SWEN30006
Software Modelling and Design

Operation Contracts
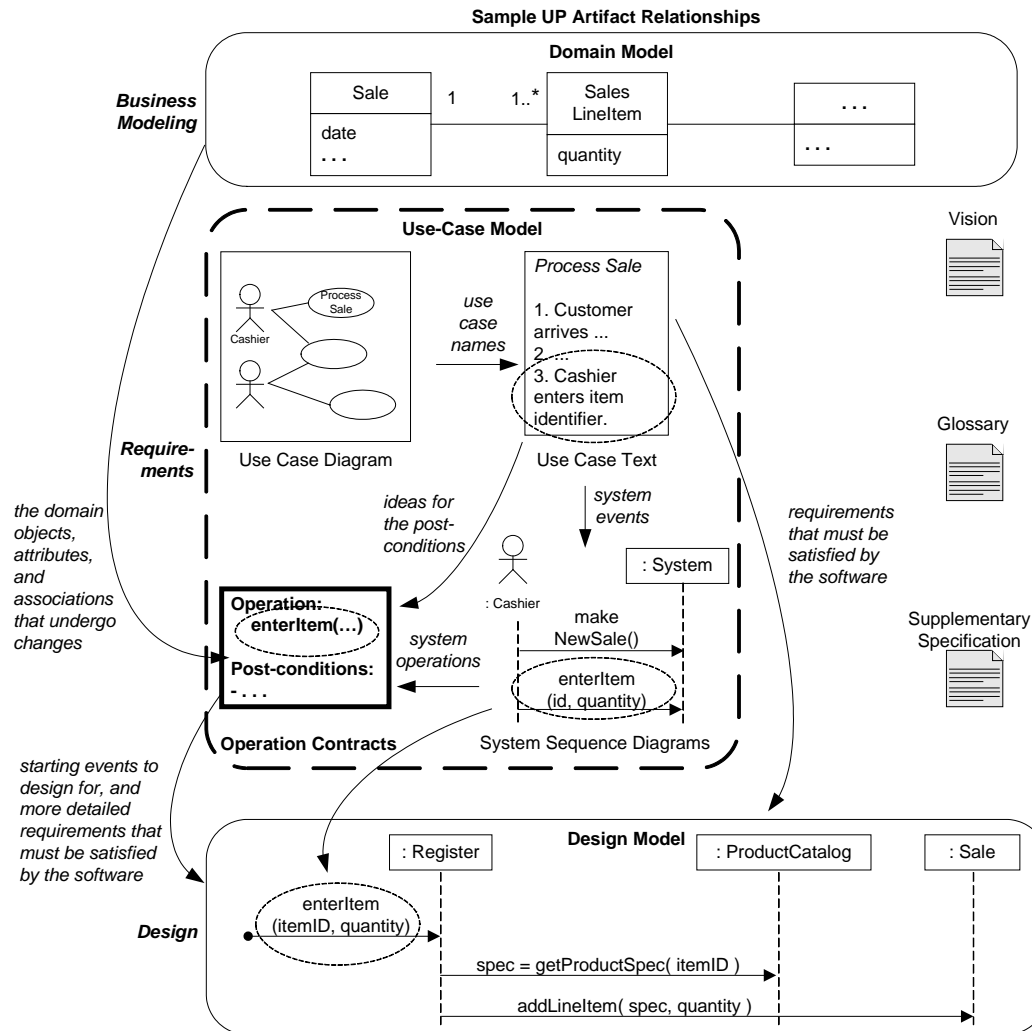
Larman Chapter 11

When ideas fail, words come in very handy.
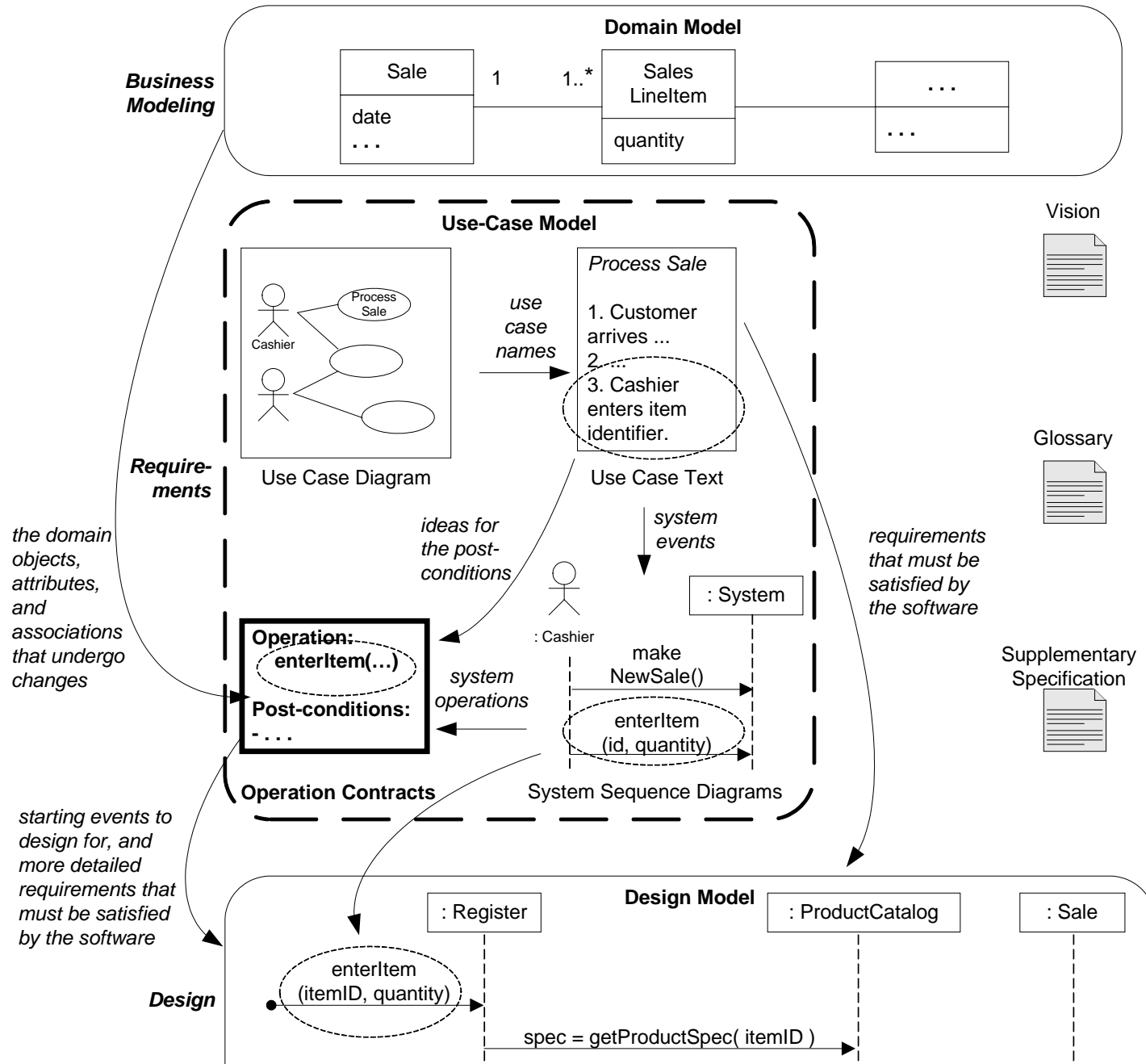*—Johann Wolfgang von Goethe*

Objectives

On completion of this topic you should be able to:

❑ Define system operations.
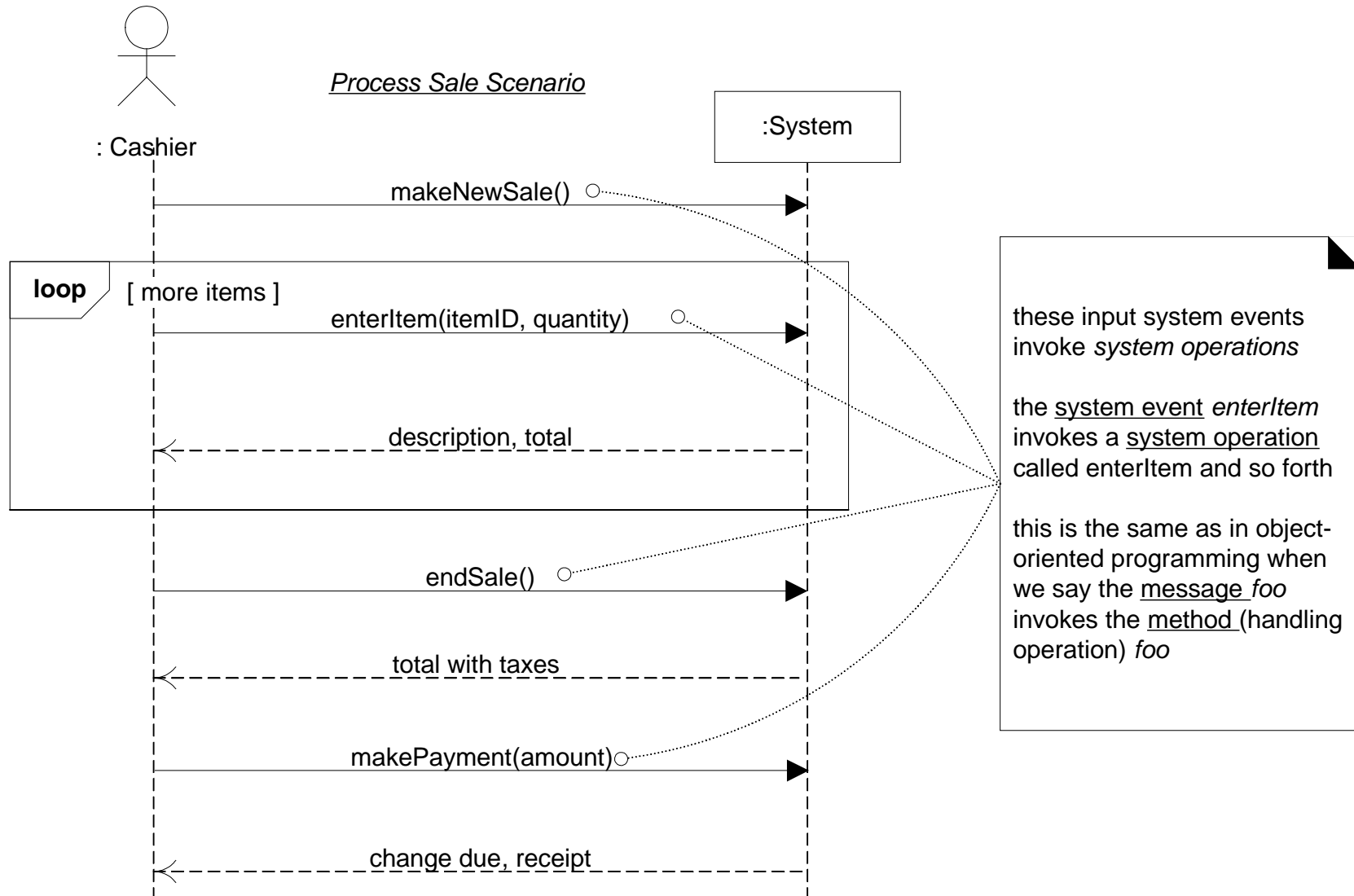
❑ Create contracts for system operations.

Sample UP Artefact Influence

Sample UP Artifact Relationships

Sample UP Artifact Relationships

System Ops handle Input Events

Process Sale Scenario

: Cashier   :System

makeNewSale()

loop [more items]

enterItem(itemID, quantity)

description, total

endSale()

total with taxes

makePayment(amount)

change due, receipt

these input system events invoke *system operations*

the system event *enterItem* invokes a system operation called enterItem and so forth

this is the same as in object-oriented programming when we say the message *foo* invokes the method (handling operation) *foo*

Contract CO2: enterItem

		Comments *(not part of contract)*
Operation:	enterItem(itemID: ItemID, quantity: integer)	
Cross References:	Use Cases: Process Sale	
Preconditions:	There is a sale underway.	
Postconditions:	• A SalesLineItem instance sli was created.	*instance creation*
	• sli was associated with the current Sale.	*association formed*
	• sli.quantity became quantity.	*attribute modification*
	• sli was associated with a ProductDescription, based on itemID match.	*association formed*

Contract Sections

Operation:	*Name of operation and parameters.*
Cross References:	*Use cases within which this operation can occur.*
Preconditions:	*Noteworthy assumptions about system state or objects in the Domain Model before execution of the operation. These are non-trivial assumptions the reader should be told.*
Postconditions:	*Most important section. State of objects in the Domain Model after completion of the operation.*

Complex operations,
not all operations

Postconditions

- ❑ describe changes by the operation to the state of objects in the domain model

  - ○ instances created or deleted

  - ○ changed attribute values

  - ○ associations formed or broken

- ❑ what happened? (not how!)

- ❑ past tense

  - ○ (better) A SalesLineItem was created

  - ○ (worse) Create a SalesLineItem

Analogy: Spirit of Pre/Post Conditions

The Stage and Curtains

The system and its objects on a theatre stage:

1. Before operation, take a picture of the stage.
2. Close the curtains, and apply the system operation (*background noise: clanging, voices, …*)
3. Open the curtains and take a second picture.
4. Compare before and after, express <u>essential</u>
   - starting elements as pre-conditions
   - changes as post-conditions

   Update domain model where required

Contract CO4: makePayment

Operation:	makePayment(amount: Money)	*Comments (not part of contract)*
Cross References:	Use Cases: Process Sale	
Preconditions:	There is a sale underway. Sale.isComplete is true.	*Domain addition + endSale postcondition*
Postconditions:	• A Payment instance p was created .	*instance creation*
	• p.amountTendered became amount.	*attribute modification*
	• p was associated with the current Sale .	*association formed*
	• The current Sale was associated (log of completed sales) with the Store.	*association formed*

Validation of payment:
Credit: authorisation
Cheque: licence/id
Cash: ??