Analysis Class Diagrams

PB007 Software Engineering I

Bruno Rossi

21. 10. 2015

Analysis Class Diagram

A **Class Diagram** is a static view of the classes, their attributes, operations and relationships.

The **Analysis Class Diagram** shows the analysis classes that represent the concepts in the business domain of the system to be modelled, i.e. it does not go into the implementation details.

Properties of analysis classes

Good designed analysis classes should have the following characteristics:

- a name that reflects well the purpose;
- they have a small number (3-5) of responsibilities/operations;
- they are not isolated from other classes;
- have high cohesion;
  Example: class *ShoppingCart*, operations *addItem(), removeItems(), displayContents(), receiptOfPayment(),printInvoice()*
- they have few links to other classes (coupling);

Be careful about:

- a large number of small classes;
- a small number of very large classes;
- functions/procedures that are created as separate classes;
- classes that govern/manage other classes. Often they are called *system*, *controller*, or *manager*;
- a complex inheritance hierarchy (max 2 levels);

Finding Analysis Classes

Analysis of nouns and verbs:

- gathering available resources (specifications, documentations, use cases, ...);
- nouns are candidates for class or attributes;
- verbs or verb patterns are candicate for methods/operations within classes;
- watch out for **hidden** classes/concepts;

CRC (Class, Responsabilities, Collaborators) Analysis

- group activity involving brainstorming;
- on a card that represents one class, you can add the class name, the responsabilities and the collaborators (other classes that are in relation with the class);

Relations between classes

The basic relationships include:

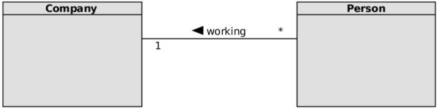
- Generalization
- Association
- Dependency

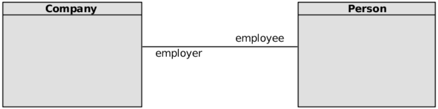
Association is the semantic relationship between classes.
It is represented by the following **attributes**:

- name
- name of roles
- multiplicity
- navigability of associations

Relations between classes II

Name of the association and names of roles:

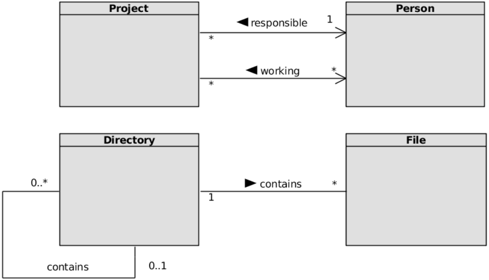




You can only use one option, not both. Generally the name of the association is preferred, but it depends if you want to specify the names of the roles to make the relationships clearer.

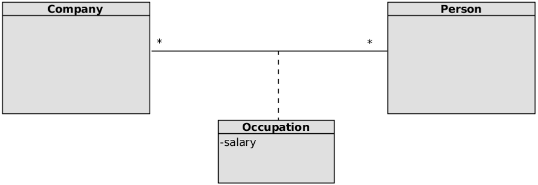
Relations between classes III

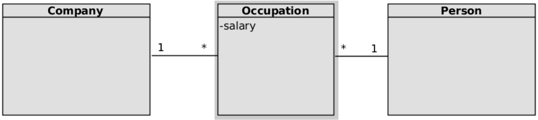
Multiple and reflexive associations:



Relations between classes IV

Association class:





To generate a class diagram

1. Find classes, basic attributes, operations and co-workers
2. Determine inheritance levels between classes
3. Capture relationships by means of associations
4. Name associations or roles
5. Determine the multiplicity of associations
6. Capture dependencies
7. Add additional attributes and operations that are in the domain

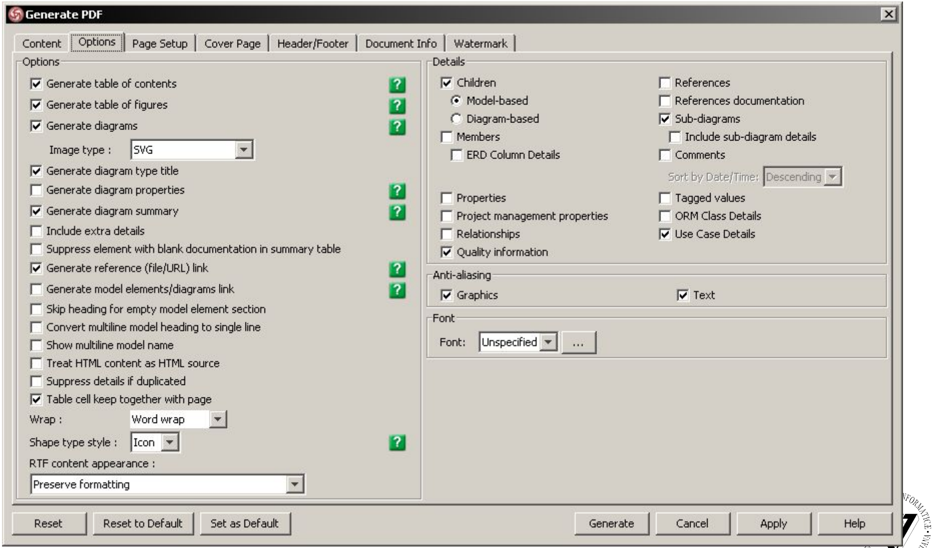
Resources

- www.uml.org.cn/umlapplication/pdf/crcmodeling.pdf
- www.agilemodeling.com/artifacts/classDiagram.htm
- sourcemaking.com/uml/modeling-it-systems/structural-view/class-diagram
- http://sourcemaking.com/uml/modeling-it-systems/structural-view/constructing-class-diagrams

Tasks

- Correct the mistakes from the previous task;
- Look into the specifications for information for the analysis diagrams, classes, basic attributes, operations and relationships;
- Draw the analysis class diagram including inheritance and names of associations/roles;
- Upload the **pdf report** to the folder (**Week 05**).
- **Deadline:** Mon 26.10.15 23:59 (Groups 2,3)

Customization of PDF Reports



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 ...