

More System Behavior

System Behavior and Contracts

Operation Contracts

In Class Contracts

Discussion Questions

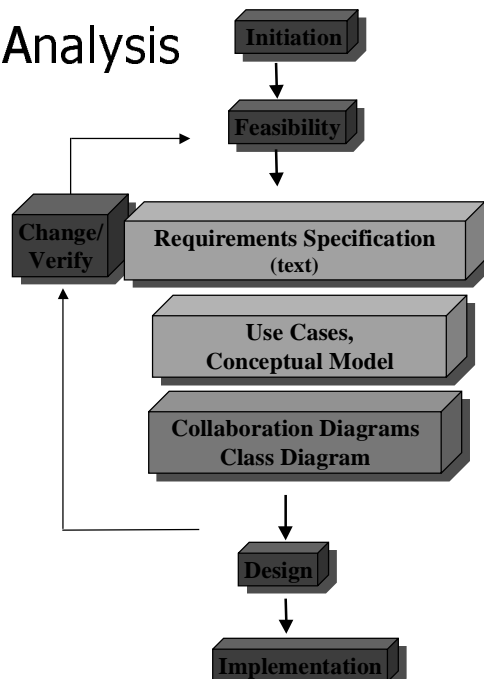
©1998 William N. Robinson

Questions Answered in This Lecture

- How does a contract add to the behavioral descriptions provided by use cases or sequence diagrams?
- What is the purpose of an operation contract?
- How is an operation contract constructed?

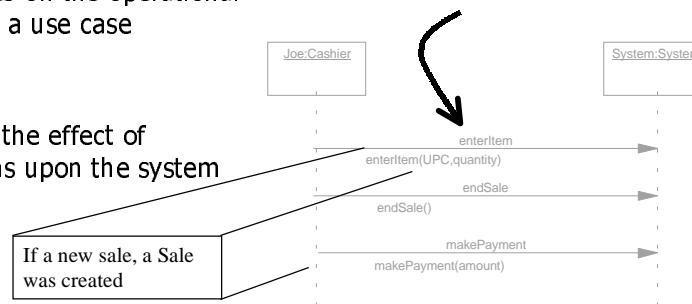
Object-Oriented Analysis

- Textual documentation
 - Describes system
 - Rationale
 - Requirements
- Multiple system models
 - Processing depicted in different views
 - Use case diagram (graph)
 - Sequence Diagram / Contracts (text)
 - Data depicted in different views
 - Conceptual model (graph)
 - Class diagram (graph)



Details with Contracts

- Use Case
 - *A sequence of transactions in a system whose task is to yield a measurable value to an individual actor of the system*
- Sequence Diagram
 - Elaborates on the operational details of a use case
- Contract
 - Describe the effect of operations upon the system



Contract

- Contract
 - a document that describes what an operation commits to achieve
 - a specification of behavior of *what* will happen with little regard for *how* it will be achieved
 - Often specified in terms of condition that must or may hold before the operation (pre-conditions) and condition must hold after the operation (post-conditions)
- System Operation Contract
 - a document that describes changes in the state of the overall system when a system operation is invoked

Example of Contract Sections

- **Name:**
 - enterItem(upc: number, quantity: integer)
- **Responsibilities:**
 - Enter (record) sale of an item and add it to the sale. Display the item description and price.
- **Type:**
 - System
- **Requirements:**
 - R1.1, R1.3, R.19
- **Notes:**
 - Use efficient database access.
- **Exceptions:**
 - If the UPC is not valid, indicate that it was an error.
- **Output:**
- **Pre-conditions:**
 - UPC is known to the system.
- **Post-conditions:**
 - If a new sale, a Sale was created (instance creation)
 - If a new sale, the new Sale was associated with the POST (association formed)
 - A SalesLineItem was created (instance creation)
 - The SalesLineItem.quantity was set to quantity (attribute modification)
 - The SalesLineItem was associated with a ProductSpecification, based on UPC match (association formed)

Building a Contract

- For each system operation construct a contract
 - ❶ Write Responsibilities section
 - informally describing the purpose of the operation
 - *Derive responsibilities from associated Requirements*
 - ❷ Write the Post-conditions section
 - declaratively describing changes in terms changes to object found in the conceptual
 - may need to add classes to the conceptual model
 - Describe post-conditions in past tense, consider these types of changes
 - Instance creation and deletion
 - Attribute modification
 - Associations formed and broken
 - ❸ Write the Pre-conditions section
 - ❹ Write notes and exceptions

Contract Guidelines

- Write declarative statements
 - Do not be lured into writing code
 - It's up to the designer or programmer to spend their time figuring out how the operation should be performed
 - Consider the following paradigm
 - Pre-conditions: state of (key) things before the operation
 - operation occurs: *doesn't matter how*
 - Post-conditions: state of (key) things after the operation
 - Consider the three common categories of post-conditions
 - » Especially, forming associations
- Completeness
 - Create your best guess, but don't be too concerned with fine details as they are best defined during design
- Consistency
 - Consider the context supplied by the Requirements, Use Case, and Sequence Diagrams
 - Do not create operations that do more than is expected in the context

Contract in System Architect

The image shows a UML diagram at the top with two lifelines: 'Cashier:Cashier' and 'POST:System'. A message arrow labeled 'enterItem' points from 'Cashier:Cashier' to 'POST:System'. Below the diagram are two screenshots of the System Architect software interface. The left screenshot shows the 'Modify Definition (UML Class)' dialog for the 'enterItem' class, with fields for Name, Description, Business Name, To Class, To Object, and Operation. The right screenshot shows the 'Modify Definition (Method)' dialog for the 'enterItem' method, with fields for Name, Responsibilities, Notes, Exceptions, Output, Pre-conditions, and Post-conditions. The 'Post-conditions' field contains several lines of text describing the state of the system after the method execution.

Distribution of Submarine Supplies Requirements



We are losing business due to our poor order processing and delivery. We mainly serve as a distribution center that consolidates produce and meats for submarine shops. However, we are not meeting our business objectives of profit, quality, responsiveness. Moreover, we are not as efficient as we would like, as our works spend an inordinate amount of time trying to use our current system. Things are a bit chaotic now...

- Your assignment
 - Draw a System Sequence Diagram
 - Focus on a specific subsystem

**Use
System Architect**

Discussion Questions

- I think I can define requirements and create an initial UML product specification, but how do I find the requirements?
 - That is, how do I do Requirements Elicitation?
- This seems to be a bunch of low level (detailed) analysis and description, can't we do analysis at a higher level, say based on prior experience?
 - ... using prior experience of analysis in general?
 - ... using prior experience of information systems analysis?