



Modeling elements

MSc 2008/2009

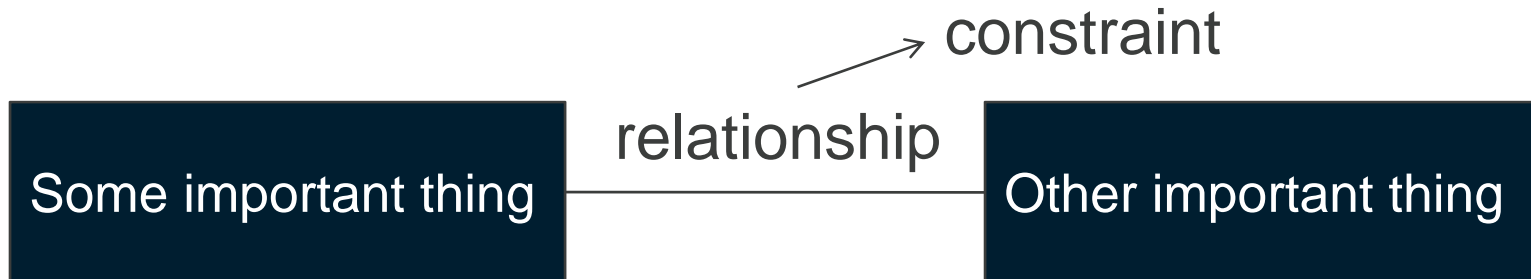
Lecture 3/4

- **Introduction and motivation.**
- Conceptual modeling.
- Data and process models.
- General principles.
- Model engineering.
- Modeling languages.
- Use cases.

- **Modeling elements.**
- General principles.
- Basic elements and their properties: entities, attributes, associations, specialization/generalization.
- Relationships and their characteristics.

- Abstraction
 - Ignoring certain aspects in order to simplify the handling of something or to better understand other aspects.
 - The modeler decides what it is important or not and then chooses a representation that is more tractable than the original.
 - A representation of something can not be greater than that something.

- Models should be divisible.
- Components/modules/chunks should be highly cohesive.
- Use informative names.
- Chunks should have low coupling.



The node is a non-trivial thing, easy to find in the domain, with a technological equivalent, with high cohesion and low coupling.

Candidates for nodes:

- steps in flowcharts.
- functions in data flow diagrams.
- states in state machine diagrams.
- things or entities in ER models, knowledge bases.
- classes in OO modes.
- types.

Relationships/associations/reasons/properties/attributes hold between instances of the entities.

Constraints/axioms/restrictions/rules further specify the nature of relationships.

- An entity represents a set of instances.
- An entity should be highly cohesive, precisely nameable, relevant.
- An entity should have a strong identity.



- Interview: talk to subject matter experts.
- Documentation: read what experts have written about the subject matter, read the requirements documentation, read proposals and invitations to tender.
- Observation and reflection.

- Typical candidates for entities: **NOUNS**.
 - But: actors of use cases do not necessarily correspond to entities.
- Other terms as well
 - Gerund: „My eyes glazing over...” ~ withdrawal
 - Verbs: an association which starts to take on attributes and associations of its own turns into an entity: „Officer arrests suspect“.
 - Verbs: events: „Illness episode“.
 - Passive form: re-formulate in active form.
 - No pronouns.

- An entity should represent one thing, all of that thing and nothing but that thing.
- You can prove cohesion by
 - Giving the entity a representative name.
 - Noun (+ adjective, sometimes however also captured as attribute value).
 - Blackmail victim, robbery victim.
 - Blue car, red car.
 - **Cars** is not cohesive.
- Avoid ambiguous terms.
 - Manager, handler, processor, list, information, item, data...
- Instances vs classes of things.
- Naming the entity set or the instance you currently see.
 - Database modeling vs. OO modeling (root objects).
- Identity ~ individuality: entities change values, but are still the same entity
 - Child/Adult: age

- Go out too far vs. going down too far.
- Investigate homonyms and synonyms.
 - Can medicine and drug be considered synonyms?
 - Do they have the same properties/characteristics/attributes/relationships?
 - Do they have a critical mass of commonalities?

- Two types of principal characteristics:
 - Measurable properties: attributes.
 - Inter-entity connections: associations.
- Use associations to model something with an identity.
- Arrest details as attribute of the suspect vs. Arrest as an association.
 - Do we measure degrees of arrestedness or do we want to be able to distinguish between arrests?
- Color of an image as attribute vs. entity.
- A „pointing finger“ rather than a „ruler“ indicates identity.

- An attribute is a measurable property of an entity.
 - Scalar values: choice from a range of possibilities.
 - An attribute is NOT a data structure. It is not complicated to measure.
- Value of attributes: integer, real numbers, enumerations, text.
- Attributes do NOT exhibit identity.
- Attributes should have precise representative names.
- Public vs. private (OO).
- Derived attributes.

Witness

```
name:text  
age: integer  
eyesight:  
enum{...}
```

- Interview: talk to subject matter experts.
- Documentation: read what experts have written about the subject matter, read the requirements documentation, read proposals and invitations to tender.
- Observation and reflection.
- Nouns in „-ness“
 - Velocity-ness, job-ness, arrested-ness...
- „How much, how many“ test.
 - If you evaluate this, then it is probably an attribute.
 - If you enumerate these, it is probably an entity.
- Do NOT collapse one-to-one relationships.
- Status attributes are problematic because of open-ended range or fixed, but very large possible values, or because of complex state dependencies.

- Age abstracted as an integer.
- Latitude and longitude: real numbers/NSEW.
- Names abstracted as text.

- Associations are relationships where entities instances are aware of, and characterized by, other entity instances.
- Characterizing relationships vs. use relationships.
 - The former are part of the domain model.
- Properties of associations: reflexivity, cardinality, functional, inverse-functional, discontinuous multiplicity, many-to-many, all values from, some values of, transitivity, symmetry etc.
- Arity
- Derived associations.
- Associations can stand for any way in which one entity can be characterized by another. Label them.

Examples

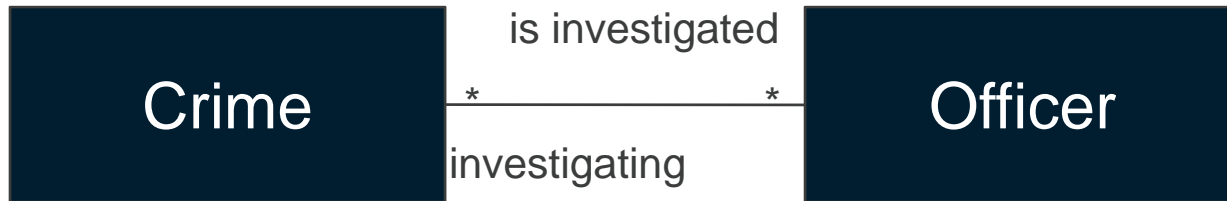


- Interview: talk to subject matter experts.
- Documentation: read what experts have written about the subject matter, read the requirements documentation, read proposals and invitations to tender.
- Observation and reflection.
- Verbs, verbal phrases and things that could have been verbs.
 - „*The butler murdered the duchess*“

- *Never draw more than three associations without going back and starting to label them.*
- The most common way to label associations is with role names.
- Nouns, adjectives.
- Verbs: indication of time's passing.
 - Short-term, one-to-one associations should be named with present participles.
 - Longer-term, one-to-many associations should be named with past participles, or with the simple present third-person singular.

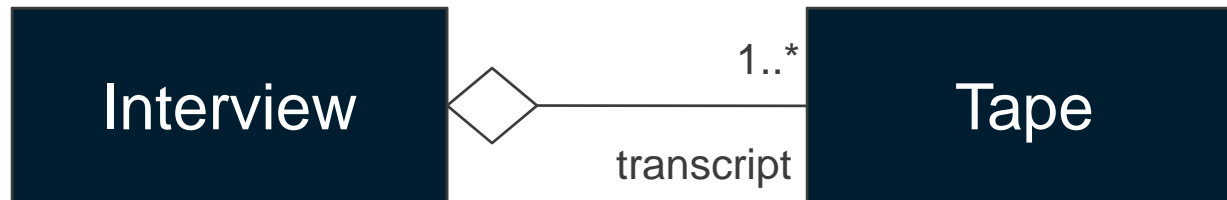
Examples





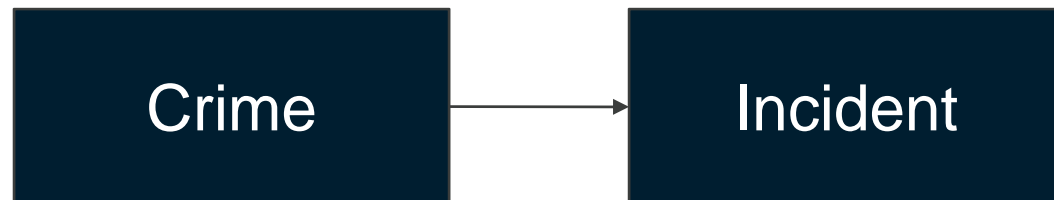
- Aggregations are relationships where entities instances are aware of, and characterized by, other entity instances.
- Difference between aggregation and association:
 - An association describes a relationship between instances of one or more classes.
 - An aggregation describes a part-whole relationship.
 - Existence-dependence

- Provides a reliable test to distinguish between an aggregation and an association.
- Component is mandatory for the existence of the composit.



- 1..* vs. existence-dependence.

- Classification is a relationship between entities ~ sets of instances.
- It is done to communicate understanding, to reduce redundancy, to improve normalization.
- *Generalization/specialization, typing, is-a-kind-of, is-a, inheritance.*
- Some kind of things is a kind of another kind of thing and yet they couldn't both be said to be instances of the same kind of thing.
- The Liskov Substitutability Principle (LSP): If an instance of the supertype were expected and an instance of the subtype turned up it would be acceptable, suitable or substitutable in most contexts.
- *Is the ellipse a kind of circle or a circle a kind of ellipse?*
- *Is a warrant an arrest?*



- Top-down, bottom-up, middle-out.
- Are all instances of entity A also instances of entity B?
- Are all A's also B's?
- Roles.
- Difference between classifications, associations, and aggregations.

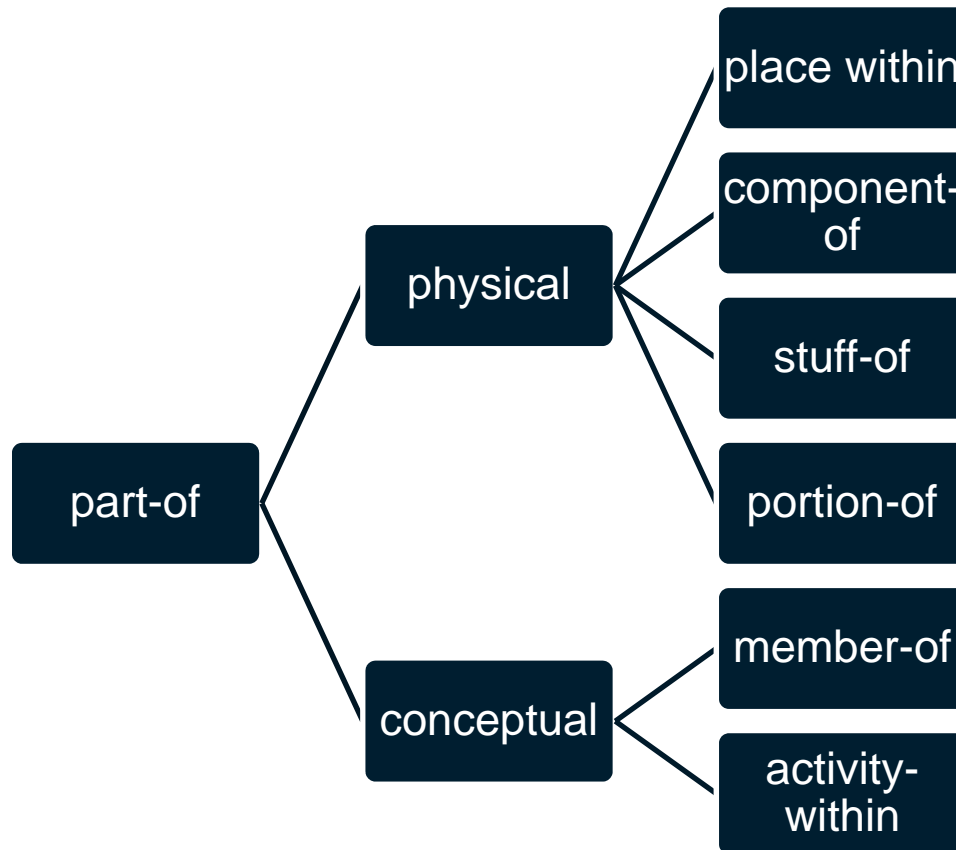


BASIC RELATIONSHIPS AND THEIR PROPERTIES

- Capture the intuitive semantics of the is-a relationship.
- Functional inclusion
 - A chair is-a piece of furniture.
 - A hammer is a tool.
- State inclusion
 - Polio is a disease.
 - Hate is an emotion.
- Activity inclusion
 - Tennis is a sport.
 - Murder is a crime.
- Action inclusion
 - Lecturing is a form of talking.
 - Frying is a form of cooking.
- Perceptual inclusion
 - A cat is a mammal.
 - An apple is a fruit.

- Essential properties of an entity are inherited by its subentities.
- All instances of a subentity are instances of the superentity.
- Reflexivity.
- Antisimmetry.
- Transitivity.

- Capture part-of relationships.



The Alps are part-of Europe

A chapter is part of a book.

A chair is partly wood.

A yard is part of a mile.

A tree is part of a forest.

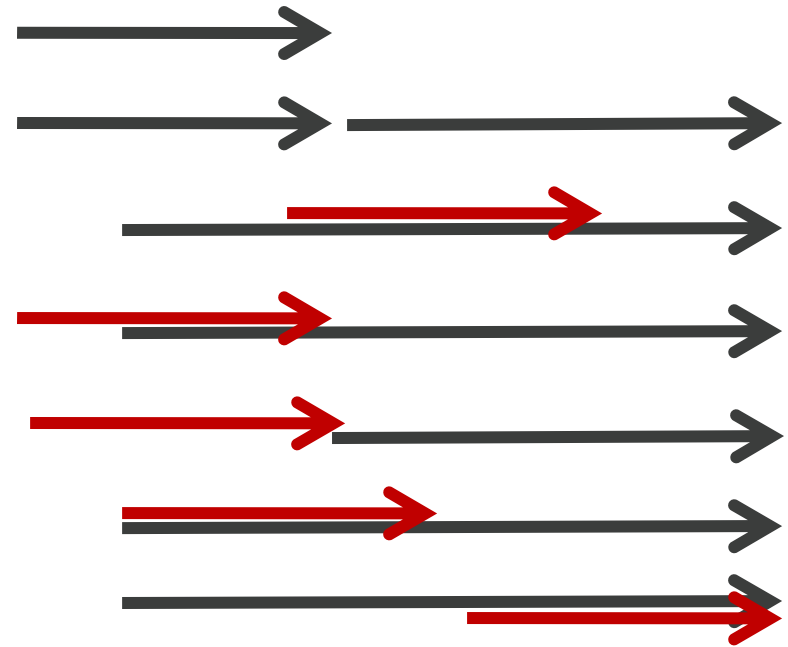
Paying is part of shopping.

- Irreflexivity.
- Asymmetry.
- Transitivity.

relationship	separable	functional	homeomeric
place-within	-	-	-
component-of	+	+	-
stuff-of	-	-	-
portion-of	+	-	+
member-of	+	-	-
activity-within	-	+	-

Temporal relationships


- X equals Y.
- X before Y, Y after X.
- X during Y, Y contains X.
- X overlaps Y.
- X meets Y.
- X starts Y.
- X finishes Y.



- Irreflexivity.
- Asymmetry.
- Transitivity (before, during, starts, finishes).

Spatial relationships

• left-of, right-of. 

• above, below. 

• in-front-of, behind. 

• inside. 

• outside. 

• between. 

• far. 

• near. 

• touching. 

• beside. 

• disjoint 

• intersect 

• coincident 

- Reflexivity (coincident).
- Irreflexivity (all others).
- Symmetry (near, far, beside).
- Asymmetry (left-of, right-of, above, behind, inside).
- Transitivity (left-of, right-of, above, behind).
- Mutual exclusivity (examples: below, coincident, inside; touching, far; intersect, inside).

- depends-on-causally.
- depends-on-existentially.

- Transitivity.

- They differ from other families of relationships in that they do not depend solely on the nature of the meaning of the entities they relate.
- Agent-Action. Dog-bark; artist-paint.
- Agent-uses-Instrument. Skier-uses-skis.
- Agent-Object. Bakes-uses-flour.
- Action-Recipient. Laydown-bed.
- Action-Instrument. Paint-brush.

- Irreflexivity.
- Asymmetry.

- General principles.
- Basic elements and their properties: entities, attributes, associations, specialization/generalization.
- Relationships and their characteristics.

- Object-Oriented Analysis and Design (John Deacon, Addison-Wesley).
 - Chapters 5-8.
 - <http://www.johndeacon.net/OOAandD/index.asp#TutorLecturerResources>
- IDEF5 Method Report:
<http://www.idef.com/pdf/Idef5.pdf>

- a) Describe the automotive domain using 10-20 entities, attributes and relationships.
- b) Model the following statements.
- Barack Hussein Obama is the nominee of the Democratic Party for the office of President of the United States in the 2008 general election.
 - Peter saw Van Gogh's sunflowers in an MOMA exhibition at the Louvre in December last year.

- c) In the context of financial accounting what might Customer, Investment, Cruise Ship and Tanker have in common? In the context of a mathematics package, what might Circle and Ellipse have in common? In the context of a fighter-pilot training simulator, what might Bird Flock, Detonation and Cannon Round have in common?

Thank You!

Questions?