Systems Development

Lecture 1: Introduction and System Choice

Contents

- Introduction to the course
- Object-oriented analysis and design
- System Choice
- Varying system definition elements
- Challenges in this activity

Contents

▶ Introduction to the course

- Lecturer
- Course overview

▶ Object-oriented analysis and design

▶ System Choice

▶ Varying system definition elements

▶ Challenges in this activity

Lecturer

▶ MSc in Computer Science, Aalborg University, 1984
Master thesis on Methods for Analysis
In collaboration with Mentor informatik

▶ PhD in Informatics, University of Oslo, 1989
Thesis on Work practices for analysis and design in system development

▶ Teacher and researcher, Department of Computer Science, Aalborg University, 1988-

▶ Guest researcher at SUNY Binghamton (1991) and University of Auckland (2002-2003)

▶ Research and teaching in:

- System development and system development methods
- Object-oriented methods – OOA&D
- Prototyping
- Usability evaluation
- Interaction design

OBJECT ORIENTED ANALYSIS & DESIGN

Lars Mathiassen
Andreas Munk-Madsen
Peter Axel Nielsen
Jan Stage
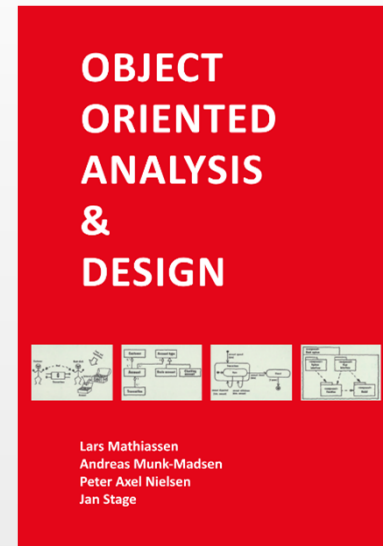
Teaching Assistants

▶ Four teaching assistants

- Jens Nicolaj Andersen
- Mathias Kirkegaard Rasmussen
- Simon Mathiasen
- Theodor Risager

▶ They will

- Assist during exercise hours
- Comment on submitted assignments

Course Overview

▶ 5 ECTS – 12 lectures

▶ **Knowledge**
The student should gain knowledge of the following theories and methods:

- Object-oriented modelling in analysis and design: modelling of context (application domain and problem domain), object-oriented concepts: class, object, event, structure, function, use patterns, component, component architecture, UML: class diagram, state chart diagram, sequence diagram, diagram for use patterns
- Modelling with patterns: patterns for modelling application and problem domains, patterns for composing components, specifically the patterns for analysis: object-descriptor, hierarchy, stepwise-role, materials, procedure, specifically the patterns for design: collection, layered, observer, client-server, model-view-controller
- System development methods: waterfall method and model-driven development, iterative method and prototype-driven development, activities in systems development and relations between activities
- Systems practices: the relation between methodology and practice, strengths and weaknesses of model-driven and prototype-driven development

▶ **Skills**

- be able to explain accurately, using the concepts and modelling language of the discipline
- be able to model the requirements to a system, its context and all its various parts (model, features and interfaces)
- be able to model a system design at component level and describe relations between components

▶ **Competences**

- be able to apply concepts, patterns and modelling language to describe a specific system that solves a well-defined task

Form

▸ Individual reading, exercises and quizzes before lectures

▸ Lectures providing overview of the topic

▸ Small exercises and discussion during lectures

▸ Group exercises after lectures (next slide)

▸ Submitted assignments (you will get feedback)

▸ Exam:
- Written exam
- 4 hours
- January
- Open book
- Computers and mobile phones are NOT permitted
- External censor
- Graded by the 7 point scale

▸ Examples of written exams: exam set + solution 2 weeks later

Group Exercises

Analysis and design of systems in all exercises

▶ Your specific group project

- Focus on this
- Keep up with the pace of the course (defer details)

▶ General exercises

- An extra offer
- Especially until you have chosen your project

Workload

▶ 5 ECTS ≈ 137,5 hours

▶ Distribution of effort

• Lectures	24 h
• Preparation (reading and individual exercises)	48 h
• Group exercises	24 h
• Submitted assignment	12 h
• Exam and preparation for exam	29,5 h

Literature

Primary literature:

▶ Mathiassen, L., Munk-Madsen, A., Nielsen, P. A. and Stage, J.
  (2018) *Object-Oriented Analysis & Design*.
  Metodica. (OOA&D)

Secondary literature:

▶ Bo Dahlbom and Lars Mathiassen. 1993. *Computers in Context:
  The Philosophy and Practice of Systems Design*. Blackwell
  Publishers, Inc., Cambridge, MA, USA.

▶ A number of papers and excerpts from textbooks

Contents

▸ Introduction to the course

▸ Object-oriented analysis and design

- Activities and method
- Basic concepts
- Principles
- Method

▸ System Choice

▸ Varying system definition elements

▸ Challenges in this activity

System Development: Activities and Method

▶ Analysis:

- Understand a system, its context, and the conditions for its implementation
- To determine system requirements

▶ Design:

- Produce a system design without significant uncertainties

▶ Implementation:

- Realize a design on a technical platform

▶ Method:

- Guidelines for carrying out system development activities

▶ Example

- Guidelines for work processes (OOA&D)
- Guidelines for documentation (UML)

▶ A method can be applied under different approaches

- Waterfall
- Iterative

What is Object-Orientation?

▸ Object:
  • An entity with:
    identity, state, and behavior
▸ An object belongs to a class

▸ Class:
  • A description of a collection of objects sharing: structure, behavioral pattern, and attributes
▸ Each class contains a set of objects
  • We refer to them as the objects of the class
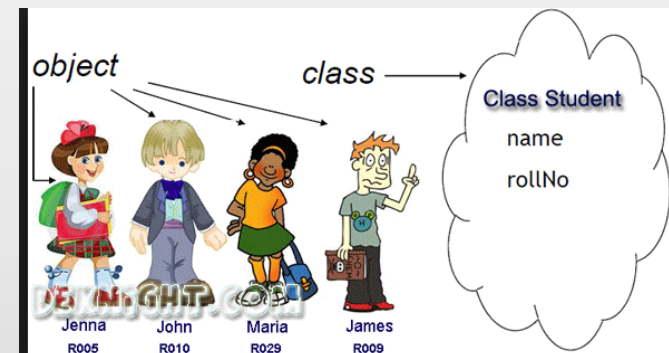
Example: Object and Class

▶ Object:
- identity: myChair
- state: by dining table, free
- behaviour: bought, moved to, … , sat down on, got up from, …, moved to, …, sold
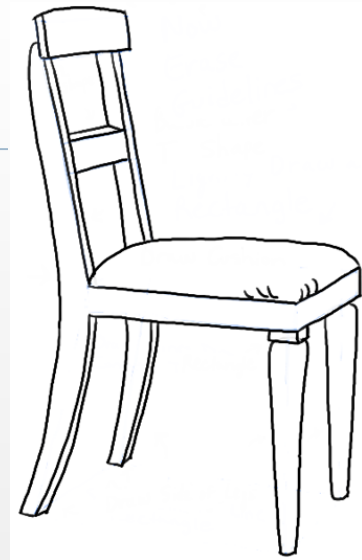
▶ Class:
- structure: has an owner
- attributes: position, vacant
- behavioural pattern: buy + {move | sit down on + get up from}* + sell

Example: Warehouse

▶ A large collection of articles stored in separate positions

▶ An article can be entered into, moved within and removed from the warehouse

▶ Describe an object and the relevant class

Example: Gravel Pit

- In the gravel pit, dirt is sorted into sand, gravel and rocks
- The sorted materials are stored in piles
- <u>Describe an object and the relevant class</u>

Objects in Analysis and Design

▶ Analysis

- Phenomena outside the computer system
- Identity: identifies an object
- State: properties of an object and its relation to other objects
- Behavior: the events an object has performed or suffered

▶ Design (and implementation)

- Phenomena inside the computer system
- Identity: gets access to an object
- State: attribute values and values of relations
- Behavior: the operations an object can perform on request and offers to other objects

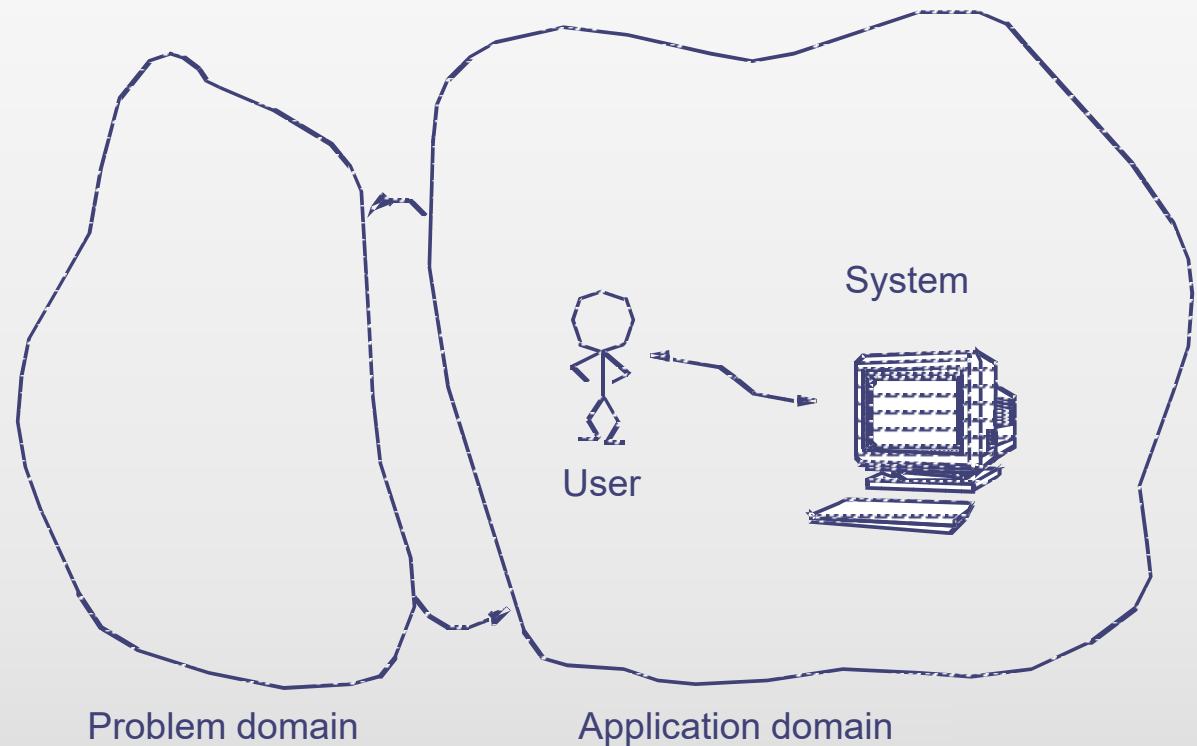▶ More on this column when we get to design

Model the Context

Focus on an IT system
and its context

Problem domain:
That part of a context
that is administrated,
monitored, or controlled
by a system

Application domain:
The organization that
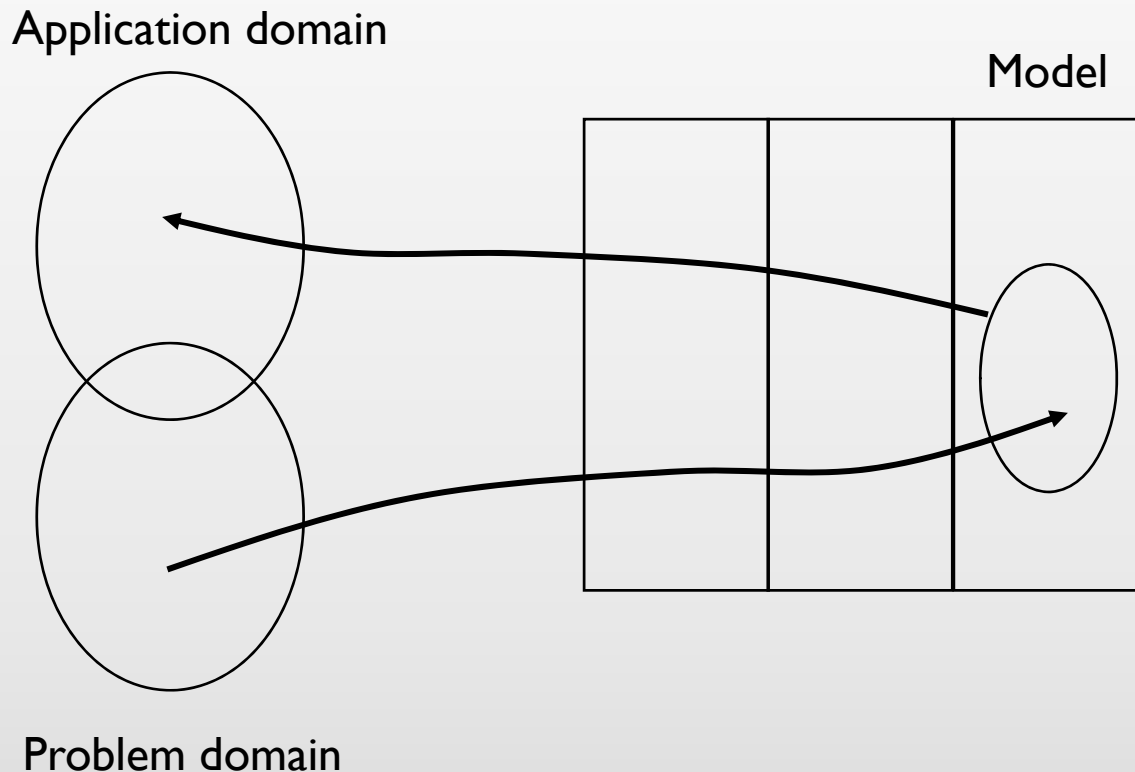administrates, monitors,
or controls a problem
domain

System

User

Problem domain

Application domain

A Model of the Problem Domain

Model:

An updated representation of the state in the problem domain

User:

Is in the application domain and gets information about the problem domain mediated through the model.
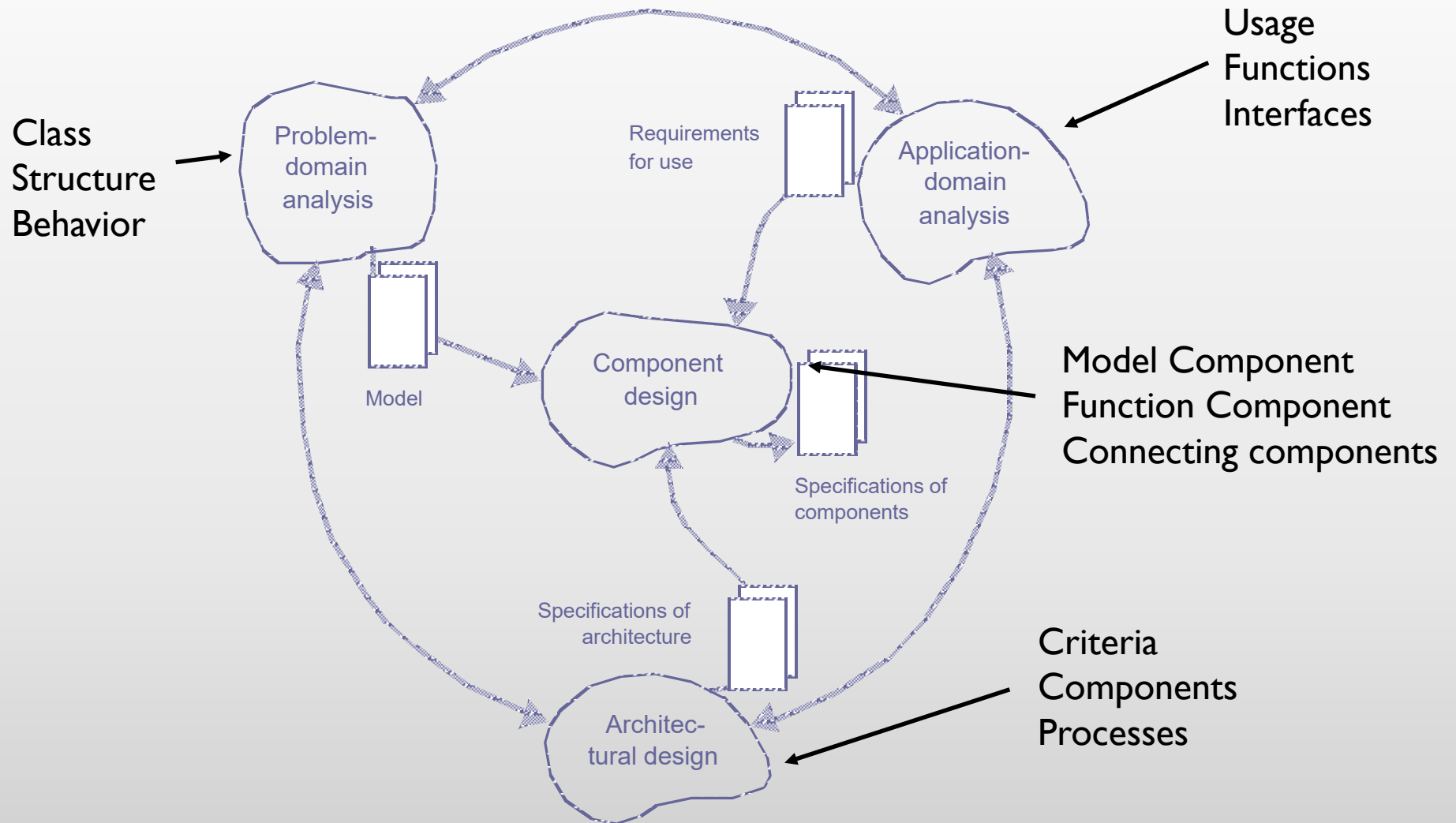
Application domain

Model

Problem domain

Context and Interplay

▶ <u>For a system controlling the temperature in a greenhouse, describe its problem domain, application domain and the interplay based on the model</u>

▶ A solution:

- PD: the greenhouse and its temperature

- AD: The gardener

- A temperature sensor connects the model to the problem domain

- Two possible control mechanisms

  ▪ A display outside the greenhouse and manual control of the windows

  ▪ An automatic control of the windows based on the temperature

Sprinkler

Window motor

Temperature and humidity sensor

Heater

The Method as a Whole

Class
Structure
Behavior

Problem-domain analysis

Requirements for use

Application-domain analysis

Usage
Functions
Interfaces

Model

Component design

Specifications of components

Model Component
Function Component
Connecting components

Specifications of architecture

Architec-tural design

Criteria
Components
Processes

The OOA&D Method

| Purpose	• To determine system requirements. • To produce a system design without significant uncertainties. • To understand a system, its context, and the conditions for its implementation.
Concepts	• Object: En entity with identity, state, and behavior. • Class: A description of a collection of objects sharing structure, behavioral pattern, and attributes. • Problem domain: That part of a context that is administrated, monitored, or controlled by a system. • Application domain: The organization that administrates, monitors, or controls a problem domain. • System: A collection of components that implements modeling requirement functions, and interfaces
Principles	• Model the context. • Emphasize the architecture. • Reuse patterns. • Tailor the method to suit specific projects.
Results	• An analysis document and a design document.

Contents

▶ Introduction to the course

▶ Object-oriented analysis and design

▶ System Choice

- Purpose, activities and result
- Situation and rich pictures
- Create new ideas
- Define systems

▶ Varying system definition elements

▶ Challenges in this activity

The System's Context

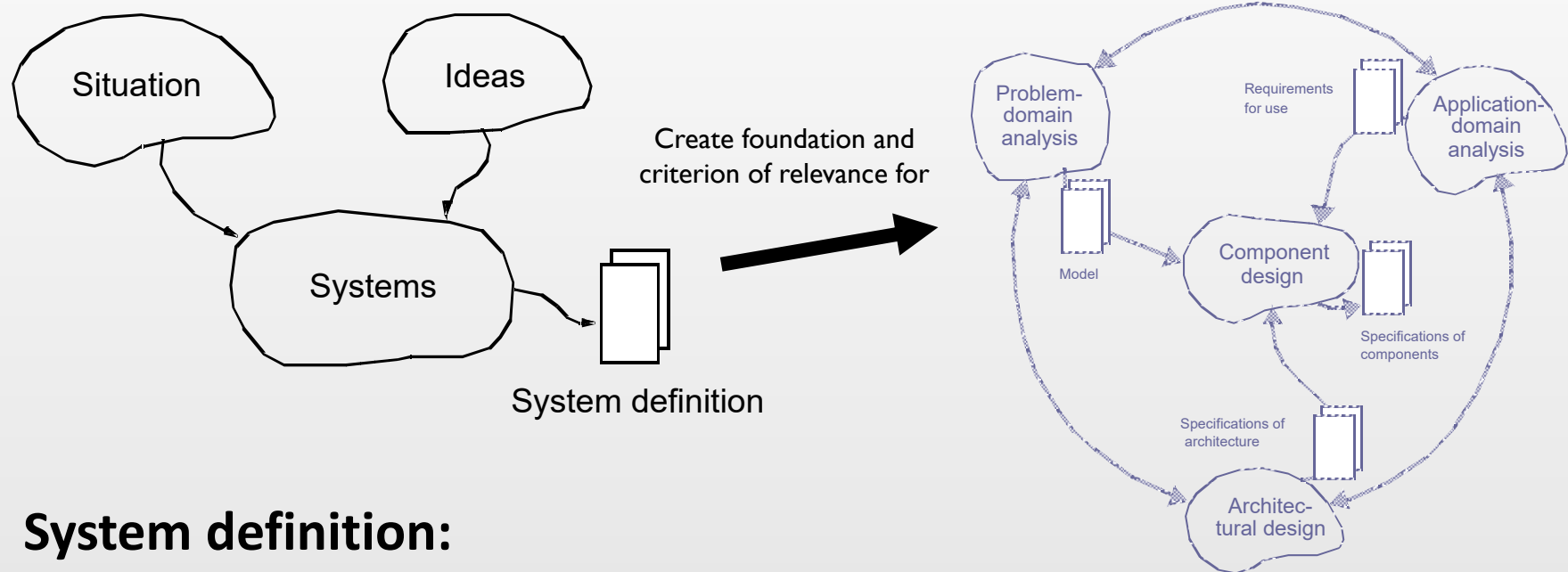▸ The purpose of the System Choice activity is to define the system in its context = AD+PD

Problem domain   Application domain
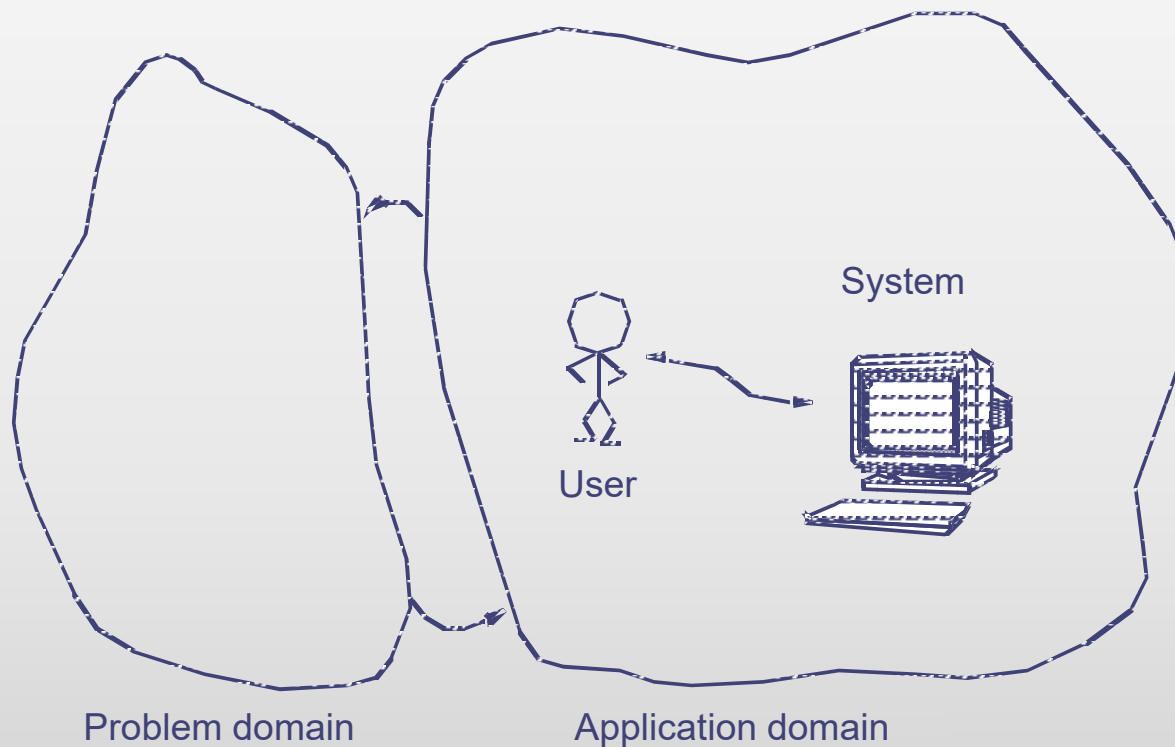
System Choice: Result

F   The system can register a new customer with credit card information, compose a meal with a number of orders for food shops and bars, select food or drink items in each order of a meal, make payment of a meal, and register the table where a customer wants his/her orders delivered.

A   A system provided as an app, which is used by customers who want to order food and drinks at S-Food's street food market. The customers come to S-Food at their own initiative, and their only relation to S-Food is that they download and use their app to order and pay for food and drinks.

C   The app will be developed by S-Food's own IT department in cooperation with S-Food's sales department, the food shops and bars, and customers that will be selected to represent the whole customer segment. It may be necessary to resolve conflicting requirements between these different groups. The app will be used by users with very different levels of IT skills.

T   The app is running on each customer's smartphone. It communicates through a wireless network with a server that registers what the individual customer has ordered and paid. On the smartphone, there is always a copy of all meals from the current day, so they are available if the wireless network should fail. The app includes a QR code reader.

O   Customer, Food Shop, Bar, Order, Meal, Food item, Drink item.

R   The system is primarily an administrative tool that is responsible for registering all customers and their orders, and facilitating secure payment of these with the customers' credit cards. Secondarily, it is a communication medium that customers use to request delivery of orders from the food shops and bars.

System Choice: Purpose and Activities

Situation

Ideas

Systems

System definition

Create foundation and criterion of relevance for

Problem-domain analysis

Requirements for use

Application-domain analysis

Model

Component design

Specifications of components

Specifications of architecture

Architectural design

System definition:

A concise description of a computerized system expressed in natural language

Situation

Situation

Ideas

Systems

System definition
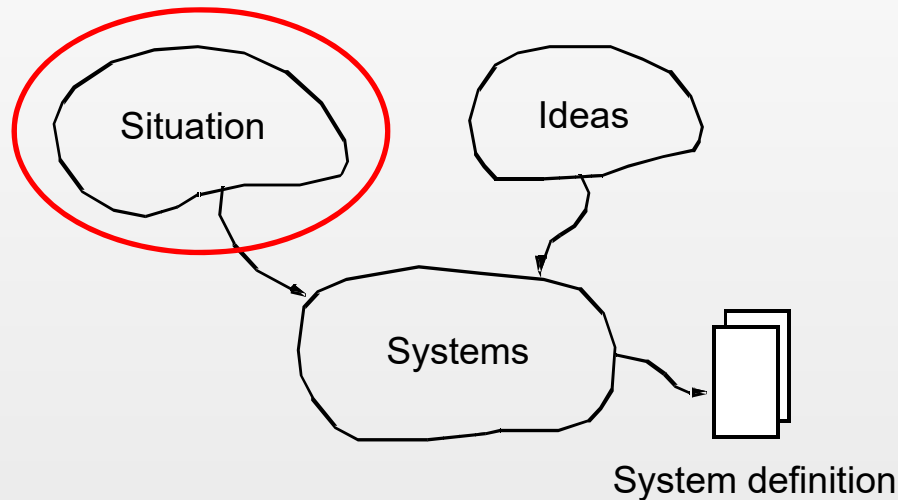
Describe the situation:

‣ The context of the system (AD and PD)

Maybe also:

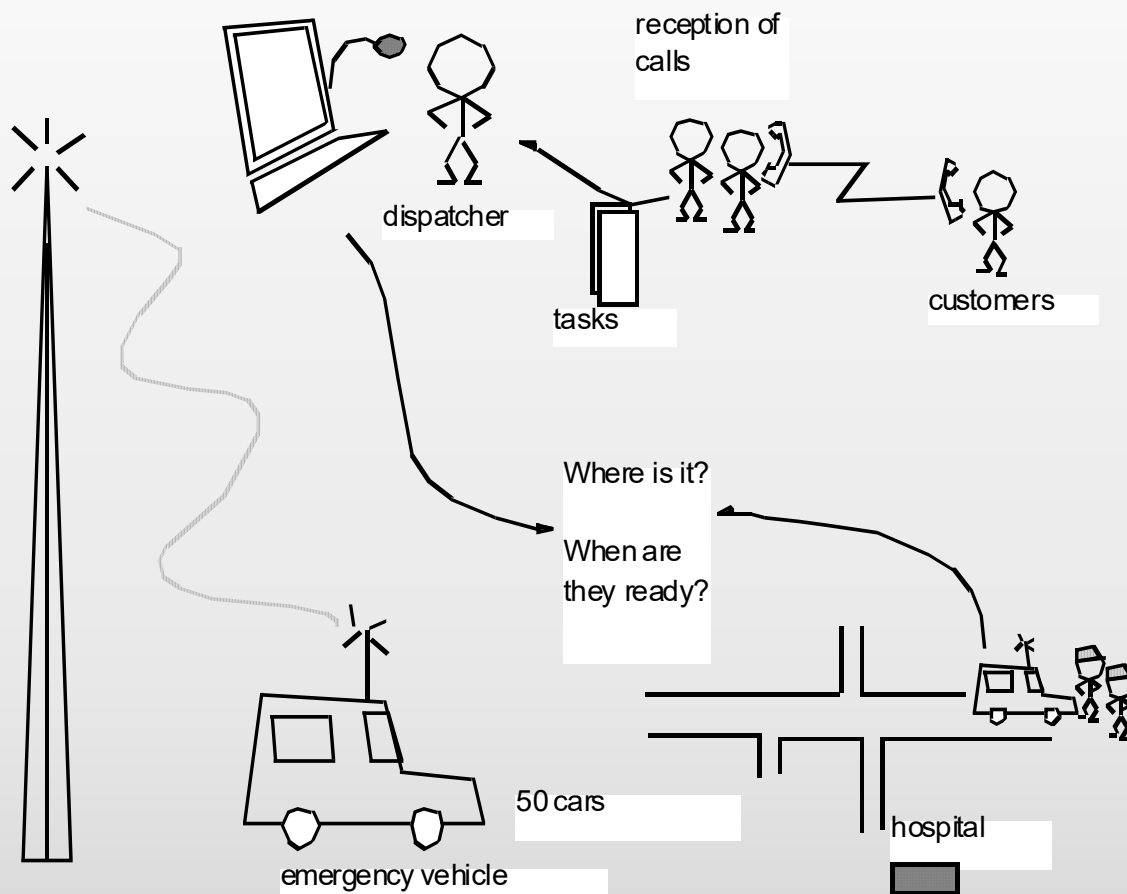‣ The system

‣ The conditions for development of the system

We describe this on an overall level in a rich picture

Example: Rescue Station (Chapter 21)

Rich Picture

reception of calls

dispatcher

tasks

customers

Where is it?

When are they ready?

50 cars

emergency vehicle

hospital

- ▸ The situation is described by a rich picture
- ▸ Focus:
  - on actors and the role of the system
- ▸ Entities:
  - People, roles and tasks, locations
- ▸ Processes such as:
  - work, production, information processing, exchange, planning, control, development, organizational change
- ▸ Structures such as:
  - production, use, application, communication, deal, ownership, membership, relationship, power

Ideas

Situation

Ideas

Systems

System definition
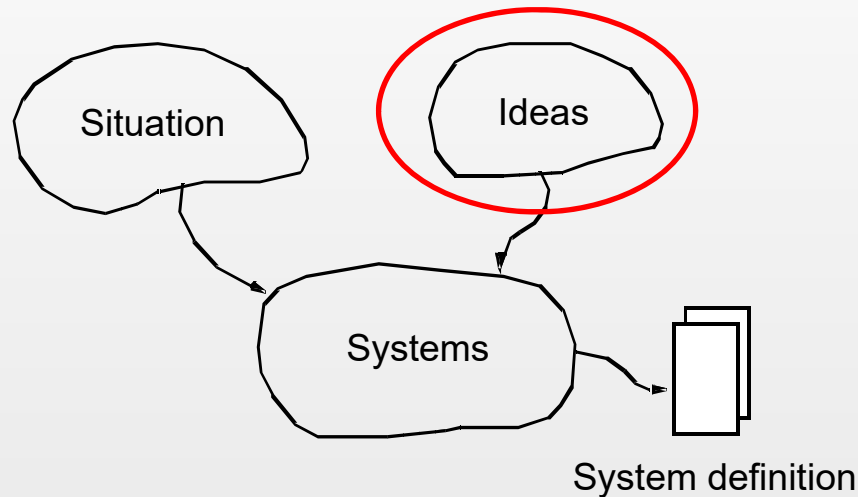
- Complement the description of the existing situation with new ideas
- The software developers are often best able to suggest changes

Create Ideas

- Exemplars (e.g. for a financial system):
  - Study an existing system
  - Look at spreadsheets
  - Examine standard packages
  - Consider SAP
- Metaphors (e.g. for a library):
  - Storage
  - Supermarket
  - School

- Experiments with prototypes:
  - Planning
  - Development
  - Preparation
  - Test
  - Summarizing
- More about prototypes in the DEB course

Systems

- Situation
- Ideas
- Systems
- System definition
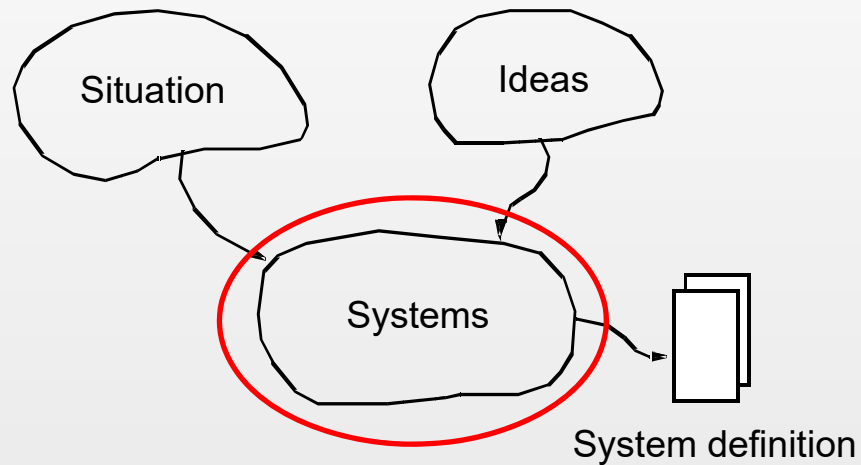
▶ Based on
- The description of the situation
- The new ideas

describe the system to be developed on an overall level

Street Food: Situation

System Definition (FACTOR) 1

Functionality: The system functions that support the application-domain tasks.

Application domain: Both the parts of an organization that administrate, monitor, or control a problem domain; and the users and their work tasks.

Conditions: The conditions under which the system will be developed and used.

F The system can register a new customer with credit card information, compose a meal with a number of orders for food shops and bars, select food or drink items in each order of a meal, make payment of a meal, and register the table where a customer wants his/her orders delivered.

A   A system provided as an app, which is used by customers who want to order food and drinks at S-Food's street food market. The customers come to S-Food at their own initiative, and their only relation to S-Food is that they download and use their app to order and pay for food and drinks.

C   The app will be developed by S-Food's own IT department in cooperation with S-Food's sales department, the food shops and bars, and customers that will be selected to represent the whole customer segment. It may be necessary to resolve conflicting requirements between these different groups. The app will be used by users with very different levels of IT skills.

System Definition (FACTOR) 2

Technology: Both the technology used to develop the system and the technology on which the system will run.

Objects: The main objects in the problem domain.

*Responsibility*: The system's overall responsibility in relation to its context.

T The app is running on each customer's smartphone. It communicates through a wireless network with a server that registers what the individual customer has ordered and paid. On the smartphone, there is always a copy of all meals from the current day, so they are available if the wireless network should fail. The app includes a QR code reader.

O Customer, Food Shop, Bar, Order, Meal, Food item, Drink item.

R The system is primarily an administrative tool that is responsible for registering all customers and their orders, and facilitating secure payment of these with the customers' credit cards. Secondarily, it is a communication medium that customers use to request delivery of orders from the food shops and bars.

Understanding Responsibilities

▸ Kindberg and Fox (2002)

▸ The divide between the responsibilities of the system and the user is a "semantic rubicon": "[that] demarcates responsibility for decision-making between the system and the user."

System responsible for this

User responsible for this

Explicit mechanisms for transferring responsibilities

▸ For the user to understand the semantic rubicon it requires

- defining the knowledge the system can have of the physical world and of user(s) behavior – through sensing and user interaction
- defining the knowledge the user has of the system and how they might influence it
- defining the mechanisms and permissible interactions for one to influence the other and to transfer responsibilities between them
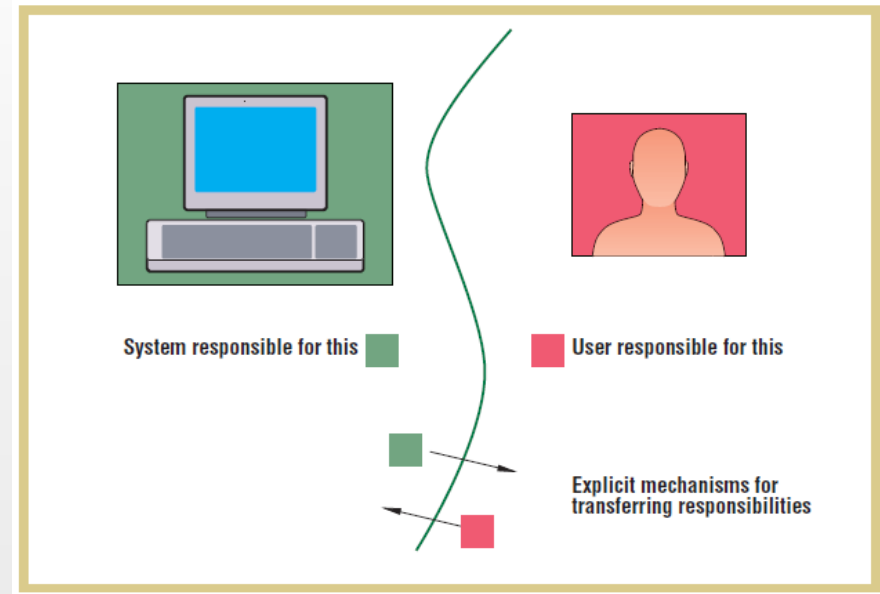
Exploring Responsibilities

▶ Possible responsibilities:

- Automatic system
- Support tool
- Proposer
- …

▶ Example systems:

- Making a schedule for a school
- Airplane landing system
- Greenhouse temperature control

▶ <u>For an example system: how would different responsibilities work?</u>

Evaluation and Choice

▸ Do the system choice activity as early as possible

▸ Iterate within the activity and in subsequent activities

▸ The purpose is to choose <u>one</u> system

▸ The customer/user is responsible for the choice

▸ The system developers provide support and organize the process

▸ Approach: question overall decisions and evaluate alternatives – and do it with the prospective users

System Choice: Summary

Purpose	• To agree on the overall system characteristics.
Concept	• System definition: A concise description of a computerized system expressed in natural language.
Principles	• Appreciate the situation. • Cultivate new ideas. • Define alternative systems.
Results	• A system definition that fulfills the FACTOR criterion.

Contents

- Introduction to the course

- Object-oriented analysis and design

- System Choice

- Varying system definition elements

  - Three examples

- Challenges in this activity

Exploring Alternative System definitions

▶ The system developers explore different alternative systems by changing elements of the system definition

▶ The two most interesting to change:

- Functionality
- Responsibility

Street Food

▶ Functionality 1

- The system can register a new customer with credit card information, compose a meal with a number of orders for food shops and bars, select food or drink items in each order of a meal, make payment of a meal, and register the table where a customer wants his/her orders delivered

▶ Functionality 2

- The system can compose a single order for a food shop or bar, with selection of food or drink items
- The order is picked up at the shop and paid for manually

▶ Responsibility 1

- The system is primarily an administrative tool that is responsible for registering all customers and their orders, and facilitating secure payment of these with the customers' credit cards. Secondarily, it is a communication medium that customers use to request delivery of orders from the food shops and bars

▶ Responsibility 2

- Same as Responbility 1. In addition, the system will suggest more healthy alternatives to the food selected and provide the difference in calories

Bank

▸ The classical bank:
- A bank clerk uses the system to register customer transactions
- For example withdrawal of money

▸ The modern bank:
- A customer uses the system to make transactions

▸ Exercise in lecture 2:
How will the differences be reflected in the system definition, i.e. which element of the system definition is changed (and how)?

1. november 1979 konventerte to av bankens filialer til IDA — Kongsgård og Vågsbygd. Bildet er hentet fra sistnevnte.

Kontakt Find afdeling
Valuta Find automat
Log på
Overfør Konto
Betal

Danske Bank

Warehouse

▶ <u>Exercise in lecture 2:</u>

▶ What would the system definition for a traditional warehouse be?

▶ How should we change it with a modern warehouse like IKEA?

  • Which elements (FACTOR) are changed?

Contents

▸ Introduction to the course

▸ Object-oriented analysis and design

▸ System Choice

▸ Varying system definition elements

▸ **Challenges in this activity**

Challenges in this Activity

▶ Difficulties differentiating:

- F – A
- F – R
- T – C

▶ Work in exercises for this activity

- Write text for each element of the FACTOR criterion
- Experiment with variations on selected elements
- If desired, write it out as a coherent text (see textbook)