

SWEN30006

Software Modelling and Design

RELATING USE CASES

Larman Chapter 30

*Why do programmers get Halloween and Christmas mixed up?
Because OCT(31) = DEC(25)*

Objectives

On completion of this topic you should be able to:

- ❑ Relate use cases with *include* and *extend* associations, in both text and diagram formats.

Use Case Include Relationship

Use *include* when you are repeating yourself in two or more separate use cases and you want to avoid repetition (or when a use case is too long).

- ❑ Most common and important relationship
- ❑ Arises when partial behaviour is common across several use cases
- ❑ Refactor common part into a *subfunction* use case
- ❑ (Hyper)link to subfunction use case to avoid duplication
- ❑ Included case may apply in its own right, e.g. Search

Use Case Include Example (1a)

UC1: Process Sale

(the including use case)

...

Extension Points: *VIP Customer*, step 1. *Payment*, step 7

Main Success Scenario:

1. Customer arrives at a POS checkout with goods and/or services to purchase.

...

7. Customer pays and System handles payment.

...

Extensions:

...

7b. Paying by credit: Include Handle Credit Payment.

7c. Paying by check: Include Handle Check Payment.

...

Use Case Include Example (1b)

UC7: Process Rental (the including use case)

...

Extensions:

6b. Paying by credit: Include Handle Credit Payment.

...

Use Case Include Example (1c)

UC12: Handle Credit Payment

(the included use case)

...

Level: Subfunction

Main Success Scenario:

1. Customer enters their credit account information.
2. System sends request to external Payment Authorization Service System, and requests payment approval.
3. System receives payment approval, signals approval to Cashier
4. ...

Extensions:

- 2a. System detects failure to collaborate with external system:
1. System signals error to Cashier.
 2. Cashier asks Customer for alternate payment.

...

Use Case Include Example (2)

UC1: Process FooBars

(the including use case)

...

Main Success Scenario:

...

Extensions:

a*. At any time, Customer selects to edit personal information:

Include Edit Personal Information.

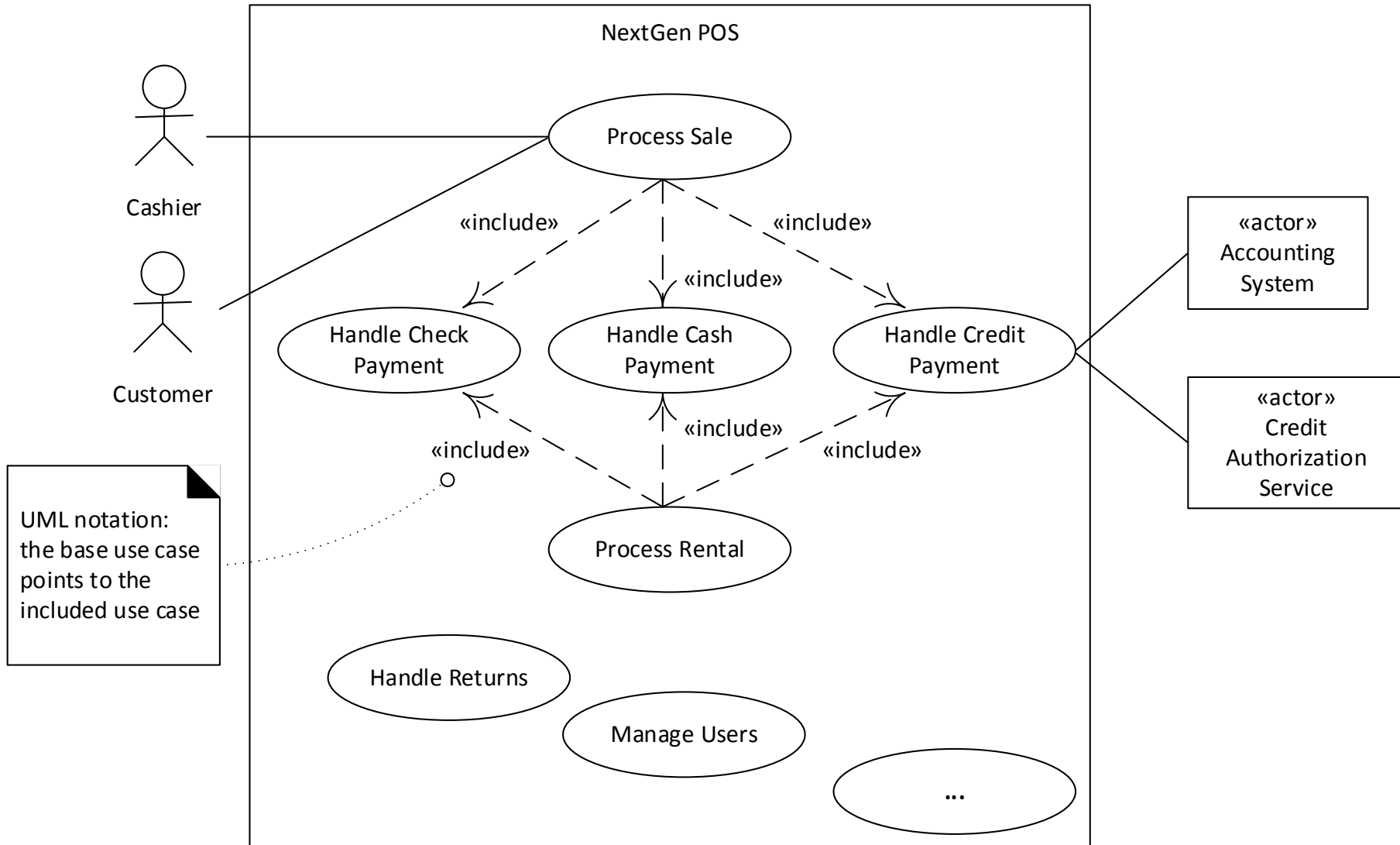
b*. At any time, Customer selects printing help:

Include Present Printing Help.

2-11. Customer cancels: *Include Cancel Transaction Confirmation.*

...

Use Case Include Relationship



Use Case Extend Relationship

Use *extend* when you want to add new extensions or conditional steps to a use case but can't or don't want to add to the text in the use case.

- ❑ Use when modifying *base use case* is undesirable
 - e.g. base is in current build, extension in next
- ❑ Base use case doesn't refer to *extending use case*
- ❑ Extending use case describes its relationship with the base use case
 - e.g. when extension is applicable
- ❑ Extending use case doesn't apply in its own right

Use Case Extend Example (1a)

UC1: Process Sale

(the base use case)

...

Extension Points: *VIP Customer*, step 1. *Payment*, step 7

Main Success Scenario:

1. Customer arrives at a POS checkout with goods and/or services to purchase.

...

7. Customer pays and System handles payment.

...

Use Case Extend Example (1b)

UC15: Handle Gift Certificate Payment

(the extending use case)

...

Trigger: Customer wants to pay with gift certificate.

Extension Points: Payment in *UC1: Process Sale*

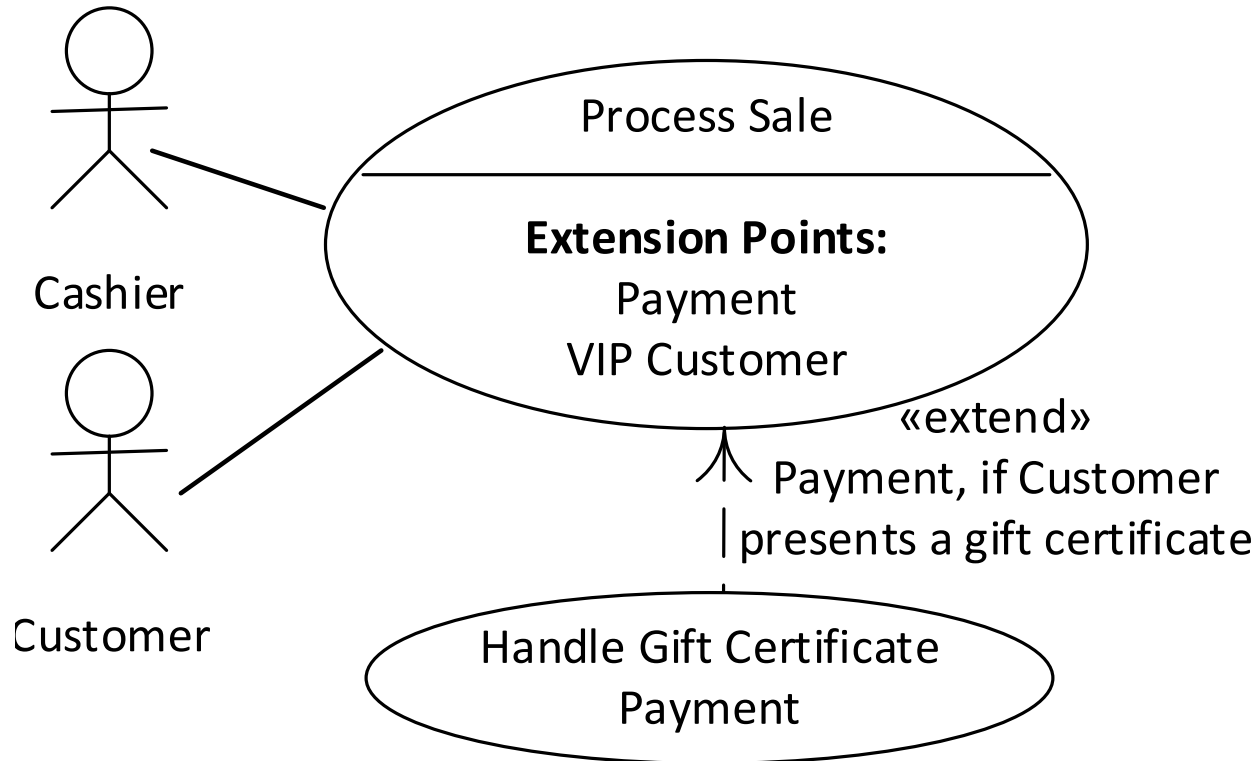
Level: Subfunction

Main Success Scenario:

1. Customer gives gift certificate to Cashier.
2. Cashier enters gift certificate ID.

...

Use Case Extend Relationship

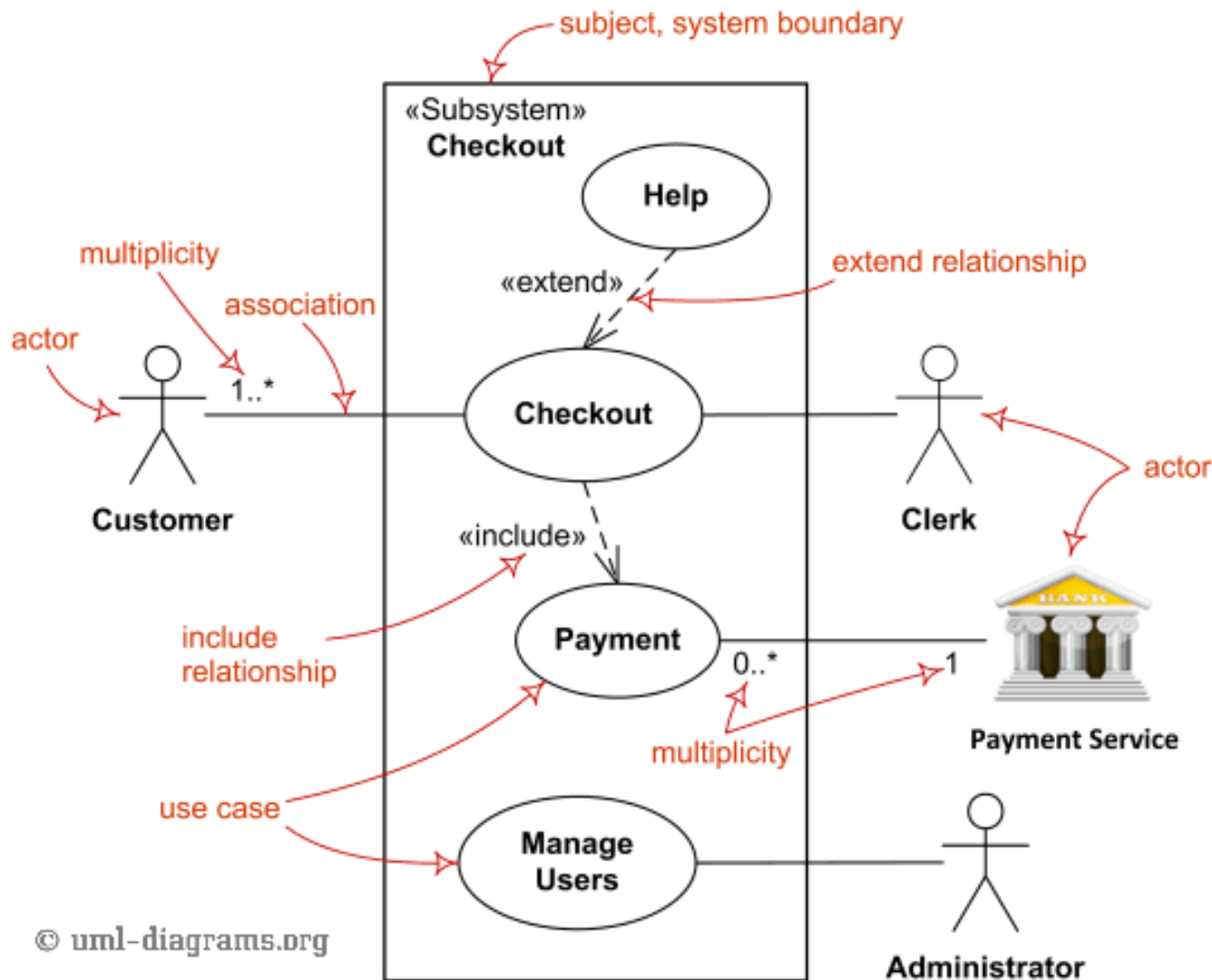


UML notation:

1. The extending use case points to the base use case.

2. The condition and extension point can be shown on the line.

Example with Extend and Include



Guidelines

- ❑ Don't agonize over relating use cases in diagrams: text is more important
- ❑ Use include relationships freely to avoid repetition or structure long use cases
- ❑ Avoid extend relationships, preferring to modify text of use case unless good reason not to
 - i.e. in most cases just update the *Extensions* section of the use case
- ❑ Primary actor may engage with included use case, but not with extending use case