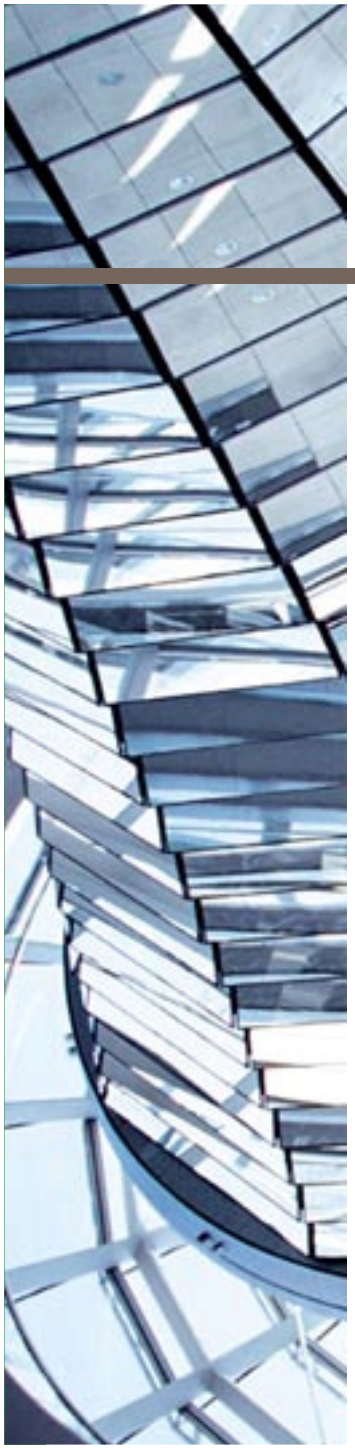


2. A reference architecture for business information systems

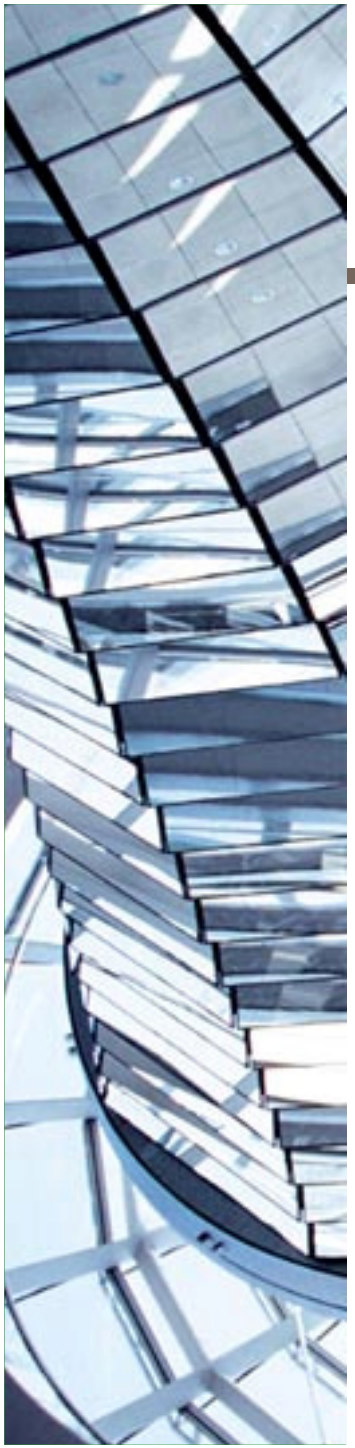
Reference Architectures and Patterns

Winter Semester 2008 / 2009
Prof. Dr. Bernhard Humm
Darmstadt University of Applied Sciences
Department of Computer Science

h_da
HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

The lecture in the context of the entire course

1. Introduction
2. A reference architecture for business information systems
3. Application kernel facade
4. Persistence and transaction
5. Authorization
6. Client architecture
7. Other reference architectures: SOA, BI, systems integration, …



Agenda

➔ **Architectural Viewpoints**

Software Categories

Components and Interfaces

Reference Architecture for Business Information Systems

Literature

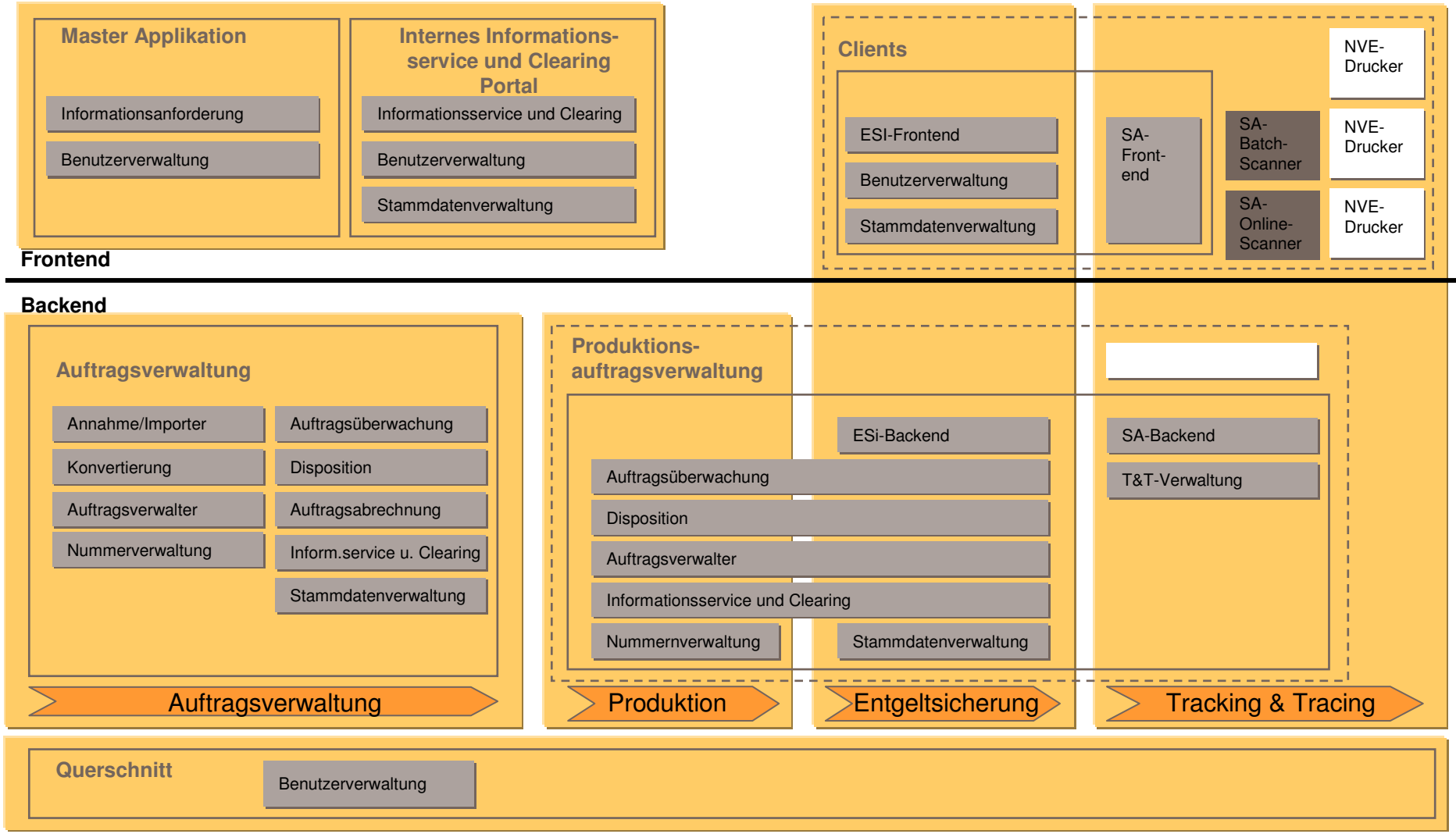
Architecture viewpoints separate concerns of different stakeholders

A Architecture (Business Application Architecture)	**T Architecture** (Technical Architecture)	**TI Architecture** (Technical Infrastructure)
■ Structure of an application from the business point of view ■ Independent of techical issues ■ Different for every application ■ Application entitites like Customer, Account, etc.	■ Glue between A an TI architecture ■ Specifies the virtual machine on which the A archictecture will be executed	■ Hardware (computers, networks, etc.) and systems software (operating systems, application servers, database servers, etc.) and programming languages used

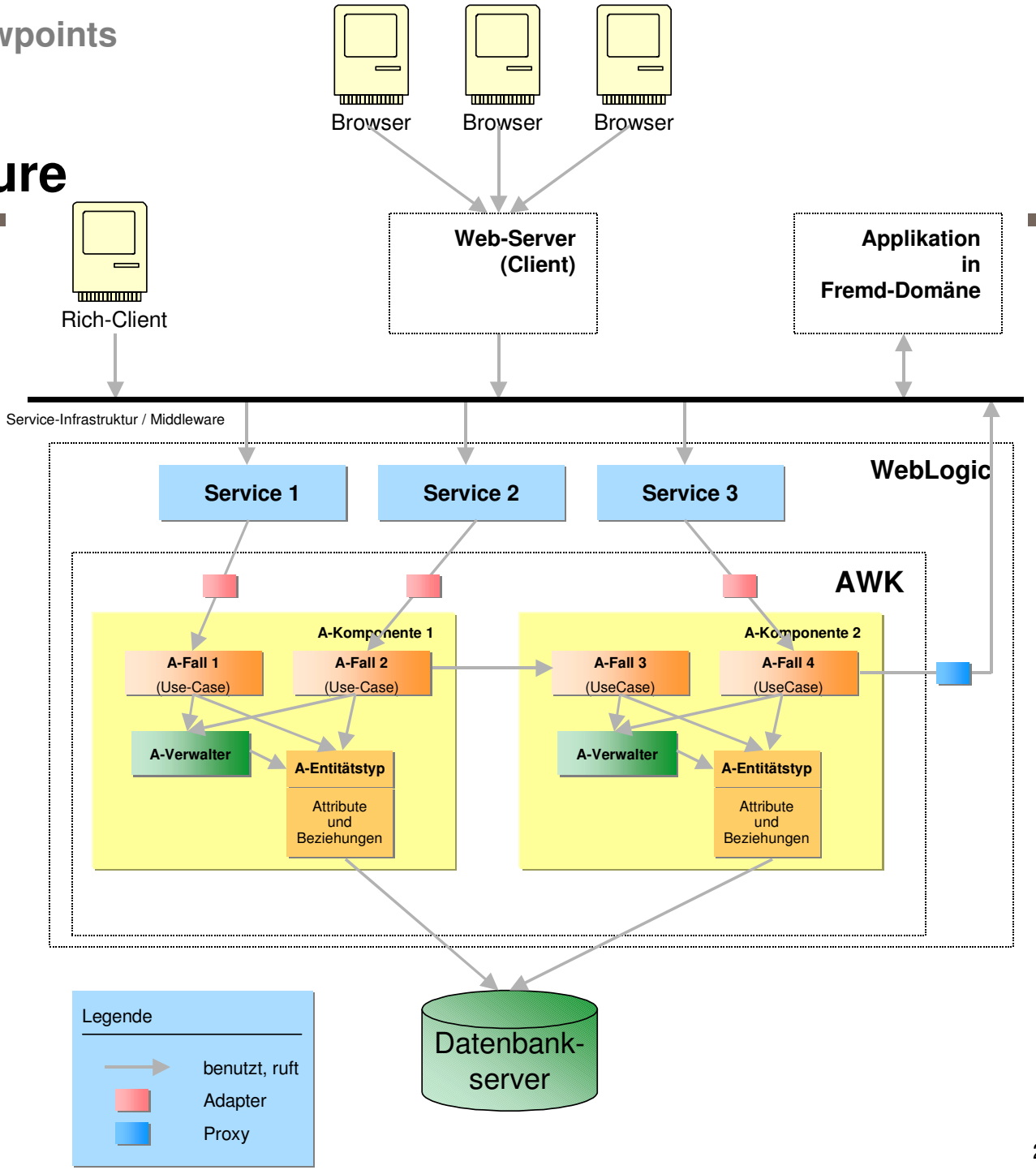
Example project in the domain of logistics

- Customer:
  large international logistics provider
- Project: order management
- Volume Release 1:
  > 30 person years
- Time:
  April, 2003 – January, 2004 (Release 2 currently being implemented)

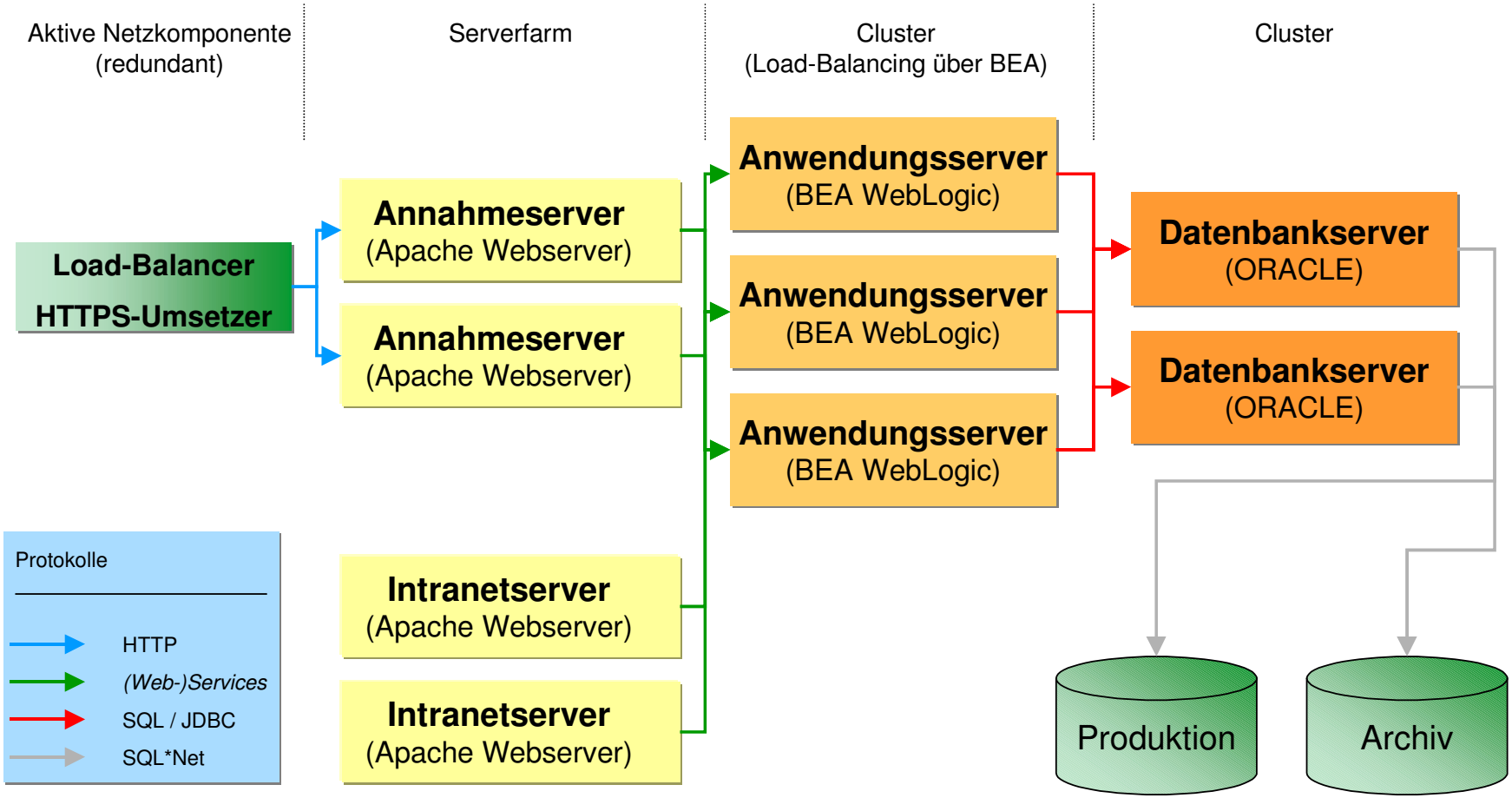
Example A Architecture (business application architecture)



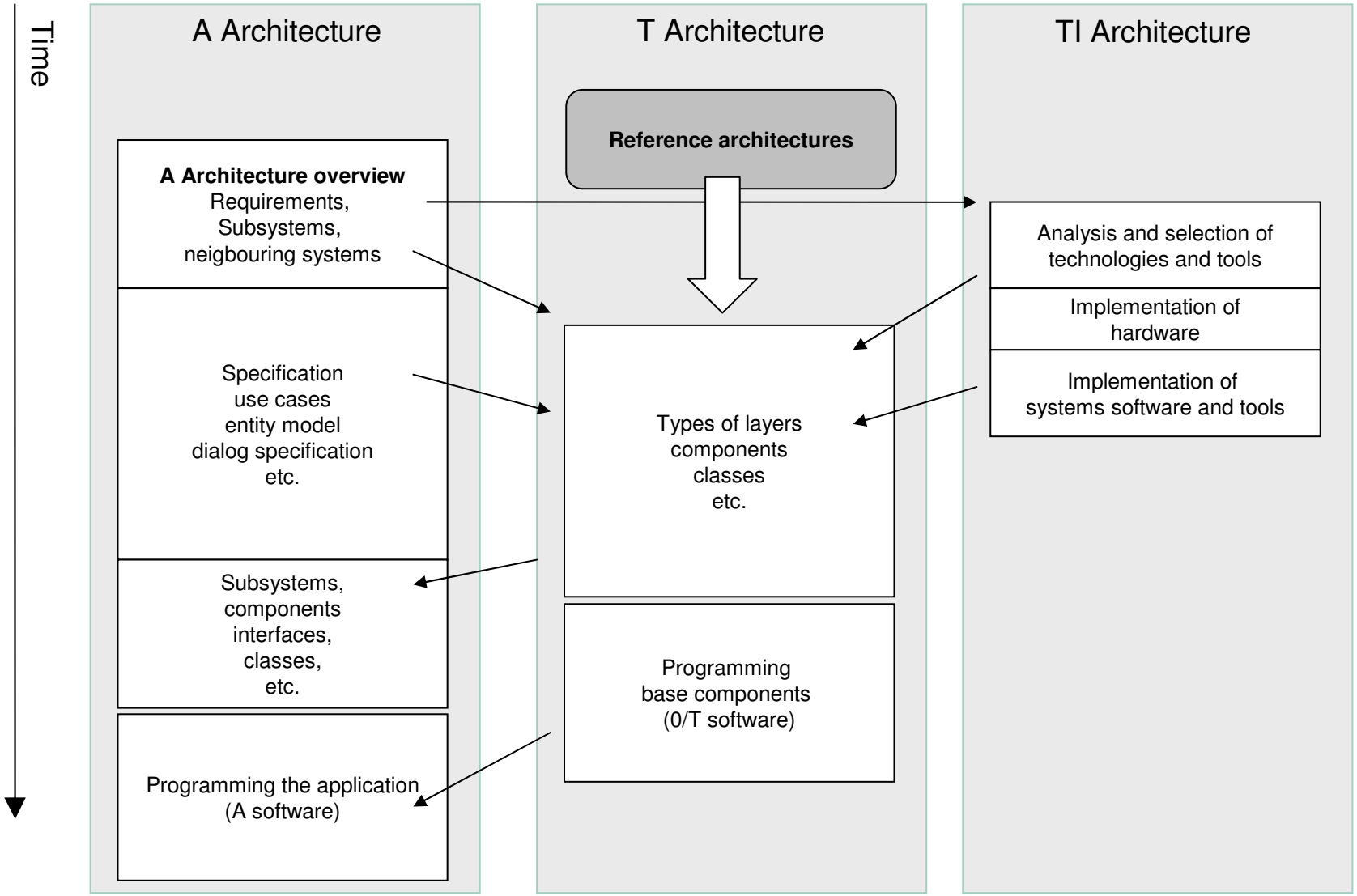
Example
T Architecture

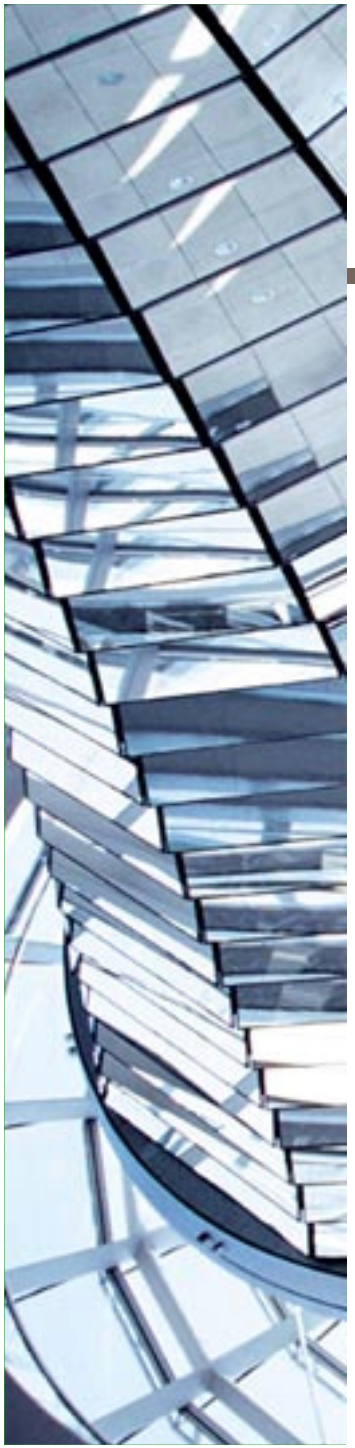


Example TI architecture (technical infrastructure)



Separating the architectural viewpoints eases parallelizing project tasks





Agenda

Architectural Viewpoints

➔ **Software Categories**

Components and Interfaces

Reference Architecture for Business Information Systems

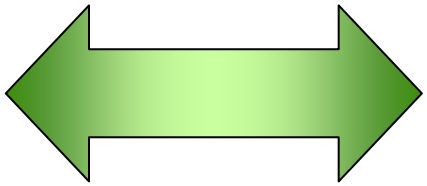
Literature

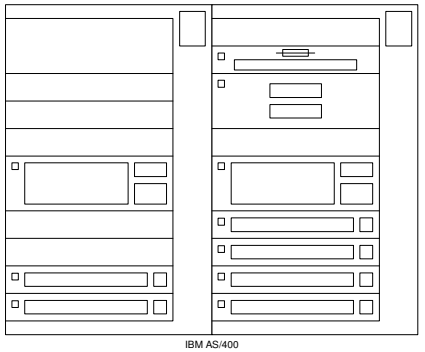
Idea behind software categories

Software that changes at a different pace
should be separated into different modules

Parnas, 1972

Example: Software for salary payment (human resources)





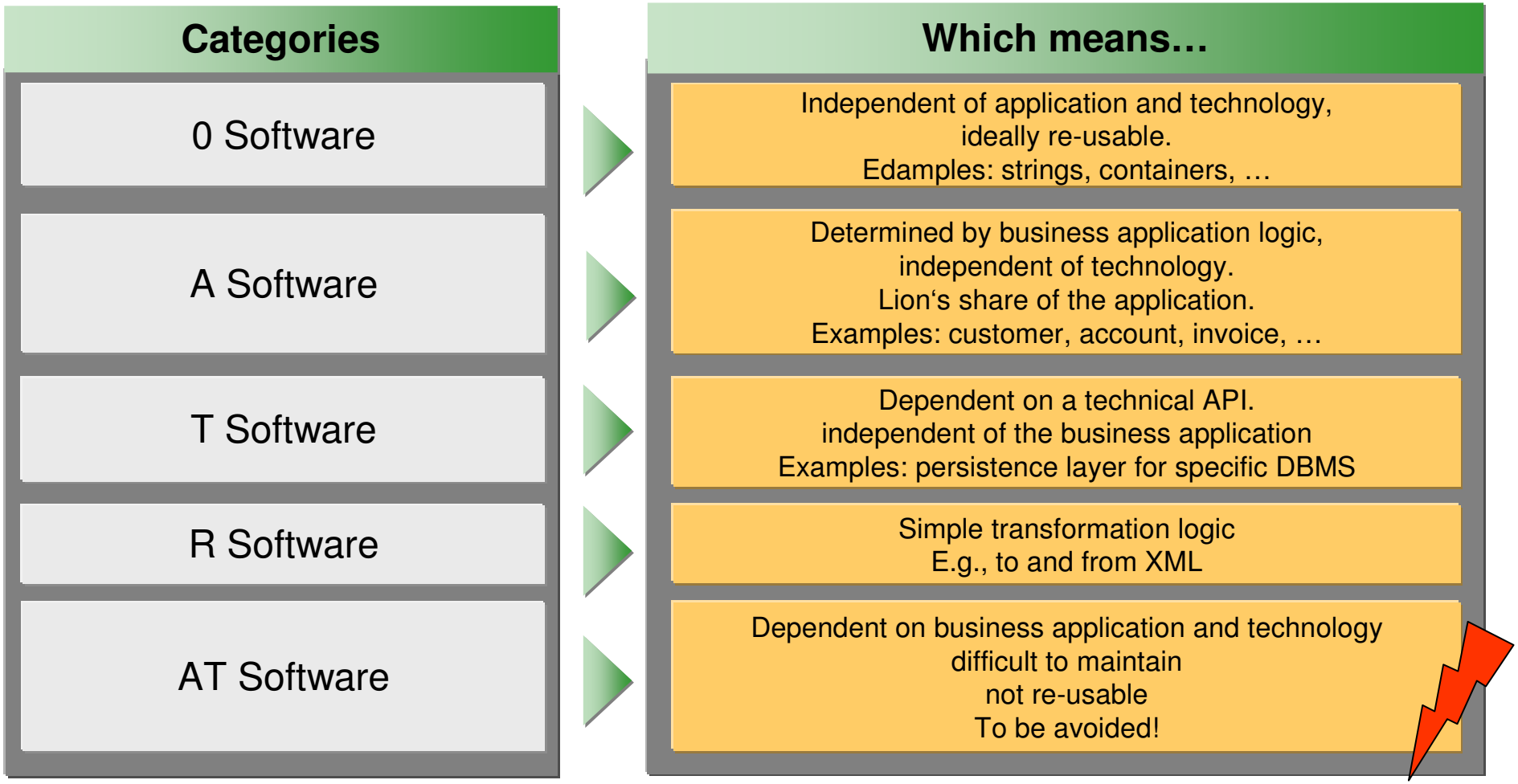
Business changes

- New legal requirements (taxes)
- New company payment policies

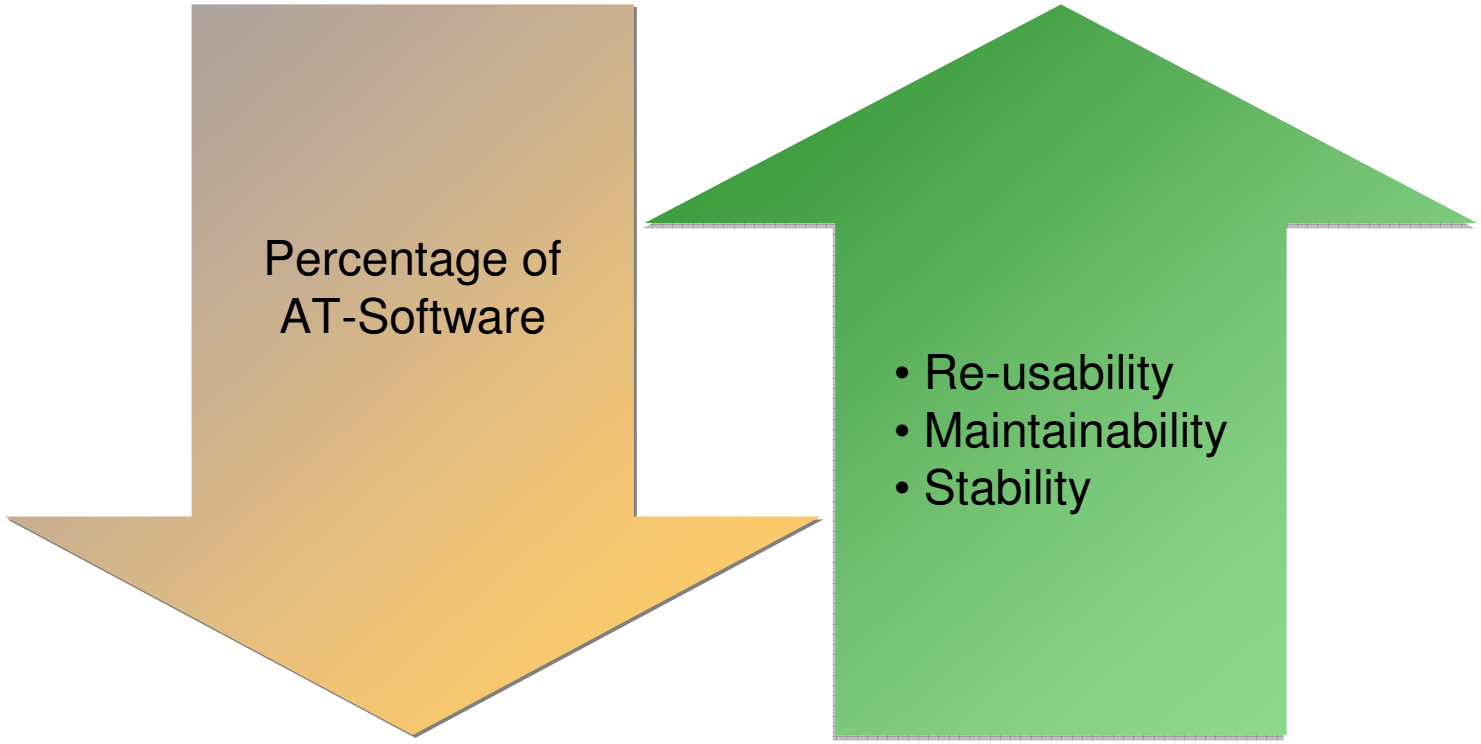
Technical changes

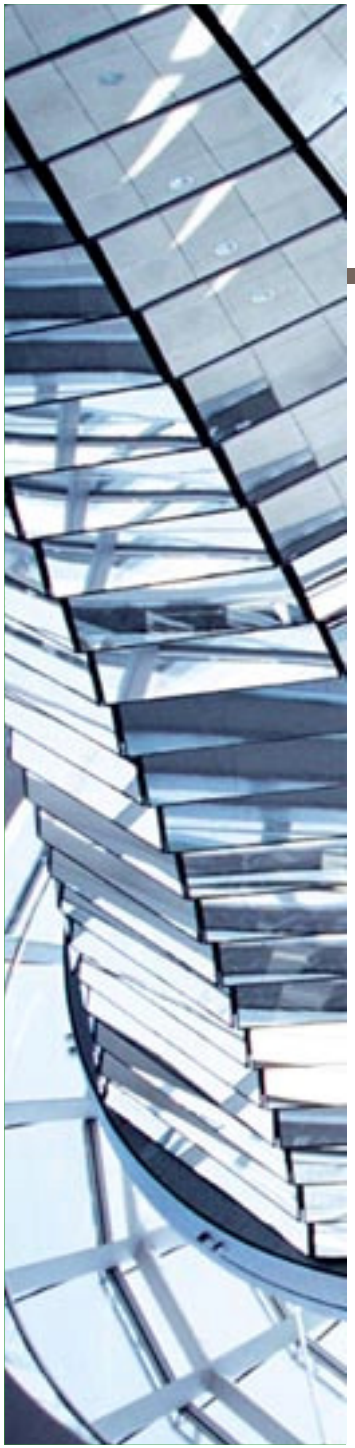
- New application server version
- New database version

Software categories



Software categories as a measure for software quality





Agenda

Architectural Viewpoints

Software Categories

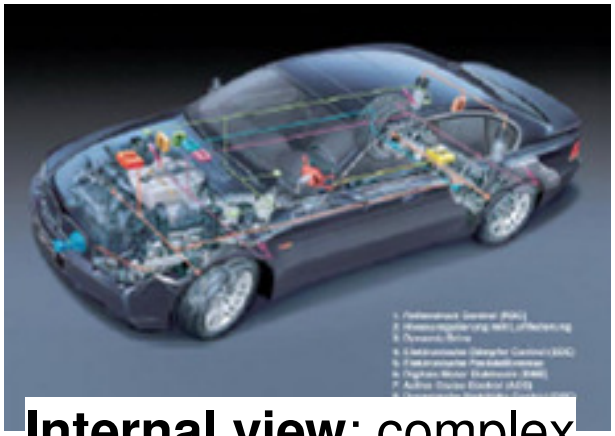
➔ **Components and Interfaces**

Reference Architecture for Business Information Systems

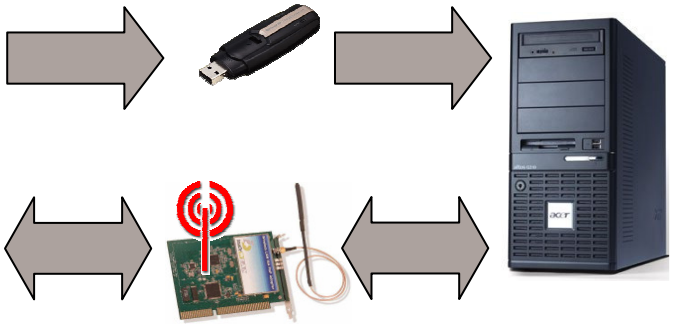
Literature

Component-oriented design of an automobile

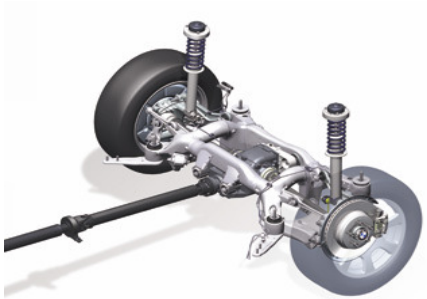
External view (interface) of the driver: as simple as possible



Internal view: complex

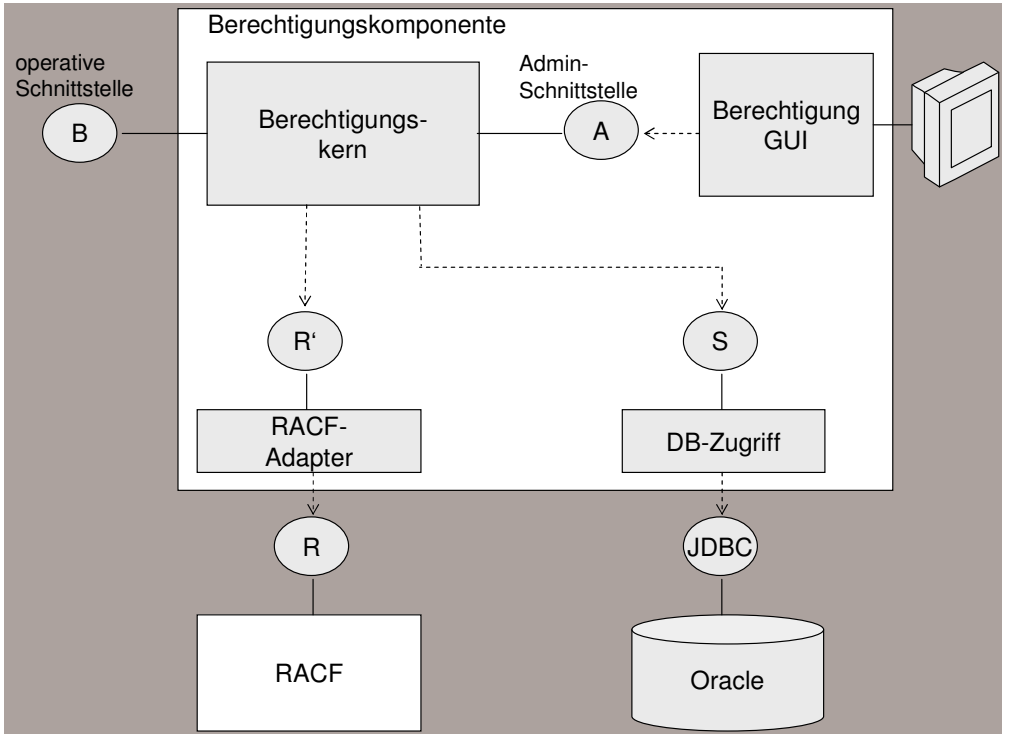


External view (interface) of workshop: as simple as possible



Component-oriented software design
Example: Authorization component

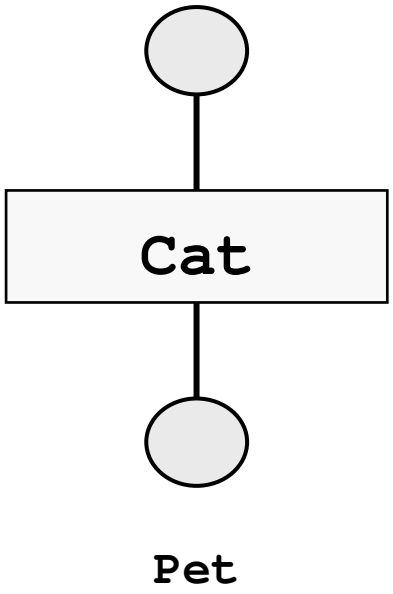
External view: operational interface of application programmer (as simple as possible) operative

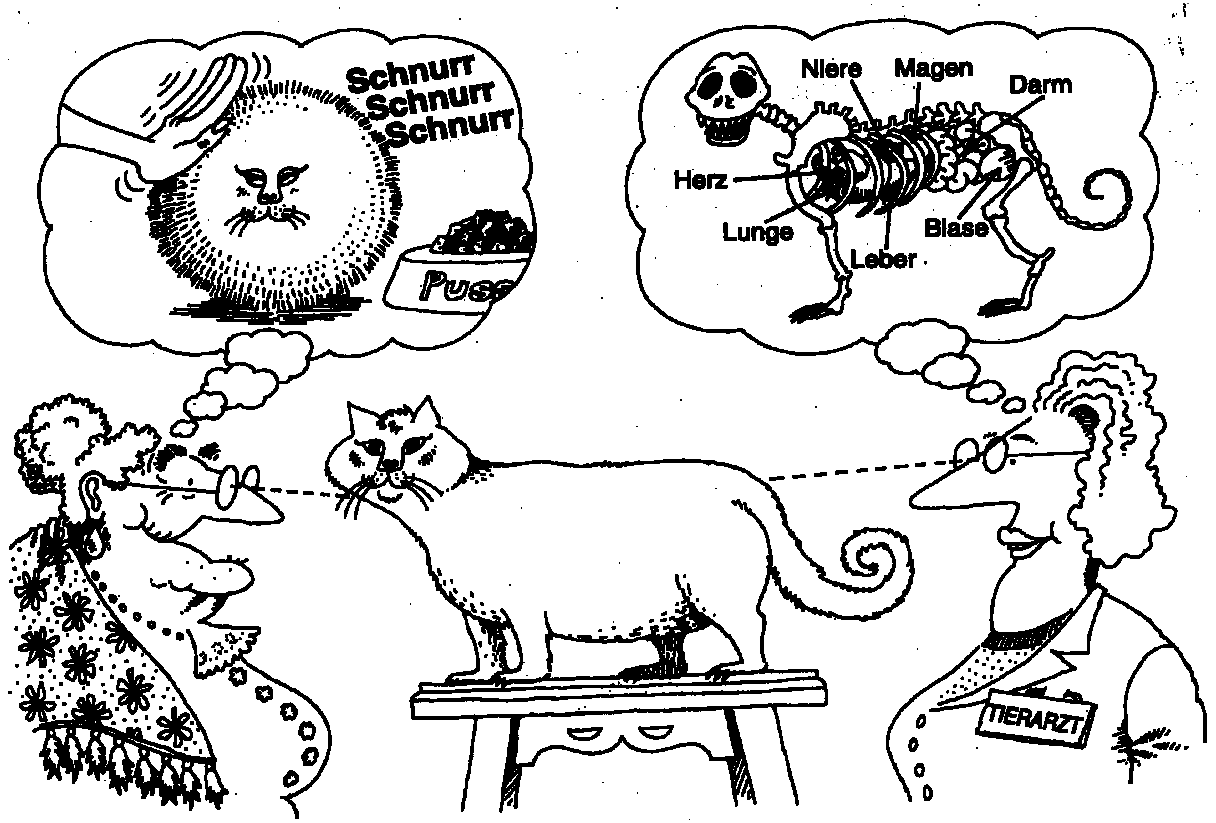


External view: administrative interface (as simple as possible)

Internal view: complex

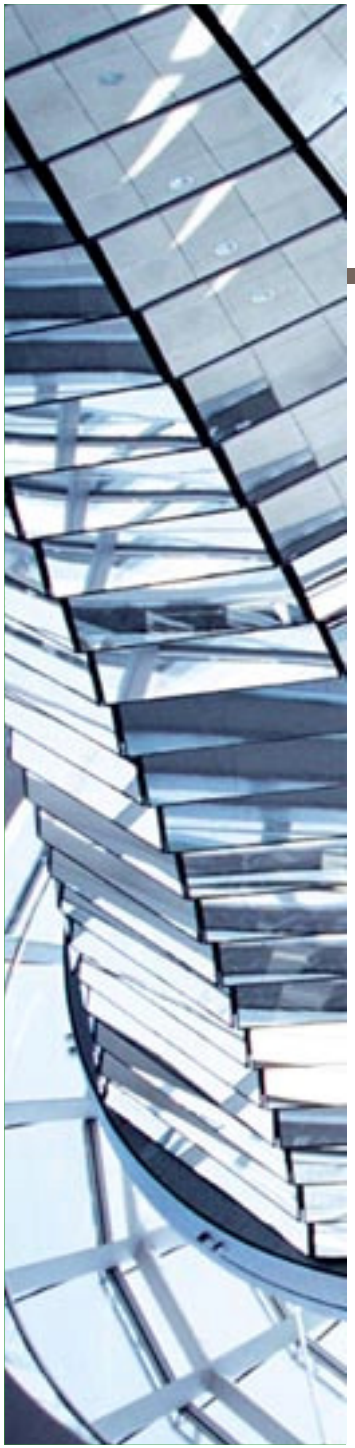
Interfaces present abilites of a component to different user groups






```
class Cat implements Patient, Pet {…}
```

Source: Roger King



Agenda

Architectural Viewpoints

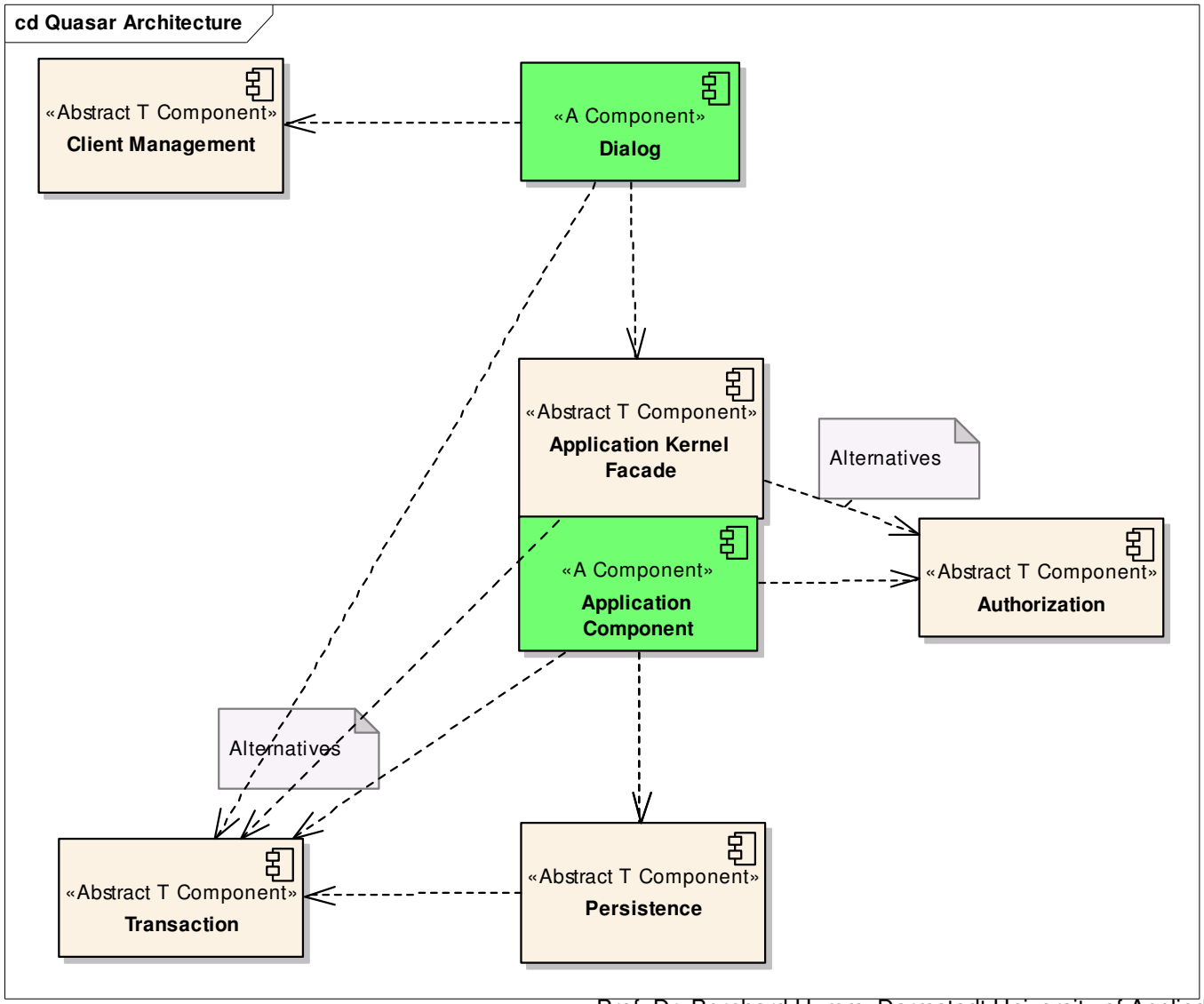
Software Categories

Components and Interfaces

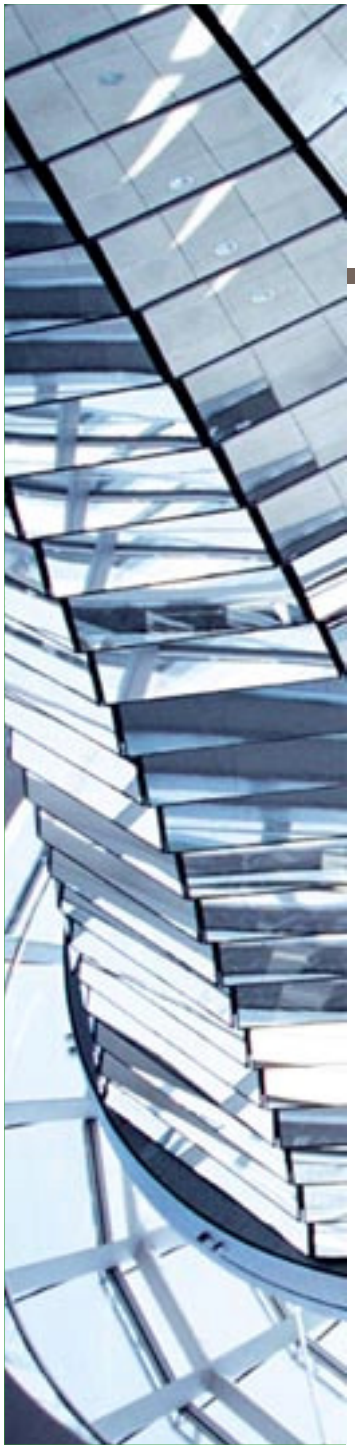
➔ **Reference Architecture for Business Information Systems**

Literature

Reference architecture for business information systems Quasar (Quality Software Architecture)



- Separation of A Software and T Software on the component level
- Identification of the main technical components
- Identification of dependencies



Agenda

Architectural Viewpoints

Software Categories

Components and Interfaces

Reference Architecture for Business Information Systems

➔ **Literature**

References for this lecture

Sections of „Teil 1“ dealing with today‘s lecture:

- 1.2 Kategorien (Blutgruppen)
- 1.3 Schnittstellen und Komponenten
- 2. Architekturbegriffe