Strategic dependency model (2)

- **Dependee**
  - Actor who is depended upon on a dependency relationship.

- **Depender**
  - The depending actor on a dependency relationship.

- **Dependum**
  - Element around which a dependency relationship centers.
KAOS
Constructs of Operation model

- **Operation**
  - An *input/output* relation over *objects*
  - Define state transition

- **Operationalisation**
  - Relationship between goal and operation

- **Performs**
  - Agent performs operations
Use case templates (3)
(Wiegers, 2004)

<table>
<thead>
<tr>
<th>Use Case ID:</th>
<th>Use Case Name:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created By:</td>
<td>Last Updated By:</td>
</tr>
<tr>
<td>Date Created:</td>
<td>Date Last Updated:</td>
</tr>
</tbody>
</table>

- **Actors**: a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks

- **Description**: the reason for and outcome of this use case, the sequence of actions and the outcome of executing the use case

- **Trigger**: the event that initiates the use case

- **Pre-condition**: list any activities that must take place, or any conditions that must be true, before the use case can be started

- **Post-condition**: the state of the system at the conclusion of the use case execution
Levels of authority

- **Top management**
  - establishes goals
  - does long-range planning
  - determines new market & product developments
  - decides on mergers & acquisitions.

- **Middle management**
  - sets objectives
  - allocates & controls resources
  - does planning
  - measures performance

- **Lower management**
  - supervises day-to-day operations
  - takes corrective action when necessary.

- **Operational level**
  - performs day-to-day operations
Goal Modeling

- **(Hard) Goals:**
  - Describe functions that must be carried out. E.g.
    - Satisfaction goals
    - Information goals

- **Softgoals:**
  - Cannot really be fully satisfied. E.g.
    - Accuracy
    - Performance
    - Security
    - ...

- **Also classified temporally:**
  - Achieve/Cease goals
    - Reach some desired state eventually
  - Maintain/Avoid goals
    - Keep some property invariant
  - Optimize
    - A criterion for selecting behaviours

- **Agents:**
  - Owners of goals
  - Choice of when to ascribe goals to agents:
    - Identify agents first, and then their goals
    - Identify goals first, and then allocate them to agents during operationalization

- **Modelling Tips:**
  - Multiple sources yield better goals
  - Associate stakeholders with each goal
    - Reveals viewpoints and conflict
  - Use scenarios to explore how goals can be met
  - Explicit consideration of obstacles helps to elicit exceptions
Scenario Types

• Current state and desired state
• Positive and negative
• Misuse scenarios
• Descriptive, exploratory and explanatory
• Instance, type and mixed scenarios
• System-internal, interaction, and context scenarios
• Main, alternative, and exception scenario
Case Description

The AUTOservice company has no software intensive system to manage and organize the AUTOservice work and to store information about the cars to repair. The major goals to achieve are:

- New car to repair registered;
- Car repairing status is up to date;
- Maintaining tracking of repairing schedules;
- Information about car conditions (e.g., particular defects) is recorded.

The major stakeholders are

- **owner** who would like to know all the information about the cars to repair but she has no other tasks to fulfil in or intention to use the information system
- **technician** who can read data related to the car such as general data, repairing status, particular defects. She will be also able to work with schedules (e.g., for repairing) and add other notes
- **manager** who can enter and update information about new car to repair in the AUTOservice, create repairing schedules, print generated reports
Task1: What are the social relationships between the stakeholders of the AUTOservice system?

To support your answer, create a strategic dependency model (using the j* modelling language), where social viewpoint of the given case is illustrated.
Task 1: What are the social relationships between the stakeholders of the AUTOservice system?

To support your answer, create a strategic dependency model (using the i* modelling language), where the social viewpoint of the given case is illustrated.
Task 2: Use KAOS modelling languages and refine goal “Information about car conditions (e.g., particular defects) is recorded” to the goal hierarchy

• Containing at least 4 hierarchy levels
• Including at least 2 alternative refinements
• Your model should separate between requirements and expectations
Task 2: Use KAOS modelling languages and refine goal “Information about car conditions (e.g., particular defects) is recorded” to the goal hierarchy

- Containing at least 4 hierarchy levels
- Including at least 2 alternative refinements
- Your model should separate between requirements and expectations
Task 3: Create a use case diagram to illustrate functions of the AUTOservice system

- In the ideal case, the use case diagram could potentially show how some requirements from task 2 are operationalized.

---

G1.1 Information about car conditions is recorded

G1.1 Run diagnostics

G1.2 Perform visual check

G1.3 Prepare car condition report

TRACES from

G1.1 Information about car conditions is recorded

Diagnostic system

Technician

Report controller
Task 3: Create a use case diagram to illustrate functions of the AUTOservice system

- In the ideal case, the use case diagram could potentially show how some requirements from task 2 are operationalised.
Task 4: Select **one use case** from diagram created in Task 3 and illustrate its scenarios by filling in this **use case template**
Task 4: Select one use case from diagram created in Task 3 and illustrate its scenarios by filling in this use case template

<table>
<thead>
<tr>
<th>Use case ID:</th>
<th>UC#1: Record information about car condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date created:</td>
<td>16.11.2020</td>
</tr>
<tr>
<td>Actors:</td>
<td>AUTOservice system, Technician,</td>
</tr>
<tr>
<td>Description:</td>
<td>In this use case the information about the car defects is collected. It is done by analyzing the visual look of the car, by running the diagnostic and by generating the defect report.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Car is moved to examination frame.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>The car is in service, car is not functioning as required.</td>
</tr>
<tr>
<td>Postcondition:</td>
<td>Defects are collected</td>
</tr>
</tbody>
</table>
| Main flow:    | 1. Technician plugs and starts diagnostic system  
                2. Diagnostic system runs diagnostic (includes UC#1.1)  
                3. Diagnostic system sends results to Report controller  
                4. Technician performs visual check (includes UC#1.2)  
                5. Technician enters results of the visual check to Reporting controller  
                6. Reporting controller prepare car condition report (includes UC#1.3)  
                7. Technician prints out the car condition report |
| Alternative flow: | None                                      |
| Priority:     | Not-known                                   |
| Assumptions:  | Owner allowed to perform car reparation and collect information about the car. |