Pragmatic Product Leader Course

Become a Product Manager with Superpowers

UNIT 6

B.O.O.M Step 1

Interviews

- Interviews are a must across the phases
- In iterative processes, interviews are conducted in each phase

Format	What	When	Benefits	Disadvantages
One-on-one interviews		During the Initiation and Discovery phases	Is easy to organize	Reconciling discrepancies is time consuming
Brainstorming	Group interview for enlisting new ideas	During Initiation phase and whenever the project is "stuck"	Breaks old ways of thinking	Does not yield detailed requirements
Joint application development (JAD)	Group interview to gather requirements	During Discovery phase	Simplifies reconciling of discrepancies, decreasing analysis time; can be used to create various deliverables, including the following: ■BRD ■Proof of concept ■Strategy ■Screens ■Decision tables	Difficult to get all the interviewees in one room at same time Group-think
Structured walkthrough	Group interview to verify requirements	During Discovery phase, after early draft of requirements is available. During Construction phase, to review technical specifications and, on iterative projects, to verify remaining requirements.	Moves testing forward, reducing the impact of mistakes	Difficult to get interviewees in one room at same time Group-think

B.O.O.M. Step 1 - Initiation

Key components of the PRD produced during this phase are as follows:

- Business use-case descriptions, including business use-case diagrams
- Role map
- System use-case diagram
- Initial class diagram, describing key business classes

B.O.O.M. Step 1 - Initiation

You will cover the following, in the step 1:

1a Model Business use cases
- i Identify business use cases (business use-case diagram)
- ii Scope business use cases (activity diagram)

1b Model System use cases
- i Identify actors (role map)
- ii Identify system use-case packages (system use-case package diagram)
- iii Identify system use cases (system use-case diagram)

1c Begin structural model (class diagrams for key business classes)

1d Set baseline for Discovery (PRD/initiation)

1a Model Business use cases

i Identify business use cases (business use-case diagram)
ii Scope business use cases (activity diagram)

B.O.O.M. Step 1 - Initiation

1a Model business use cases

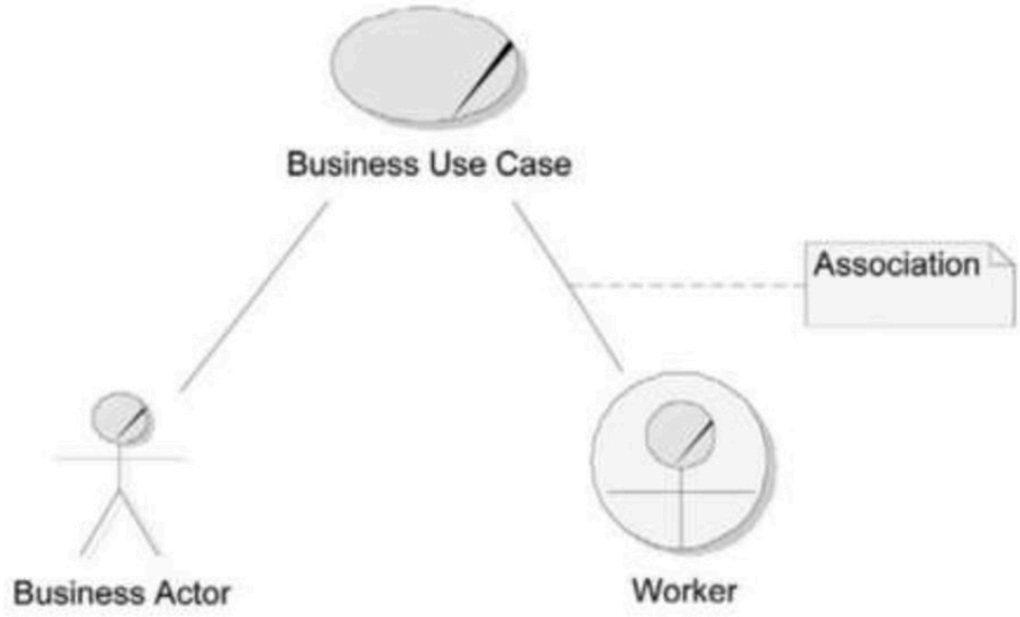
- Specific workflow in the business - an interaction that a stakeholder has with the business that achieves a business goal
- Document business use cases using business use-case diagrams
    - Describe the players in each business use case
    - Use text or workflow diagram (such as an activity diagram) to describe the interaction between the players and the business as the use case is played out

B.O.O.M. Step 1 - Initiation

1a Model business use cases > (i) Identify Business use cases (business use case diagram)

A business use-case diagram provides an overview of business processes and services (business use cases) and the entities that use those services or participate in their implementation.

- Business actor
- Worker
- Association



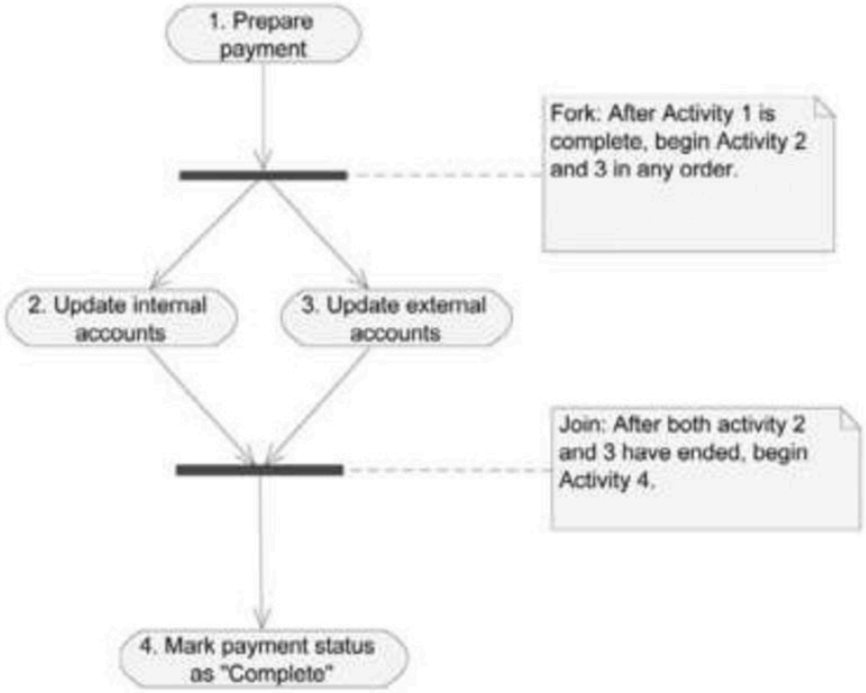
B.O.O.M. Step 1 - Initiation

1a Model business use cases > (ii) Scope business use cases (Activity diagrams)

Activity diagram is one of the most useful diagrams to depict and understand workflow.

Diagrams for depicting workflow:

- System flowchart
- Swimlane workflow diagram
- Sequence diagram
- Activity diagram



Activity diagram

B.O.O.M. Step 1 - Initiation

1a Model business use cases > (ii) Scope business use cases (Activity diagrams)

States

Objects may be considered to be in various states during their lifetimes.

To find out what these states are, simply ask the stakeholders to tell you what statuses they consider a business object to be in.

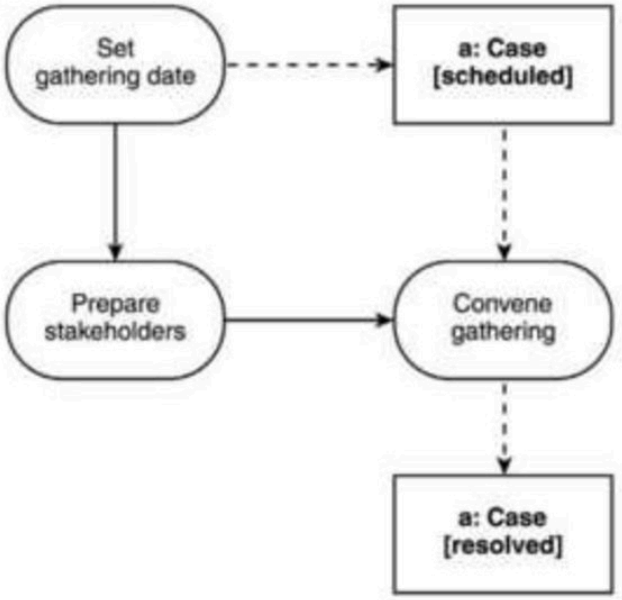


Fig. a: Activity diagram with states

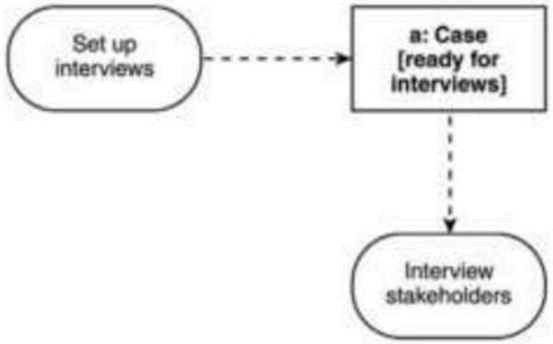


Fig. b: Activity diagram with states

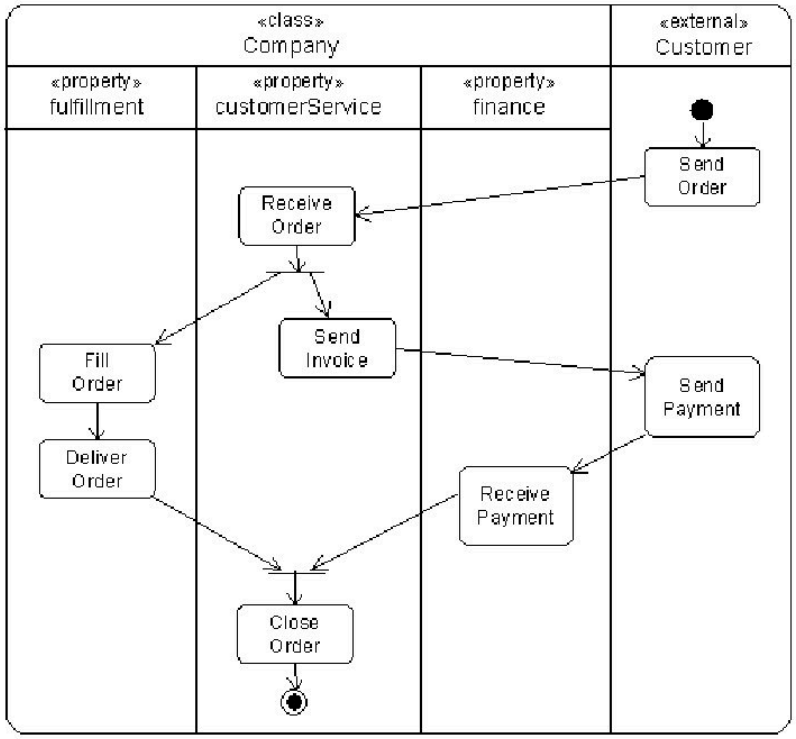
B.O.O.M. Step 1 - Initiation

1a Model business use cases > (ii) Scope business use cases (Activity diagrams)

Activity diagrams with partitions (Swimlanes)

To indicate who performs each activity, you add partitions (commonly referred to as swimlanes) to the activity diagram

A partition is depicted as a column (or row) on an activity diagram. Each partition represents a stakeholder (business actor or worker) who carries out some activity.



1b Model system use cases

B.O.O.M. Step 1 - Initiation

1b Model system use cases

System use cases (often just called use cases) help you imagine the IT system from a user perspective, by focusing on the user's goals.

Standardise common issues

- How users are documented (role map)
- Break up the user requirements into manageable pieces (system use-case diagrams)

B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(i) Identify actors (Role Map)**

Identify the IT system's users, or *actors*.
Earlier, we spoke of actors in relation to *business* use-case modeling. In this context, **actor is a role played by a person or system that interacts with the IT system.**
Similar terms: External agent/external entity, Stakeholder

Finding actors: To find actors go through your list of business actors and workers, eliminating any who don't interact with IT system. Then add any external systems and human users who are required because of the technology.

The role map: Diagram used to standardise the treatment of users and external systems. It is a restricted form of use-case diagram.

B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(i) Identify actors (Role Map)**

FAQs

Why identify actors and why do it now?
To focus on user's needs

If a user only receives reports from the system, is that user an actor?
Yes (a bit controversial though)

How to handle system use cases that aren't started by anybody, but just start up automatically at a given time? Where's the actor?
Define an actor called Time to act as the initiator of these use cases (There is also controversy about this issue. Some practitioners, for example, prefer to see no actor and some prefer to indicate the actor who has asked that the use case be initiated at that time.)

If a customer calls in a request and a customer service representative keys it in, which one is the actor?
Only the actor who directly interacts with the computer system is considered an actor. In this case, it would be the CSR.

B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(i) Identify actors (Role Map)**

Modeling actors with overlapping roles

Using generalisation relationship between actors.
Any time the phrase “a kind of ” comes up in the discussion of actors, think about using the generalization relationship.

For example, a Bookkeeper and an Accountant are two kinds of Accounting Staff. Exactly how you draw the generalization depends on how the roles overlap.

We’ll look at two types of situations:

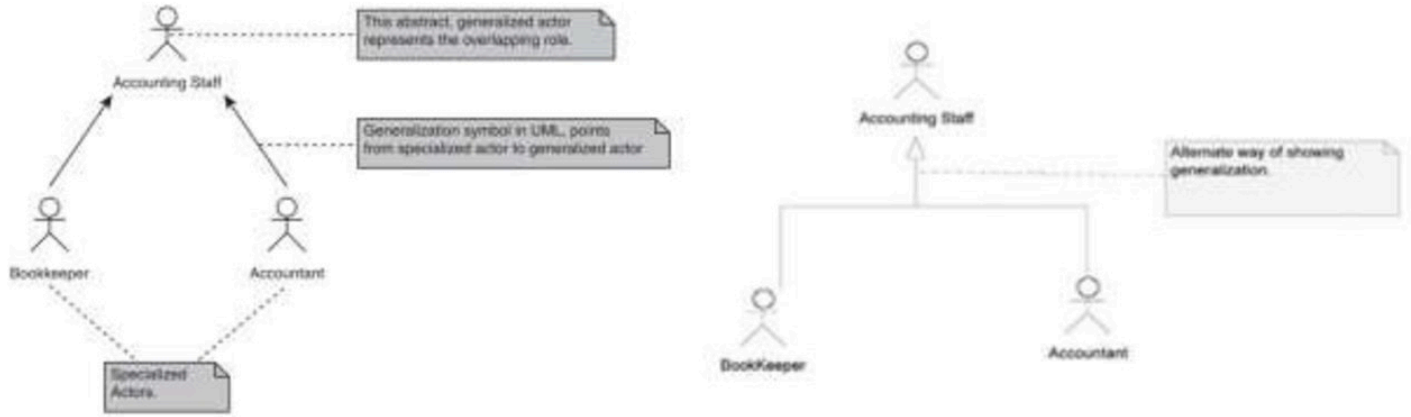
- Actors whose roles partially overlap
- An actor whose role completely encompasses another’s

B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(i) Identify actors (Role Map) > Overlapping roles**

Actors whose roles partially overlap

When two actors have some overlap in their roles, but each actor can do things with the system that the other can't, model the actors as specialised actors and invent an *abstract generalized actor* to represent the overlap. The term generalised implies that the specialised actors inherit something from the generalised actor. The term *abstract* means that the invented actor is not real.

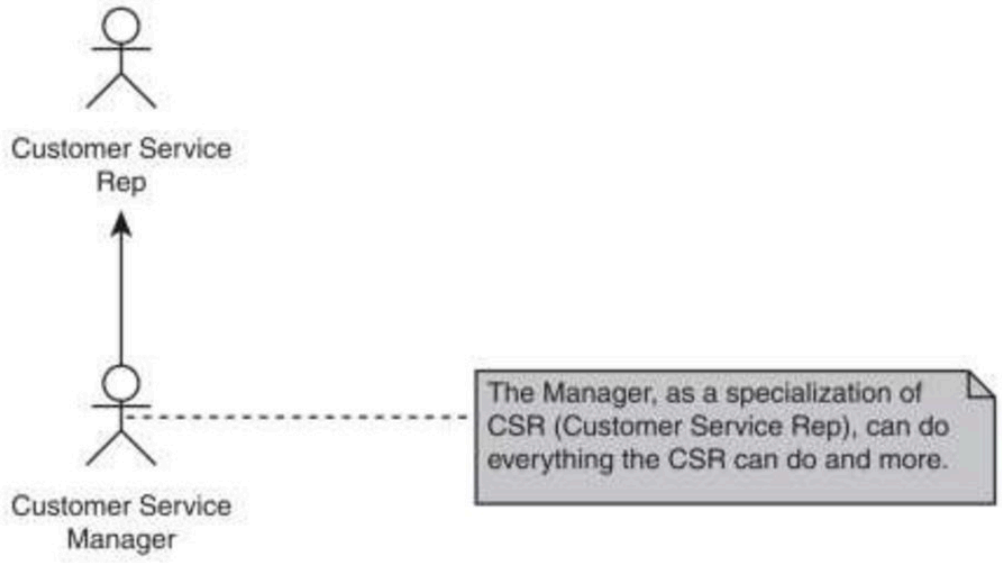


B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(i) Identify actors (Role Map) > Overlapping roles**

Actor whose role totally encompasses another's

When two actors have some overlap in their roles, but each actor can do things with the system that the other can't, model the actors as specialised actors and invent an *abstract generalized actor* to represent the overlap. The term generalised implies that the specialised actors inherit something from the generalised actor. The term *abstract* means that the invented actor is not real.



B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(ii) Identify system use-case packages (system use-case diagram)**

Create system use-case packages, to support a number of business use cases. A system use-case package is a collection of system use cases and the diagrams that describe them.

What criteria are used to group system use cases into packages?

- Group system use cases by the main actor who uses them
- Create a system use-case package for each business use case

Naming use-case packages

Use a noun phrase or according to the business use case it supports (this makes tracking easier)

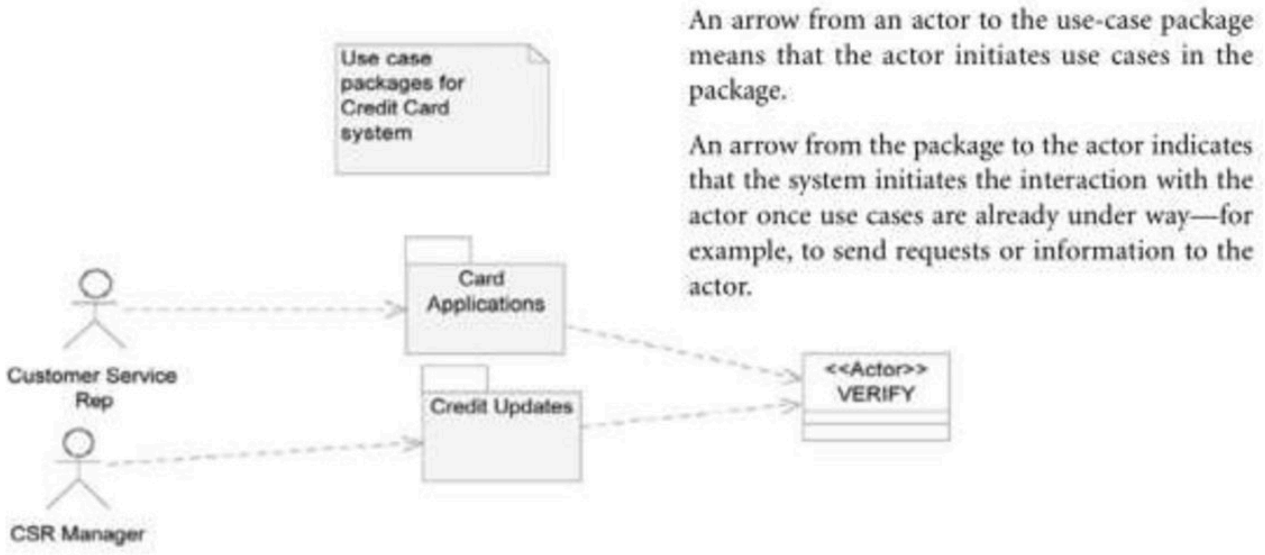
B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(ii) Identify system use-case packages (system use-case diagram)**

Diagramming system use-case packages

Employ use-case diagrams though it looks a little odd in that it does not depict any actual use cases.

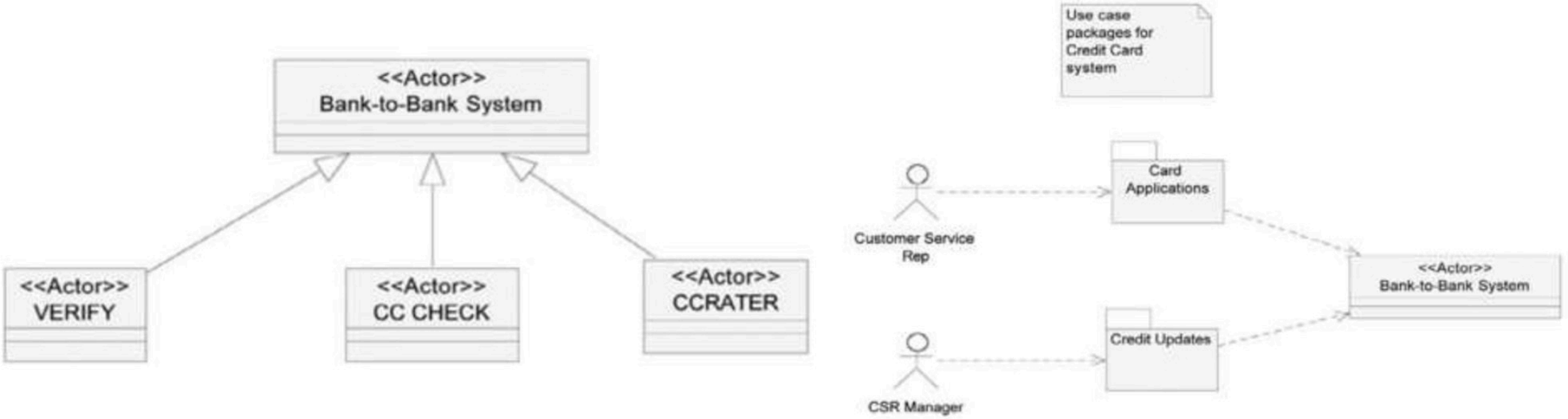
This diagram indicates that a customer-service representative can initiate use cases relating to card applications and that a CSR manager initiates updates to credit. In both cases, the system under design will need to be able to communicate with VERIFY (an external system that verifies the application against a person's credit record).



B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(ii) Identify system use-case packages (system use-case diagram)**

What if a use-case package is connected to all of the specialised actors of a generalised actor?



B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(iii) Identify system use-cases (system use-case diagram)**

A system use case is an interaction between an actor and a computer system.

- Identify the system use cases that go into the packages.
- Review the activities described by the *business use cases* and identify the ones that fall within the scope of IT system.
- Once identified, group them into system use cases.
- Imagine the system, how will someone sitting at a terminal actually use this system? What result is the user trying to achieve from the computer system with each interaction?
- **Each of these results, expressed as a user goal, is a system use case**
- For example, for a web banking system, some system use cases are “View Transaction history”, “Transfer Funds”, and “Pay Bill”

B.O.O.M. Step 1 - Initiation

1b Model system use cases > **(iii) Identify system use-cases (system use-case diagram)**

What Is the Purpose of Segmenting the User Requirements into System Use Cases?

System use cases become the central tool that governs the management of the project. With their user perspective, they keep the team focused on the user throughout the project.

Here's how:

- The requirements are written from the user's point of view
- System use cases help ensure that the user receives useful functionality with each release when a project is managed iteratively
- System use cases lead to user interfaces that are organised from a user perspective
- System use cases yield a set of test cases that encompass the ways users use the system

B.O.O.M. Step 1 - Initiation

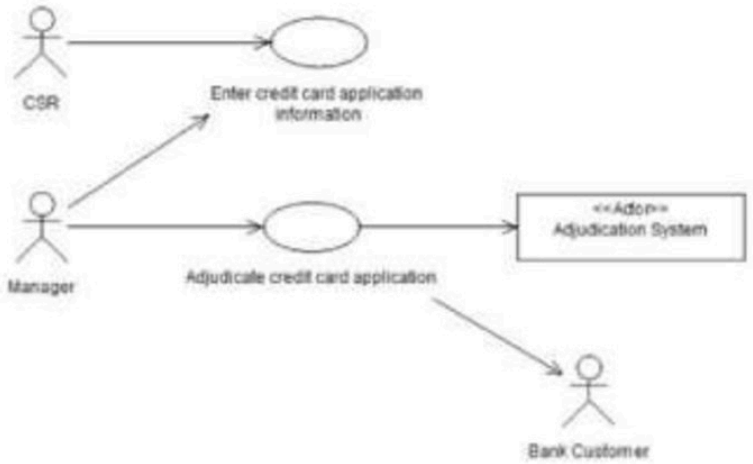
1b Model system use cases > **(iii) Identify system use-cases (system use-case diagram)**

Modeling System Use Cases

- After identifying what system use cases are required to support a business use case, you document your findings in a system use case diagram.
- Create one (or more if necessary) system use-case diagram for each system use-case package.
- The system use-case diagram shows which actors participate in each system use case.
- The diagram does not show sequencing (To show sequencing, use an activity diagram instead)

Figure shows:

- Primary actor
- Secondary actor
- System use case

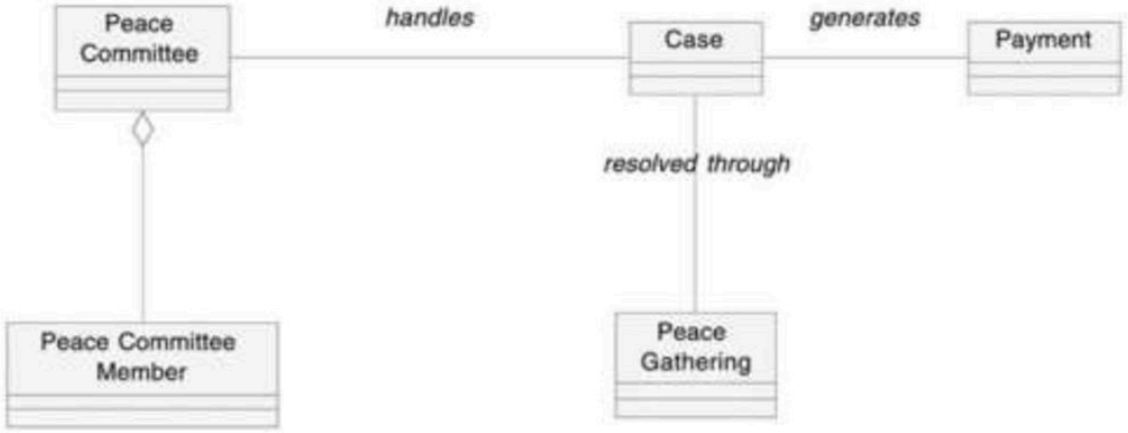


1c. Begin Structural Model

(class diagrams for key business classes)

B.O.O.M. Step 1 - Initiation

1c Begin Structural Model



Structural model is used to describe the Structure of a system using classes.

Here's what the figure means:

- A Peace Committee handles a case.
- A case is resolved through Peace Gatherings.
- A case generates payment(s).
- A Peace Committee consists of Peace Committee members.

Other information, such as the number of payments per case, can also be added to the structural model at this stage.Since this is still early in the project, expect changes to be made to the structural model as the project progresses.

1d Set baseline for Discovery

(PRD/initiation)

B.O.O.M. Step 1 - Initiation

1d Set baseline for discovery

Once the Initiation phase of the project is over, you need to “baseline” your analysis.

By baselining your documentation, you ensure that if changes are requested later, you’ll be able to check whether they represent a change from the original scope of the project.

Keep in mind, however, the exception for agile projects, where the requirements are not baselined unless they are under development. The analysis up to this point also becomes the starting point for the next phase of the project, the Discovery phase.

To conclude

Thank you.