
Introduzione a UML

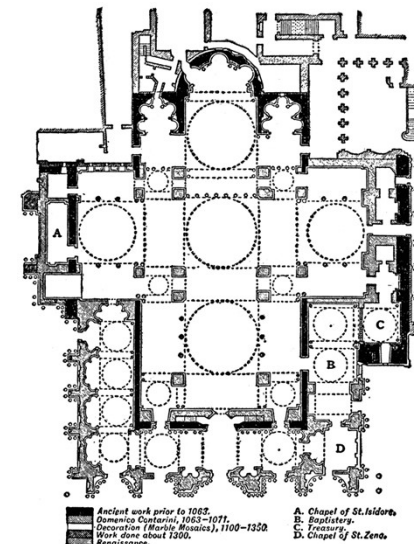
UML: casi d'uso

narrative dei casi d'uso



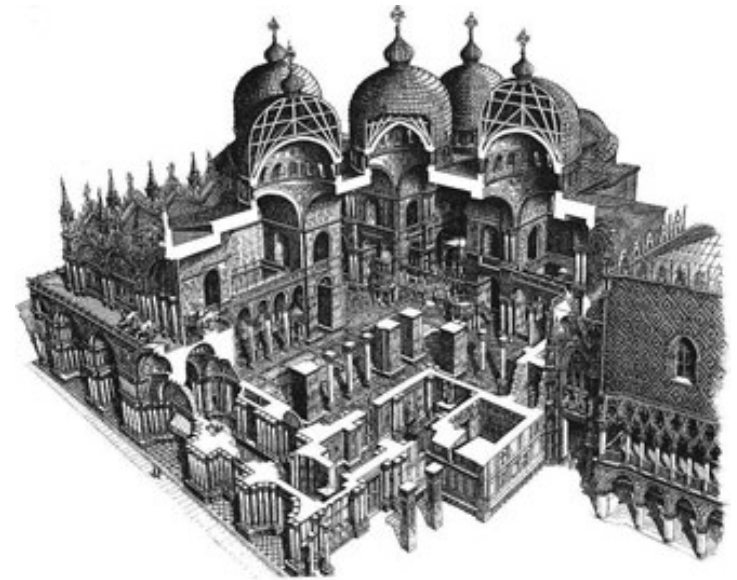
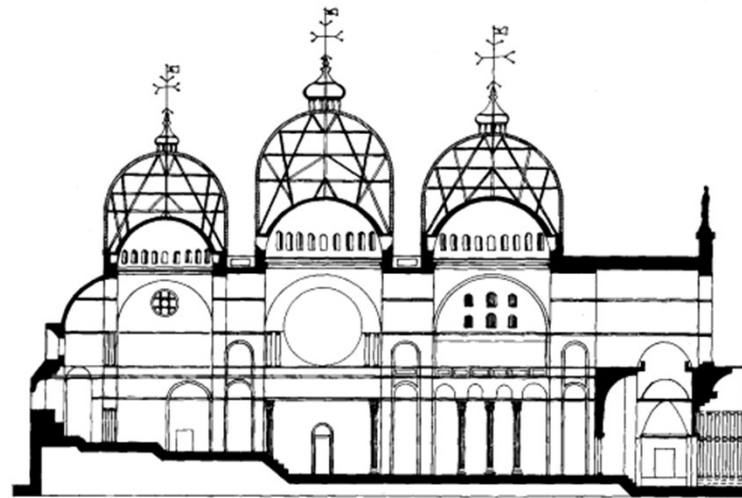
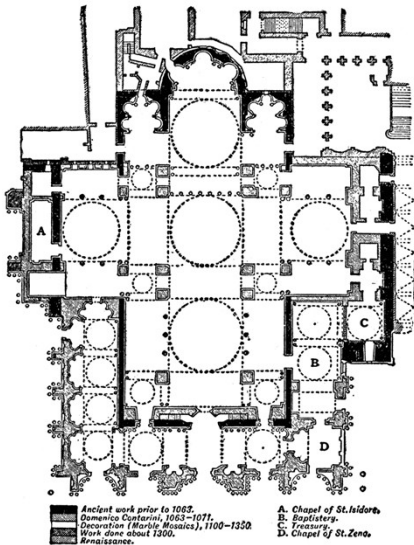
Cosa è un modello

- Astrazione del sistema (o del dominio) usato per specificarne struttura e/o comportamento
- Un modello:
 - cattura aspetti importanti e astrae dai dettagli
 - è espresso con un formalismo che ne rende facile l'uso e la comprensione



Diverse viste su un modello

Un sistema è descritto non da una singola vista (prospettiva) ma da un insieme di viste che nel loro insieme definiscono un modello sufficientemente completo



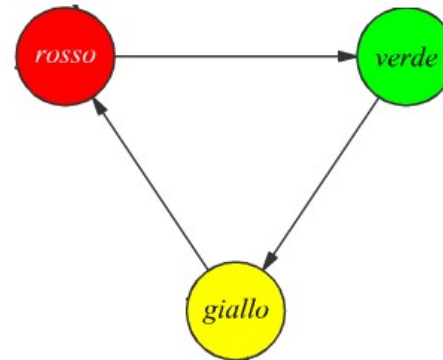
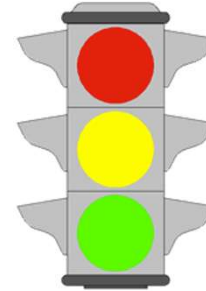
Ancora sui modelli

- Strumento di documentazione, comunicazione e discussione
 - Fondamentale per un processo di progettazione collaborativo, al centro del moderno sviluppo sw
- 1. Come si modella un sistema?
- 2. Come si rappresenta un modello?
- 3. Come si usa un modello?

1. Come si modella un sistema?

Si decide se fornire:

- modello statico
- modello dinamico
- entrambi



In ogni caso si decide a che livello di astrazione porsi

2. Come si rappresenta un modello?

- Con un linguaggio
 - Formale o
 - Semi-formale

3. Come si usa un modello?

- Un modello può essere:
 - Bozza (sketch)
 - *modello non completo, usato per descrizioni iniziali*
 - *Si possono rendere visibili soluzioni alternative*
 - Progetto dettagliato (blueprint)
 - *contiene dettagli sufficienti per permettere agli sviluppatori di creare un ready-to-run system senza prendere decisioni di progettazione*
 - Eseguitabile
 - *talmente completo e preciso da permettere di generare il codice automaticamente dal modello*

UML

Unified Modeling Language: linguaggio di modellazione unificato

- Serve per supportare la descrizione e il progetto di sistemi software in particolare applicazioni Object Oriented, ma non solo
- Famiglia di notazioni grafiche
 - Permettono di descrivere diversi modelli da diversi punti di vista
 - Sono correlate (descritte da un meta-modello comune)
 - Sono generalmente facilmente comprensibili a clienti e utenti

UML

- UML è adatto a descrivere aspetti diversi di un progetto software:
 - Descrizione strutturale (statica) vs Descrizione di un comportamento (dinamica)
 - Bozza di progetto vs progetto finale
 - Descrizione del dominio vs descrizione della struttura del codice

Storia di UML

. . . 1994: molti linguaggi e metodi di modellazione OO.

1994: primo tentativo di unificazione - il linguaggio Fusion

1994: Booch e Rumbaugh si uniscono nella Rational Corporation dando vita a quello che chiamano UML

1996: Object Management Group (OMG) lancia una prima proposta di standardizzazione di UML

1997: OMG approva lo standard UML 1.0

2000: UML 1.4

2006: UML 2.0

2006 : Model Driven Architecture (MDA)

Dal 2017, stabile alla versione 2.5.1

<https://www.omg.org/spec/UML/>

Modello statico e dinamico

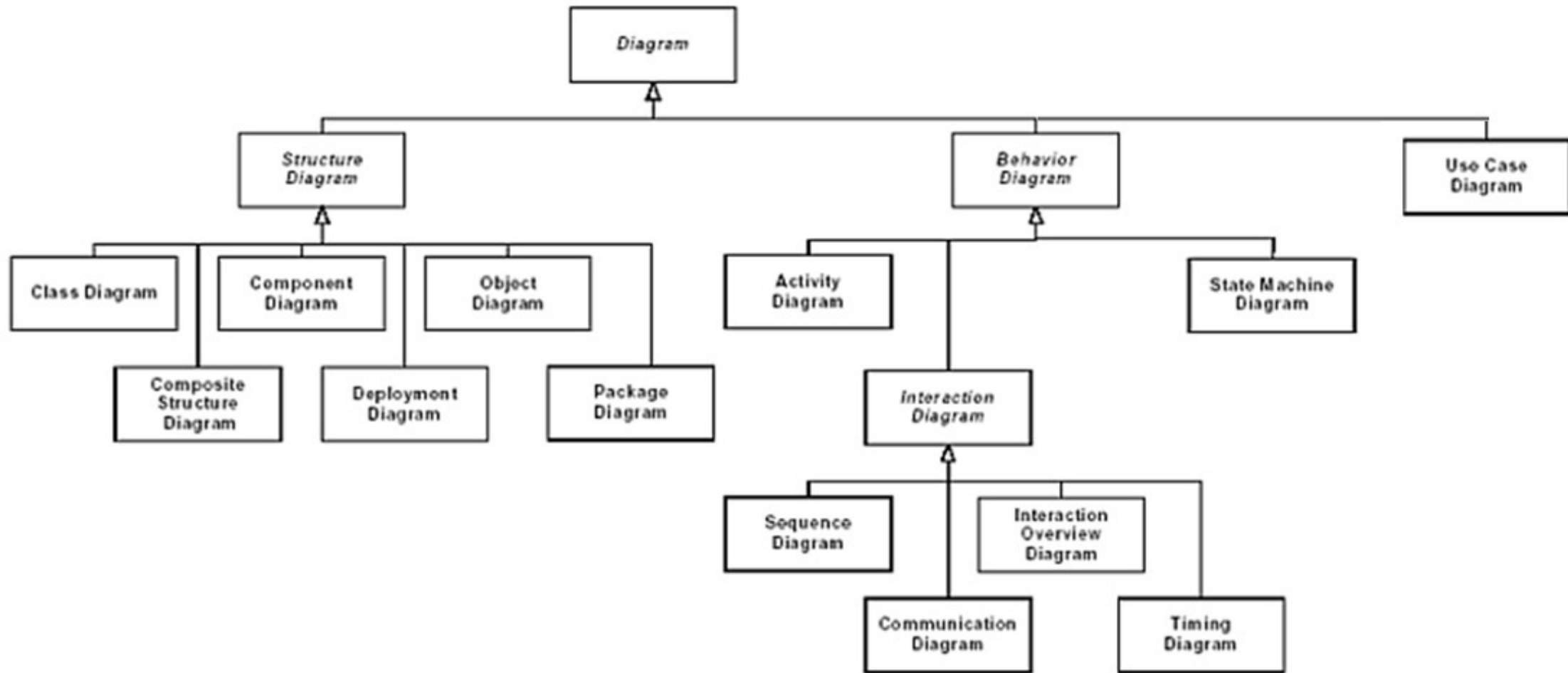
Modello statico

- Entità e relazioni per descrivere:
 - Concetti del dominio
 - Componenti architettura
 - Classi di realizzazione
- Assenza degli aspetti dipendenti dal tempo

Modello Dinamico

- Modella il comportamento delle entità descritte nel modello statico

Diagrammi in UML: 14 diversi



Diagrammi che vedremo nel corso

- Diag. dei casi d'uso
- Modello statico
 - Diag. delle classi
 - Diag. degli oggetti
 - Diag. delle componenti
 - Diag. di dislocazione
- Modello dinamico
 - Diag. di macchina a stati
 - Diag. di attività
 - Diag. di sequenza

Diagramma:

- Rappresentazione grafica
 - di un insieme di elementi del modello
- Grafo
 - nodi = elementi del modello
 - archi = relazioni fra elementi

Strumenti per UML

Esistono tantissimi strumenti per UML:

https://en.wikipedia.org/wiki/List_of_Uniformed_Modeling_Language_tools

Noi useremo: Visual paradigm,
in particolare la
Free Community Edition

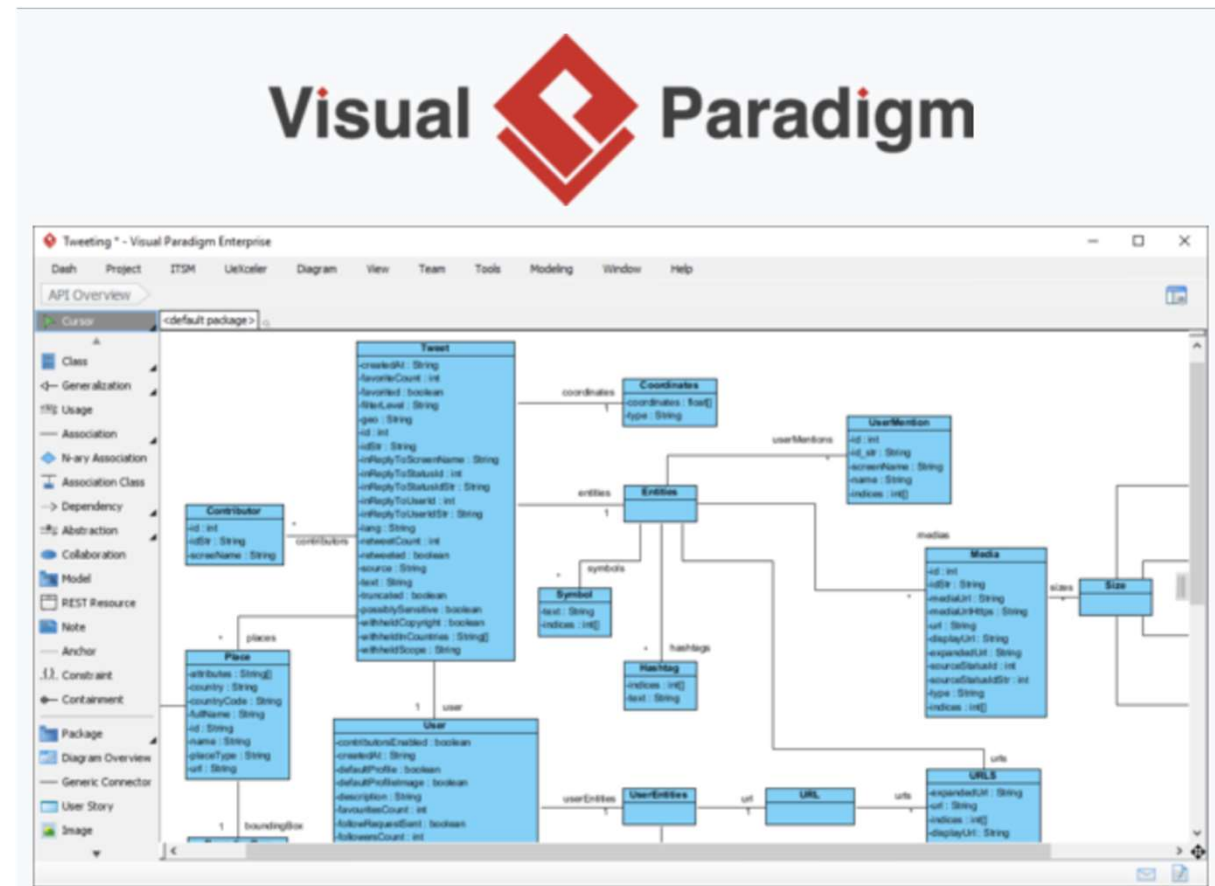


Diagramma dei casi d'uso



Contenuti

- **Modellazione dei Requisiti:**
 - aiuta a raccogliere i requisiti funzionali, elaborarli e documentarli
- **Modello statico:**
 - diagramma dei casi d'uso
- **Modello dinamico:**
 - narrative associate ai casi d'uso

Diagramma dei casi d'uso

- Descrive i **requisiti funzionali** del sistema
- Cattura le funzionalità che un sistema deve offrire, **visto dall'esterno**:
 - *i compiti che un utente può realizzare con l'aiuto del sistema.*
- Un **attore** è un'entità esterna al sistema, che interagisce direttamente con esso in un determinato ruolo
- Tre tipologie di attori
 - utente umano in un determinato ruolo
 - altro sistema
 - tempo (attore speciale)

Diagramma dei casi d'uso

- Un *caso d'uso* è
 - una funzionalità o un servizio offerto dal sistema a uno o più attori
 - formalmente: un compito che un attore può svolgere con l'aiuto del sistema
 - espressa come un insieme di SCENARI
- Uno *scenario* è
 - Una sequenza di interazioni (scambi di messaggi) tra sistema e attori

Costruzione del diagramma dei casi d'uso

La modellazione dei requisiti usando il diagramma dei casi d'uso prevede i seguenti passi:

1. Individuare il confine del sistema
2. Individuare gli attori
3. Individuare i casi d'uso
4. Individuare le relazioni attore-caso d'uso
5. Specificare il caso d'uso con una descrizione testuale (narrativa)

Diagramma dei casi d'uso

Il diagramma è composto da:

- **Attori:** hanno un nome (Maiuscolo, è una classe)
- **Casi d'uso:** ha un nome (Maiuscolo, è un verbo che denota il compito)
- **Relazioni:** relazioni significative tra gli attori e casi d'uso
 - Rappresentano interazione (sequenze di messaggi).
- **Confine del sistema:** un rettangolo disegnato intorno ai casi d'uso per indicare il confine del sistema oggetto del modello

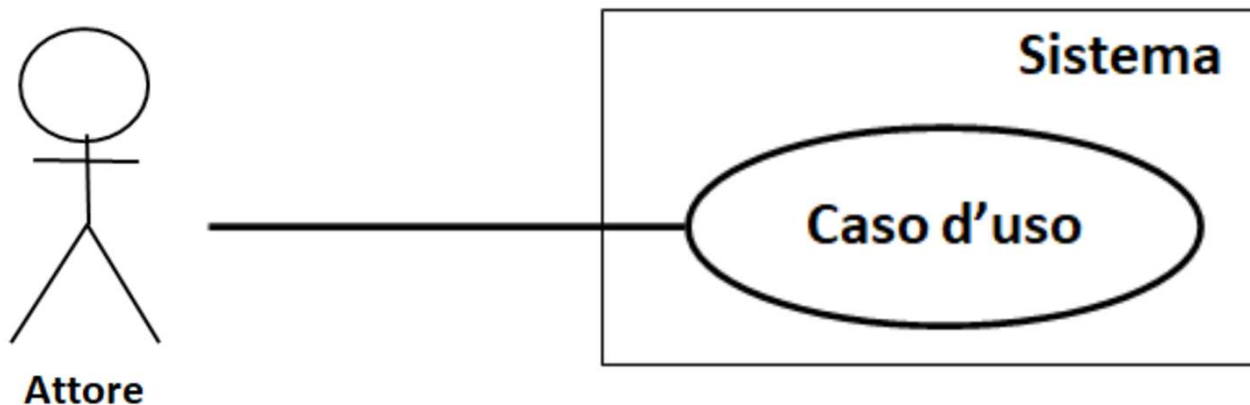
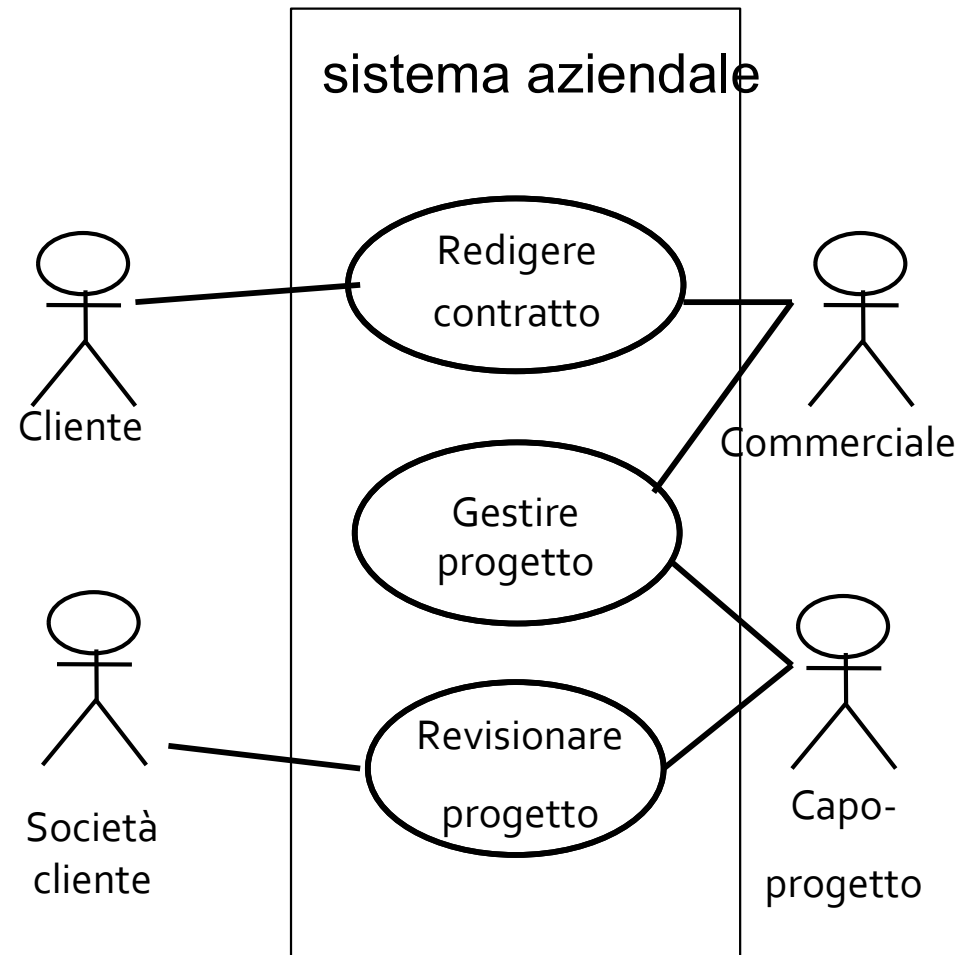
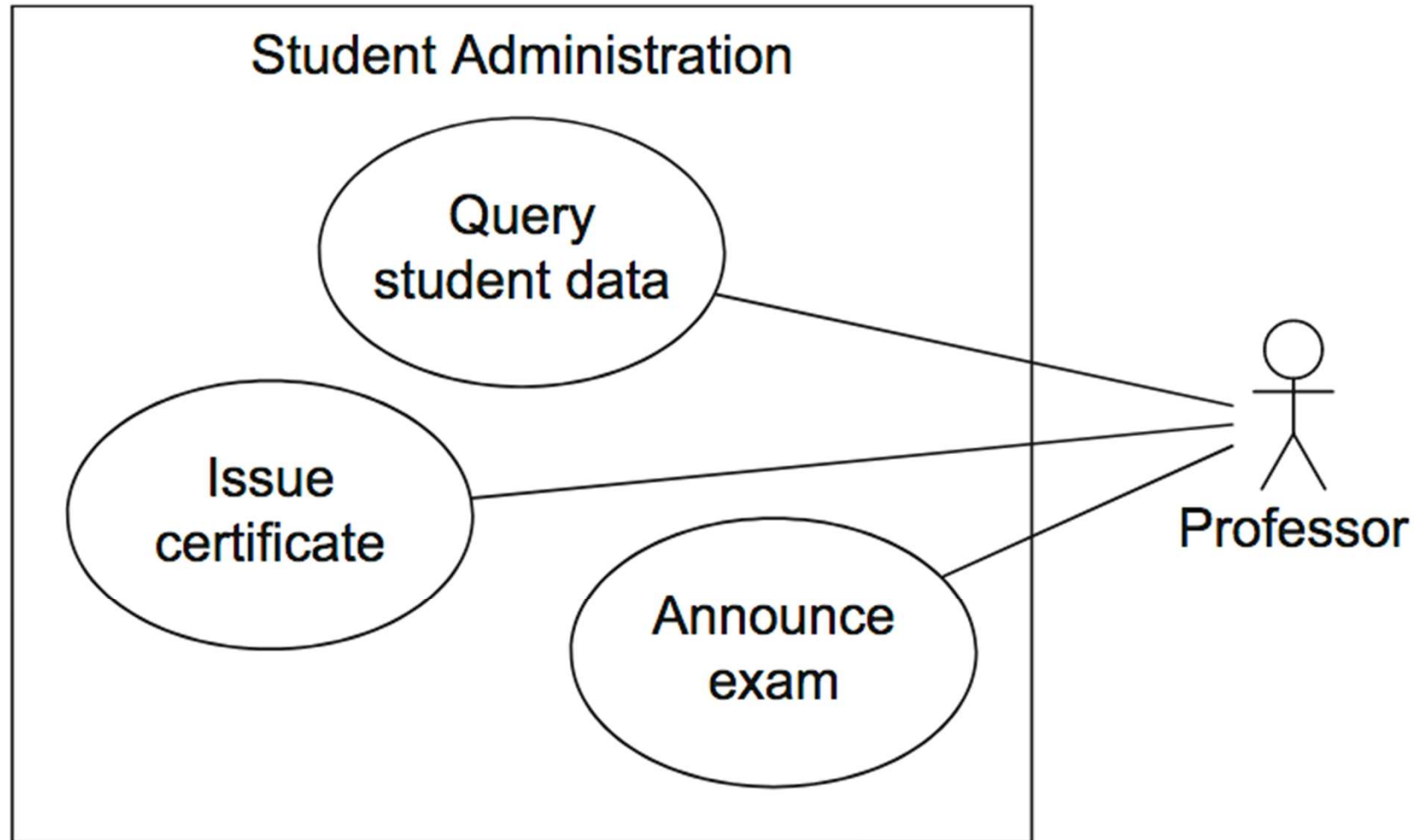


Diagramma dei casi d'uso

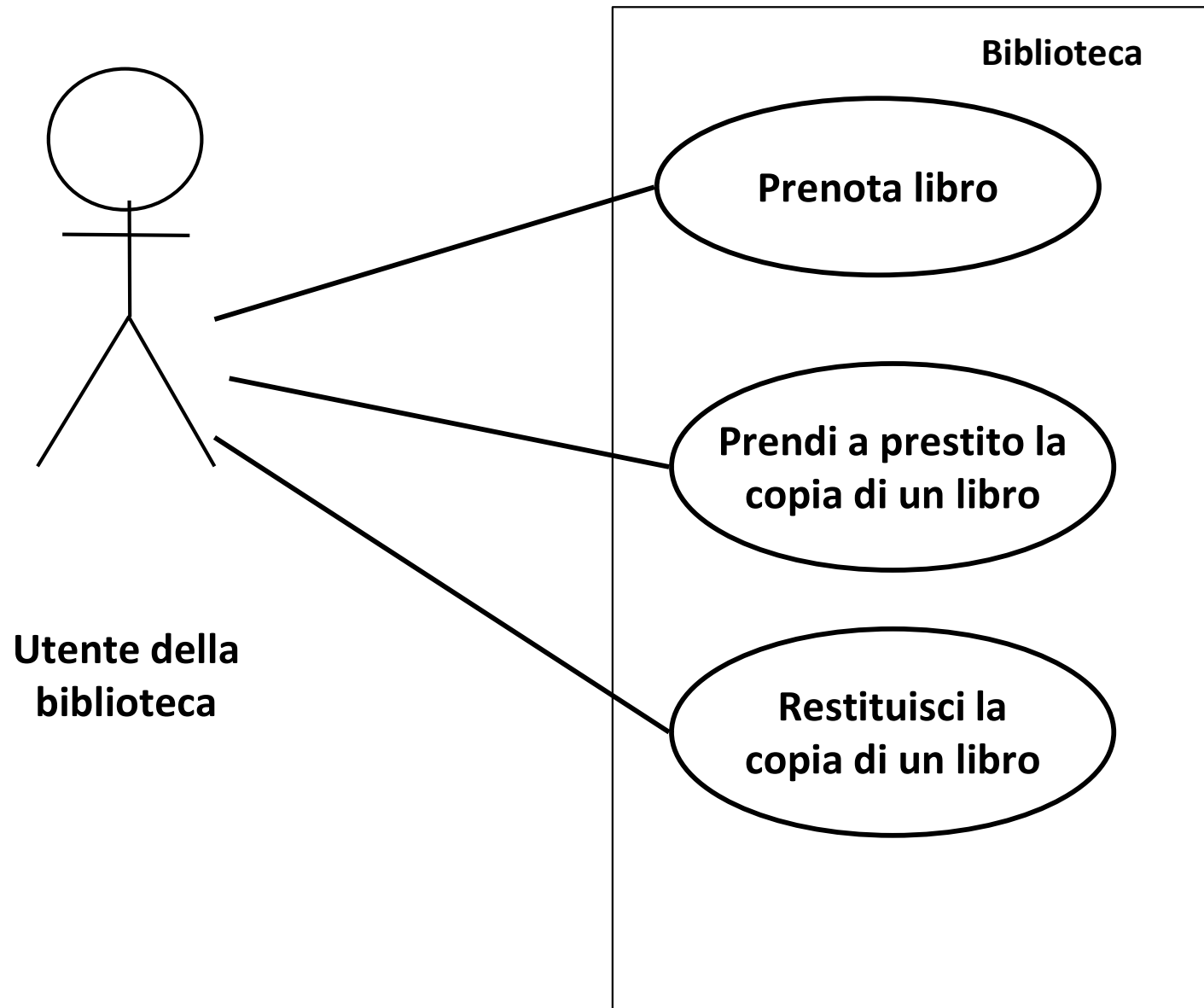
- L'associazione attori—casi d'uso è molti a molti: un attore può essere associato a più casi d'uso e viceversa
- Un caso d'uso è iniziato sempre da uno e un solo attore, detto principale
 - **Casi d'uso non collegati a un attore sono in casi particolari che vedremo**
 - **Eventualmente Tempo**



Esempio di diagramma dei casi d'uso

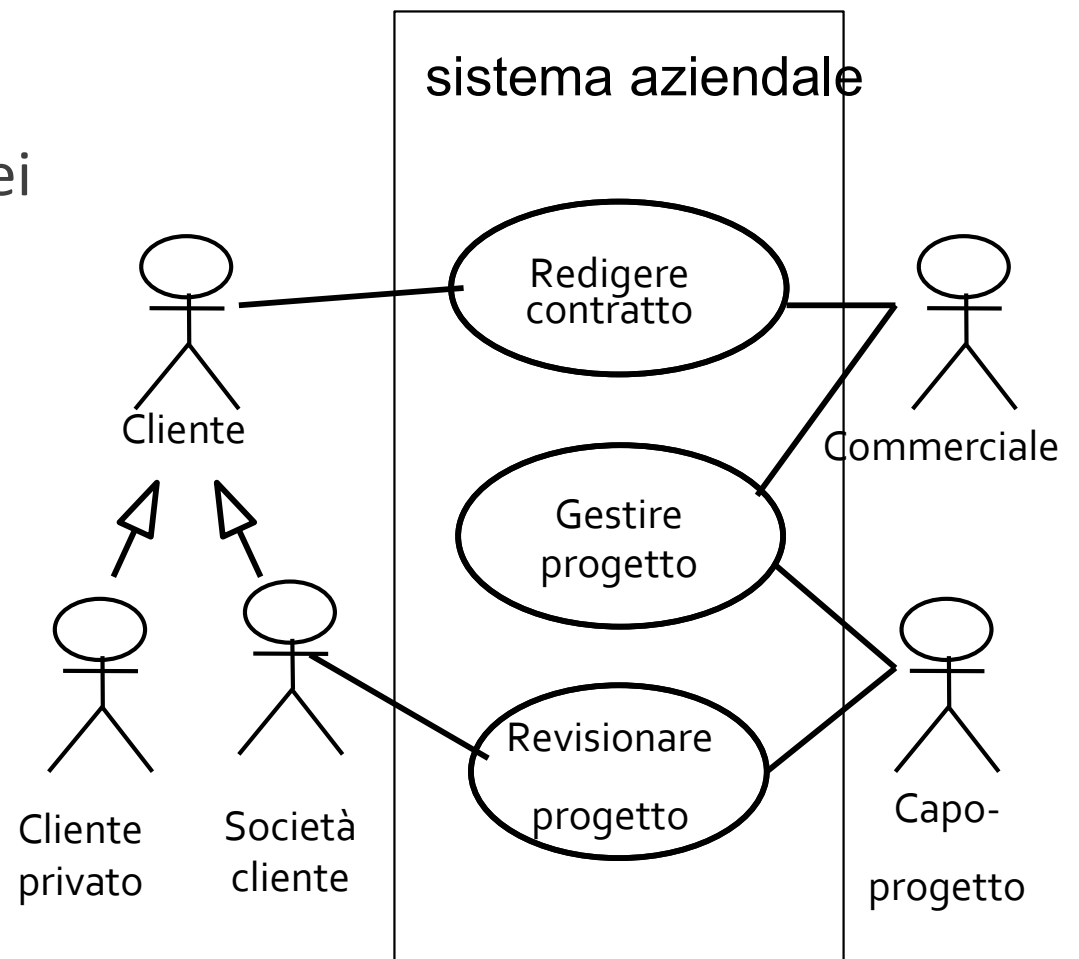


Esempio di diagramma dei casi d'uso



Esempio di diagramma dei casi d'uso

- I clienti possono essere privati o società
- Clienti e commerciali partecipano alla redazione dei contratti
- Società e capiprogetto partecipano alle revisioni
- Commerciali e capiprogetto gestiscono i progetti
- N.B.: Non è chiaro in questi esempi chi sia l'attore principale: lo chiarisce la narrativa



Narrativa di un caso d'uso

- Documento che descrive il modello dinamico
 - gli scenari rilevanti di un caso d'uso
 - dal punto di vista degli attori
 - Compreso chi lo inizia, i.e. chi è l'attore principale (aka primario)
- Caratteristiche
 - Inizio, fine, passi intermedi
 - Condizioni eccezionali
 - Precondizioni, postcondizioni

Narrativa di un caso d'uso

Nome:
ID:

Nome del caso d'uso
identificatore

Breve descrizione:

Due righe riassuntive

Attore primario:

Attore che avvia il caso d'uso

Attori secondari:

Altri attori che interagiscono con il caso d'uso

Precondizioni:

Devono valere prima dell'esecuzione del caso d'uso

*Sequenza degli
eventi principale:*

Sequenza di passi

Postcondizioni:

Della sequenza principale

*Sequenze alternative degli
eventi:*

Errori, ramificazioni e interruzioni nella sequenza principale

Narrativa di un caso d'uso: esempio

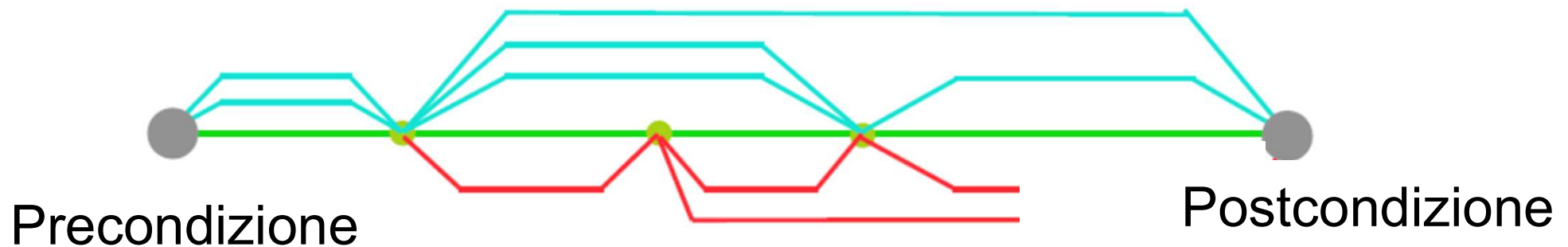
<i>Nome:</i>	Estendi il prestito
<i>Breve descrizione:</i>	Un utente desidera prolungare il periodo di prestito di un libro
<i>Attore primario:</i>	Utente della biblioteca
<i>Attori secondari:</i>	Nessuno
<i>Precondizioni:</i>	Il libro è già assegnato all'utente
<i>Sequenza degli eventi principale:</i>	<ol style="list-style-type: none">1. L'utente chiede di prolungare il prestito2. Il Sistema accerta che nessuno l'abbia prenotato3. Il Sistema accerta che l'utente sia ammesso al prestito4. Il Sistema aggiorna utente e stato del libro
<i>Postcondizioni:</i>	Prestito esteso, stato utente aggiornato
<i>Sequenze alternative degli eventi:</i>	Il libro è già prenotato, utente non ammesso al prestito

Narrativa di un caso d'uso: esempio

<i>Nome:</i>	Estendi il prestito
<i>Breve descrizione:</i>	Un utente desidera prolungare il periodo di prestito di un libro
<i>Attore primario:</i>	Utente della biblioteca
<i>Attori secondari:</i>	Nessuno
<i>Precondizioni:</i>	Il libro è già assegnato all'utente
<i>Sequenza degli eventi principale:</i>	<ol style="list-style-type: none">1. L'utente chiede di prolungare il prestito2. Il Sistema verifica se qualcuno ha prenotato il libro3. If (prenotato)<ol style="list-style-type: none">3.1 Il sistema avverte l'utente4. else<ol style="list-style-type: none">4.1 Il Sistema accerta che l'utente sia ammesso al prestito4.2 Il Sistema aggiorna utente e stato del libro
<i>Postcondizioni:</i>	Prestito esteso, stato utente aggiornato OR libro già prenotato e utente avvisato
<i>Sequenze alternative degli eventi:</i>	Utente non ammesso al prestito

Scenario

- Uno scenario è un'istanza di un caso d'uso: una sequenza di passi che produce un risultato osservabile da uno o più attori
- Gli scenari descritti dalla sequenza principale degli eventi sono quelli che portano alla postcondizione



Precondizioni e postcondizioni: nota importante

- Pre- e postcondizioni sono asserzioni che devono essere vere in uno stato.
- Sono quindi predicati o formule logiche che valgono vero o falso in uno stato: precondizione prima di iniziare il caso d'uso, postcondizioni alla fine
- Non sono MAI azioni

Esempio sbagliato ed es. corretto

Nome: Registra account

Attore principale: Utente

Precondizione: L'utente ha cliccato sul link "Registrati" nella pagina principale del sito web.

Sequenza principale degli eventi:

1. Il sistema mostra il modulo di registrazione con i campi nome, cognome, email, password e conferma password.
2.

E' **sbagliato** perché quanto scritto nella precondizione è la prima azione dell'utente, che inizia il caso d'uso

Nome: Registra account

Attore principale: Utente

Precondizione: L'utente vuole registrarsi

Scenario principale:

1. L'utente clicca sul link "Registrati" nella pagina principale del sito web.
2. Il sistema mostra il modulo di registrazione con i campi nome, cognome, email, password e conferma password.
3. ...

Corretto

Relazione tra preconditione, postcondizione, sequenza principale e sequenza alternativa

- Parallelo con le triple di Hoare:
- {Precondizione} sequenza principale {Postcondizione}
- Quindi:
 - per ogni stato σ che soddisfa la preconditione, l'esecuzione del caso d'uso a partire da σ termina producendo uno stato σ' che soddisfa la postcondizione
- ... a meno che non si verifichi qualche imprevisto come quelli elencati nella sequenza alternativa, in questo caso la postcondizione non è garantita

Sintassi della sequenza principale degli eventi (S.P.E.)

- La S.P.E. elenca i passi che compongono il caso d'uso.
- La sequenza è numerata e ogni passo dovrebbe avere la struttura:
<numero>. <soggetto><azione><complementi>
- Il primo passo (attivazione) è sempre compiuto dall'attore principale.
 1. L'utente chiede di prolungare il prestito
 2. Il sistema controlla
 3.

Sintassi della sequenza principale degli eventi (S.P.E.)

Nella S.P.E. si possono descrivere condizioni e cicli.

1. Se (espressione booleana)

1.1

1.2

Altrimenti [OPZIONALE]

1.3

2. For (espressione di iterazione)

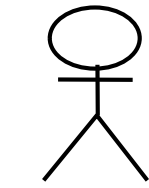
2.1

3. While (espressione booleana)

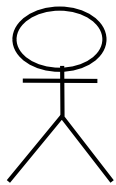
3.1 ...

3.2 ...

Casi d'uso: generalizzazione tra attori



Cliente



Cliente privato



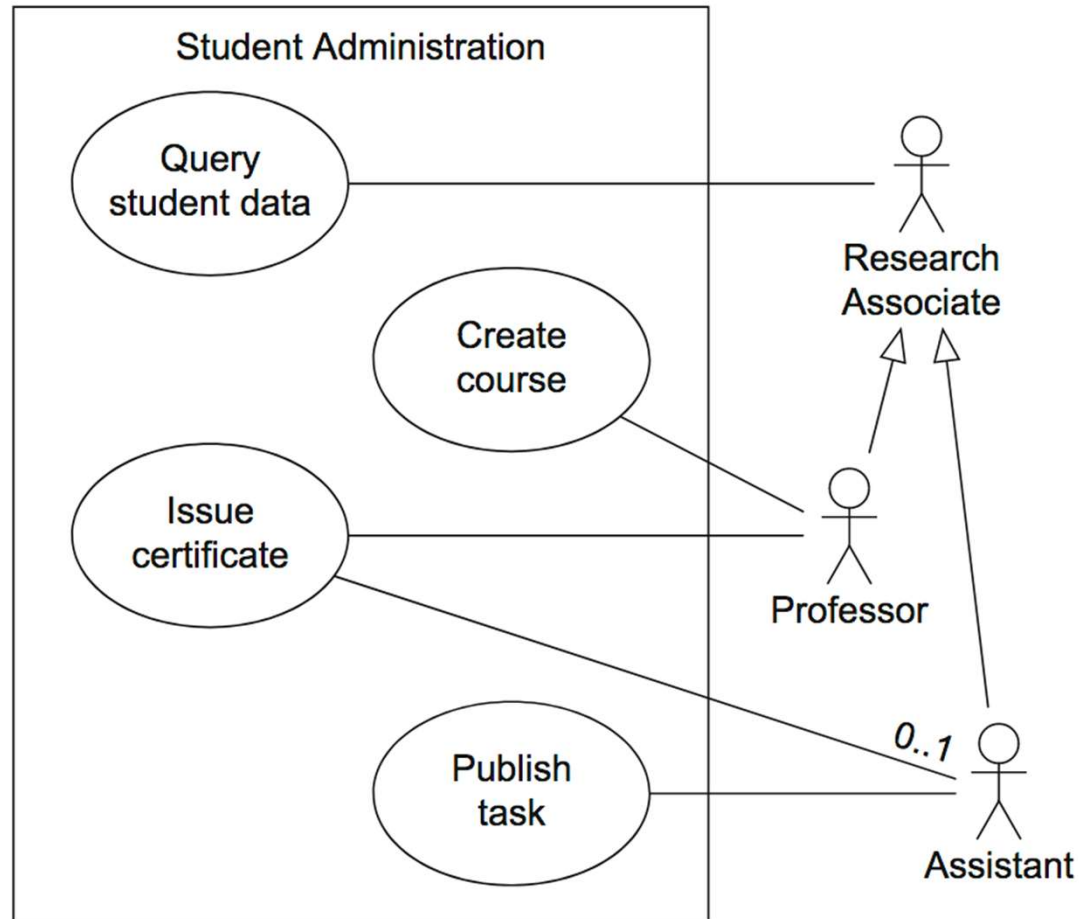
{abstract}
Research
Associate



Professor Assistant

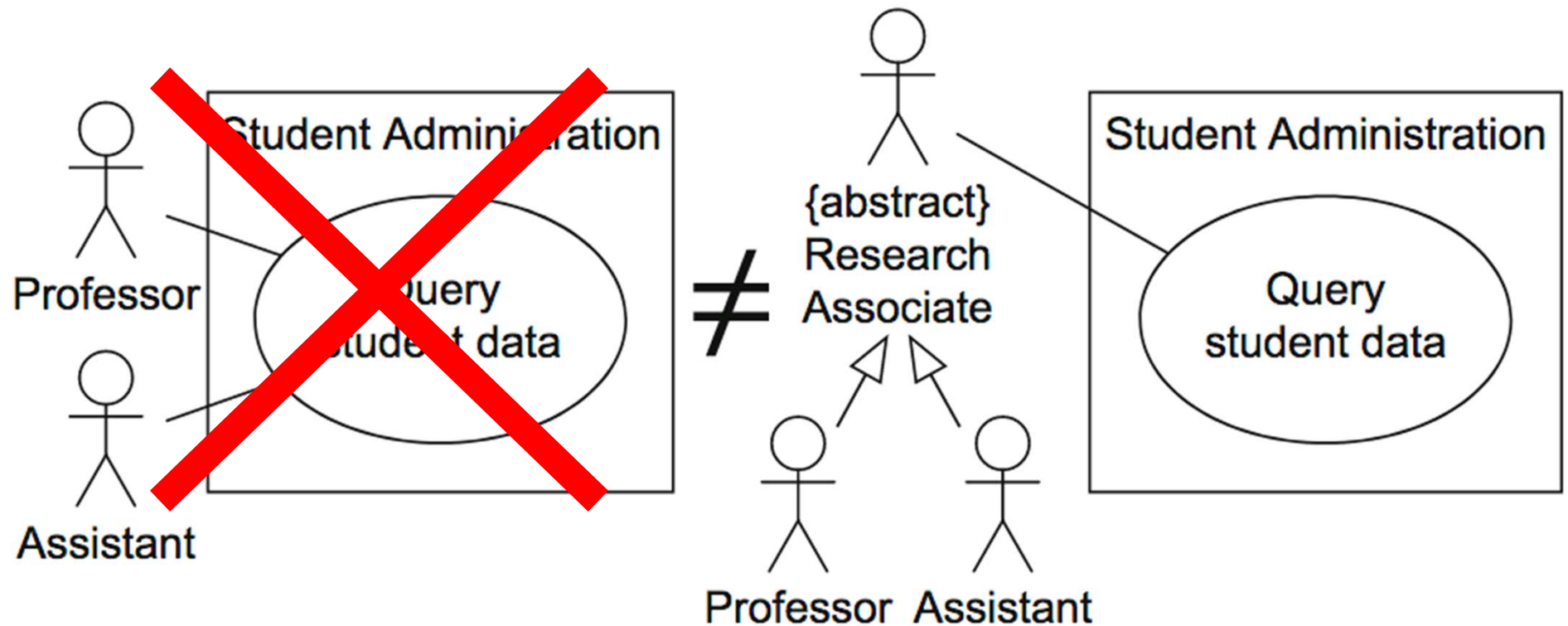
È possibile che
l'attore padre
sia astratto

Esempi di generalizzazione

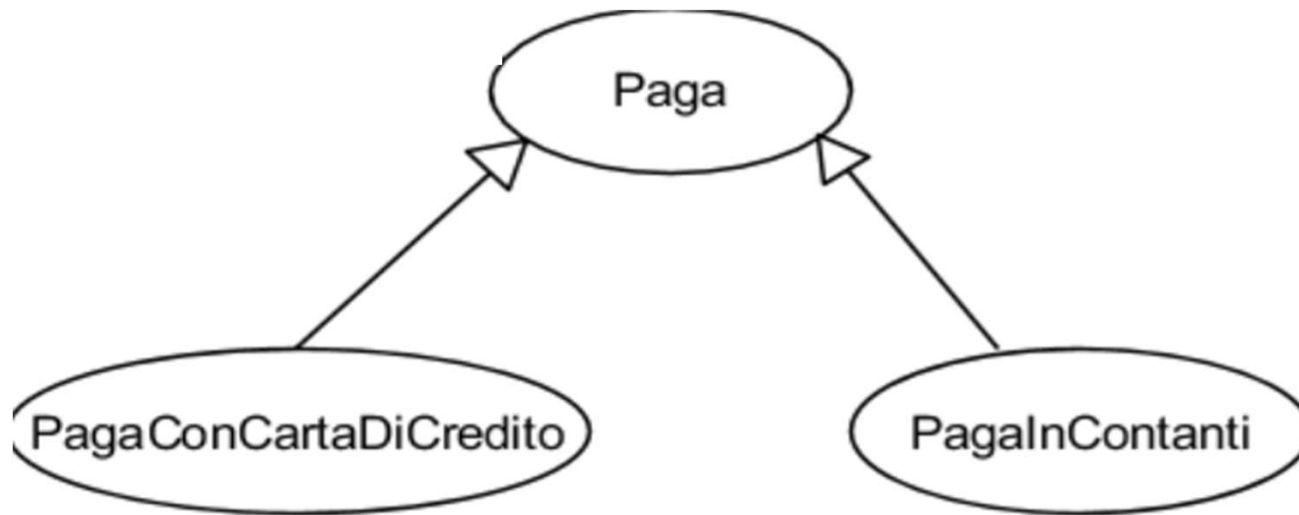


Esempi di generalizzazione

L'introduzione di un attore astratto qui serve per soddisfare il vincolo di avere un solo attore principale

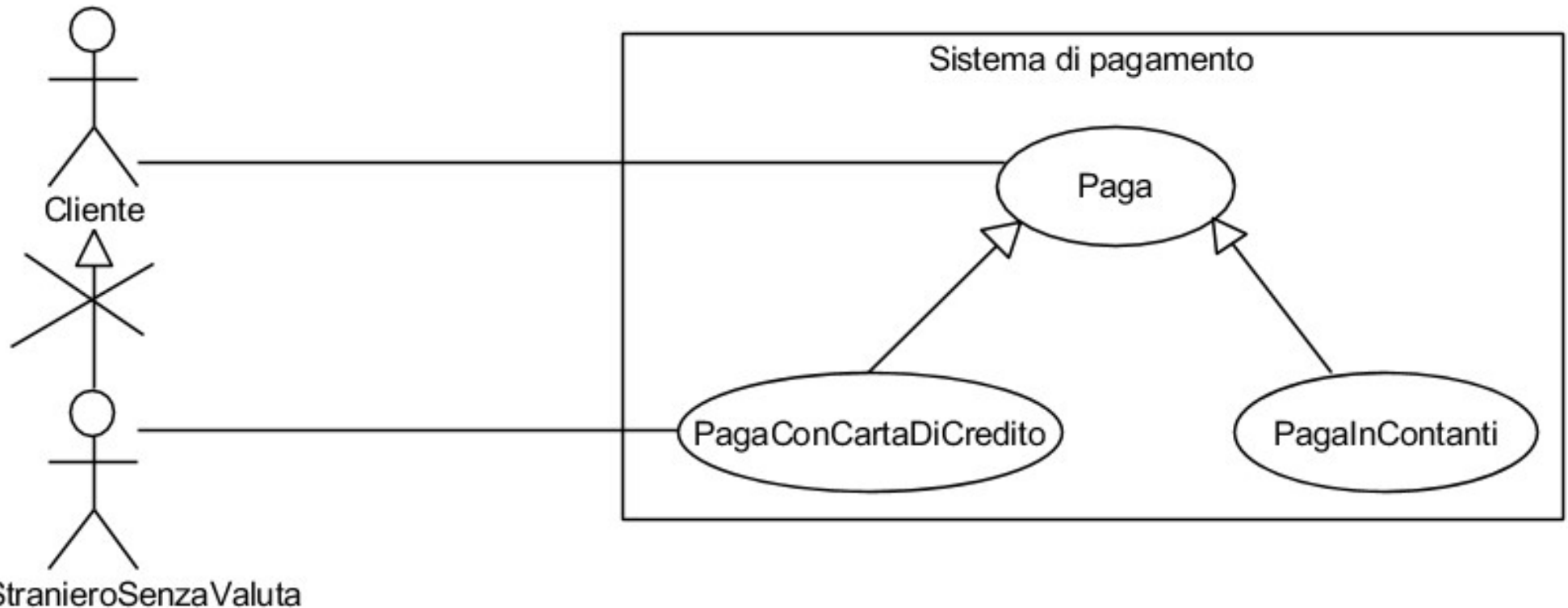


Generalizzazione dei casi d'uso

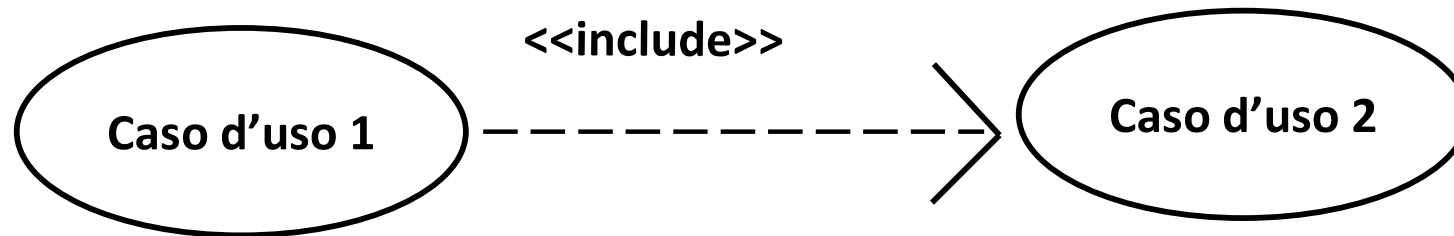


Attenti alla generalizzazione...

Usando la generalizzazione fare attenzione che il classificatore specializzato eredita tutte le relazioni del classificatore padre (Liskov), quindi il ClienteStraniero NON eredita da Cliente

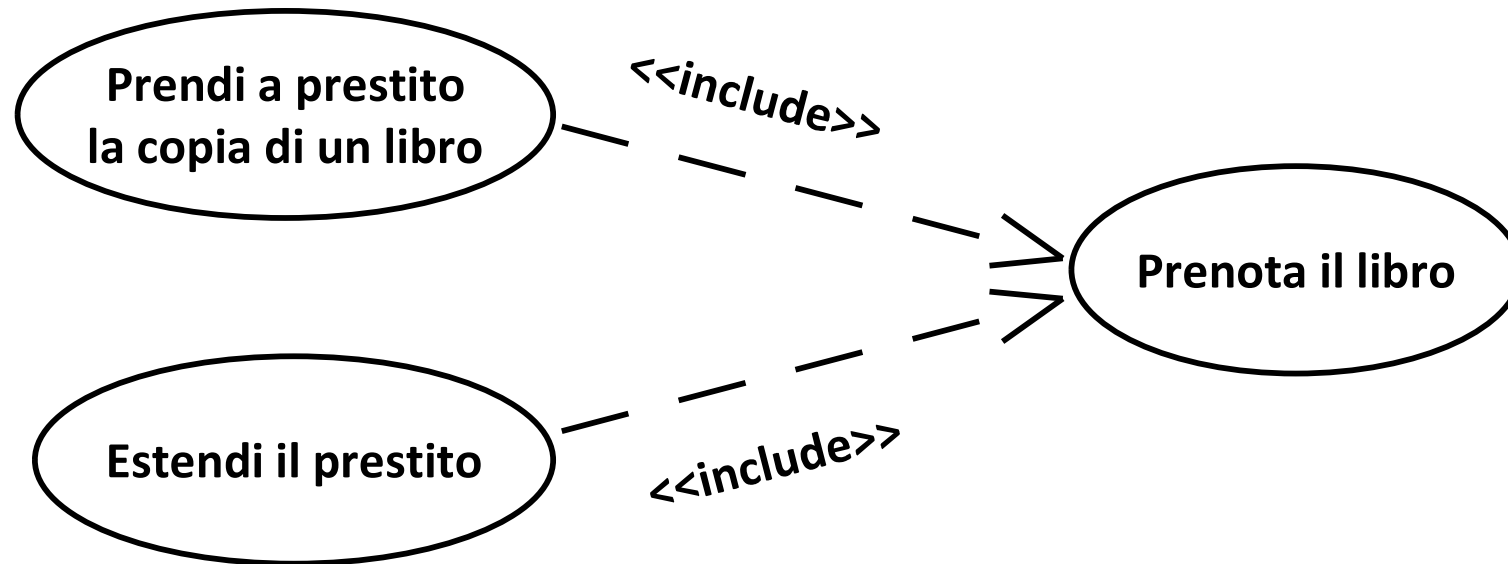


Inclusione di caso d'uso



- Il caso d'uso 1 incorpora l'interazione descritta dal caso d'uso 2 come parte dell'interazione che esso (1) descrive
- La relazione *include* punta al caso d'uso che si include
- Simile ad una chiamata di funzione
 - La narrazione del caso 1 dovrebbe invocare l'esecuzione del caso incluso (2)
- <<include>> è uno stereotipo
 - Gli stereotipi sono keywords tre « » che annotano elementi di un diagramma, per precisarne il significato
 - In questo caso specifica che la **relazione di dipendenza** ----> è un'inclusione

Esempio di inclusione



Sia per prendere in prestito la copia di un libro che per estendere il prestito occorre prenotare il libro

N.B. Questo è un uso corretto della relazione di inclusione: non usatela per fare decomposizione funzionale di un caso d'uso

Inclusione nella narrativa

1. ...

2. ...

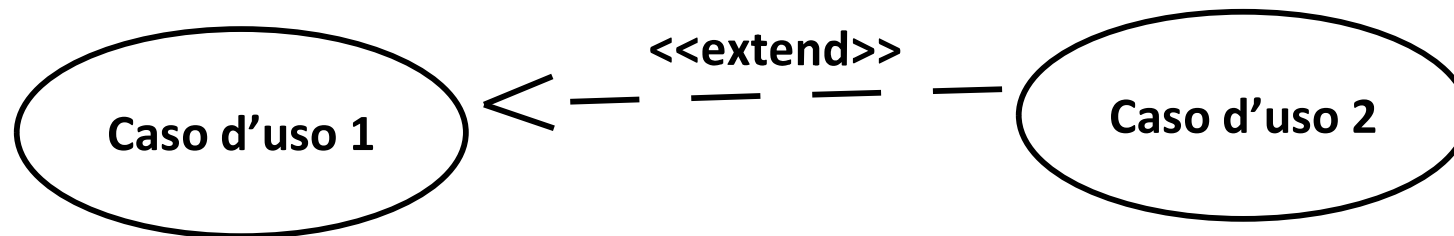
3. **Include (nome caso incluso)**

4. ...

- Il caso d'uso incluso è specificato a parte
 - Può essere istanziabile (completo) (avviato da un attore)
 -o non istanziabile (eseguito solo quando incluso)

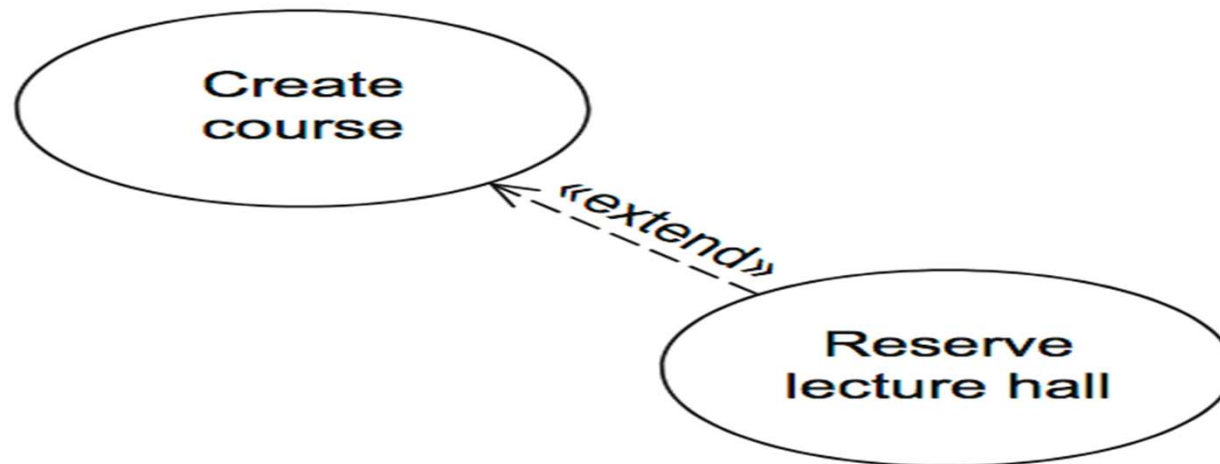
Estensione di un caso d'uso

Attenzione al verso della freccia
opposto all'include



- Il caso d'uso 1 può incorporare l'interazione descritta dal caso d'uso 2 (non dipende da esso)
- La relazione *extend* punta al caso d'uso che viene esteso
 - La narrazione del caso 1 può in qualche caso invocare l'esecuzione del caso (2)
- <<extend>> è anch'esso uno stereotipo

Esempio di Estensione



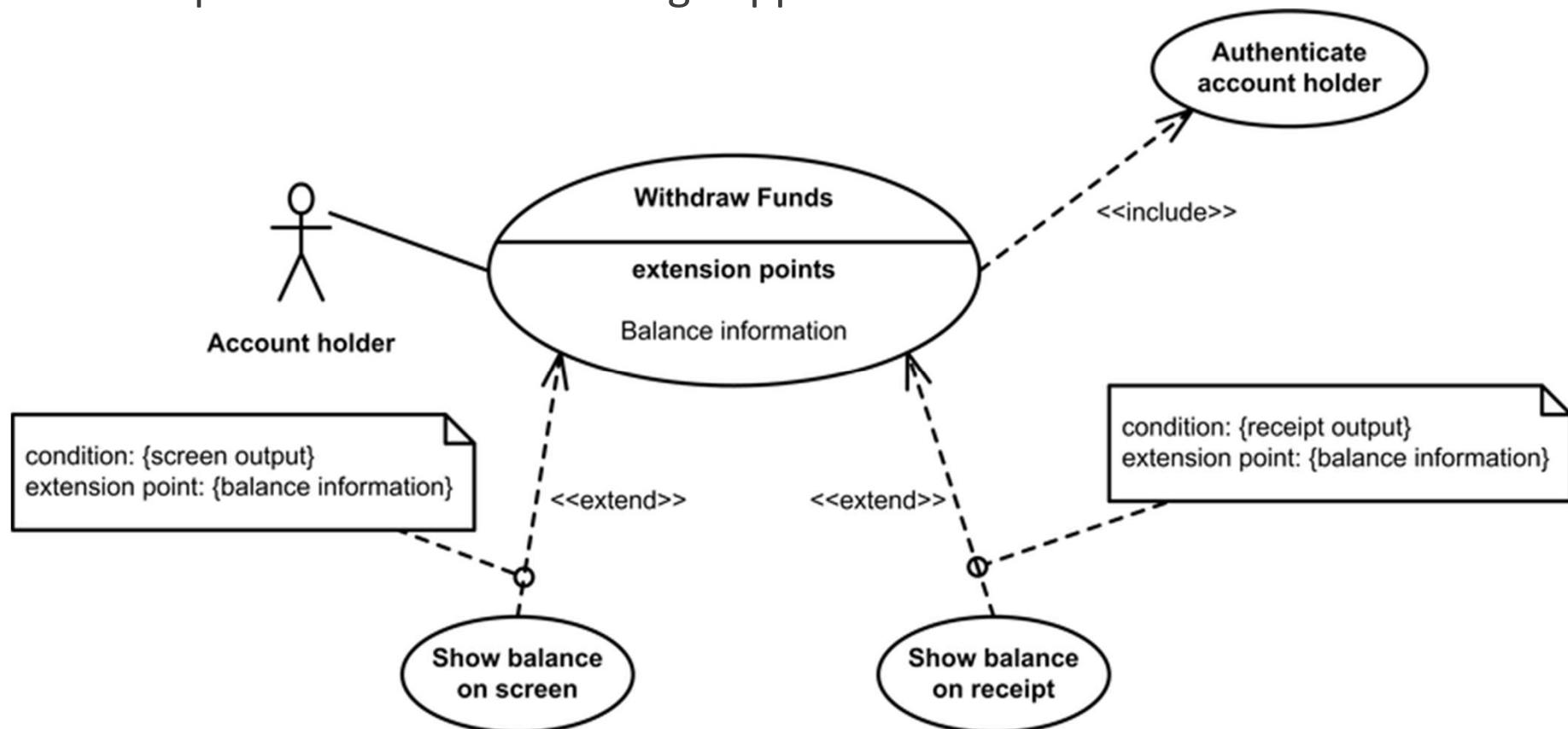
Quando si crea un nuovo corso **si può** prenotare un'aula ma **tale prenotazione non è normalmente parte della creazione del corso**

Altrimenti la prenotazione sarebbe stata un passo della sequenza principale degli eventi e la definizione di due casi separati un'inutile ed erronea decomposizione funzionale

Extension points e condizioni

Le estensioni sono opzionali dal punto di vista del Caso d'uso base → esiste una notazione aggiuntiva che aiuta a identificare quando e dove inserire l'estensione.

Possiamo collegare un vincolo alla freccia "extend", indicando la condizione che deve essere vera perché l'estensione venga applicata.



Syllabus

UML@Classroom, Martina Seidl, Marion Scholz, Christian Huemer, Gerti Kappel, Springer Verlag, 2015

Cap 1-2-3

Homework

- Installate Visual paradigm e disegnate il diagramma dei casi d'uso del seguente sistema (chiavi magnetiche)
- Per motivi di sicurezza, un'organizzazione ha deciso di realizzare un sistema secondo il quale a ogni dipendente è assegnata una chiave magnetica per accedere (aprire) determinate stanze. I diritti di accesso dipenderanno in generale dalla posizione e dalle responsabilità del dipendente. Quindi sono necessarie operazioni per modificare i diritti di accesso posseduti da una chiave se il suo proprietario cambia ruolo nell'organizzazione.