Object Oriented Analysis

Identifying Use cases
Object Analysis: Classification
Identifying object relationships, Attributes and Methods.

Identifying the use cases: Goals

The use-case approach to object-oriented analysis and the object-oriented analysis process.

- Identifying actors.
- Identifying use cases.
- Documentation.

What Is Analysis?

Analysis is the process of transforming a problem definition from a **fuzzy set** of facts and myths into a coherent statement of a system's requirements.

Analysis (contd..)

- The main objective of the analysis is to capture:
  - a **complete**, **unambiguous**, and **consistent picture** of the requirements of the system and
  - what the system **must do** to satisfy the users' requirements and needs.

Where Should We Start?

1. Examination of existing system documentation.
2. Interviews.
3. Questionnaire.
4. Observation.

Requirements Difficulties

Three most common sources of requirements difficulties are:

1. Incomplete requirements.
2. Fuzzy descriptions (such as fast response).
3. Unneeded features.

Use-Case Driven Object-Oriented Analysis: The Unified Approach

- The OOA phase of the UA uses the following to describe the system.
  - Actors-external factors interact with the system
  - Usecases-scenarios that describe how actors use the system

The Object-Oriented Analysis (OOA) Process

The process consists of the following steps:

1. Identify the actors:

– Who is using the system?

– Or, in the case of a new system, who will be using system?

The OOA Process (Con't)

2. Develop a simple business process model using UML activity diagram.

- **Activity diagrams** are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.
- In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control

The OOA Process (Con't)

3. Develop the use case:

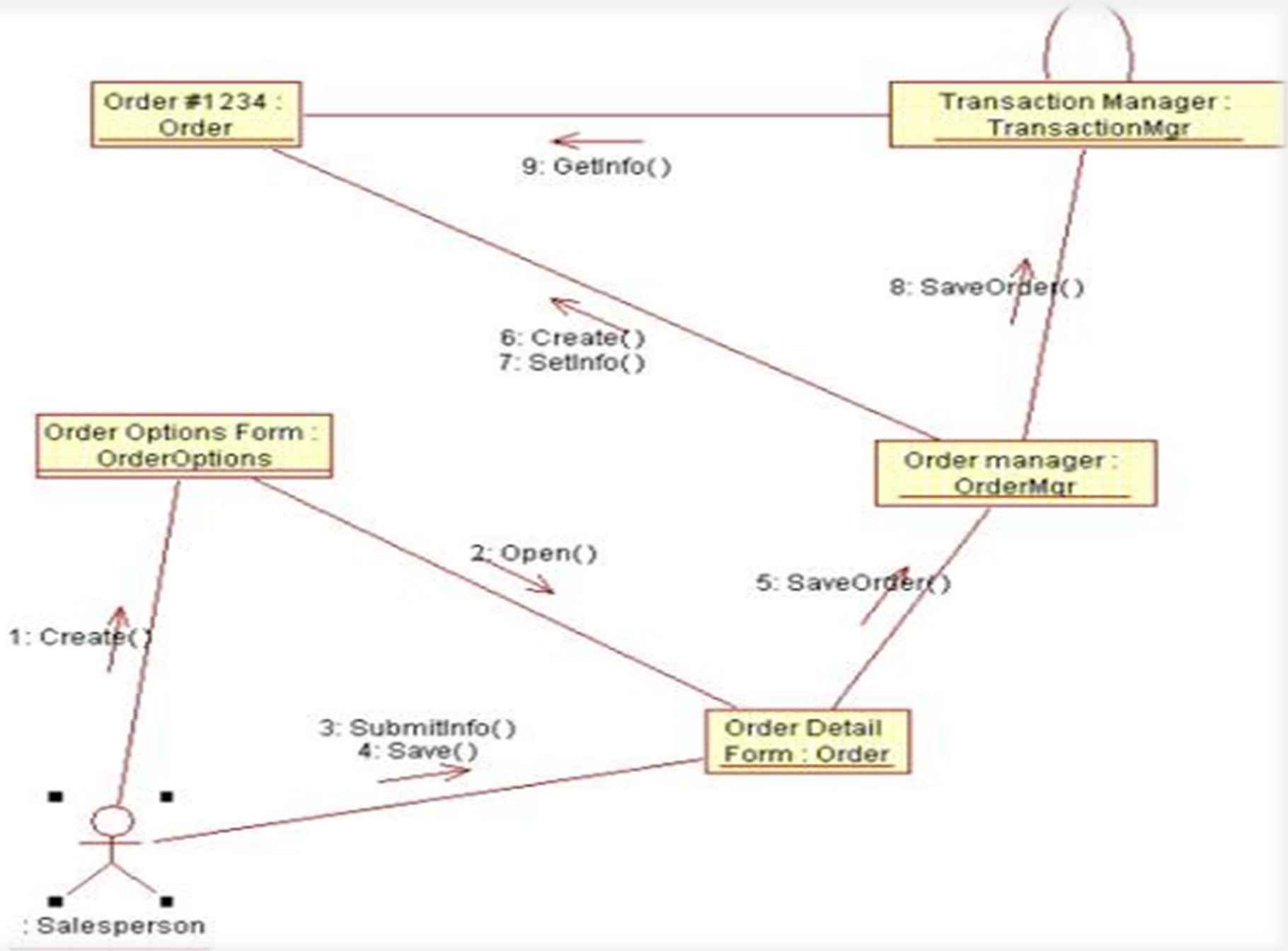
- What the users are doing with the system?
- Or, in the case of a new system, what users will be doing with the system?

Use cases provide us with comprehensive documentation of the system under study.

The OOA Process (Con’t)

4. Prepare interaction diagrams:

- ❑Determine the sequence.
- ❑Develop collaboration diagrams.


Order #1234 : Order
Transaction Manager : TransactionMgr
9: GetInfo()
8: SaveOrder()
6: Create()
7: SetInfo()
Order Options Form : OrderOptions
Order manager : OrderMgr
2: Open()
5: SaveOrder()
1: Create()
3: SubmitInfo()
4: Save()
Order Detail Form : Order
: Salesperson

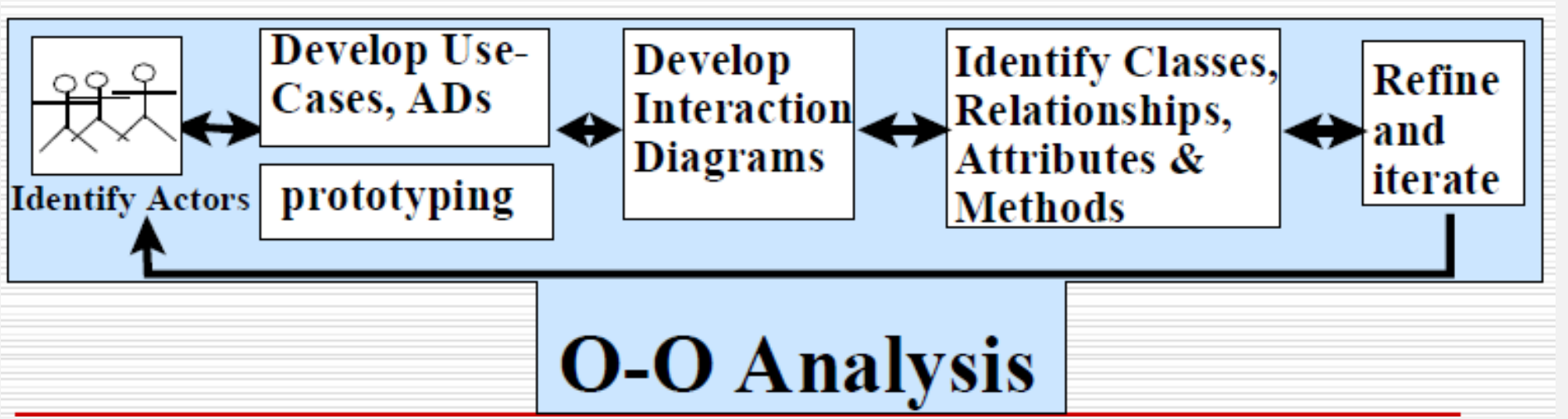
The OOA Process (Con't)

5. Classification—develop a static UML class diagram:

- Identify classes.
- Identify relationships.
- Identify attributes.
- Identify methods.

The OOA Process (Con't)

6. Iterate and refine: If needed, repeat the preceding steps.



Use Case Model

- Use cases are scenarios for understanding system requirements.
- Use case defines what happens in the system when a use case is performed.
- The use-case model tries to systematically identify uses of the system and therefore the system's responsibilities.

Use Cases Under the Microscope

- "A **Use Case** is a sequence of transactions in a system whose task is to yield results of measurable value to an individual actor of the system."

Use Case Key Concepts

- Use case Use case is a special flow of events through the system.
- Actors An actor is a user playing a role with respect to the system.
- In a system This simply means that the actors communicate with the system's use case

Use Case Key Concepts (Con't)

- Measurable value A use case must help the actor to perform a task that has some identifiable value.

- Transaction A transaction is an atomic set of activities that are performed either fully or not at all.

Use Associations

- The use case association occurs when you are describing your use cases and notice that some of them have common subflows.

- The use association allows you to extract the common subflow and make it a use case of its own.

Extends Associations

- The extends association is used when you have one use case that is similar to another use case but does a bit more or Is more specialized; in essence, it is like a subclass

Types of Use Cases

- *Essential Use Cases* are expressed in an ideal form that remains relatively free of technology and implementation details; design decisions are deferred and abstracted, especially those related to the user interface.
- *Concrete or Real Use Case* concretely describes the process in terms of its real current design, committed to specific input and output technologies and so on.

Types of Use Cases

- *Abstract Use Case* is not complete and has no actor that initiates it but is used by another use cases.

IDENTIFYING THE ACTORS

- The term ***actor*** represents the role a user plays with respect to the system.
- When dealing with actors, it is important to think about roles rather than people or job titles.
- The actors is as important as identifying classes, structures, associations, attributes and behavior.

IDENTIFYING THE ACTORS

- Actors can be found through the answers to the following questions.

➢ Who is using the system ?. or,

➢ Who affected by the system? Or,

➢ Which groups need help from the system to perform a task?.

➢ Who affects the system? Or ,

IDENTIFYING THE ACTORS

- Which user groups are needed by the system to perform its functions?. These functions can be both main functions and secondary function such as administration.
- Which external hardware or other systems use the system to perform task?.
- What problems does this application solve?.
- How do users use the system(use case)?What are they doing with the system

GUIDELINES FOR FINDING THE USECASES

1. For each actor, find the tasks and functions that the actor should be able to perform. The use case should represent a course of events that leads to a clear goal.
2. Name the use cases
3. Describe the use case briefly by applying terms with which the user is familiar .this makes the description less ambiguous.

Dividing Use cases Into Packages

- Use case represents a particular scenario in the system.
- A design is broken down into packages.

EX::In a library system, the various scenarios involve a supplier providing books or a member doing research or borrowing books, Do research and purchase books

DEVELOPING EFFECTIVE DOCUMENTATION

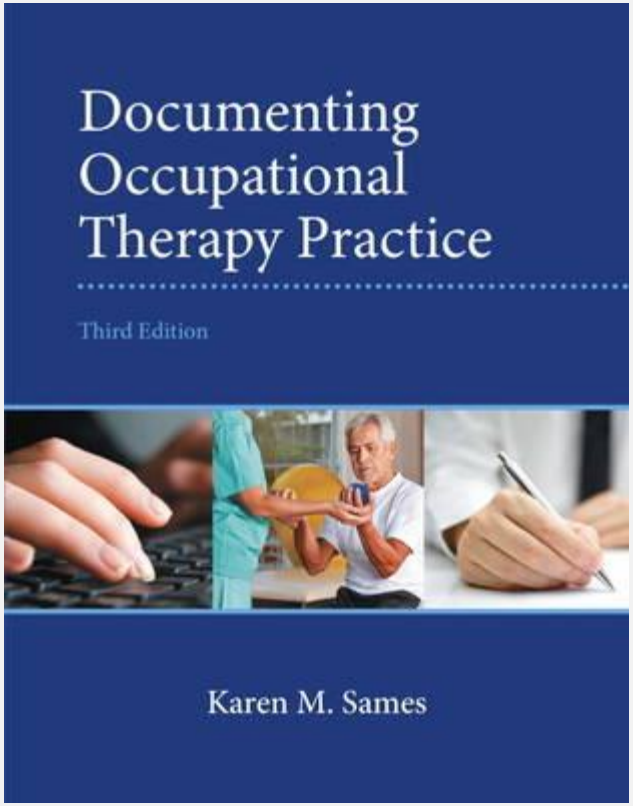
- A document can serve as a communication vehicle among the project's team members, or it can serve as an initial understanding of the requirements.
- In many projects, documentation can be an important factor in making a decision about committing resources.

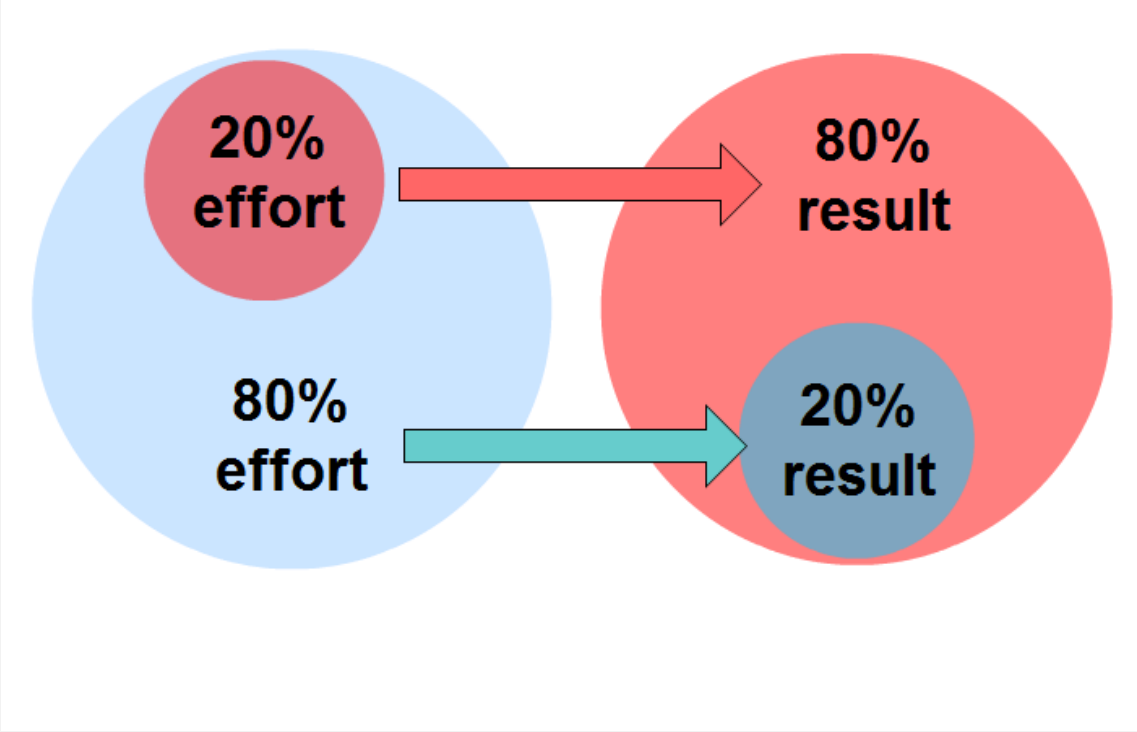
Organization Conventions For Documentation

- The documentation depends on the organization's rules and regulations.
- Too little documentation invites disaster
- Too much documentation, transfers energy from the problem solving tasks to a mechanical and unrewarding activity.

Guidelines for developing Effective documentation

- Common cover
- 80-20 rule
- Familiar vocabulary
- Make the documentation as short as possible
- Organize the document


Documenting
Occupational
Therapy Practice
Third Edition
Karen M. Sames


20%
effort
80%
result
80%
effort
20%
result


question
add
concept
mental
arithmetic
styles
schooling
sensation
auditory
stories
studious
one
math
communicate
learning
questions
teaching
mathematics
reading
vocabulary
learn
writing
students
speaking
make
friends
fun
generally
subtract
cool
short
three
children
good
sports
consciousness

References

- http://www.tutorialspoint.com/uml/uml_use_case_diagram.htm
- http://gyan.fragnel.ac.in/~surve/OOAD/UCD/UCD_Types.html