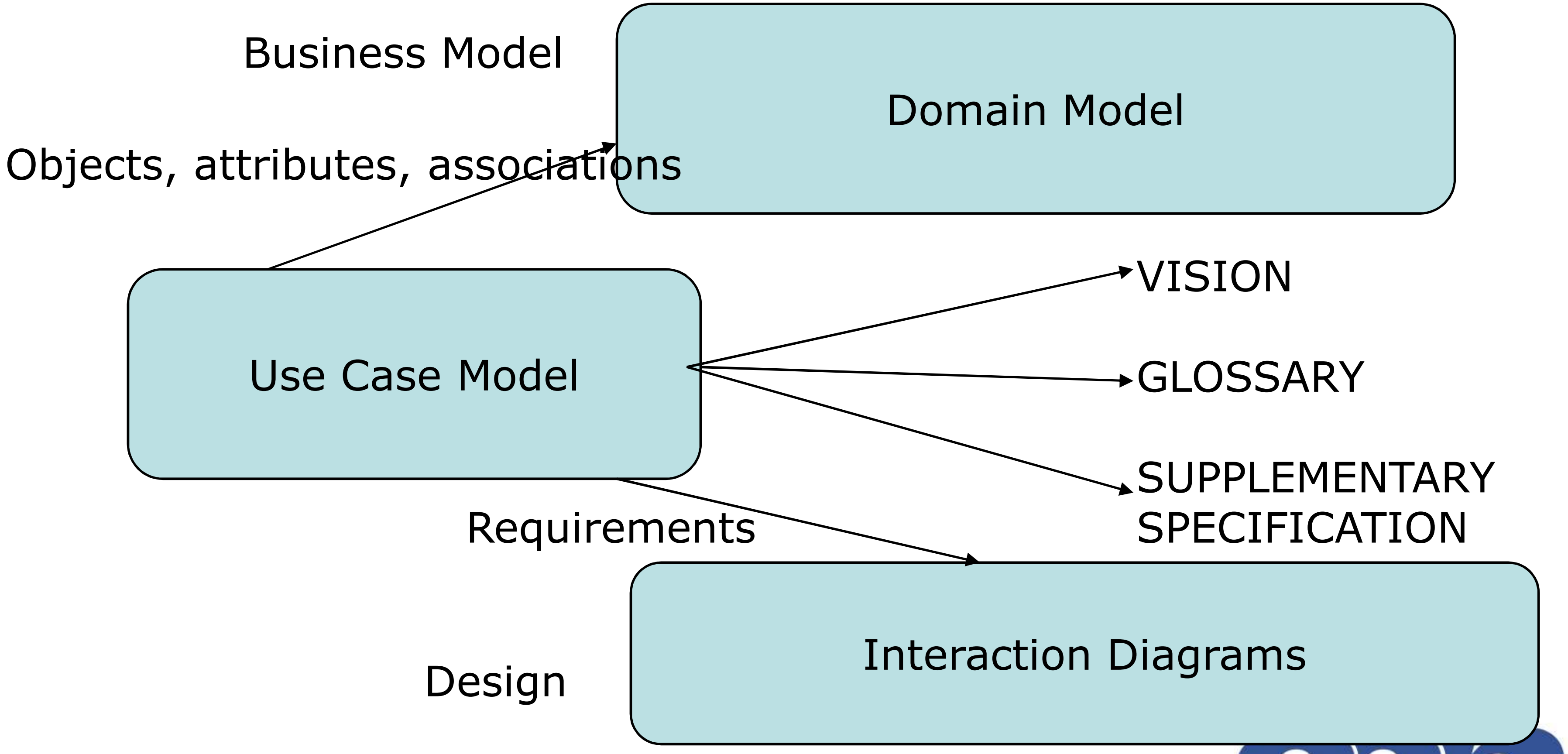
Use Cases

Define the Problem

- The most critical question:
- “Is this the right system to make?”

Use Case Relationships



Business Model

Domain Model

Objects, attributes, associations

VISION

Use Case Model

GLOSSARY

SUPPLEMENTARY SPECIFICATION

Requirements

Interaction Diagrams

Design

Use Cases are not Diagrams

- Use Cases may have a diagram associated with them, and a use case diagram is an easy way for an analyst to discuss a process with a subject matter expert (SME).
- But use cases are primarily text. The text is important. The diagram is optional.

Emphasize Goals

- Investigating goals rather than tasks and procedures improves information gathering by focusing on the essence of requirements—the intent behind them.

- Seeing requirements as identifying tasks to be done has a strong bias toward reproducing the existing system, even when it is being replaced because it is seriously defective.

Why Use Cases?

- Simple and familiar storytelling makes it easier, especially for customers, to contribute and review goals.
- Use cases keep it simple (KISS)
- They emphasize goals and the user perspective.
- New use case writers tend to take them too seriously.

Actors or Use Case First?

- Because you have to understand each part of Use Cases, the parts are presented separately. But those who create use cases switch back and forth. The text describes use cases substantially before paying attention to actors. Typically, both actors and use cases are identified early and then examined to see if more use cases can be found from the actors, or more actors found by examining the use cases.

Identify Use Cases

- Capture the specific ways of using the system as dialogues between an actor and the system.
- Use cases are used to
  - Capture system requirements
  - Communicate with end users and Subject Matter Experts
  - Test the system

Specifying Use Cases

- Create a written document for each Use Case
  - Clearly define intent of the Use Case
  - Define Main Success Scenario (Happy Path)
  - Define any alternate action paths
  - Use format of Stimulus: Response
  - Each specification must be testable
  - Write from actor’s perspective, in actor’s vocabulary

www.usecases.org Template

- Name
- Primary Actor
- Scope
- Level
- Stakeholders and Interests
- Minimal Guarantee
- Success Guarantee
- Main Success Scenario
- Extensions

- This is the basic format used in the text and in Alistair Cockburn's *Writing Effective Use Cases* (Addison Wesley, 2000, ISBN 0201702258).
- I prefer to modify it slightly to use the actor actions and system response in tabular form. Larman calls this the Two-Column Variation.

Optional Items

- You can add some of the following items
  - Trigger (after Success Guarantee)
- (at end:)
  - Special requirements
  - Technology and Data Variations
  - Frequency of Occurrence
  - Open Issues

Use Cases

The first step in getting what you want is to decide what you want.

What they are

- Use cases are text stories (not diagrams!) used to discover and record requirements
- If a diagram clarifies the text, use it

Definitions

- Actor – something with a behavior, such as a person, an input device, etc.
- Scenario – Specific sequence of actions and interactions between actors. (also called a use-case instance)
- Use Case is a collection of related success and failure scenarios that describe an actor using a system to support a goal

Three Formats

- Brief – Terse, one-paragraph summary, usually the main success scenario. Create during early requirements phase.
- Casual – Informal paragraph format. Can cover various scenarios in multiple paragraphs.
- Fully-dressed – All steps and variations written in detail. Has supporting sections, success guarantees, main scenario, alternate scenarios, etc.

Scope

- Defines how broad the use case is. This can be for the whole system, as in the POS example, or narrow, as in a use case for creating a journal entry in an accounting system.

Level

- User-goal: Scenarios that let a user get something done. Corresponds to an elementary business process.
- Subfunction: smaller steps required to support a user goal.

Primary Actor

- The person (or sometimes object) that calls upon system services to fulfill a goal. (When might an actor not be a person?)

Stakeholders and Interests

- The stakeholders are people who have a reason to want this system. The Interests are their reasons for wanting it and what they expect from it.
- You could view the system as a contract between various stakeholders.

Preconditions and Success Guarantee

- These should be non-obvious.
- Preconditions state what must ALWAYS be true before you can start the scenario. This often defines the success of another use case.
- Success guarantees state what must be true on successful completion of the use case.

Main Success Scenario

- This satisfies the interests of the stakeholders. You get your groceries, the store gets your money, inventory is reduced, etc.
- Steps:
  - An interaction between actors
  - Validation (by the system)
  - State change to the system

Extensions or Alternate Flows

- These are usually the majority of the text. Remember Murphy’s Law ☺
- These include all other possible outcomes, both success and failure.

Performing Another Use Case

- Use cases can branch to other use cases. For example, if a POS system rejects a bar code, the cashier can request alternate lookup.
- Denote this by underlining: Cashier performs <u>Find Product Help </u>to get item ID and price

Technology and Data Variations

- Technical variations on how something must be done:
  - Scan bar code
  - Key item ID
  - RFID
- Avoid early design decisions; keep things general.

Finding Use Cases

1. Choose the system boundary
2. Identify the primary actors
3. Identify the goals for each primary actor
4. Define use cases that satisfy these goals

Questions to Find Actors and Goals

- Who starts and stops the system?
- Who does user and security management?
- Who does system administration?
- Is “time” an actor because the system does something in response to a time event?
- How are software updates handled?
- Who gets notified of problems?