

SWEN30006

Software Modelling and Design

USE CASES

Larman Chapter 6

*The indispensable first step to getting the things
you want out of life: decide what you want.*

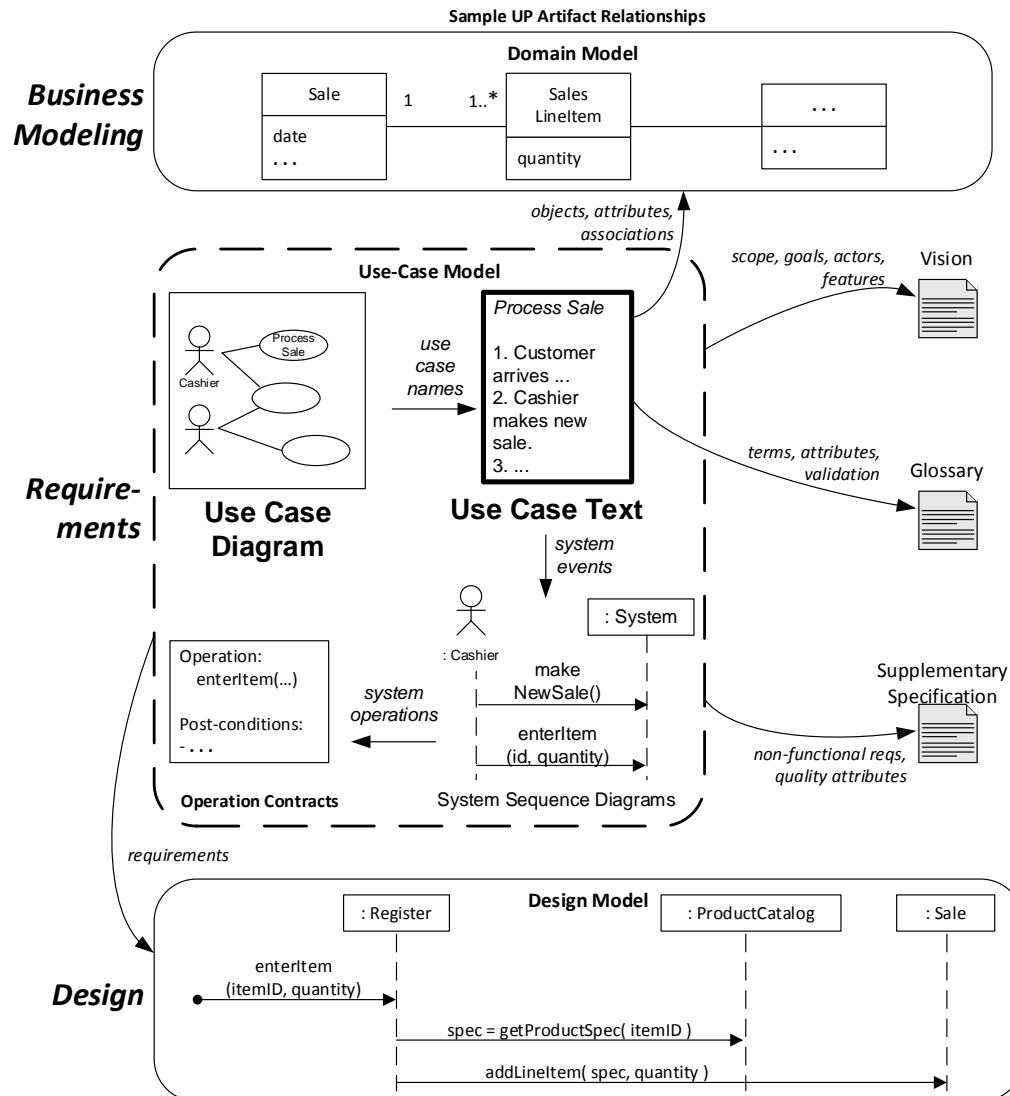
—Ben Stein

Objectives

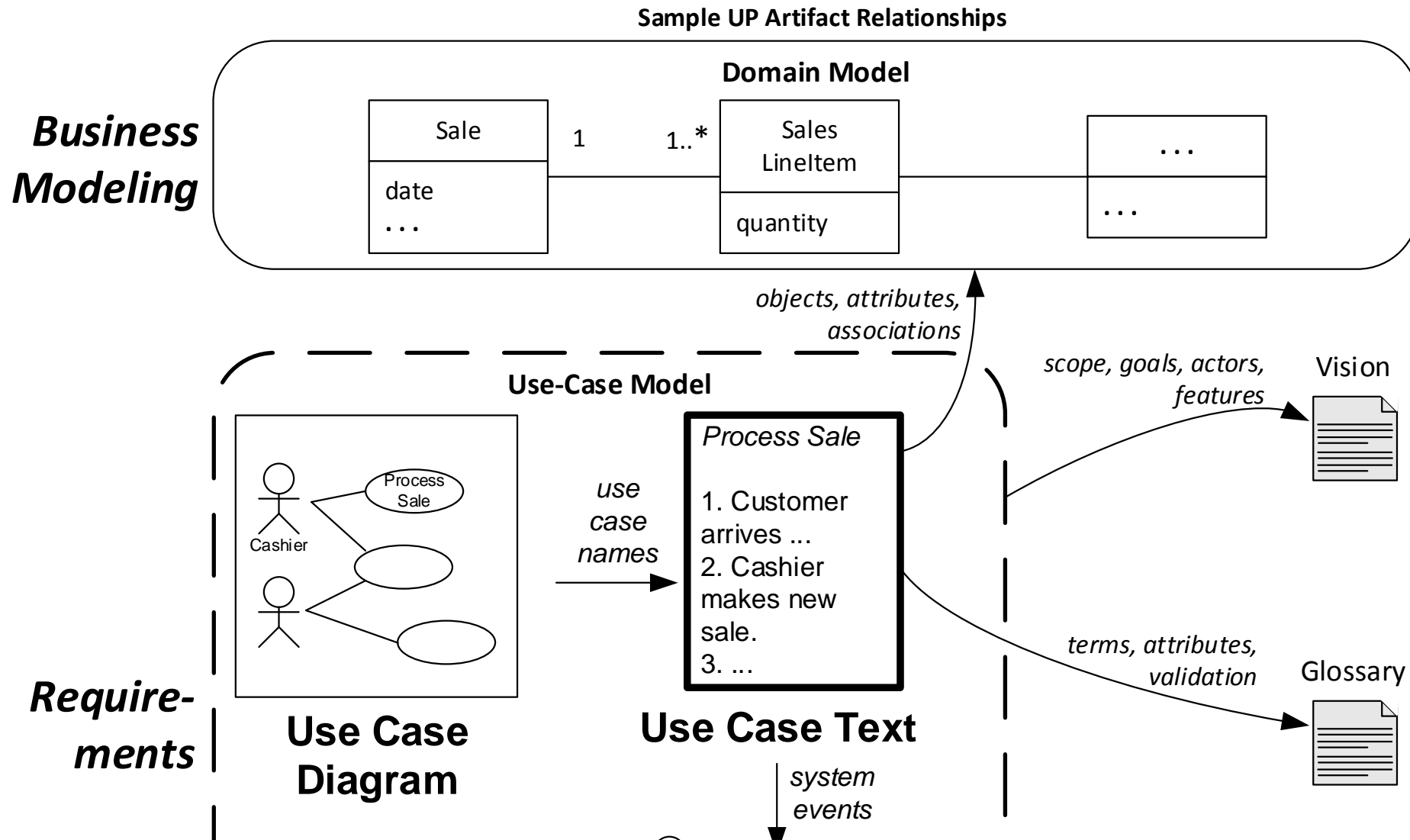
On completion of this topic you should be able to:

- ❑ Identify and write use cases.
- ❑ Use the brief, casual, and fully dressed formats, in an essential style.
- ❑ Apply tests to identify suitable use cases.
- ❑ Relate use case analysis to iterative development.
- ❑ Recognise and understand use cases, in their different forms, and use case diagrams

Use Case Influence in UP



Use Case Influence in UP



Example: Process Sale (brief)

A customer arrives at a checkout with items to purchase. The cashier uses the POS system to record each purchased item. The system presents a running total and line-item details. The customer enters payment information, which the system validates and records. The system updates inventory. The customer receives a receipt from the system and then leaves with the items.

Example: Pre-Operative Planning (brief)

The surgeon inspects and annotates an OWS x-ray; judgement is made regarding the femur neck resection lines and location of replacement components. Decisions are also made by the surgeon regarding type and size of replacement components, and the accessories needed for the operation. An operation plan is prepared to reflect decisions made during Pre-Operative Planning. Orders for the components and accessories are generated. A Fad may also be prepared and some details of the Pre-Operative Plan are added to Patient Details.

Definitions

- ❑ **SuD**: system-under-discussion
- ❑ **actor**: something with behaviour, such as a person, computer system or organisation (e.g. a cashier).
- ❑ **scenario** (or **use case instance**): a specific sequence of actions and interactions between actors and the SuD.
- ❑ **use case**: a collection of related success and failure scenarios that describe an actor using the SuD to support a goal.

Note:

- use cases are text, not diagrams.
- use cases are not object-oriented.

Example: Handle Returns (casual)

Main Success Scenario: A customer arrives at a checkout with items to return. The cashier uses the POS system to record each returned item ...

Alternate Scenarios:

If the customer paid by credit, and the reimbursement transaction to their credit account is rejected, inform the customer and pay them with cash.

If the system detects failure to communicate with the external accounting system, ...

Handle Returns (fully dressed 1)

Main Success Scenario:

1. Customer at checkout with items to return.
2. Cashier starts new return.
3. Cashier enters item id.
4. ...

Cashier repeats steps 3-4 until indicates done.

5. ...
6. Cashier reimburses customer.

Handle Returns (fully dressed 2)

Alternate Scenarios:

6a. Customer paid by credit:

1. Cashier requests credit reimbursement.

- 2a. Cashier reports success/provides receipt.

- 2b. Cashier reports failure/provides cash.

6b. Customer paid by cash:

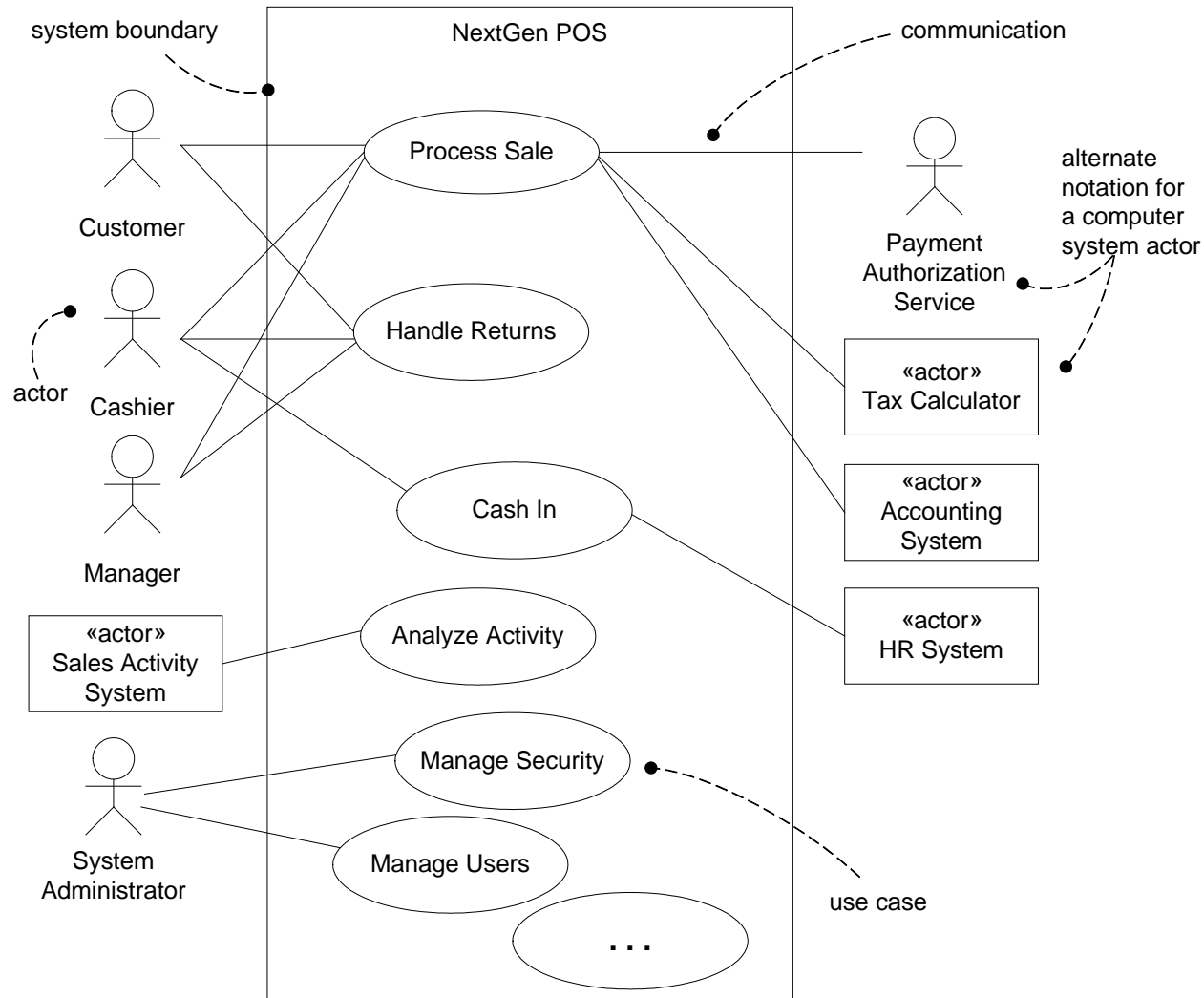
1. Cashier reimburses customer's cash.

*a. At any time, comms failure with external accounting system: ...

Fully Dressed Use Cases

Use Case Section	Comment	Use Case Section	Comment
Use case Name	Start with a verb.	Main Success Scenario	A typical, unconditional happy path scenario of success.
Scope	The system under design.	Extensions	Alternate scenarios of success or failure.
Level	“user-goal” or “subfunction”	Special Requirements	Related non-functional requirements.
Primary Actor	Calls on the system to deliver its services.	Technology and Data Variations List	Varying I/O methods and data formats.
Stakeholder and Interests	Who cares about this use case, and what do they want?	Frequency of Occurrence	Influences investigation, testing, and timing of implementation.
Preconditions	What must be true on start, <i>and</i> worth telling the reader.	Miscellaneous	Such as open issues.
Success Guarantee	What must be true on successful completion, <i>and</i> worth telling the reader.		

Use Case Diagram: NextGen POS



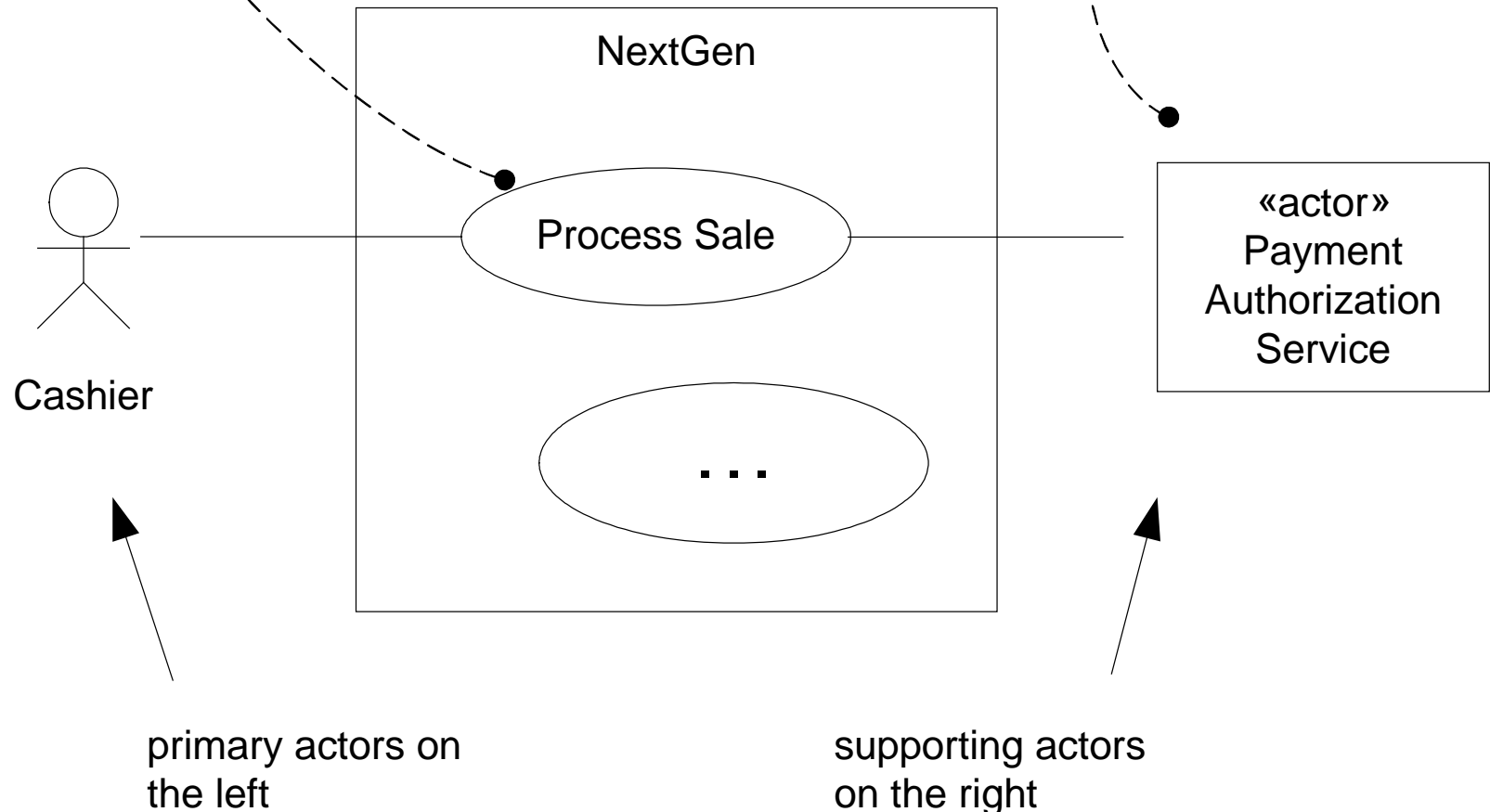
Three Kinds of Actors

- ❑ **primary**: has user goals fulfilled through using services of the SuD (e.g. the cashier).
 - user goals drive the use case
- ❑ **supporting**: provides a service (e.g. information) to the SuD (e.g. an automated payment authorization service). Often a computer system, but could be an organization or person.
 - clarifies external interfaces and protocols
- ❑ **offstage**: has an interest in the behaviour of the use case, but is not primary or supporting (e.g. a government tax agency).
 - ensure all interests identified/satisfied

Notation Suggestions

For a use case context diagram, limit the use cases to user-goal level use cases.

Show computer system actors with an alternate notation to human actors.



Finding Useful Use Cases

❑ Boss Test

- Your boss asks, “What have you been doing all day?”
You reply: “<Performing use case>”. Is your boss happy?

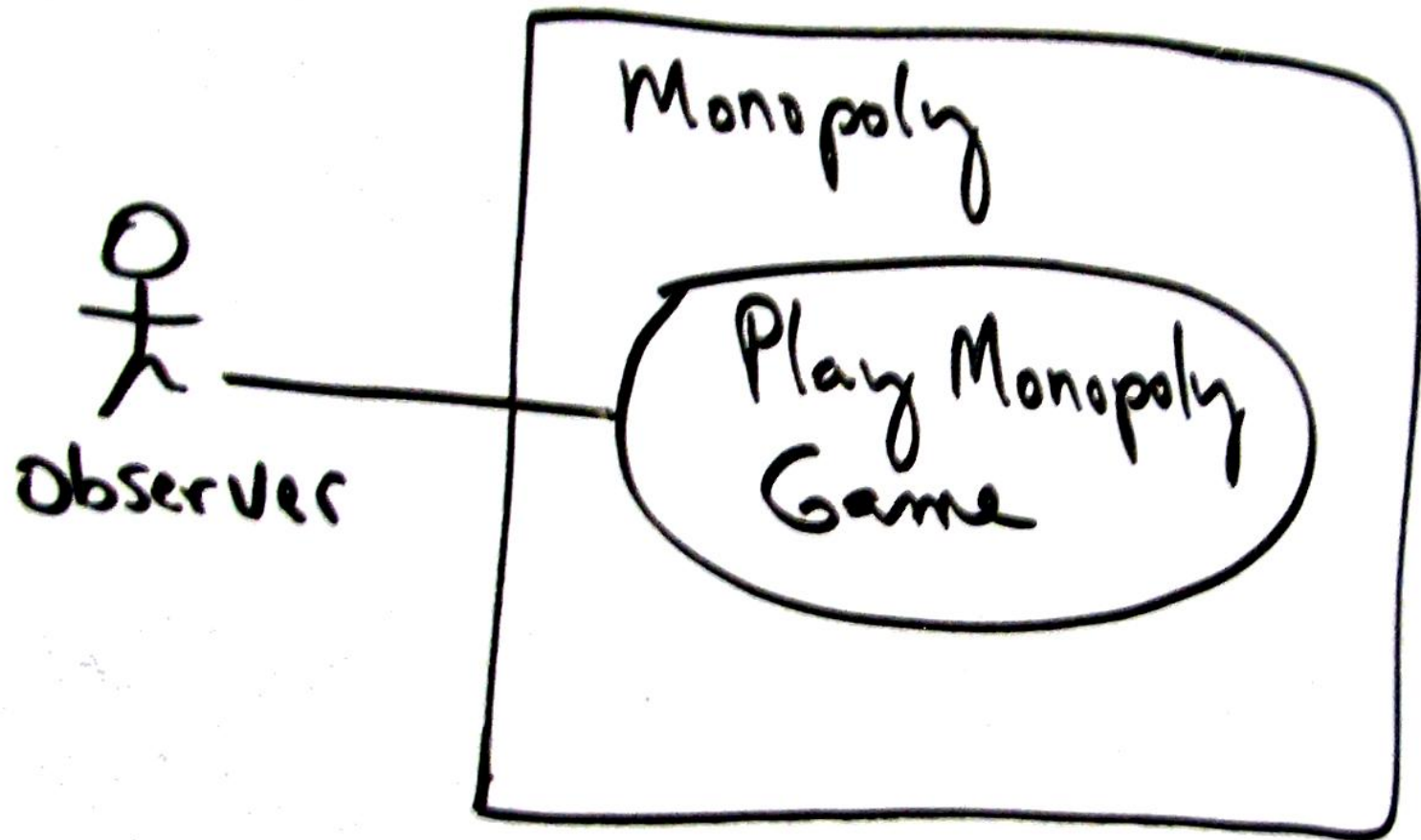
❑ Elementary Business Process (EBP) Test

- A task performed by one person in one place at one time, in response to a business event, which adds measurable business value and leaves the data in a consistent state.

❑ Size Test

- Very seldom a single action/step; typically many steps; fully dressed often require 3–10 pages of text.

Use Case Diagram: Monopoly



Summary

- ❑ *Use Cases* are text descriptions of an *actor* using the system to achieve a goal.
- ❑ Use Cases cover both success and failure scenarios.
- ❑ Use cases can be *brief*, *casual*, or *fully dressed*.
- ❑ *Use Case Diagrams* summarise use cases and show some relationships between them.