Dasar Pengembangan Sistem Informasi

System Modeling

Tim Pengampu Mata Kuliah
Dasar Pengembangan Sistem Informasi

Topics covered

- Context models
- Interaction models
- Structural models
- Behavioral models

System Modeling

- System modeling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system.
- System modeling may represent a system using graphical notation, e.g. the Unified Modeling Language (UML).
- System modelling helps the analyst
  - understand the functionality of the system
  - models are used to communicate with customers.

System Models Purposes

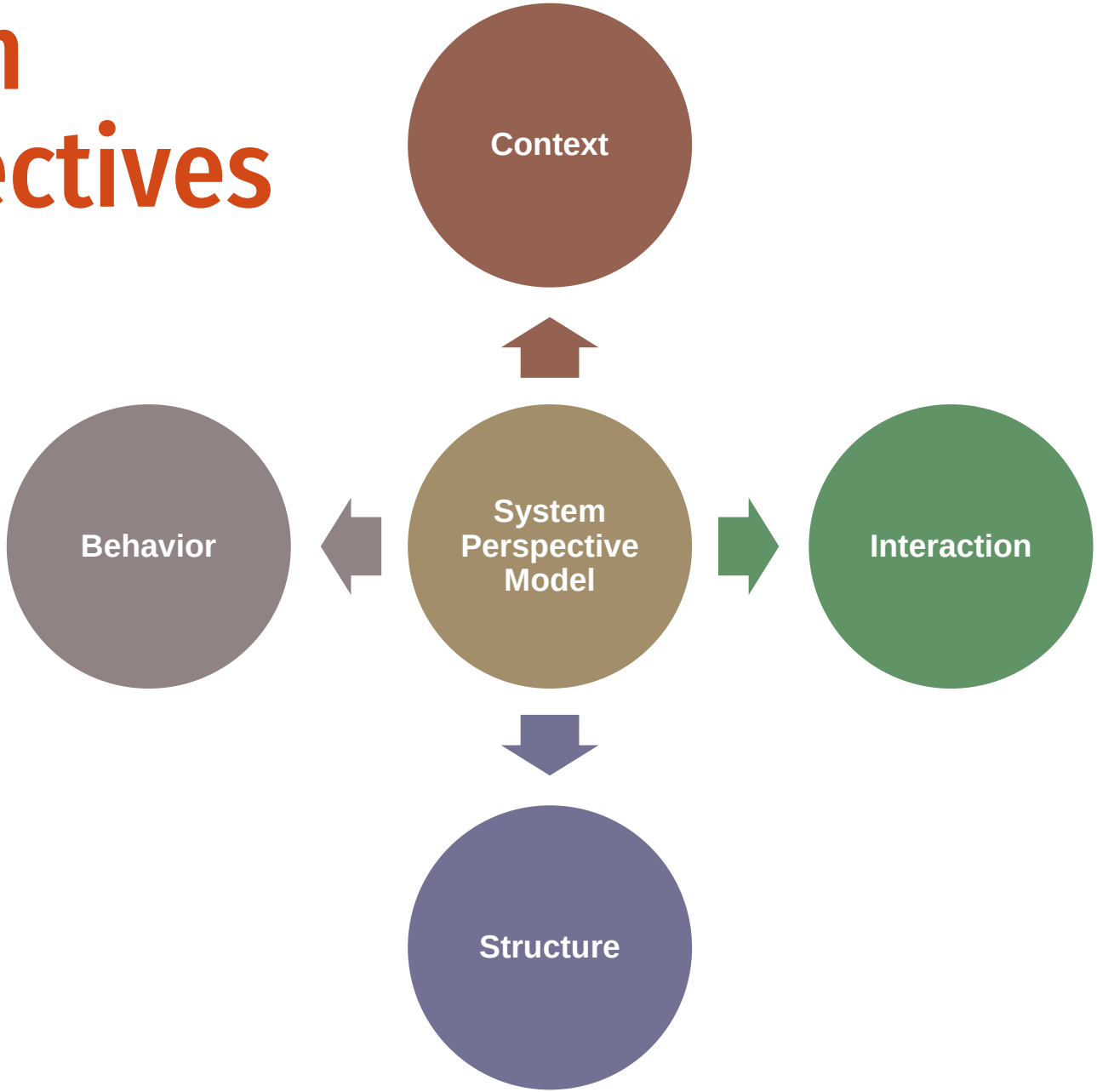
Existing System

- They help clarify what the existing system does and can be
- Used as a basis for discussing its strengths and weaknesses.
- These then lead to requirements for the new system

Proposed System

- help explain the proposed requirements to other system stakeholders.
- discuss design proposals
- to document the system for implementation

System Perspectives Model



System Perspectives

- An context/external perspective
  - model the context or environment of the system.
- An interaction perspective
  - model the interactions between a system and its environment, or between the components of a system.
- A structural perspective
  - model the organization of a system or the structure of the data that is processed by the system.
- A behavioral perspective
  - model the dynamic behavior of the system and how it responds to events.

Why Use of Graphical Models?

- As a means of facilitating discussion about an existing or proposed system
  - Incomplete and incorrect models are OK as their role is to support discussion.
- As a way of documenting an existing system
  - Models should be an accurate representation of the system but need not be complete.
- As a detailed system description that can be used to generate a system implementation
  - Models have to be both correct and complete.

Object-Oriented Model using UML

- Activity diagrams, show the activities involved in a process or in data processing .
- Use case diagrams, show the interactions between a system and its environment.
- Sequence diagrams, which show interactions between actors and the system and between system components.
- Class diagrams, which show the object classes in the system and the associations between these classes.
- State diagrams, which show how the system reacts to internal and external events.

Structured Approach Diagram Types

- Data Flow Diagram, shows the data flow and processes inside the system.
- ER Diagram, shows data objects with their relationships
- State Transitions Diagrams, which show how the system reacts to internal and external events.

Context Models

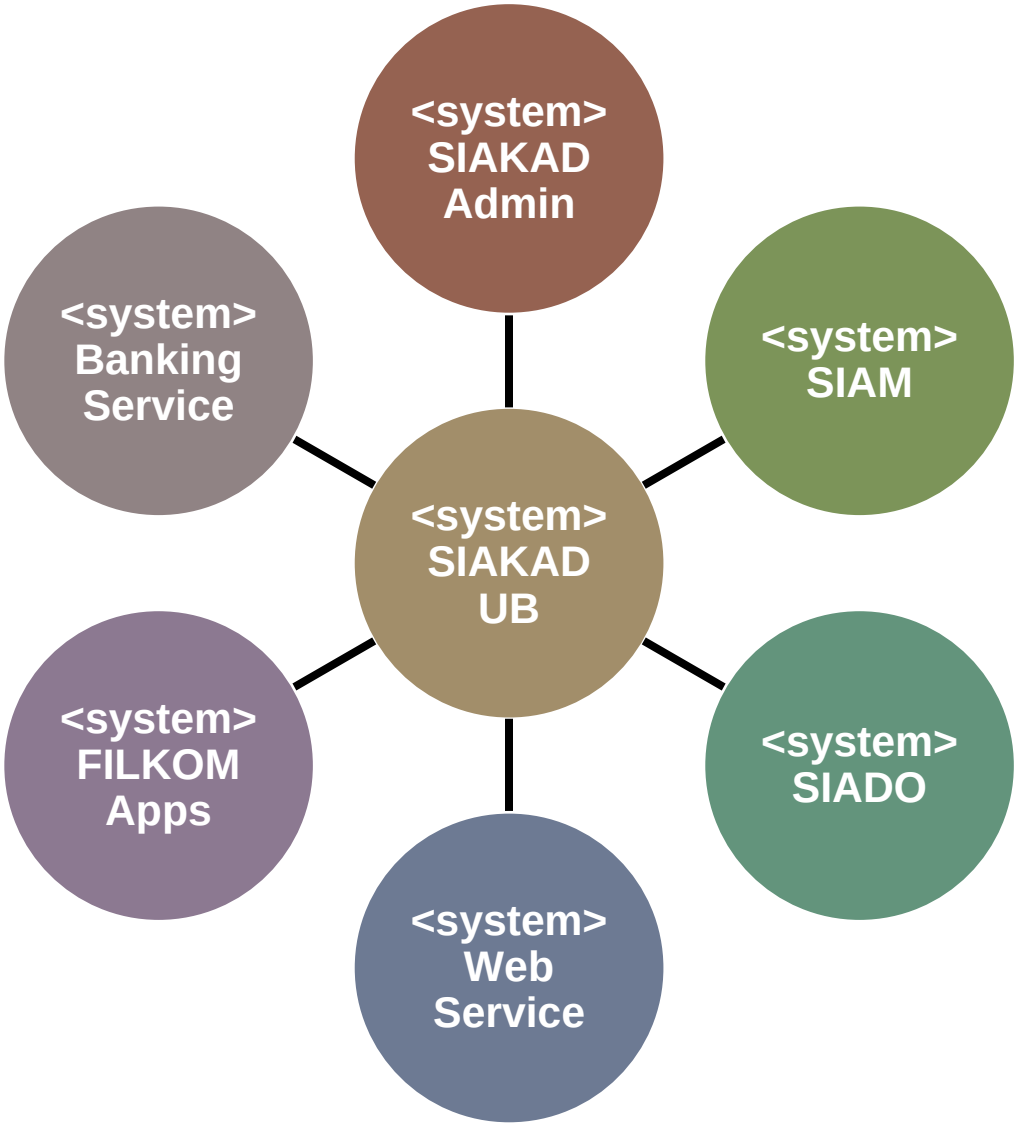
Context models

- Context models are used to illustrate the operational context of a system - they show what lies outside the system boundaries.
- Social and organisational concerns may affect the decision on where to position system boundaries.
- Architectural models show the system and its relationship with other systems.

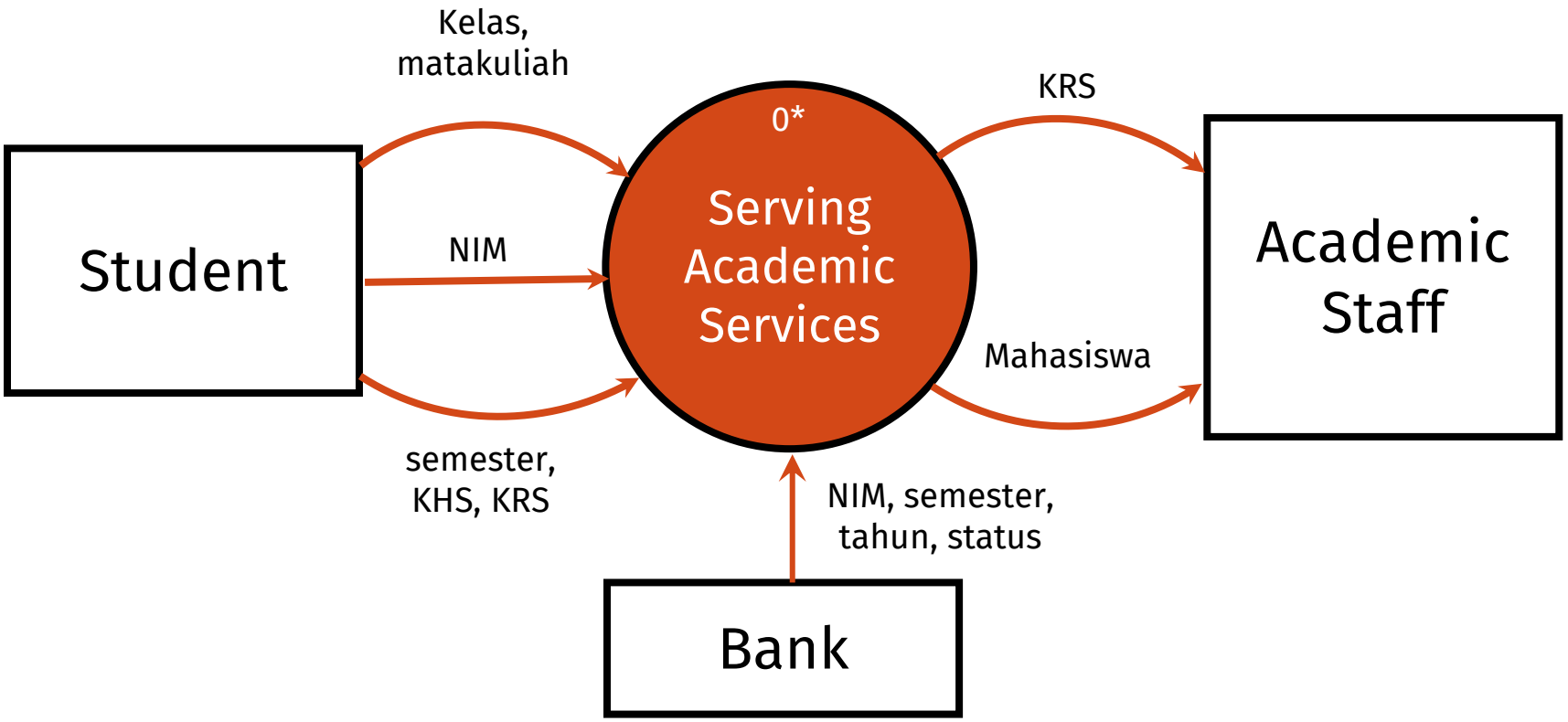
System boundaries

- System boundaries are established to define what is inside and what is outside the system.
  - They show other systems that are used or depend on the system being developed.
- The position of the system boundary has a profound effect on the system requirements.
- Defining a system boundary is a political judgment
  - There may be pressures to develop system boundaries that increase / decrease the influence or workload of different parts of an organization.

Example: Context of SIAKAD UB

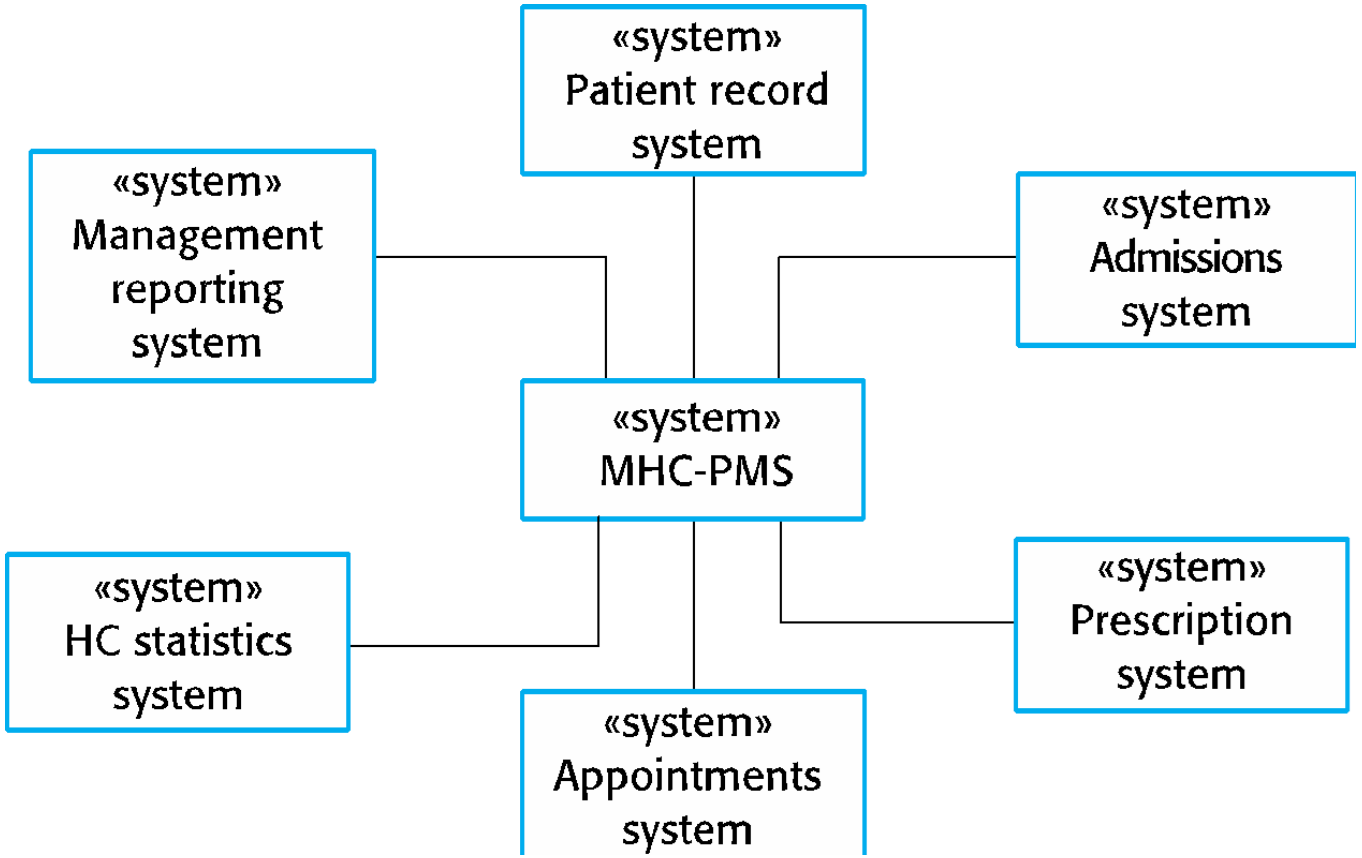


Example: Context Diagram



The Context of the MHC-PMS

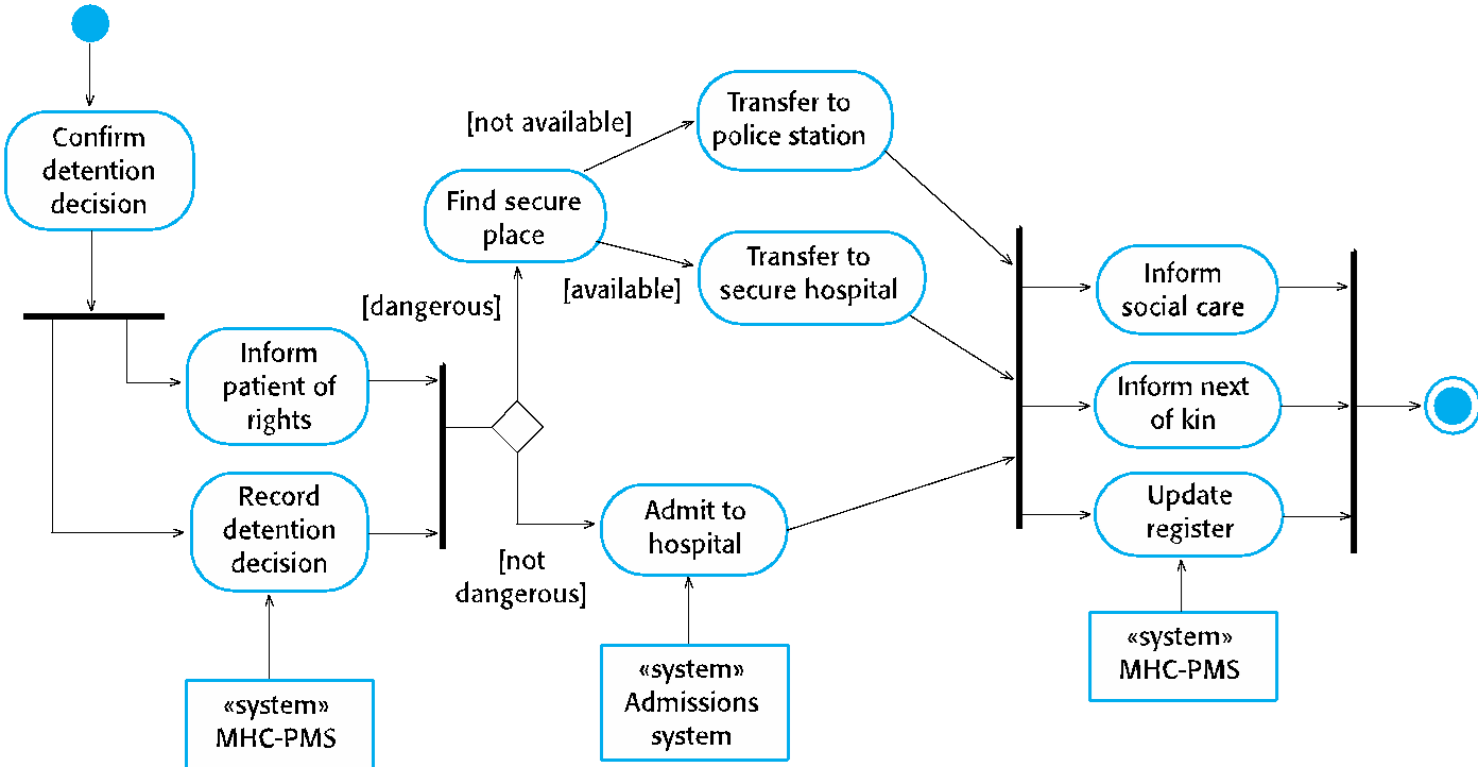
Mental Health Care – Patient Management Systems



Process perspective

- Context models simply show the other systems in the environment, not how the system being developed is used in that environment.
- Process models reveal how the system being developed is used in broader business processes.
- UML activity diagrams may be used to define business process models.

Process model of involuntary detention



Activity Diagram

FILKOM
FAKULTAS ILMU KOMPUTER

Interaction Models

Interaction models

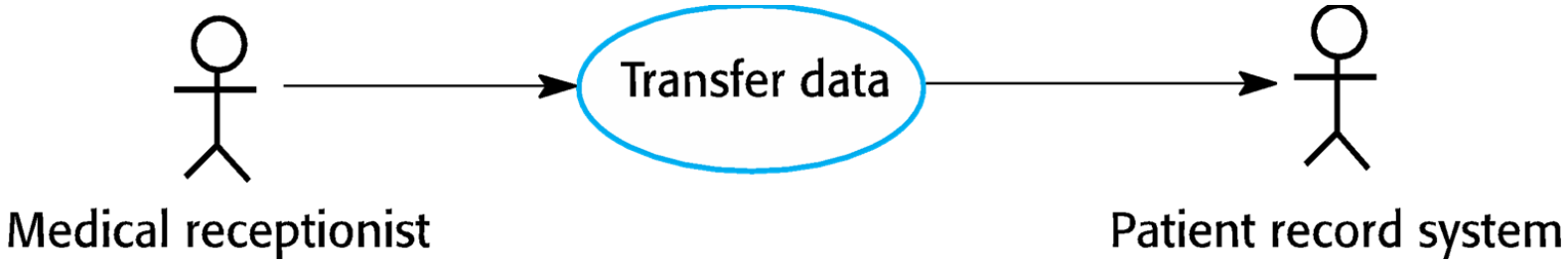
- Modeling user interaction is important as it helps to identify user requirements.
- Modeling system-to-system interaction highlights the communication problems that may arise.
- Modeling component interaction helps us understand if a proposed system structure is likely to deliver the required system performance and dependability.
- Use case diagrams and sequence diagrams may be used for interaction modelling.

Use Case Modeling

- Use cases were developed originally to support requirements elicitation and now incorporated into the UML.
- Each use case represents a discrete task that involves external interaction with a system.
- Actors in a use case may be people or other systems.
- Represented diagramatically to provide an overview of the use case and in a more detailed textual form.

Transfer-data use case

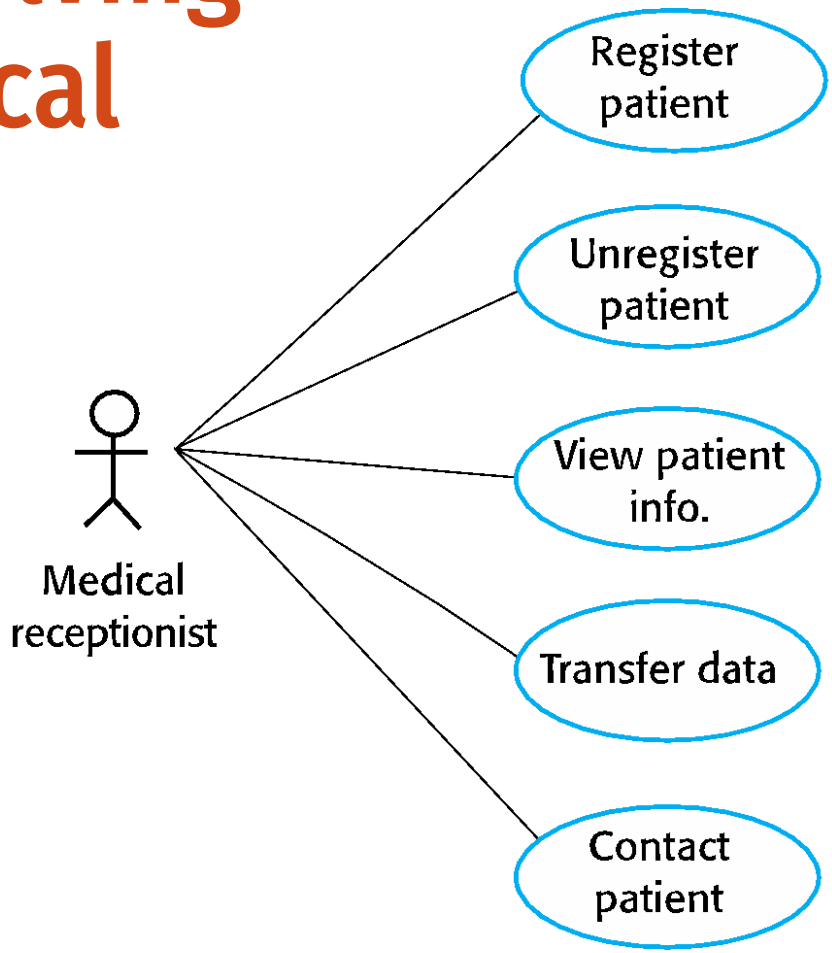
- A use case in the MHC-PMS



Tabular description of the ‘Transfer data’ use-case

MHC-PMS: Transfer data	
Actors	Medical receptionist, patient records system (PRS)
Description	A receptionist may transfer data from the MHC-PMS to a general patient record database that is maintained by a health authority. The information transferred may either be updated personal information (address, phone number, etc.) or a summary of the patient’s diagnosis and treatment.
Data	Patient’s personal information, treatment summary
Stimulus	User command issued by medical receptionist
Response	Confirmation that PRS has been updated
Comments	The receptionist must have appropriate security permissions to access the patient information and the PRS.

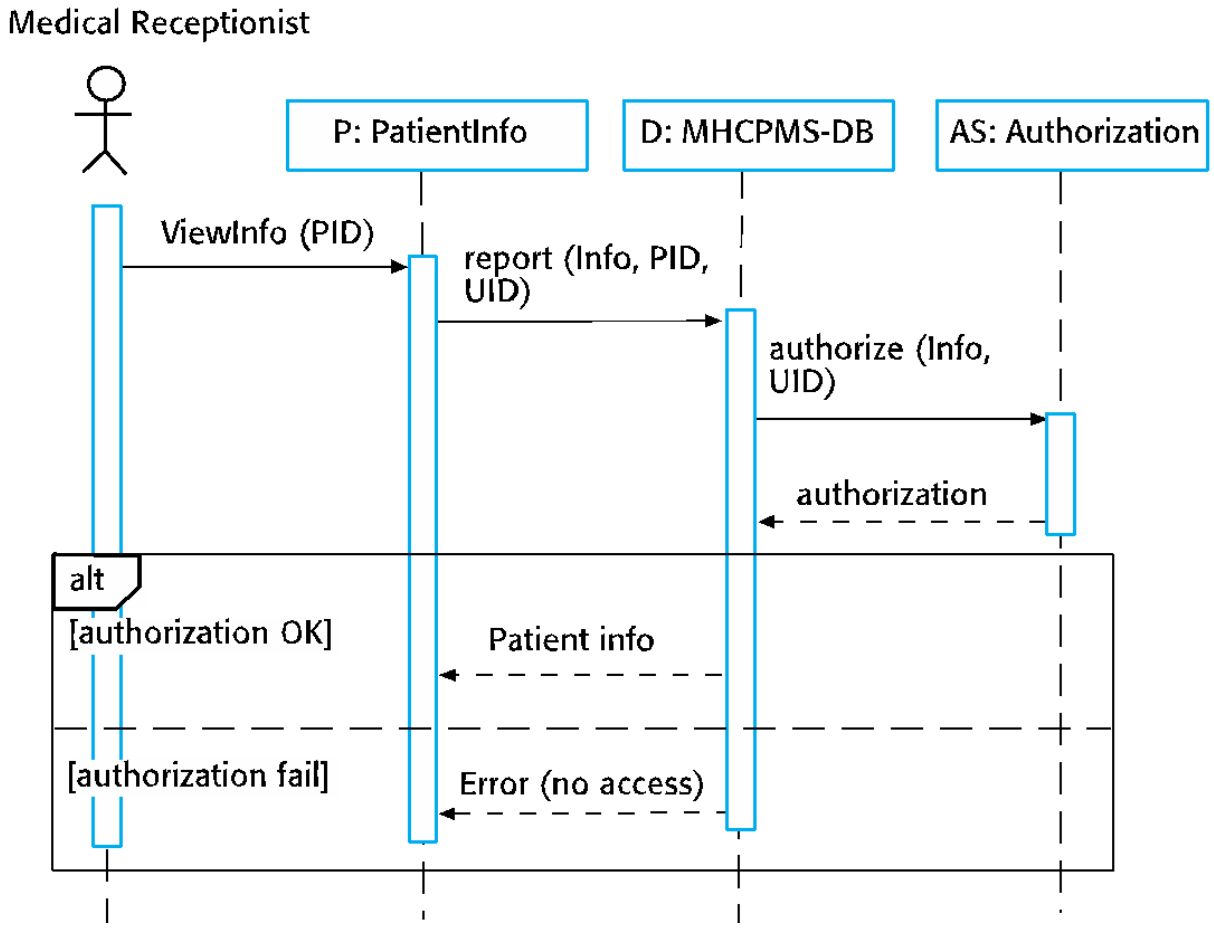
Use cases in the MHC-PMS involving the role 'Medical Receptionist'



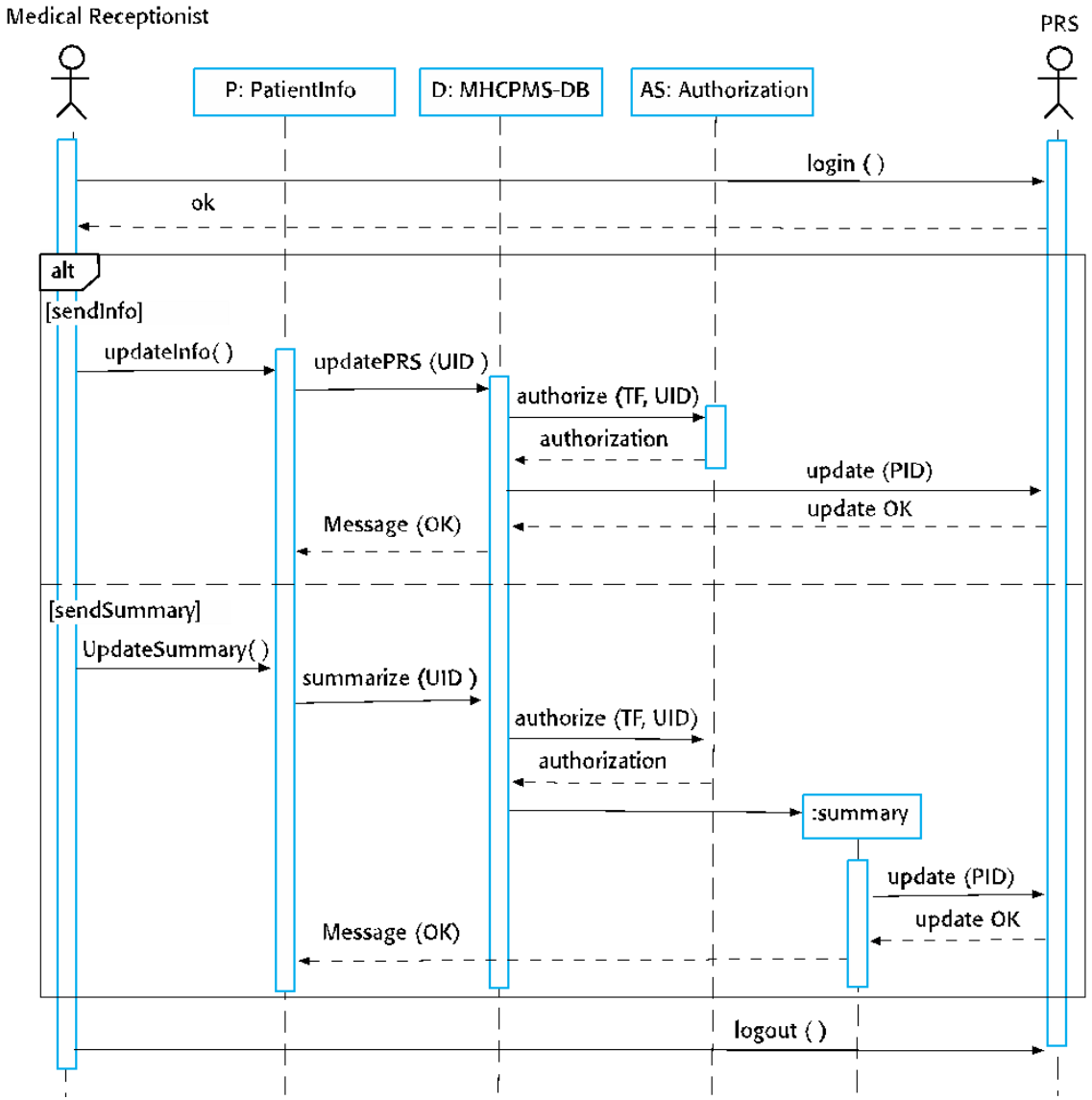
Sequence diagrams

- Sequence diagrams are part of the UML and are used to model the interactions between the actors and the objects within a system.
- A sequence diagram shows the sequence of interactions that take place during a particular use case or use case instance.
- The objects and actors involved are listed along the top of the diagram, with a dotted line drawn vertically from these.
- Interactions between objects are indicated by annotated arrows.

Sequence diagram for View patient information



Sequence Diagram for Transfer Data



Structural Models

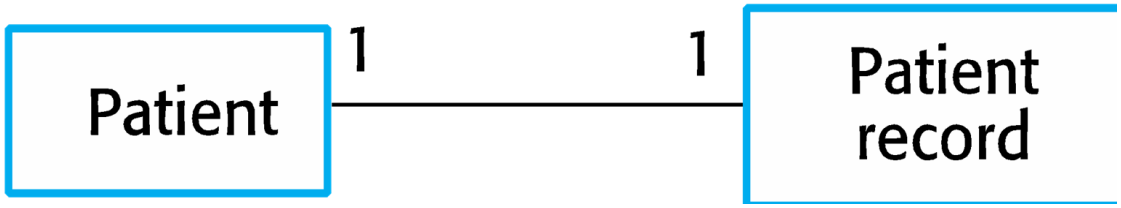
Structural models

- Structural models of software display the organization of a system in terms of the components that make up that system and their relationships.
- Structural models may be static models, which show the structure of the system design, or dynamic models, which show the organization of the system when it is executing.
- You create structural models of a system when you are discussing and designing the system architecture.

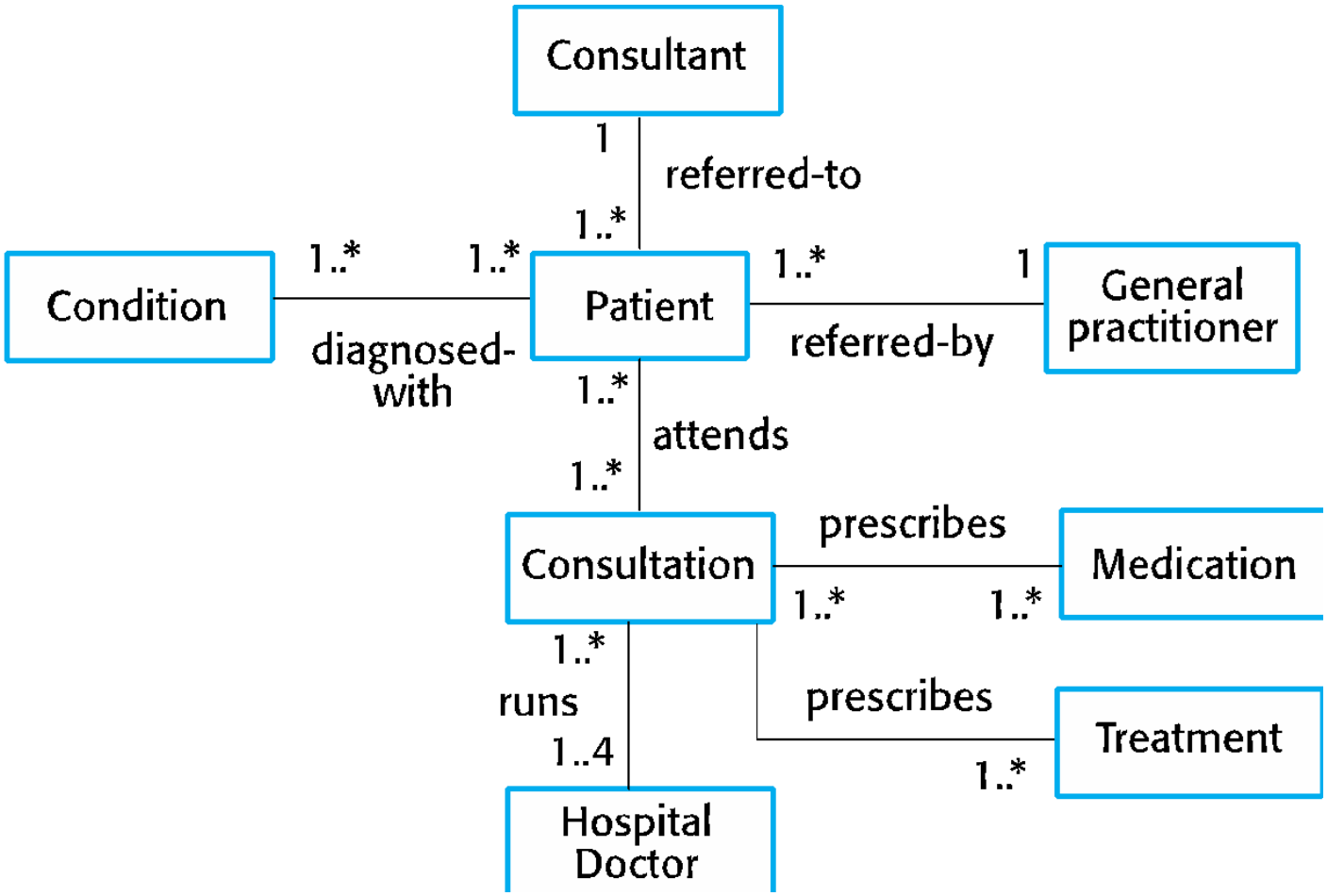
Class diagrams

- Class diagrams are used when developing an object-oriented system model to show the classes in a system and the associations between these classes.
- An object class can be thought of as a general definition of one kind of system object.
- An association is a link between classes that indicates that there is some relationship between these classes.
- When you are developing models during the early stages of the software engineering process, objects represent something in the real world, such as a patient, a prescription, doctor, etc.

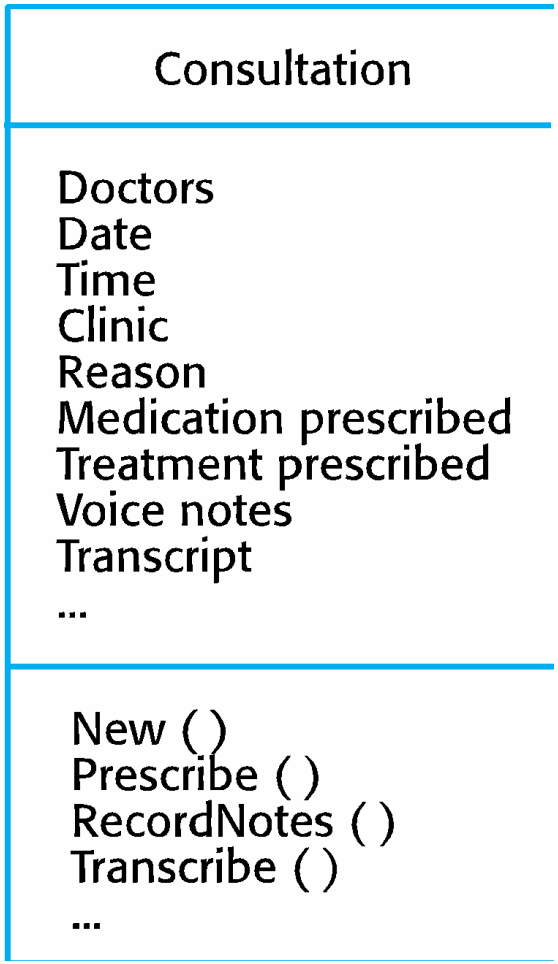
UML classes and association



Classes and associations in the MHC-PMS



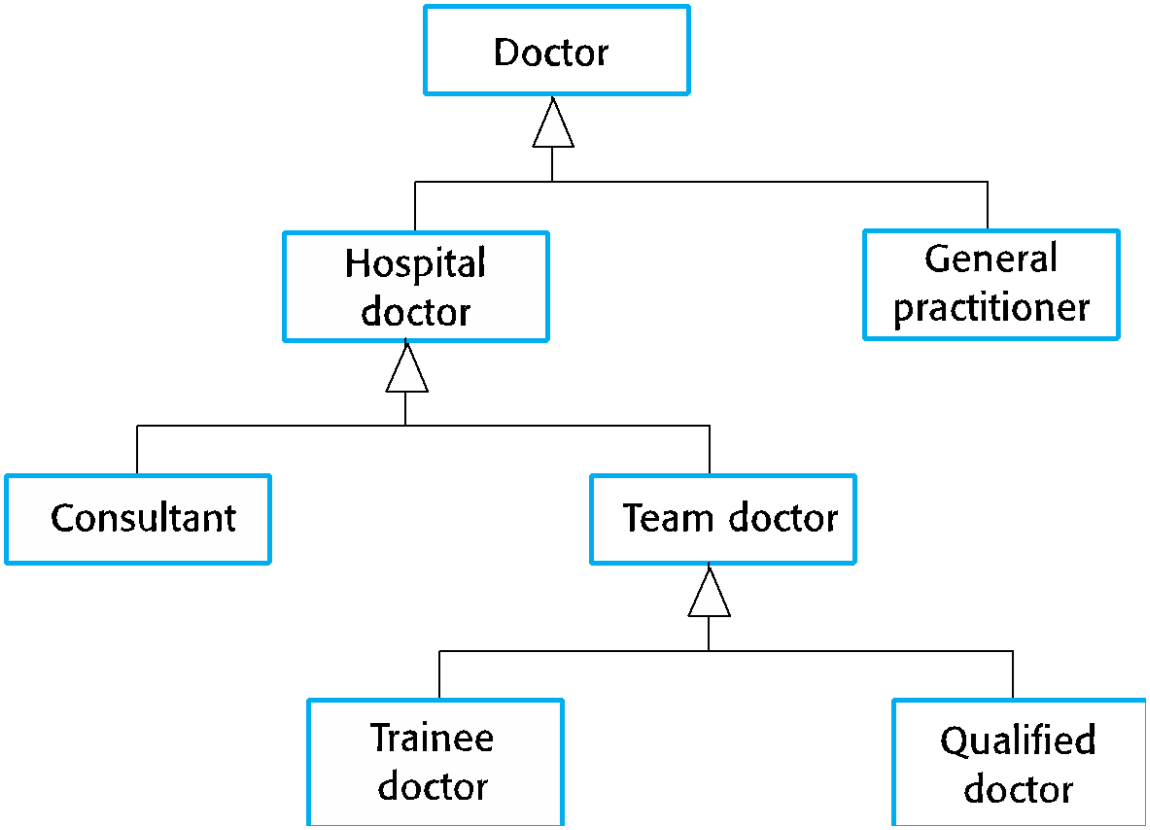
The Consultation class



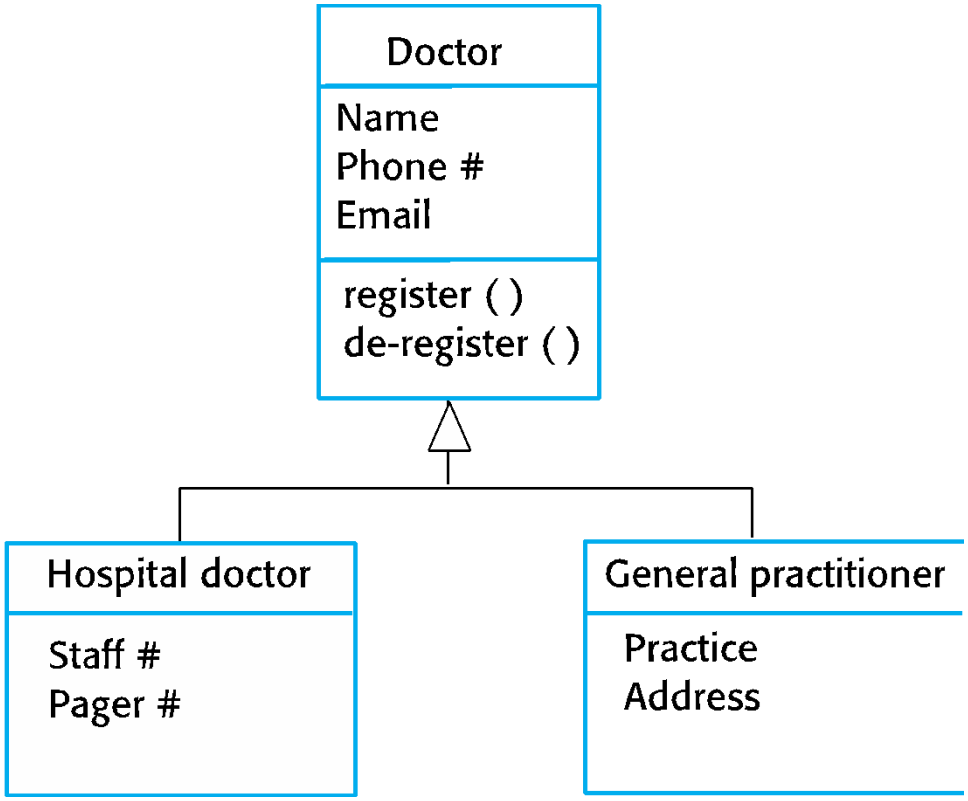
Generalization

- Generalization is an everyday technique that we use to manage complexity.
- In a generalization, the attributes and operations associated with higher-level classes are also associated with the lower-level classes.
- The lower-level classes are subclasses inherit the attributes and operations from their super-classes. These lower-level classes then add more specific attributes and operations.

A Generalization Hierarchy



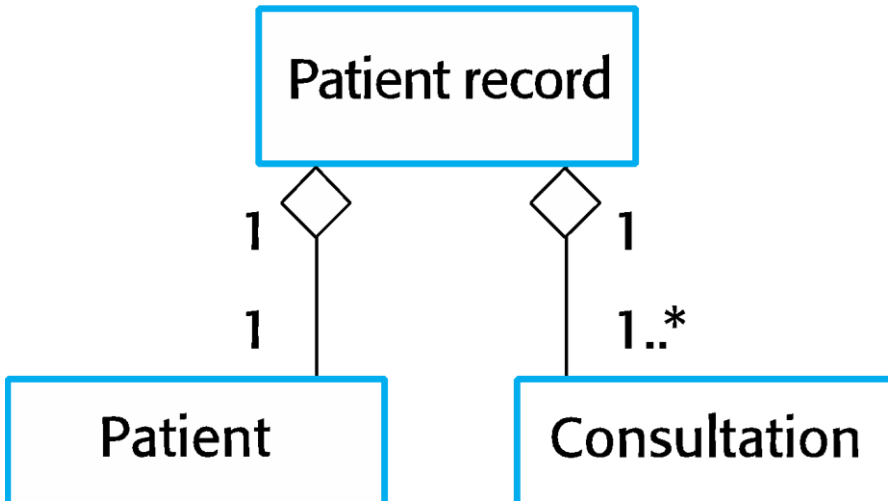
A generalization hierarchy with added detail



Object class aggregation models

- An aggregation model shows how classes that are collections are composed of other classes.
- Aggregation models are similar to the part-of relationship in semantic data models.

The Aggregation Association



Behavioral Models

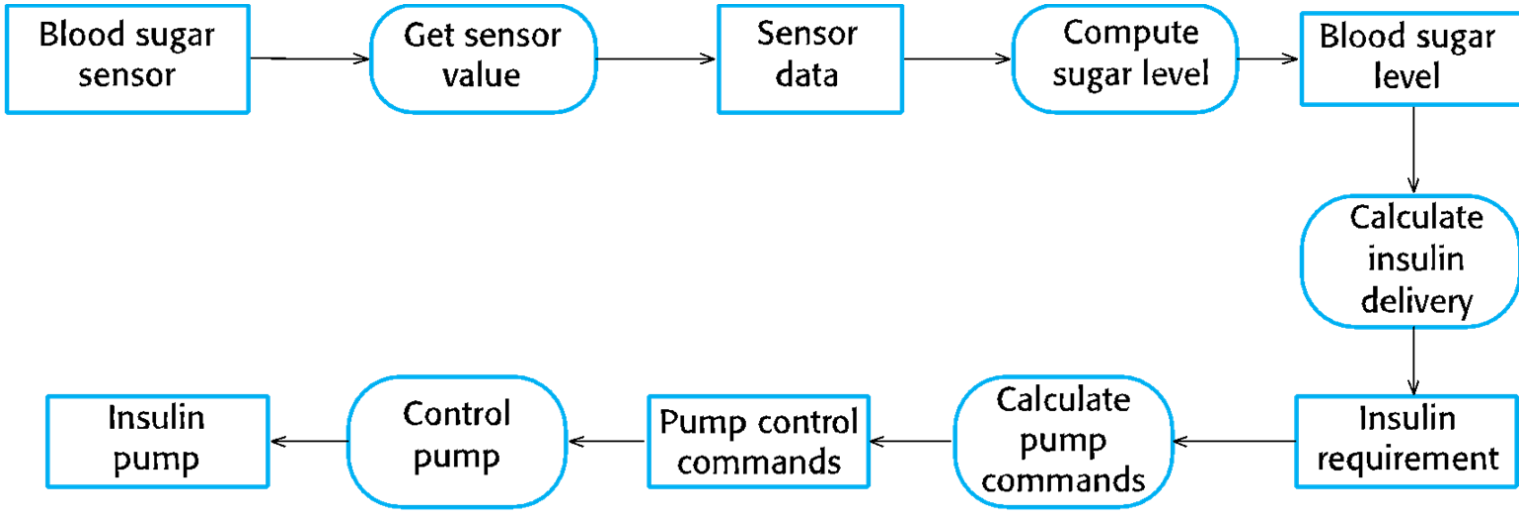
Behavioral models

- Behavioral models are models of the dynamic behavior of a system as it is executing.
- Showing what happens or what is supposed to happen when a system responds to a stimulus from its environment.
- You can think of these stimuli as being of two types:
  - Data Some data arrives that has to be processed by the system.
  - Events Some event happens that triggers system processing. Events may have associated data, although this is not always the case.

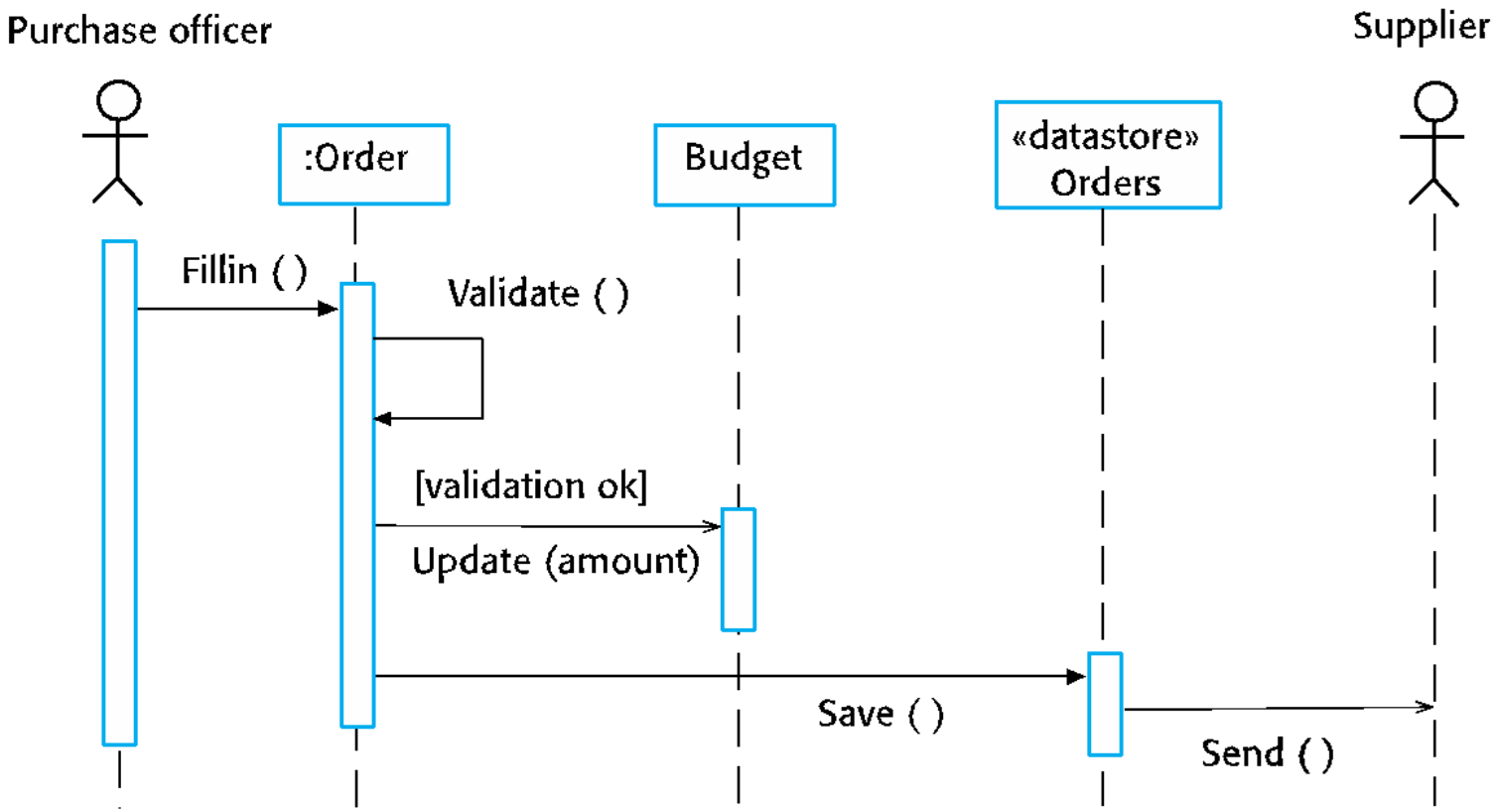
Data-driven modeling

- Many business systems are data-processing systems that are primarily driven by data. They are controlled by the data input to the system, with relatively little external event processing.
- Data-driven models show the sequence of actions involved in processing input data and generating an associated output.
- They are particularly useful during the analysis of requirements as they can be used to show end-to-end processing in a system.

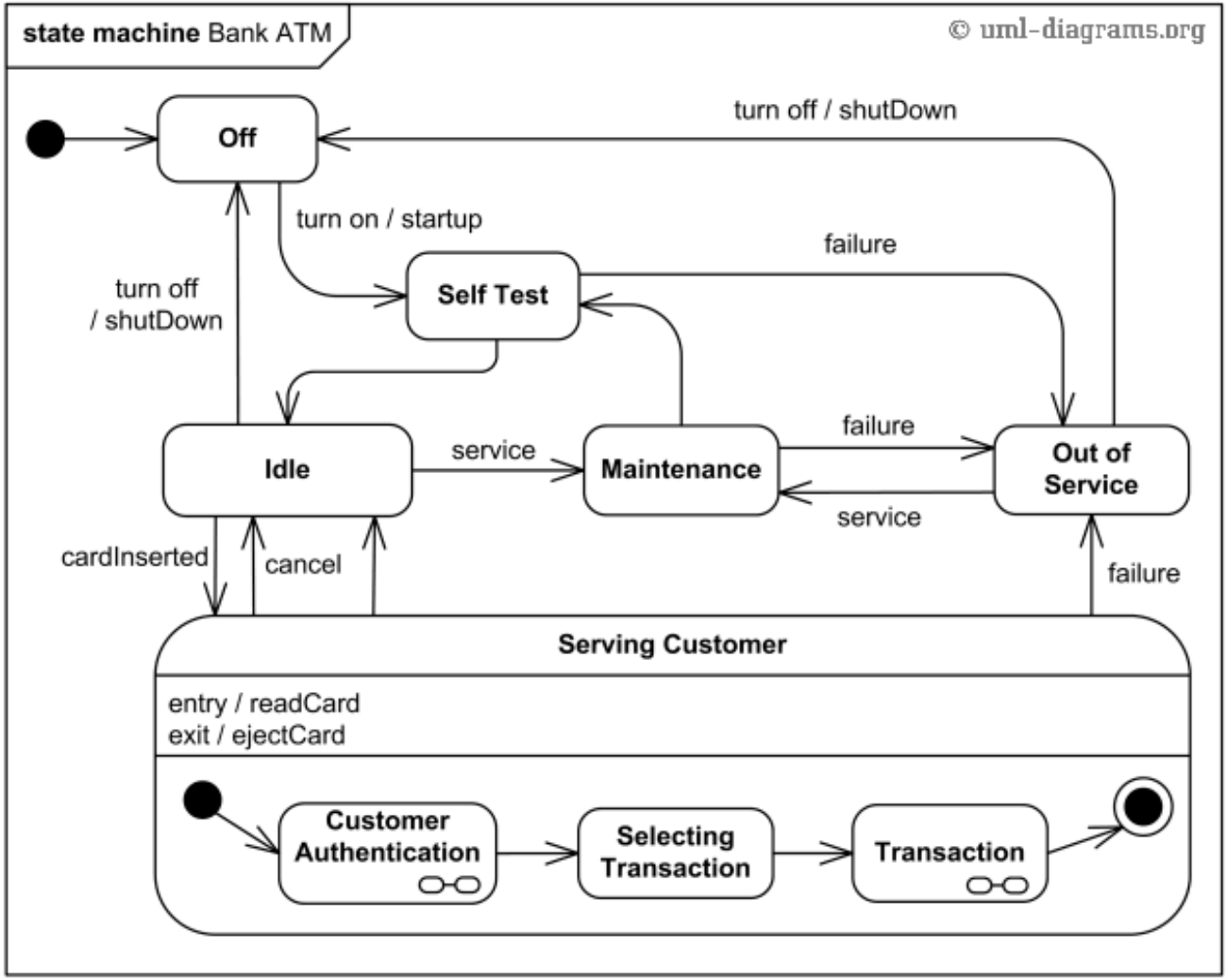
An activity model of the insulin pump's operation



Order processing



ATM State Diagram



Key points

- Behavioral models are used to describe the dynamic behavior of an executing system. This behavior can be modeled from the perspective of the data processed by the system, or by the events that stimulate responses from a system.
- Activity diagrams may be used to model the processing of data, where each activity represents one process step.
- State diagrams are used to model a system's behavior in response to internal or external events.

Questions?