

Advanced use case diagrams, Textual specification

PB007 Software engineering I

Stanislav Chren

27. 9. 2021



Relationships in use case diagrams

In addition to the communication association between actors and use cases, there are other types of possible relationships:

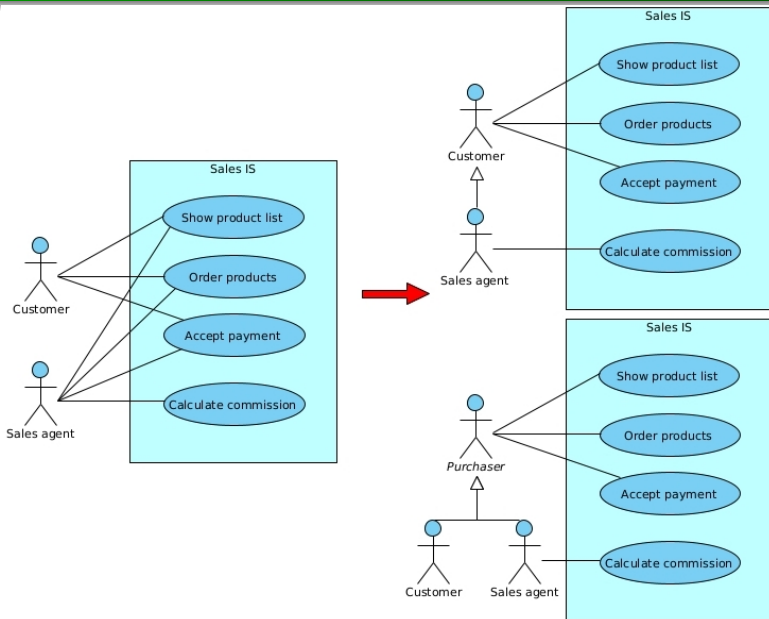
- actor generalization
- use case generalization
- `«include»`
- `«extend»`



Actor generalization (inheritance) is the relationship between the general and specialized actors.

- General actors are often abstract, i.e. they do not represent a real role in the system.
- Specialized actor inherits all roles and relationships of the parent actor.
- Each expected occurrence of the parent actor can be substituted by any of its descendant actors.
- It is usually suitable when multiple actors share some of the use cases.
- Simplifies the semantics of the diagram.

Actor generalization - example



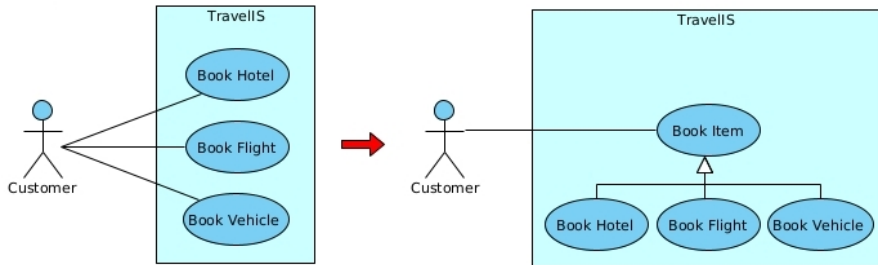
Use case generalization

Use case generalization (inheritance) is the relationship between the general and specialized use cases.

- Specialized use case inherits the properties of its parent and it can add new properties or override inherited ones.
Note.: it cannot overload parent's extension points.
- The text specification of inherited use cases should mark the changes from the parent use case.
- Parent use case can be abstract (*recommended*), i.e. it has either no specification or incomplete specification of the event flow.



UC generalization - example



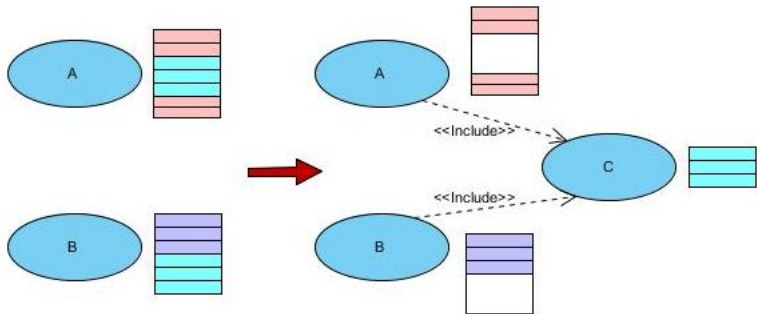
«include»

«include» allows to extract steps that are repeated in multiple use cases into separate use case.

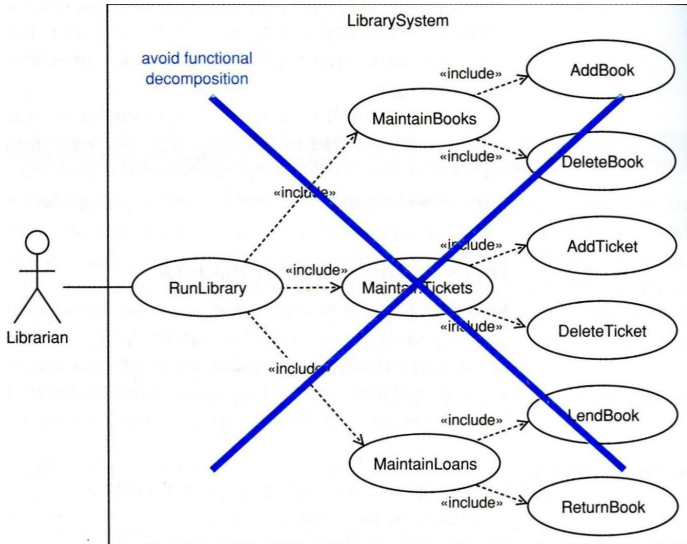
- The base use case is incomplete without all the included use cases.
- The included use case may or may not be complete.
- It should not be misused for functional decomposition



«include»



«include»



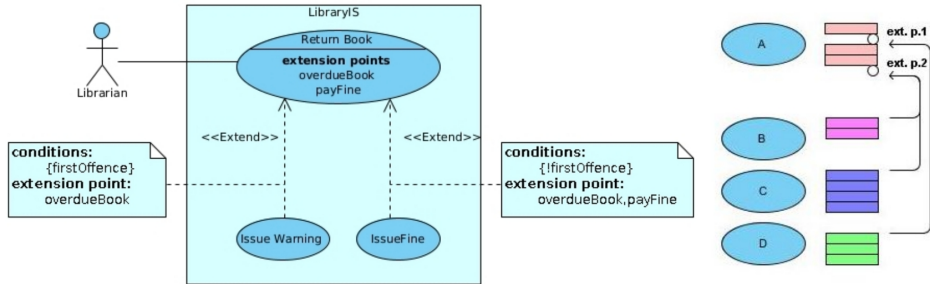
«extend»

The «extend» relationship allows to add a new behaviour into an existing use case.

- The base use case contains *extension points* within the flow of events.
- The extension use case specifies to which extension points it is attached.
- The base use case does not know about its extension use cases.
- The extension use case is usually incomplete. It can contain multiple segments, each specifying their extension points.
- Multiple extension use cases can share the same extension points. In such case, it is appropriate to define conditions to determine which of them should be used in given situation.



«extend»



Textual specification of use cases

Textual specification of use cases should contain:

- Name (ID)
- A brief description
- The primary actors
- The secondary actors
- Preconditions
- The main flow of events
- Post-conditions
- An alternative flow of events



Flow of events

Main flow(Primary Scenario) of events is a sequence of steps in the interaction between actors and the system in the ideal case (i.e. without errors, interruptions, etc.)

- It always starts with an action from the primary actor. It is recommended to use the form:
1. The use case begins when, <actor><function>.
- The individual steps should be short, precise and understandable. Use the form:
<step id><actor/system><action>
- You can modify the flow with IF, FOR, WHILE, GOTO etc.

The **alternative flow** represents deviations from the main flow caused by the errors and interruptions. It can be also used to capture more complex branching.



- http://sce.uhcl.edu/helm/rationalunifiedprocess/process/artifact/ar_ucmod.htm
- http://www.karonaconsulting.com/downloads/UseCases_IncludesAndExtends.pdf



Tasks

- Review the initial use case diagram from the previous session. Fix any problems and consider the use of advanced relationships (include, extend, generalizácia).
- Briefly document all use cases. This will help to better understand the project.
- Choose 3 use cases (ideally the more complex ones) for detailed textual specification. The selection should be approved by the seminar tutor. The chosen use cases should use different colour in VP.
- Generate a **PDF report** and upload it to the homework vault (**Seminar 03**)

Deadline (in the given day before the next seminar until 23:59):

- Saturday (Groups 03, 04)
- Monday (Group 11)
- Tuesday (Groups 06, 07)
- Wednesday ([06:00 AM] Groups 08, 09)



Rules for report submission

- 1 Submit the PDF report, not the VP source file and not an exported image.
- 2 PDF report must be created using the procedure shown on the seminars including the report settings.
- 3 The name of the PDF report file should be *lastname1-lastname2-lastname3* of the team members.
- 4 PDF report must contain all diagrams modelled until now.
- 5 PDF report must be uploaded to the homework vault by the specified deadline.
- 6 PDF report must be uploaded to the correct homework vault. The name of the homework vault is always specified on the slides.
- 7 Each team uploads only a single PDF report for the whole team.
- 8 Submitted diagrams must be clear and readable.
- 9 Submitted diagrams should not contain serious mistakes. At least, they should not contain mistakes mentioned in the *Catalogue of common mistakes*.



VP report settings

Generate PDF

Content Options Page Setup Cover Page Header/Footer Document Info Watermark

Options

- ☒ Generate table of contents ?
- ☒ Generate table of figures ?
- ☒ Generate diagrams ?
Image type : SVG
- ☒ Generate diagram type title
- ☐ Generate diagram properties ?
- ☒ Generate diagram summary ?
- ☐ Include extra details
- ☐ Suppress element with blank documentation in summary table
- ☒ Generate reference (file/URL) link ?
- ☐ Generate model elements/diagrams link ?
- ☐ Skip heading for empty model element section
- ☐ Convert multiline model heading to single line
- ☐ Show multiline model name
- ☐ Treat HTML content as HTML source
- ☐ Suppress details if duplicated
- ☒ Table cell keep together with page
- Wrap : Word wrap
- Shape type style : Icon ?
- RTF content appearance :
Preserve formatting

Details

- ☒ Children
 - ☒ Model-based
 - ☐ Diagram-based
- ☐ Members
 - ☐ ERD Column Details
- ☐ Properties
- ☐ Project management properties
- ☐ Relationships
- ☒ Quality information
- ☐ References
- ☐ References documentation
- ☒ Sub-diagrams
- ☐ Include sub-diagram details
- ☐ Comments
- Sort by Date/Time: Descending
- ☐ Tagged values
- ☐ ORM Class Details
- ☒ Use Case Details

Anti-aliasing

- ☒ Graphics
- ☒ Text

Font

Font: Unspecified ...

Reset Reset to Default Set as Default Generate Cancel Apply Help