Use case

Complementi di UML

Cos’è UML?

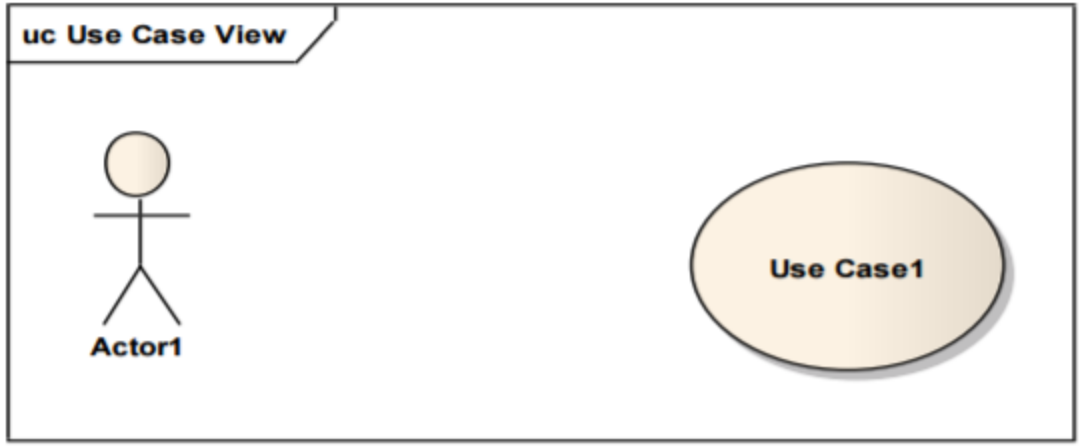
Il Linguaggio di Modellazione Unificato (UML) è un linguaggio nato per specificare, visualizzare e documentare modelli di sistemi di software a oggetti.

UML non è un metodo di sviluppo, cioè non dice cosa fare prima e dopo o come progettare un sistema, ma aiuta a visualizzare un progetto e, soprattutto, a comunicarne le proprietà.

UML è controllato da Object Management Group (OMG) ed è lo standard industriale per descrivere graficamente il software.

Attore - Actor

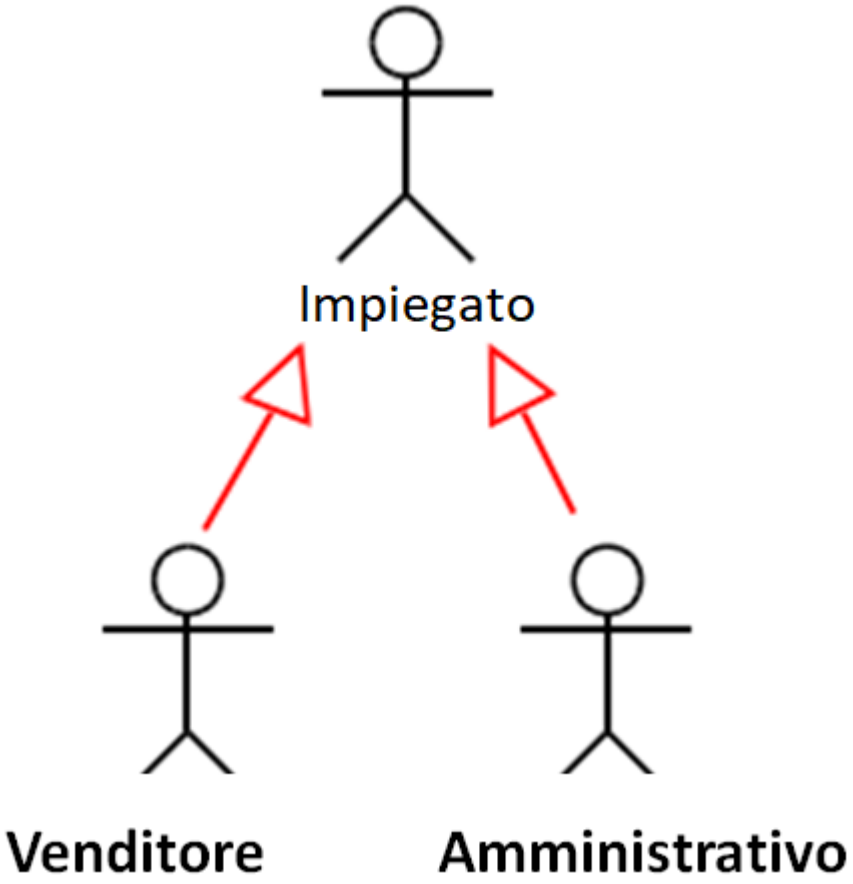
- Un attore è qualsiasi persona o sistema esterno che deve interagire con il sistema in esame
- Gli attori NON sono parte del sistema da sviluppare
- Gli attori possono essere ATTIVI o PASSIVI. Se attivi, avviano uno *Use Case*, se passivi ne ricevono solo informazioni.
- In UML, uno Use Case è graficamente rappresentato da un ovale, mentre un attore viene rappresentato da un omino stilizzato



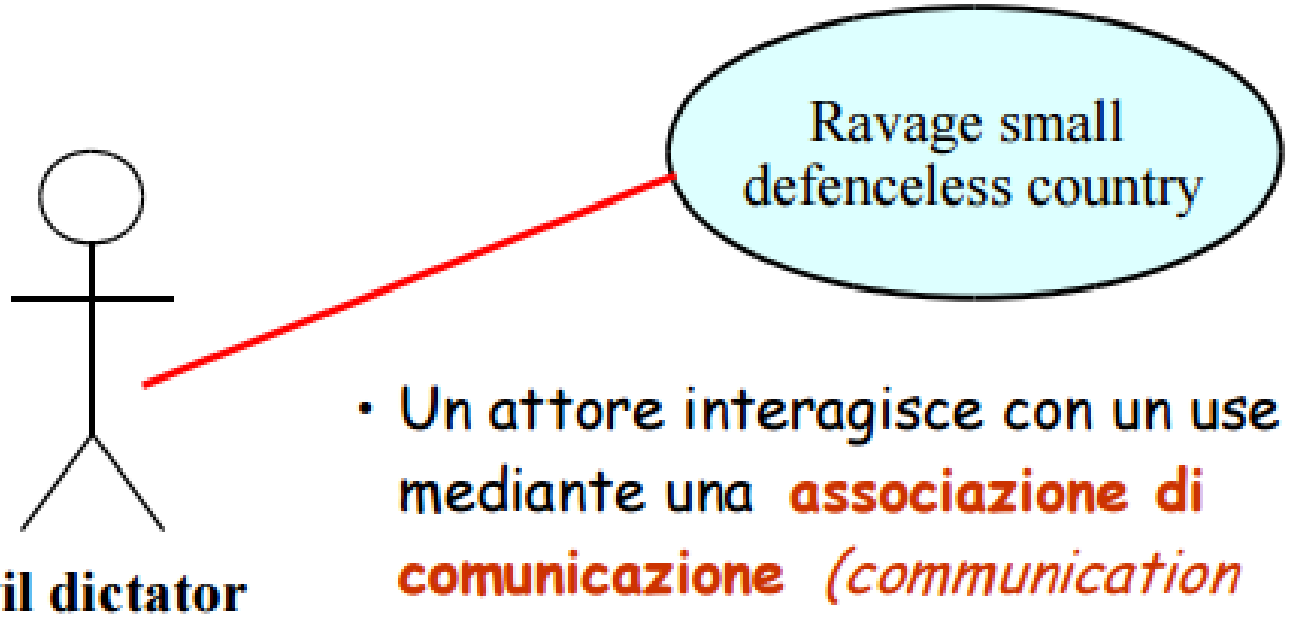
Associazioni tra casi d’uso e attori

- Una sola relazione è presente: l’associazione
- Un’associazione tra un attore e uno Use Case indica
  - che l’attore e lo use case comunicano tra loro, che ognuno può inviare e ricevere messaggi
  - Indica che un attore, svolgendo un particolare ruolo, interagisce con il sistema
- L’interazione è descritta dal caso d’uso associato (c’è una parte testuale)
- Non indica un flusso di dati, insomma non è un data-flow diagram!

Generalizzazione tra attori



Relazione tra attore e use case: **associazione**



- Un attore interagisce con un use case mediante una **associazione di comunicazione** *(communication association).*
- Questa è **normalmente** una relazione uno a uno, priva di direzione.

Relazioni tra casi d’uso

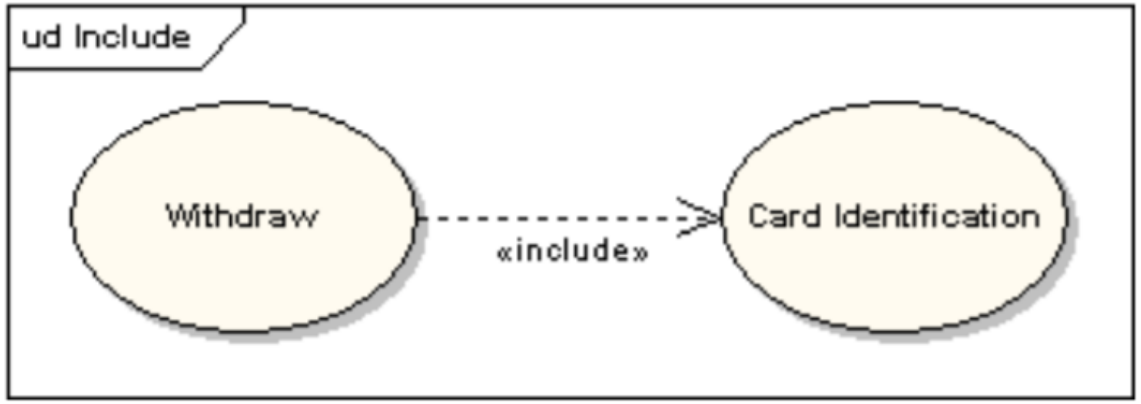
Ci sono poche relazioni: è un modello semplice, senza grossa complessità sintattica

- Relazione di inclusione «include»
- Relazione di generalizzazione
- Relazione di estensione «extend»

Cerchiamo di capire bene però queste poche relazioni

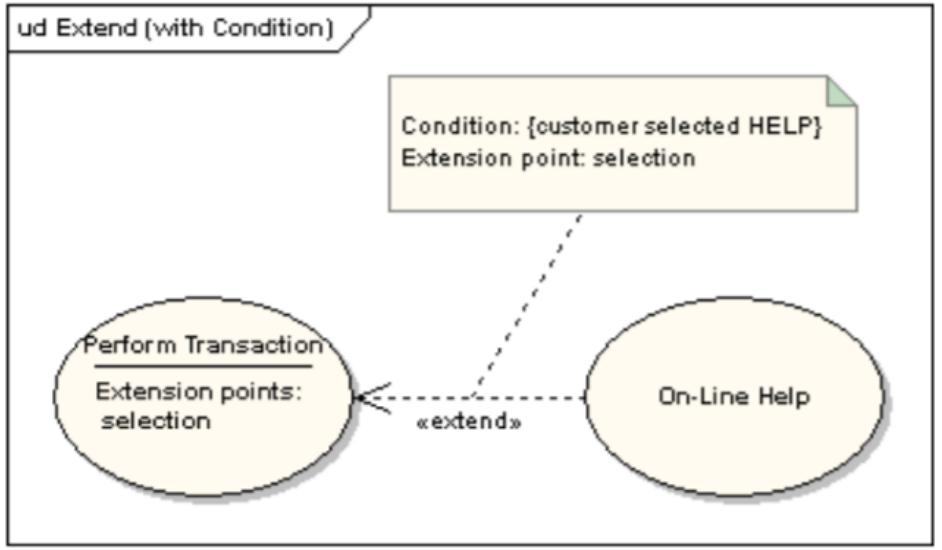
Relazioni di dipendenza tra casi d’uso

- Inclusione «include»
- Indica che il caso d’uso principale incorpora esplicitamente il comportamento di un altro caso d’uso subordinato
- Il caso d’uso principale indica l’esatto punto in cui il caso d’uso subordinato viene incluso
- Al termine dell’esecuzione del caso d’uso subordinato, il caso d’uso principale riprende dal punto in cui è stato sospeso



Relazioni di dipendenza tra casi d'uso - Estensione

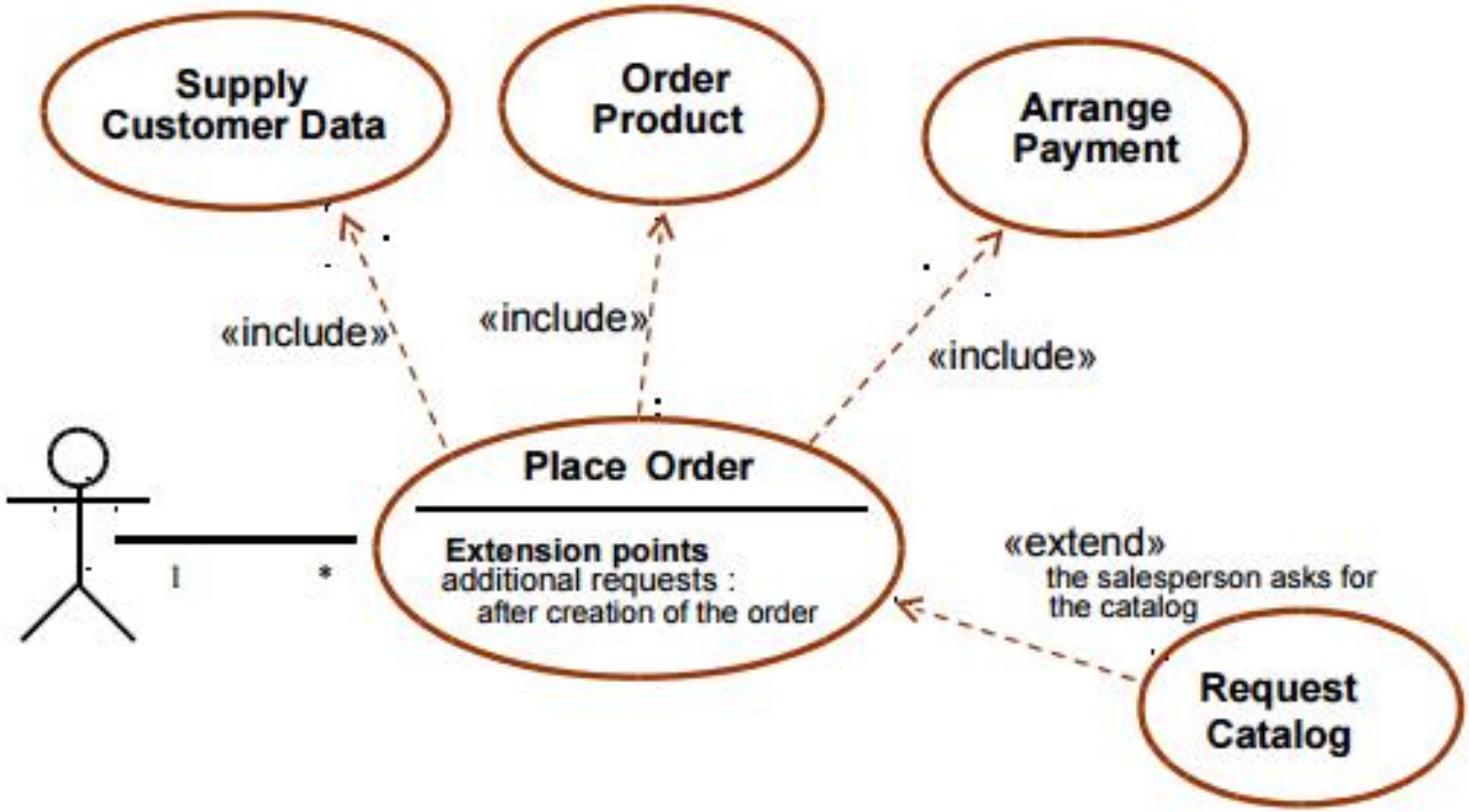
- Estensione «extend»
- Indica che il caso d'uso subordinato estende il comportamento del caso d'uso principale, aggiungendo la logica necessaria per gestire eccezioni, flussi di lavoro alternativi, ecc.
- Il caso d'uso principale indica l'esatto punto in cui il caso d'uso subordinato viene incluso (detto "punto di estensione")
- Al termine dell'esecuzione del caso d'uso subordinato, il caso d'uso principale riprende dal punto in cui è stato sospeso
- Occorrono (se non banali):
  - Punti di estensione
  - Condizioni di attivazione



Relazioni di dipendenza tra casi d’uso - Estensione

- Da non confondere con l’ereditarietà (!)
- Il caso d’uso «estensione» continua il comportamento del caso d’uso di base, inserendovi delle azioni
- Il caso d’uso di base dichiara tutti i possibili punti di estensione, anche a livello grafico, generalmente con una nota o con una descrizione testuale all’interno dell’ovale del caso d’uso base
- Simile alla gestione degli interrupt hardware (gestione eccezioni)
- È una *vera* estensione solo se serve a completare il (una parte del) caso d’uso principale
- L’estensione *vive* all’interno del caso d’uso principale; in altre parole, l’esecuzione dell’estensione è ancora parte dell’esecuzione del caso d’uso principale

Relazione tra use case



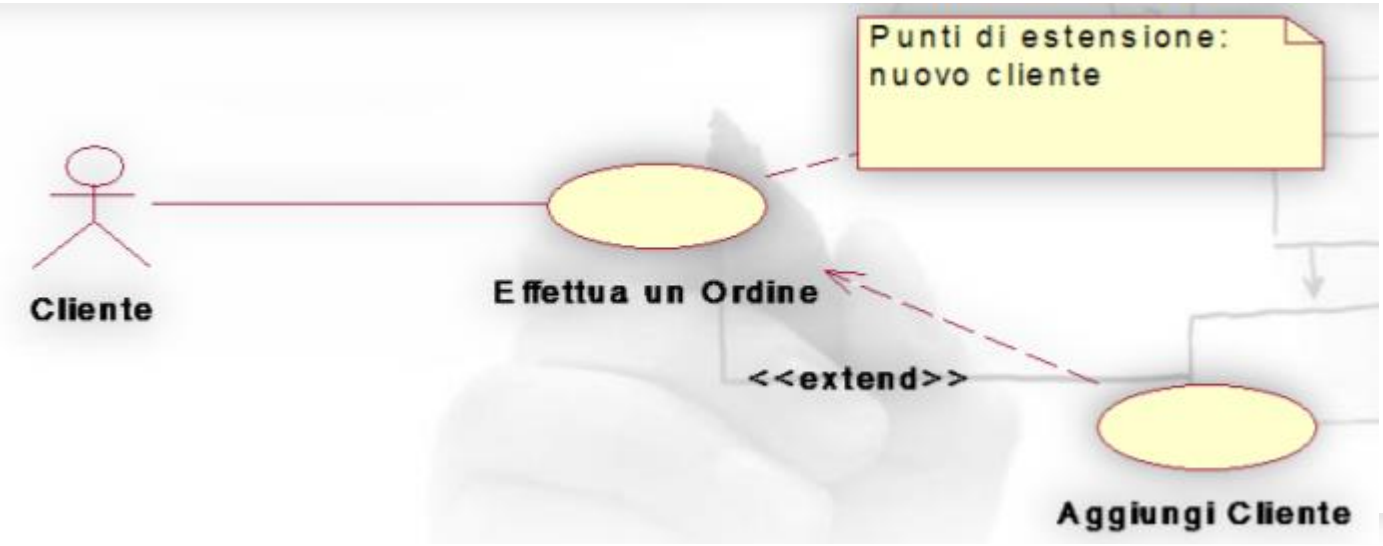
Relazioni tra casi d’uso - riepilogo

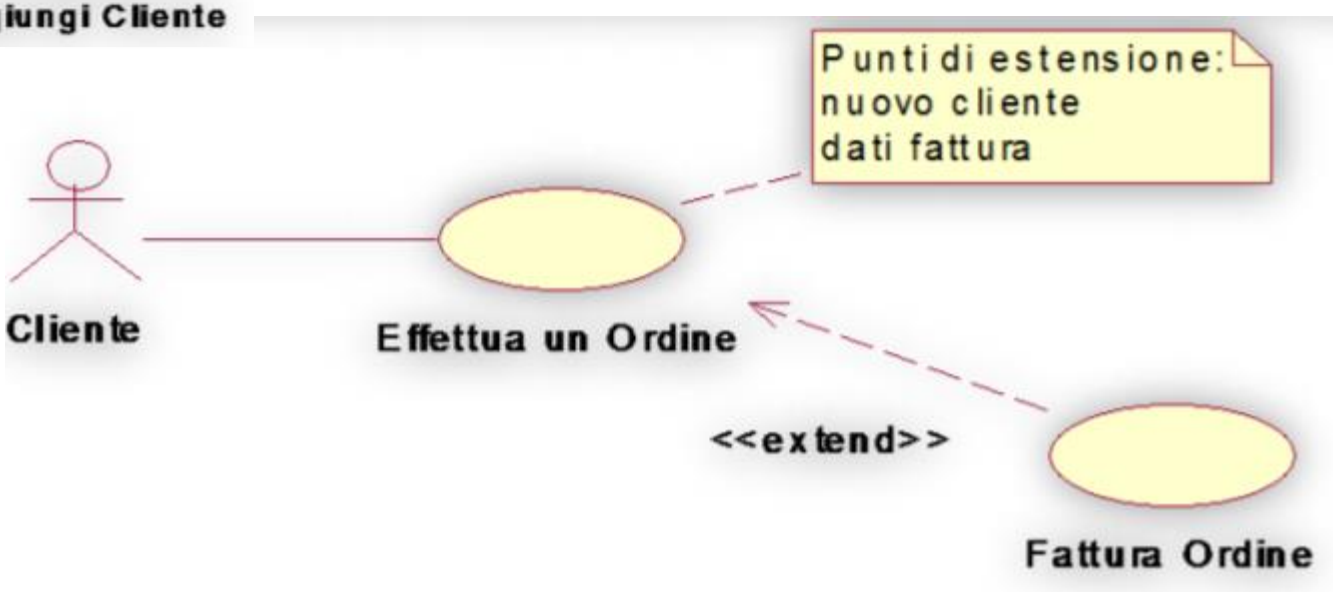
- **extend** - a dashed arrow indicating an addition to functionality of the base case.

- **include** - a dashed arrow indicating a calling relationship like a function call.

- **generalization** - a hollow arrowhead indicating inheritance.

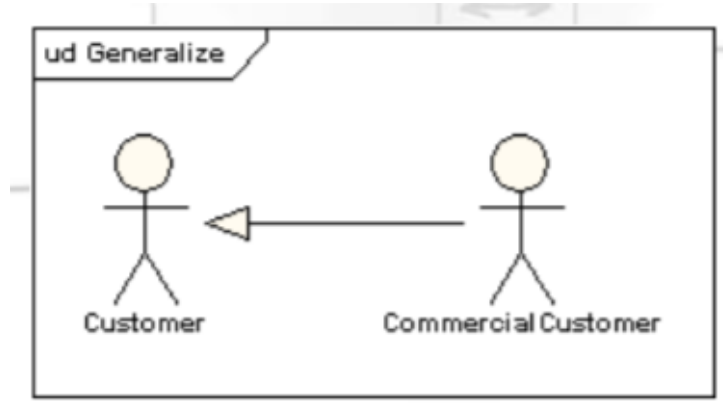
Esempi di estensione

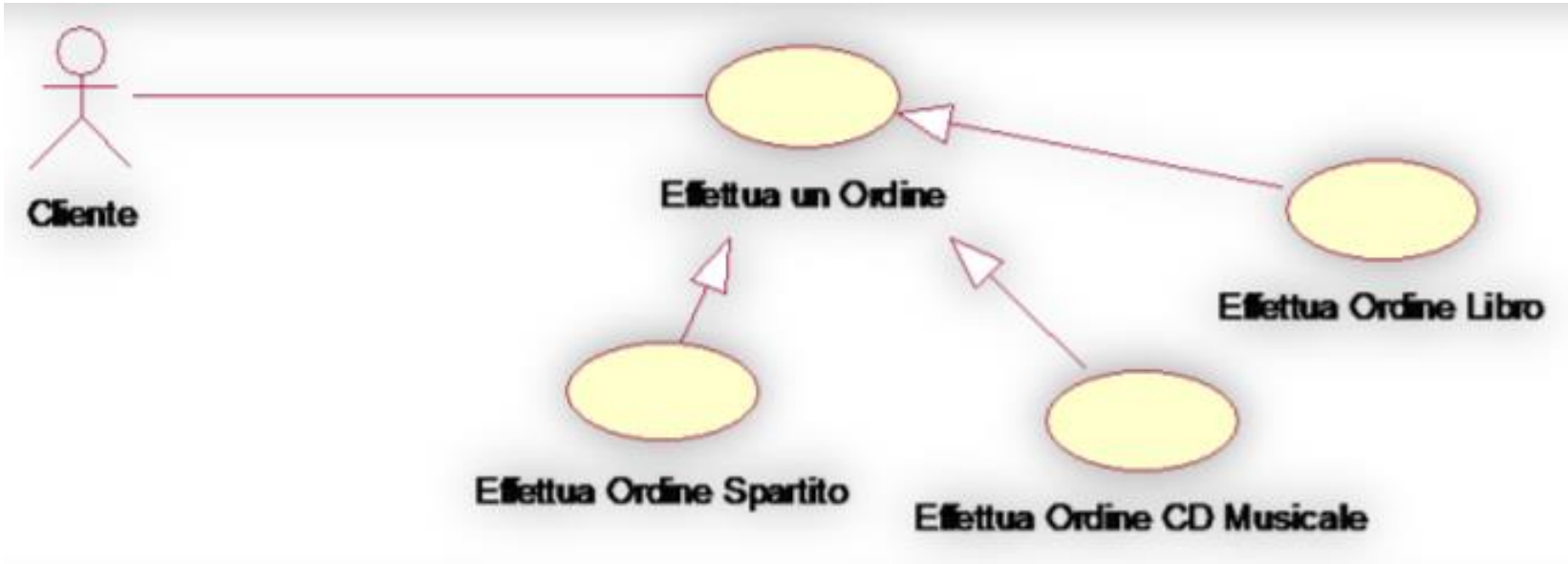




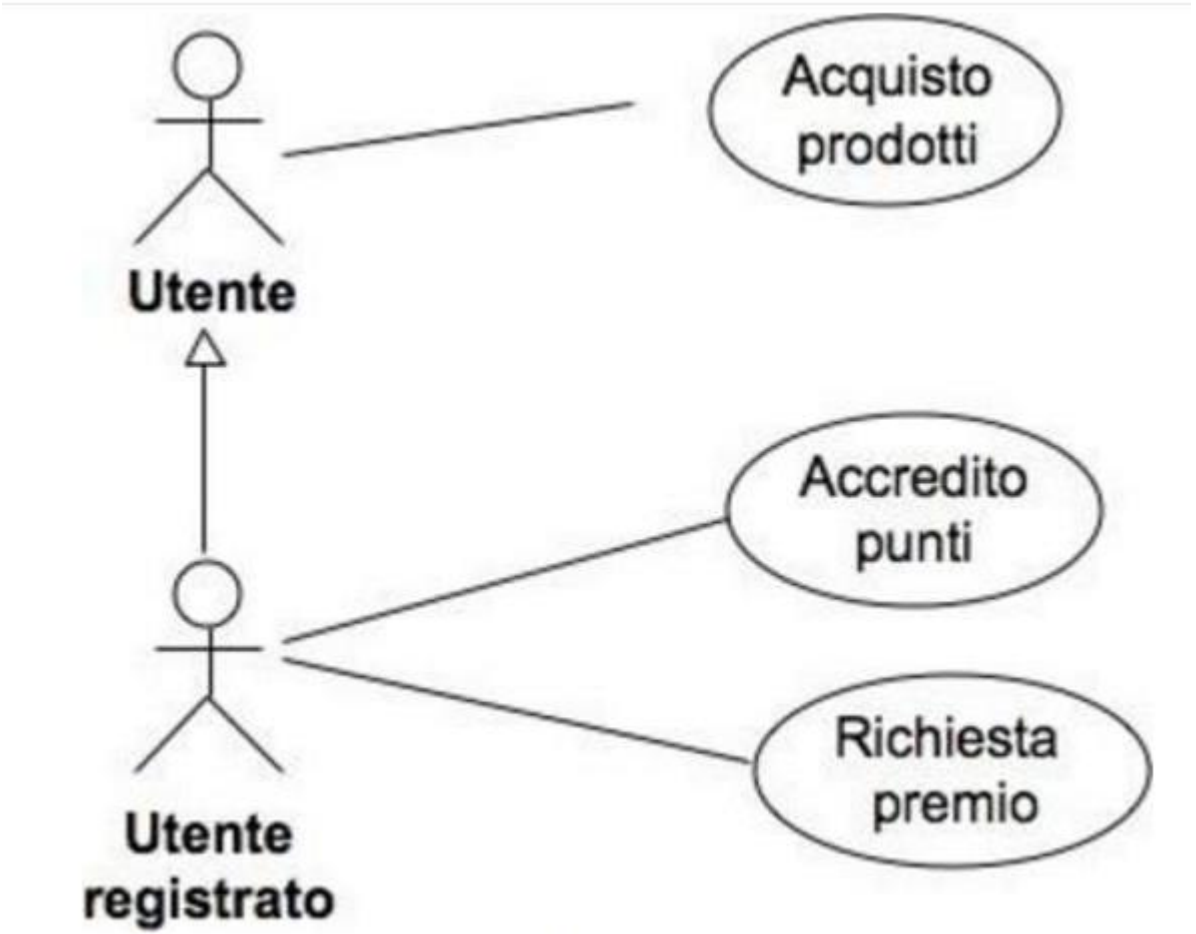
Relazioni di dipendenza tra casi d’uso

- Generalizzazione
  - Specifica gerarchie (categorie) di attori e/o casi d’uso
  - concettualmente, è la classica relazione di ereditarietà





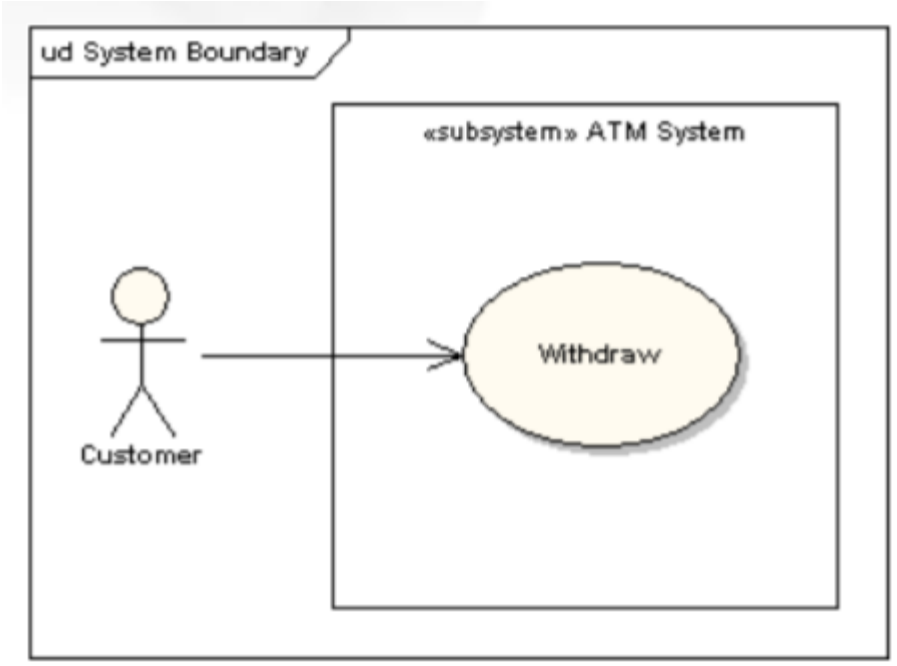
Generalizzazione tra attori - esempio



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

System boundary

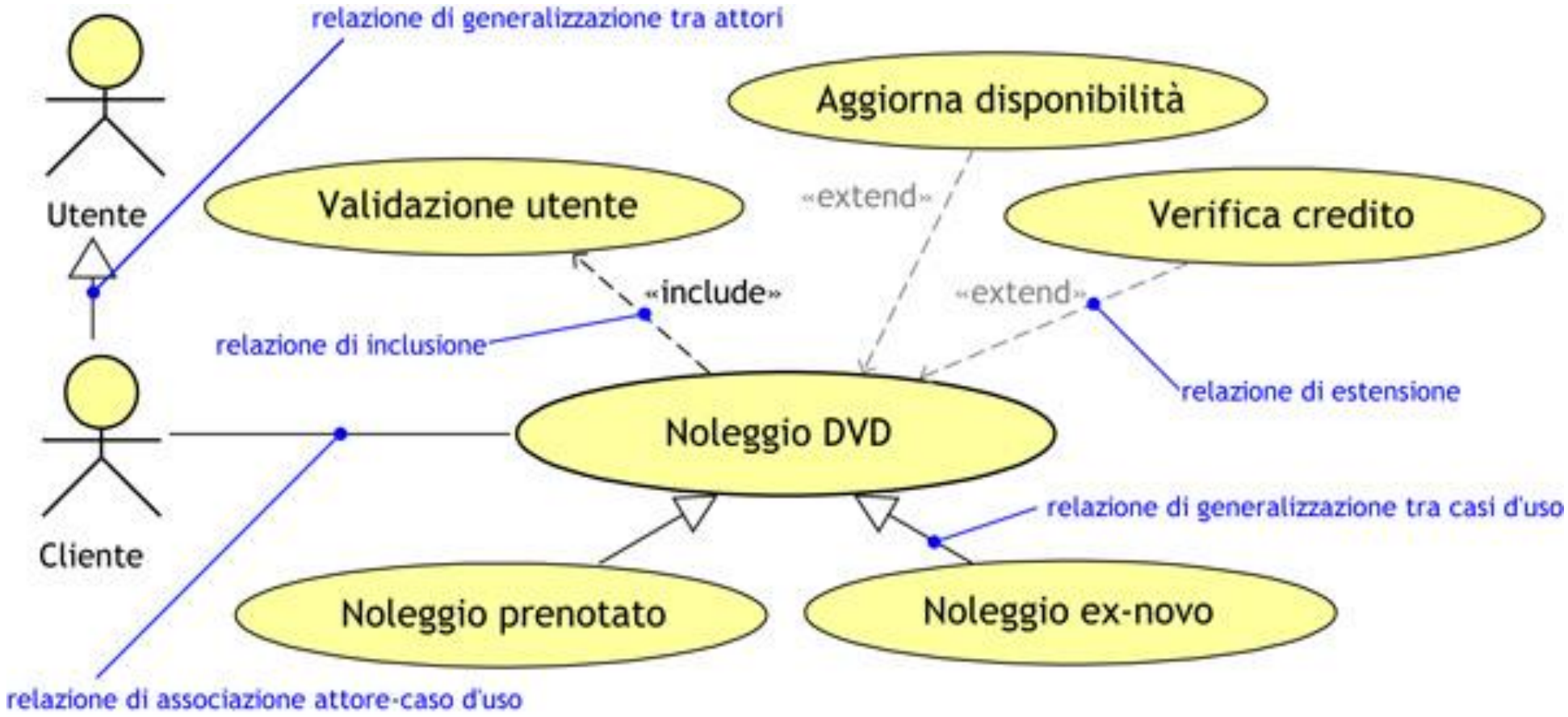
- E’ buona pratica visualizzare gli use case come se fossero racchiusi dentro ad una scatola (system boundary) che descrive il confine tra il sistema da sviluppare e il resto del mondo
- Gli attori che interagiscono con il sistema vengono rappresentati all’esterno della scatola



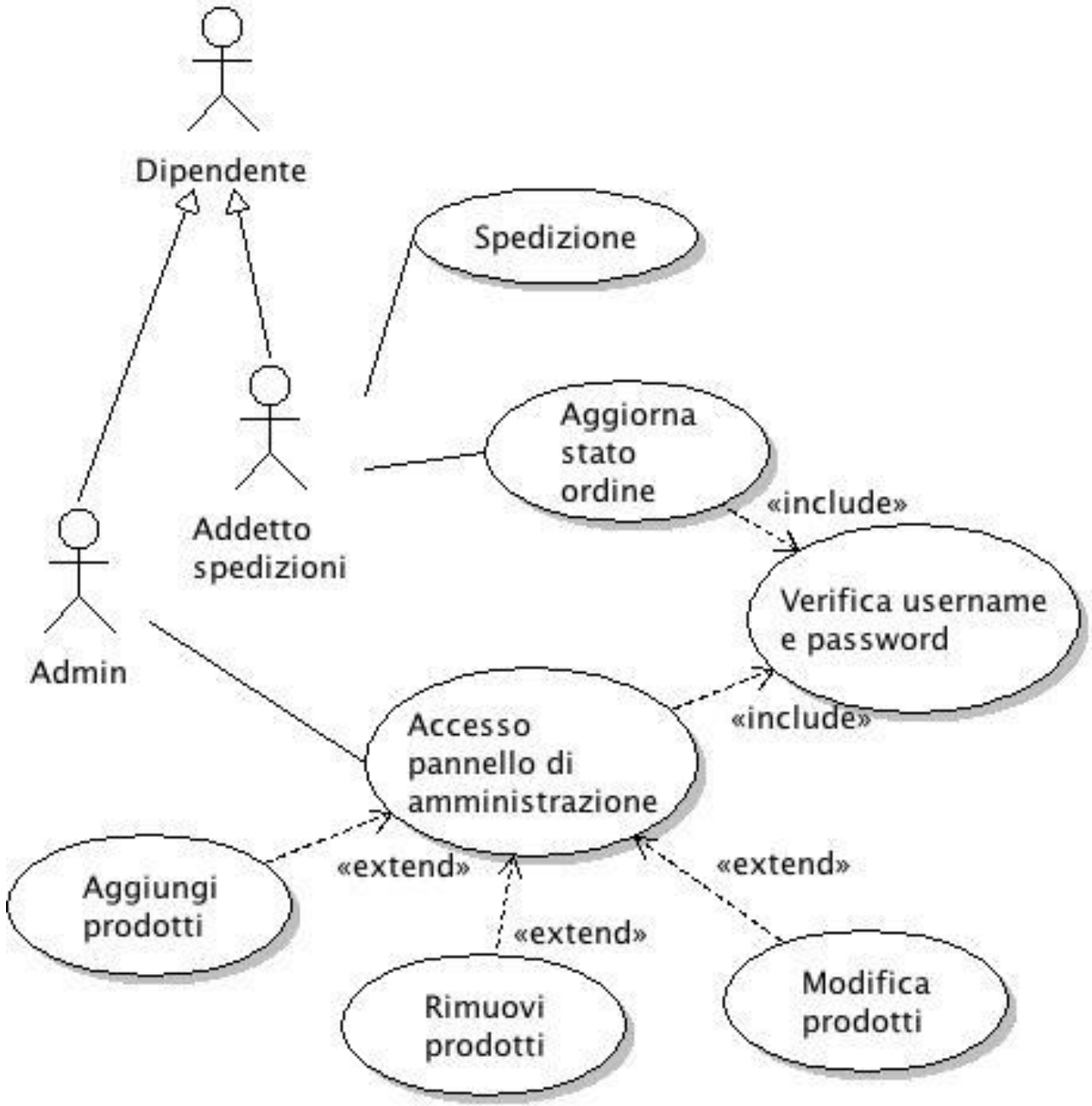
Riassumendo …

- Perché è importante modellare i casi d'uso?
- E' il primo momento nel ciclo di vita del software in cui costruiamo delle rappresentazioni del software
- Aiuta a esplicitare e comunicare a tutti gli stakeholder i requisiti funzionali del sistema (analizzare, identificare, descrivere gli usi tipici del sistema da parte dei suoi utilizzatori)
- Permette di considerare anche tutta l'eventuale logica derivante dalla gestione di errori, eccezioni, flussi alternativi
- Autorizza a validare i requisiti utente (essendo il modello dei casi d'uso piuttosto semplice, con pochi costrutti, e semantica precisa ma intuitiva, diventa un valido strumento per assicurarci di aver sviluppato un sistema software corrispondente alle vere necessità dell'utente).

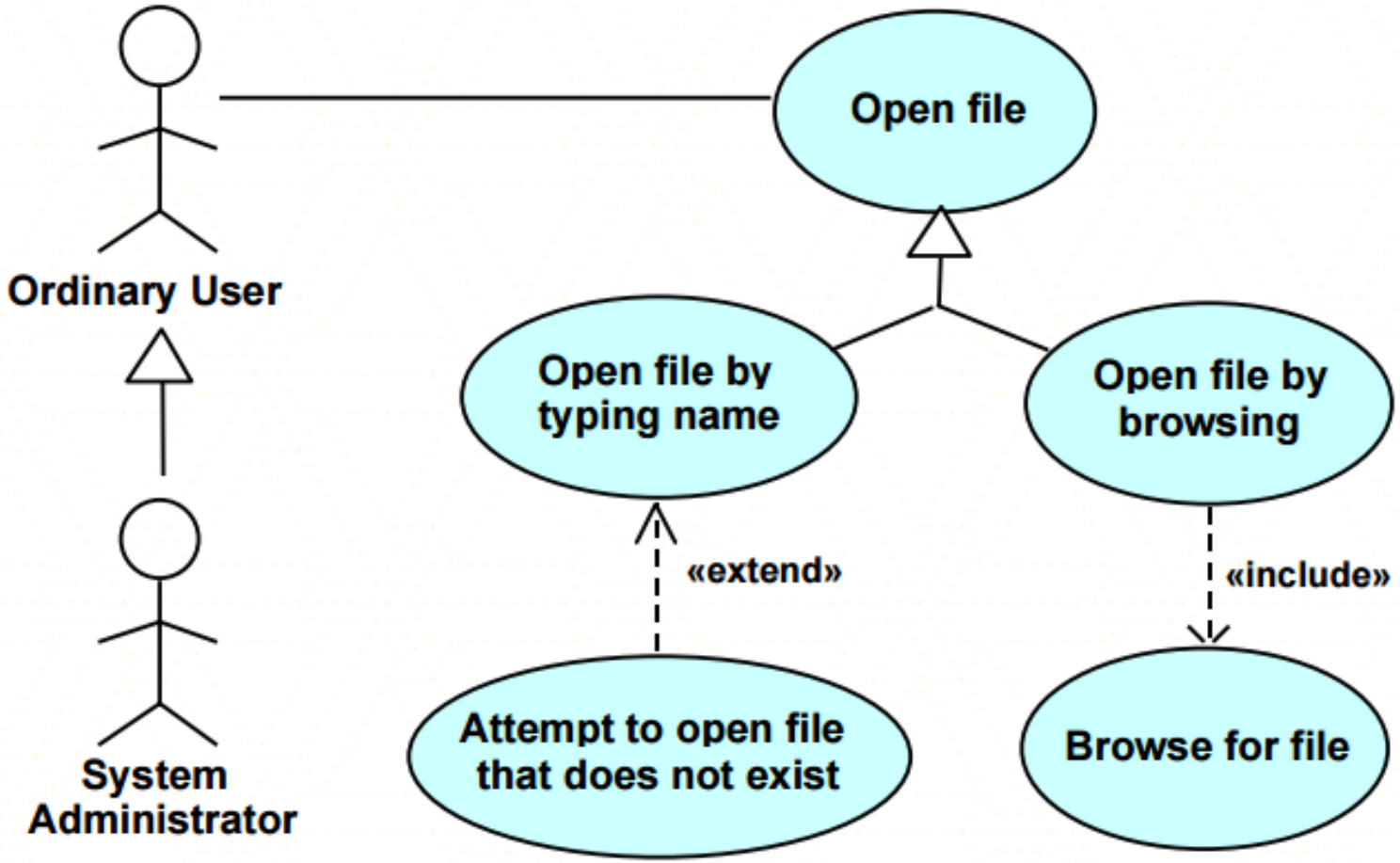
Esempio 1


relazione di generalizzazione tra attori
Aggiorna disponibilità
Utente
Validazione utente
«extend»
Verifica credito
«include»
«extend»
relazione di inclusione
relazione di estensione
Noleggio DVD
Cliente
relazione di generalizzazione tra casi d'uso
Noleggio prenotato
Noleggio ex-novo
relazione di associazione attore-caso d'uso

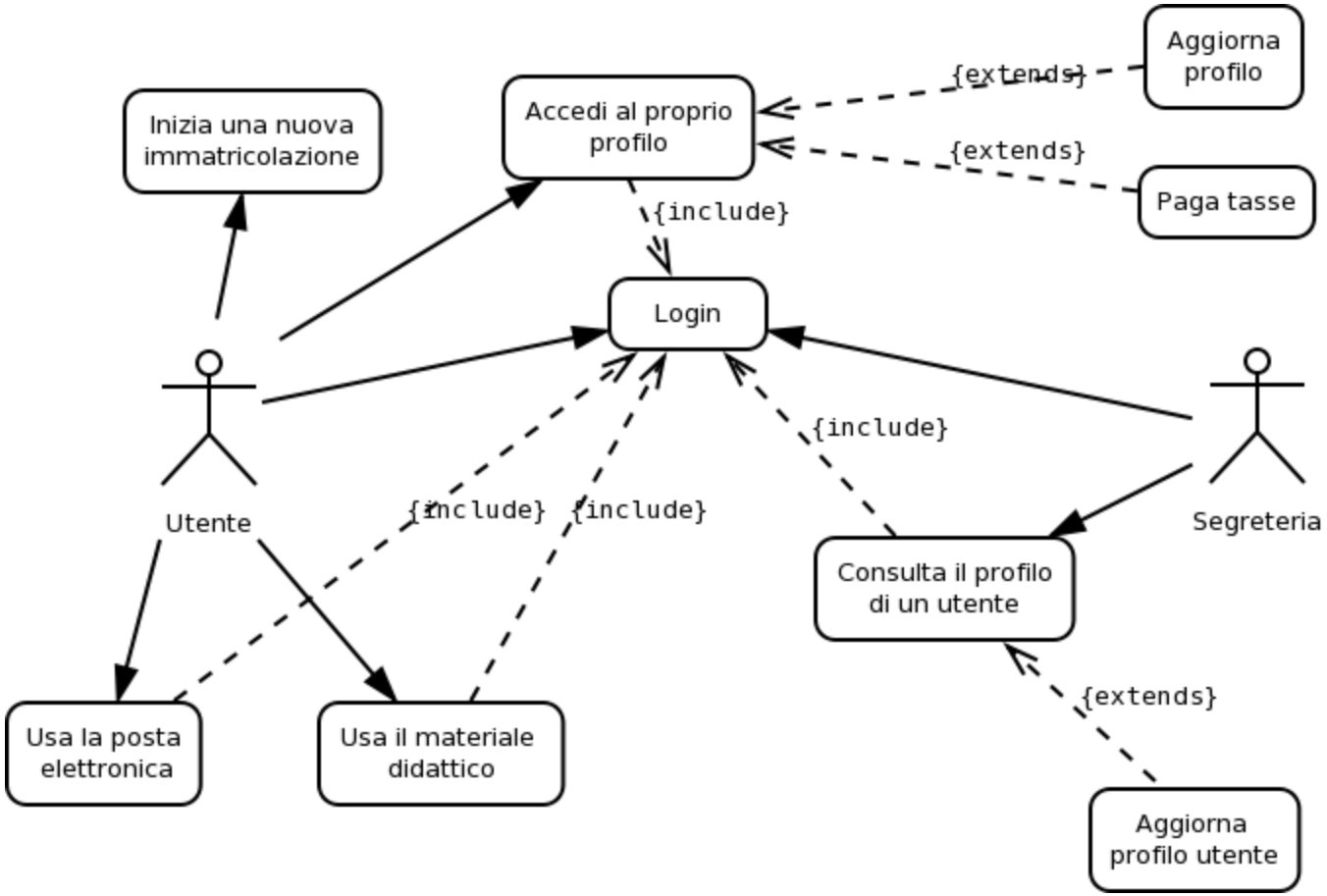
Esempio 2


Dipendente
Spedizione
Aggiorna stato ordine
«include»
Addetto spedizioni
Verifica username e password
Admin
Accesso pannello di amministrazione
«include»
«extend»
«extend»
Aggiungi prodotti
«extend»
Rimuovi prodotti
Modifica prodotti

Esempio 2


Open file
Ordinary User
Open file by typing name
Open file by browsing
«extend»
«include»
System Administrator
Attempt to open file that does not exist
Browse for file

Esempio 3


Aggiorna profilo
{extends}
Inizia una nuova immatricolazione
Accedi al proprio profilo
{extends}
Paga tasse
{include}
Login
{include}
{include}
{include}
Utente
Segreteria
Consulta il profilo di un utente
{extends}
Usa la posta elettronica
Usa il materiale didattico
Aggiorna profilo utente

Fosse davvero tutto qui…

CASO D'USO: *Codice*	*Nome del caso d'uso.*	Data:
		Versione: 0.00.000
Descrizione:	*Descrizione generale del caso d'uso (scope).*	
Priorità:	*Priorità attribuita al caso d'uso dagli utenti.*	
Durata:	*Ordine di grandezza stimata della durata del caso d'uso.*	
Attore primario:	*Nome*	
	Interessi nell'esecuzione del caso d'uso	
Attori secondari:	*Nome*	
	Interessi nell'esecuzione del caso d'uso	
Precondizioni:	*Descrizione.*	
Garanzie:	Minime: *descrizione.*	
	Successo: *descrizione.*	
Avvio:	*Evento che innesca l'avvio del caso d'uso.*	
Scenario principale.		
1.	*<STEP 1>*	
2.	*<STEP 2>*	
3.	*<STEP 3>*	
4.	*<STEP 4>*	
Primo scenario alternativo.		
1.1.	*Elenco delle azioni da eseguire come alternativa a quanto prescritto nel primo passo.*	
Secondo scenario alternativo.		
3.1.	*Elenco delle azioni da eseguire come alternativa a quanto prescritto nel terzo passo.*	
Primo scenario di errore.		
2.1.	*Elenco delle azioni da eseguire nel caso in cui si verifichi una condizione di errore durante l'esecuzione del secondo passo.*	
Annotazioni.		
4.	*Annotazioni relative al punto 4 dello scenario principale.*	