterministic (at any time)
  - statistical (consider a time interval or certain propability)

$$QoS_{min} <= P <= QoS_{max}$$

**Predictable Service**

- **consider history**
  - from the very beginning of calculation
  - in a shifting time window
- **"if it was like that in the last ..., you can rely on ..."**

**Best Effort Service**

- **no or just partial guarantees**

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

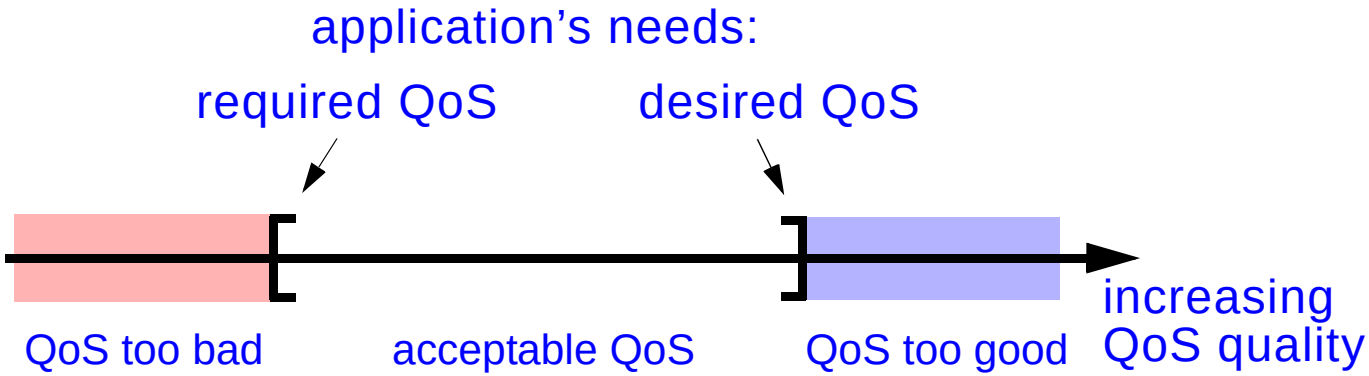
# QoS Intervals

**Parameter values result in**

- **acceptable regions**
- **inacceptable regions**

**of QoS**

**(here:) in one-dimensional intervals**



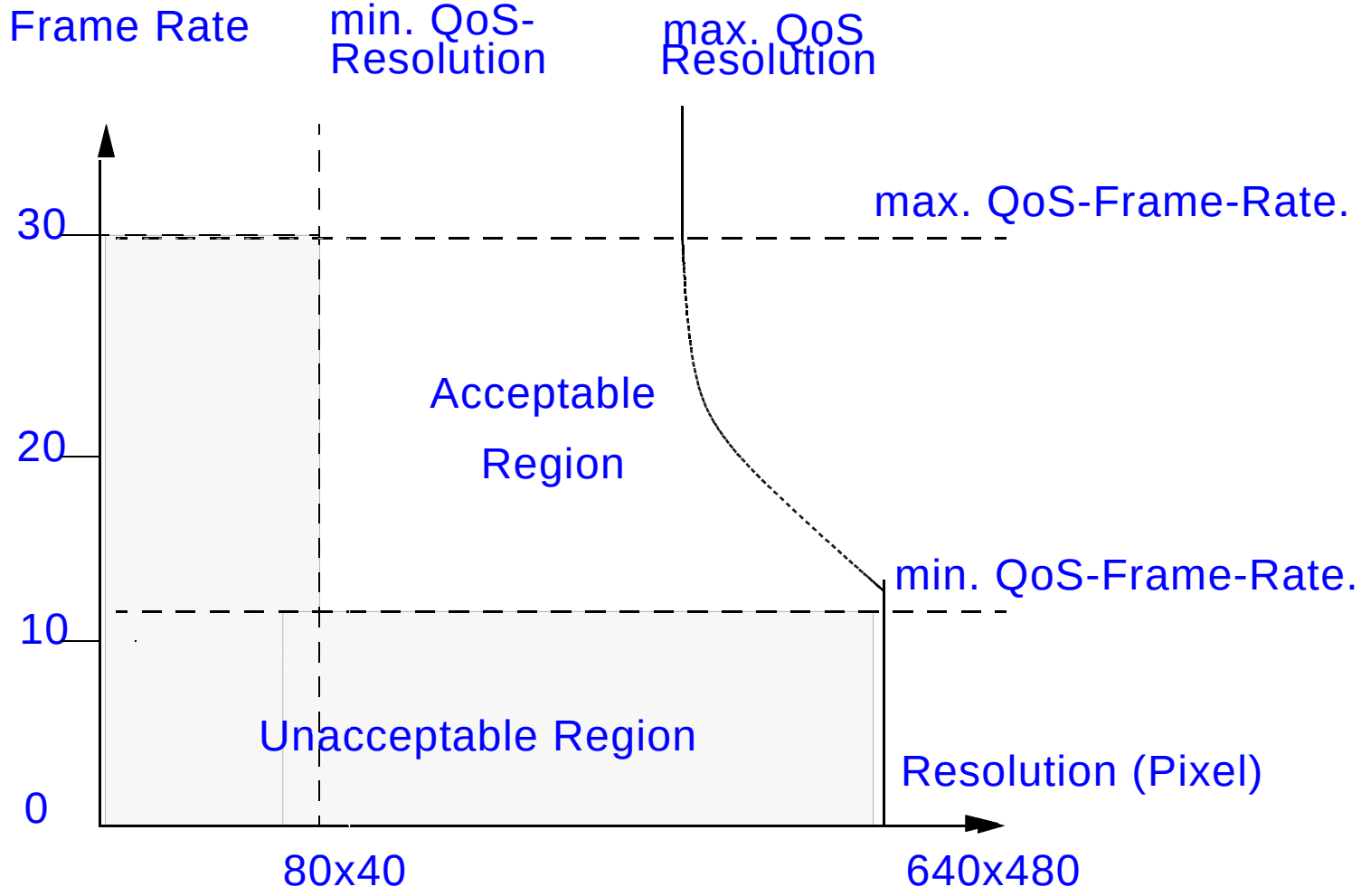

**note:**

- **below required QoS level - no reasonable service**
- **above required QoS level - unnecessary resource consumption / costs**

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

# QoS intervals

**also: multidimensional intervals**




http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

Scope

Contents

# 4. Resources

## Classification

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**by functionality**

- **active resources**
  - actively fulfill a certain task
  - e.g. processor, network adapter
- **passive resources**
  - provide “space”
  - e.g. memory, frequency spectrum, filesystem

**by availability for concurrent usage**

- **exclusive**
- **shared**

**by occurence**

- **single**
- **multiple**

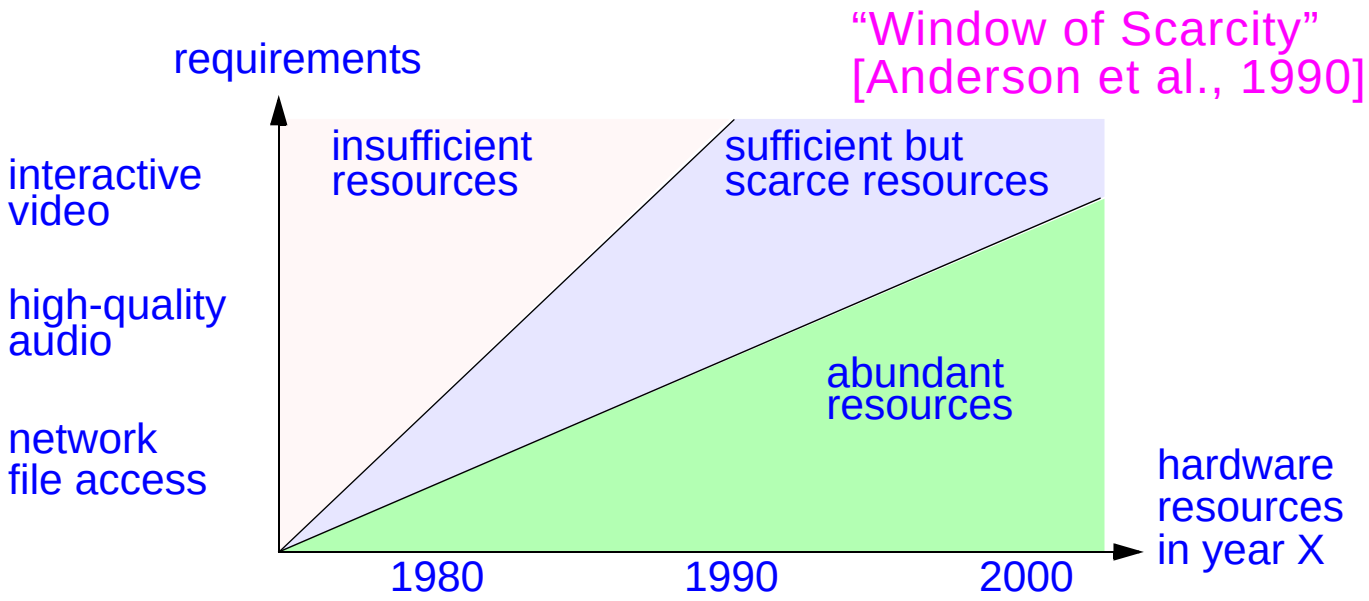
**Common parameter:**

- **Capacity - allows quantitative description**

# Resources - Availability

**Starting point:**

- **scarce, but sufficient resources**




**Goal:**

- **provide best service at lowest possible costs**

**Conclussion**

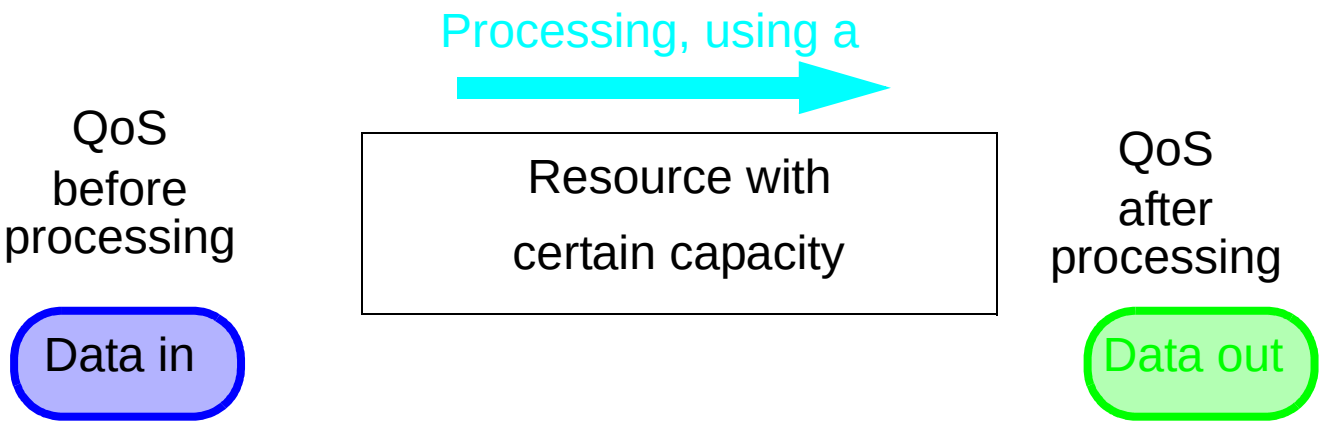
- **need for resource management**

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

# Relationship: QoS - Resources

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

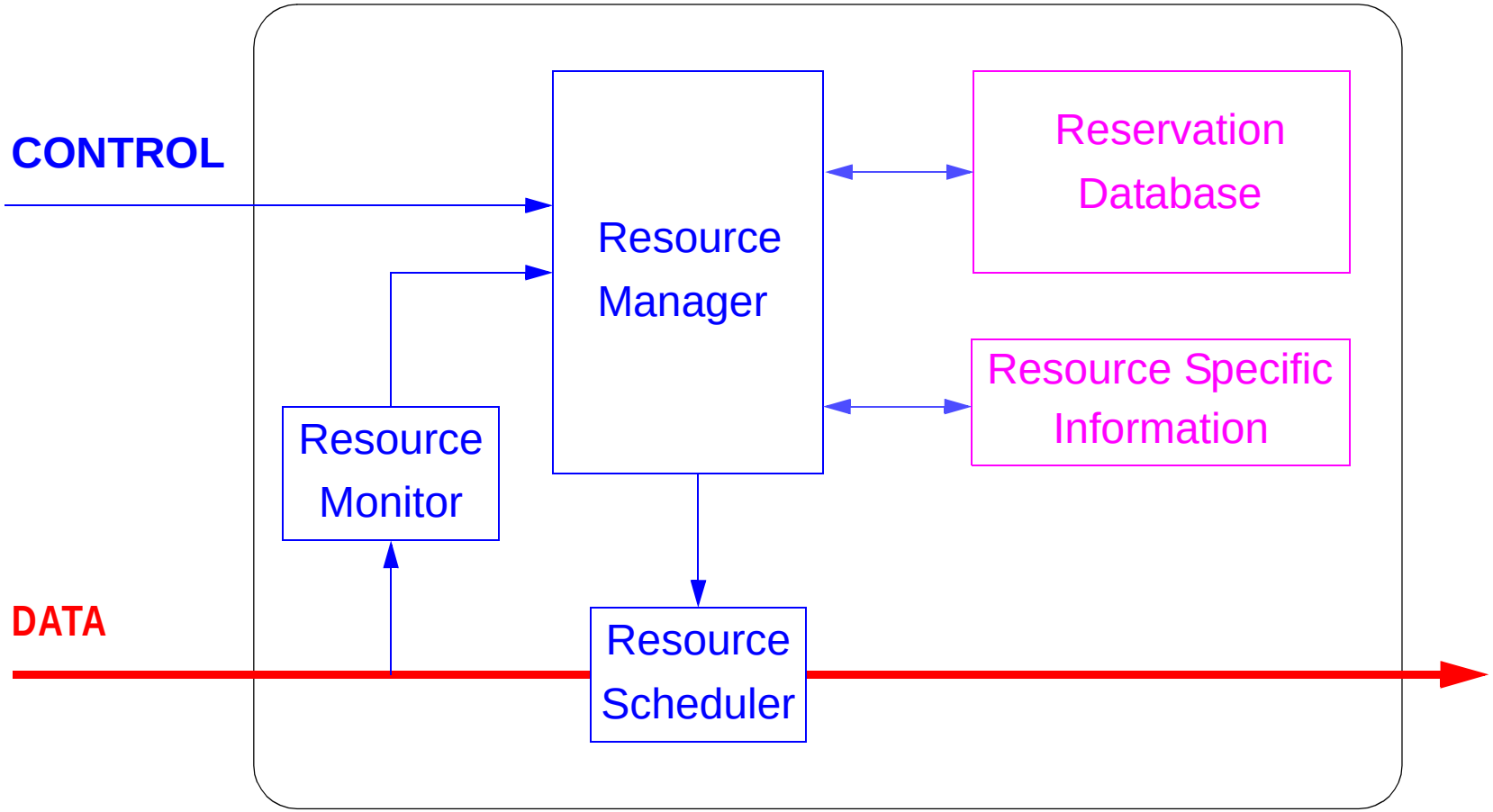
**Model:**




Scope

Contents

# Architecture




http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

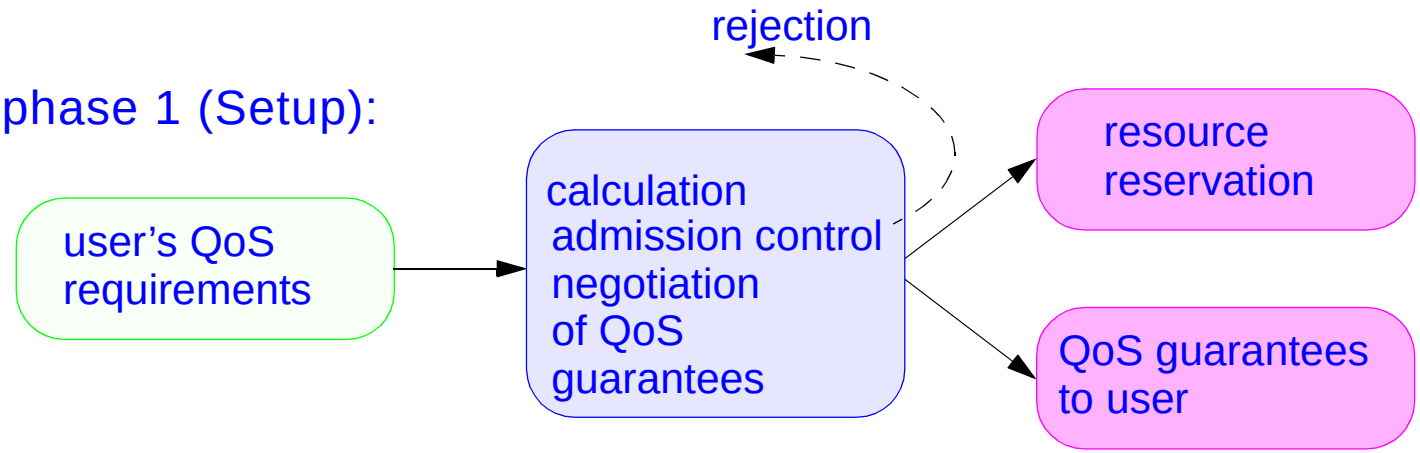
Scope

Contents

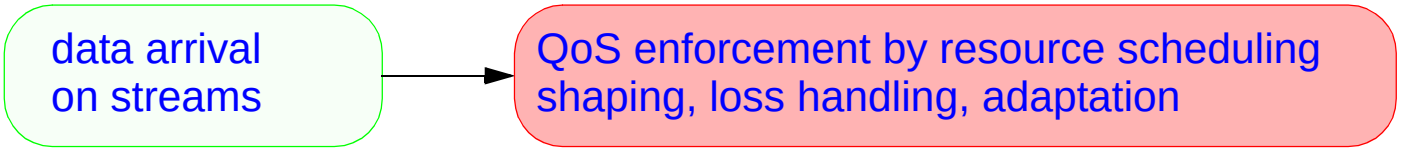
# Resource Management Phases

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

phase 1 (Setup):




phase 2 (Data processing):

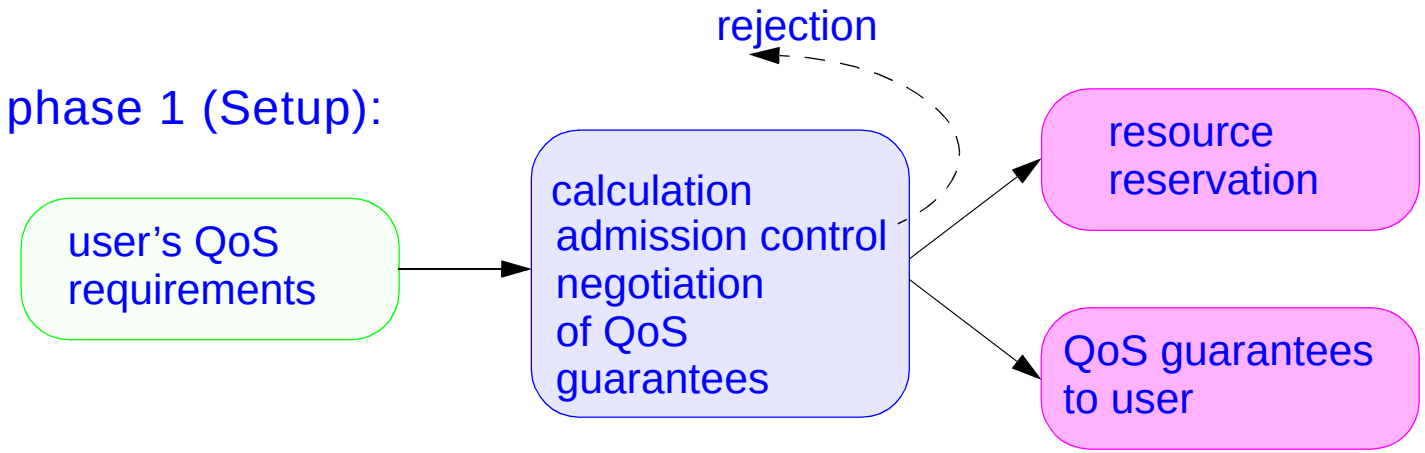



Scope

Contents

# 5.1 QoS Provisioning - Setup Phase

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser




**Definition of required parameters**

- **implicit or explicit by application or user**

**Distribution and Negotiation**

**Translation between different layers**

- **especially if they use different semantics / notations**

**Transformation**

- **QoS parameter => Resource requirements**

**Allocation and coordination of resources**

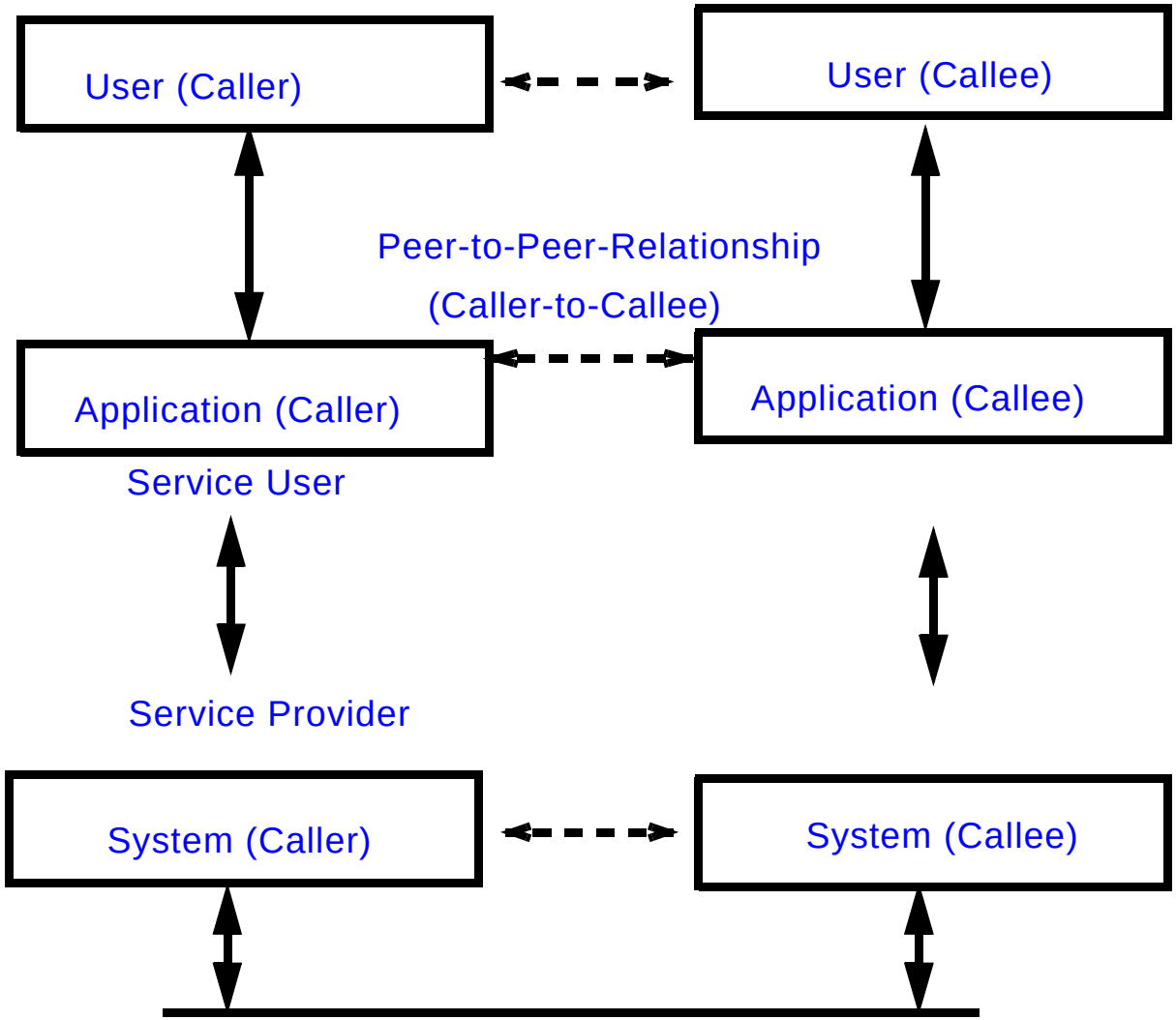
- **along path(s) source(s) => sink(s)**

Scope

Contents

# QoS Calculation and Negotiation

**Model:**




http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

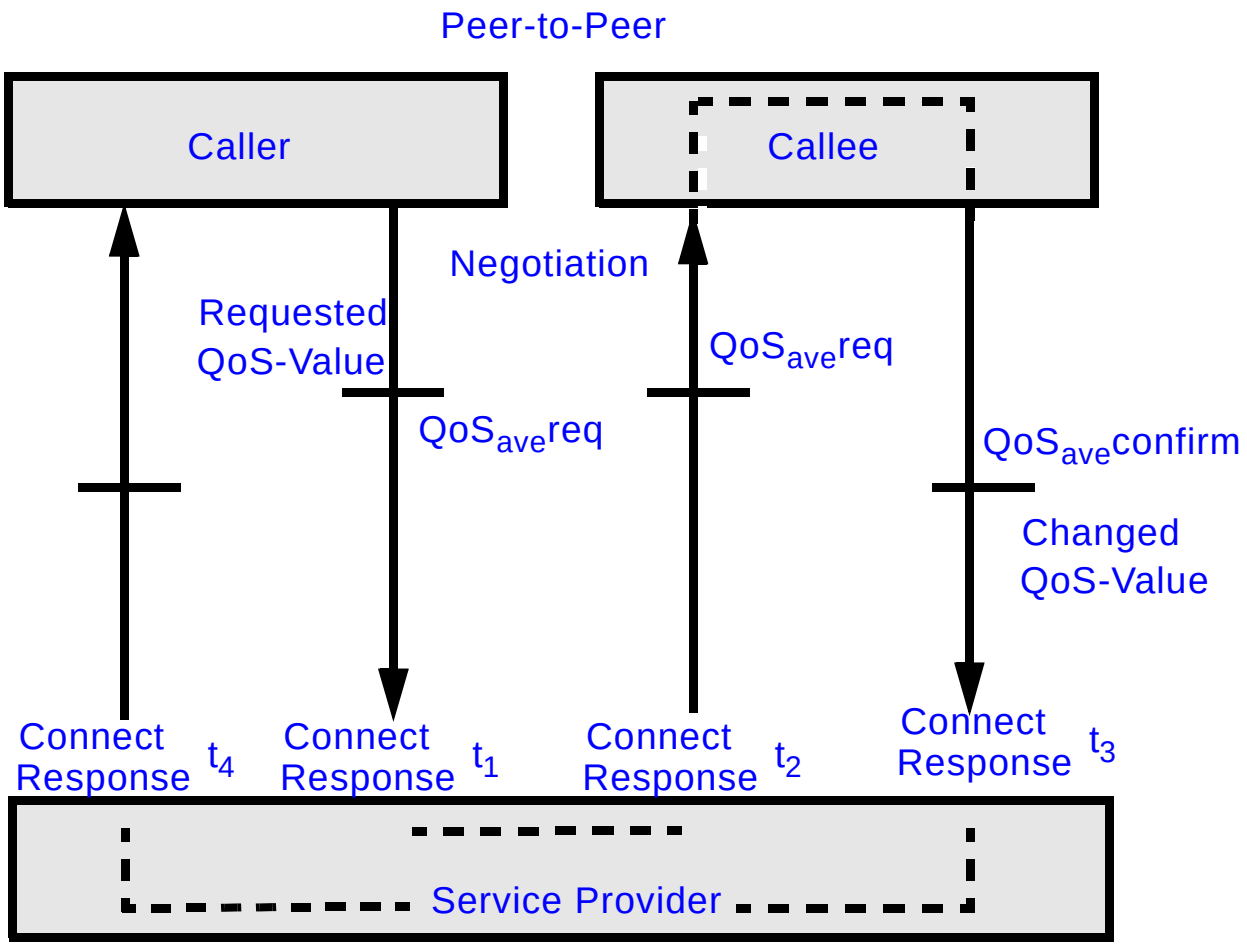
Scope

Contents

# QoS Negotiation

**bilateral peer-to-peer**

- **service provider may not modify requested QoS parameters**
- **only service user at receiver side may modify (lower) value(s) on confirm**




http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

Scope
Contents

# QoS Negotiation (cont.)

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**bilateral layer-to-layer**

- **only between adjacent parts**
  - between local service users and providers
  - between sender and network

**unilateral**

- **no modification of requested QoS parameters**
- **but just accept or reject**
- **receiver may accept QoS parameter though it can't meet it**
  - example: color TV broadcast

**hybrid**

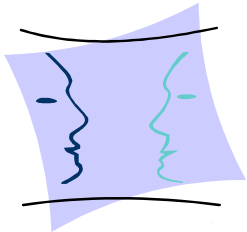
- **uses unilateral mode at a certain bilateral layer-to-layer-negotiation**
  - example: broadcast/multicast communication => heterogenity of receivers

**further:**

- **trilateral for information exchange or: ... for a limited target value**

**commonly accepted for connection-oriented layered distrib. services:**

- **initiator requests QoS, each entity accepts OR reduces QoS parameters**
- **final "Connect-Confirm" w/ overall minimum QoS, initator accepts/rejects**
- **(NOTE: "connection-oriented" vs. "flow-based" battle, see networking)**



Scope

Contents

# Admission Control

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**check, whether requested resources are and will be available**

**especially important for shared resources:**

- **CPU**
- **network paths**
- **buffer space: memory**

**simple rule:**

> **check whether sum of resources already in use and new request(s) is less or equal available resource capacity**

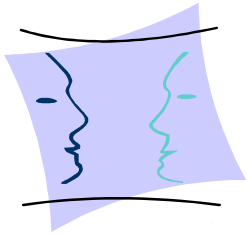
**... may be applied to**

- **availability of buffers (spatial)**
- **bandwidth (net)**

**more complex: check for schedulability**

**note:**

- **strong relationship with Pricing / Billing**
- **efficient mechanisms will use “economic feedback” to prevent users from requesting whatever they can get**



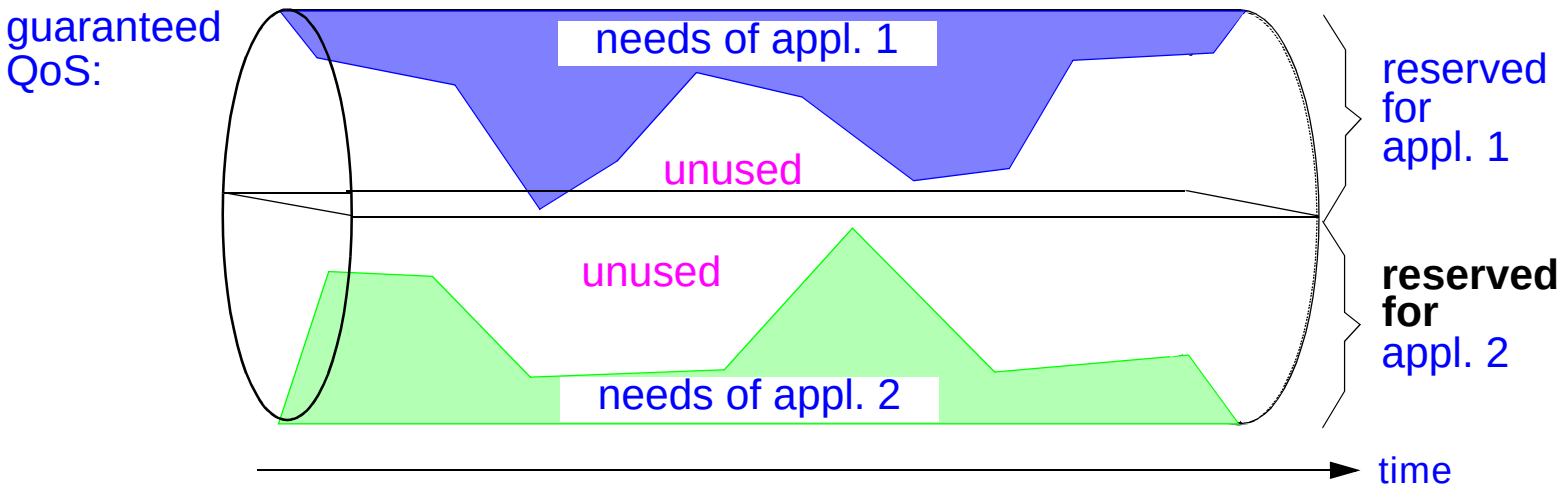
Scope

Contents

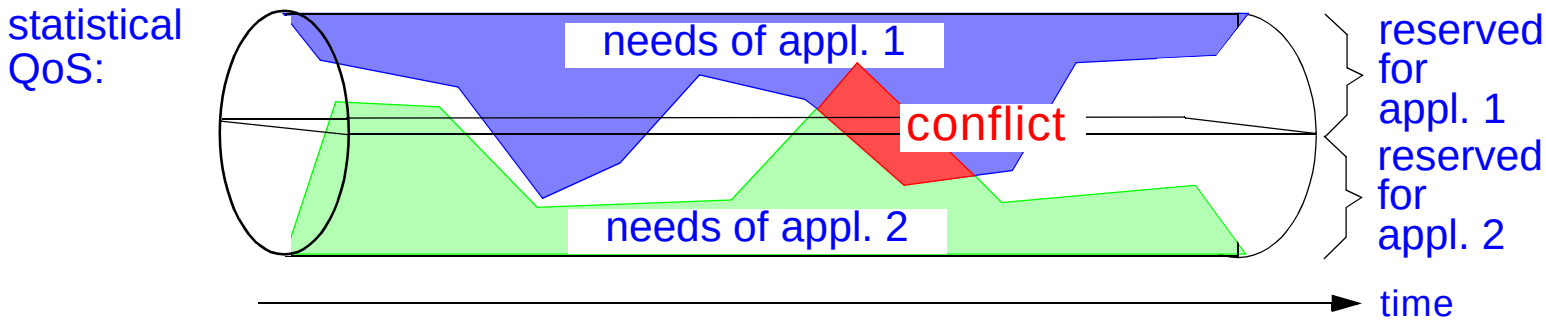
# Resource Reservation

**Fundamental concept for reliable enforcement of QoS guarantees**

- **pessimistic - results in Quaranteed QoS**




- **optimistic - results in statistical QoS**




- may use monitoring and react on overload conditions (e.g. CPU load)

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

# Resource Reservation Aspects - Example

**Example: Communication System ===> variety of aspects**

**Reservation Model**

- **Sender-initiated**
- **Receiver-initiated**
- **Explicit vs. Implicit**
- **Out-of-Band vs. In-Band**

**Reservation Style**

- **Semantics and Notation**
- **Heterogenity and Multicast-Support**

**Reservation Protocols**

- **IP V.5: ST-II**
- **RSVP (Resource reSerVation Protocol)**

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

# 5.2 QoS Provisioning - Data Processing Phase

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

phase 2 (Data processing):

data arrival on streams → QoS enforcement by resource scheduling shaping, loss handling, adaptation

**maintain resource reservations**

**use:**

- adequate traffic shaping (to ensure characteristics of processed data)
- scheduling
- feedback and adaption

**mechanisms**

**note: concurrent requests !**

Scope

Contents

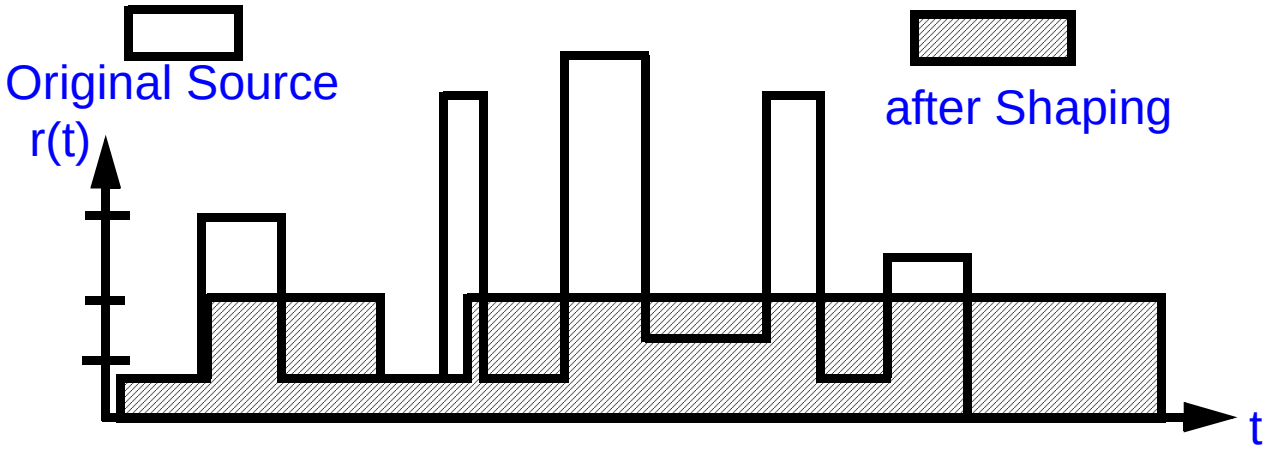
# Shaping

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

## Characteristics of Multimedia Traffic

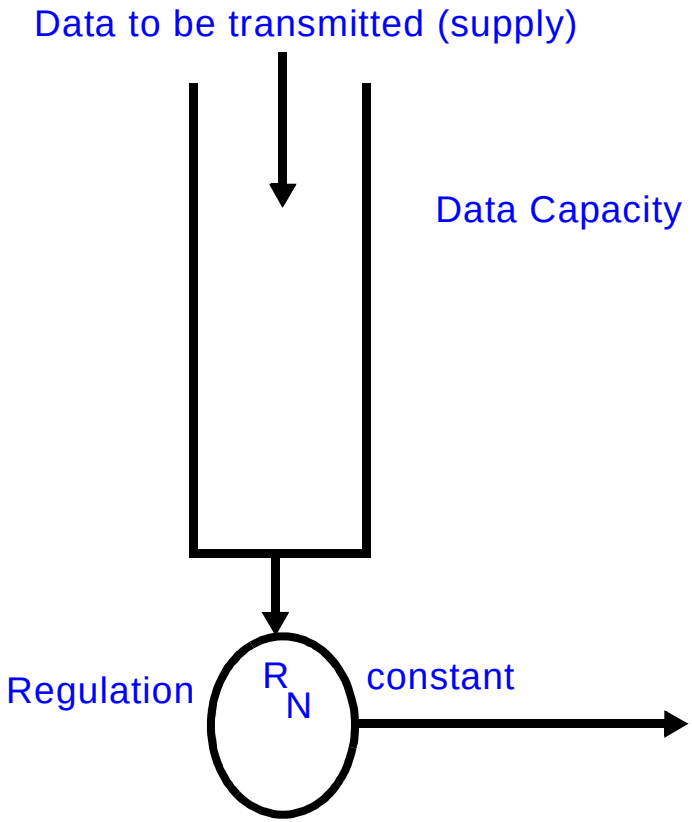
- **bursty (remember lecture Compression)**
- **concurrent requests may cause problems though guarantees could be met (e.g. buffer overflow)**

## Basic principle



# Shaping - Leaky Bucket Algorithm, (r,T) shaping

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser




**Leaky Bucket: Bucket Size**

- **determines maximum capacity till overflow (drop) *and* possible delay**

**(r, T) shaping:**

- **frames of T bits (system-wide), fraction r assigned per connection**
- **within interval T, sender may not send more than r Bits**
- **if "current packet" would exceed r -> wait for next interval (not economic)**

Scope

Contents

# Shaping: Other Algorithms

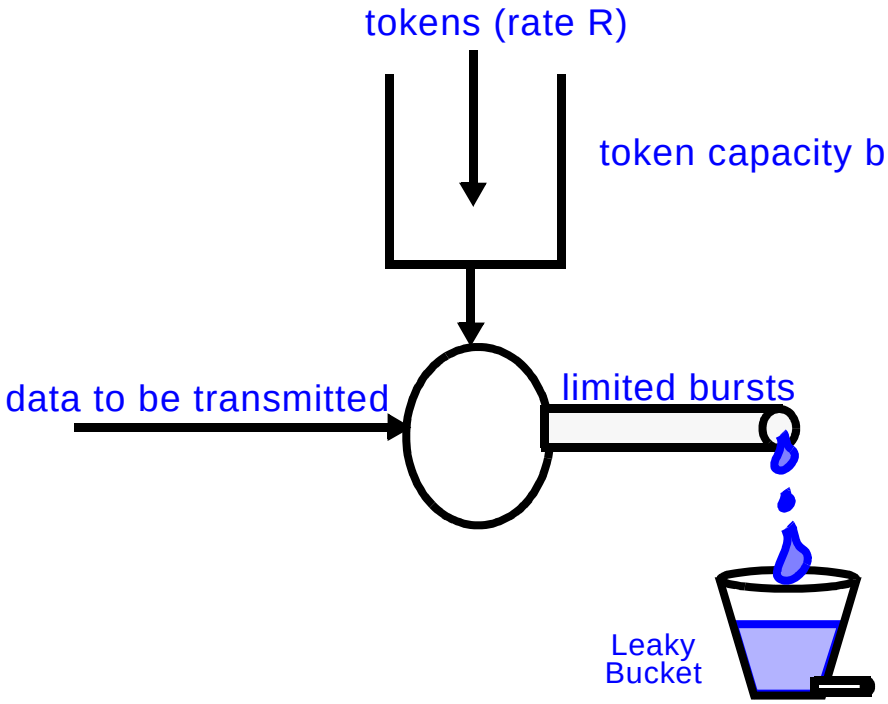
http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Token Bucket Algorithm**

- tokens drop into bucket with rate R, bucket capacity $\beta$
- packet (="burst") size determines no. of tokens (<=$\beta$) used
- effect: limited-size bursts allowed (within t, no more than $\beta$+R*t tokens)

**Token Bucket Algorithm with Leaky Bucket Rate Control**

- problem w/ token bucket: burst directly released into network
- therefore, put "burst" into another (leaky!) bucket
- --> burst drop out of leaky bucket smoothly




Scope

Contents

# Loss Handling

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

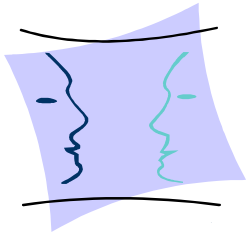
**Error Detection**

- **by means of redundancy / checks / analysis**

**Loss Handling**

**2 basic categories:**

- **(partial) Retransmission (problem: media stream stalls during retransmit)**
  - in-packet redundancy (cyclic redund. check CRC) for detection of error only
  - automatic repeat request ARQ: non (or neg.) acknowledged packets resent
  - three facets:
  - Go-back-N (error -> "reset" link to state before erroneous Xmit)
  - Selective Retransmission (error: buffer further packets until reXmit)
  - Using partially error-free streams (reXmit "reasonable" no. of packets)
- **Prevention (formerly considered "only viable solution" for multimedia)**
  - Forward Error Correction (FEC): in-packet redundancy for correction!
  - Priority Coding: --> mix of FEC "degrees" or of FEC and CRC
- **Multimedia-proof retransmission:**
  - Slack ARQ: buffer received packets before presentation!
  - buffer time long enough to support reXmit in hi-speed LANs
  - for, e.g., voice streams of type voice-pause-voice-....: re-buffer after pause



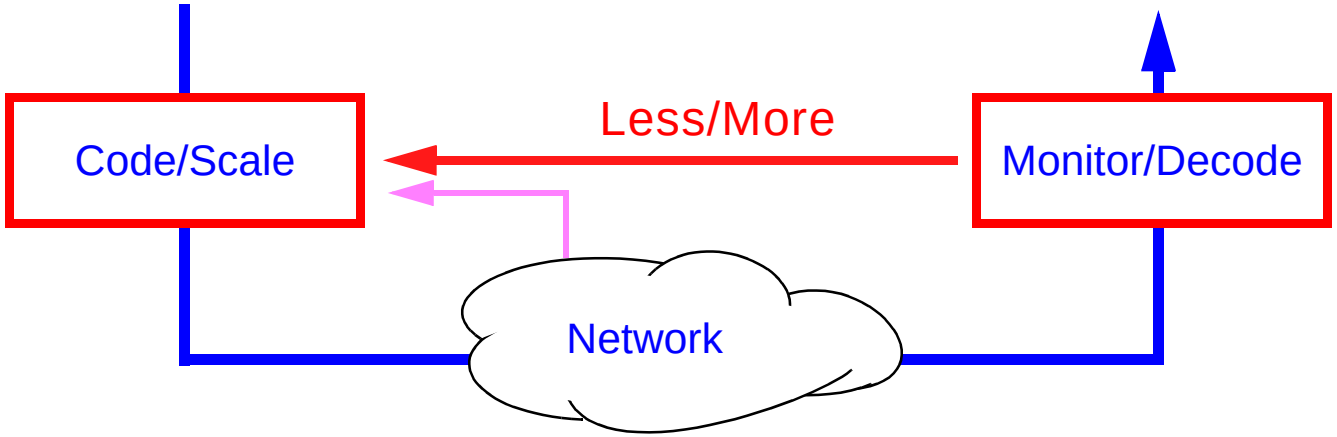
# Adaption - Feedback Control

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Monitor load of network and local end-system resources**

**If significant changes occur, take appropriate actions to reduce generated load**

- **Explicit communication – receiver informs sender to slow down**
- **Completely in network on a hop-by-hop basis**
- **By feedback from congested network nodes to sender**




**Variety of possible reactions**

- **e.g. Layered transmission**
- **Degredation, ...**

Scope

Contents

# 6. QoS Architectures

http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

## Examples (Communication Layer)

- **Heidelberg Transport System (HeiTS)**
  - uses ST-II (IPv5)
- **Internet Integrated Services**
  - use existing infrastructure, but deploy dedicated handling of Flows in Transfer System
  - Resource Reservation Protocol RSVP to support heterogenous needs
- **Differentiated Service**
  - Granularity based on TOS IP Header Field
  - Define Service Classes, Negotiate Service Level Agreements and ensure dedicated treatment of Flows that behave as described
- **IPv6**
  - QoS support as one design criteria
  - dedicated header fields to allow classification / dedicated treatment of flows

# 7. Conclusion

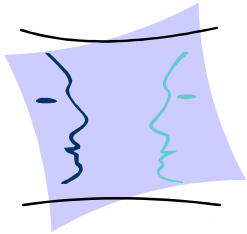
http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

**Realtime- and Multimedia Systems**

**Quality of Service - Definition and Concepts**

**Resources and Resource Management / QoS provisioning in 2 phases**

- **QoS specification, calculation and negotiation:**
  - Requirements of the application
  - Functions to calculate QoS guarantees
  - Guarantees returned by the system
  - Reservation of resource capacities
- **QoS enforcement:**
  - Scheduling of resource access
  - Monitoring and adequate actions (shaping, scaling ...)



Scope

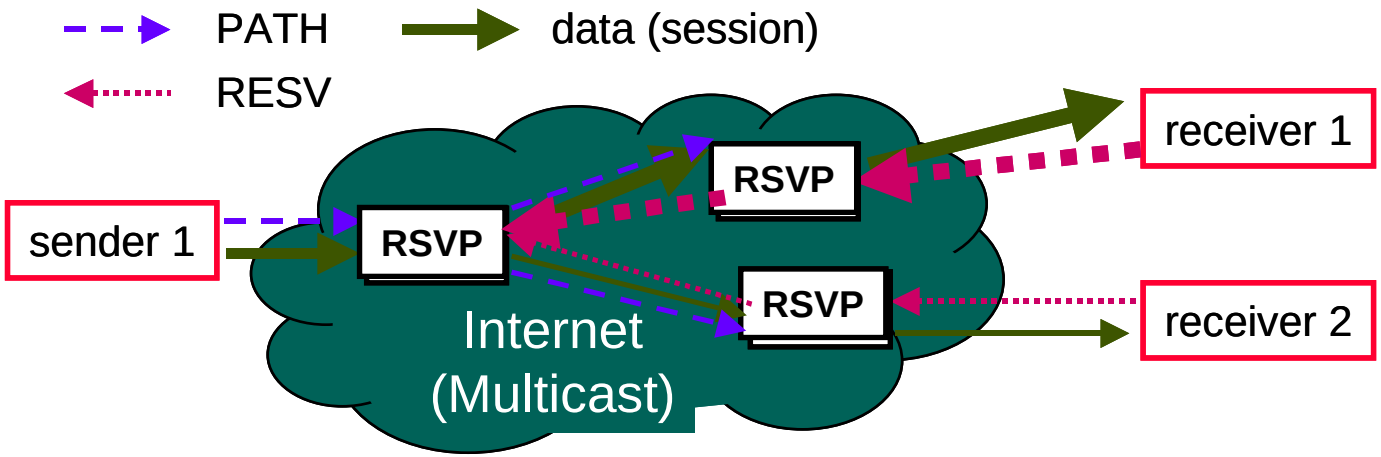
Contents

# Outlook: Communication Protocols

**Reservation / QoS initiated by: sender, receivers, netowork**

**example RSVP: receiver-initiated**

- **sender sets up paths (requested by receivers?)**
- **receivers periodically send reservation (RSVP) packets**
- **paths (~flows) may, e.g., represent video multicast**
- **receivers might not be able to present / receive in full resolution**
- **receivers may just drop video -> RSVP packets periodically**
-




http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser

Scope

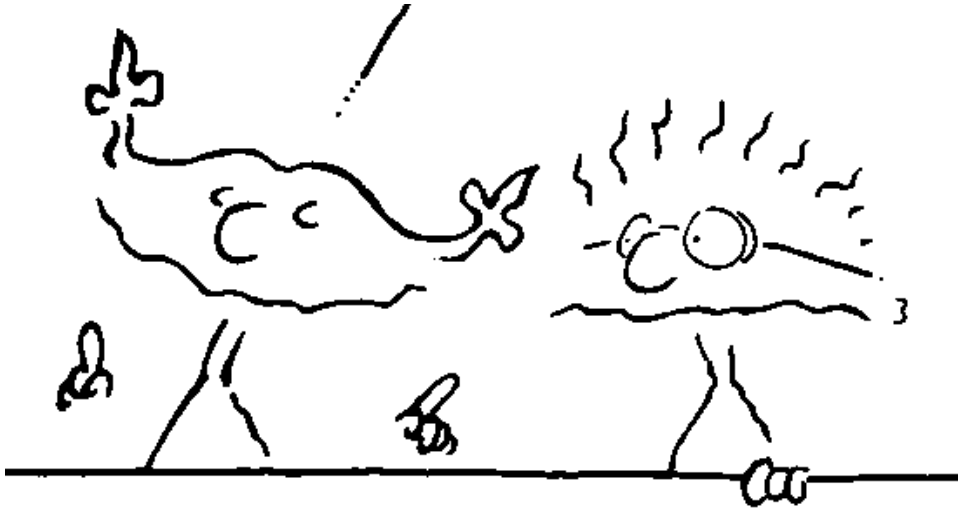
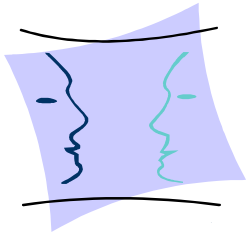
Contents

# QoS vs. Human Perception

**Rule-of-Thumb: Eye integrates, Ear differenciates (cf. chapter MM sync.)**

**--> tolerant visual perception 1: overlay separation**

**--> tolerant visual perception 2: completion**



http://www.kom.e-technik.tu-darmstadt.de
http://www.tk.informatik.tu-darmstadt.de
© R. Steinmetz, M. Mühlhäuser