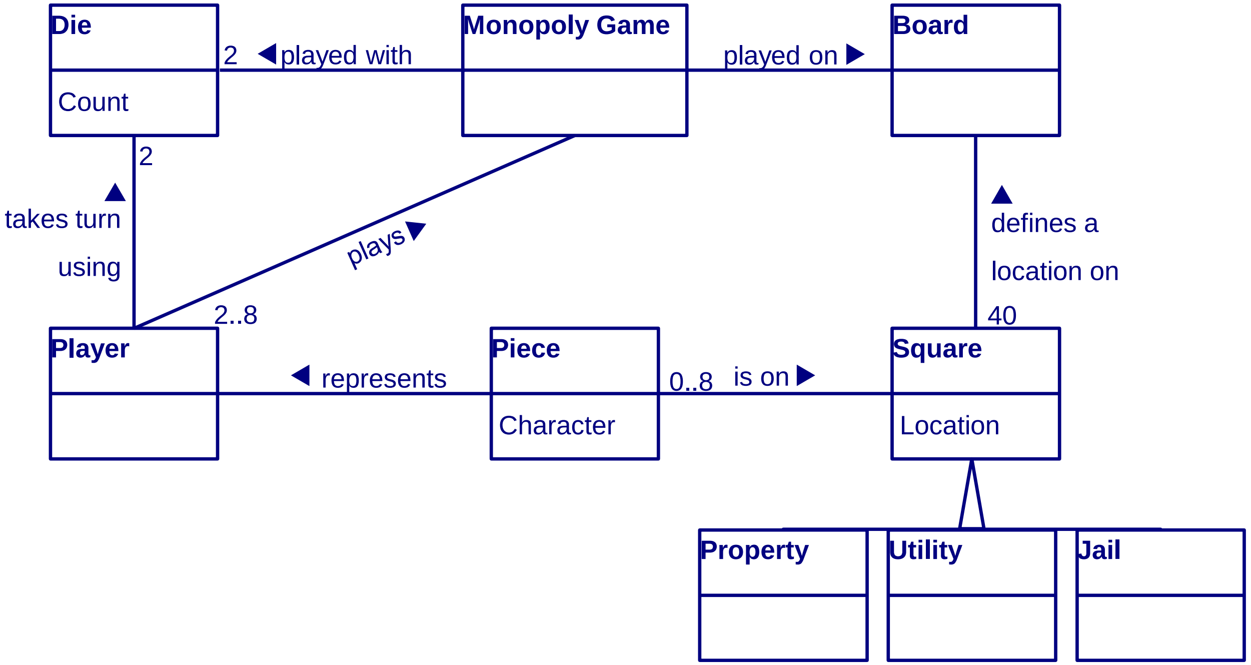
Domain Analysis



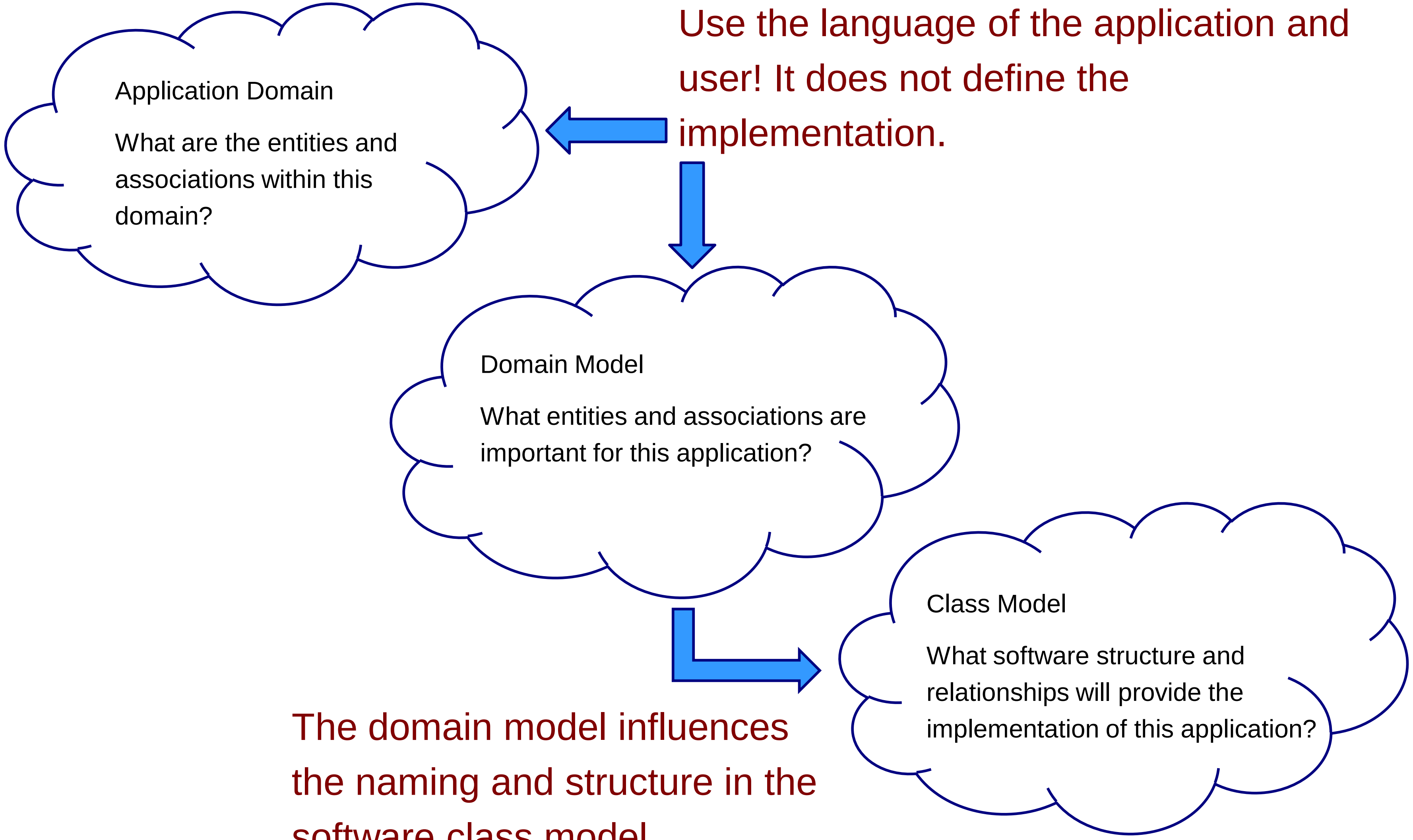
SWEN-261
Introduction to Software Engineering

Department of Software Engineering
Rochester Institute of Technology

The domain for a software system defines the context in which the software operates.

- This is also referred to as the *application domain*.
  - ***Retail sales***
  - ***Banking***
  - ***Customer contact management***
  - ***Checkers playing***
- The domain model describes the ubiquitous world in which the system's experts and users exist and work on a daily basis.
  - ***Domain entities***
  - ***Domain language***
  - ***Associations/relationships between entities***

Domain analysis provides an understanding of the application problem space.



Application Domain

What are the entities and associations within this domain?

Use the language of the application and user! It does not define the implementation.

Domain Model

What entities and associations are important for this application?

Class Model

What software structure and relationships will provide the implementation of this application?

The domain model influences the naming and structure in the software class model.

The domain model identifies important aspects of the application not the implementation.

- Only use vocabulary from the problem statement
  - ***For example, a unique identifier needed to store data with no meaning to the user would not be in a domain model.***
- Establishes a common understanding of the problem for customer/user and software team

Domain model definition starts with an analysis of the nouns in the domain.

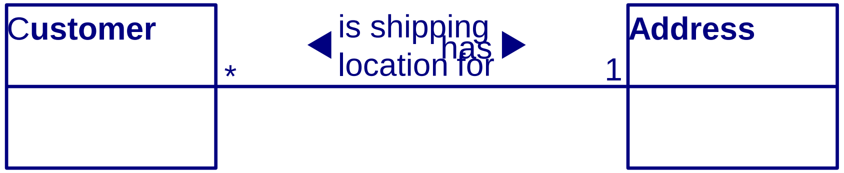
- The steps in the noun analysis include
  - ***Identify the nouns in the problem statement and language of the domain experts and users.***
  - ***Identify any words that might be specializations of other nouns.***
  - ***Identify any nouns that might be attributes or properties of other nouns.***
  - ***Identify any other associations between nouns.***

The domain model is typically drawn using a simplified class diagram notation.

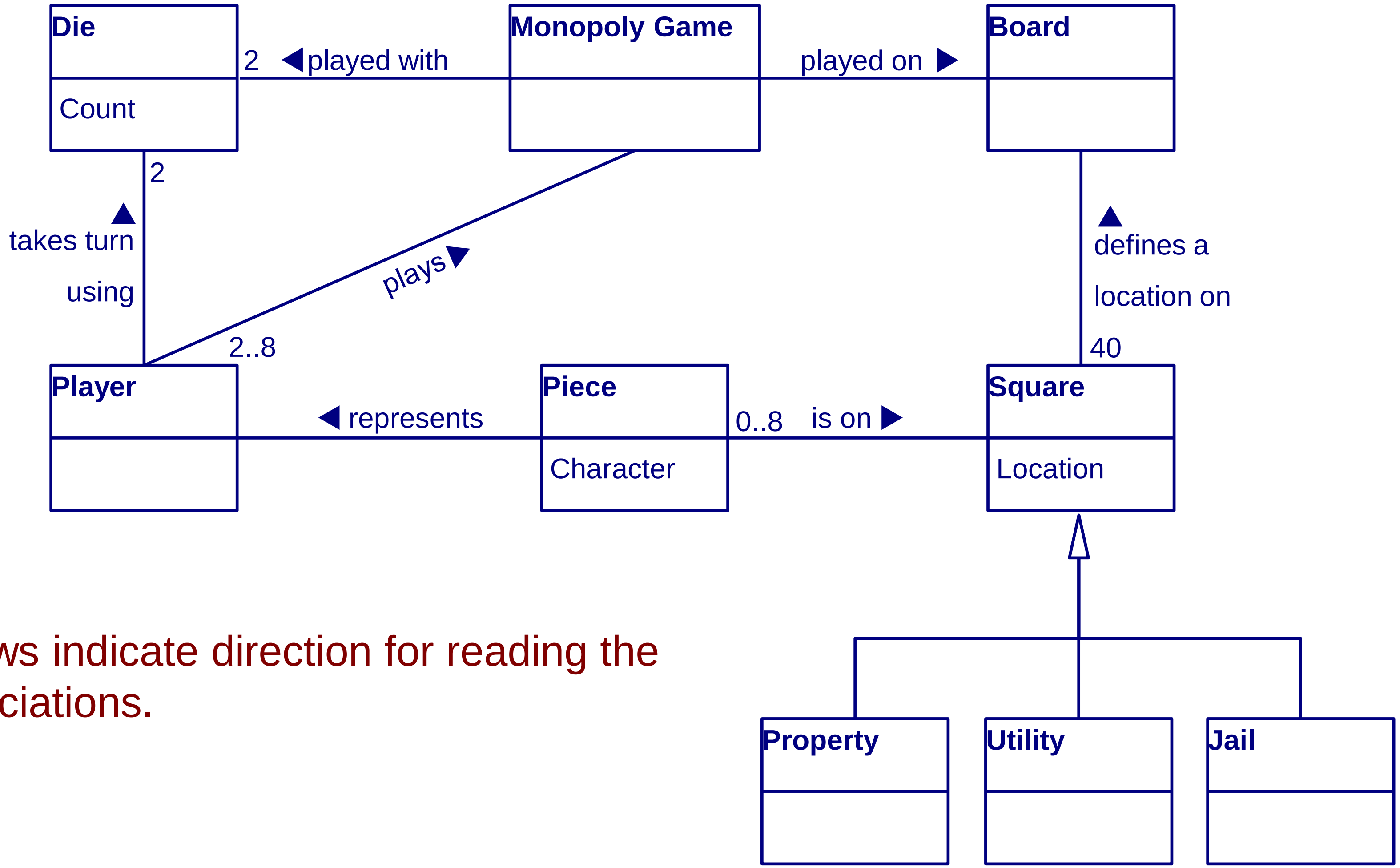
- Show the following information
  - ***Domain entities***
  - ***Attributes in domain entities***
  - ***Associations between domain entities***
- Use user vocabulary
  - ***Attributes do not indicate data type***
- Associations come from the problem statement
  - ***Place label on the association line***
  - ***Usually completes a phrase between two domain entities: DE_1 association DE_2 (LineItem records-sale-of Product)***
  - ***Indicate multiplicity, if known***
  - ***Use inheritance, if appropriate***

An association should describe the relationship between two domain entities.

- All associations should have an arrow to indicate the direction to read the association.
- Use the active voice for the verb when possible.
- An association of "has" or "contains" does not describe much about the relationship.
  - ***Reverse the direction and rephrase the association***



This partial domain model for a game of Monopoly demonstrates these ideas.



Arrows indicate direction for reading the associations.

Domain analysis continues through the project.

- The domain model continues to evolve as you learn more about the project.
  - ***Working on the project gives you a different understanding of the domain.***
  - ***New features change your understanding of the domain.***
  - ***When user stories are refined during backlog refinement more details may come out about the domain.***
- Keep your domain model up-to-date so that there is always a common understanding between the development team and Product Owners.