

# 08 . Contratti

Sviluppo di Applicazioni Software

---

Matteo Baldoni

a.a. 2023/24

Università degli Studi di Torino - Dipartimento di Informatica

## Attenzione:



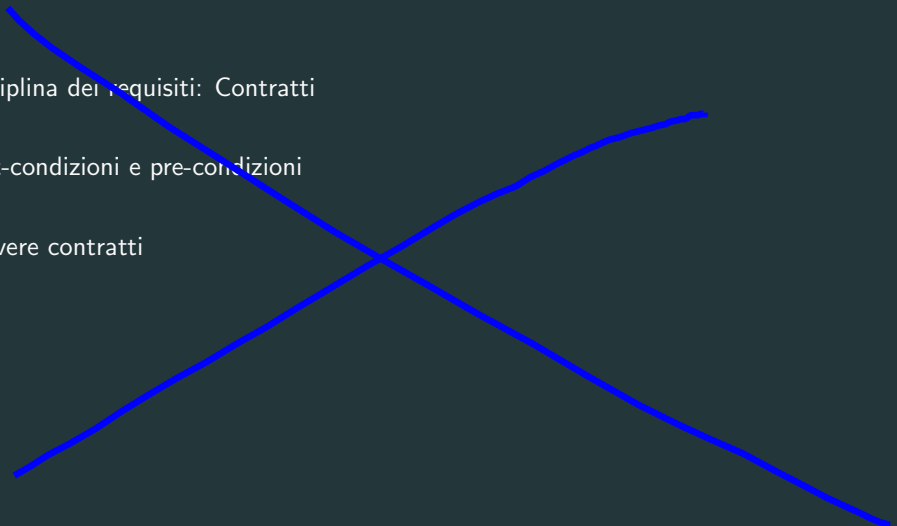
©2024 Copyright for this slides by Matteo Baldoni. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

<https://creativecommons.org/licenses/by/4.0/>

## Si noti che

questi lucidi sono basati sul libro di testo del corso “C. Larman, *Applicare UML e i Pattern*, Pearson, 2016” e parzialmente sul materiale fornito da Viviana Bono, Claudia Picardi e Gianluca Torta dell’Università degli Studi di Torino.

# Table of contents

- 
1. Disciplina dei requisiti: Contratti
  2. Post-condizioni e pre-condizioni
  3. Scrivere contratti

## Disciplina dei requisiti: Contratti

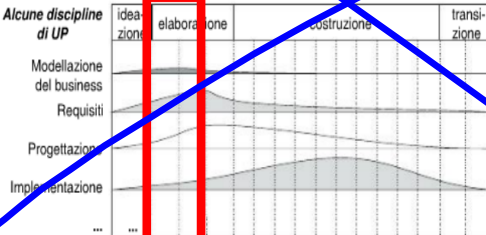
---

# UP maps

**Tabella 2.1** Scenario di Sviluppo di esempio (i – inizio; r – raffinamento).

Disciplina	Pratica	Elaborato Iterazione →	Ideazione II	Elaboraz. Et..En	Costr. Cl..Cn	Transiz. T1..T2
Modellazione	modellazione agile	Modello		i		
Requisiti	workshop requisiti esercizio sulla visione	Modello dei Casi d'Uso	i	r		
		Specifica Supplementare Glossario	i	r		
Progettazione	modellazione agile sviluppo guidato ai test	Modello di Progetto Documento dell'Architettura Software Modello dei Dati		i	r	
Implementazione	sviluppo guidato dai test programmazione a coppie integrazione continua standard di codifica	...				
Gestione del progetto	gestione del progetto agile riunioni Scrum giornaliere	...				
...						

*Alcune discipline  
di UP*



L'impegno relativo  
nelle discipline cambia  
a seconda delle fasi.

Questo esempio è solo  
un suggerimento, non è  
da prendere alla lettera.

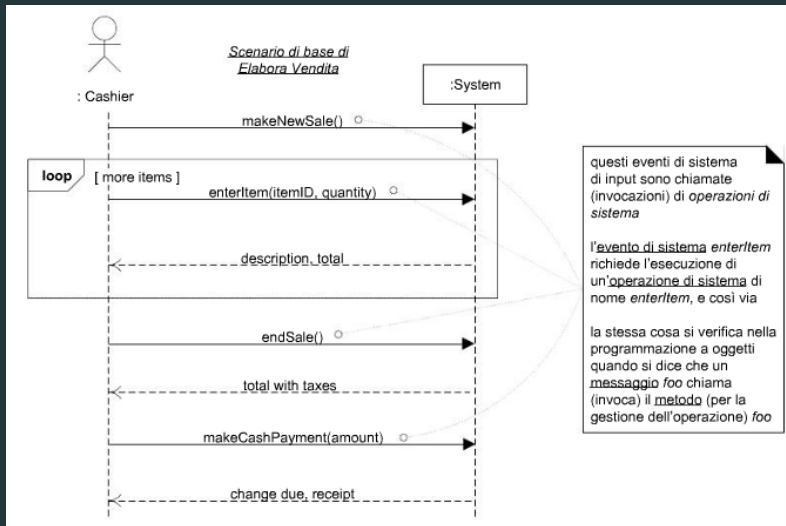
# Operazioni di sistema

- Le operazioni di sistema sono operazioni che il sistema, considerato come un componente a scatola nera, offre nella sua interfaccia pubblica.
- Le operazioni di sistema possono essere identificate mentre si abbozzano gli SSD. Gli SSD mostrano eventi di sistema.

## Eventi e operazioni

Un evento di sistema implica che il sistema definisca un'operazione di sistema per gestire quell'evento.

# Eventi e operazioni di sistema



## I contratti delle operazioni

usano pre-condizione e post-condizione per descrivere nel dettaglio i cambiamenti agli oggetti (*concettuali*) in un modello di dominio.

**Nota:** possono essere considerati parte del Modello dei Casi d'Uso, poiché forniscono maggiori dettagli dell'analisi sull'effetto delle operazioni di sistema implicate dai casi d'uso.

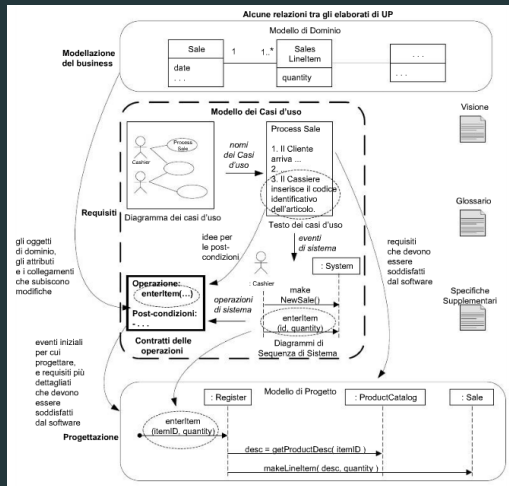
Non sono menzionati esplicitamente in UP.



# Relazioni tra elaborati di UP

I principali input per i contratti sono le operazioni di sistema identificate negli **SSD**, il **modello di dominio**, e l'esame del dominio da parte degli esperti.

I contratti servono come **input per la progettazione a oggetti**.



### Contratto CO2: enterItem

---

**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

**Pre-condizioni:** è in corso una vendita s

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (*creazione di oggetto*).
- sli è stata associata con la Sale (vendita) corrente s (*formazione di collegamento*).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (*formazione di collegamento*).
- sli.quantity è diventata quantity (*modifica di attributo*).

- **Operazione:** Nome e parametri (*firma*) dell'operazione.
- **Riferimenti:** Casi d'uso in cui può verificarsi questa operazione.
- **Pre-condizioni:** Ipotesi significative sullo stato del sistema o degli oggetti nel Modello di Dominio prima dell'esecuzione dell'operazione. Si tratta di ipotesi non banali, che dovrebbero essere comunicate al lettore.
- **Post-condizione:** È la sezione più importante. Descrive i cambiamenti di stato degli oggetti nel Modello di Dominio dopo il completamento dell'operazione.

## Post-condizioni e pre-condizioni

---

# Definizione di post-condizione

## Le post-condizioni

descrivono i cambiamenti nello stato degli oggetti del modello di dominio. I cambiamenti dello stato del modello di dominio comprendono gli **oggetti creati**, i **collegamenti formati** o **rotti**, e gli **attributi modificati**.

Le post-condizioni **non sono azioni da eseguire nel corso dell'operazione**; si tratta di **osservazioni** sugli oggetti del modello di dominio che risultano al **termine** dell'operazione.

# Definizione di post-condizione

## Le post-condizioni

descrivono i cambiamenti nello stato degli oggetti del modello di dominio. I cambiamenti dello stato del modello di dominio comprendono gli oggetti creati, i collegamenti formati o rotti, e gli attributi modificati.

Le post-condizioni **non** sono azioni da eseguire nel corso dell'operazione; si tratta di osservazioni sugli oggetti del modello di dominio che risultano al termine dell'operazione.

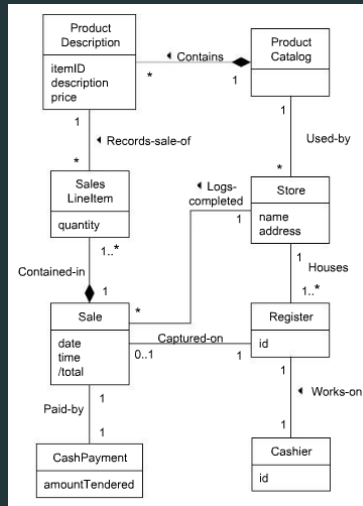
**Nota:** è un punto di vista **concettuale**, non di implementazione: si possono creare gli oggetti corrispondenti alle classi del **Modello di Dominio**, come si possono formare i collegamenti corrispondenti alle associazioni del **Modello di Dominio**.

# Adottare un punto di vista concettuale e della conoscenza

## Il contratto per un'operazione di sistema

descrive (specialmente la post-condizione) i cambiamenti nello stato del sistema causato dall'esecuzione di un'operazione di sistema. Questo cambiamento va espresso in **termini di oggetti e collegamenti del Modello di Dominio**, secondo un **punto di vista concettuale** (ovvero si parla di oggetti e collegamenti del mondo reale o nel modello di interesse).

A lato il Modello di Dominio di riferimento negli esempi di seguito.



## Contratti: che cosa vs come

### Un contratto

è uno strumento ottimo nell'**analisi dei requisiti** o nell'analisi orientata agli oggetti per descrivere in modo molto dettagliato i cambiamenti richiesti dall'esecuzione di una operazione di sistema (in termini di oggetti del Modello di Dominio), **senza però descrivere come** devono essere ottenuti questi cambiamenti.

La progettazione si può rimandare e ci si può concentrare sull'analisi di che cosa deve accadere, anziché su come deve essere ottenuto.



# Contratti: che cosa vs come

## Contratto CO2: enterItem

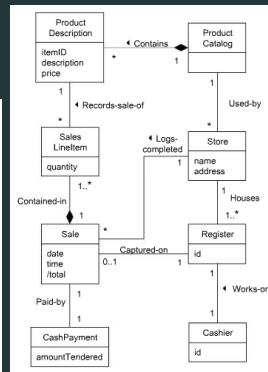
**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

**Pre-condizioni:** è in corso una vendita s

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (*creazione di oggetto*).
- sli è stata associata con la Sale (vendita) corrente s (*formazione di collegamento*).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (*formazione di collegamento*).
- sli.quantity è diventata quantity (*modifica di attributo*).



©C. Larman. Applicare UML e i Pattern. Pearson, 2016.

Si noti che

non viene fatto alcun commento su come venga creata l'istanza *SalesLineItem* o su come venga associata a una *Sale*.

# Contratti: che cosa vs come

## Contratto CO2: enterItem

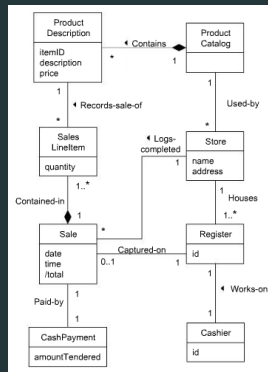
**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

**Pre-condizioni:** è in corso una vendita s

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (*creazione di oggetto*).
- sli è stata associata con la Sale (vendita) corrente s (*formazione di collegamento*).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (*formazione di collegamento*).
- sli.quantity è diventata quantity (*modifica di attributo*).



©C. Larman. Applicare UML e i Pattern. Pearson, 2016.

Le post-condizioni vanno espresse con il tempo del verbo al passato

“**È stata creata** una SalesLineItem” piuttosto che “Creare una SalesLineItem” oppure “Viene creata una SalesLineItem”.

# Contratti: che cosa vs come

## Contratto CO2: enterItem

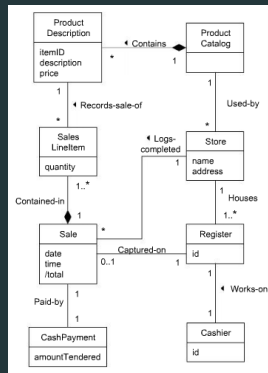
**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

**Pre-condizioni:** è in corso una vendita s

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (*creazione di oggetto*).
- sli è stata associata con la Sale (vendita) corrente s (*formazione di collegamento*).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (*formazione di collegamento*).
- sli.quantity è diventata quantity (*modifica di attributo*).



©C. Larman. Applicare UML e i Pattern. Pearson, 2016.

Si noti l'assegnazione

di un nome all'istanza, q questo permetterà di fare riferimento alla nuova istanza nelle altre post-condizioni.

# Contratti: che cosa vs come

## Contratto CO2: enterItem

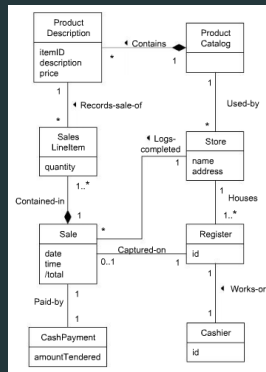
**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

**Pre-condizioni:** è in corso una vendita s

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (*creazione di oggetto*).
- sli è stata associata con la Sale (vendita) corrente s (*formazione di collegamento*).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (*formazione di collegamento*).
- sli.quantity è diventata quantity (*modifica di attributo*).



©C. Larman. Applicare UML e i Pattern. Pearson, 2016.

## Formazione o rottura di collegamenti

La nuova *SalesLineItem* deve essere stata collegata alla sua *Sale* corrente e collegata anche a una sua *ProductDescription*.

# Contratti: che cosa vs come

## Contratto CO2: enterItem

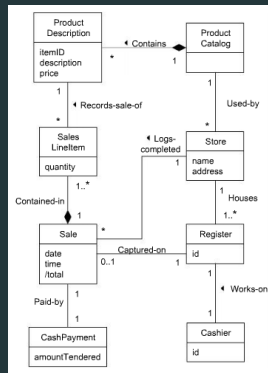
**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

**Pre-condizioni:** è in corso una vendita s

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (*creazione di oggetto*).
- sli è stata associata con la Sale (vendita) corrente s (*formazione di collegamento*).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (*formazione di collegamento*).
- sli.quantity è diventata quantity (*modifica di attributo*).



©C. Larman. Applicare UML e i Pattern. Pearson, 2016.

## Modifica di attributi

La *quantity* della *SalesLineItem* creata deve essere diventata pari al parametro *quantity*.

# Definizione di pre-condizione

## Le pre-condizioni

descrivono le ipotesi significative sullo stato del sistema **prima** dell'esecuzione dell'operazione.

Una **descrizione sintetica** dello stato di avanzamento del caso d'uso.

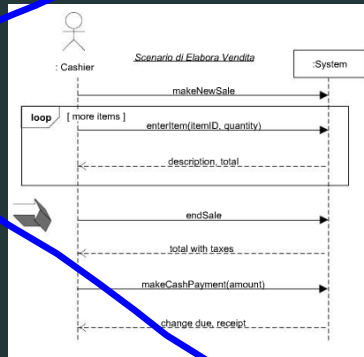
# Definizione di pre-condizione

## Le pre-condizioni

descrivono le ipotesi significative sullo stato del sistema **prima** dell'esecuzione dell'operazione.

Una **descrizione sintetica** dello stato di avanzamento del caso d'uso.

Per esempio, l'operazione *enterItem* viene eseguita quando una vendita è già iniziata ma non è ancora conclusa. Lo stato di avanzamento del caso d'uso si può riassumere con la pre-condizione "è *in corso una vendita*".



# Definizione di pre-condizione

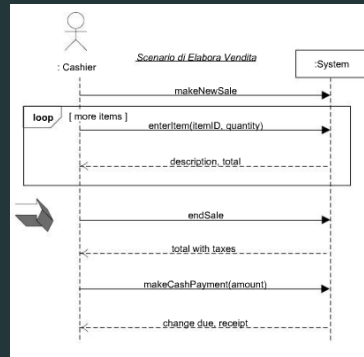
## Le pre-condizioni

descrivono le ipotesi significative sullo stato del sistema **prima** dell'esecuzione dell'operazione.

Una **descrizione sintetica dello stato di avanzamento del caso d'uso.**

Per esempio, l'operazione *enterItem* viene eseguita quando una vendita è già iniziata ma non è ancora conclusa. Lo stato di avanzamento del caso d'uso si può riassumere con la pre-condizione "è *in corso una vendita*".

**Nota:** nelle pre-condizioni è utile indicare gli **oggetti rilevanti a quel punto del caso d'uso, in particolare quelli che si vogliono citare nelle post-condizioni**. Ad esempio, "sì è stata associata con la Sale corrente s", la vendita corrente *s* va citata nelle pre-condizioni.





## Scrivere contratti

---

# Come creare e scrivere i contratti

Si proceda come segue:

1. Identificare le operazioni di sistema dagli SSD
2. Creare un contratto per le operazioni di sistema complesse o i cui effetti sono probabilmente sottili, o che non sono chiare dai casi d'uso
3. Per descrivere le post-condizioni si utilizzino le seguenti sotto-categorie:
  - creazione o cancellazione di oggetto (o istanza)
  - formazione o rottura di collegamento
  - modifica di attributo

# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita

## Contratto CO1: makeNewSale

**Operazione:** makeNewSale()

**Riferimenti:** casi d'uso: Elabora Vendita

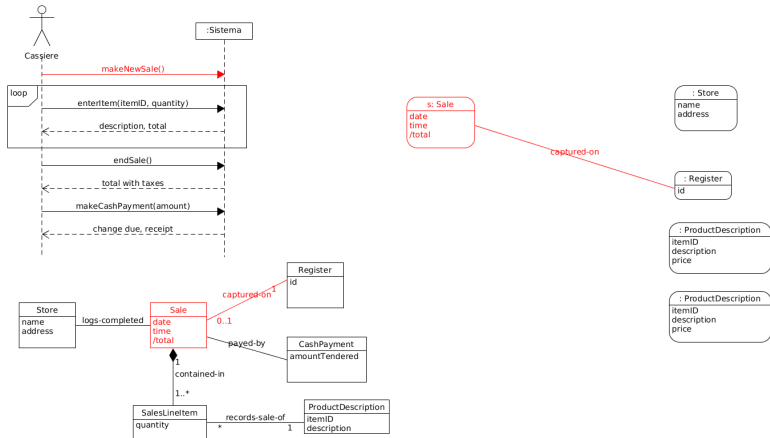
**Pre-condizioni:** nessuna

**Post-condizioni:**

- è stata creata un'istanza s di Sale (creazione di oggetto).
- s è stata associata con il Register (formazione di collegamenti).
- gli attributi di s sono stati inizializzati (modifica di attributi).

© C. Larman. Applicare UML e i Pattern. Pearson,

2016.



# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita

## Contratto C02: enterItem

**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

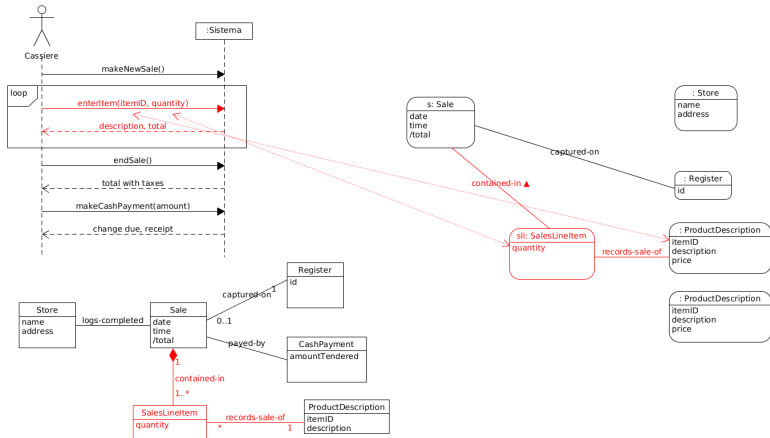
**Pre-condizioni:** è in corso una vendita s.

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLinItem (creazione di o).
- sli è stata associata con la Sale (vendita) corrente s (formazione di collegamento).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (formazione di collegamento).
- sli.quantity è diventata quantity (modifica di attributo).

©C. Larman. Applicare UML e i Pattern. Pearson,

2016.



# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita

## Contratto C02: enterItem

**Operazione:** enterItem(itemID: ItemID, quantity: integer)

**Riferimenti:** casi d'uso: Elabora Vendita

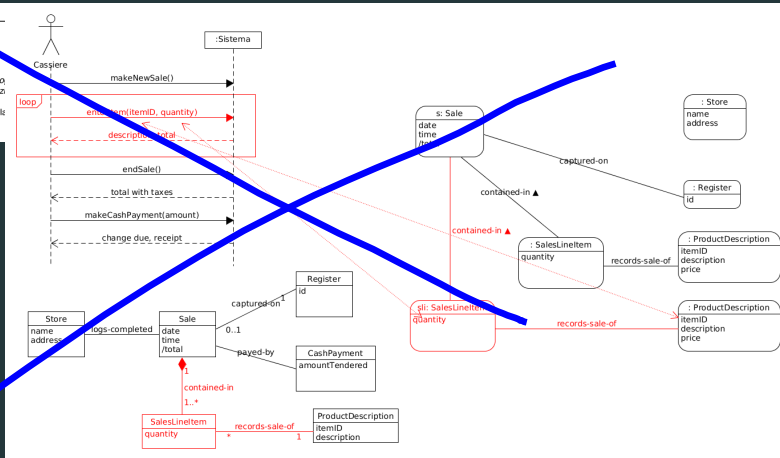
**Pre-condizioni:** è in corso una vendita s.

**Post-condizioni:**

- è stata creata un'istanza sli di SalesLineItem (creazione di o).
- sli è stata associata con la Sale (vendita) corrente s (formazione di collegamento).
- sli è stata associata con una ProductDescription, in base alla corrispondenza con itemID (formazione di collegamento).
- sli.quantity è diventata quantity (modifica di attributo).

©C. Larman. Applicare UML e i Pattern. Pearson,

2016.



# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita

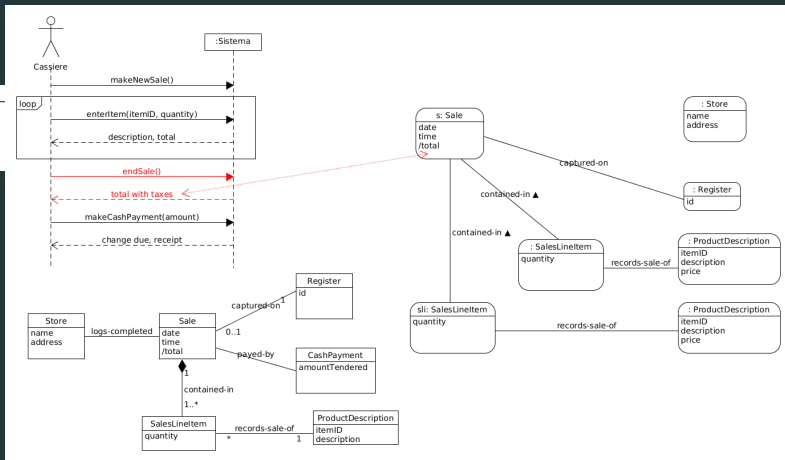
*Di questo contratto ne  
parliamo tra poco...*

## Contratto CO3: endSale

**Operazione:** endSale()  
**Riferimenti:** casi d'uso: Elabora Vendita  
**Pre-condizioni:** è in corso una vendita s.  
**Post-condizioni:** — nessuna.

©C. Larman. Applicare UML e i Pattern. Pearson,

2016.



# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita

## Contratto CO4: makeCashPayment

**Operazione:** makeCashPayment( amount: Money )

**Riferimenti:** casi d'uso: Elabora Vendita

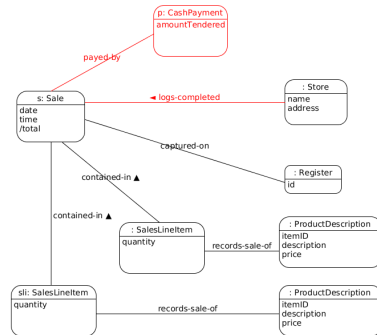
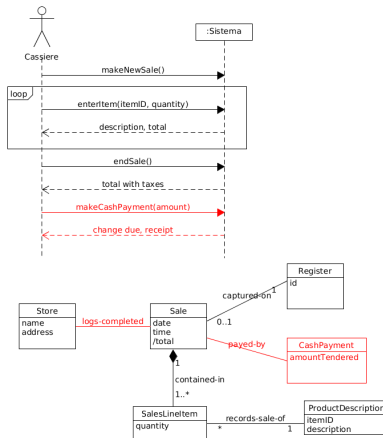
**Pre-condizioni:** è in corso una vendita s.

**Post-condizioni:**

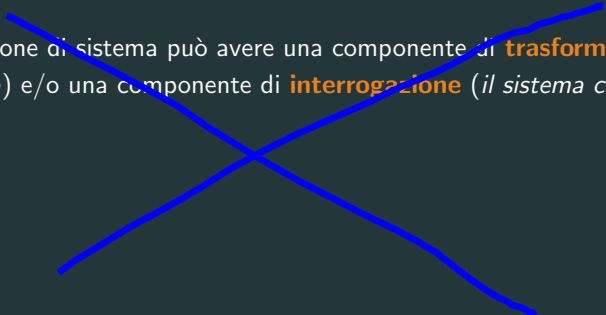
- è stata creata un'istanza p di CashPayment (creazione di oggi)
- p è stata associata con la Sale (vendita) corrente s (formazione collegamento).
- p.amountTendered è diventato amount (modifica di attributo).
- la Sale corrente s è stata associata con lo Store (formazione collegamento); (s è stata aggiunta al registro storico delle vendite completate).

©C. Larman. Applicare UML e i Pattern. Pearson,

2016.



Ogni operazione di sistema può avere una componente di **trasformazione** (*il sistema cambia il proprio stato*) e/o una componente di **interrogazione** (*il sistema calcola e restituisce valori*).





Ogni operazione di sistema può avere una **componente di trasformazione** (*il sistema cambia il proprio stato*) e/o una **componente di interrogazione** (*il sistema calcola e restituisce valori*).

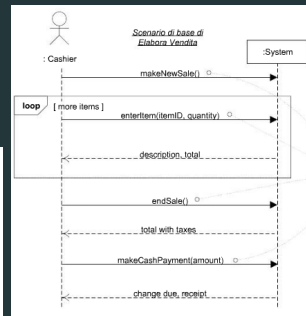
## Un'operazione di sistema

ha **post-condizioni solo se implica una trasformazione**ne, mentre **non ha post-condizioni** se si tratta semplicemente di un'interrogazione.

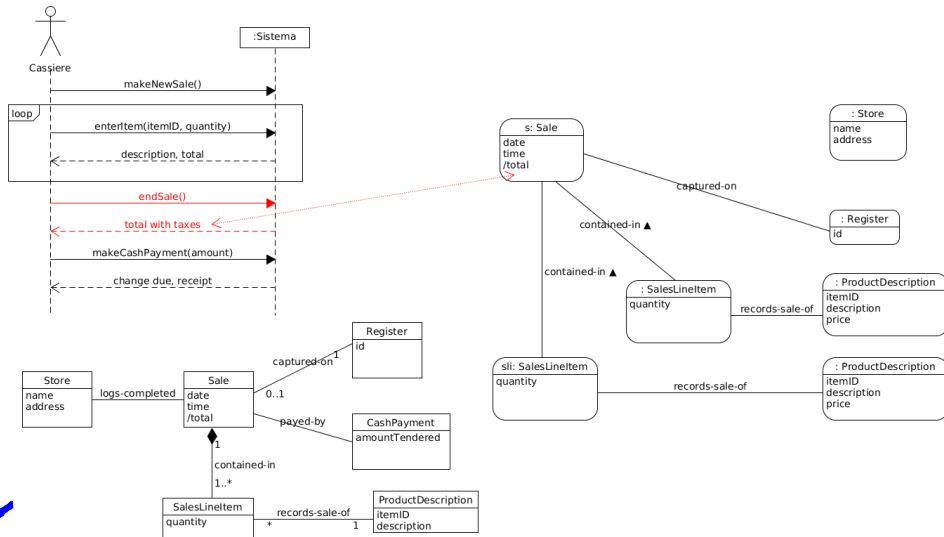
# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita

## Contratto CO3: endSale

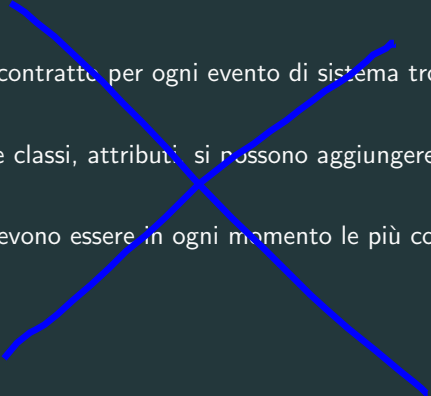
**Operazione:** endSale()  
**Riferimenti:** casi d'uso: Elabora Vendita  
**Pre-condizioni:** è in corso una vendita s.  
**Post-condizioni:** – nessuna.



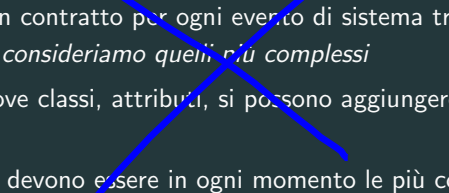
# Esempio: contratti per POS NextGen, caso d'uso Elabora Vendita



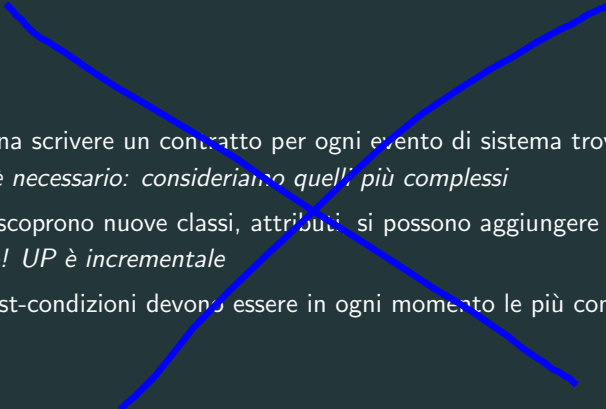
## Come creare e scrivere i contratti

- 
- Bisogna scrivere un contratto per ogni evento di sistema trovato nel SSD?
  - Se si scoprono nuove classi, attributi, si possono aggiungere nel modello di dominio?
  - Le post-condizioni devono essere in ogni momento le più complete possibili ?

## Come creare e scrivere i contratti

- 
- Bisogna scrivere un contratto per ogni evento di sistema trovato nel SSD?  
*Non è necessario: consideriamo quelli più complessi*
  - Se si scoprono nuove classi, attributi, si possono aggiungere nel modello di dominio?
  - Le post-condizioni devono essere in ogni momento le più complete possibili ?

## Come creare e scrivere i contratti

- 
- Bisogna scrivere un contratto per ogni evento di sistema trovato nel SSD?  
*Non è necessario: consideriamo quelli più complessi*
  - Se si scoprono nuove classi, attributi, si possono aggiungere nel modello di dominio?  
*Ovvio! UP è incrementale*
  - Le post-condizioni devono essere in ogni momento le più complete possibili ?

# Come creare e scrivere i contratti

- Bisogna scrivere un contratto per ogni evento di sistema trovato nel SSD?  
*Non è necessario: consideriamo quelli più complessi*
- Se si scoprono nuove classi, attributi, si possono aggiungere nel modello di dominio?  
*Ovvio! UP è incrementale*
- Le post-condizioni devono essere in ogni momento le più complete possibili ?  
*Non è necessario: UP iterativo ed incrementale*