

Corso di Laurea in Ingegneria Informatica

Corso di Ingegneria del Software

Il linguaggio UML Casi d'uso

Sommario

Il linguaggio UML: Introduzione

Casi d'uso

Diagramma dei casi d'uso

Esempi

Riferimenti:

J.Arlow, I. Neustadt: UML 2 e Unified Process,
2° edizione, McGraw-Hill. **Capp.** 4, 5.

C. Savy: Da C++ a UML: guida alla progettazione,
McGraw-Hill. **Cap.** 32.

UML (Unified Modeling Language)

- ✓ L' **UML** è un linguaggio di modellazione utilizzabile per **specificare, progettare, rappresentare o documentare** qualunque tipo di sistema:
 - sistemi software, hardware, organizzativi;
 - NON necessariamente software object-oriented.

- ✓ L' UML **non (!)** ...
 - ... è un linguaggio di programmazione;
 - ... fornisce una *metodologia* di modellazione,
ma piuttosto la sintassi da utilizzare per costruire modelli;
 - ... è un processo di sviluppo.



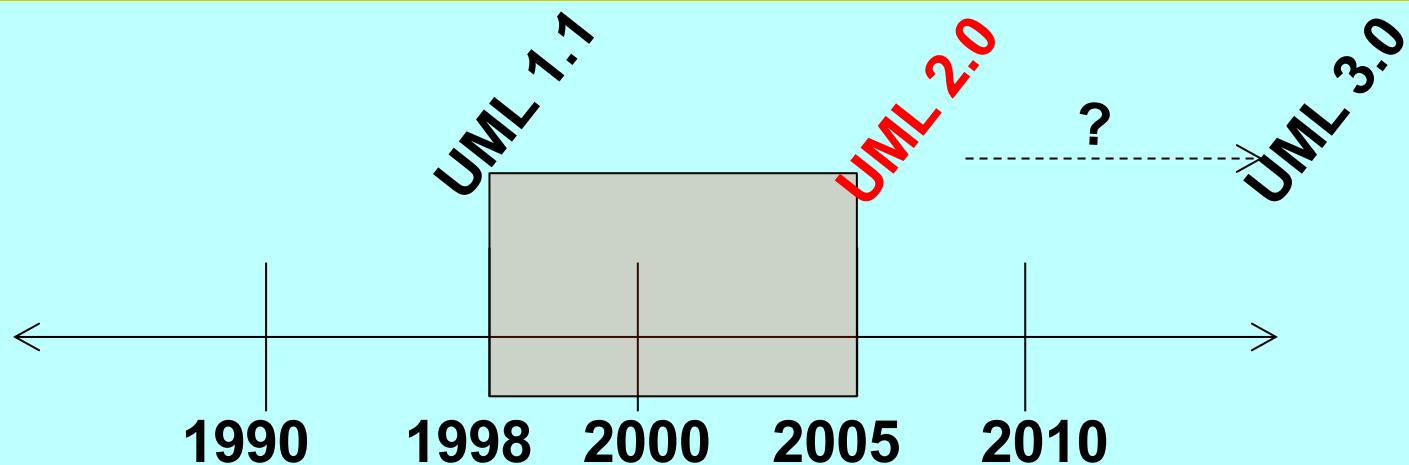
Alcuni cenni storici

- ✓ Prima del **1994** esistevano diversi linguaggi di modellazione visuale e metodologie OO:
 - **Grady Booch (metodo Booch)** e **Jim Rumbaugh** (Object Modeling Tecjnicue, **OMT**) erano i leader nel mercato dei linguaggi di modellazione visuale;
 - **Ivar Jacobson** era di gran lunga avanti nel campo delle metodologie. Introdusse il concetto di *use cases*;
 - Altri players: Schlaer/Mellor “Recursive Design” approach, Coad/Yourdon “Prototype-oriented” approach.

Dal 1994 in poi ...

- ✓ **1994**: Booch e Rumbaugh si uniscono alla Rational Corporation per lavorare sull' UML.
- ✓ Due eventi significativi avvengono nel **1995**:
 - viene rilasciata la prima descrizione pubblica del metodo unificato Booch+Rumbaugh: **Unified Method**;
 - **Jacobson** si unisce allo *unified team*.
- ✓ Durante il **1996** Booch, Rumbaugh e Jacobson (chiamati i Three Amigos) lavorano al nuovo metodo rinominato **Unified Modeling Language**
- ✓ **1997** UML diviene uno **standard OMG** (*Object Management Group*) dal novembre del 1997 i cui autori sono Booch, Rumbaugh e Jacobson.

UML timeline



QUESTION

✓ Transizione da UML 1.x a UML 2:

- rafforza sintassi e semantica degli elementi di modellazione a livello di meta-modello UML;
- introduce nuovi notazioni e ne migliora le pre-esistenti;
- include nuovi concetti di origine non- e pre-OO: attività, macchine a stati, strutture composite,

Il Meta-modello: the meta-levels game

Is this a pipe?



Credits: Bran Selic

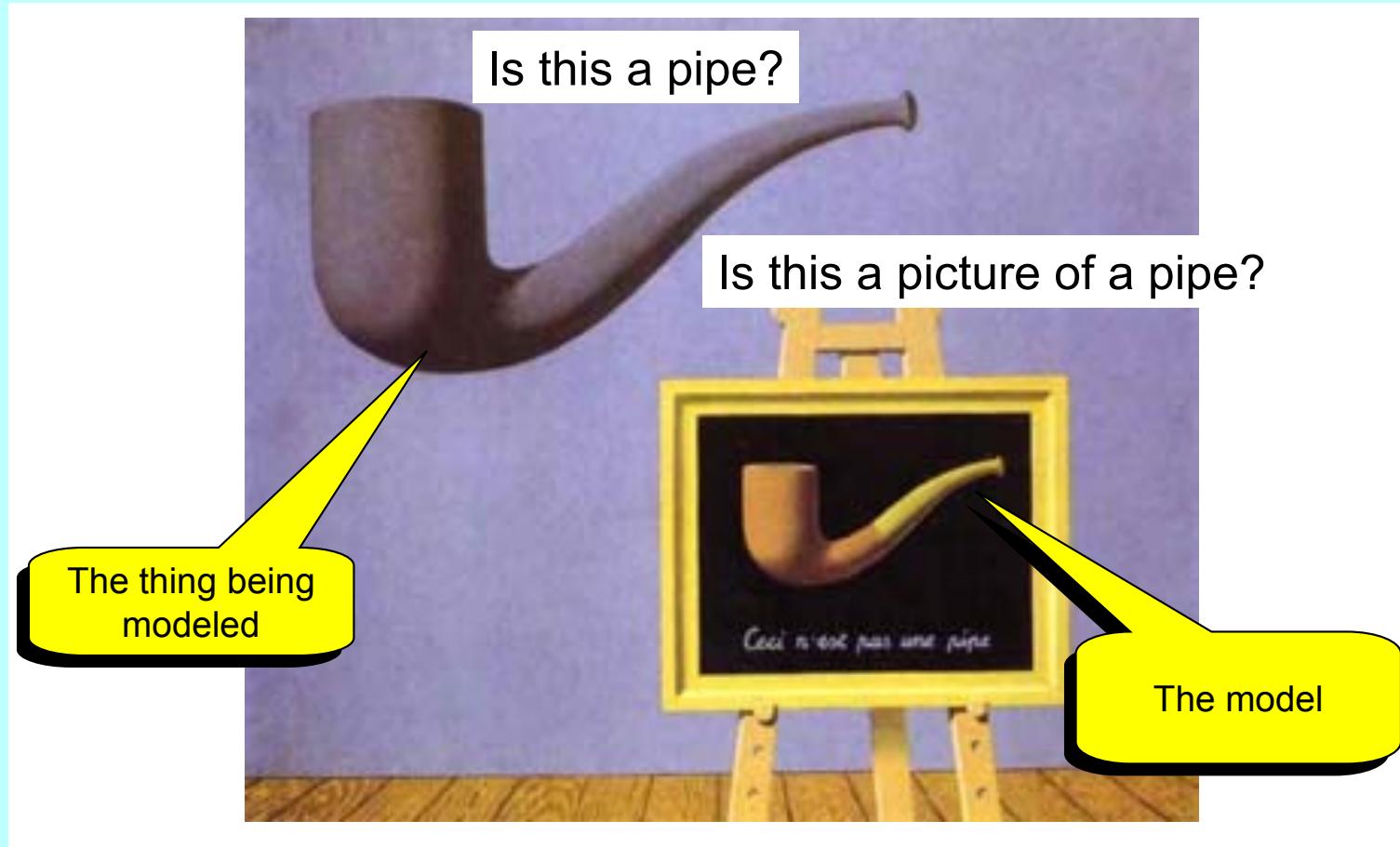
Il Meta-modello: the meta-levels game

No, this is a picture of a pipe



Credits: Bran Selic

Il Meta-modello: the meta-levels game



Credits: Bran Selic

Il Meta-modello: conclusioni

♦ **Meta-levels can be very confusing**

- It is easy to confuse a specification/model with the actual thing that is being modeled
- ...especially in informal colloquial usage

♦ **Meta-levels can extend to infinity**

- picture (model) of a picture (model) of a picture (model)...

♦ **There is a singularity at the origin (level 0)**

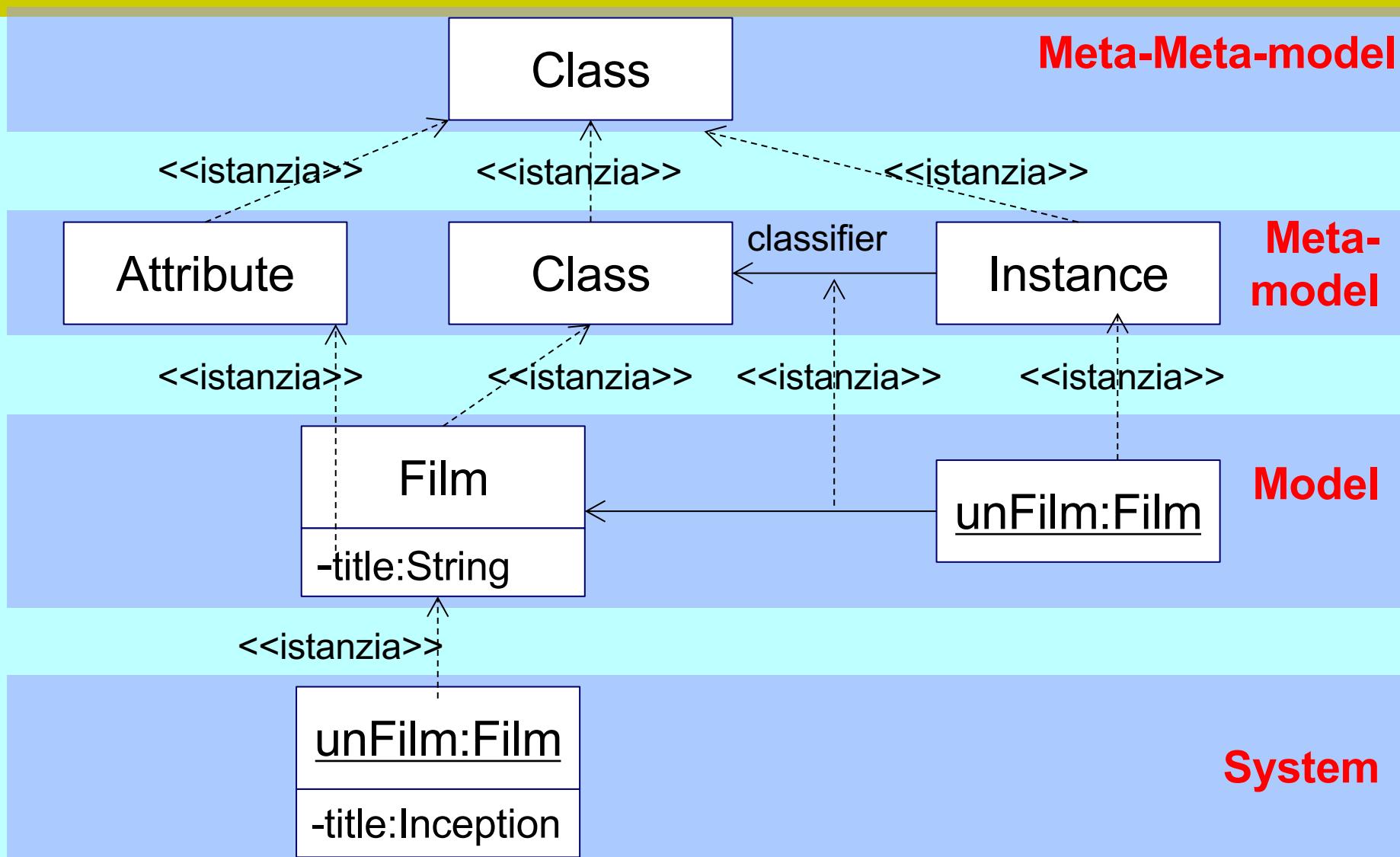
- Physical reality is not a model
- All levels above this level represent models

Credits: Bran Selic

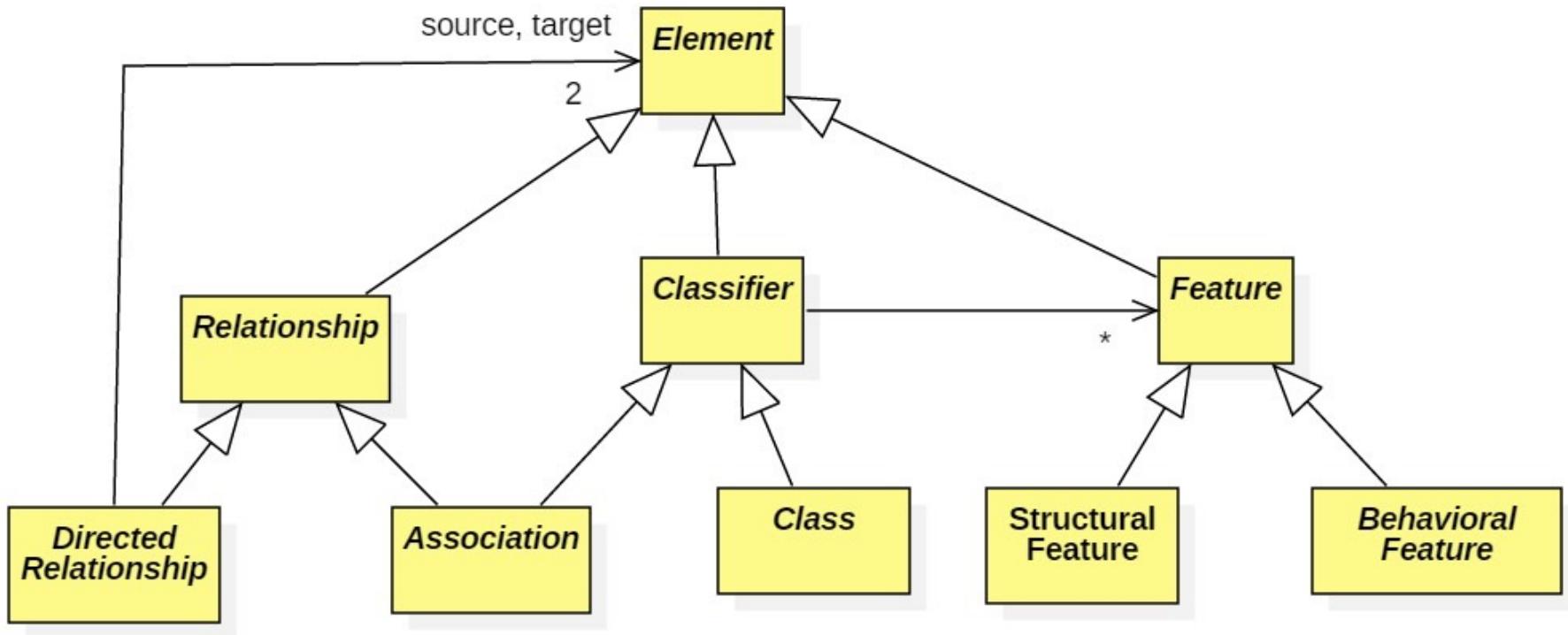
Meta-modello UML

- ❖ L' UML è basato su un **meta-modello**
 - Un meta-modello è un linguaggio per specificare modelli. Esso fornisce una singola, comune e definitiva struttura della sintassi e della semantica degli elementi UML (descrizione dell' UML *in* UML).
- ❖ Utilizzando gli elementi del meta-modello è possibile creare i modelli per i sistemi da rappresentare
- ❖ Il meta-modello è composto da numerosi elementi – **oggetti, attributi, relazioni** - collegati tra loro secondo regole precise che permettono di verificare la correttezza dei modelli creati

Meta-modello UML: Livelli



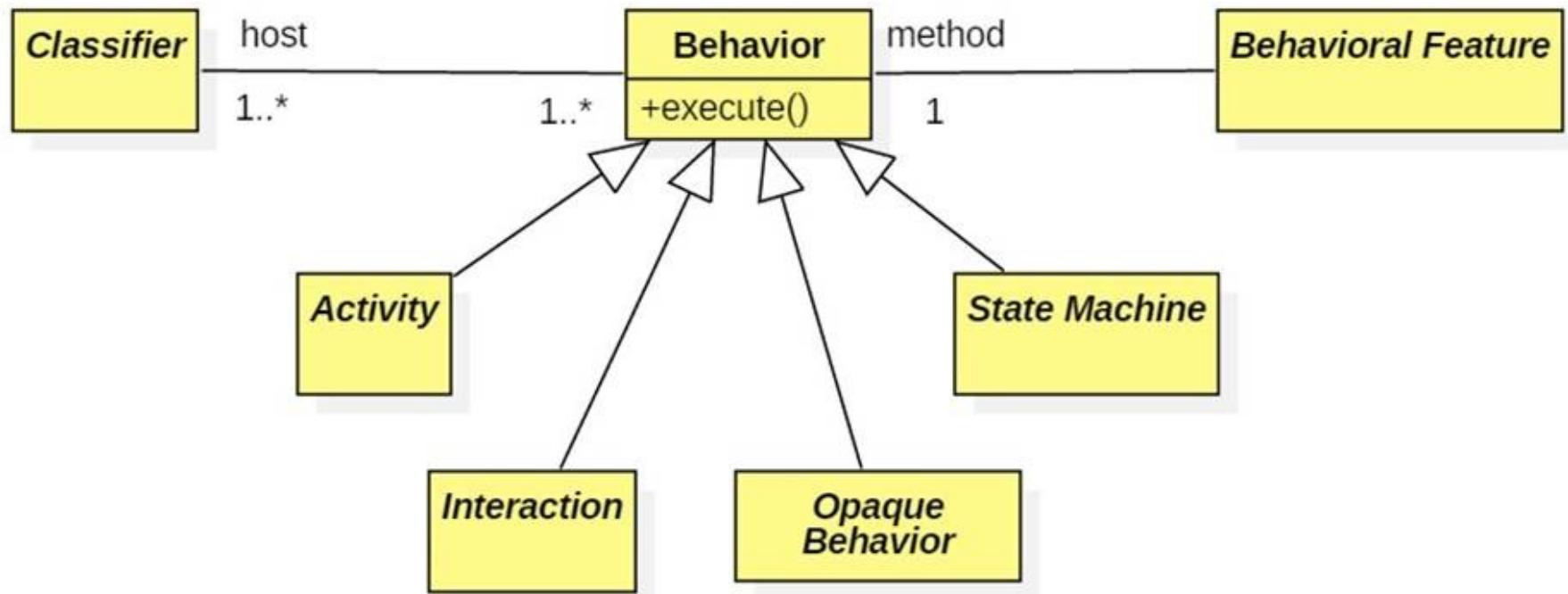
Meta-modello UML: Estratto



Riferimenti:

The Unified Modeling Language; Kirill Fakhroutdinov; <http://www.uml-diagrams.org/>.
OMG Unified Modeling Language; <http://www.omg.org/spec/UML/> .

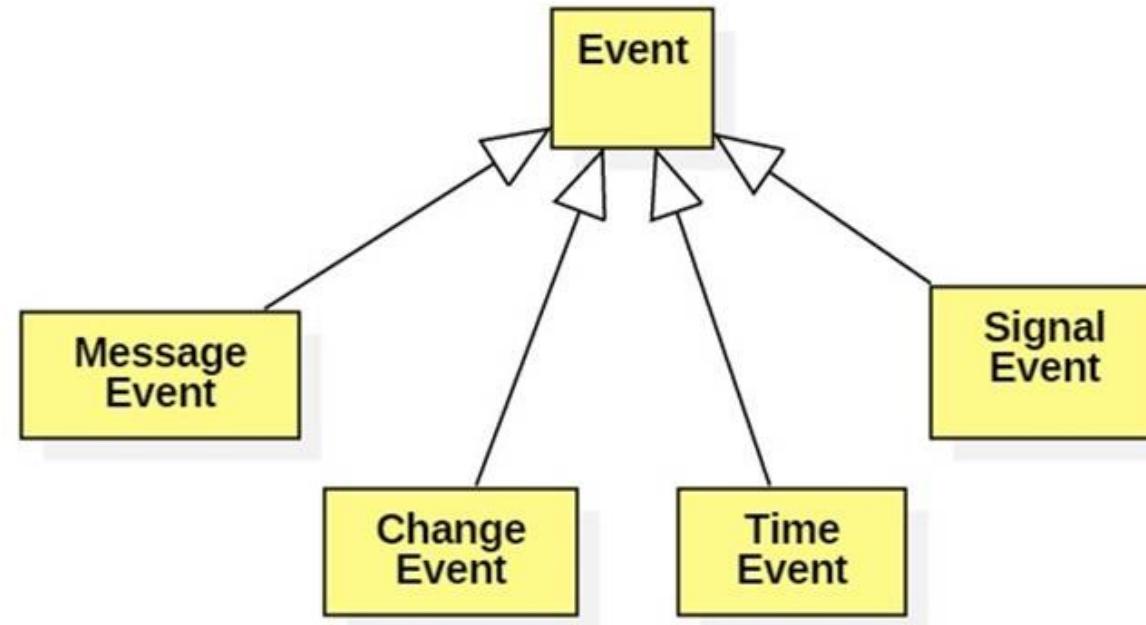
Meta-modello UML: Estratto



Riferimenti:

The Unified Modeling Language; Kirill Fakhroutdinov; <http://www.uml-diagrams.org/>.
OMG Unified Modeling Language; <http://www.omg.org/spec/UML/> .

Meta-modello UML: Estratto



Riferimenti:

The Unified Modeling Language; Kirill Fakhroutdinov; <http://www.uml-diagrams.org/>.
OMG Unified Modeling Language; <http://www.omg.org/spec/UML/> .

Meta-modello UML

Principali elementi:

- **Classifier**: ad es., *classi, use case, per la modellazione strutturale e Relationships*
- **Structural e Behavioraul Features** (*ad es. attributi e operazioni nel caso delle classi*)
- **Behaviour**: ad es., *interazioni, activity, state machine, per la modellazione comportamentale*
- **Event**: catturano “occorrenze” (*attivano un behaviour*)
- Altri elementi
- Informativi
 - Note
- Di raggruppamento e contenimento
 - Ad es., package

I diagrammi di UML

- ❖ Ogni diagramma è progettato per fornire a sviluppatori e clienti (attori) la **vista** di un sistema software da diverse prospettive e a vari gradi di astrazione.
- ❖ **Diagrammi per modellare gli aspetti statici e dinamici (comportamentali):**
 - **Diagramma dei casi d'uso (use case diagram)**
 - Diagramma delle classi (class diagram)
 - Diagramma di sequenza (sequence diagram)
 - Diagramma di comunicazione (communication diagram)
 - Diagramma di stato (state–chart diagram)
 - Diagramma delle attività (activity diagram)
- ❖ **Diagrammi di implementazione:**
 - Diagramma dei componenti (component diagram)
 - Diagramma di distribuzione (deployment diagram)

I diagrammi in UML 2

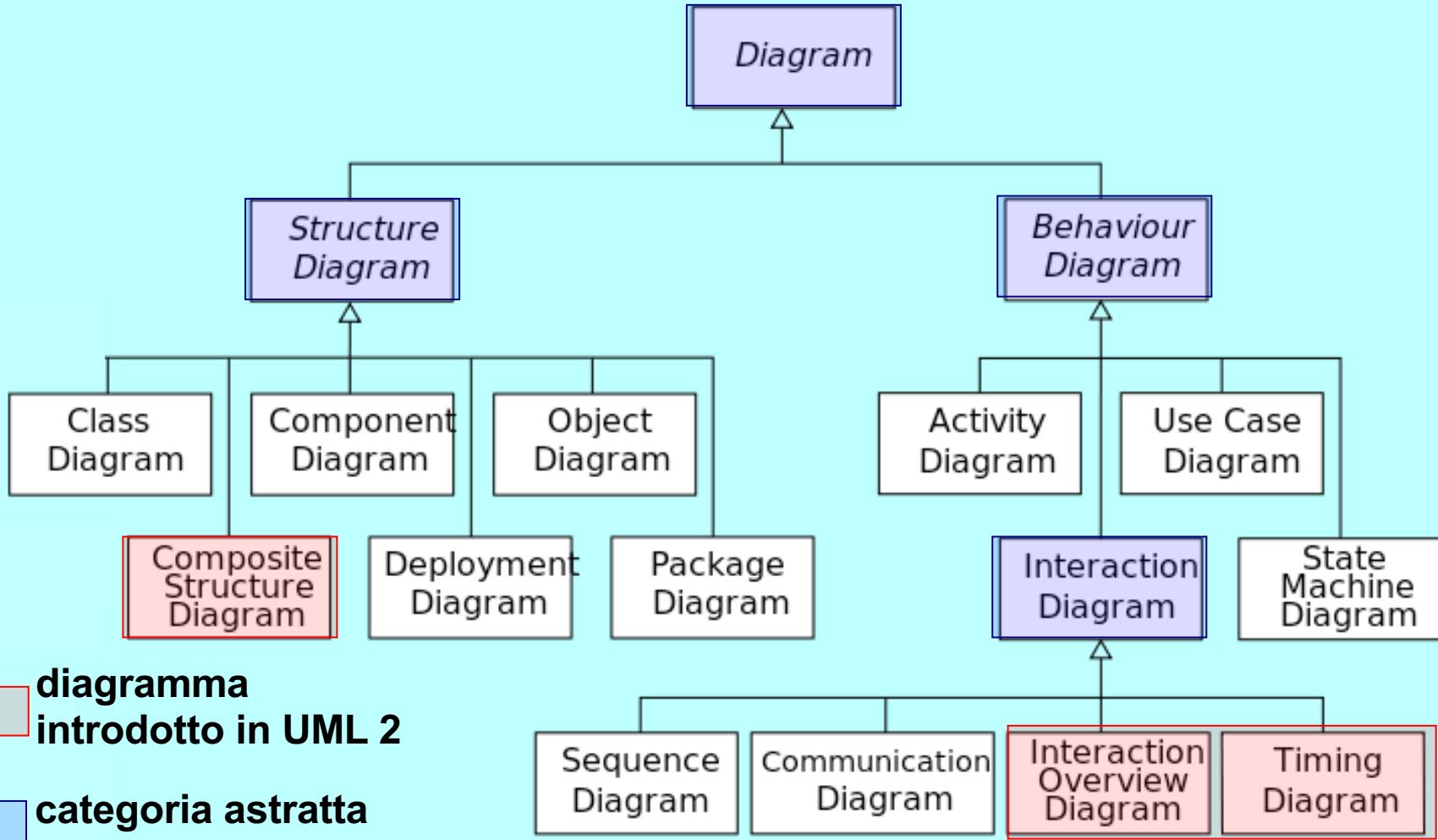


diagramma
introdotto in UML 2

categoria astratta
di diagramma

Modellazione dei casi d'uso

I Casi d'Uso: cosa sono?

- ♦ Sono storie scritte, testuali, di qualche attore che usa un sistema per raggiungere degli obiettivi. Sono utilizzate per scoprire e registrare i requisiti di un sistema
- ♦ Inflenzano molti aspetti di un progetto, dall'analisi alla progettazione ad oggetti
- ♦ Esempio di Caso d'uso: **Elabora Vendita**
 - *Un cliente arriva alla cassa con alcuni articoli da acquistare. Il cassiere usa un POS per registrare ogni articolo acquistato. Il sistema mostra il totale e i dettagli per ogni articolo. Il cliente inserisce informazioni sul pagamento, che il sistema convalida e registra. Il sistema aggiorna l'inventario. Il cliente ottiene una ricevuta ...*

Generalità

- ◆ L'analisi dei casi d'uso viene impiegata in fase di analisi per modellare il comportamento esterno del sistema da sviluppare, senza specificare come esso viene realizzato
 - Il sistema viene visto, al suo interno, come una **scatola nera** (black box)
 - Il comportamento del sistema è analizzato dal punto di vista dei suoi possibili **utenti**
 - ◆ Per questo motivo si presta particolarmente ad essere compilato a valle di interviste al committente

Caso d'uso

- ✓ Un **caso d'uso** è una tipica interazione tra un attore ed il sistema per svolgere un'unità di lavoro utile
- ✓ Un caso d'uso non rivela l'organizzazione interna del sistema (non dice come è fatto il sistema, ma cosa deve fare)
- ✓ L'insieme dei casi d'uso rappresenta le funzionalità del sistema (i servizi che esso offre agli attori)
- ✓ La modellazione dei casi d'uso è solitamente effettuata in forma grafica, spesso comprensibile anche ai “non addetti ai lavori”, attraverso un diagramma UML dei casi d'uso

Caso d'uso

- ◆ La descrizione di un caso d'uso specifica cosa accade nel sistema in seguito ad uno stimolo
- ◆ Generalmente lo **stimolo** parte da un **attore**, ma può anche essere il sistema stesso ad iniziare il caso d'uso (es.: aggiornamento automatico inventario)
- ◆ Un caso d'uso corrisponde ad un compito
 - che l'attore chiede al sistema di eseguire (l'attore inizia il caso d'uso)
- oppure
 - che il sistema esegue autonomamente (il sistema inizia il caso d'uso).

Modellazione dei casi d'uso

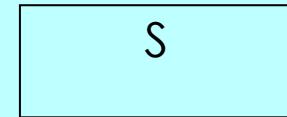
Un modello dei casi d'uso consiste di:

- Confini del sistema e nome del sist. (Subject)
- Attori
- Casi d'uso
- Comunicazione tra attori e casi d'uso
- Relazioni tra attori, relazioni tra casi d'uso

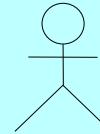
Elementi del diagramma dei casi d'uso

In un diagramma UML dei casi d'uso (*use case diagram*) si rappresentano i seguenti elementi:

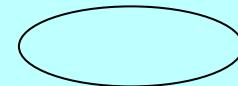
- Confine del sistema e *SubjectS*



- Attori



- Casi d'uso



- Comunicazione (tra attori e casi d'uso)



- Relazioni tra attori, relazioni tra casi d'uso

Subject

♦ **Subject (o Confine del sistema)**

- Rappresenta il limite tra ciò che è interno al sistema da sviluppare e ciò che è esterno ad esso.
- È rappresentato da un rettangolo etichettato col nome del sistema.

♦ Il rettangolo è il perimetro che delimita il sistema

- Ciò che è interno ai confini andrà progettato, realizzato, verificato e validato, e costituirà il prodotto software

Sistema di vendite

Attore

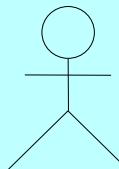
Gli attori sono esterni al sistema.

Un attore di un caso d'uso può essere:

- Una classe di persone fisiche (es. Fornitore)
- Un altro sistema software (es. Sistema di contabilità)
- Un dispositivo hardware esterno (es. Sensore)
- Il tempo (es.: per una funzionalità periodica)

Un attore di un caso d'uso è detto:

- ♦ **Attore primario**, se fornisce lo stimolo che avvia il caso d'uso
- ♦ **Attore secondario**, se interagisce con il caso d'uso dopo che è stato avviato.



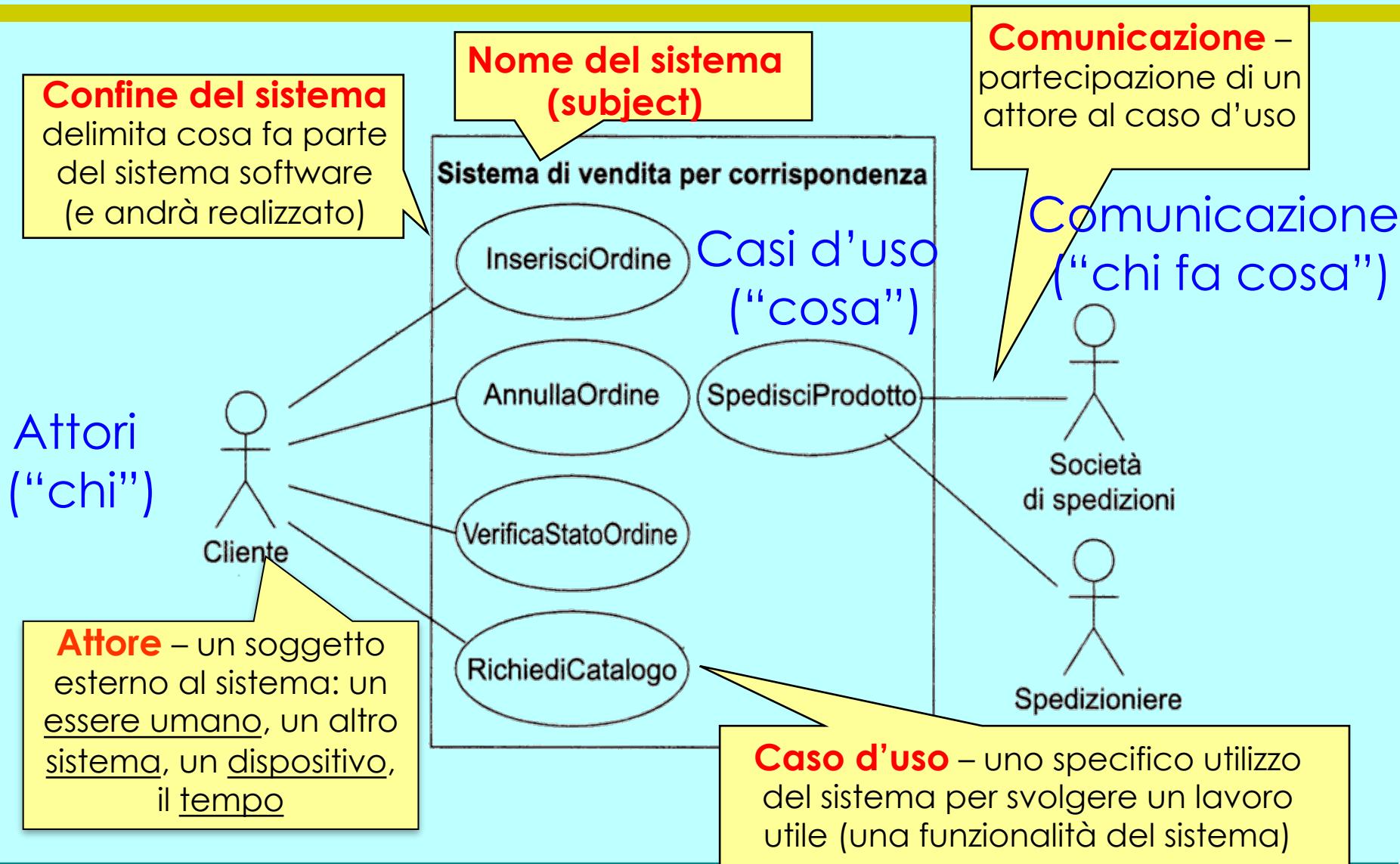
Nome Attore

Caso d'uso

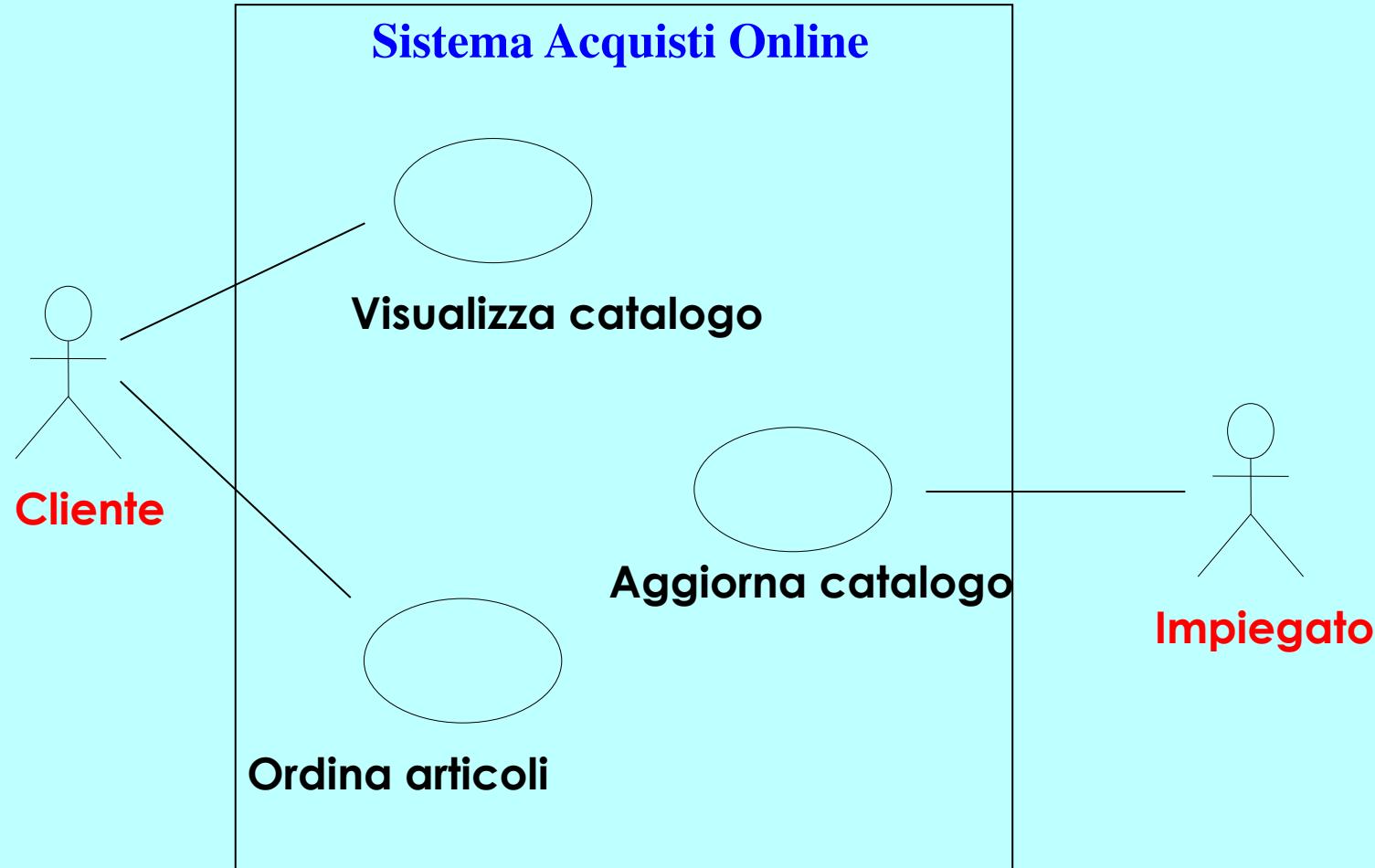
- ◆ È una unità di lavoro utile (una elaborazione) che il sistema esegue stimolato dall'attore primario (se esiste) o comunque interagendo con uno o più attori
- ◆ Es. di caso d'uso: Stampa Cedolino Stipendio
- ◆ Ragionare sui casi d'uso aiuta a scoprire i requisiti funzionali del sistema
 - Tipicamente si procede per raffinamenti successivi

Nome del caso d'uso

Diagramma UML dei casi d'uso

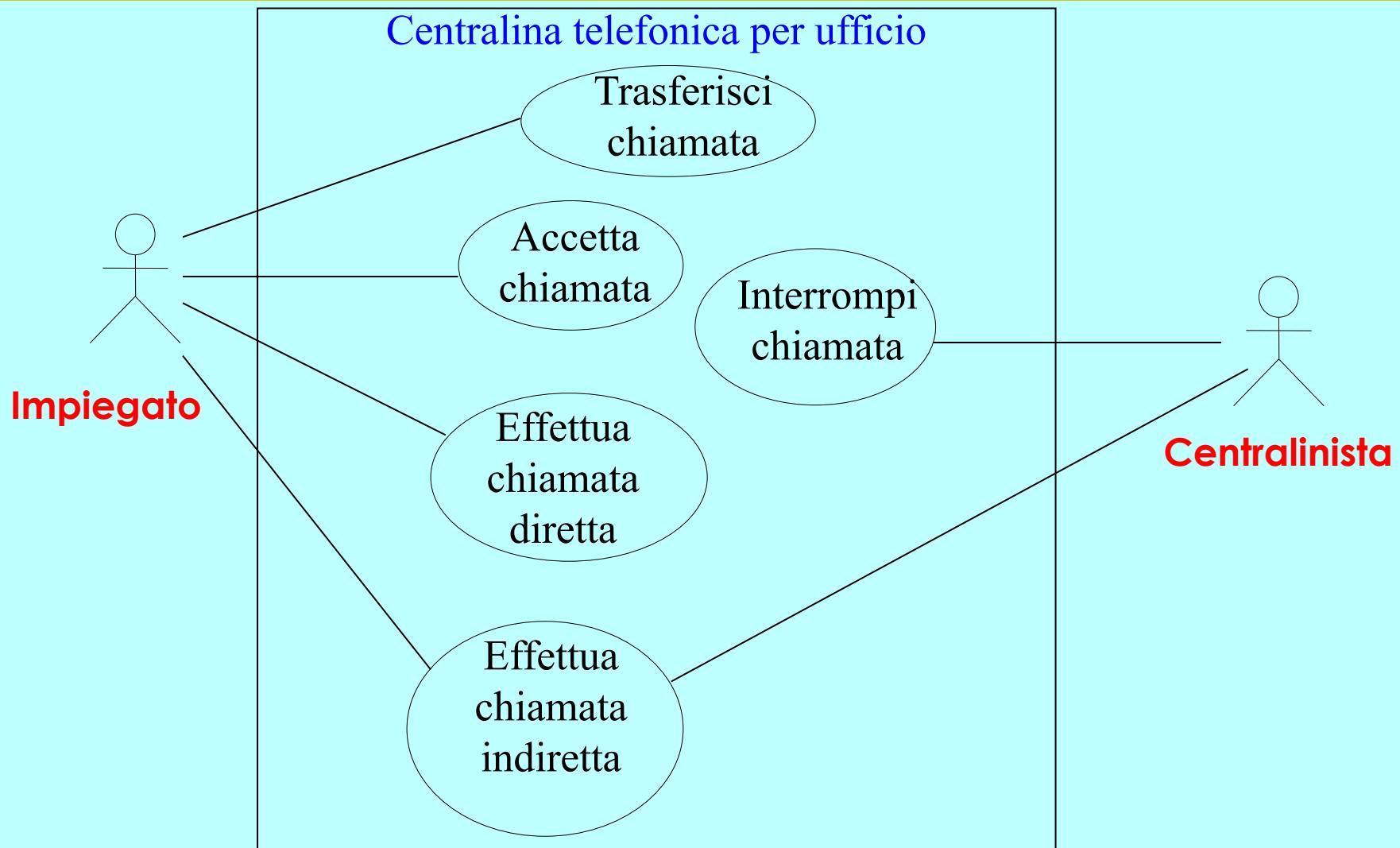


Esempio: sistema di acquisti on-line



Esempio

Centralina telefonica per ufficio



Scenari

- ◆ Un caso d'uso va descritto tramite un insieme di **scenari** di interazione tra gli attori ed il sistema.
- ◆ Uno scenario è una sequenza di azioni ed interazioni fra sistema ed attori.
- ◆ Esempio: Caso d'uso “Effettua Ordine”
 - il **cliente** richiede l'elenco dei prodotti
 - il **sistema** propone i prodotti disponibili
 - il **cliente** sceglie i prodotti che desidera
 - il **sistema** fornisce il costo totale dei prodotti selezionati
 - il **cliente** conferma l'ordine
 - il **sistema** comunica l'accettazione dell'ordine

Scenari

- ◆ Uno **scenario** definisce cosa accade nel sistema in seguito all'evento di innesco:
 - come e quando il caso d'uso inizia
 - chi inizia il caso d'uso
 - interazione tra attore/i e caso d'uso e cosa viene scambiato
 - come e quando c'è bisogno di dati memorizzati o di memorizzare dati
 - come e quando il caso d'uso termina
- ◆ Per ciascun caso d'uso sono previsti scenari **normali** e scenari **alternativi** (o eccezionali)
- ◆ Stili di descrizione:
 - testuali, con un flusso chiaro di eventi da seguire
 - diagrammatici: diagramma di stato, di sequenza, di interazione
 - ◆ Essendo diagrammi utili anche in fase di progettazione, verranno presentati in futuro

Scenari e Casi d'uso

- ◆ Informalmente, un caso d'uso è una collezione di scenari correlati, di successo e fallimento, che descrivono un attore che usa un sistema per raggiungere un obiettivo.
- ◆ Esempio di scenari per “Effettua un Ordine”:
 - 1) Scenario di successo (normale): il cliente effettua normalmente l'ordine
 - 2) Scenario alternativo: il cliente non conferma l'ordine ...
 -

Specifiche di uno scenario

- ✓ Per ciascun caso d'uso, è opportuno documentare gli scenari, ad es. con una scheda strutturata per ciascuno scenario
- ✓ UML non definisce uno standard, ogni azienda/organizzazione ne definisce uno proprio

nome del caso d'uso	Caso d'uso: Paga IVA
identificatore del caso d'uso	ID: 1
breve descrizione	Breve descrizione: Pagamento dell'IVA alla fine del trimestre fiscale
gli attori coinvolti nel caso d'uso	Attori primari: Tempo
lo stato del sistema prima che il caso d'uso possa iniziare	Attori secondari: Fisco
i passi effettivi del caso d'uso	Precondizioni: 1 Si è concluso un trimestre fiscale Sequenza degli eventi principale: attore implicito tempo 1. Il caso d'uso inizia quando si conclude un trimestre fiscale. 2. Il sistema calcola l'ammontare dell'IVA dovuta al Fisco. 3. Il sistema trasmette un pagamento elettronico al Fisco.
lo stato del sistema quando l'esecuzione del caso d'uso è terminata	Postcondizioni: 1 Il Fisco riceve l'importo IVA dovuto.
sequenze degli eventi alternative	Sequenza degli eventi alternativa: Nessuna.

La descrizione dei casi d'uso con pre e post-condizioni

Nome del caso d'uso:

Attori coinvolti (primari –che avviano il caso d'uso e secondari)

Descrizione breve

Pre-condizioni: *specificano lo stato del sistema prima di eseguire il caso d'uso*

Sequenza di eventi:

Percorso Normale

Post-condizioni: *specificano lo stato del sistema dopo l'esecuzione del caso d'uso*

Altri casi d'uso correlati

Sequenza di eventi:

Percorsi alternativi

Esempio di scenario

Caso d'uso effettua ordine

- ◆ Pre-condizioni: *Un utente ha effettuato il login*
- ◆ Flusso di eventi:
 - **Percorso Normale:**
 1. Il cliente sceglie di effettuare un ordine
 2. Il cliente inserisce nome ed indirizzo
 3. IF il cliente inserisce solo il CAP, il sistema visualizza la città
 4. Il cliente inserisce i codici dei prodotti da ordinare
 5. Il sistema visualizza la descrizione ed il prezzo di ciascun prodotto
 6. Il sistema calcola il totale dell'ordine man mano che i prodotti sono inseriti
 7. Il cliente inserisce i dati della carta di credito
 8. Il cliente preme il tasto di Conferma ordine
 9. Il sistema verifica le informazioni, salva l'ordine tra gli ordini pendenti, ed invia le informazioni per il pagamento al sistema dei pagamenti
 10. Quando il pagamento viene confermato, l'ordine viene contrassegnato come Confermato, il codice ordine viene inviato al cliente e il caso d'uso termina

Caso d'uso effettua ordine

◆ Percorsi alternativi:

- Al punto 9, se qualche informazione risulta scorretta, il sistema chiederà al cliente di correggere l'informazione scorretta.
- 9.a Il Codice della carta di credito risulta non valido
 - ◆ Il sistema segnala l'errore e chiede un'altra forma di pagamento
- Un percorso alternativo viene descritto da due parti (condizione e gestione)

◆ Post condizioni: *l'ordine è stato salvato nel sistema e segnato come Confermato*

Notazione con formato a due colonne

.... preambolo	
Scenario di successo	
Azione dell'Attore	Responsabilità del sistema
1. Il cliente arriva alla cassa	
2. Il cassiere inizia una nuova vendita	
3. Il cassiere inserisce il codice prodotto	
	4. Il sistema registra la vendita dell'articolo e mostra il suo prezzo
Il cassiere ripete i passi 3 e 4 finchè non indica che ha terminato	
	5. Il sistema mostra il totale con le imposte calcolate
....	

Linee guida per la descrizione degli scenari

1. Scrivere in stile *essenziale*, senza riferimento all'interfaccia utente.
 - Concentrarsi sullo scopo reale dell'utente e non su UI
2. Descrivere casi d'uso concisi e completi
 - Esempio: Il Sistema autentica il Cassiere e non: il sistema richiede username e password
3. Descrivere casi d'uso a scatola nera
 - Concentrarsi sulle responsabilità del sistema (cosa fa, non come) e non anticipare scelte realizzative
 - Esempio errato:
 - ◆ Il sistema memorizza la vendita in una base dati (o peggio: Il Sistema esegue una istruzione SQL Insert per la vendita...)

Linee guida per la descrizione degli scenari

- ❖ Nella descrizione di un caso d'uso NON devono essere indicati dettagli che rivelino le scelte di progetto del software
 - Quando si pensa ai casi d'uso, non si è ancora affrontato alcun aspetto della progettazione
 - Gli aspetti algoritmici non dovrebbero comparire: interessa sapere quale problema sarà affrontato, non la soluzione che verrà scelta (in una fase successiva)
 - Non possono esserci riferimenti a specifici file/ moduli/ specifiche interfacce utente, a meno che essi non rappresentino dei **vincoli** (ad esempio dei sistemi già esistenti con i quali bisogna integrarsi)
- ❖ Il diagramma dei casi d'uso dovrebbe SEMPRE essere tracciato e approvato PRIMA di partire con le fasi di progetto

Ulteriori aspetti del Diagramma dei Casi d'uso

- ♦ Relazioni fra attori
- ♦ Relazioni fra casi d'uso

Relazioni tra attori o tra casi d'uso

In un diagramma dei casi d'uso si possono specificare relazioni tra attori e relazioni tra casi d'uso

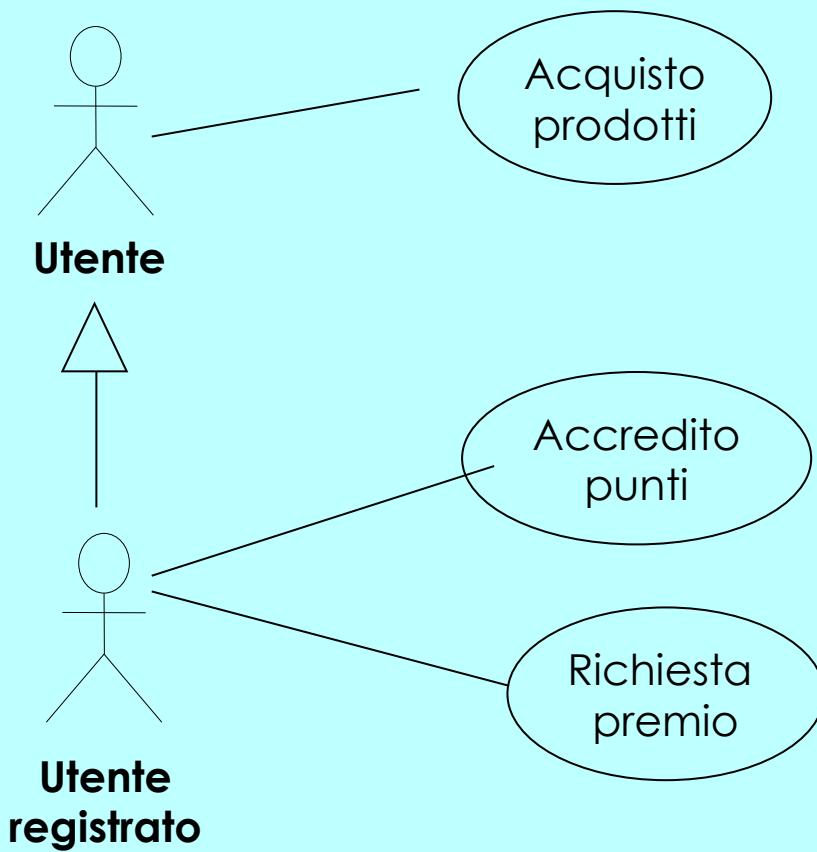
Relazioni tra attori:

- relazione **di generalizzazione tra attori**

Relazioni tra casi d'uso:

- relazione **di generalizzazione**
- relazione **d'uso <<include>>**
- relazione **d'estensione <<extends>>**

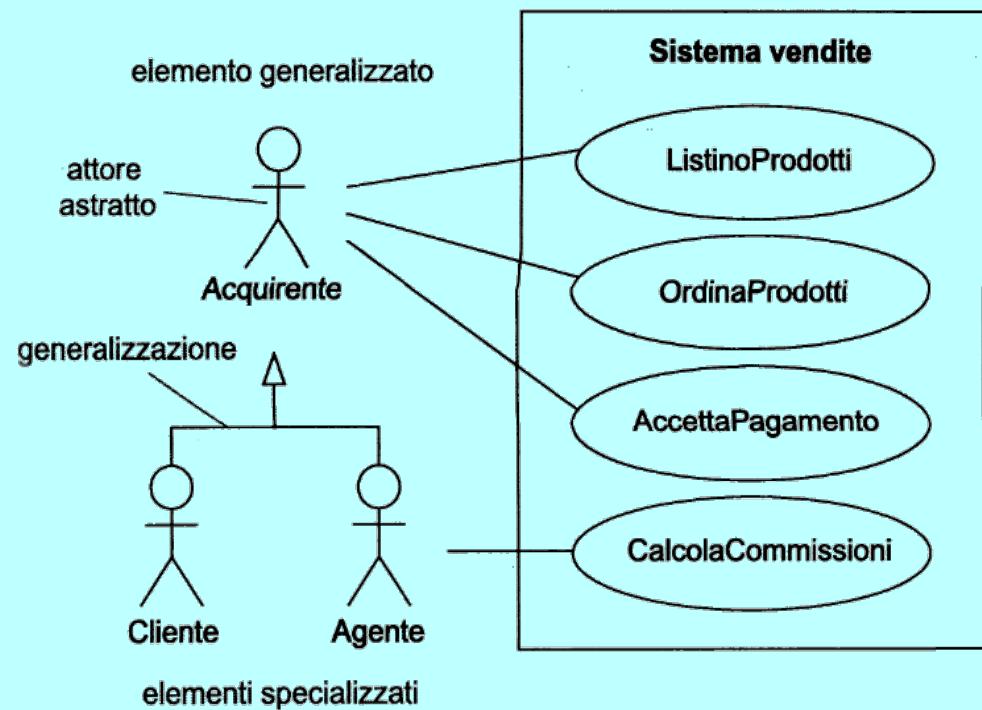
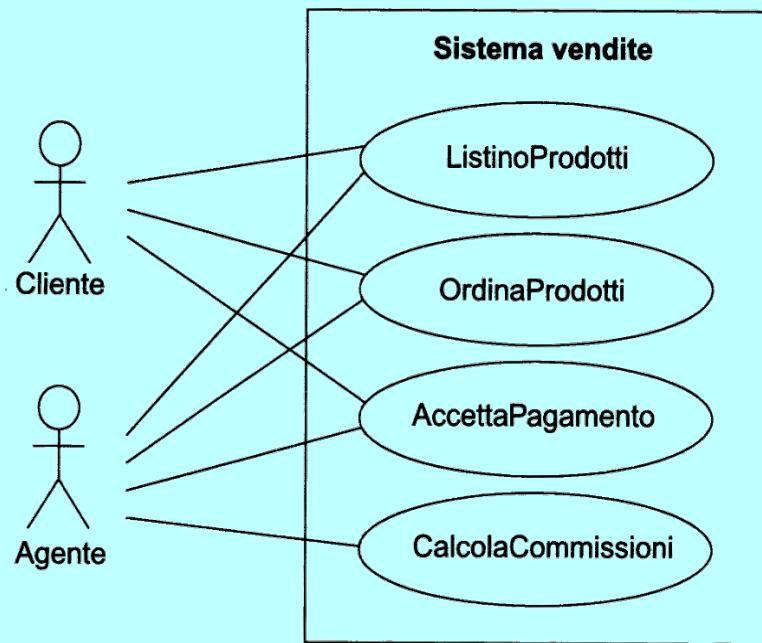
Generalizzazione di attori



L'attore figlio conserva le proprietà del padre oltre a possedere sue caratteristiche particolari.

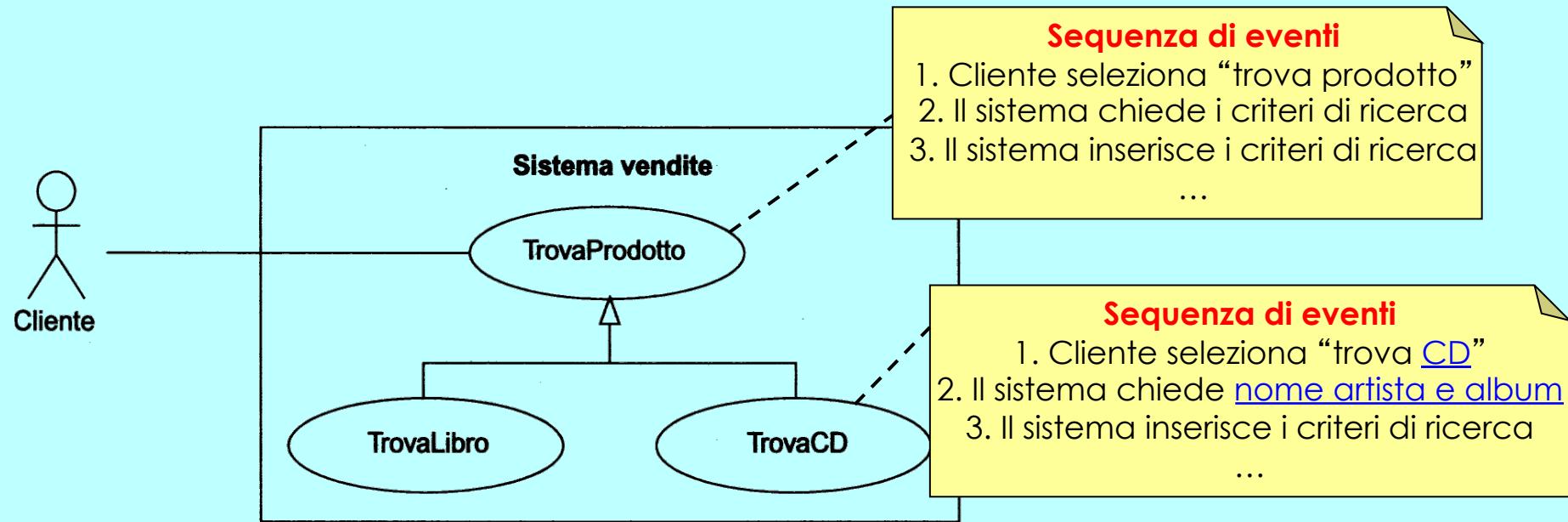
Generalizzazione fra attori

La **generalizzazione tra attori** permette di astrarre ruoli comuni a più attori e di semplificare i diagrammi. Gli attori specializzati (o figli) ereditano i ruoli e le relazioni dell'attore generalizzato.



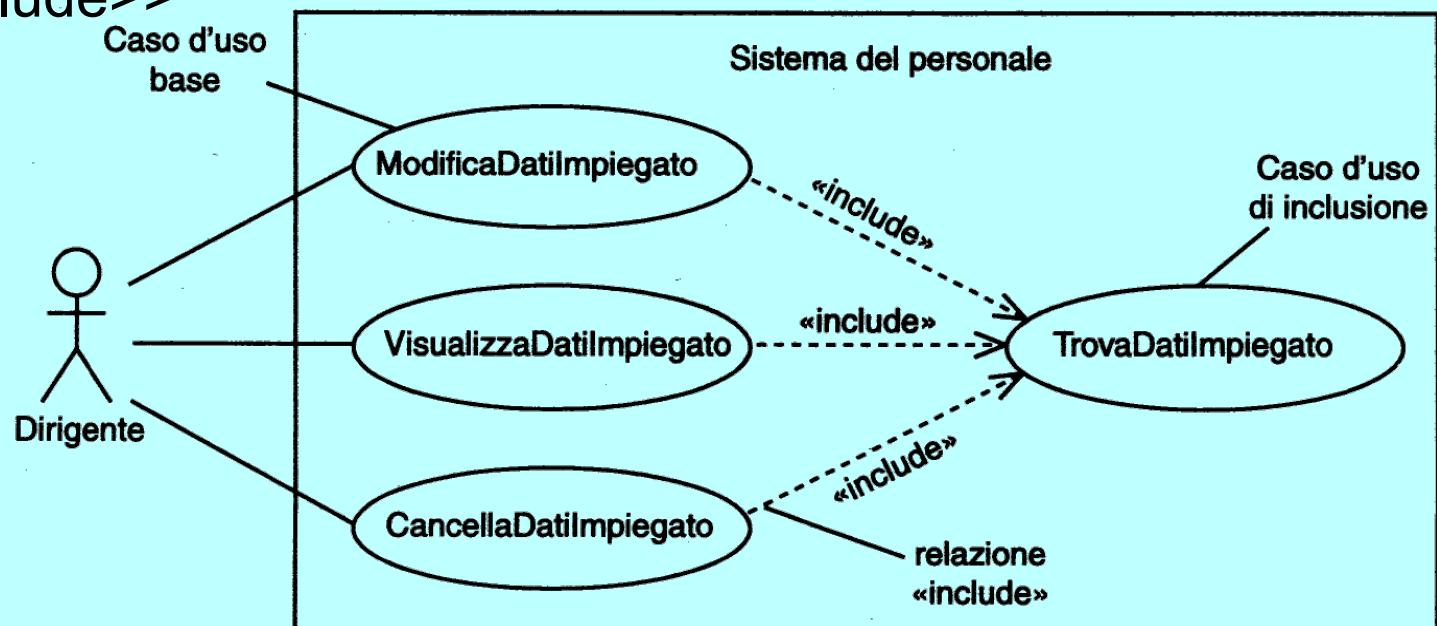
Generalizzazione fra casi d'uso

- ◆ Simile alla generalizzazione fra attori
- ◆ Un **caso d'uso figlio** eredita comportamento e significato del **caso d'uso padre**
- ◆ Il caso d'uso figlio può aggiungere o modificare il comportamento del padre (es. Caso d'uso TrovaProdotto)



Relazione di inclusione tra casi d'uso

- La **relazione d'uso (inclusione)** formalizza i casi in cui più casi d'uso racchiudono una serie di azioni comuni.
Il **comportamento comune a più casi d'uso** diventa un caso d'uso che è incluso nei casi d'uso di partenza.
- Il caso d'uso base è incompleto senza il caso incluso.**
- Rappresentato graficamente come una dipendenza stereotipata come <<include>>



Punto di inclusione negli scenari

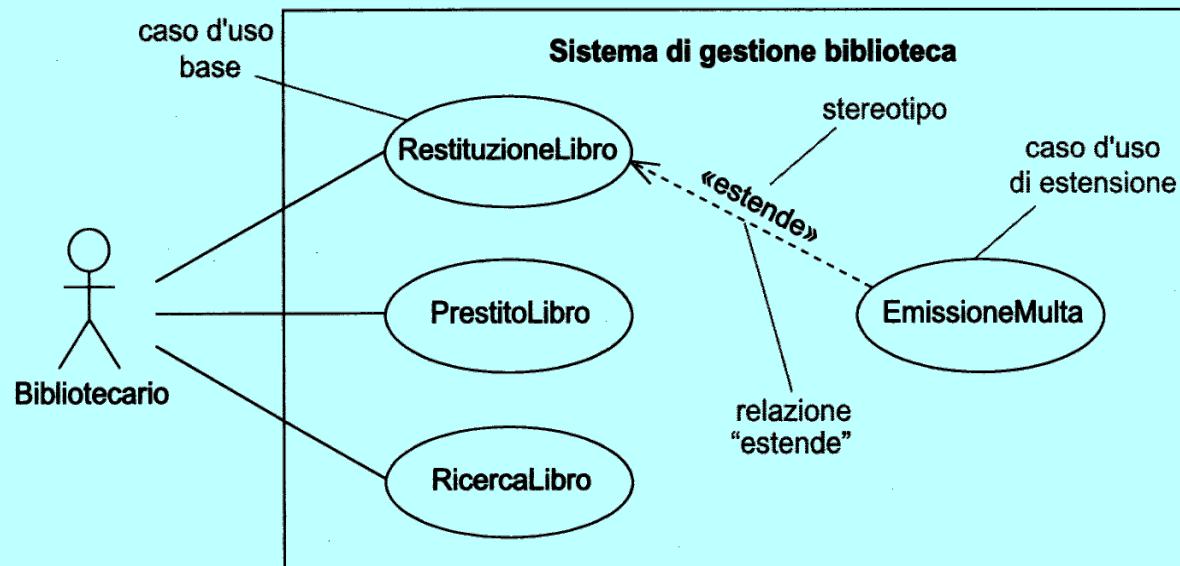
Caso d'uso: ModificaDatilmpiegato	
ID:1	
Breve descrizione: Il Dirigente modifica i dati dell'impiegato.	
Attori primari: Dirigente	
Attori secondari: Nessuno.	Inclusione Ha un significato simile ad una chiamata di procedura
Precondizioni: 1. Il Dirigente è stato autenticato dal sistema.	
Sequenza degli eventi principale: 1. include (TrovaDatilmpiegato) 2. Il sistema mostra i dati dell'impiegato 3. Il Dirigente modifica i dati dell'impiegato ...	
Postcondizioni: 1. I dati dell'impiegato sono stati modificati.	
Sequenze degli eventi alternative: Nessuna.	

Caso d'uso: VisualizzaDatilmpiegato
ID:2
Breve descrizione: Il Dirigente visualizza i dati dell'impiegato.
Attori primari: Dirigente.
Attori secondari: Nessuno.
Precondizioni: 1. Il Dirigente è stato autenticato dal sistema.
Sequenza degli eventi principale: 1. include (TrovaDatilmpiegato). 2. Il sistema mostra i dati dell'impiegato. ...
Postcondizioni: 1. Il sistema ha mostrato i dati dell'impiegato.
Sequenze degli eventi alternative: Nessuna.

Caso d'uso: CancellaDatilmpiegato
ID:3
Breve descrizione: Il Dirigente cancella i dati dell'impiegato.
Attori primari: Dirigente.
Attori secondari: Nessuno.
Precondizioni: 1. Il Dirigente è stato autenticato dal sistema.
Sequenza degli eventi principale: 1. include (TrovaDatilmpiegato). 2. Il sistema mostra i dati dell'impiegato. 3. Il Dirigente cancella i dati dell'impiegato. ...
Postcondizioni: 1. I dati dell'impiegato sono stati cancellati.
Sequenze degli eventi alternative: Nessuna.

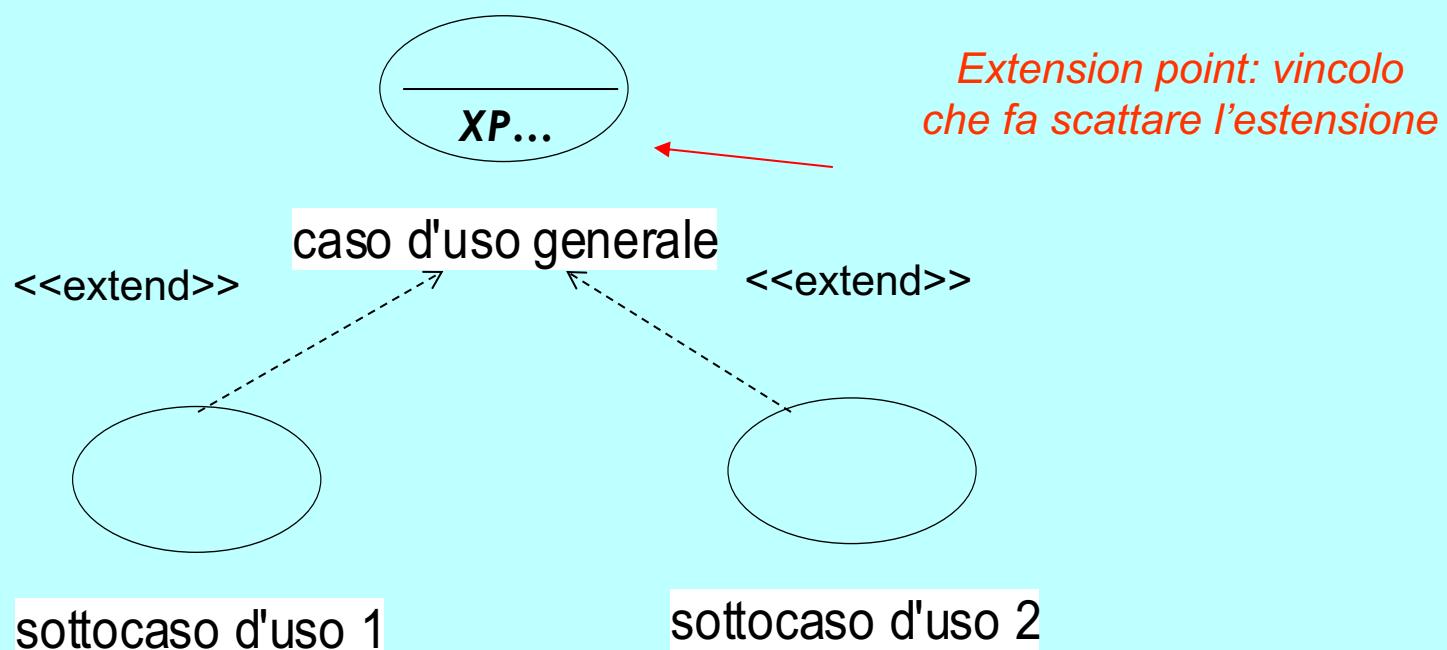
Relazione di estensione tra casi d'uso

- ◆ Si usa per modellare una sequenza opzionale di eventi oppure casi eccezionali.
- ◆ La sequenza opzionale definisce un nuovo caso d'uso che estende il caso d'uso di partenza e ne varia il comportamento “normale”.
 - Nel caso d'uso esteso (base) si **aggancia** ad uno o più **punti d'estensione**
 - I casi d'uso di estensione aggiungono un comportamento in corrispondenza di questi punti
- ◆ Il caso d'uso base si può svolgere anche senza i casi d'uso d'estensione



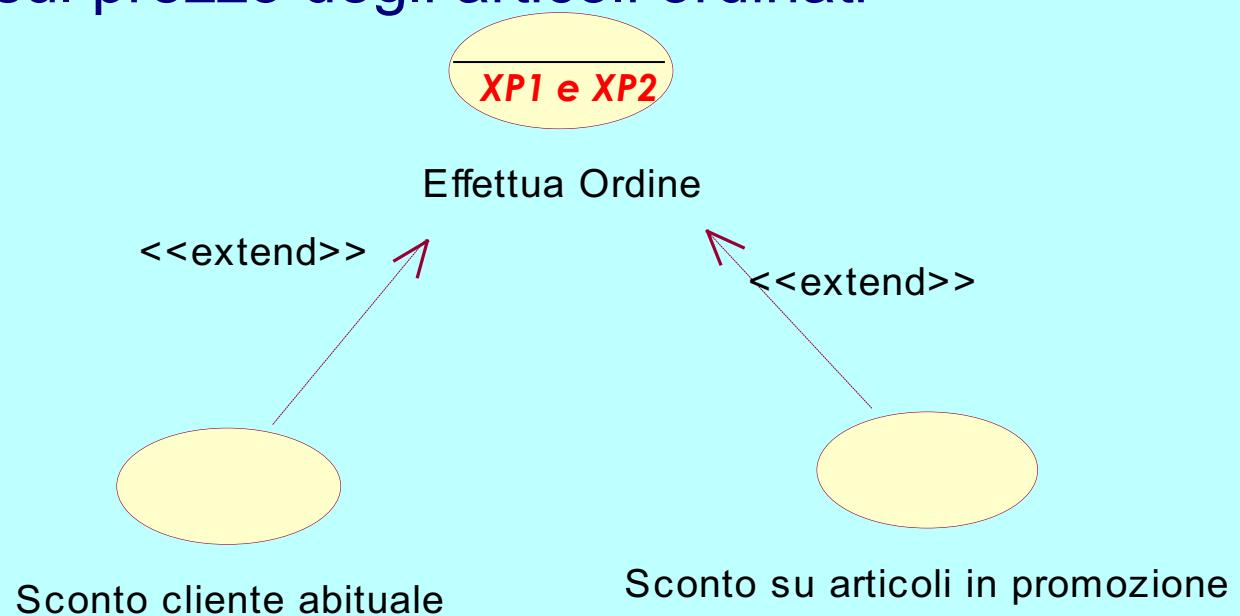
Relazione di estensione

- ◆ Creando estensioni separate per le sequenze opzionali o i casi eccezionali, la descrizione del caso d'uso di base resterà semplice
- ◆ Si rappresenta con dipendenze stereotipate con <<extend>>
 - Partono dal caso d'uso che estende e arrivano al caso d'uso base



Esempio di casi d'uso d'estensione

- Il caso d'uso ‘**Effettua ordine**’ può presentare comportamento aggiuntivo in due casi:
 - Quando il cliente sceglie prodotti in promozione, il prezzo dell’articolo sarà opportunamente scontato
 - Quando il cliente è un cliente abituale, gli verrà praticato uno sconto sul prezzo degli articoli ordinati

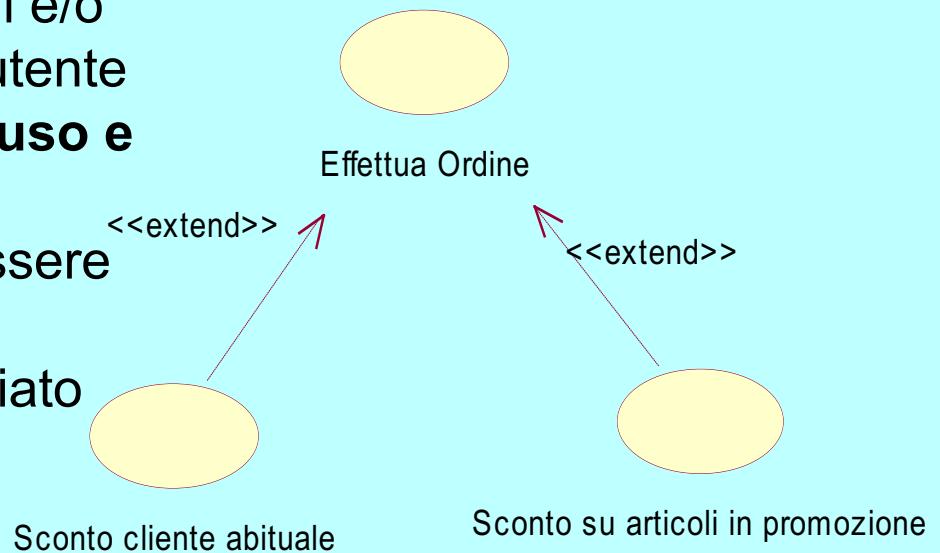


Proprietà dell'estensione

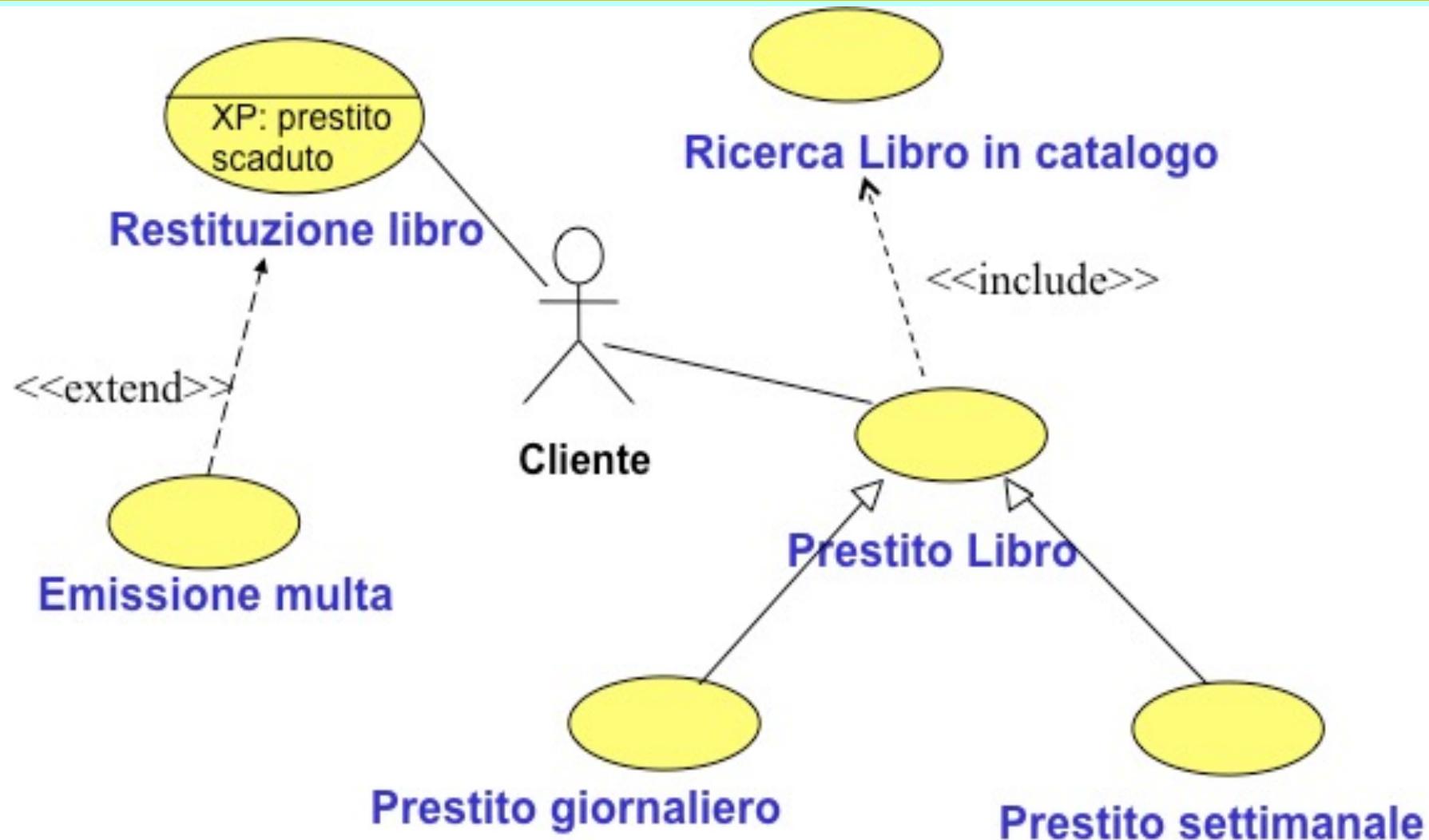
- ◆ L'estensione NON contiene informazioni sull'ordine con il quale i due casi d'uso sono effettuati
- ◆ I casi d'uso che estendono possono anche essere accessibili direttamente da un attore
 - In tal caso nel diagramma dei casi d'uso ci sarà una comunicazione (un segmento) tra l'attore e il caso d'uso esteso

Estensione vs Scenari Alternativi

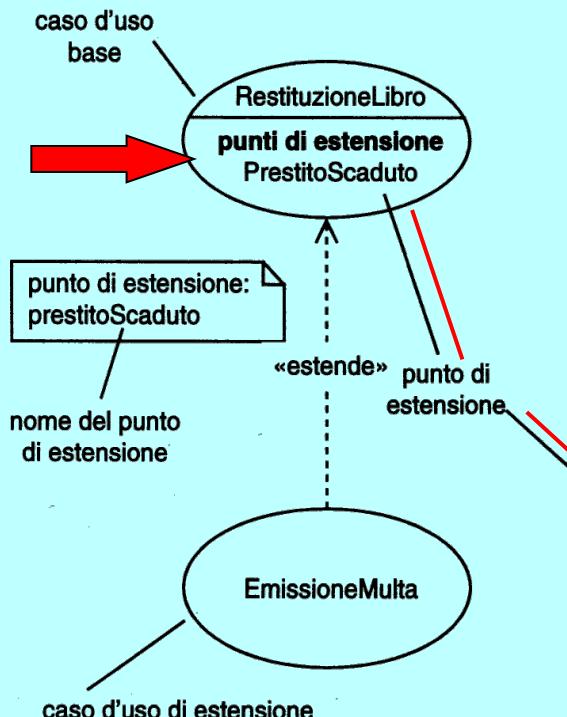
- ♦ L'esempio visto si poteva anche risolvere utilizzando un unico caso d'uso (Effettua Ordine) con due scenari alternativi (corrispondenti ai due casi d'uso d'estensione)
- ♦ Come scegliere tra le due possibili soluzioni modellistiche?
 - La soluzione **con tre casi d'uso** è più utile nel caso in cui i casi d'uso d'estensione abbiano ulteriori legami e/o siano direttamente richiamabili dall'utente
 - La soluzione **con un solo caso d'uso e più scenari** fornisce una vista più compatta del sistema, e potrebbe essere preferibile se si vuole realizzare un modello dei casi d'uso meno dettagliato



Esempio



Punto di estensione negli scenari



Caso d'uso: RestituzioneLibro
ID: 9
Breve descrizione: Il Bibliotecario restituisce un libro preso in prestito.
Attori primari: Bibliotecario.
Attori secondari: Nessuno.
Precondizioni: 1. Il Bibliotecario è stato autenticato dal sistema.
Sequenza degli eventi principale: 1. Il Bibliotecario inserisce l'ID dell'utente che restituisce il libro. 2. Il sistema mostra i dettagli dell'utente che restituisce il libro, incluso un elenco di tutti i libri ancora da restituire. 3. Il Bibliotecario individua il libro restituito fra quelli in prestito. punto di estensione: prestito scaduto 4. Il Bibliotecario rimette il libro a posto ...
Postcondizioni 1. Il libro è stato restituito.
Sequenze degli eventi alternative: Nessuna.

Caso d'uso di estensione: EmissioneMultta
ID: 10
Breve descrizione: Segmento 1: il Bibliotecario registra e stampa una multa.
Attori primari: Bibliotecario.
Attori secondari: Nessuno.
Precondizioni del segmento 1: 1. Il prestito è scaduto.
Sequenza degli eventi del segmento 1: 1. Il Bibliotecario inserisce i dati della multa nel sistema. 2. Il sistema stampa la multa.
Postcondizioni del segmento 1: 1. La multa è stata registrata nel sistema. 2. Il sistema ha stampato la multa.

Estensione

Il caso d'uso base si comporta come un **framework** modulare in cui è possibile inserire estensioni in punti di estensione predefiniti

Uso di If, While e For negli scenari

Le alternative possono essere descritte attraverso **If** oppure attraverso percorsi alternativi

Cicli **While** o **For** possono essere usati per racchiudere gruppi di passi che devono essere ripetuti più volte

Ad esempio il passo 4 dello scenario precedente:

4. Per (**For**) ciascun codice prodotto inserito
 - a) il sistema fornirà la descrizione ed il prezzo
 - b) il sistema aggiungerà il prezzo al totale

End

Uso di If, While e For negli scenari

Caso d'uso: TrovaProdotto
ID:3
Breve descrizione: Il sistema individua alcuni prodotti in base ai criteri di ricerca specificati dal Cliente e li mostra al Cliente
Attori primari: Cliente
Attori secondari: Nessuno
Precondizioni: Nessuna
Sequenza degli eventi principale 1. Il caso d'uso inizia quando il Cliente seleziona "Trova prodotto". 2. Il sistema chiede al Cliente i criteri di ricerca. 3. Il Cliente inserisce i criteri di ricerca richiesti. 4. Il sistema ricerca i prodotti che soddisfano i criteri specificati dal Cliente. 5. Se il sistema trova uno o più prodotti. 5.1. Per ogni prodotto trovato 5.1.1. Il sistema mostra l'immagine ridotta del prodotto. 5.1.2. Il sistema mostra l'elenco delle caratteristiche del prodotto. 5.1.3. Il sistema mostra il prezzo del prodotto. 6. Altrimenti 6.1 Il sistema comunica al cliente che non sono stati trovati prodotti che soddisfano i criteri specificati.
Postcondizioni: Nessuna
Sequenze degli eventi alternative: Nessuna

Relazione fra requisiti e casi d'Uso

◆ Requisito funzionale:

- funzionalità, o aspetto di dettaglio di una funzionalità, richiesta dal committente

◆ Caso d'Uso:

- modalità di utilizzo del sistema da parte di un utilizzatore (attore)

◆ Ogni caso d'uso può soddisfare più requisiti funzionali

◆ Un requisito funzionale può dare origine a più casi d'uso

◆ Ad ogni caso d'uso possono venire associati più requisiti non funzionali

◆ I requisiti funzionali sono pensati dal punto di vista del sistema, i casi d'uso da quello dell'attore

Tracciabilità requisiti-casi d'uso

- ◆ È importante incrociare gli archivi dei requisiti funzionali e dei casi d'uso per verificare la reciproca copertura:
 - Ossia che ogni requisito sia coperto da almeno un caso d'uso e viceversa (in generale la relazione è molti-a-molti)

		Caso d'uso				Matrice di Tracciabilità
		UC1	UC2	UC3	UC4	
Requisito	R1	X				
	R2		X	X		
	R3			X		
	R4		X		X	
	R5					