

## Announcements

Pick up exercise on table

Hand in new HW up here (is your name on it?)

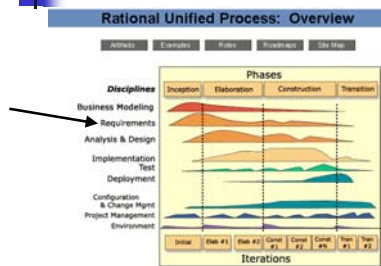
Last time: 3 chapters

This time: part of 1 chapter

## Use Cases for Requirements

- Today's material: Part of chapter 6
- Recall:
  - "Requirements are capabilities...to which the system...must conform" – p. 41
    - This is the author's opinion
    - I prefer to make a clear conceptual distinction between
      - **requirements** and **specifications**
      - The conceptual distinction is important
        - regardless of names (i.e. **requirements, specifications**)
  - Requirements is a UP **discipline** (Fig. 2.4)
  - ...it is done throughout the project
    - (Unlike in the waterfall model)

## Disciplines in the RUP (almost fig. 2.4)



## Use Cases - Background

- A use case is a "case of use"
  - ...that is, (generalized) example of usage
- Precise translation from the Swedish is "usage case"
- Invented by Ivar Jacobson in 1986
- Major advance by Alistair Cockburn in book
  - *Writing Effective Use Cases*, 2001
  - Textbook discussion is based on that book

## What is a Use Case

- A use case is a set of system use "stories"
- As a way to communicate among humans...
  - Stories are very flexible
  - Stories are a human-friendly
- Thus, use cases are a good way to communicate system needs
- Use cases are stories (not diagrams)
  - Use case *diagrams*: nice, but not the main point

## Use Cases for System Needs

- Because use cases are human friendly
  - ...they are a good way to capture needs
  - They are stories about how a system can meet needs
- Example (POS system):
  - Customer brings items to checkout counter. Cashier records each item. System displays each item & price, and current total. Customer provides payment somehow. System validates it, logs it, and updates inventory data. System prints receipt and customer waves a cheery goodbye. (p. 46)

## Stories can Have Titles...

- The title of that use case could be
  - "Process a customer with items to buy"
- Consider the registration system
  - A title of a use case might be
    - "Process a student with courses to sign up for"
- Consider a general UML editor system
  - Write down a title for one use case...
  - (We'll have more to say about these in the future)

## Use Cases are Appropriately General

- Too specific: Joe Schmoe goes to the grocery store at 11:00 a.m. on Thursday to buy stuff for breakfast. In his basket he puts beer, potato chips, and a copy of the book "Healthy Breakfasts." Then he...
- A use case instance:
  - one path through a use case.
  - Example: someone goes to the store, puts items into a cart, proceeds to the cash only express line...
  - Another example: . . .
- (A scenario: same as a use case instance)
- A use case: set of "related...scenarios"

## Use Case – What Is It?

- Set of related scenarios
  - (i.e. set of related use case instances)...
  - ...in which **actor(s)** use the system...
  - ...to meet some **need**
- **Actor**: something that behaves in some way
  - E.g. a cashier, a computer, a retail store, a customer...
  - What actors exist in a UML editing system?
- **Need**: a goal that the system can support
  - Buying things
  - Returning things
  - Determining what needs to be restocked
- Example: use case as set of use case instances...

## Use Case as Set of Use Case Instances

- Use case title: Handle Returns
  - **Main** use case instance (scenario)
  - **Main** success scenario
    - Customer has items to return, cashier scans in each, gets total, gives refund
  - **Credit card expired** use case instance
    - Customer paid with credit card, but now can't refund to the credit card account, so give cash
  - **Bar code missing** scenario
    - ...

## Example of Use Case – brief format

Customer brings items to checkout counter. Cashier records each item. System displays each item & price, and current total. Customer provides payment somehow. System validates it, logs it, and updates inventory data. System prints receipt and customer waves a cheery goodbye. (see p.46)

- This "brief format" use case actually contains several use case instances...

## Use Case Instances in the Use Case

- Write down a use case instance (scenario) for the UML use case
- then...we'll type in a few...
  - Make use case diagram
    - Create, drag, drop a stick figure
    - User, mouse, computer
  - Load and modify previously created diagram
    - Use file browser to search for diagram in a .uml file, click to bring it up, ...
    - Monitor, file system

- What is the actor for each of those?
  - Make use case diagram
    - Create, drag, drop a stick figure
    - User, mouse, computer
  - Load and modify previously created diagram
    - Use file browser to search for diagram in a .uml file, click to bring it up, ....
    - Monitor, file system

## Features Vs. Valuable Features

- "...the software industry is littered with failed projects that did not deliver what people really needed" – p. 48
- Requirements (hence, use cases) should
  - ...focus on adding value and reaching goals
  - (...not focus on laundry lists of features)
- Otherwise, the system could end up with:
  - features that don't add much value;
  - goals that aren't actually met

## Use Cases – One Requirements Tool

- In the FURPS+ model, use cases
  - ...are primarily oriented toward one letter
  - Which letter? (see next slide)

## FURPS+ Model for Requirements

- F is for Functionality
  - "features, capabilities, security"
- U is for Usability
  - "human factors, help, documentation"
- R is for Reliability
  - MTBF, "recoverability, predictability"
- P is for Performance
  - "response times, throughput, accuracy, availability, resource usage"
- S is for Supportability
  - "adaptability, maintainability, internationalization, configurability"
- + is for...

## Types of Use Cases I

- Black-box use cases
  - Describe what system does
  - Do not describe how system does it
  - Example: "system records the sale"
- White-box use cases
  - Describe mechanism
  - Example: system records sale to database
  - Example: system issues SQL INSERT command
- One of these is recommended – which?
- (Why?)

## Types of Use Cases II

- Brief – 1 paragraph of main use case instance
- Casual – Multiple paragraphs
  - each is a different use case instance (scenario)
- Fully dressed – Full details
  - ...of each step
  - ...of each variation
  - ...with preconditions and postconditions
    - (what are those?)
    - What is a precondition/postcondition for the UML editor?
  - See table, pp. 50-53

## 1-Column Vs. 2-Column

- Used for “fully dressed” use cases
- 2-Column:
  - Left column is for things outside actors do
  - Right column is for things the system does
  - Makes the interaction with the system clear
- 1-Column:
  - More compact
- I'd say try 2-column format first...author now prefers 1-column in his work – p. 54
- Various other format options exist

## Essential Vs. Concrete Style

- See section 6.11

## Example

(See p. 50 ff.)

## The Fully Dressed Use Case

- Primary actor – the main system user
- Stakeholders and their interests –
  - “the use case captures all and only the behaviors related to satisfying the stakeholders’ interests”
  - ...thus telling us what should be in a use case

## More Use Case Sections...

- Precondition –
  - something that must hold
  - ...before a use case begins
- Postcondition –
  - something that must hold
  - ...after a use case ends
- (Postcondition also called success guarantee)

## More Parts of the Use Case

- Main Success Scenario (“Basic Flow”)
  - “Happy path” typically with no branches
  - It’s what you hope will happen
- Extension (“Alternate Flow”)
  - All the (‘not-so-happy’) branches off the main success scenario
  - Can be several of them; steps numerically keyed to Basic Flow; longer than Basic Flow



## More Parts of the Use Case

- Special Requirements
  - Constraints that aren't things the system does
- Technology and Data Variations List
  - Requirements that are almost design-like in flavor
  - We're not designing, but maybe we can't avoid it entirely...
  - ...this is where these aspects go



## ...If Time Allows

- List some use cases for the UML editor system
- For one of them, list use case *instances*