



Systems Development



Lecture 5: Usage

Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
- ▶ The Usage activity
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
- ▶ The Usage activity
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

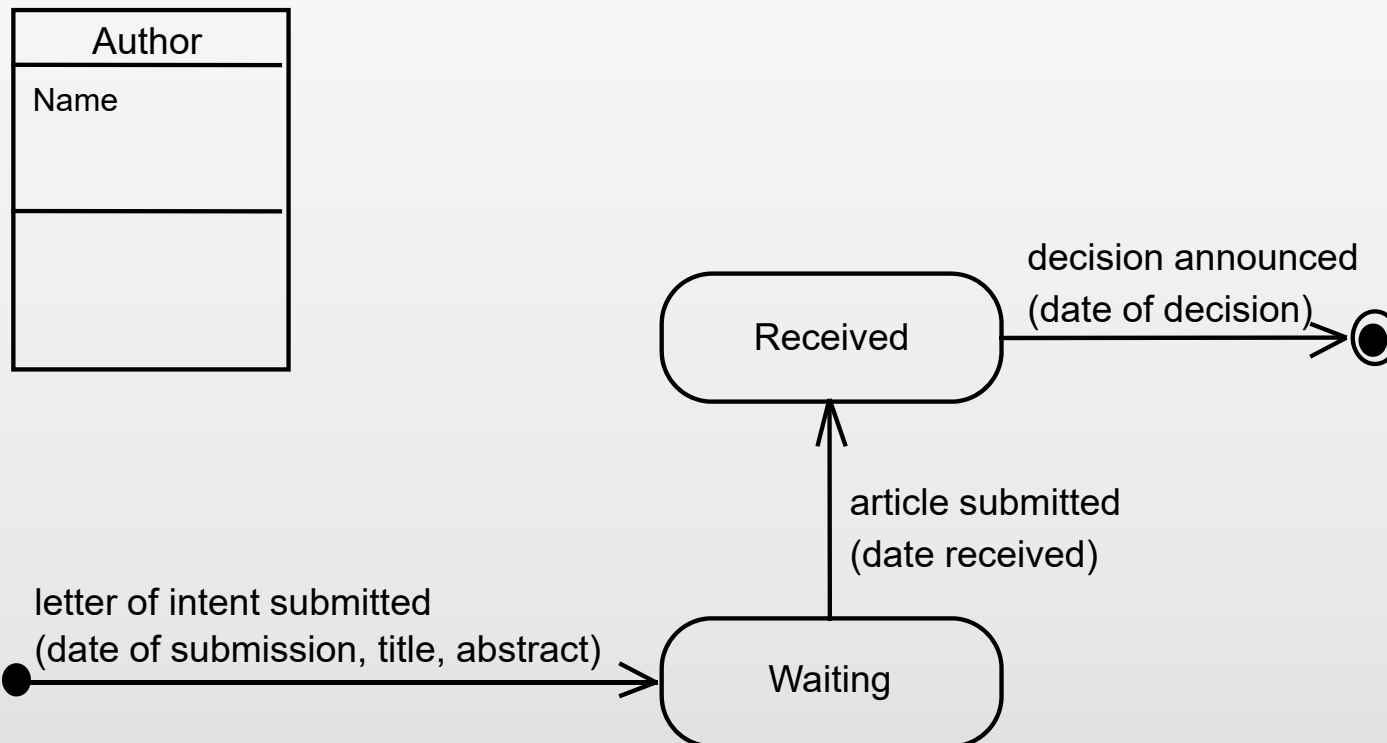
Exam Sets from Previous Exams

- ▶ Exam sets from previous exams will be published on the moodle page (see the bottom of the front page)
- ▶ The first set is there now
- ▶ You have the opportunity to try to solve the exam set individually before you see the solution
- ▶ The solution to the exam set is published approximately 2 weeks later
- ▶ At the same time, the next set is published

Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
 - The Behaviour activity
- ▶ Application Domain analysis
- ▶ The Usage activity
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

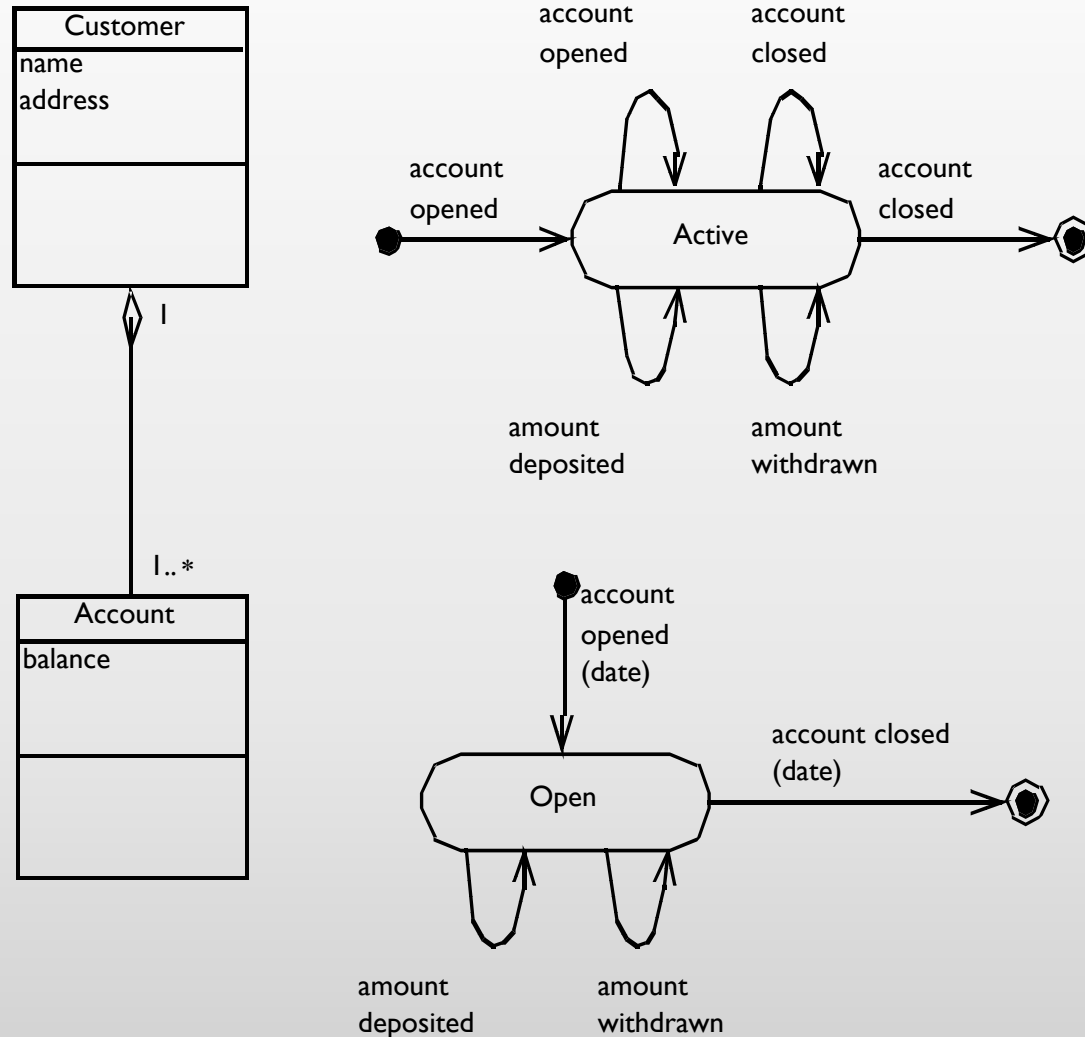
Behaviour: Result



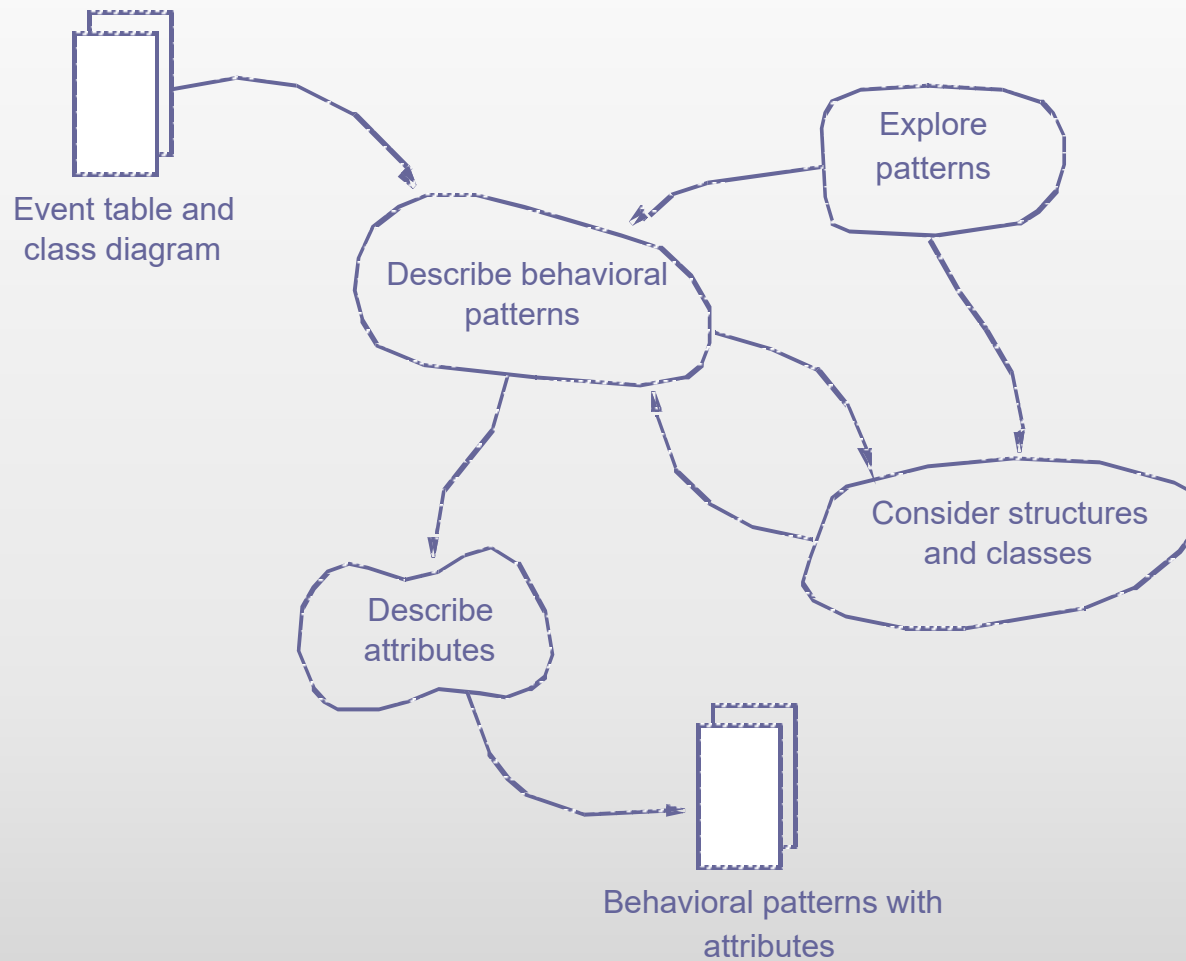
Common Events: Event Table

	Customer	Assistant	Apprentice	Appointment	Plan
reserved	*	*		+	*
cancelled	*	*		+	
treated	*			+	
employed		+	+		
resigned		+	+		
graduated			+		
agreed		*	*		*

Common Events: State Chart Diagrams



Behaviour:Activities



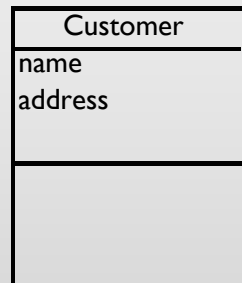
Explore Patterns

- ▶ Stepwise relation
- ▶ Stepwise role
- ▶ Composite

Describe Attributes

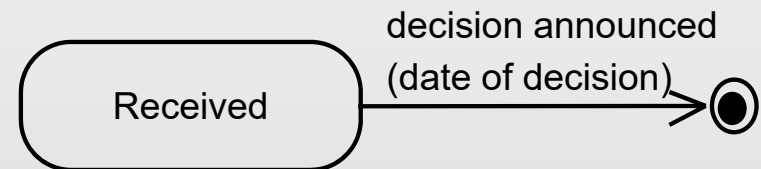
▶ Class attributes:

- What are the general characteristics of the class?
- What basic data must be captured about objects from this class?
- What results from an event trace must be captured?



▶ Event attributes:

- What time did the event occur?
- Which amount did it concern?



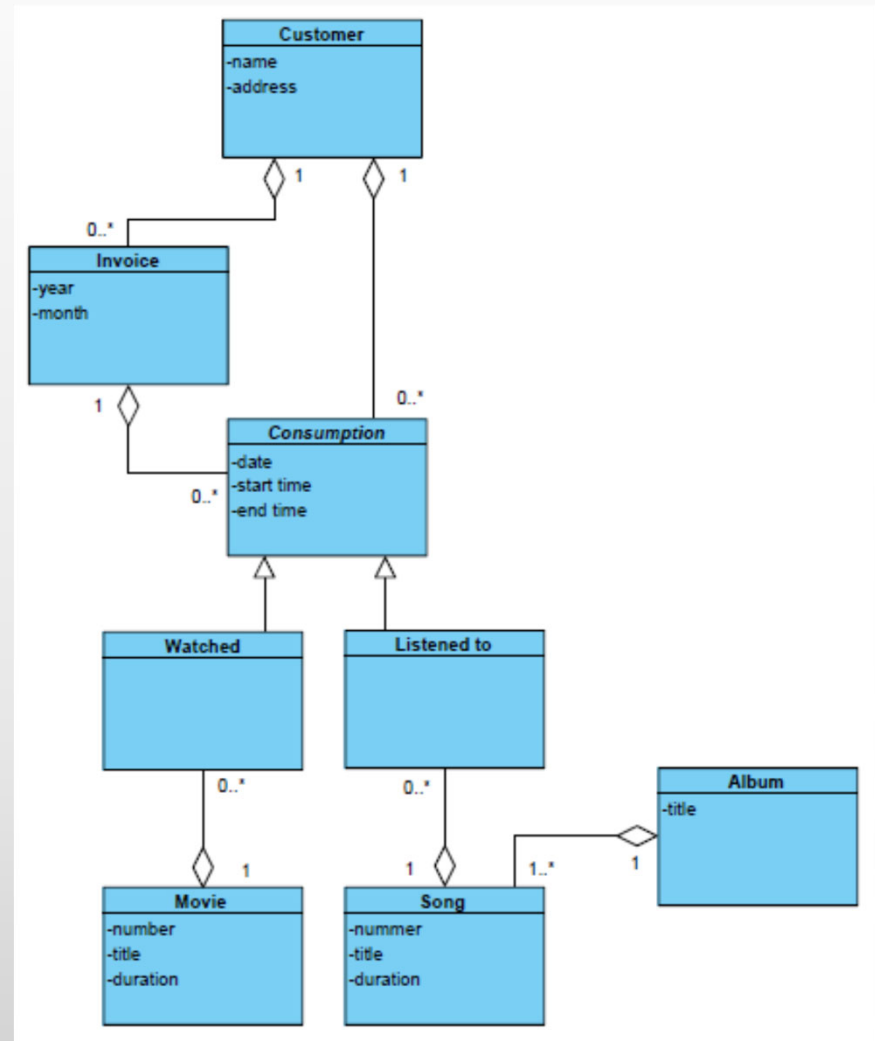
Behaviour: Summary

Purpose	<ul style="list-style-type: none">• To model the dynamics of a problem domain.
Concepts	<ul style="list-style-type: none">• Event trace: A sequence of events involving a specific object.• Behavioral pattern: A description of possible event traces for all objects in a class.• Attribute: A descriptive property of a class or an event.
Principles	<ul style="list-style-type: none">• Create behavioral patterns from event traces.• Study common events.• Derive class attributes from behavioral patterns.
Results	<ul style="list-style-type: none">• A behavioral pattern with attributes for every class in a class diagram

Class Diagram for the Streaming Service

► System definition:

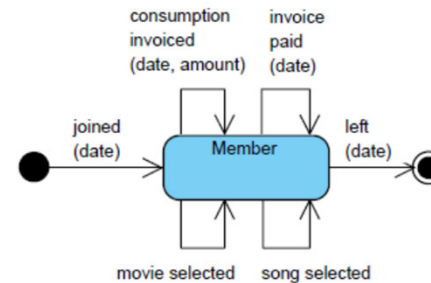
- F: register the movies the customers see and the records they listen to, and support payment by customers for their consumption of movies and songs.
- A: will be used by the administrative personnel that is employed by the organization that provides the streaming service.
- C: developed for the administrative personnel.
- T: PC platform with typical tools.
- O: Customer, Movie, Song.
- R: registration, administration and payment of customers' consumption.



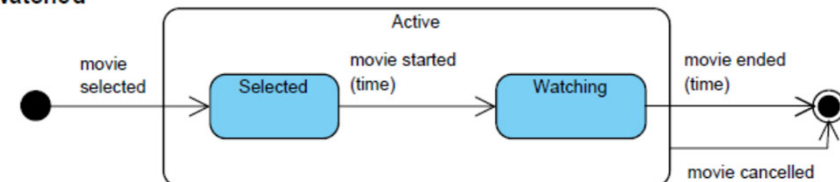
Statechart Diagrams

- ▶ Make statechart diagrams for the following classes:
- ▶ Customer
- ▶ Movie
- ▶ Watched

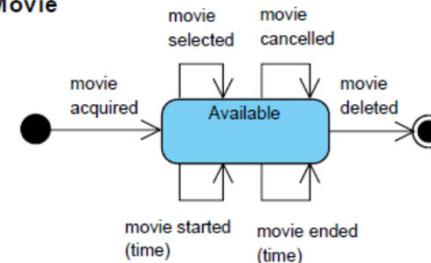
Customer



Watched



Movie



Quiz 4 Overview

Quiz 4

Average

4.68 (of 6.00) of 70 finished attempts (of 159)

Best result (0.67-1.00)

- 2 (0.96) Match the three diagrams with the right notation: (sequence, selection, iteration)
- 3 (0.89) Which of these statements are correct? (sequence, selection, iteration)
- 6 (0.86) Given the class diagram and behavioral pattern shown in the diagram; which of the given solutions is a legal behavioral pattern for sub class 1?
- 4 (0.76) Choose the legal paths in the statechart diagram shown below

Middle result (0.34-0.66)

- 1 (0.66) A behavioral pattern:
- 5 (0.54) Which of these statements are true for the Stepwise Relation Pattern?

Worst result (0.00-0.33)

None

Quiz 4 Answers

▶ 1. A behavioural pattern

A behavioral pattern:

Select one or more:

- ☐ a. describes the use of the target system
- ☒ b. captures the dynamic character of the problem domain
- ☐ c. captures the dynamic character of the application domain
- ☒ d. describes possible event traces for objects of a class
- ☐ e. describes event traces for specific objects

▶ 5. Which of these statements are true for the Stepwise Relation Pattern?

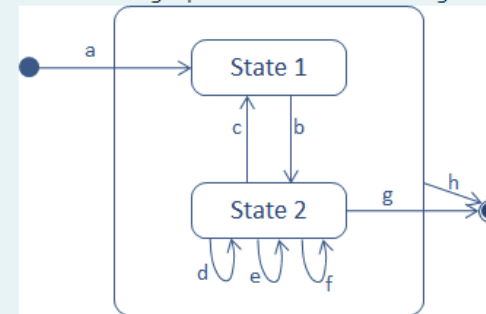
Which of these statements are true for the Stepwise Relation Pattern?

Select one or more:

- ☐ a. Association and generalization are used as the structures in the generic pattern
- ☐ b. Aggregation and generalization are used as the structures in the generic pattern
- ☐ c. It describes how a whole changes as its parts become active
- ☒ d. Aggregation and association are used as the structures in the generic pattern
- ☒ e. It describes how elements in a hierarchy are related in a sequential manner

▶ 4. Choose the legal paths in the statechart diagram shown below

Choose the legal paths in the statechart diagram shown below



Select one or more:

- ☐ a. abcbceeffbh
- ☐ b. abcfeg
- ☐ c. acbddffeg
- ☒ d. abfffeedg
- ☒ e. abdddeeefffg
- ☐ f. ahgfedcb
- ☒ g. abh
- ☒ h. abcbdddch

Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
 - Results
 - Key concepts
 - Using the model
 - Activities
- ▶ The Usage activity
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

Application Domain Analysis: Results

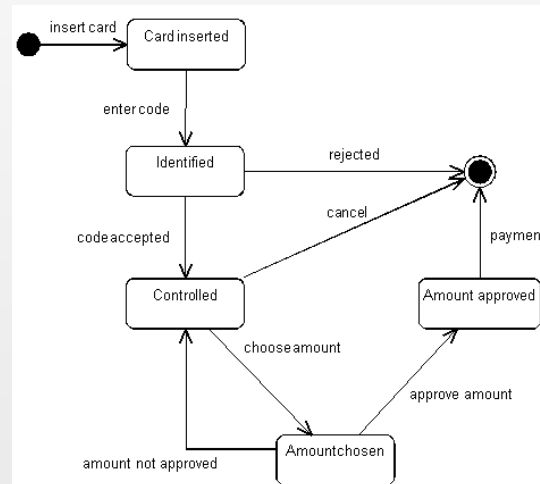
► Actors and Use cases

Account Owner

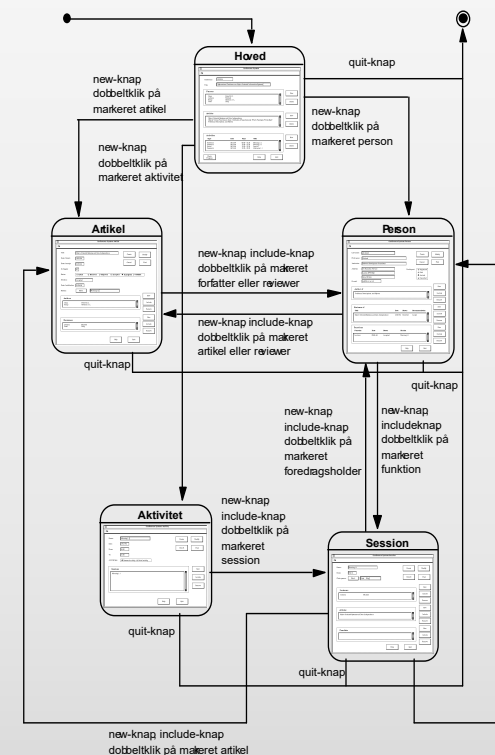
Purpose: A person who owns an account. The account owner's basic need is to be able to make payments with his plastic card.

Characteristic: The system's users include many and very different account owners

Examples: Account owner A is insecure in the use of a plastic card as a form of payment. A originally received his card because it was the only possibility...



► User interfaces



► Functions

Lav plan	Særdeles kompleks	Opdatering
Konsekvensberegning plan	Kompleks	Signalering
Find kundeoplysninger	Middel	Aflæsning
Sæt indhold i plan	Kompleks	Opdatering
Slet plan	Simpel	Opdatering
Lav reservation	Middel	Opdatering
...

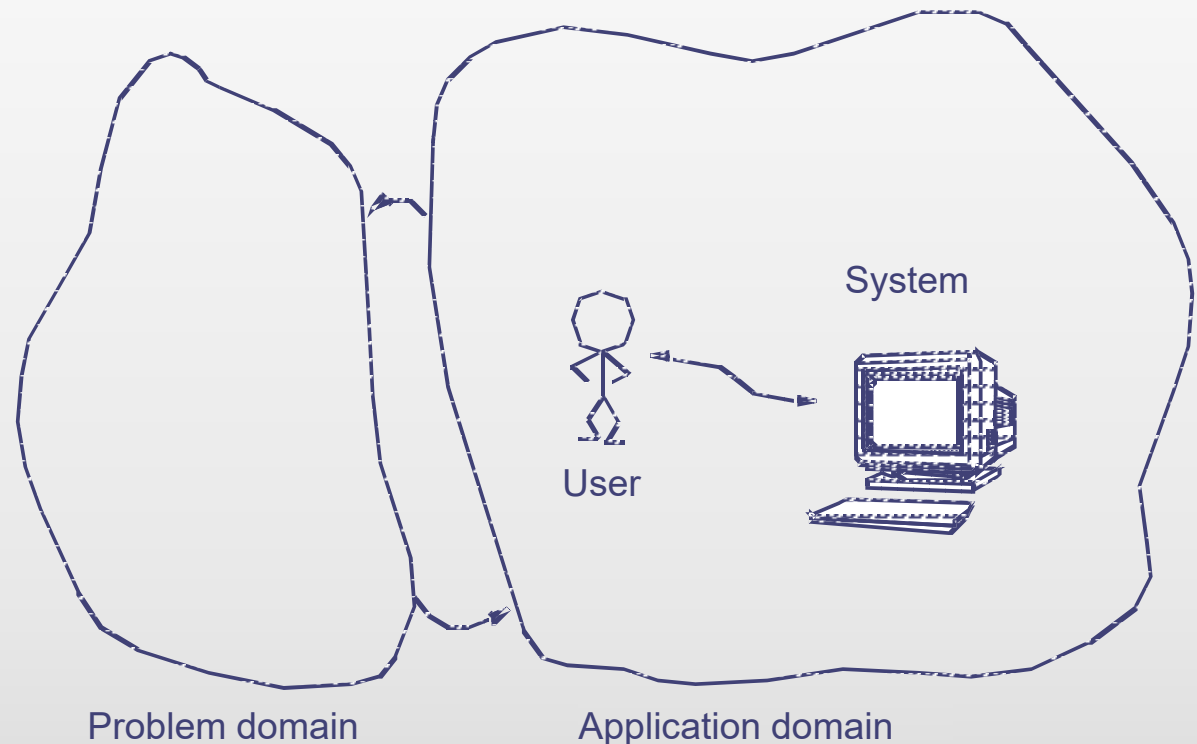
Application Domain Analysis: Key Concepts

Application domain:

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

New Concepts:

- ▶ Actors (users and other systems)
- ▶ Use cases
- ▶ Functions
- ▶ Interfaces



Using The Model of the Problem Domain

Model:

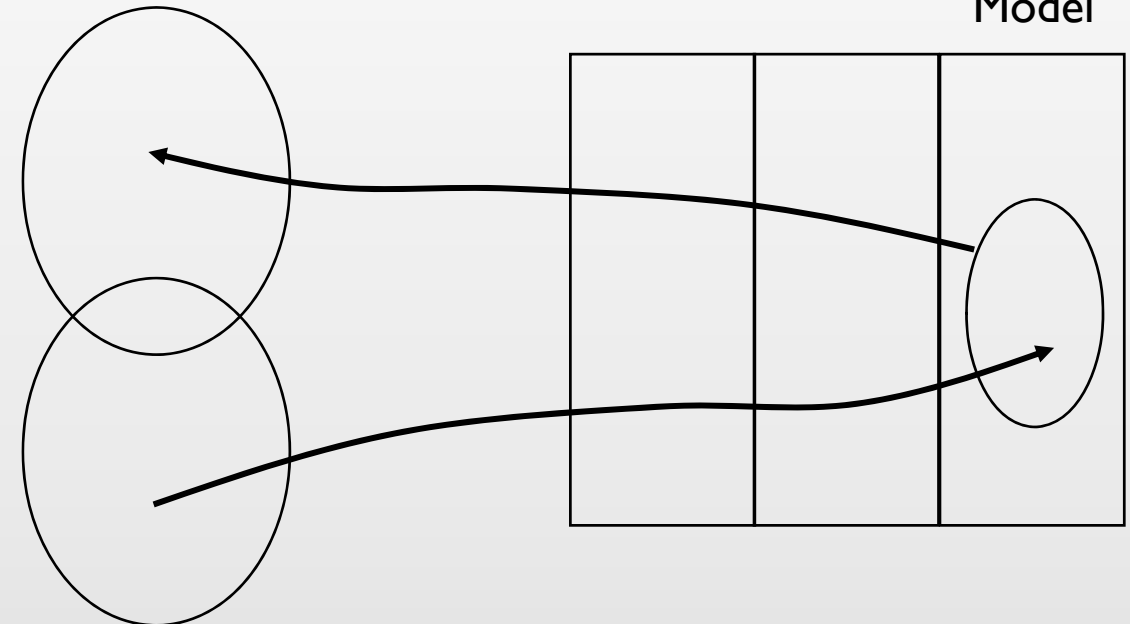
A description of classes, objects, structures, and behavior in a problem domain

Provides an updated representation of the state in the problem domain

- ▶ In analysis, we describe the problem domain by an object-oriented model
- ▶ We also describe how the model is used in the application domain by the actors

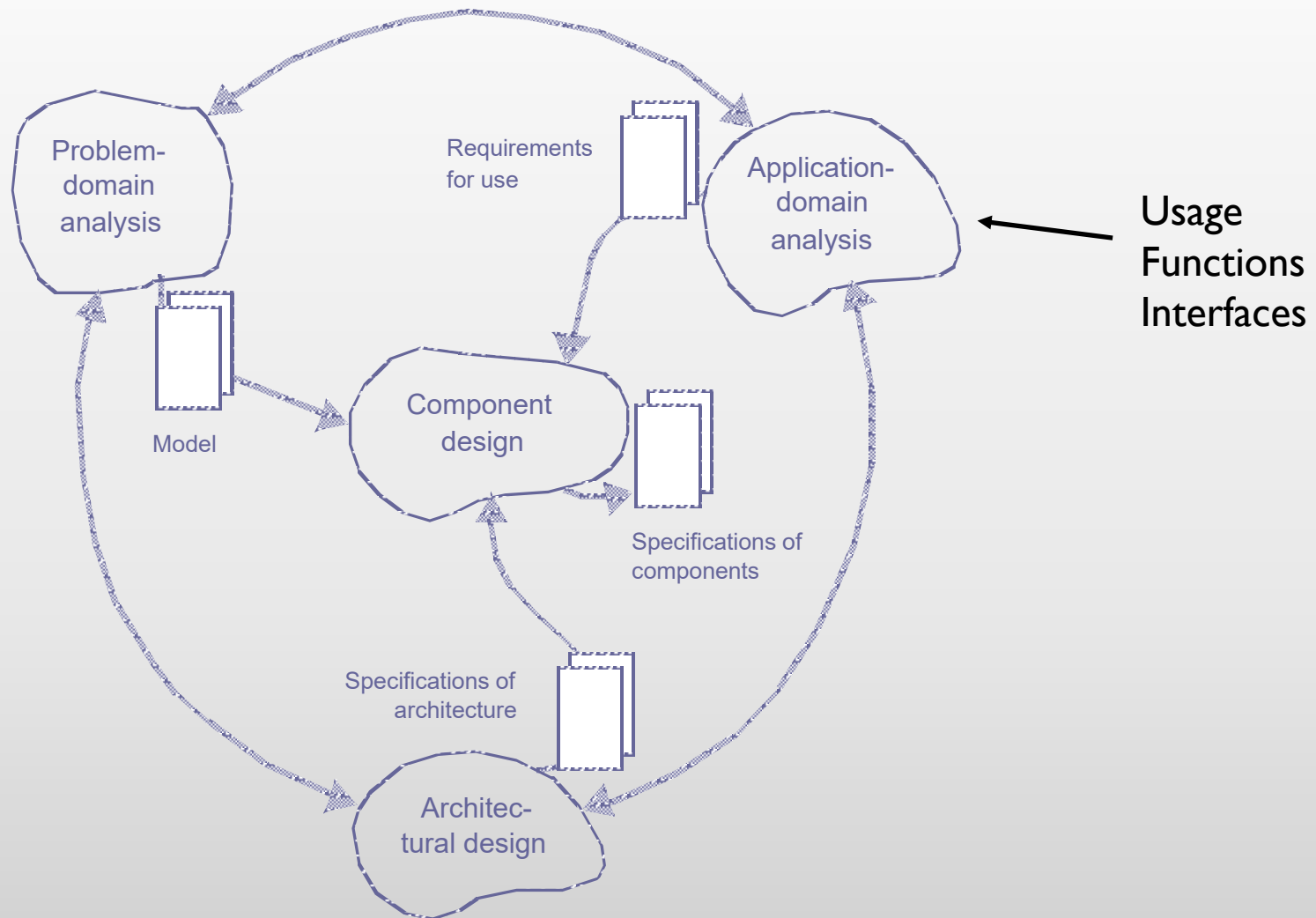
Application domain

Model



Problem domain

Application Domain Analysis: Activities

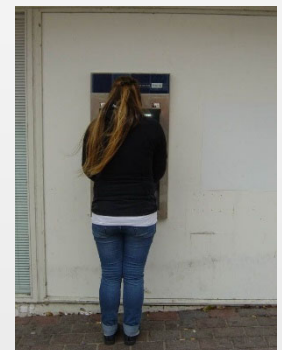


Stable versus Transient Properties

- ▶ Why did we start with analysis of the model?
- ▶ Compare the model, functions and interfaces between a classical bank and a modern internet-based bank
- ▶ Model: the same
- ▶ Functions: the same basic functions, some new ones added, e.g. mobilepay
- ▶ Interfaces: completely changed; new customer interface



1. november 1979 konverterte to af bankens filialer til IDA — Kongsgård og Våsthygd. Billedet er hentet fra slægtstevnte.



Stable properties

Transient properties



Model

Functions

Interfaces

Application Domain Analysis: Summary

Purpose	<ul style="list-style-type: none">• To determine a system's usage requirements.
Concepts	<ul style="list-style-type: none">• Application domain: An organization that administrates, monitors, or controls a problem domain.• Requirements: A system's externally observable behavior.
Principles	<ul style="list-style-type: none">• Determine the application domain with use cases.• Collaborate with users.
Results	<ul style="list-style-type: none">• A complete list of the system's overall usage requirements.

Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
- ▶ The Usage activity
 - Results
 - Key concepts
 - Activities
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

Usage: Results

Actor table

Use Cases	Actors			
	Account owner	Creditor	Administrator	Liquidity Monitor
Payment	X	X		
Cash Withdrawal	X			
Money Transfer	X	X	X	
Account information	X		X	X
Credit information		X	X	
Registration			X	
Monitoring			X	
Fault processing			X	

Actor

Account Owner

Purpose: A person who owns an account. The account owner's basic need is to be able to make payments with his plastic card.

Characteristic: The system's users include many and very different account owners

Examples: Account owner A is insecure in the use of a plastic card as a form of payment. A originally received his card because it was the only possibility...

Use case

Cash Withdrawal

Use Case: Cash withdrawal is started by the account owner, when he wishes to use his credit card to withdraw cash from an ATM. The account owner inserts his card in an ATM, and is then requested via the screen to type his PIN code. The screen will either show a polite denial, the card will be rejected from the ATM and the process will be cancelled; or the screen will show a menu requesting the account owner to choose an amount of money by typing on the ATM's keyboard. A new screen requests the account owner to approve the transaction. If the transaction is not approved the account owner is again requested to type an amount. Otherwise the use case ends by the ejection of the card, and the desired amount of money being paid.

Objects: (to be added later)

Functions: (to be added later)

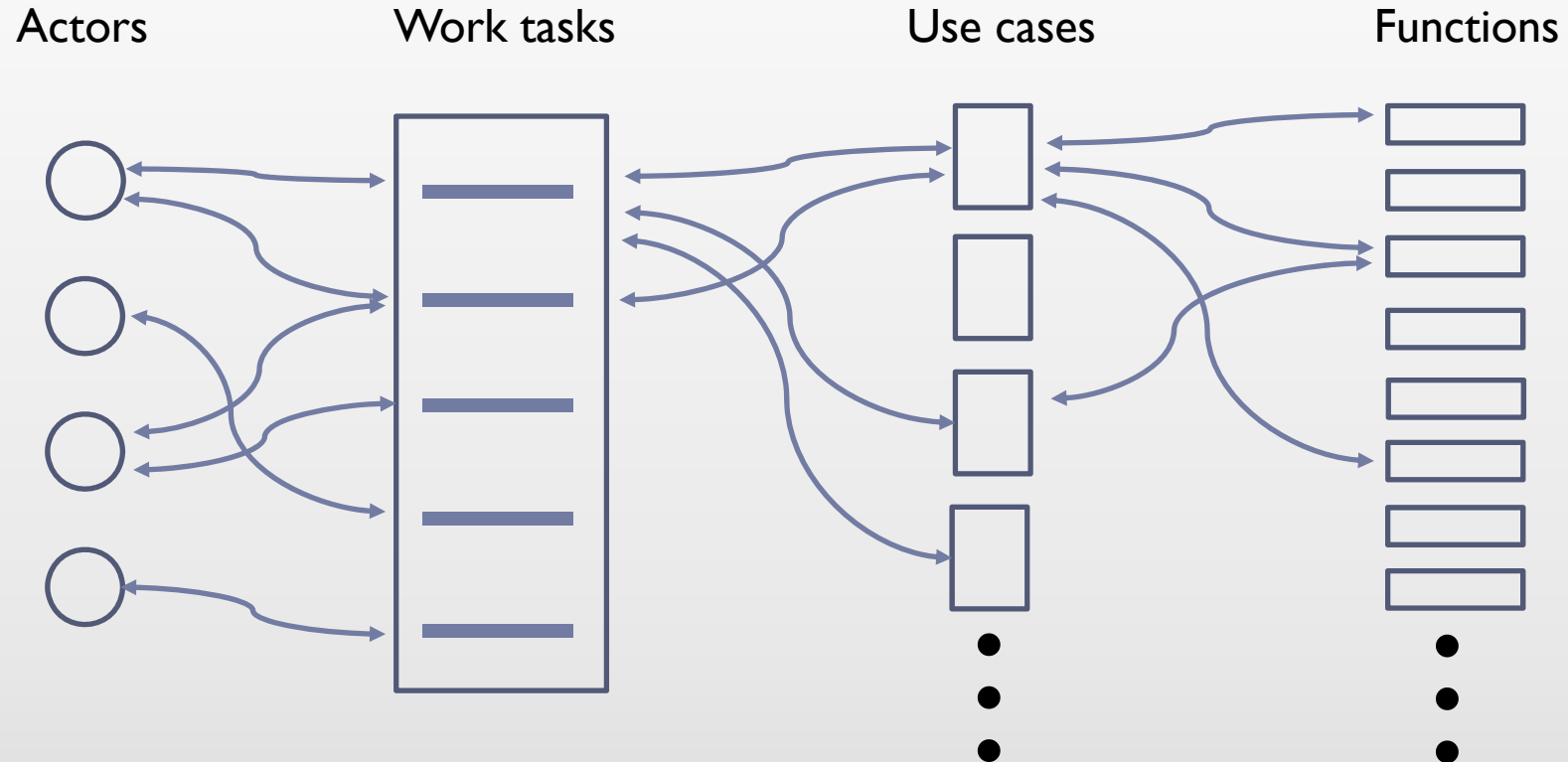
Start from Work Tasks

- ▶ What tasks exist in the application domain?
- ▶ What is the division of labor?
- ▶ How are the different tasks delimited?
- ▶ Describe the work tasks:
 - Name and content
 - Purpose
 - How is it assigned?
 - Who performs it?
 - Relationships to other tasks
 - Result

Administration System Work Tasks

- Establish new conference
- Detailed planning of conference
- Administration of participants
- Registration of person
- Administration of articles
- Information to the committees
- Information to participants, authors, and reviewers

Actors, Work Tasks, Use Cases and Functions



Key Concepts: Actor

- ▶ Identify actors
 - Determine the distribution of roles of the works tasks related to the system
 - Consider human actors
 - Consider other systems as actors
- ▶ Describe actors
 - Make actor specification

Account Owner

Purpose: A person who owns an account. The account owner's basic need is to be able to make payments with his plastic card.

Characteristic: The system's users include many and very different account owners

Examples:

Account owner A is insecure in the use of a plastic card as a form of payment. A originally received his card because it was the only possibility for getting an ID card for his checks. A only withdraws money from the ATM in emergency situations.

Account owner B is technologically curious and uses the system often, optimally, and to the limit of its abilities. B has never had major problems in understanding the possibilities of the system, and B also examines the possibilities that are not obviously available.

Key Concepts: Use Case (I)

- ▶ Identify use cases where the system is used to carry out part of a work task
- ▶ Describe use cases
 - As text
 - As a statechart diagram

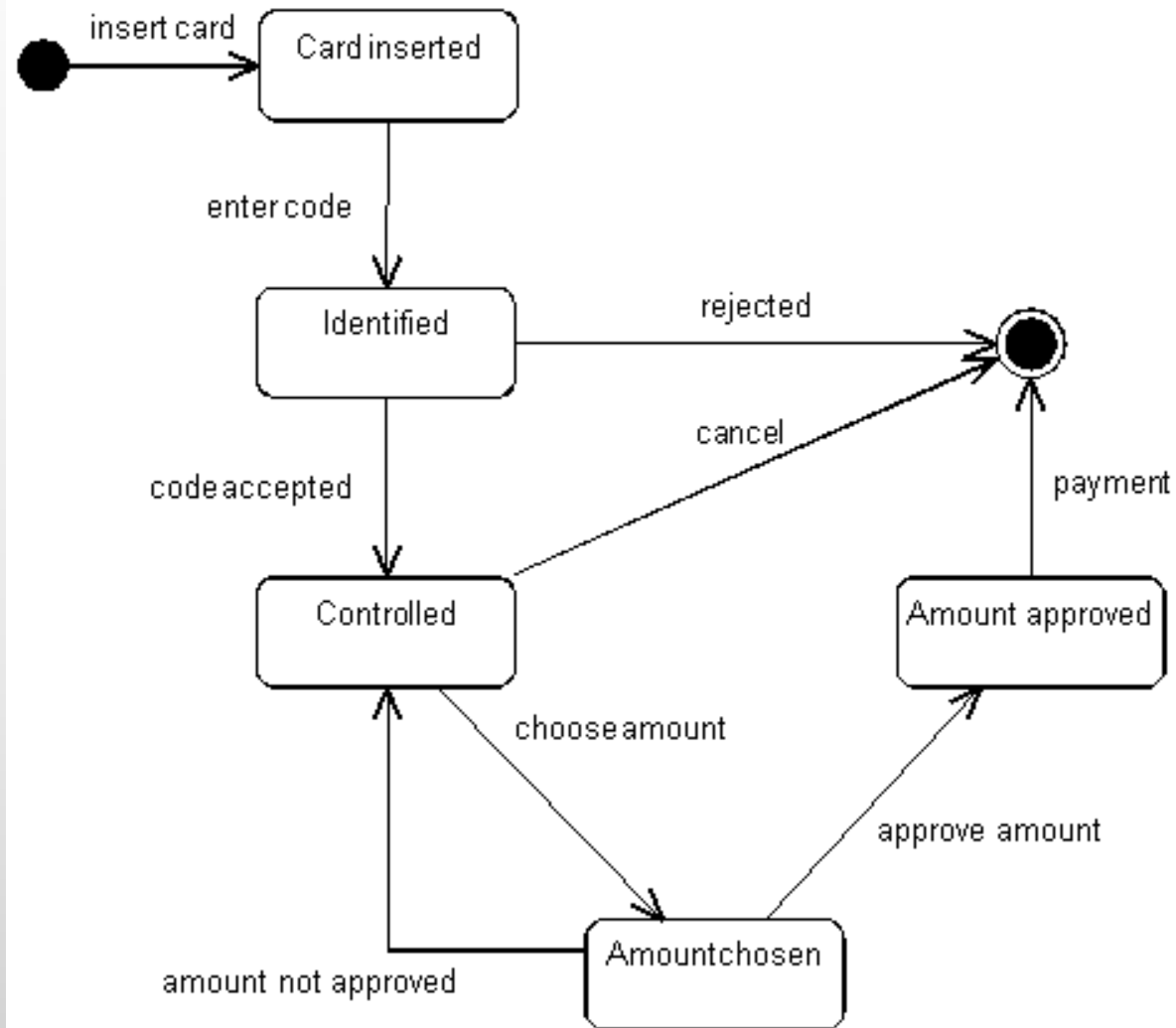
Cash Withdrawal

Use Case: Cash withdrawal is started by the account owner, when he wishes to use his credit card to withdraw cash from an ATM. The account owner inserts his card in an ATM, and is then requested via the screen to type his PIN code. The screen will either show a polite denial, the card will be rejected from the ATM and the process will be cancelled; or the screen will show a menu requesting the account owner to choose an amount of money by typing on the ATM's keyboard. A new screen requests the account owner to approve the transaction. If the transaction is not approved the account owner is again requested to type an amount. Otherwise the use case ends by the ejection of the card, and the desired amount of money being paid.

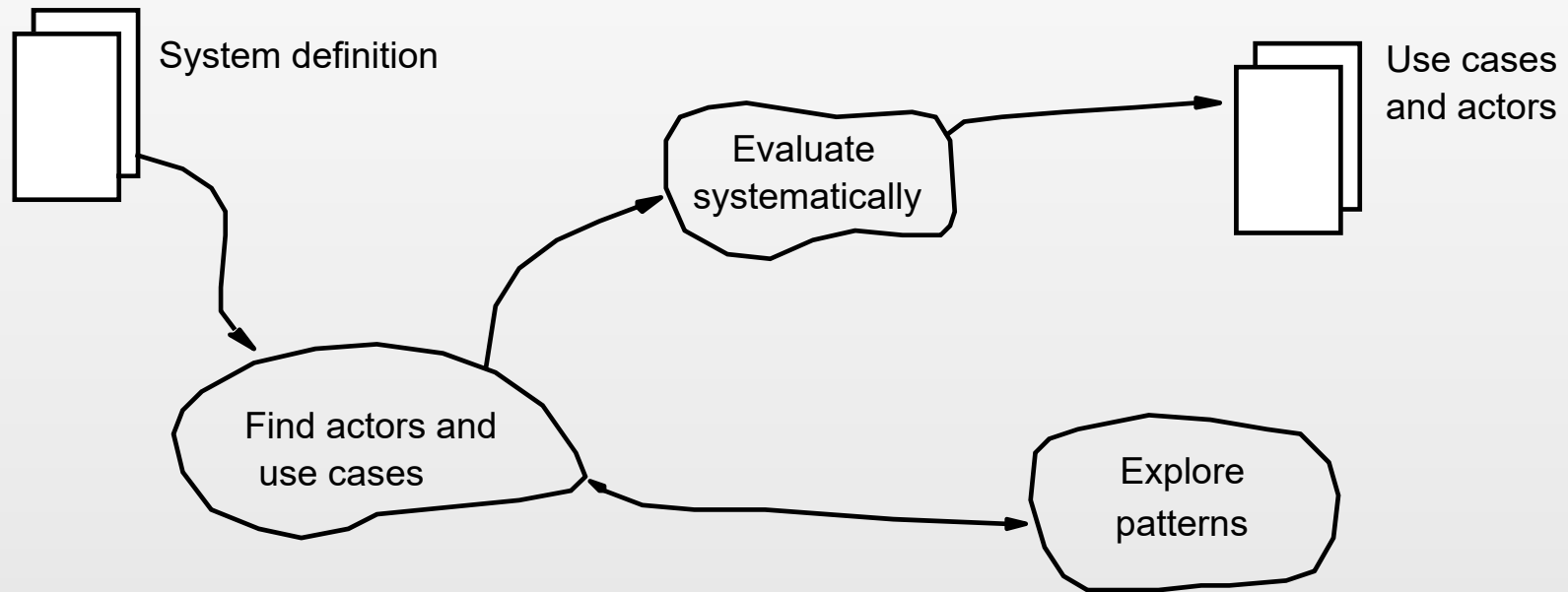
Objects: (to be added later)

Functions: (to be added later)

Key Concepts: Use Cases (2)



Usage: Activities



Evaluate Systematically

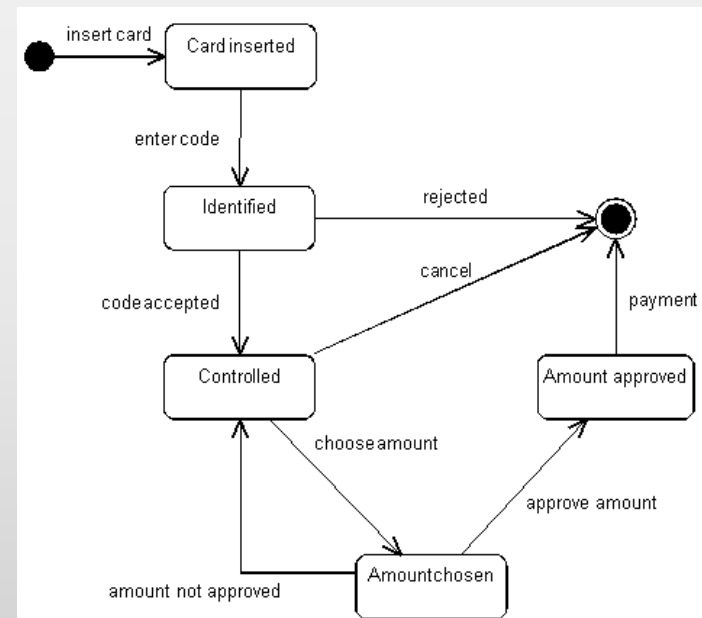
► Systematic review

- Use cases should be simple and constitute a coherent whole
- The description of actors and use cases should provide understanding and overview
- Use cases should be described in enough detail to enable identification of functions and interface elements

► What could be functions and interface elements for Cash Withdrawal?

► Experiment with prototypes

- A use case is best evaluated through planned prototype experiments



Usage: Summary

Purpose	<ul style="list-style-type: none">• To determine how actors interact with a system.
Concepts	<ul style="list-style-type: none">• Actor: An abstraction of users or other systems that interact with the target system.• Use case: A pattern for interaction between the system and actors in the application domain.
Principles	<ul style="list-style-type: none">• Determine the application domain with use cases.• Evaluate use cases in collaboration with users.• Assess social changes in the application domain.
Results	<ul style="list-style-type: none">• Descriptions of all use cases and actors.

Contents

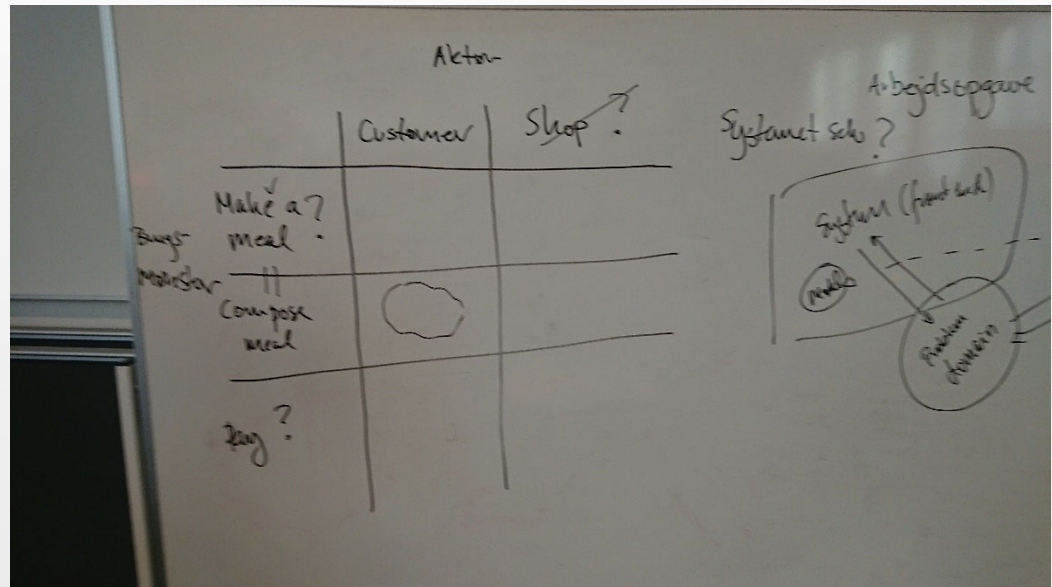
- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
- ▶ The Usage activity
- ▶ **Example: street food and streaming**
- ▶ Explore patterns
- ▶ Challenges in this activity

Street Food

- 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. Secondly, it is a communication medium that customers use to request delivery of orders from the food shops and bars.

Street Food Usage

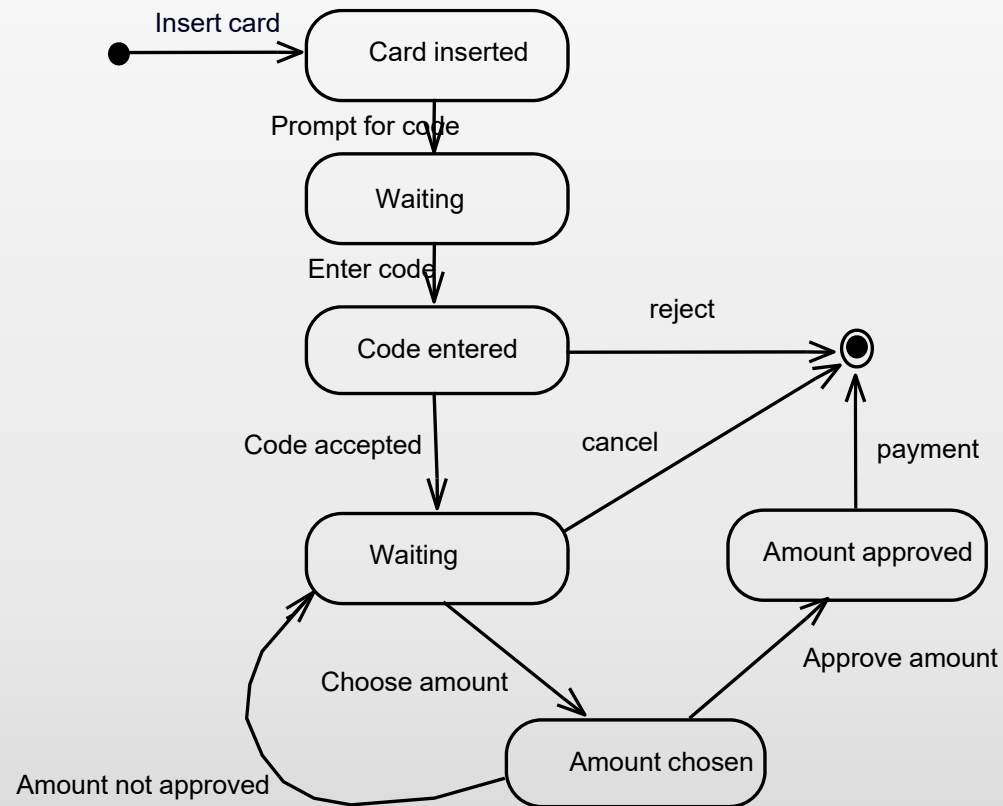
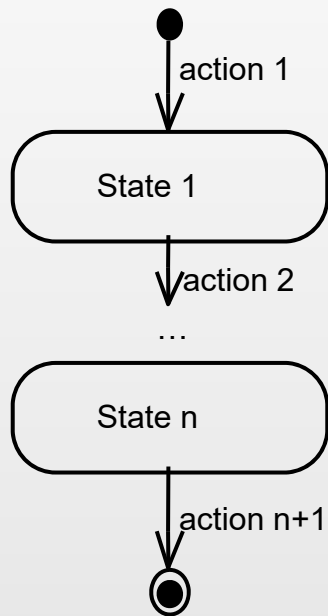
- ▶ Name actors
- ▶ Name use cases
- ▶ Compile an actor table
- ▶ Define one actor
- ▶ Define one use case



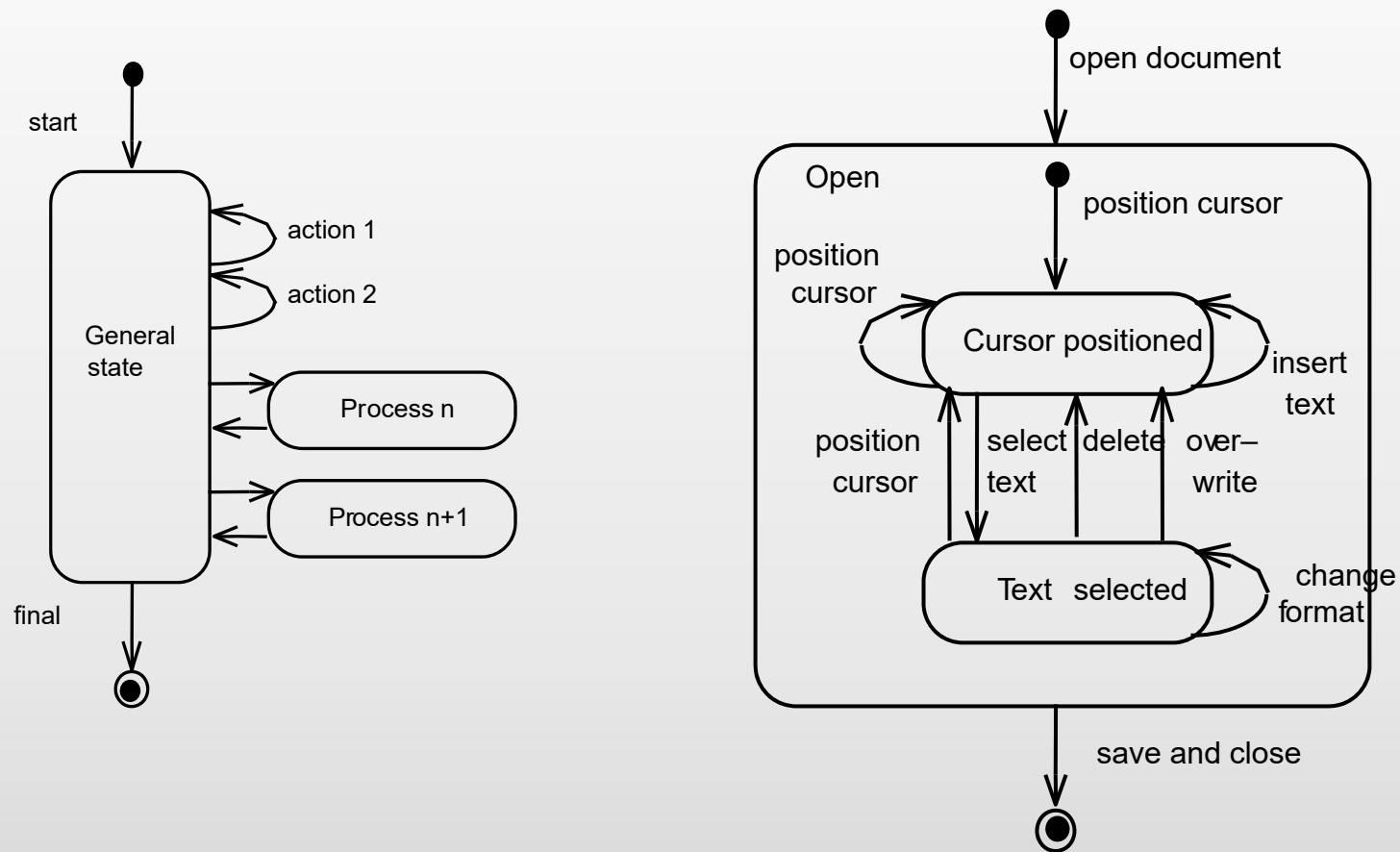
Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
- ▶ The Usage activity
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

Explore Patterns: Procedural



Explore Patterns: Material



Contents

- ▶ Exam sets from previous exams
- ▶ Summary of last lecture
- ▶ Application Domain analysis
- ▶ The Usage activity
- ▶ Example: street food and streaming
- ▶ Explore patterns
- ▶ Challenges in this activity

Appreciate the Difference: Actors and Classes

	Application domain	Problem domain
Static	Actors	Classes and Structure
Dynamic	Use cases	Behavioral patterns

Work in Exercises for this Activity

- ▶ Start from work tasks (from A in system definition)
- ▶ For each work task, find the key actors
- ▶ Describe key use cases for these actors
- ▶ Complete the actors
- ▶ Complete the use cases
- ▶ Compile a complete actor table (overview)

Use Cases	Actors			
	Account owner	Creditor	Administrator	Liquidity Monitor
Payment	X	X		
Cash Withdrawal	X			
Money Transfer	X	X	X	
Account information	X		X	X
Credit information		X	X	
Registration			X	
Monitoring			X	
Fault processing			X	