Domain Models

Prof. Dr. Robert Mertens

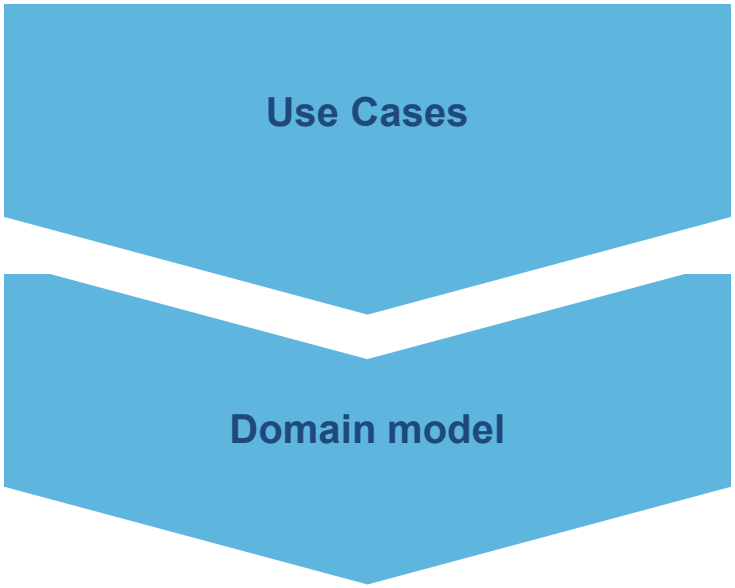
SoSe 2020

Introduction an Background

Introduction

… or: „What role do domain models play?“

- Most important model in OO analysis
  - Use cases are not OO
  - OO beacuase ob objects in model
- Inspiration for software objects
  - In Design
  - In domain layer
- Optional n agilen approaches

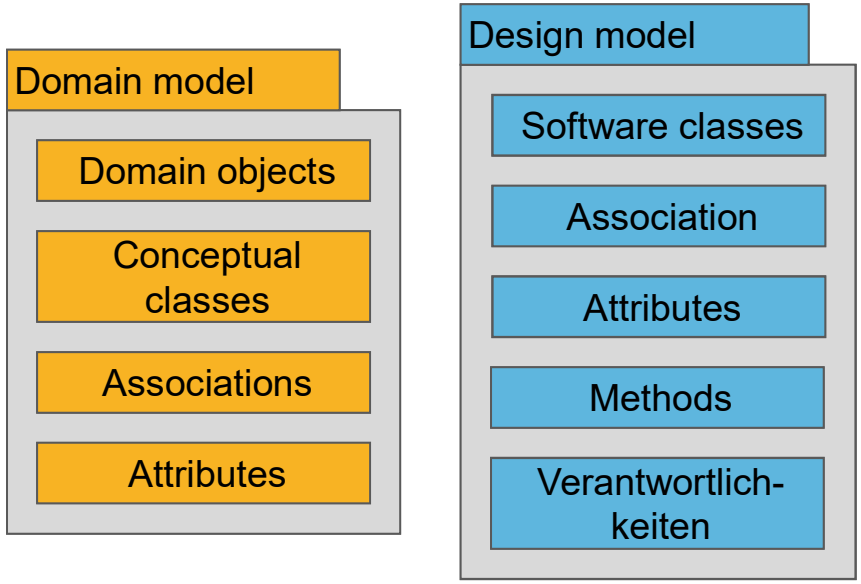


„Which objects belong in the domain model is determined by the use cases.“

Definition Domain model

… or: „What separates domain model from domain layer?“

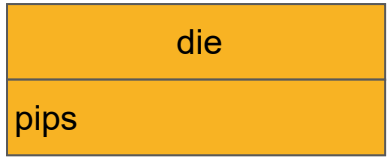
- Visuall Representation
  - Real objects from domain
  - Conceptual classes
  - Visuelles Vokabular
- Domain model ↔ domain layer
  - Concept perspektive
  - Software perspektive
  - Class diagrams used for both
- Domain model ↔ data modell
  - Also Objects without attributes possible
  - Objects with only behavior possible



Symbol, Intension, Extension

… or: „Model vs. reality – what is the difference?“

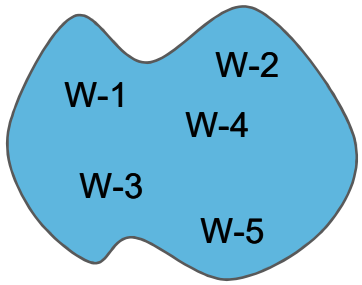
- Models are abstractions
  - Only the relevant aspects
  - All relevant aspects
- Terms from logic
  - Symbol
  - Intension
  - Extension



„Class diagrams a a form of symbolic representation. “

„A die has six sides which are numbered on through six.“

„‚Intension‘ means a definition.“

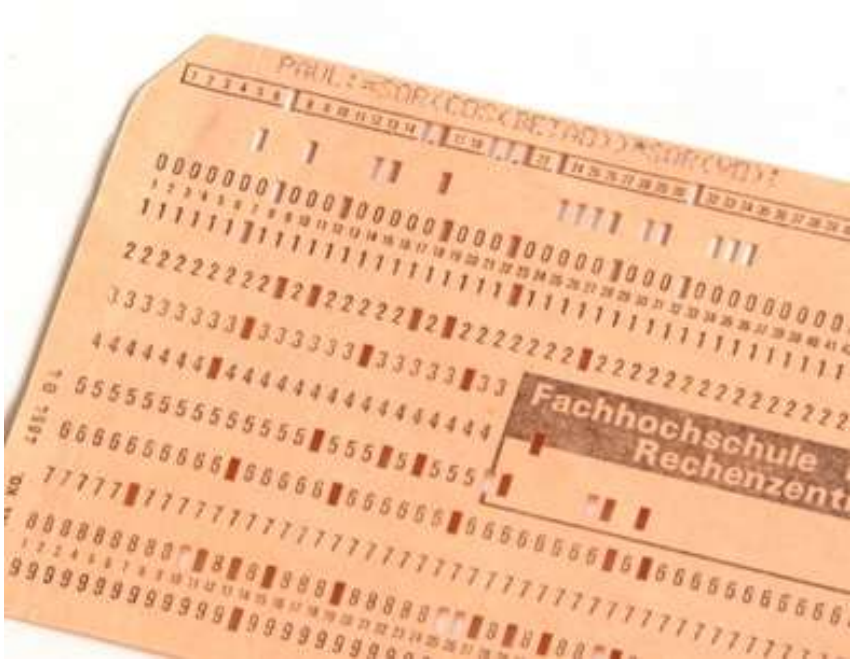


„‚Extension‘ means all positive examples in the world.“

Motivation

… or: „What is d domain model good for?“

- Understanding the domain
  - Key concepts
  - Vocabulary of domain experts
- Bridging representational gaps
  - Understanding program logic
  - Program changes
  - Maintenance



„In the early years even programmers struggled when trying to understand programs. Similar Names and structures in both worlds facilitate understanding. “

Modeling

Recommendation: Model in three Steps

… or: „How to build a domain model?“

1. Find conceptual classes
2. Draw class diagrams
3. Add associations and attributes

„Building a domain model is a creative process. Discussions and sketches play an important role. “

Modeling

Finding conceptual classes

How to Identify Conceptual Classes

… or: „Where do conceptual classes come from?“

Strategies

Existing Models

- Reusable
- Have to be adapted
- Method is easy and proven
- Book: Analysis Patterns, David Fowler
- Book: Data Model Patterns, David Hay
- Book: Model Resource Book (Band 1&2), Len Stevenson

Category lists

- Standard categories as inspiration
- Most important categories first
- Can be combined with nominal pharases method

Nominal phrases

- Linguistic analysis
- Starting point: Uses cases, additional documents, expert knowledge
- Result: conceptual classes, attributes
- Ingnore classes out of focus
- Problem: Ambiguities and synonyms

Example Category List (Part I)

… or: „Focus on business applications!“

Business transaction	This is what it is all about, the money is here	Sale, reservation
Transaction line item	Part of the business transaction	Sales line item
Product or service	Transactions are for something (for instance a product or service)	Item
Transaction records	Where are thre transactions recorded?	Ledger, register
Actors	All those involved, organizations, too.	Cashier, customer
Place	Where the transaction takes place.	Store
Ereignisse	Often additional info like location and time is needed.	Sale, reservation
Physical object	Especially important for device control drivers or simulations	Item sold, register, cash
Descriptions	Describes the product or services	Product description
Catalogue	Descriptions can be found there	Product catalogue

Example Category List (Part I)

… or: „More categories.“

Container	For things or information	Store, Container
Things in container	Often what the transaction is about	Product
Neighboring systems	Systems thar are part of the transaction	Payment authorization servise
Records	Records for payments, work, contracts, legal matters	Maintenance protocol
Financial instrument	What payment is used?	Cash, credit line
Secheduls, manuals	Documents used to structure work	Maintenance plans, change lists

„Redudancies are there on purpose. His lowers the risk of forgetting something important. “

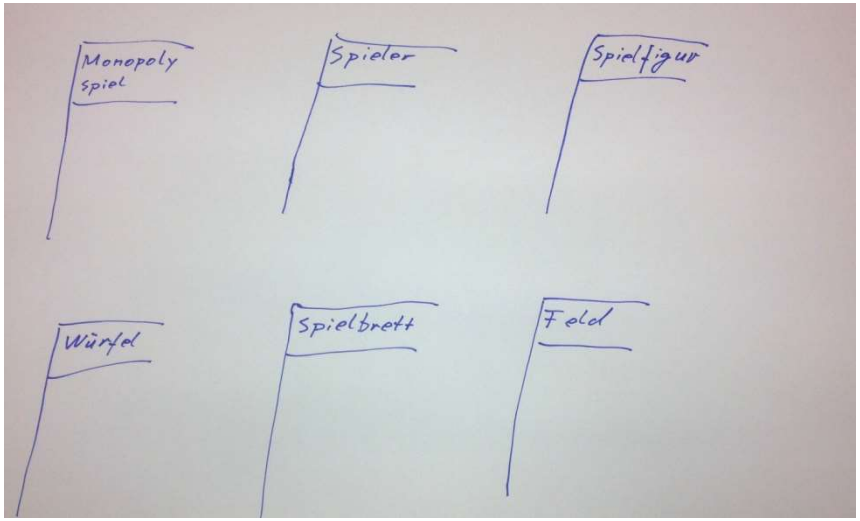
Modeling

Drawing class diagrams

Example Monopoly

… or: „How exactly is modeling done?“

- Brainstorming directly in UML
- Classes can be extendet
  - Leave space
  - Right and bottom are left open
- Not complete at first try
  - More domain classes in design
  - More in development



„Non-electronic media like whiteboards are good tools for this step, too.“

Reports as Objects

… or: „Does double always mean redundant?“

- Generally not advisable
  - Information is already stored
  - Duplicates lead to problems
- Special roles change the rules
  - Receipt for money-back
  - Warranty

„receipts store redundant information but they act in a double role for additional puposes. “

Analogy Cartographer

… or: „What should be in the model?“

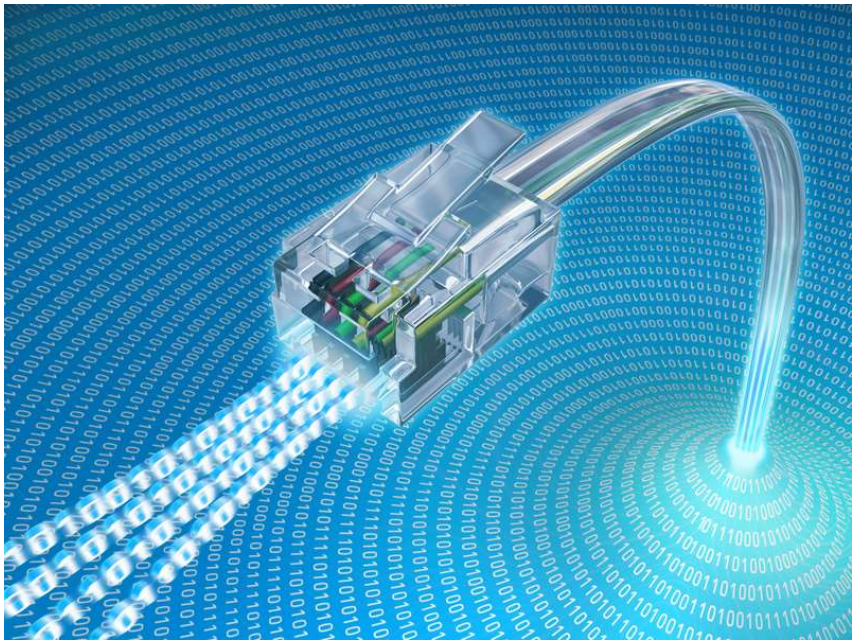
- Facilitate orientation
  - Names in local language
  - In model: expert‘s terms
- Only relevant information
- Do not add anything

„Old maritime maps often showed sea monster. Did they realy exist? “

Non-physical entities

… or: „How to model a port?“

- No natural analogy
- No similar models
- Ask the experts
  - Vocabulary
  - Concepts



„The telecommunications domain is a good example: terms like communications protocol or connection can hardly be fathomed without the help of experts.“

Attribute or class

… or: „When to use an attribute and when to use a class?“

- Not just a number or text
- Real entities
  - Organization, too
  - Legal entities
- Having physical properties
  - Space
  - Mass

„A store in the real world is not only a numer or a string. Hence, it might be a good idea to represent it as a class with attributes.“

Description classes

… or: „How to sell something that does not yet exist?“

- Concrete object does not exits
  - New products in marketing
  - Custom made products
- Description as a template
  - For transport, storage etc.
  - For variable components
- Advantages
  - Reduces redundancy
  - Instances can be deleted without losing information

„Desription classes can be used to describe products or services – even when they not not yet or no longer exist.“

Modeling

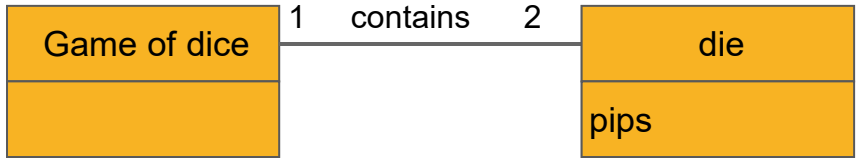
associations and attributes

Associations in the Domain Model

… or: „What exactly is meant by *association*?“

- Linking instances
  - Information need
  - Understanding of domain

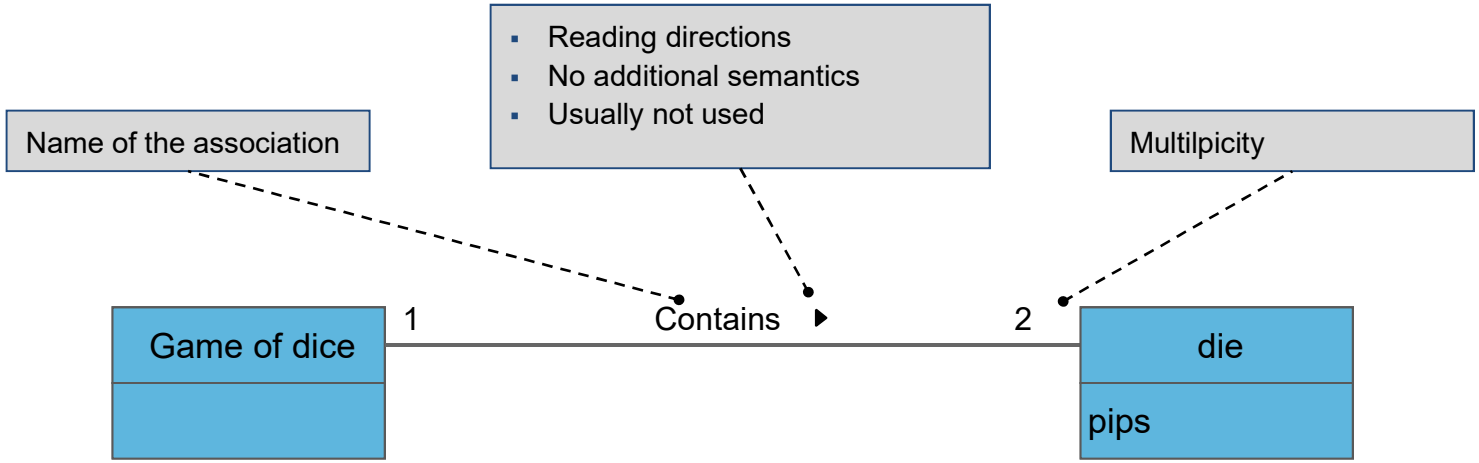
- Domain model ↔ design model
  - Associations from the real world
  - Not: documenting object structure
  - Associations are not directed



„An association links two classes.“

UML Notation for Associations

… or: „What do they look like?“



„The reading direction does not carry any additional information. In the domain model all associations are bidirectional. “

Criteria for Associations in a Diagram

… or: „Which associations make it into the model?“

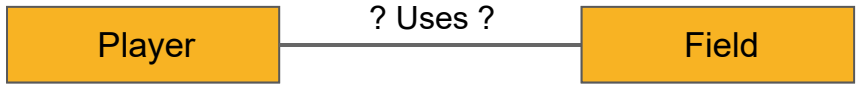
- Information is required
  - Monopoly: What lot am I on?
  - Monopoly: Whow does it belong to?
- Persistent associations
  - Even for short periods of time
- List of common associations
  - Helps finding assciations
  - Good indicator

„The number of pips determins the target of moving a token. However, the informations is only used within the turn, hence it does not need to be persisted. “

Naming Associations

… or: „Every name tells a story!“

- Telling names
  - informative
  - Readable (Class-Verb-Class)
  - As exact as possible
- Writing conventions
  - Hyphen
  - CamelCase
  - Always: first letter capital

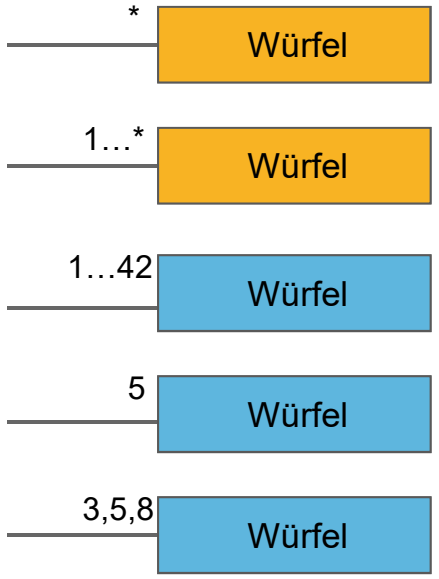


„Verbs like ‚has‘ oder ‚uses‘ are often ambiguous Itthe player‘s token on a Feld or does he or she own it?“

Rolls (Syntax)

… oder: „What do the numbers mean?“

- Multiplicity
  - At any time
- Name
- Navigation direction
  - Not used in domain model

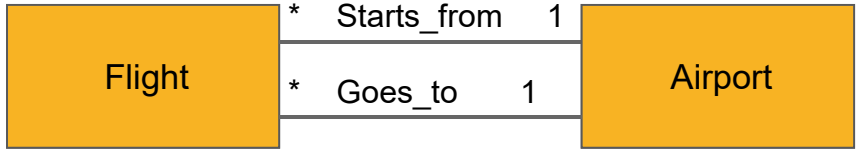


„Choosing multiplicity dpends on the intentions of those using them. Here limits can be set that guide software develoment.“

More than one Association Between Classes

… or: „Is that possible?“

- Not uncommon
- Use it for independent relations
- Simple syntax



„In case of independent associactions each one is represented by ist own connecting line with a separate role.“

List of Common Associations

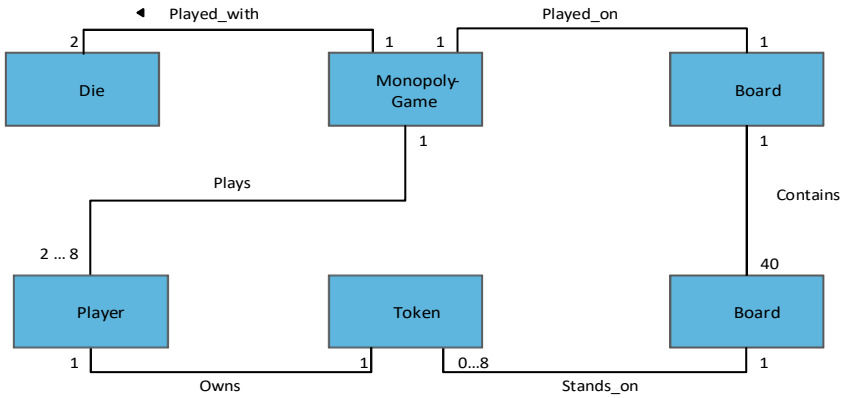
… or: „Do we have a template for these as well?“

Transaction A is connected to tansaction B	Payment—sale, reservation—cancellation
A is a line item of Transaction B	SalesLineItem—sale
A is produkt for transaktion (or line item) B	Product—LineItem, seat—reservation
A is a tole related to transaction B	Customer—payment
A is part of B (physical or logical)	Square—GamingBoard, seat—airplane
A is contained in B (physical or logical)	item—shelf, square—GamingBoard, seat—airplane
A is a description of B	ProductDescription—product
A is known/logged/recorded/reported in/to B	Sale—register, token—GamingBoard
A is member of B	Player—game, cashier—store
A is an organizational subunit of B	department—store, maintenance—airline
A uses/manages/owns B	Cashier—register, player—token, pilot—airplane
A is next to B	LineItem—LineItem, square—square

Example: Monopoly

… or: „How to use the List?“

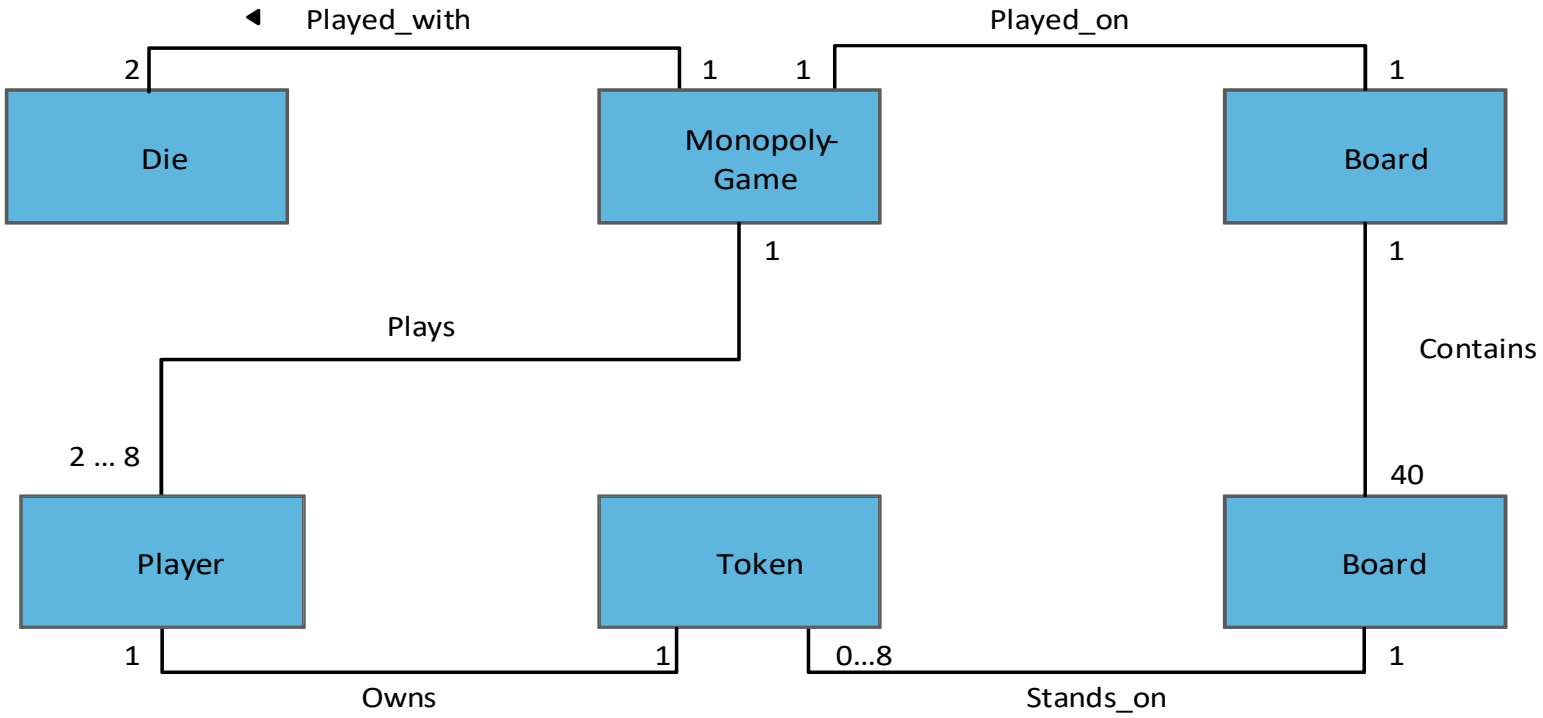
A is a tole related to transaction B
Played_with
Played_on
A is part of B (physical or logical)
Continas
A ist in B (physisch oder logisch)
Contains
A is known/logged/recorded/reported in/to B
Stands_on
A is member of B
Plays
A uses/manages/owns B
Owns



„Associations can by found with the list. The next step is giving concrete names to the associations. “

Example: Monopoly - Result

… or: „Where does it lead to?“



Attributes come from Requirements

… or: „What attributes to model?“

- Information for process steps
- Information that needs to be stored
  - Debugging
  - Logging
- Example: Use Case *Sale*
  - Sale → date/time
  - Store → name and adress
  - Cashier → ID

„A receipt for a sale containts a wealth of information: time, name and adress of the store, cashier ID, etc..“

Attribute Notation

… or: „What does it look like in the design phase?“

Complete syntax:	Visibility name : type multiplicity = default {property}

Visibility

- Private: -
- Protected: #
- Public: +
- Package: ~

Type

- Date
- Money
- String
- …

Multiplicity

- *
- n…k
- n
- n,k

Default

- Concrete value
- Initialization
- Can be overriden

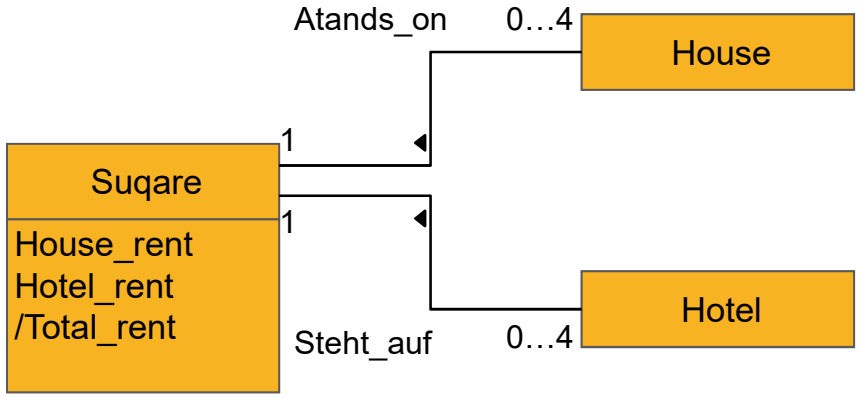
Property

- {readOnly}
- {unique}
- {readOnly, >5}
- …

Derived Attributes

… or: „What about redundant attributes?“

- Useful information
- Can be derived from other information



„Abgeleitete Attribute können aus anderen Informationen abgeleitet werden, sind aber so wichtig, dass sie als separate Attribute modelliert werden!“

Attribute Candidates

… or: „What separates classes from attributes?“

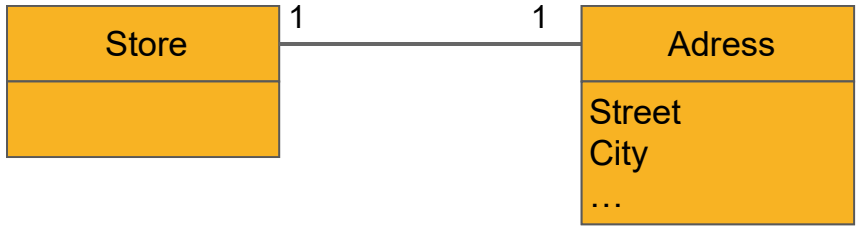
Indicators

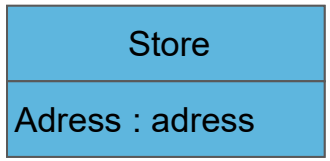
Data-types	▪ String (Text), Boolean, Date, Time, DateTime, Number, Character, etc. ▪ Primitive Datatypes
Very simple objects	▪ Adress, color, telephone number, insurance number, postal code, ISBN, enums, etc. ▪ Objects for which identity is determined by checking values and not object identity ▪ Immutables

Representing Data in the Domain Model

… or: „When to come up with new data types for attributes?“

- Multi part attribues
  - Telephon number, Names of persons
- Comes with operations
  - Validation for insurance number
- Comes with additional attributes
  - Start and and date for special offer
- Values with units
  - Currency for money
- Abstractions
  - Encoding scheme (e.g. chicken egg)





„Both models are okay: The more important the role of the complex attribute is, the more sense it makes to model them as classes.“

Recommendation: Model Units and Currencies

… or: „5000??? In Euros or in Cents?“

- Do not forget units
- Quantities as Classes
  - Units as attributes
  - Units as associations
- Example: Money
  - Amount
  - Currency

„Numbers make little sense without units!“

The perfect Domain Model

… or: „Does it exist?“

- Forgot requirements
- Domain changes oder time
- More information needed in design
- Reaction to changes depending development process model

„A domain model is heavily influenced by the requirements. When they change so does the model. As long as it helps in design the model is good.“


Thank you for your attention!