OO Analysis

ISEP / LETI / ESOFT

Topics

- Domain Model
- Concept Identification
- Association Identification
- Attribute Identification
- Generalization Pitfall
- Remarks on the Domain Model
- Domain Model Evolution
- Modeling Roles

Analysis

Analysis

- It implies:
  - Acquiring domain/business knowledge
  - Understanding the domain/business
  - Reasoning about the domain/business

- Analysis oriented by:
  - **Objects → OO Analysis**
  - (Processes or Activities → Not covered in ESOFT)

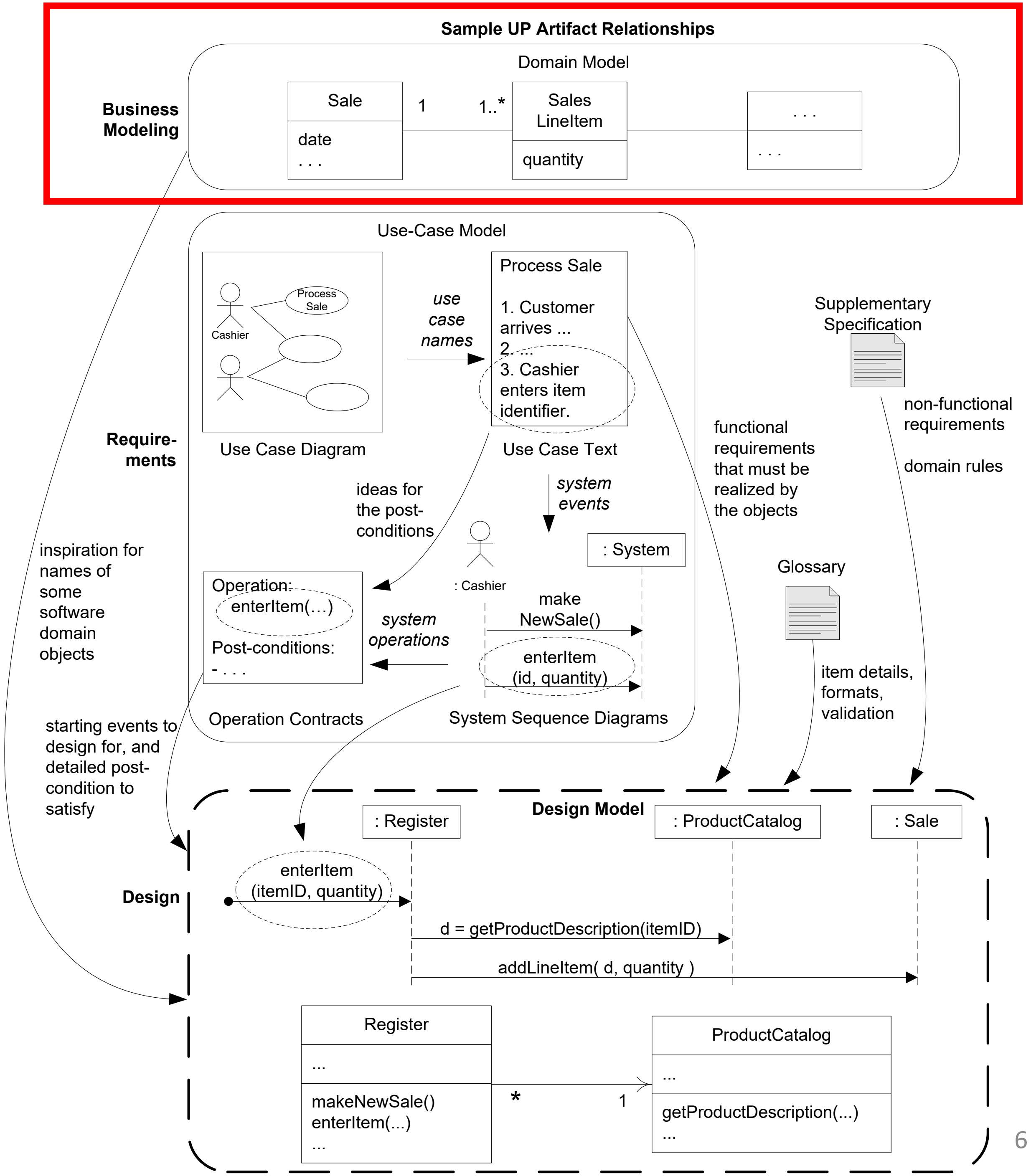
OO Analysis

- Note: Requirements Engineering **is not** an OO discipline/activity

- Which are the domain concepts/objects?

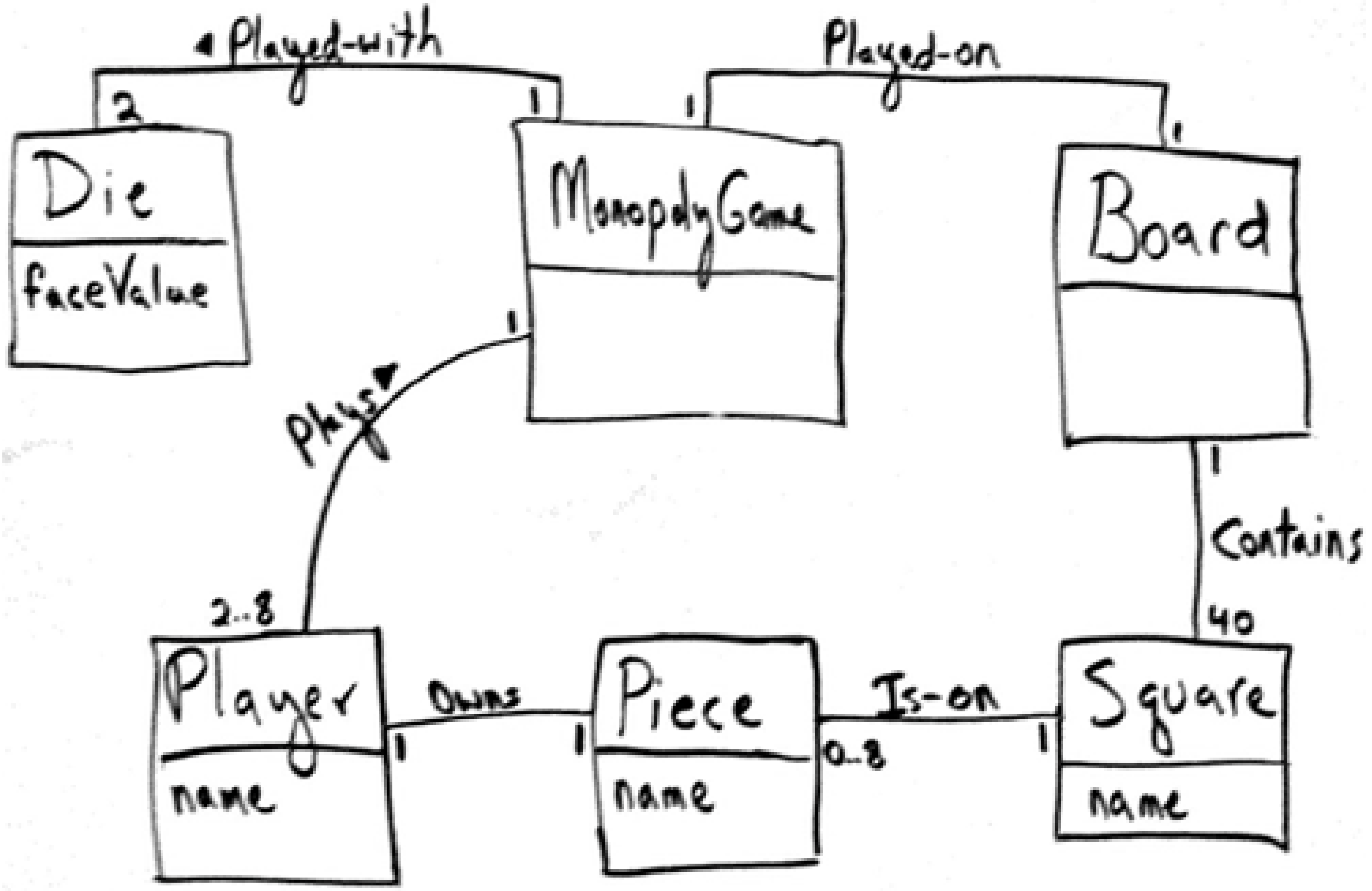
- Described in a domain (object) model
  - Classifying domain elements/concepts
  - Finding relationships between domain concepts
  - **Goal:** to narrow the representational gap between **requirements** and **design**

Artifacts Overview

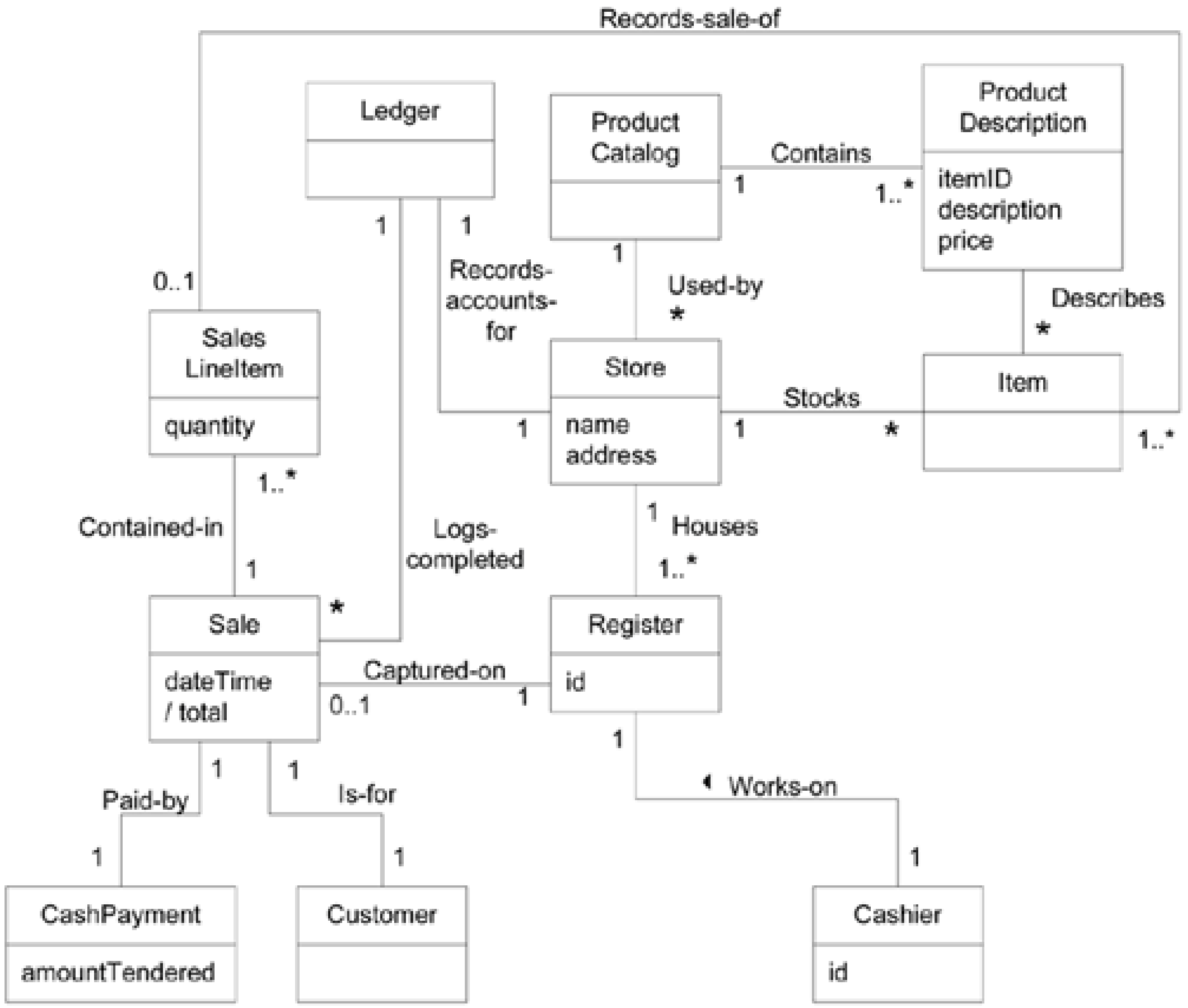


Domain (Object) Model

Example 1: Monopoly Game



Example 2: NextGen POS



Domain (Object) Model (DM)

- Artifact of the “Business Modeling” discipline
- UML visual representation of conceptual classes or real objects in the domain of interest
  - Later, they **may or may not** become software elements
  - Conceptual classes ≈ Business concepts/entities
- Input data
  - Descriptive domain artifacts. E.g.:
    - Project **Specification Document**
    - **Use Case Model** (from the Requirements Engineering activity)
  - Conversations with the SW client

Elements of the Domain Model

- **Concepts** (or conceptual classes)
  - Typically, something more complex than a number or text
  - E.g. **Student**, **Course**, **Program**
- **Associations** between concepts
  - Relationships between concepts
  - They are dependent on what matters for the specific problem
  - E.g. Student **is enrolled** in Course, Course **takes place as part of** Program
- **Attributes**
  - Data values that (partially) characterize an object
  - Typically, something that is expressed numerically, by text or by a date
  - E.g. Student **number**, **name** or **birthdate**

Conventions (1/2)

- Concept names
  - Start with a capital letter
  - In the singular

- Attribute names
  - Start with a lowercase letter

Conventions (2/2)

- Association names
  - Start with a lowercase letter
  - Chosen so that:

    Concept1 ***association name*** *Concept2*

    is a sentence with meaning

- Reading direction
  - From left to right and top to bottom
  - Otherwise, the reading direction must be indicated in the association

Concepts

How to identify concepts?

- Use a **Class Category List**
  - Focus on business information needs

- Look for **nouns** in sentences
  - **Linguistic analysis** of input data (e.g. Project Specification Document)
  - Results in a list of **candidate conceptual classes**
  - **Attention:**
    - do not apply a mechanical mapping of nouns to concepts
    - pay attention to the ambiguity of the words

- Modify/reuse existing models
  - There are templates for the most common domains like sales, stocks, finance, healthcare, etc.
  - (not covered in ESOFT)

Class Category List

- (Business) Transactions
- Transaction line items
- Products or Services related to a Transaction or Transaction line item
- Transaction Records
- Roles of People or Organizations
- Places
- Noteworthy Events
- Physical objects
- Descriptions of things
- Catalogs
- Containers of things
- Elements of containers
- (Other) Organizations
- Other (External) Systems
- Records of finance, work, contracts, legal matters
- Financial instruments
- Documents mentioned/used to perform some work

Identifying Conceptual Classes (1/5)

Platform for Outsourcing Tasks

- Excerpts from the Project Specification Document

A startup needs to develop a software product that, on the one hand, allows the registration of any interested **organization**, to be able to publish **tasks** and manage the process of assigning these tasks to **freelancers**; and, on the other hand, allow freelancers to easily access these tasks and apply for them.

This product is is expected to be accessed by several users with different roles, such as:

- Organization **Employee**: someone acting on behalf of a particular organization; employees are responsible for specifying and cataloging tasks for later publication by the organization.
- Freelancers: people who propose to carry out tasks published by organizations.

Identifying Conceptual Classes (2/5)

Platform for Outsourcing Tasks

- From the Project Specification Document

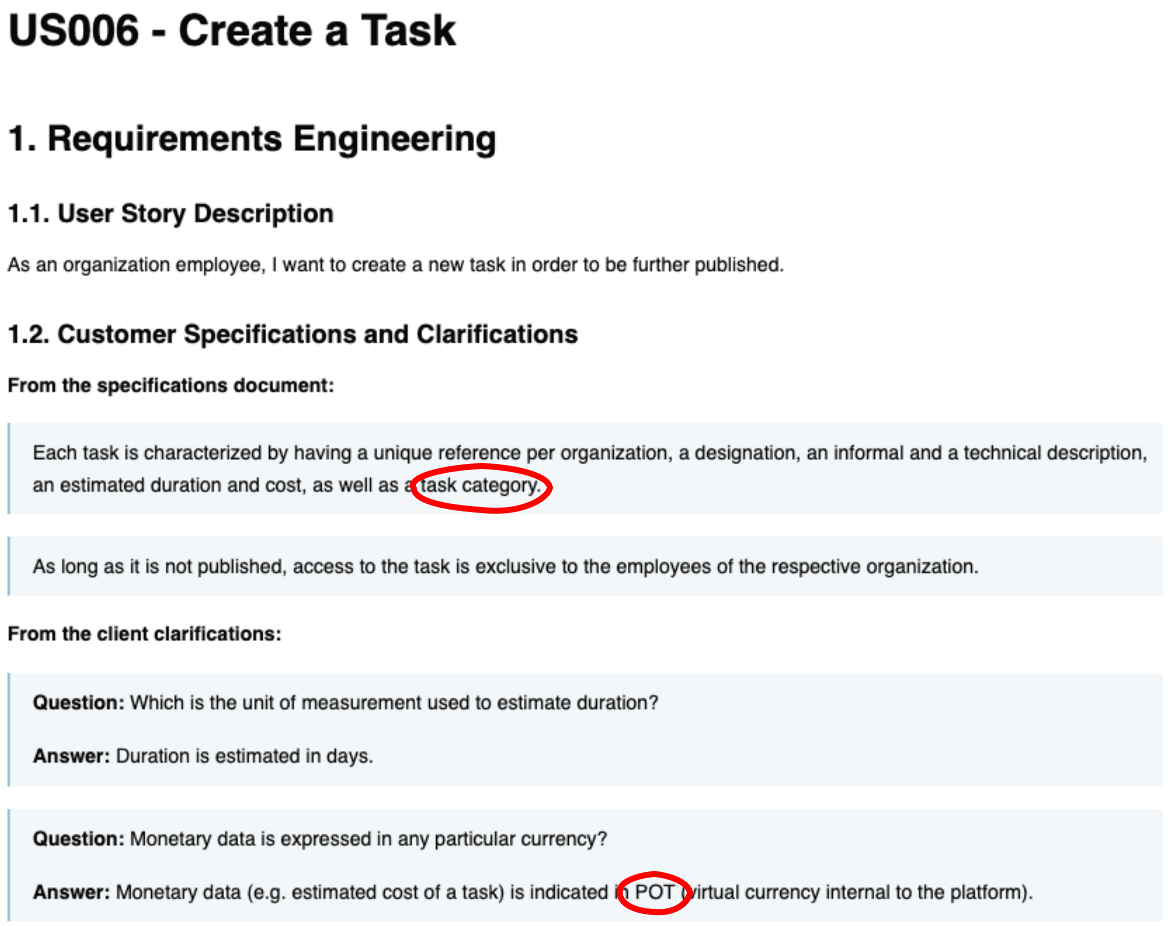
Category	Conceptual / Candidate Classes
(Business) Transactions	Task
Transaction line items	
Products or Services related to a Transaction or Transaction line	
Roles of People or Organizations	Employee, Freelancer
Places	
Noteworthy Events	
Description of things	
Catalogs	
Containers of things	
(Other) Organizations	Organization
Financial instruments	
etc.	

Identifying Conceptual Classes (3/5)

Platform for Outsourcing Tasks

- From the Requirements Engineering artifacts

Category	Conceptual / Candidate Classes
(Business) Transactions	Task
Transaction line items	
Products or Services related to a Transaction or Transaction line	
Roles of People or Organizations	Employee, Freelancer
Places	
Noteworthy Events	
Description of things	**(Task) Category**
Catalogs	
Containers of things	
(Other) Organizations	Organization
Financial instruments	**POT**
etc.	



US006 - Create a Task

1. Requirements Engineering

1.1. User Story Description

As an organization employee, I want to create a new task in order to be further published.

1.2. Customer Specifications and Clarifications

From the specifications document:

> Each task is characterized by having a unique reference per organization, a designation, an informal and a technical description, an estimated duration and cost, as well as a task category.

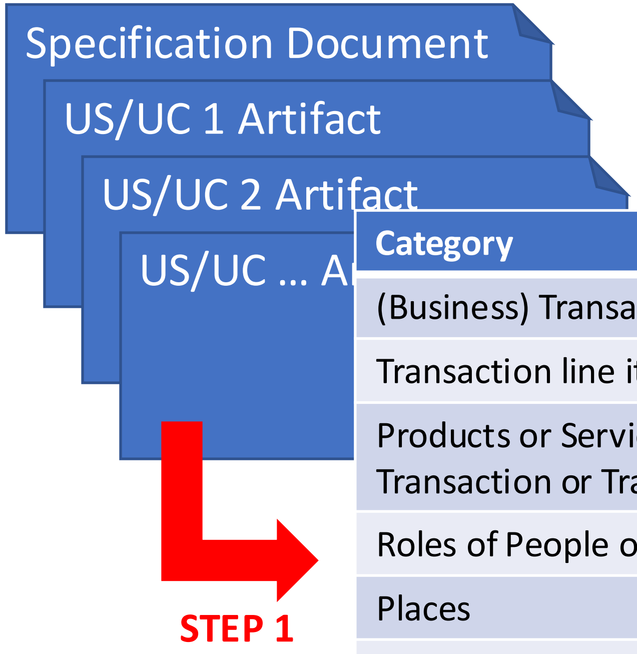
> As long as it is not published, access to the task is exclusive to the employees of the respective organization.

From the client clarifications:

> **Question:** Which is the unit of measurement used to estimate duration?
>
> **Answer:** Duration is estimated in days.

> **Question:** Monetary data is expressed in any particular currency?
>
> **Answer:** Monetary data (e.g. estimated cost of a task) is indicated in POT (virtual currency internal to the platform).

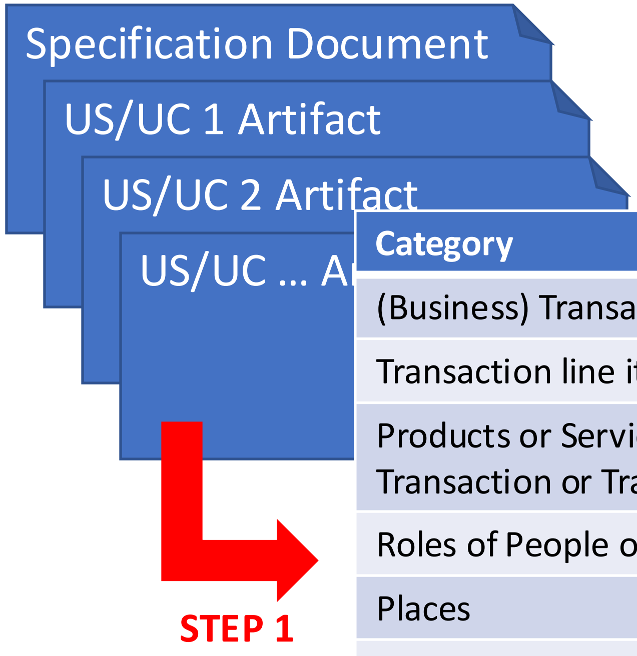
Identifying Conceptual Classes (4/5)



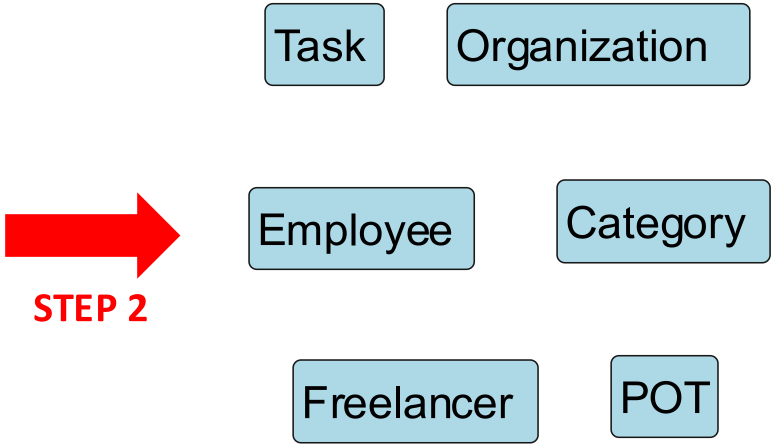
Category	Conceptual / Candidate Classes
(Business) Transactions	Task
Transaction line items	
Products or Services related to a Transaction or Transaction line	
Roles of People or Organizations	Employee, Freelancer
Places	
Noteworthy Events	
Description of things	(Task) Category
Catalogs	
Containers of things	
(Other) Organizations	Organization
Financial instruments	POT
etc.	

Identifying Conceptual Classes (5/5)

Platform for Outsourcing Tasks



Category	Conceptual / Candidate Classes
(Business) Transactions	Task
Transaction line items	
Products or Services related to a Transaction or Transaction line	
Roles of People or Organizations	Employee, Freelancer
Places	
Noteworthy Events	
Description of things	(Task) Category
Catalogs	
Containers of things	
(Other) Organizations	Organization
Financial instruments	POT
etc.	



Associations

How to identify associations?

- An association is a relationship between objects.
- Use a **List of Common Associations**, such as:
  - A is physically (or logically) part of B
  - A is physically (or logically) contained in B
  - A is a description of B
  - A is known/ captured/ recorded by B
  - A uses or manages B
  - A is related to a transaction of B
  - etc.
- Focus on **verbs** that relate two conceptual classes. E.g.:
  - “[…] registration of any interested **organization**, to be able to **publish** **tasks** [...]”
  - “[…] **employees** are responsible for **specifying** and cataloging **tasks**”

Identifying Associations (1/2)

Platform for Outsourcing Tasks

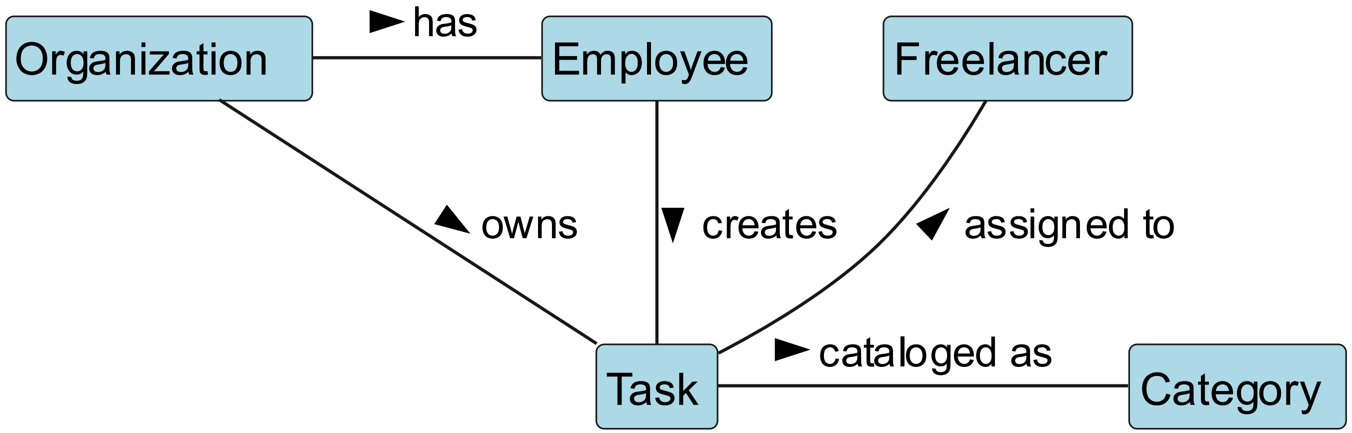
- Candidate associations captured through a table

Concept A	Association	Concept B
• Organization	• owns / publishes • has	• Task • Employee
• Employee	• works for • creates	• Organization • Task
• Task	• published by • cataloged as • created by • assigned to	• Organization • Category • Employee • Freelancer
…	…	…

Identifying Associations (2/2)

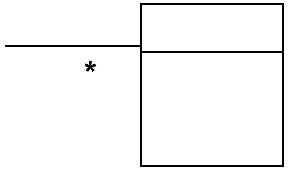
Platform for Outsourcing Tasks

- Visually representing associations
  - Select the most relevant associations for the business
  - Which direction is most relevant? E.g.:
    - Employee **works for** Organization?
    - Organization **has** Employee?

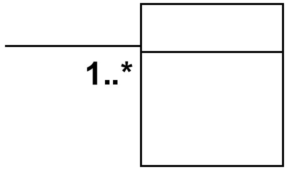


Multiplicity – Examples and Meaning

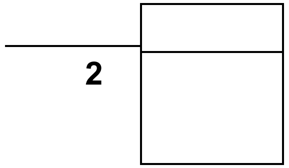
- Zero or more



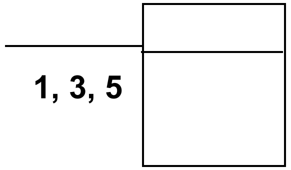
- One or more



- Exactly two



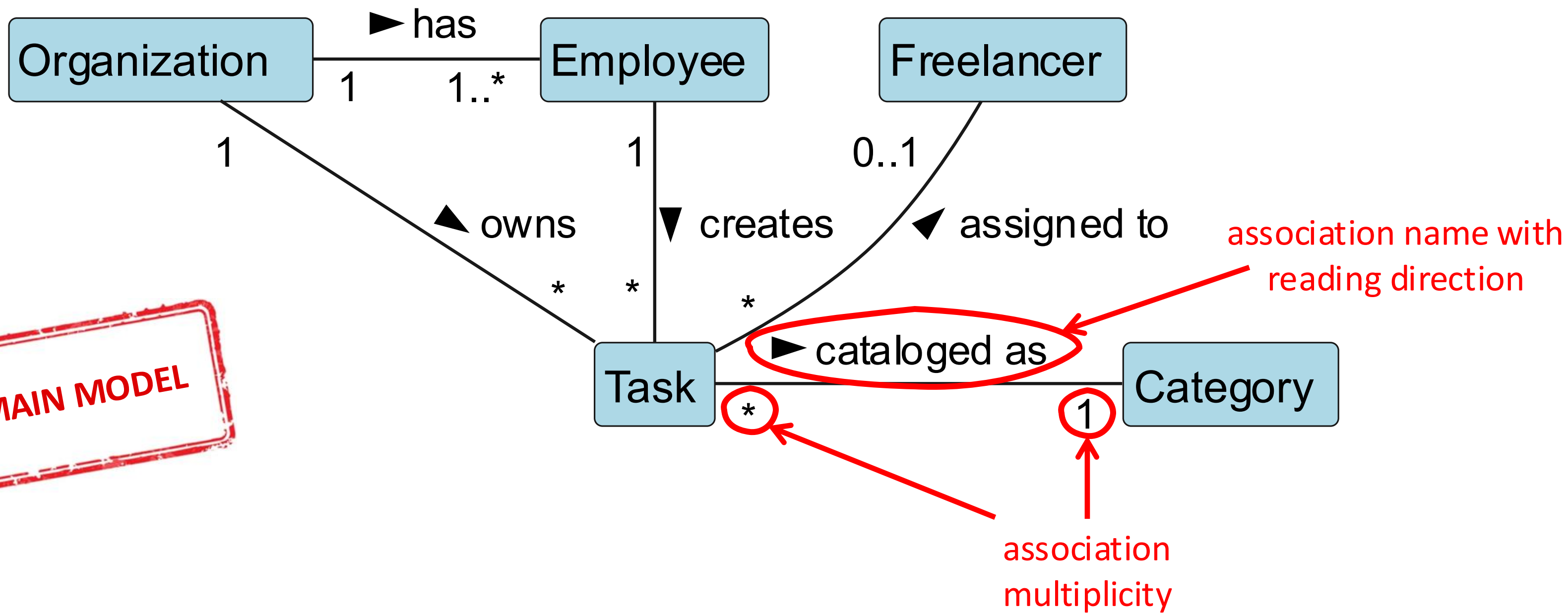
- One or three or five



Multiplicity on Associations

Platform for Outsourcing Tasks

- Adding multiplicity to associations. E.g.:
  - One Task is assigned to zero or one (0..1) Freelancer
  - One Freelancer is (or can be) assigned to several (*) Task



DOMAIN MODEL

UNIFIED MODELING LANGUAGE™

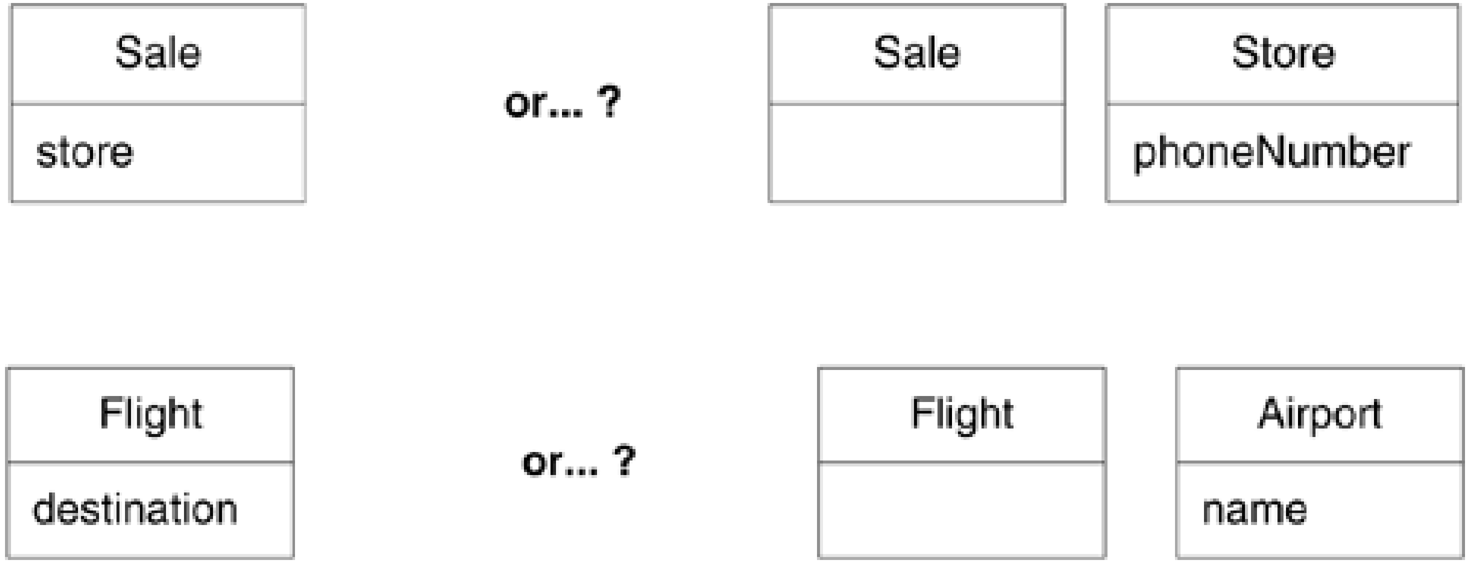
Attributes

How to identify attributes?

- Preferably they should be of a **simple and primitive type**
    - E.g. number, (unstructured) text, date, time, boolean
- Attributes should be represented as conceptual classes if:
    - the “attribute” consists of separate sections (i.e. has a structure)
        - E.g.: **address** consists of a street, a number, a postal code and a country
    - it has associated operations or behaviors
    - it has other attributes
    - it is an abstraction of one or more types
        - E.g. **barcode** is an abstraction of EAN (European Article Number) or UPC (Universal Product Code)
- **Attention**: sometimes, for readability reasons, highly experienced teams omit some of these concepts. However, **for beginners** this is discouraged and should be avoided.

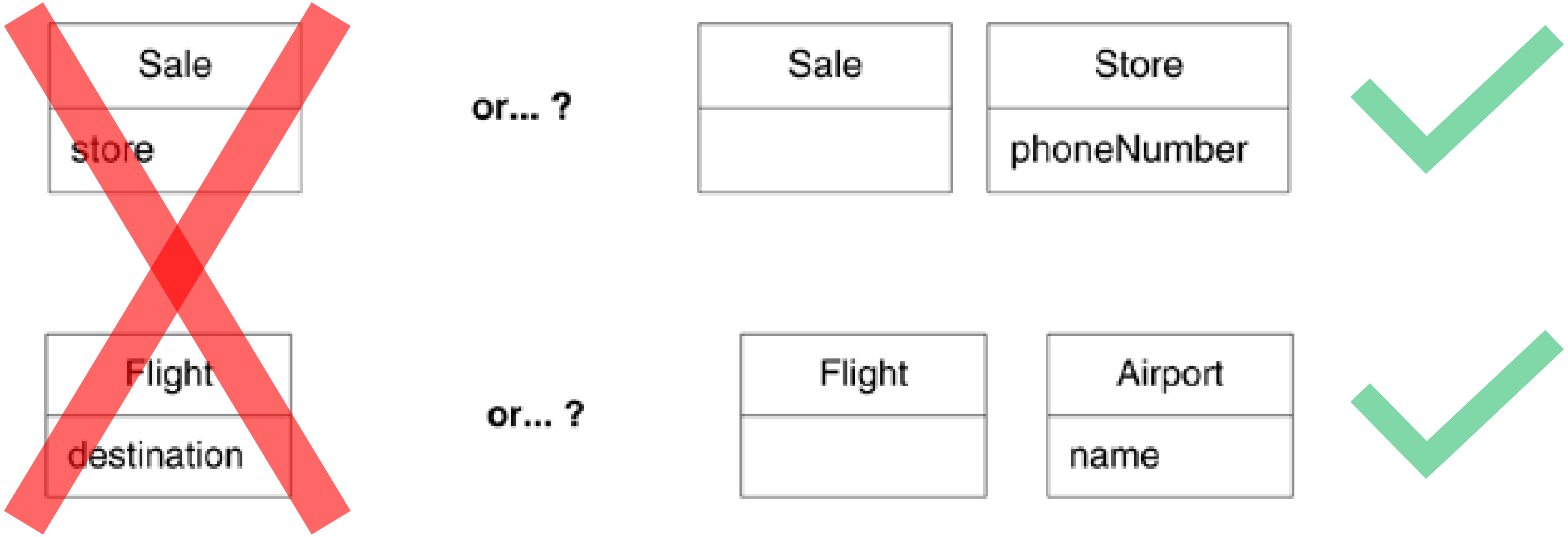
Attribute or Concept? (1/2)

- Representing as an attribute something that should be represented as a conceptual class



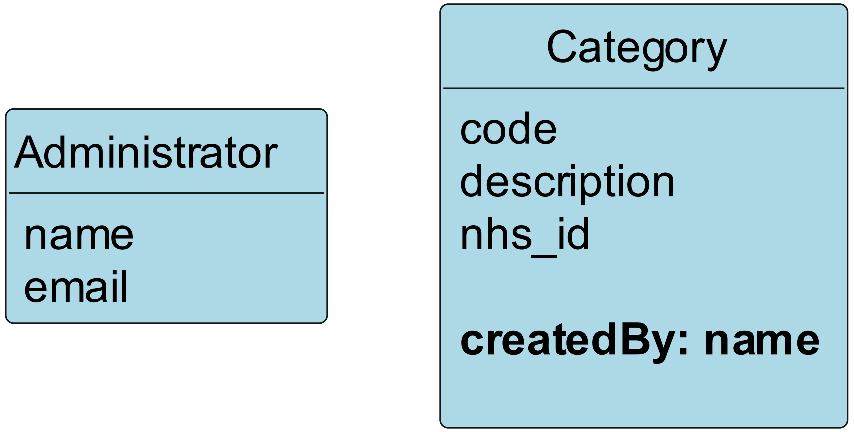
Attribute or Concept? (2/2)

- Representing as an attribute something that should be represented as a conceptual class



If, in the real world, attribute **A** is not thought of as a number or text, then **A** is most likely a conceptual class and not an attribute.

Attribute or Association? (1/2)



Administrator

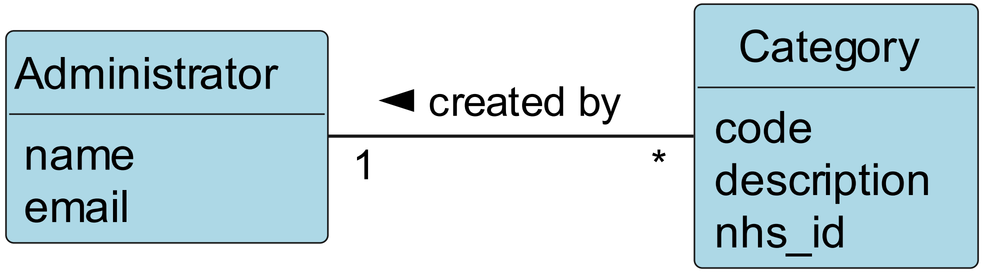
name
email

Category

code
description
nhs_id

createdBy: name

or... ?



Administrator

name
email

◄ created by

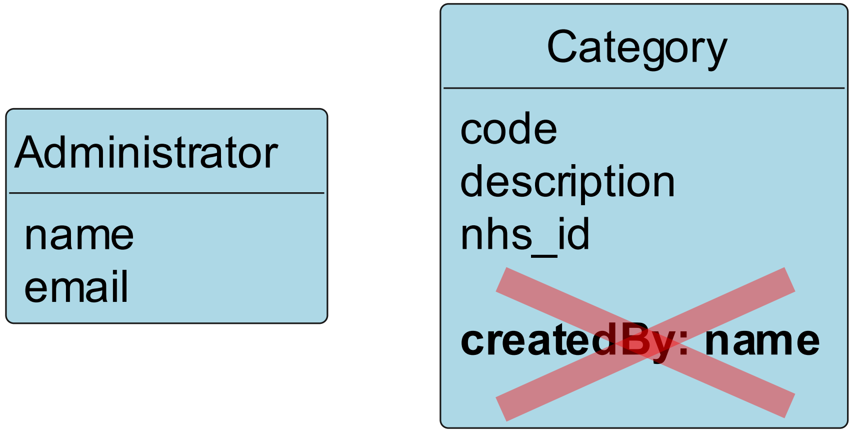
1

*

Category

code
description
nhs_id

Attribute or Association? (2/2)



Administrator

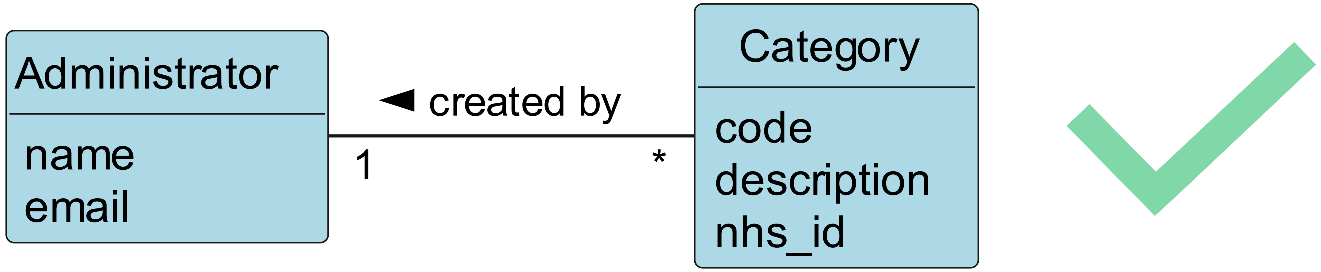
name
email

Category

code
description
nhs_id

~~**createdBy: name**~~

or… ?



Administrator

name
email

◄ created by

1

*

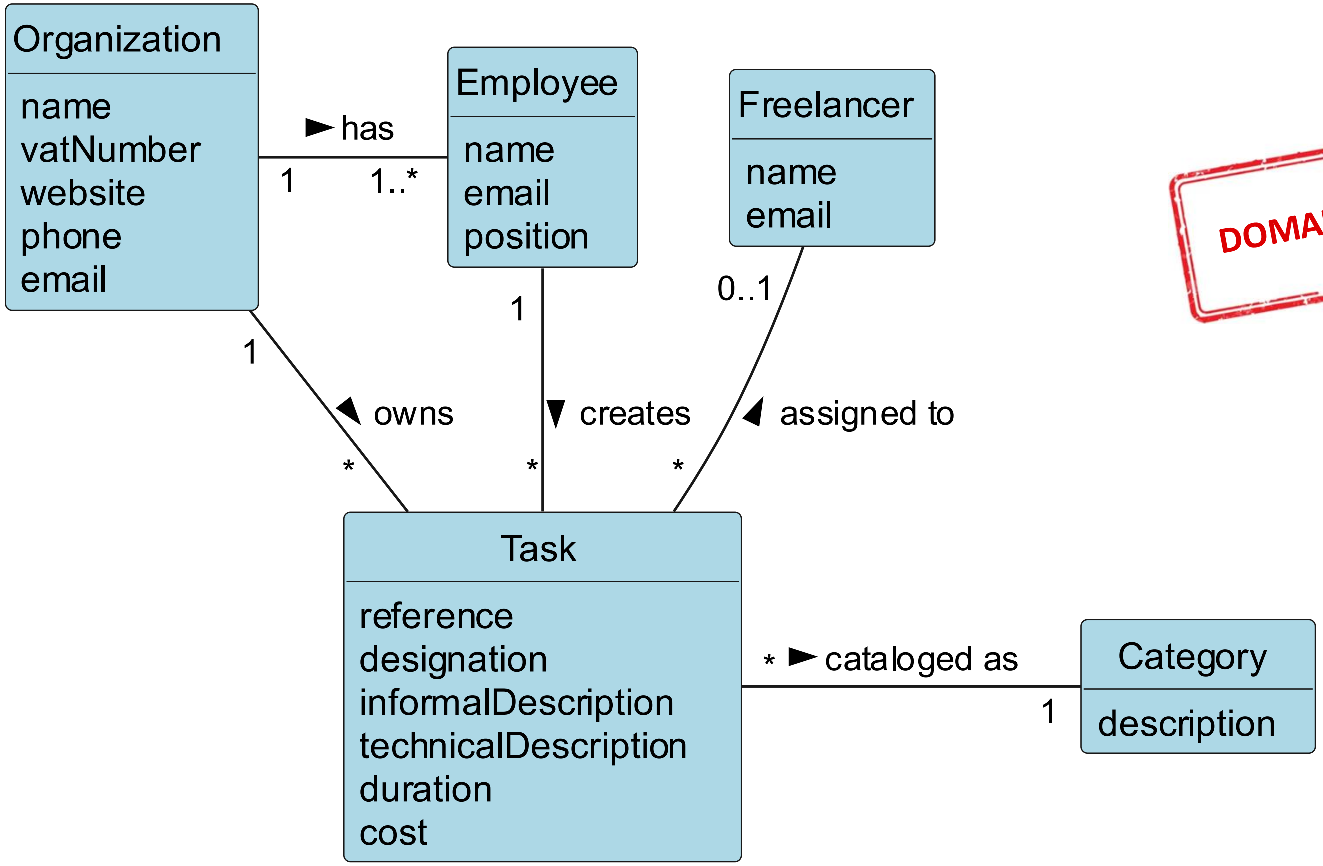
Category

code
description
nhs_id

✓

Identifying Attributes

Platform for Outsourcing Tasks



DOMAIN MODEL

UNIFIED MODELING LANGUAGE™

Are there any attribute that should be represented as a concept? Which one(s)?

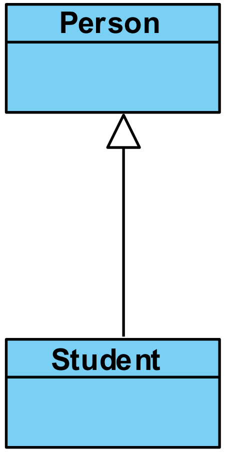
Generalization Pitfall

Generalization – Conceptual Classes

- Is there a generalization/specialization class hierarchy relationship between two conceptual classes?

- The semantic of this kind of relationship is “***is a***”
  - E.g.: a Student ***is a*** Person

- The subclass inherits the properties (attributes, operations and associations) of the superclass, and may add others
  - E.g.: *superclass* – Person; *subclass* – Student



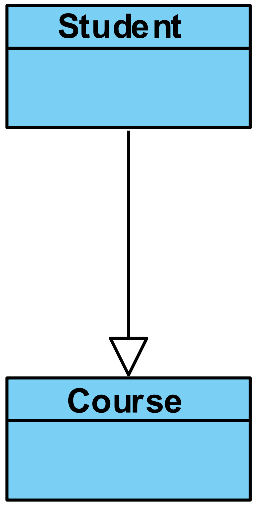
When it is useful to create conceptual subclasses?

1. Does the subclass have any **additional attributes**?
2. Does the subclass have any **additional associations**?
3. Is the subclass somehow **handled/operated differently** than the superclass?
4. Does the subclass represent something animated (e.g. an animal) that **behaves differently** from the superclass?

- Creating conceptual subclasses can be useful if the answer to at least one of these questions is “yes”.

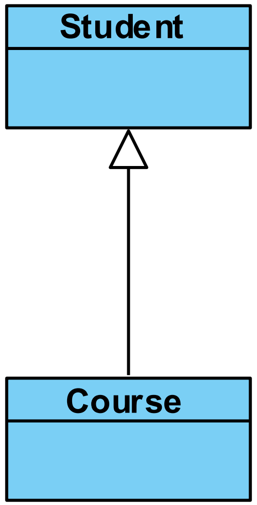
Generalization – Student and Course (1/2)

- Is there a hierarchical relationship between Student and Course or vice-versa?



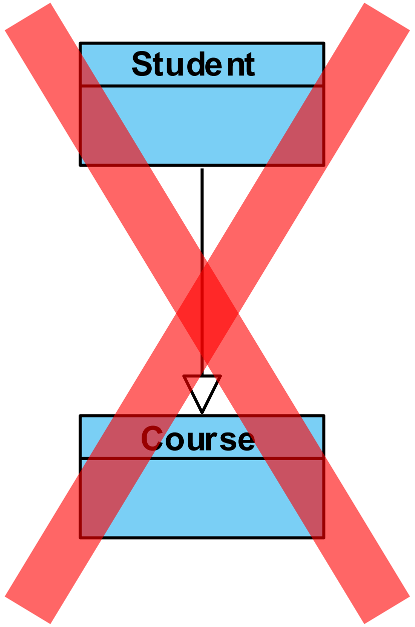
- ➢ Is 100% of the definition of the conceptual superclass applicable to the subclass?
- ➢ Does the "is a" relationship holds between such classes? **A Student is a Course?**

- ➢ Is 100% of the definition of the conceptual superclass applicable to the subclass?
- ➢ Does the "is a" relationship holds between such classes? **A Course is a Student?**

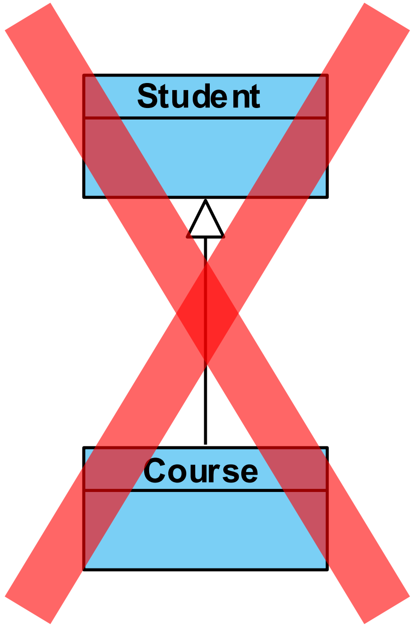


Generalization – Student and Course (2/2)

- Is there a hierarchical relationship between Student and Course or vice-versa?



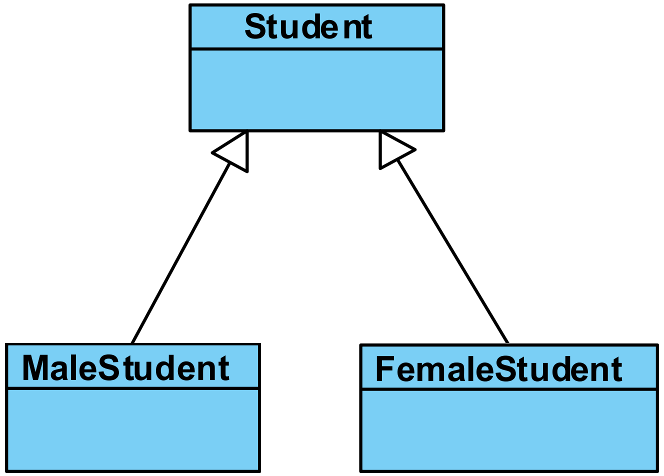
- Is 100% of the definition of the conceptual superclass applicable to the subclass?
- Does the "is a" relationship holds between such classes? **A Student is a Course?**



- Is 100% of the definition of the conceptual superclass applicable to the subclass?
- Does the "is a" relationship holds between such classes? **A Course is a Student?**

Generalization – Male/Female Student (1/2)

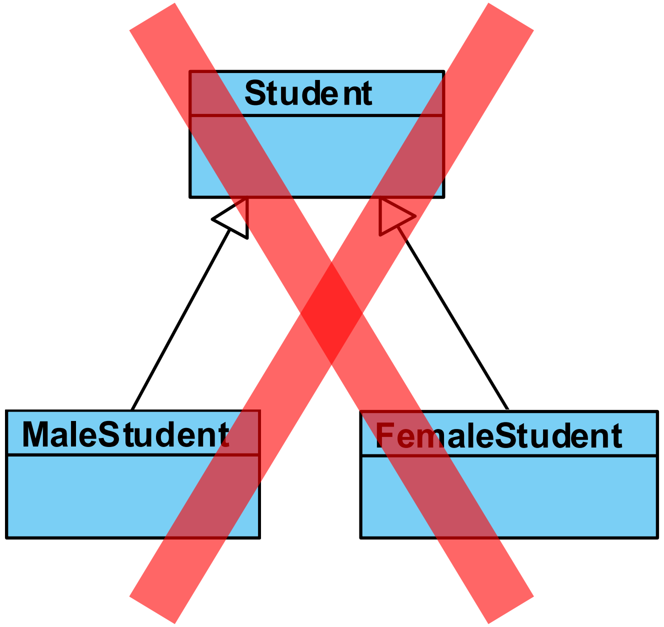
- Is there a hierarchical relationship between Student and MaleStudent or FemaleStudent?



1. Does the subclass have any **additional attributes**?
2. Does the subclass have any **additional associations**?
3. Is the subclass somehow **handled/operated differently** than the superclass?
4. Does the subclass represent something animated (e.g. an animal) that **behaves differently** from the superclass?

Generalization – Male/Female Student (2/2)

- Is there a hierarchical relationship between Student and MaleStudent or FemaleStudent?



1. Does the subclass have any **additional attributes**?
2. Does the subclass have any **additional associations**?
3. Is the subclass somehow **handled/operated differently** than the superclass?
4. Does the subclass represent something animated (e.g. an animal) that **behaves differently** from the superclass?

Generalization should be evaluated considering the domain/business context.

Remarks on the Domain Model

Only One Model per Domain?

Is My Domain Model (DM) Correct?

- There is not just one DM that is the correct one
    - **Different DM may be correct**
    - They are all approximations to the domain we are trying to understand
- DM should be seen as a **tool**:
    - **to understand the domain**
    - **to communicate**:
        - with the SW client
        - within the development team
- DM is useful when it:
    - captures abstractions and essential information needed to understand the domain, in the context of current requirements
    - assists people understand concepts, terminology and domain relationships

Last Check and Remarks

- Check your Domain Model against each User Story / Use Case
  - Are all the concepts involved in the US/UC represented in the DM?
    - Objects being created, modified or deleted
  - Are all the associations between concepts represented in the DM?
    - Association/relationships being created/modified/deleted between objects
  - Is all input data captured as attributes and/or associations between concepts?
    - Some data will be saved as attributes of one or more (new) objects
    - Other data is used to established/delete objects and/or associations between objects
  - Can you find all output data?
    - E.g. lists of objects meeting some criteria
  - Did you find anything missing?
    - YES → Revise/complete your DM
    - NO → Are you sure? Really? Then, you're probably on the right track…

Domain Model Evolution in the Project

Domain Model throughout the Project

ESOFT

- Sprint 1
  - Goal: broad and comprehensive understanding of the domain/business
  - Main Inputs:
    - Project Specification document
    - Requirement Engineering artifacts
    - Conversations with the SW Client

- Sprints 2 to 4
  - Goal: deeper and more comprehensive understanding of the domain/ business regarding the sprint backlog (i.e. user stories to develop during the sprint)
  - Main Inputs:
    - Requirement Engineering artifacts
    - Conversations with the SW Client

A New Sprint – What to do now?

- What is the impact of the new requirements on the Domain Model?
  - Do they cause any changes to existing concepts and associations?
  - Have new concepts emerged?
  - What are the new concepts?
  - How do they relate/associate with previously known concepts?
  - What are the attributes of the new concepts?

- The Domain Model should be **revised/updated** in every sprint

- The knowledge resulting from the new requirements must be reflected in the Domain Model

Scenarios to Think About

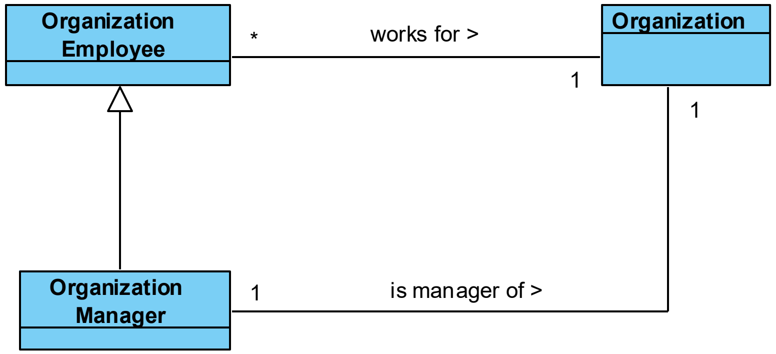
Platform for Outsourcing Tasks

- Excerpts from the Project Specification Document

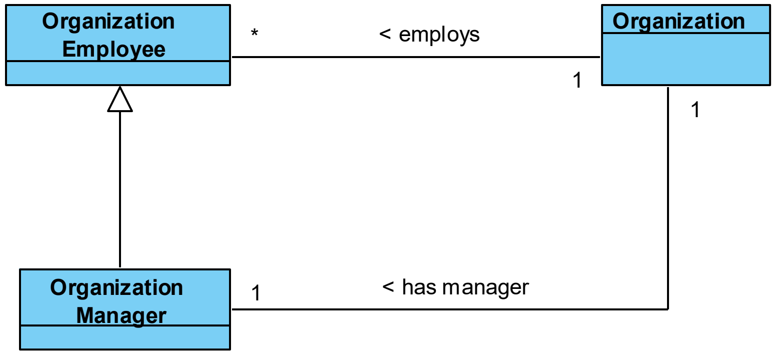
> The platform is expected to be accessed by several users with different roles, such as:
>
> - **Administrative**: these are responsible for managing the platform.
> - **Organization Manager**: person appointed as manager of the organization when registering such organization on the platform; it is assumed to be an employee of that organization, being responsible for specifying other employees of the same organization on the platform.
> - **Organization Employee**: someone acting on behalf of a particular organization; employees are responsible for specifying and cataloging tasks for later publication by the organization.
> - **Freelancers**: people who propose to carry out tasks published by organizations.

First scenario

- Considering the current domain/business, which alternative is more suitable?

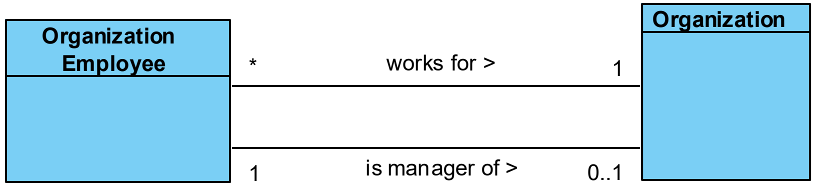


or... ?

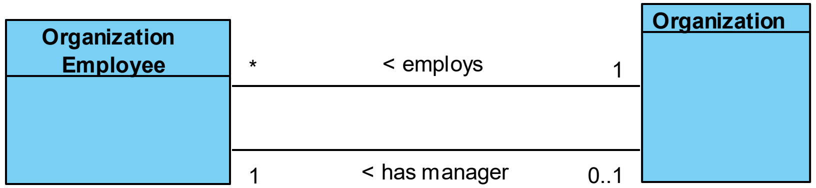


Second scenario

- Considering the current domain/business, which alternative is more suitable?

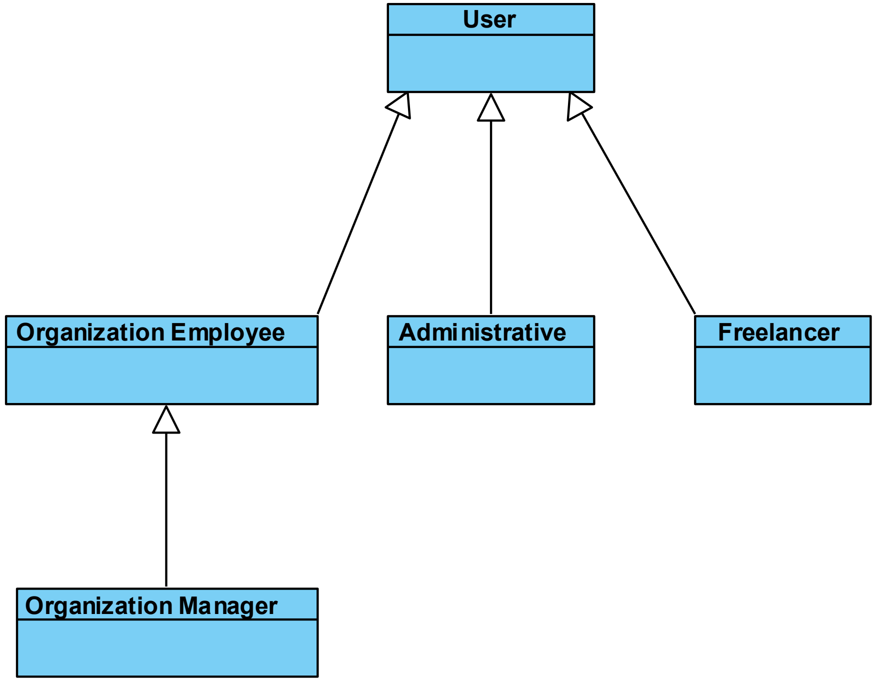


or… ?



Third scenario (1/2)

- Considering the current domain/business, can the same user be Administrative and Freelancer? Does this model support it?

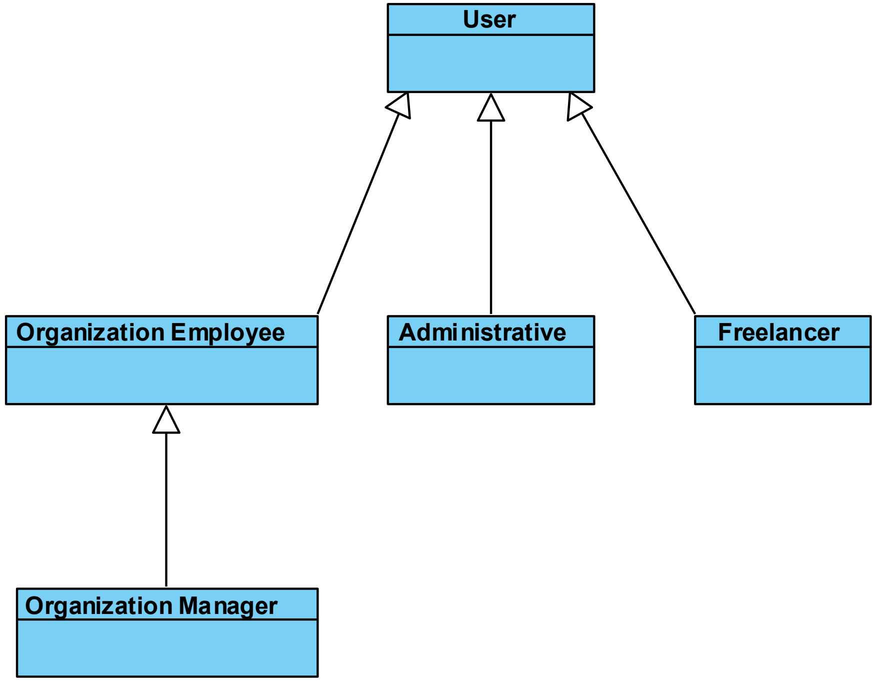


Third scenario (2/2)

- Considering the current domain/business, can the same user be Administrative and Freelancer? Does this model support it?

- Answer: **No!** Two distinct objects would be needed to represent the same real-world user.

- More about Modeling Roles in the *References & Bibliography* section



Generalization – OO Analysis vs. OO Design

- The use of generalization in the Domain Model (i.e. in Analysis) does not imply its adoption in the SW Design or Coding
  - There are other approaches (e.g. the use of interfaces)

- In OO Design, other situations for applying generalization (or polymorphism) can be identified (e.g. *implements* vs. *extends*)

Bibliography

- Fowler, M. (2003). UML Distilled (3rd ed.). Addison-Wesley. ISBN: 978-0-321-19368-1
- Larman, C. (2004). Applying UML and Patterns (3rd ed.). Prentice Hall. ISBN: 978-0-131-48906-6
- Mossé, F. G. (2008). Modeling Roles. Available on: https://objectdiscovery.com/solutions/publications/roles