



Systems Development



Lecture 2: Classes

Contents

- ▶ Summary of last lecture
- ▶ Problem domain analysis
- ▶ The Classes activity
- ▶ Challenges in this activity

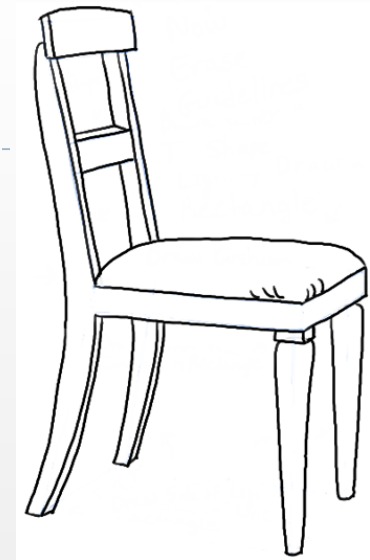
Contents

- ▶ Summary of last lecture
 - Object and class
 - Context, system and interplay
 - System choice: rich picture, ideas, FACTOR
- ▶ Problem domain analysis
- ▶ The Classes activity
- ▶ Challenges in this activity

Key Concepts: Object and Class

► Object:

- **An entity with: identity, state, and behavior**
- identity: myChair
- state: by dining table, free
- behaviour: bought, moved to, ... , sat down on, got up from, ..., moved to, ..., sold



► Class:

- **A description of a collection of objects sharing: structure, behavioral pattern, and attributes**
- structure: has an owner
- attributes: position, vacant
- behavioural pattern: buy + {move | sit down on + get up from}* + sell



Model the Context

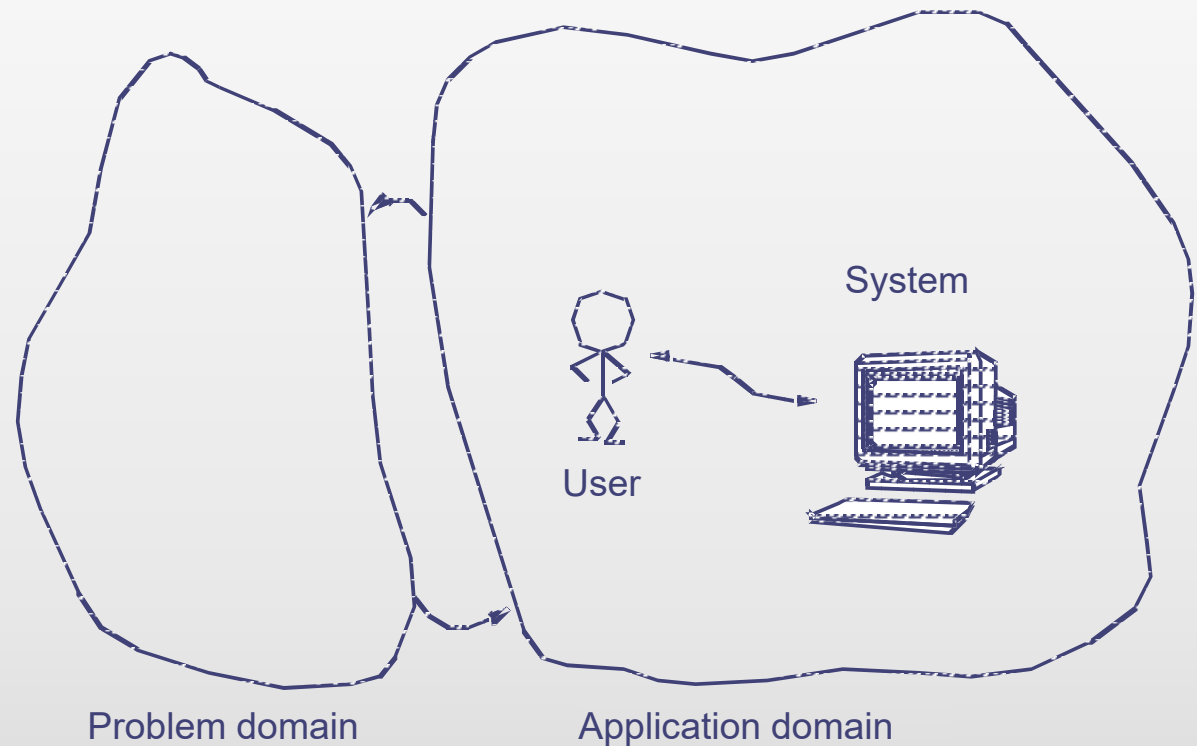
Focus on an IT system
and its context

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



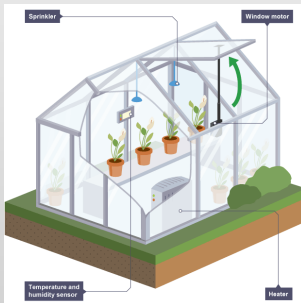
A Model of the Problem Domain

Model:

An updated representation of the state in the problem domain

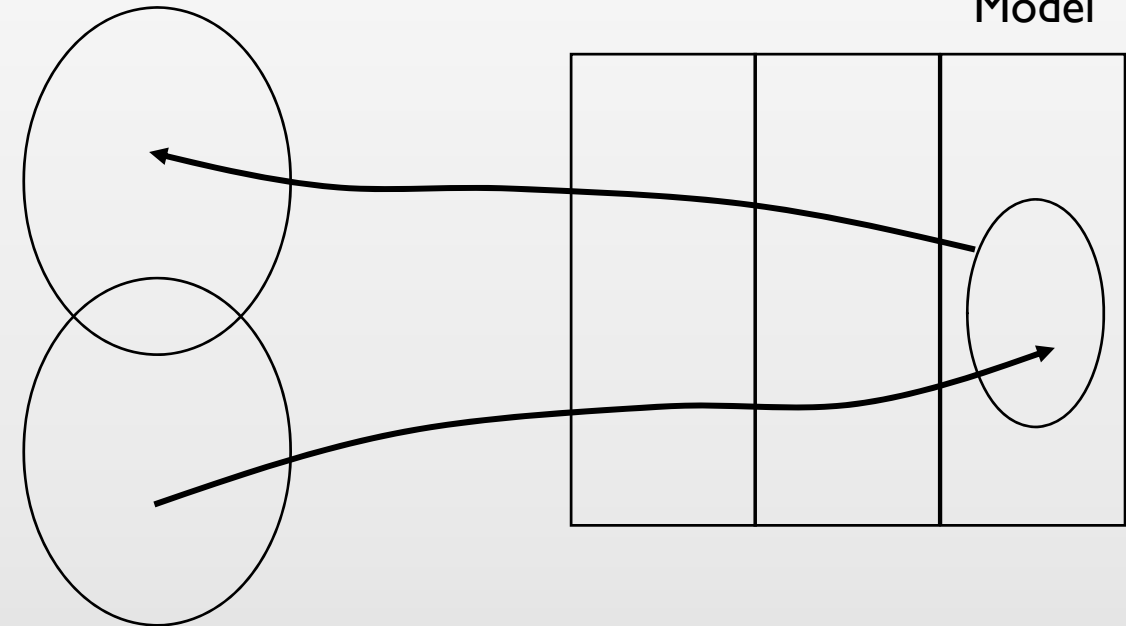
User:

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



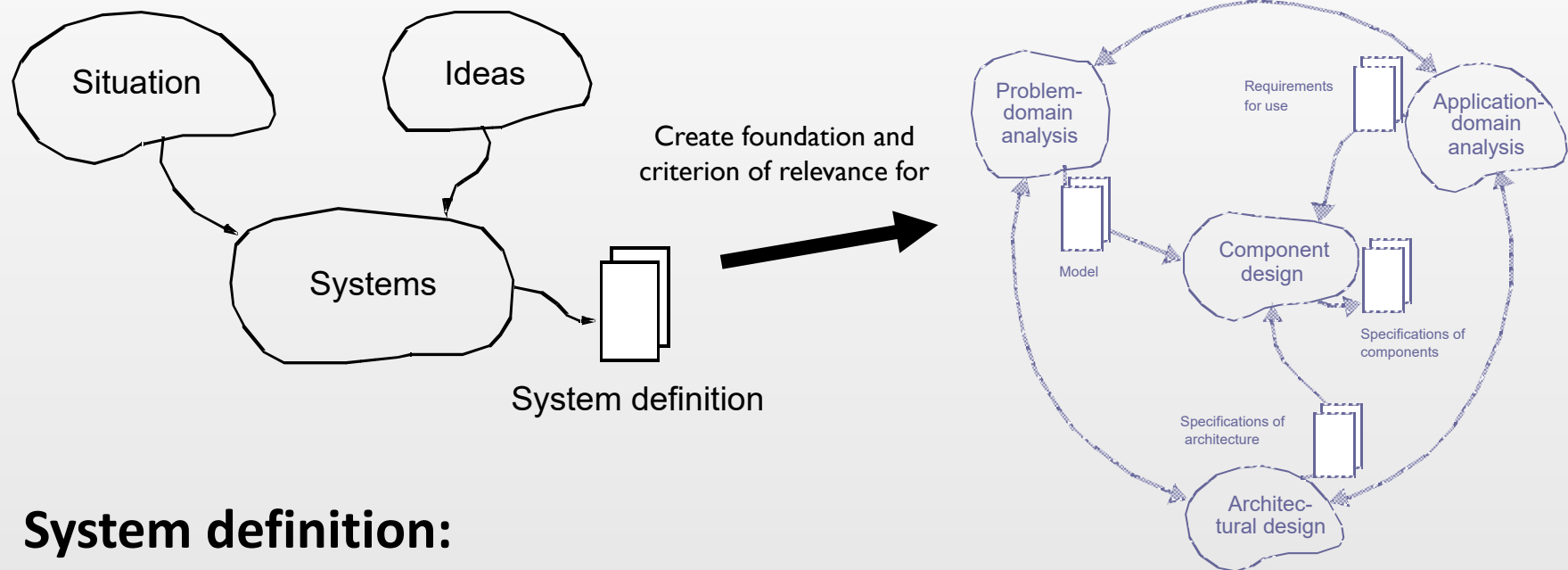
Application domain

Model



Problem domain

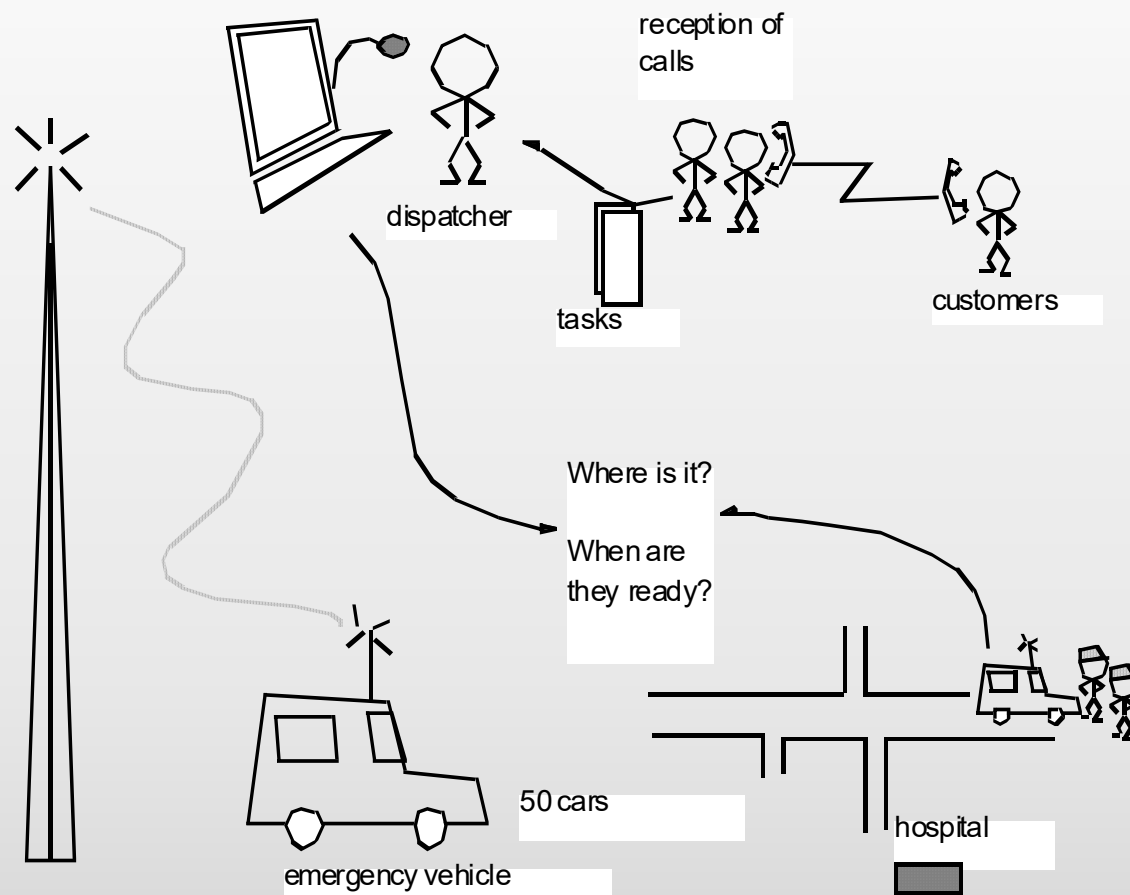
System Choice: Purpose and Activities



System definition:

A concise description of a computerized system expressed in natural language

Rich Picture



► Entities:

- People
- Roles and tasks
- Locations

Focus on stability

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

System Definition (FACTOR) I

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

Application domain: Those parts of an organization that administrate, monitor, or control a problem domain; 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. Secondly, it is a communication medium that customers use to request delivery of orders from the food shops and bars.

Difficulties with System Definition Elements

- ▶ The difference between Functionality and Responsibility
 - Functionality: The system functions that support the application-domain tasks
Support for program design. Automatic participant registration
 - Responsibility: The system's overall responsibility in relation to its context
Administrative tool, automatic device, communication medium, proposer, etc.
 - Greenhouse: (F) update and record temperature / (R) display with user intervention (monitor) or automatic system (control)
- ▶ The difference between Technology and Conditions:
 - Technology: Both the technology used to develop the system and the technology on which the system will run
Cheap PC platform with current tools, smartphone
 - Conditions: The conditions under which the system will be developed and used
Volunteer labor has widely varying administrative experience. Development to proceed despite contradictory and missing requirements
 - Greenhouse: (T) the technology for measuring, communicating and displaying / (C) in collaboration with a professional gardener and a group of other users

Exploring Alternative System Definitions

- ▶ The system developers explore different alternative systems by changing elements of the system definition
- ▶ Example: street food, traditional/modern bank or warehouse (exercise today)
- ▶ Evaluate and select the one that the customers/users want

System Choice: Summary

Purpose	<ul style="list-style-type: none">• To agree on the overall system characteristics.
Concept	<ul style="list-style-type: none">• System definition: A concise description of a computerized system expressed in natural language.
Principles	<ul style="list-style-type: none">• Appreciate the situation.• Cultivate new ideas.• Define alternative systems.
Results	<ul style="list-style-type: none">• A system definition that fulfills the FACTOR criterion.

Quiz I Overview

Quiz 1

Average

5.35 (of 9.00) of 126 finished attempts (of 175)

Best result (0.67-1.00)

9 (0.88) The FACTOR criterion covers the following items:

5 (0.71) What is/are the purpose(s) of analysis?

2 (0.69) The purpose(s) of OOA&D is/are?

Middle result (0.34-0.66)

7 (0.66) What are the sub-activities of system choice?

3 (0.65) A system for boat reservations. Problem domain.

1 (0.52) What is a system?

4 (0.50) A system for boat reservations. Application domain.

8 (0.41) Which knowledge areas can rich pictures help clarify?

6 (0.34) What is design?

Worst result (0.00-0.33)

none

Quiz I Difficult Questions

Question 1

What is a system?

Select one:

- ☐ a. A collection of components that implements requirements, functions and design
- ☐ b. A collection of components that implements problem domains, functions and interfaces
- ☒ c. A collection of components that implements requirements, functions and interfaces
- ☐ d. A collection of components that implements requirements, the application domain and interfaces

Question 3

A system for boat reservations in a boat rental company. A reservation contains the name of the renter, boat size (max. number of passengers) and a period.

Which of the following can be classes in the problem domain?

Select one or more:

- ☐ a. Server
- ☐ b. Owner of the rental company
- ☐ c. Passenger
- ☒ d. Boat
- ☐ e. Log-in
- ☒ f. Renter

Question 4

A system for boat reservations. A reservation consists of the name of the renter, the size of the boat (max. number of passengers) and a period.

Which of the following may be part of the application domain?

Select one or more:

- ☒ a. Renter
- ☐ b. Boat
- ☒ c. The owner of the boat rental
- ☐ d. Log-in
- ☐ e. Passenger
- ☒ f. IT-system with clients and server

Question 8

Which knowledge areas can rich pictures help clarify?

Select one or more:

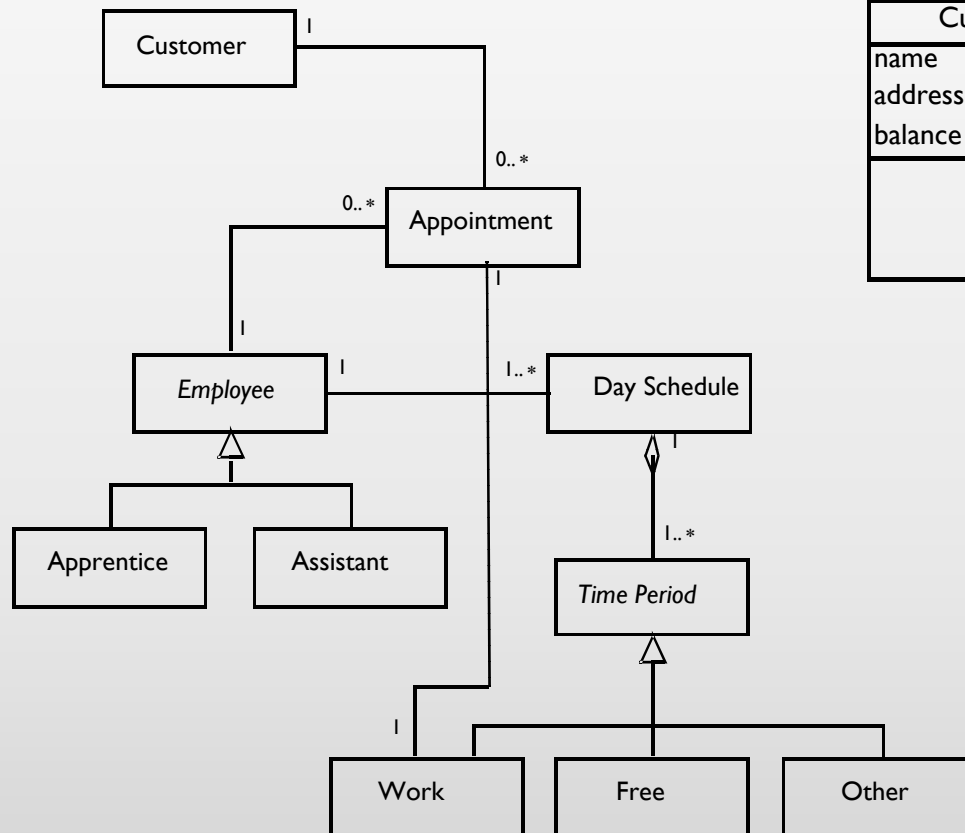
- ☒ a. Concrete experience with user's work
- ☐ b. Concrete experience with the new system
- ☐ c. Overview of technological options
- ☒ d. Relevant structures on user's work
- ☐ e. Visions and design proposals

Contents

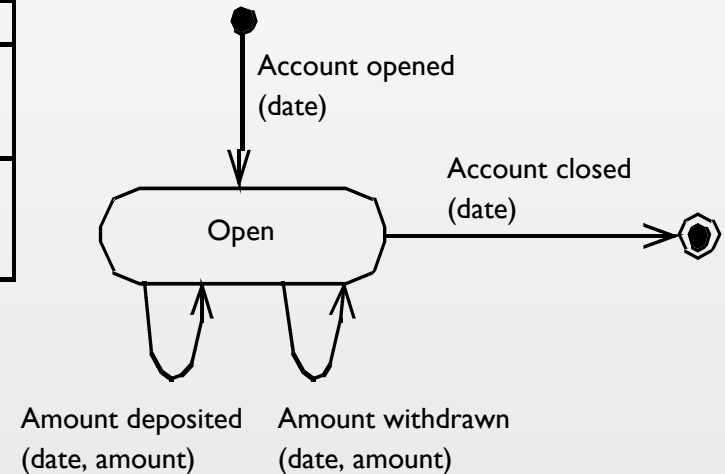
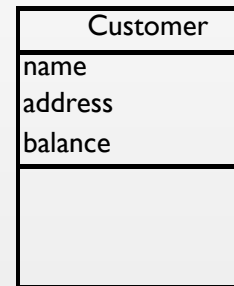
- ▶ Summary of last lecture
- ▶ Problem domain analysis
 - Results
 - Key concepts
 - Activities
- ▶ The Classes activity
- ▶ Challenges in this activity

Problem Domain Analysis: Results

Class diagram



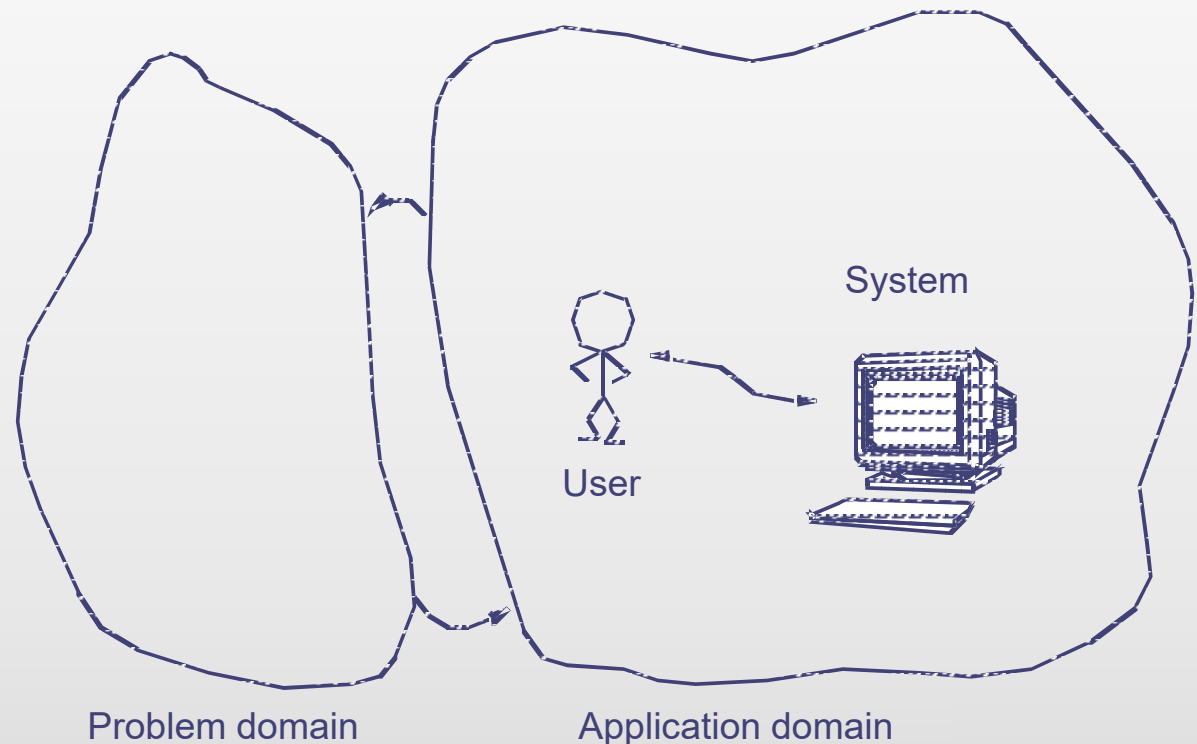
Behavioural pattern for each class



Problem Domain Analysis: Key Concepts

Problem domain:

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



The Model of the Problem Domain

Model:

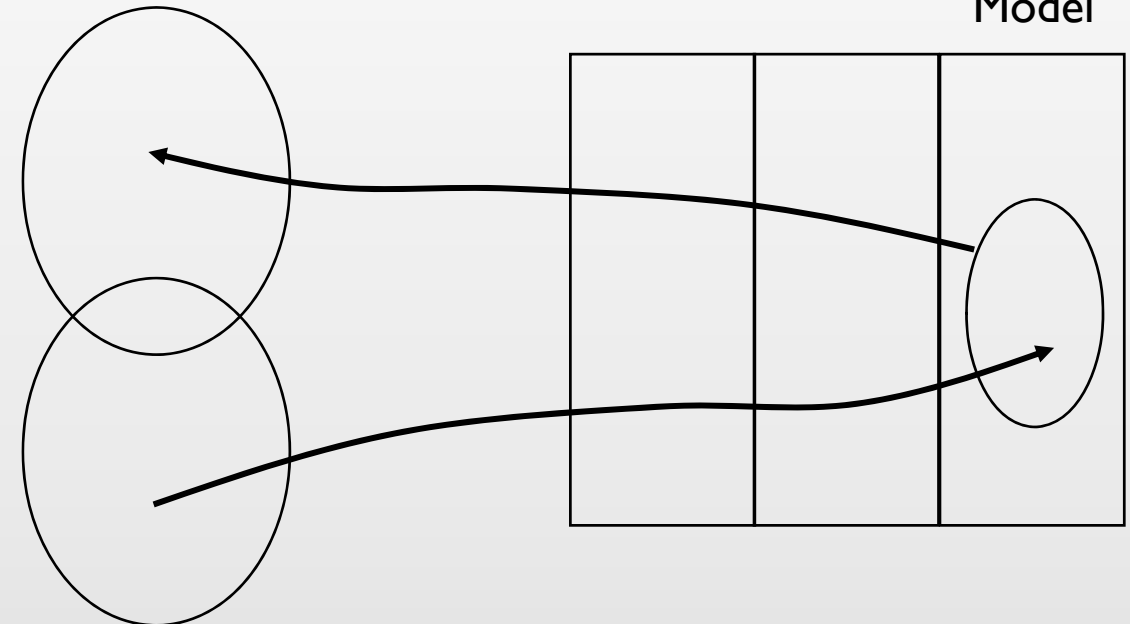
A description of classes, objects, structures, and behavior in a problem domain

Provides an updated representation of the state in the problem domain

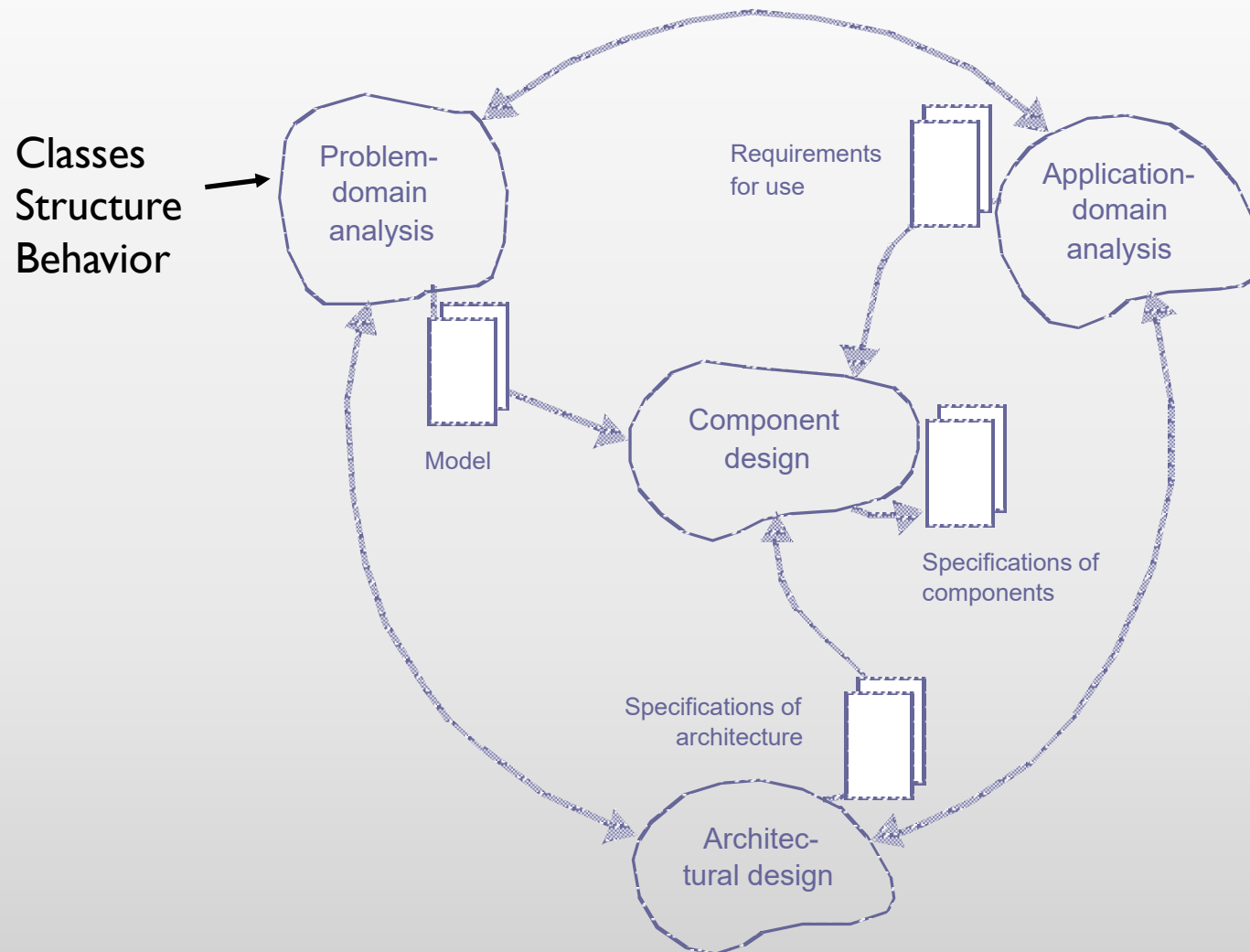
- ▶ In analysis, we describe the problem domain by an object-oriented model
- ▶ In design, we adapt this model to become the model component of the system

Application domain

Model



Problem Domain Analysis:Activities



Problem Domain Analysis: Summary

Purpose	<ul style="list-style-type: none">• To identify and model a problem domain
Concepts	<ul style="list-style-type: none">• Problem domain: That part of a context that is administrated, monitored, or controlled by a system.• Model: A description of classes, objects, structures, and behavior in a problem domain.
Principles	<ul style="list-style-type: none">• Model the real world as future users will see it.• Get an overview first, then supply details.
Results	<ul style="list-style-type: none">• A coherent model of a problem domain.

Contents

- ▶ Summary of last lecture
- ▶ Problem domain analysis
- ▶ The Classes activity
 - Results
 - Key concepts
 - Activities
- ▶ Challenges in this activity

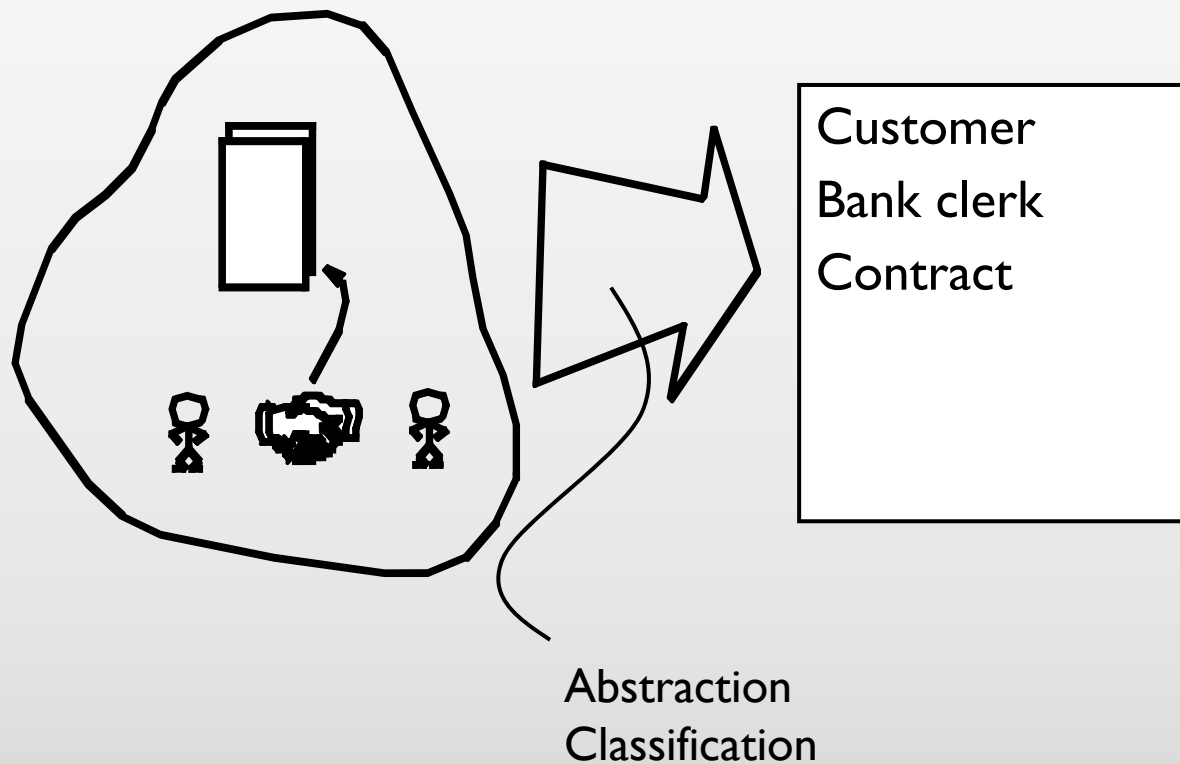
Classes: Result

Event table

	reserved	cancel	treated	employed	resigned	grad.	agreed
Customer	✓	✓	✓				
Assistant	✓	✓		✓	✓		✓
Apprentice				✓	✓	✓	✓
Appointment	✓	✓	✓				
Plan	✓						✓

Shows major classes and events in the problem domain

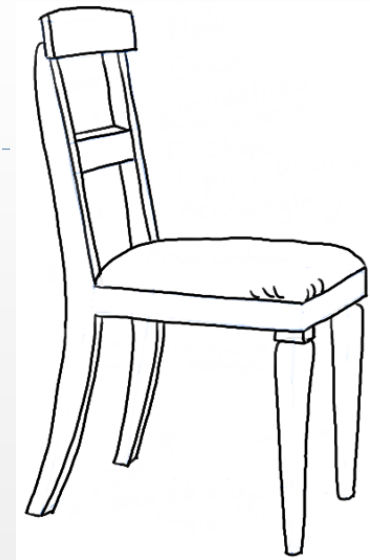
Classify Objects and Events in the Problem Domain



Key Concepts: Object and Class

► Object:

- **An entity with: identity, state, and behavior**
- identity: myChair
- state: by dining table, free
- behaviour: bought, moved to, ... , sat down on, got up from, ..., moved to, ..., sold



► Class:

- **A description of a collection of objects sharing: structure, behavioral pattern, and attributes**
- structure: has an owner
- attributes: position, vacant
- behavioural pattern: buy + {move | sit down on + get up from}* + sell

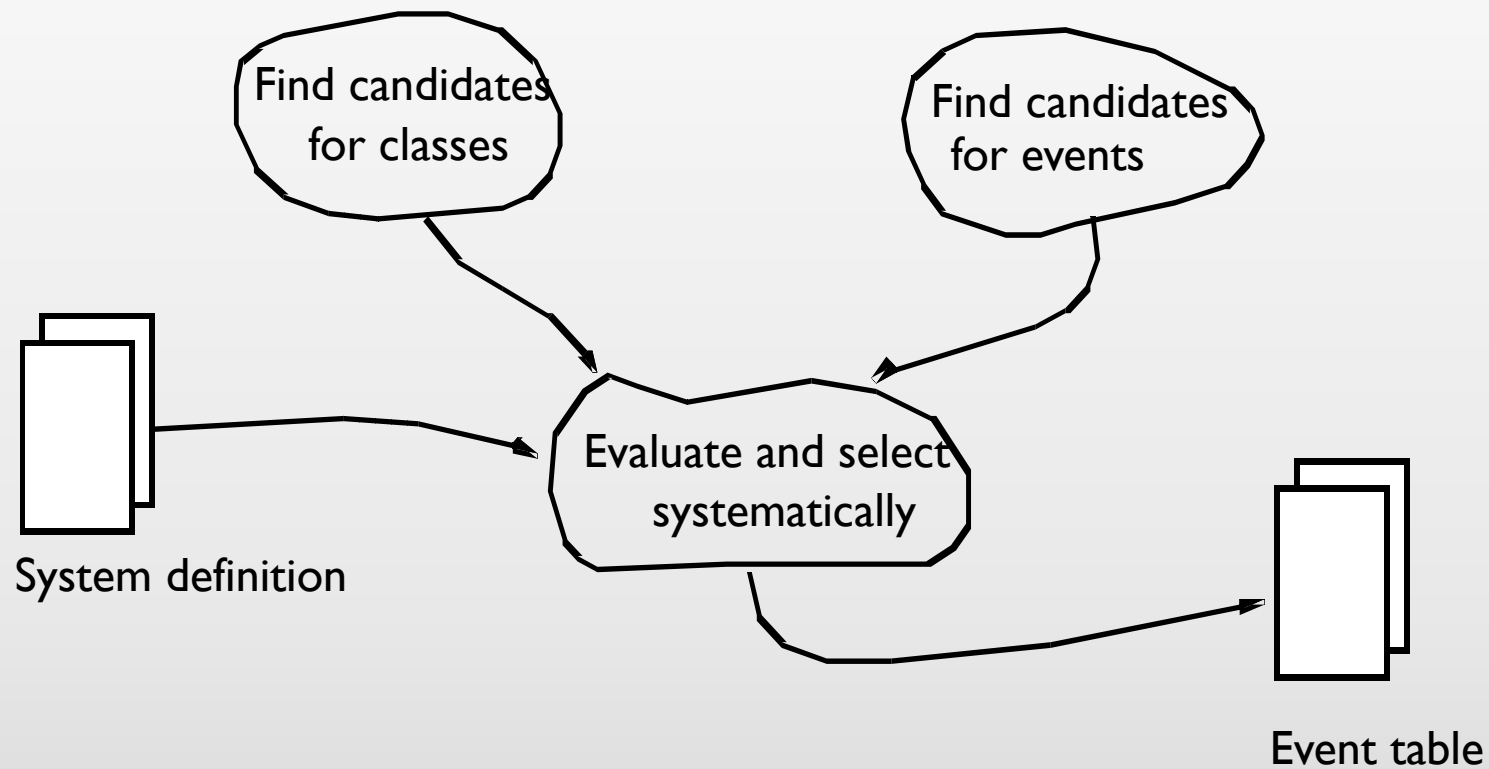


Key Concepts: Event

- ▶ Event:
An instantaneous incident involving one or more objects
- ▶ Atomic
- ▶ Common to several objects
- ▶ Unique name
- ▶ Example from IKEA
 - What are relevant events?



Classes: Activities



Find Candidates for Classes and Events

Common technique to generate candidate lists (brainstorming):

- ▶ Make a list of all potentially relevant classes/events

Consider many sources:

- ▶ Your own perception of the problem domain
- ▶ Existing descriptions (rich pictures, the system definition, etc.)
- ▶ Collaborate with prospective users

Naming:

- ▶ Must be simple and readable
- ▶ Originate in the problem domain

For classes:

- ▶ Look at nouns
- ▶ Eliminate nouns that reflect administration objects or devices
- ▶ Describe a single instance (object)

For events:

- ▶ Look at verbs
- ▶ Eliminate verbs related to the way users carry out their job – these belong to the application domain
- ▶ Indicate a single event

Example: Hair Salon

(in Chapter 20)

<i>Functionality:</i>	Support for work planning and appointments
<i>Application domain:</i>	Managing customers, their treatments, and appointments, and planning employees' work schedules
<i>Conditions:</i>	Developed in close cooperation with the employees
<i>Technology:</i>	Smaller PC or Macintosh with a large graphical screen
<i>Objects:</i>	Customers, employees, appointment, and work schedules
<i>Responsibility:</i>	Tool for reliable administration and a common mediator

Example: Candidates for Classes and Events

- ▶ Suggest classes and events for modelling a hair salon with emphasis on work planning and administration of appointments

Example: Candidates for Classes and Events

Classes

- ▶ Plan
- ▶ Customer database
- ▶ Appointment book
- ▶ Cash register
- ▶ Appointment
- ▶ Treatment performed
- ▶ Desired vacation
- ▶ Work schedule
- ▶ Boss, assistant, receptionist
- ▶ Apprentice
- ▶ Customer
- ▶ Chair
- ▶ Salon

Events

- ▶ reserved
- ▶ cancelled
- ▶ customer arrived
- ▶ treated
- ▶ payment received
- ▶ employed
- ▶ resigned
- ▶ graduated
- ▶ agreed
- ▶ material used
- ▶ item purchased
- ▶ customer picked up
- ▶ arrived workplace
- ▶ leave workplace

Systematic Evaluation

General evaluation criteria

- Is the class or event within the system definition?
- Is the class or event relevant for the problem-domain model?

Criteria for classes

- Can you identify objects from the class?
- Does the class contain unique information?
- Does the class encompass multiple objects?
- Does the class have a suitable and manageable number of events?

Criteria for events

- Is the event instantaneous?
- Is the event atomic?
- Can the event be identified when it happens?

Example: Selection of Classes and Events

- ▶ Which of the candidate classes and events should we select?

Example: Selection of Classes and Events

Classes

- ▶ Plan +
- ▶ Customer database –
- ▶ Appointment book –
- ▶ Cash register –
- ▶ Appointment +
- ▶ Treatment performed –
- ▶ Desired vacation –
- ▶ Work schedule –
- ▶ Boss, assistant, receptionist + (Assistant)
- ▶ Apprentice +
- ▶ Customer +
- ▶ Chair –
- ▶ Salon –

Events

- ▶ reserved +
- ▶ cancelled +
- ▶ customer arrived –
- ▶ treated +
- ▶ payment received –
- ▶ employed +
- ▶ resigned +
- ▶ graduated +
- ▶ agreed +
- ▶ material used –
- ▶ item purchased –
- ▶ customer picked up –
- ▶ arrived workplace –
- ▶ leave workplace –

Street Food – Where do objects belong?

	Kun i problem-området (PD)	Kun i anvendelses-området (AD)	Både i problem-området og anvendelsesområdet (PD og AD)	Hverken i problemområdet eller anvendelses-området
S-Food				
Madforretning				
Bar				
Kunde				
Administrativt personale				
Personale i forretning og bar				
Måltid				
Ordre				
Madvare				
Drikkevare				
Kreditkort				
IT-afdeling				
Smartphone				
Server				
Bord				
Stol				

Answer

	Kun i problem- området (PD)	Kun i anvendelses- området (AD)	Både i problem- området og anvendelsesområdet (PD og AD)	Hverken i problemområdet eller anvendelses- området
S-Food				X
Madforretning	X			
Bar	X			
Kunde			X	
Administrativt personale		X		(X)
Personale i forretning og bar		X		(X)
Måltid	X			
Ordre	X			
Madvare	X			
Drikkevare	X			
Kreditkort	X			
IT-afdeling				X
Smartphone		X		
Server				X
Bord	X			
Stol				X

Classes: Summary

Purpose	<ul style="list-style-type: none">• To select the elements of a problem-domain model.
Concepts	<ul style="list-style-type: none">• Object: An entity with identity, state, and behavior.• Class: A description of a collection of objects sharing structure, behavioral pattern, and attributes.• Event: An instantaneous incident involving one or more objects.
Principles	<ul style="list-style-type: none">• Classify objects in the problem domain.• Characterize objects through their events.• Have an open mind, but select critically.
Results	<ul style="list-style-type: none">• An event table with classes and related events.

Contents

- ▶ Summary of last lecture
- ▶ Problem domain analysis
- ▶ The Classes activity
- ▶ Challenges in this activity

Challenges

- ▶ Classifying classes and events is mentally challenging
 - Difficult to find the right granularity
 - Difficult to transcend the existing situation
- ▶ Selecting the relevant classes and events is often difficult
 - Use the criteria
 - Be explicit about the reason for excluding
 - Try to imagine how the information that can be provided to the user changes if a class or event is included or excluded
- ▶ Functions are often described as events – but we do usually not want to register the application of a function
 - For example: search for or access information
 - Make a preliminary function list to remember them

Work in Exercises for this Activity

- ▶ Find candidates for classes (don't discuss now)
- ▶ Find candidates for events (don't discuss now)
- ▶ Select the relevant classes and events (discuss now)
- ▶ Be precise in the argumentation about which classes and events that are in or out (use the criteria)
- ▶ Combine classes and events in an event table (which classes are involved in which events)