

# Real Time Systems – SS2017

Prof. Dr. Karsten Weronek

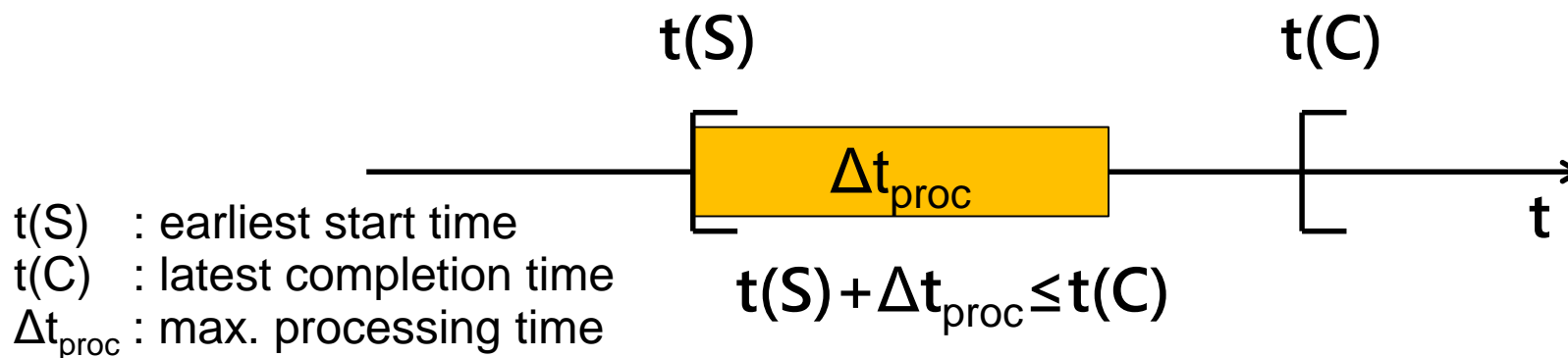
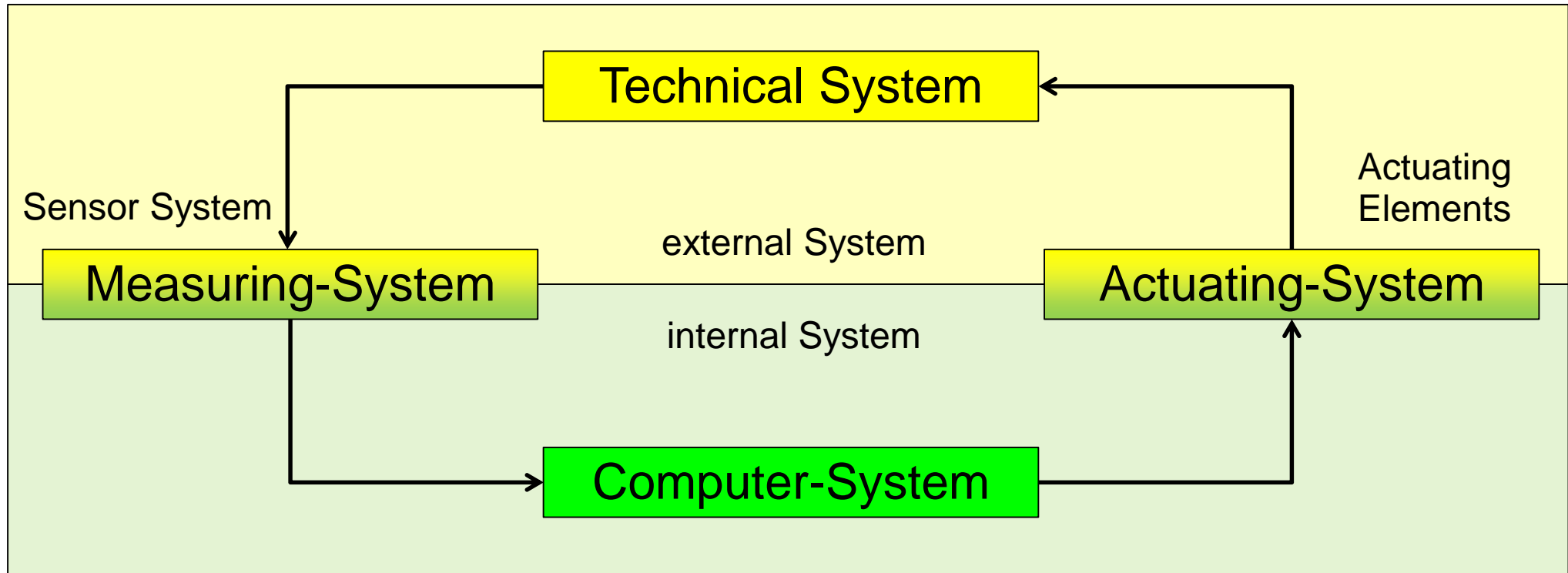
Faculty 2

Computer Science and Engineering

## Introduction

- What does Real-Time computing mean?

## Real-Time-System



- Transformation of input data is processed by the speed of arrival. ("xy is Real Time\; xy 2 MP3-encoder,rendering-engine, . . . )
- Hermann Kopetz:  
"A Real Time computer system is a computer system in which the correctness of the system behavior depends not only on the logical results of the computations, but also on the physical instant at which these results are produced." (Real Time Systems, 1999, S. 2)
- Jane Liu:  
". . . a Real Time system is required to complete its work and deliver its services on a timely basis."  
(Real Time Systems, 2000, S. 1)

## Procedures and methods for

- Development
- Implementation
- Test/Analysis

if (computer-)systems which are subject to a

„Real Time constraint“,

e.g. operational deadlines from event to system response.

## What is Real Time?

This means:

Correctness and Execution time of the results are guaranteed.

In non-Real Time Systems only the correctness of the result is guaranteed (and sometime not even that).

In other words:

An RTS will also guarantee that a certain deadline is met.

## There are three types of RTS:

### 1. Hard:

Missing a deadline is a total system failure  
(e.g. airbag in car, use case in aviation).

### 2. Soft:

The usefulness of a result degrades after its deadline,  
thereby degrading the system's quality of service.  
(e.g. warning systems, e.g. distance warner).

### 3. Firm

Infrequent deadline misses are tolerable but  
may degrade the system's quality of service.  
The usefulness of a result is zero after ist deadline.  
(e.g. ignition-point-optimizer for motor)

1. Consequence of missing deadline
2. Reliability and fault tolerance
3. Distribution: centralized or distributed RTS
4. Interactive or autonomic system
5. Hierarchical or independent systems
6. Time-driven or event-driven RTS
7. Cyclic or asynchronous scheduling

## RT-Systems often constitute Embedded Systems:

„Embedded computers are defined to be those where the computer is used as a component within a system: not as a computing engine in its own right. [...]“

(Jim Cooling: Software Engineering for RTS, 2003, S. 12, Pearson Education)

The computer is not a computer itself, but part of a system.

In other words embedded systems are central and part of the overall system.

(e.g. a missile control)

flappy but not bad:

"an embedded System is a Computer that doesn't look a computer"



- Complex interaction with the environment and other Embedded Systems
- Often operated autonomously
- High requirements on reliability
- Replace mechanical systems (e.g. regulator, stabilizer, controller)
- HW/SW-Code design: System on a chip (SoC)
- Usage of processor often not visible
- Normally working under RT constraints
- Repair cost are often higher than a new system

A very common definition of embedded System is

"is a computer system but doesn't look like a computer."