ESERCITAZIONE

Francesco Poggi

fpoggi@cs.unibo.it

A.A. 2014-2015

Outline

Cosa vedremo oggi:

- Analisi e progettazione
- Schede CRC
- Diagramma delle classi

Analisi vs. Progettazione

- L'analisi **modella** i concetti chiave del **dominio** del problema.
- La progettazione **adatta** il modello di analisi e lo **completa** affinché diventi **implementabile**.

In altre parole...

- L'analisi è vicina al **problema**.
- La progettazione è vicina alla **soluzione**.

Dal punto di vista di UML, non si usano primitive o diagrammi diversi, ma gli stessi tipi di diagramma con diversi livelli di dettaglio (i diagrammi di analisi sono più 'astratti' di quelli di progettazione).

Come procedere: Analisi

- Estrarre un insieme di classi di analisi dalla specifica del problema (ne parleremo tra poco)
- Ragionare su queste **classi**: quali **attributi** e quali **operazioni** devono fornire?
- Stendere una mappa delle **classi** e delle loro **relazioni**.
- Modellare la dinamica delle classi con i *diagrammi di comportamento*.
- Procedere per **raffinamenti successivi** fino a quando il modello rappresenta efficacemente il dominio del problema.

Come procedere: Progettazione

- Si parte dal modello di analisi che contiene classi abbastanza generiche, e lo si raffina.
- I costrutti più **astratti** di UML vengono trasformati in altri più **concreti** che possono essere implementati in un linguaggio di programmazione OO.
- Finalmente si considerano i vincoli di **piattaforma** e **linguaggio**, e i requisiti **non funzionali**.
- Le classi di analisi si trasformano in classi di progettazione (non c'è corrispondenza 1 a 1)
- Ancora una volta si procede per **raffinamenti successivi**.
- Il risultato è un modello pronto per l'implementazione.

Come estrarre le classi di analisi

- Una classe di analisi modella un concetto o entità del problema: se la specifica dei casi d'uso è buona i concetti basilari sono già in evidenza.
- I candidati più probabili sono nomi che compaiono nella specifica e nella documentazione.
- Una ragione in più per tenere un glossario di progetto: le parole nel glossario sono spesso candidati ideali per diventare classi di analisi.
- Le classi di analisi non sopravviveranno necessariamente alla progettazione.
- Due metodi molto diffusi per trovare le classi di analisi:
  - analisi nome-verbo
  - analisi CRC

Analisi nome-verbo

Si analizza tutta la documentazione disponibile, selezionando nomi e verbi.

- I **nomi**: (es: conto corrente) sono i potenziali candidati per divenire classi o attributi.
- I **'predicati nominali'**: (es: numero del conto corrente) sono i potenziali candidati per divenire classi o attributi.
- I **verbi**: (es: aprire) sono potenziali candidati a divenire responsabilità di classe.

Notate che ancora non parliamo di UML!

CRC cards

- Le carte CRC sono state proposte da Kent Beck e Ward Cunningham nel 1989 come strumento didattico per insegnare la progettazione Object-Oriented.
- Si tratta di un metodo di brainstorming iterativo che coinvolge sviluppatori, esperti, committenti.
- Servono per definire le classi principali e le loro interazioni.
- **C**lasse, **R**esponsabilità, **C**ollaborazione:
  - **Classe:** gli oggetti più importanti
  - **Responsabilità:** compiti principali da eseguire
  - **Collaborazioni:** altri oggetti che cooperano per soddisfare una responsabilità
- Si usano post-it divisi in tre sezioni chiamate proprio in questo modo.

CRC cards: esempio

Class name:	**Superclass:**	**Subclasses:**
Responsabilities	**Collaborations**	

CRC cards: esempio

Class name: Padrone	**Superclass:** Persona	**Subclasses:**
Responsabilities	**Collaborations**	
Invita	Invito, Persona, Lista	
Compra	Denaro, Negozio, Cibo, ...	
Pulisce_casa	Spugna, Straccio, ...	

Le schede posizionate su un piano, e la loro vicinanza fisica rispecchia quella logica.

Esercizio Birreria

Compilare e discutere le **schede CRC** per rappresentare il seguente dominio:

- Una birreria è frequentata dai clienti e dallo staff. In particolare, lo staff raccoglie gli ordini e consegna le birre. Si paga alla cassa (e lo staff può dare il resto se necessario). Il gestore del pub si occupa, oltre che del servizio, anche di controllare la disponibilità di ogni birra in frigo e, se necessario, aggiungerne altre.

Esercizio Birreria

Compilare e discutere le **schede CRC** per rappresentare il seguente dominio:

- Una birreria è frequentata dai clienti e dallo staff. In particolare, lo staff raccoglie gli ordini e consegna le birre. Si paga alla cassa (e lo staff può dare il resto se necessario). Il gestore del pub si occupa, oltre che del servizio, anche di controllare la disponibilità di ogni birra in frigo e, se necessario, aggiungerne altre.

Esercizio Birreria: CRC staff

Class: **Staff**	
Responsibility	*Collaborators*
Prende gli ordini	
Prende le birre	
Serve le birre	
Prende i soldi	
Dà il resto	

Esercizio Birreria: CRC staff

Class: **Staff**	
Responsibility	*Collaborators*
Prende gli ordini	Ordine Cliente
Prende le birre	Birra Frigo
Serve le birre	Birra Cliente
Prende i soldi	Cliente Cassa
Dà il resto	Cassa Cliente

Esercizio Birreria: CRC gestore

Class: **Gestore** – *Superclass*: **Staff**	
Responsibility	*Collaborators*
Controlla birre in frigo	Frigo
Aggiunge una birra in frigo	Birra Frigo

Esercizio Birreria: CRC cliente

Class: **Cliente**	
Responsibility	*Collaborators*
Ordina la birra	Staff Ordine
Riceve la birra	Staff Birra
Paga	Staff
Riceve il resto	Staff

Esercizio Birreria: CRC birra e frigo

Class: **Birra**	
Responsibility	*Collaborators*
Conosce il proprio prezzo	

Class: **Frigo**	
Responsibility	*Collaborators*
Conosce la disponibilità per ogni tipo di birra	
Permette di aggiungere una birra	

- Altri collaboratori?

Esercizio Elezioni

Compilare e discutere le schede CRC per rappresentare il seguente dominio:

- Un sistema di voto da remoto prevede due modalità: touch-screen o via tastiera. Ogni sistema permette di esprimere il proprio voto per le elezioni comunali, regionali e nazionali. Gli elettori per votare inseriscono nel sistema un codice che gli è stato fornito in precedenza e, se il codice è corretto e non ancora usato, il sistema mostra a video le possibili scelte. E l'elettore esprime la sua preferenza.

Esercizio Elezioni

Class: **SistemaDiVoto** – *Subclasses*: **Touch-Screen, Tastiera**	
Responsibility	*Collaborators*
Conosce le elezioni in corso	
Mostra le elezioni in corso	
Valida il codice di voto	
Accetta un voto	

- Chi sono i collaborator?
- Qualche altra responsability?

Esercizio Elezioni

Class: **SistemaDiVoto** – *Subclasses*: **Touch-Screen, Tastiera**	
Responsibility	*Collaborators*
Conosce le elezioni in corso	
Mostra le elezioni in corso	
Valida il codice di voto	
Accetta un voto	Voto
Permette di caricare le elezioni	Elezione

- Altri collaboratori?

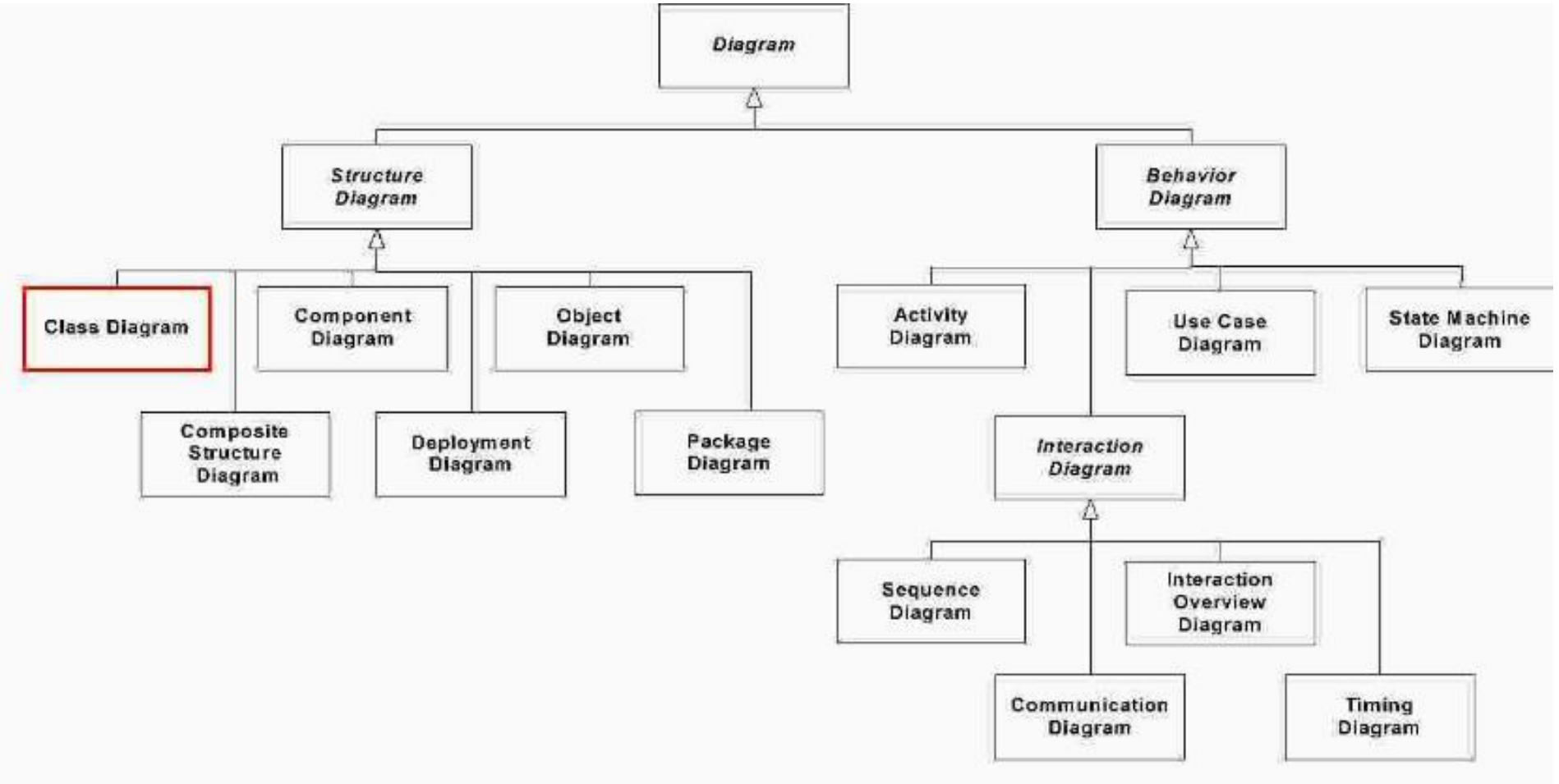
Esercizio Elezioni

Class: **Voto**	
Responsibility	*Collaborators*
Sa per quale elezione è stato espresso	
Sa quale preferenza è stata data	

Class: **Elezione** – *Subclasses*: **Comunale, Regionali, Nazionale**	
Responsibility	*Collaborators*
Conosce i candidati	

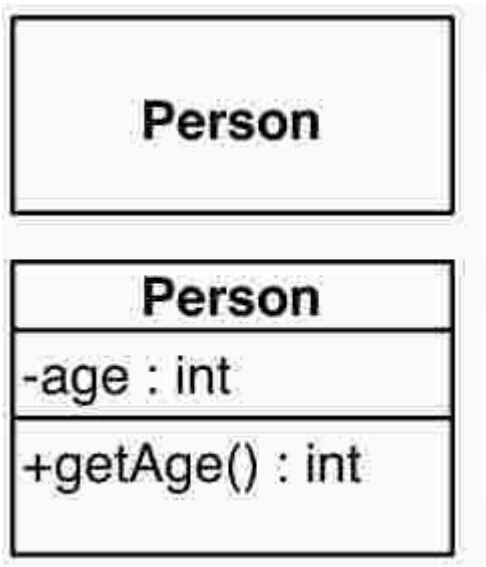
- Altri collaboratori o classi?

Diagramma delle classi



L’icona di Classe in UML

- Ha una rappresentazione grafica in forma di un rettangolo diviso in tre parti
- Le classi possono avere fino a 3 slot:
  - nome e l’eventuale stereotipo (in UpperCamelCase - obbligatorio)
  - attributi (in lowerCamelCase - opzionale)
  - operazioni (in lowerCamelCase - opzionale)
- La stessa classe può apparire con diverse quantità di ornamenti in diagrammi diversi



Attributi e Operazioni: Signature

Attributo:

visibilità nome molteplicità:tipo=valoreIniziale

Operazione:

visibilità nome (nomeParametro:tipoParametro, . . .):
tipoRestituito

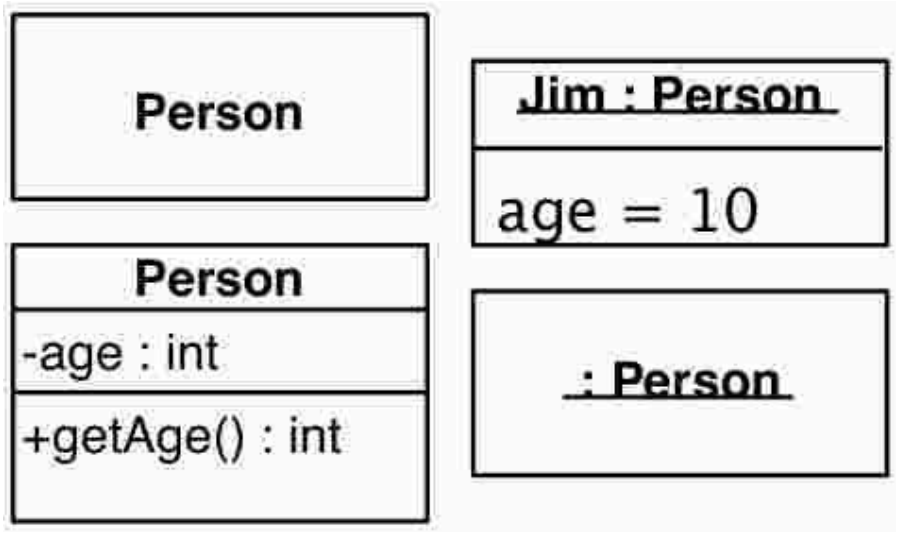
- Solo il nome è obbligatorio
- Le **classi di analisi** solitamente contengono solo quelli più importanti (quelli che risultano evidenti dall'analisi del dominio), e spesso specificano solo il nome.
- Assegnare un valore iniziale in una classe di analisi può evidenziare i vincoli di un problema.
- Le **classi di progettazione** forniscono una specifica completa (implementabile) della classe e dei suoi attributi.

Attributi e Operazioni: Tipi di Visibilità

+	public	ogni elemento che può accedere alla classe può anche accedere a ogni suo membro con visibilità pubblica
-	private	solo le operazioni della classe possono accedere ai membri con visibilità privata
#	protected	solo le operazioni appartenenti alla classe o ai suoi discendenti possono accedere ai membri con visibilità protetta
~	package	ogni elemento nello stesso package della classe (o suo sottopackage annidato) può accedere ai membri della classe con visibilità package

L’icona di Oggetto in UML

- Le istanze delle classi (oggetti) hanno una notazione molto simile
- Il titolo degli oggetti è sottolineato e del tipo ’nome: classe’, con nome opzionale.
- Gli oggetti non hanno uno slot per le operazioni, possono definire valori per gli attributi.



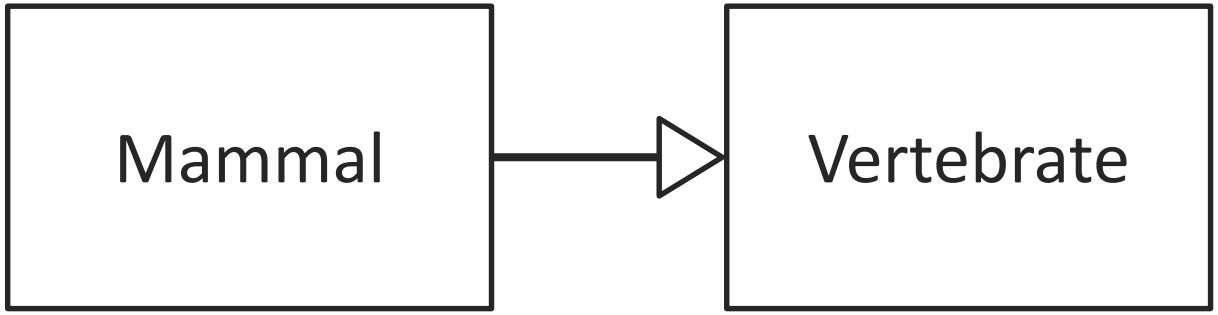
Relazione tra Classe ed Oggetto

- Si tratta di una dipendenza con stereotipo «instanceOf» (l'oggetto «istanzia» la classe).

Relazioni tra classi

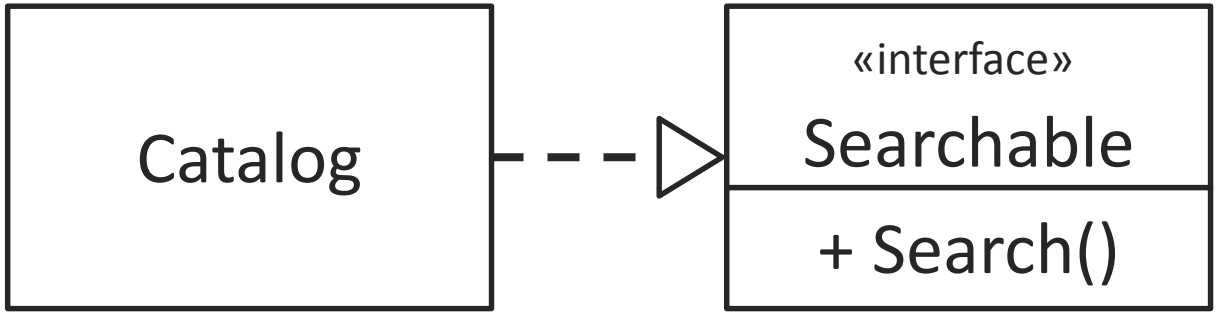
- Generalizzazione 
- Realizzazione 
- Associazione
- Dipendenza 
- Aggregazione 
- Composizione 

Generalizzazione



- Relazione tassonomica tra un elemento più generale e uno che lo specifica.
- La freccia parte dall'elemento specifico e punta verso quello più generale.
- Si tratta dell'ereditarietà in UML.
- Tra tutte le relazioni, questa è la più forte e vincolante.

Realizzazione

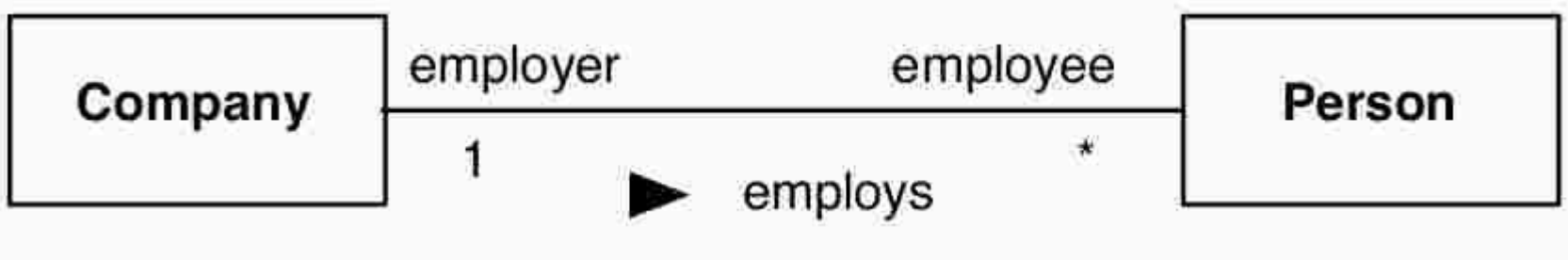


- Si tratta di una relazione semantica in cui il fornitore presenta una specifica, e il cliente la realizza (implementandola e eseguendola).
- L'esempio canonico di realizzazione è quello in cui il fornitore è un'interfaccia, e il cliente è la classe che la implementa.

Associazione

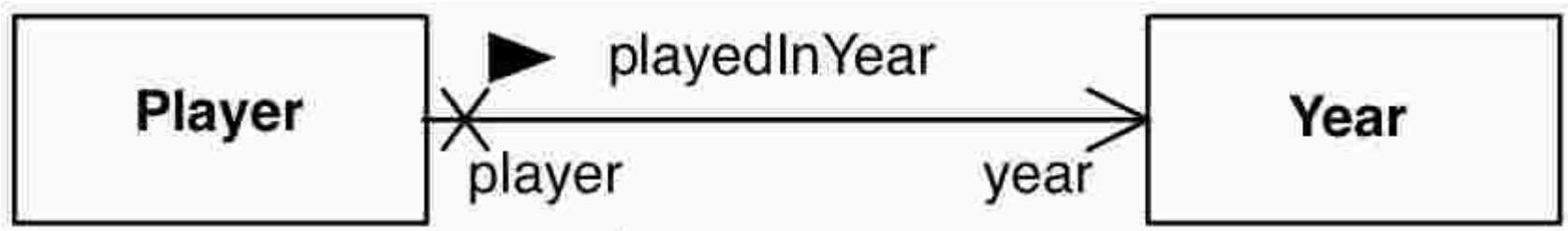
- Si tratta del tipo di relazione più generico: indica solo l'esistenza di collegamenti (link) tra le istanze delle classi.
- Rappresenta l'abilità di un'istanza di mandare messaggi a un'altra istanza.
- Può coinvolgere più di due classi e la stessa classe più di una volta.
- Tra le relazioni è anche la più flessibile e la meno vincolante.

Associazione: ornamenti



- Nome: opzionale.
- Triangolo direzionale: opzionale. Specifica la direzione in cui leggere l'associazione (aumenta la leggibilità).
- Ruoli: opzionali a ciascun estremo.
- Molteplicità: opzionale a ciascun estremo.

Associazione: Navigabilità

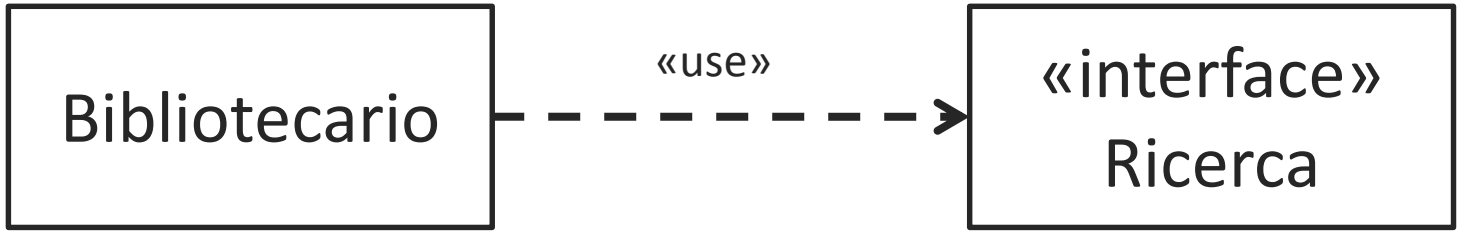


- Specifica se gli altri estremi dell'associazione possono sapere a quali istanze sono associati.
- La freccia indica navigabilità.
- La croce indica assenza di navigabilità.
- La mancanza di entrambe significa navigabilità non specificata (tipico della fase di analisi).
- Un oggetto di tipo Player sa in quali anni ha giocato, un oggetto di tipo Year non sa quali giocatori giocarono quell'anno.

Navigabilità: in pratica...

- In pratica, si tende a non specificare la navigabilità di ogni estremo di ogni relazione.
- Un metodo diffuso è il seguente:
  - non si usano le croci
  - l'assenza di frecce indica navigabilit$\mathbb{H}$ in entrambe le direzioni
  - una freccia indica navigabilità in quella direzione e assenza di navigabilità nell'altra
- Doppia navigabilità risulta indistinguibile da navigabilità non specificata, ma non è un problema in pratica

Dipendenza



- Una dipendenza è una forma più debole di relazione: tra un cliente ed un fornitore di servizio (es. tra classi e operazioni).
- Due ruoli: **supplier** (fornitore) e **client** (cliente); entrambi possono essere insiemi di elementi.
- La freccia va dal cliente verso il fornitore (e può essere indicato il tipo di dipendenza).
- Una dipendenza significa che il cliente richiede il fornitore per la propria specifica o implementazione.
- Il cliente dipende strutturalmente o semanticamente dal fornitore, e se la specifica del fornitore cambia può cambiare anche quella del cliente.

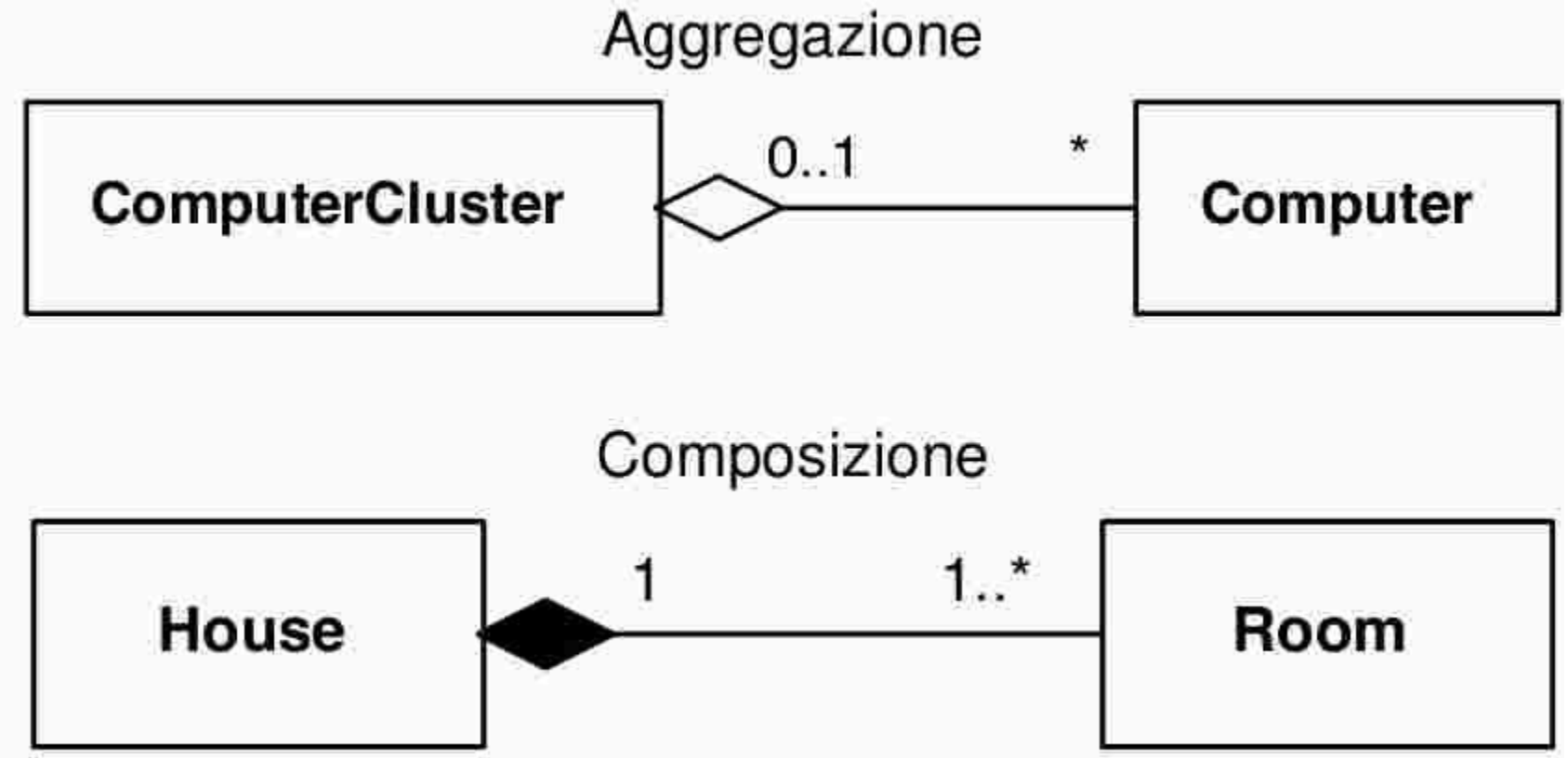
Aggregazione e Composizione

Si tratta di particolari forme di associazione che rappresentano la relazione whole-part (tutto-parte) tra un aggregato e le sue parti.

- **Aggregazione:** relazione poco forte (a. le parti esistono anche senza il tutto; b. Le parti possono appartenere a più aggregazioni. Es. i computer e il loro cluster).
- **Composizione:** relazione molto forte (le parti dipendono dal tutto e non possono esistere al di fuori di esso; es. le stanze e la casa).

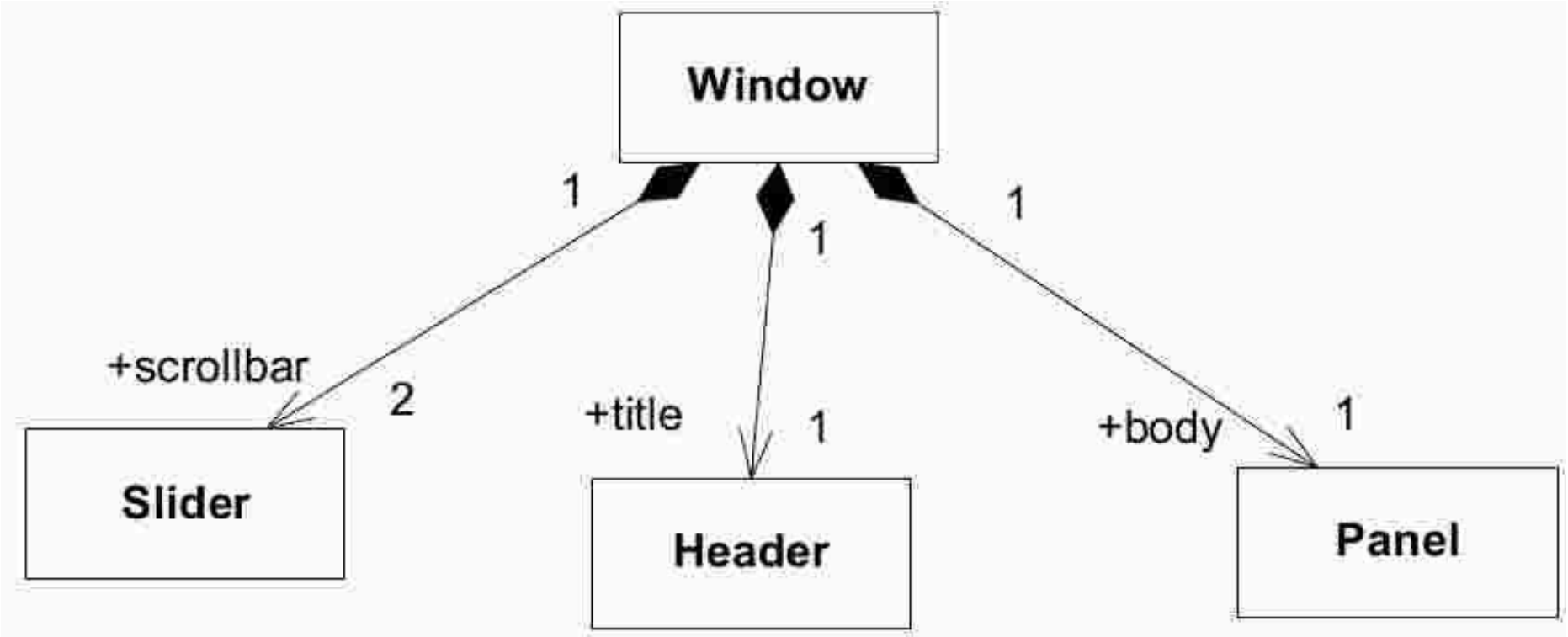
Non è sempre semplice capire quale delle due modella meglio una situazione.

Aggregazione e Composizione: Notazione 1



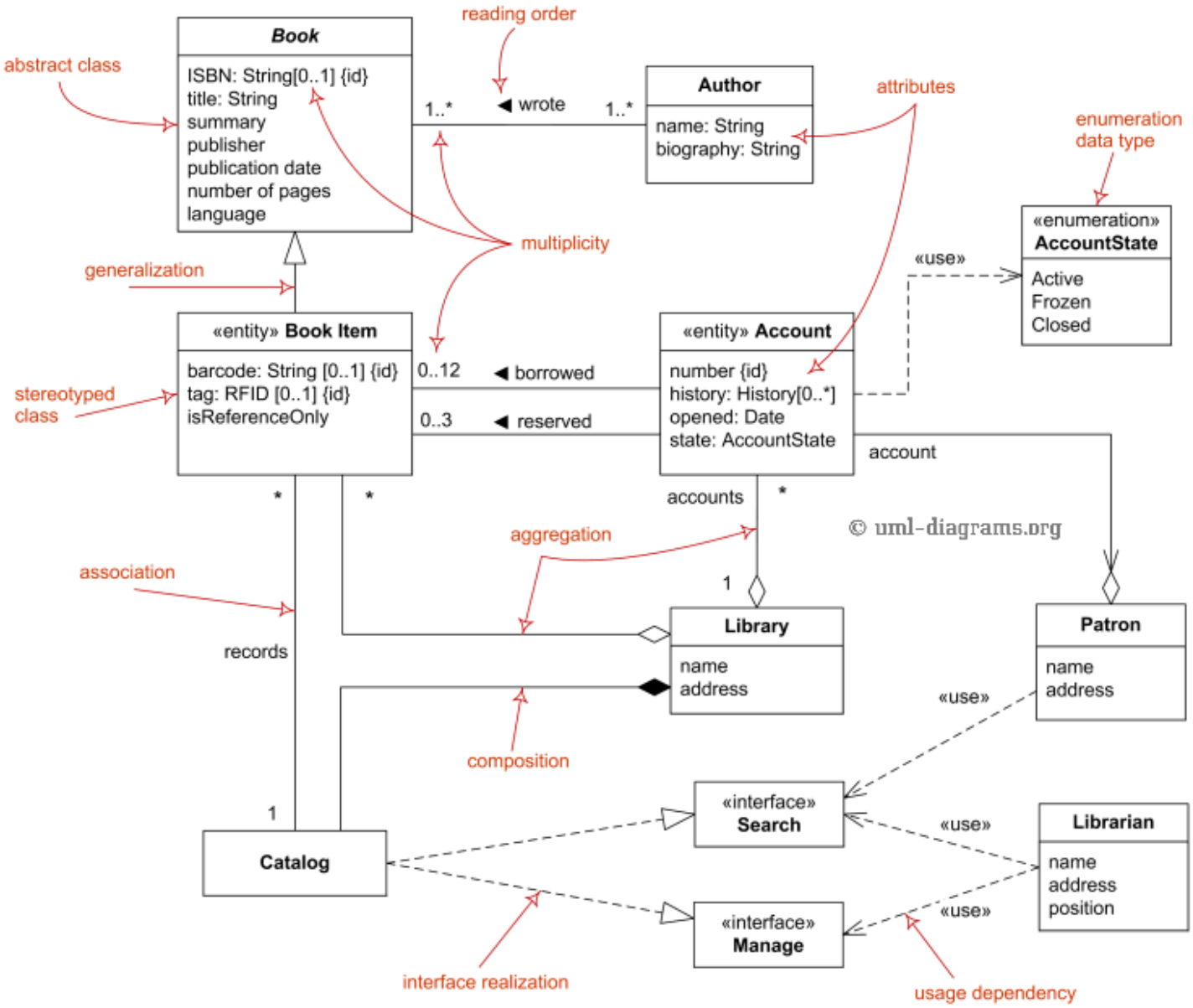
Entrambe sono rappresentate da linee con un diamante (pieno o vuoto) vicino alla classe che rappresenta l'intero

Aggregazione e Composizione: Notazione 2



Aggregazione e composizione possono essere combinate con le altre notazioni per le associazioni.

Diagramma delle classi: esempio



Esercizio

Rappresentare il seguente testo tramite un diagramma UML delle classi:

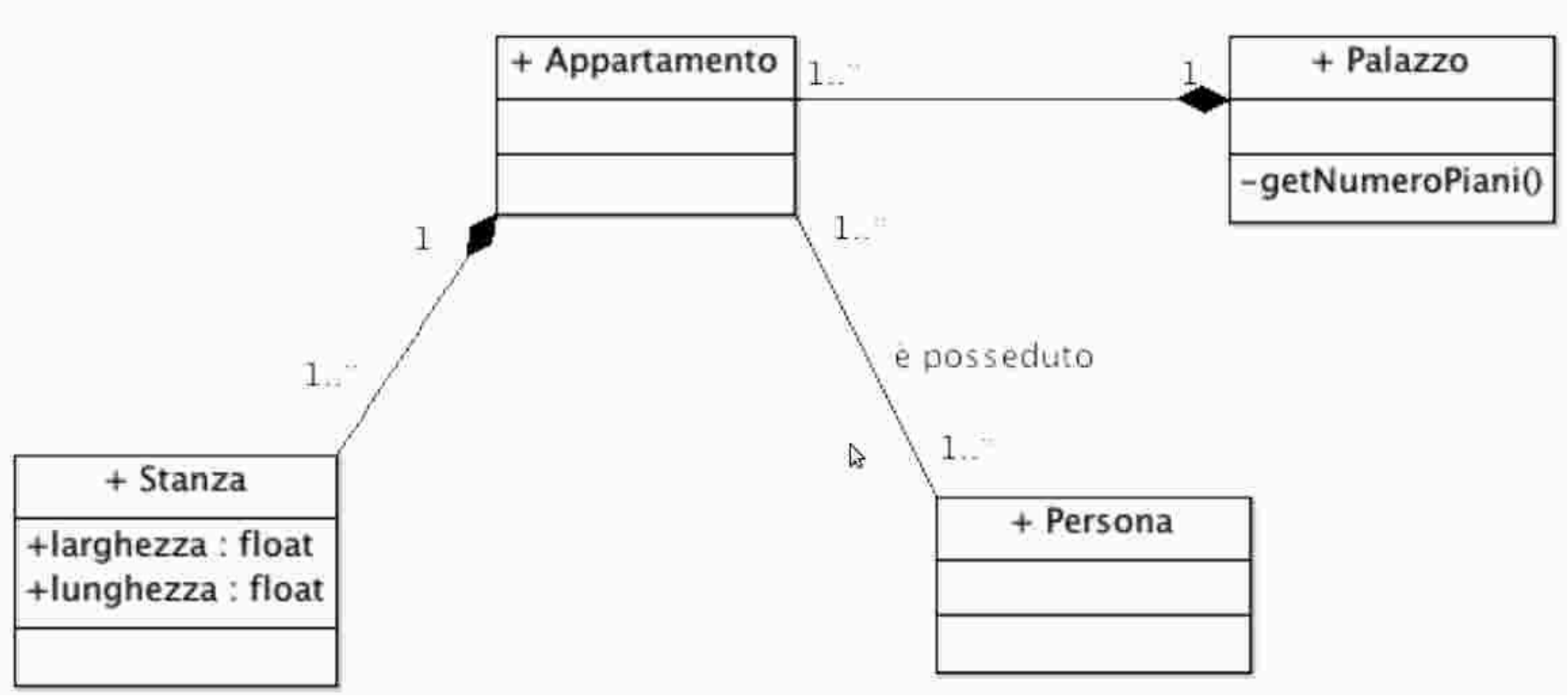
- Un appartamento è composto da una o più stanze, ciascuna delle quali ha una lunghezza e una larghezza. Un appartamento è posseduto da uno o più persone. Un palazzo ha un certo numero di piani ed è composto da uno o più appartamenti.

Esercizio

Rappresentare il seguente testo tramite un diagramma UML delle classi:

- Un **appartamento** è composto da una o più **stanze**, ciascuna delle quali ha una lunghezza e una larghezza. Un appartamento è posseduto da uno o più **persone**. Un **palazzo** ha un certo numero di **piani** ed è composto da uno o più appartamenti.

Esercizio



- Aggregazione o composizione?
- Aggiungere la classe Piano? Come cambia il diagramma?

Esercizio

Disegnare un diagramma delle classi di analisi per modellare il dominio:

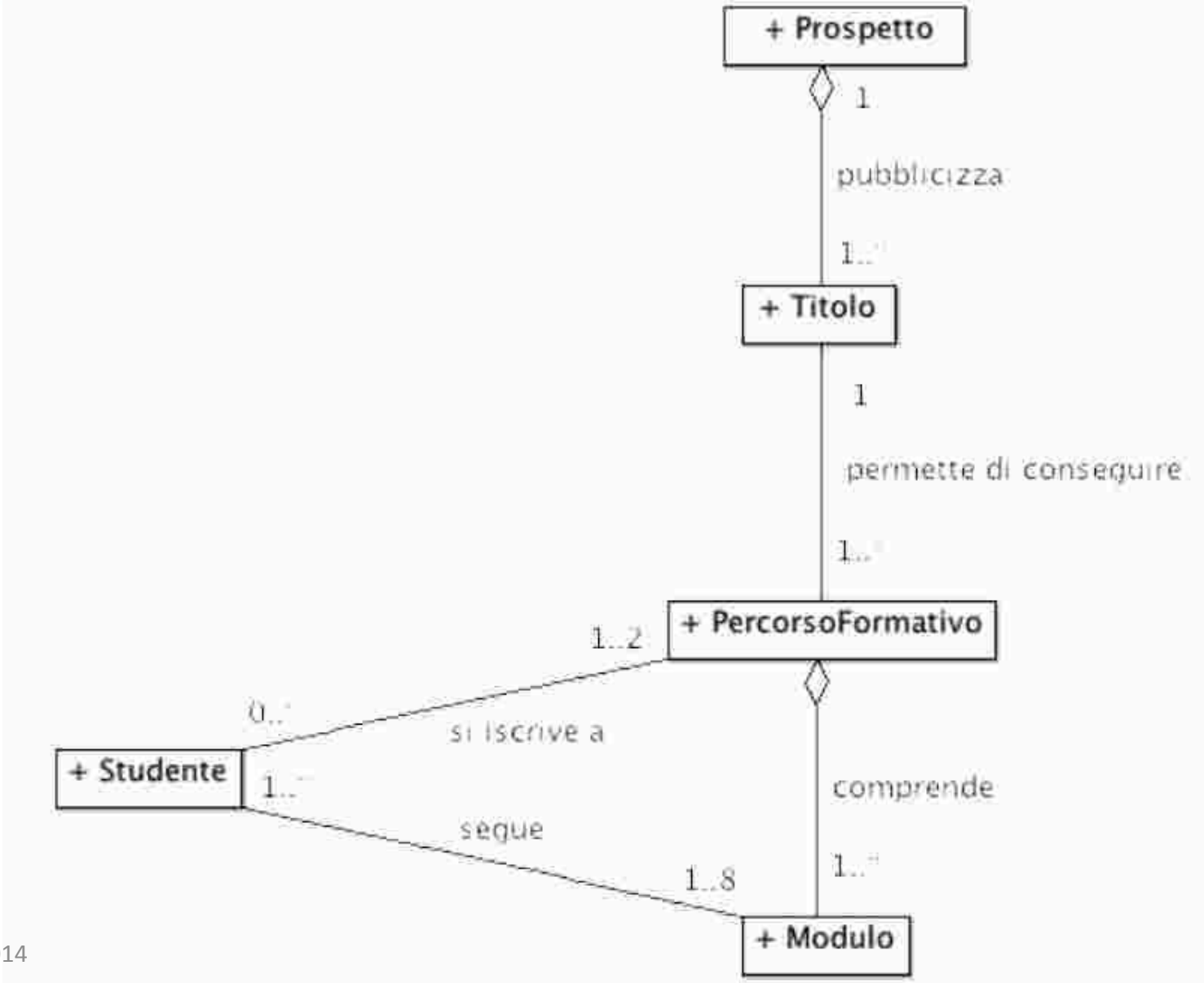
- De Montfort University (DMU) offre percorsi formativi ciascuno dei quali porta al conseguimento di un titolo di riconoscimento.
- Ogni titolo di riconoscimento è pubblicizzato nel prospetto informativo di DMU
- Ogni percorso comprende differenti moduli
- Gli studenti di un percorso seguono fino a 8 moduli all'anno
- Alcuni titoli sono 'congiunti', ad esempio uno studente può iscriversi a due differenti percorsi (come 'contabilità' e 'ragioneria')

Esercizio

Disegnare un diagramma delle classi di analisi per modellare il dominio:

- De Montfort University (DMU) offre **percorsi formativi** ciascuno dei quali porta al conseguimento di un titolo di riconoscimento.
- Ogni **titolo** di riconoscimento **è pubblicizzato** nel **prospetto** informativo di DMU
- Ogni percorso **comprende** differenti **moduli**
- Gli **studenti** di un percorso **seguono** fino a 8 moduli all'anno
- Alcuni titoli sono 'congiunti', ad esempio uno studente **può iscriversi** a due differenti percorsi (come 'contabilità' e 'ragioneria')

Esercizio


+ Prospetto
1
pubblicizza
1..*
+ Titolo
1
permette di conseguire
1..*
+ PercorsoFormativo
1..2
0..*
si iscrive a
+ Studente
1..*
segue
1..8
comprende
1..*
+ Modulo

Esercizio

Completare il diagramma per rappresentare (parte 1):

- La DMU è composta di 6 Facoltà
- Ogni facoltà definisce un numero di materie (‘contabilità’, ‘ragioneria’, etc.) di cui si occupano moduli differenti e che sono insegnate in percorsi differenti.

E successivamente (parte 2):

- Il consiglio di Facoltà è composto da studenti e da staff accademico o ammistrativo
- Lo staff accademico insegna un numero arbitrario di moduli
- Lo staff accademico supervisiona diversi studenti, ciascuno dei quali segue un percorso formativo
- Alcuni rappresentanti dello staff amministrativo sono consiglieri ma non insegnano

Esercizio

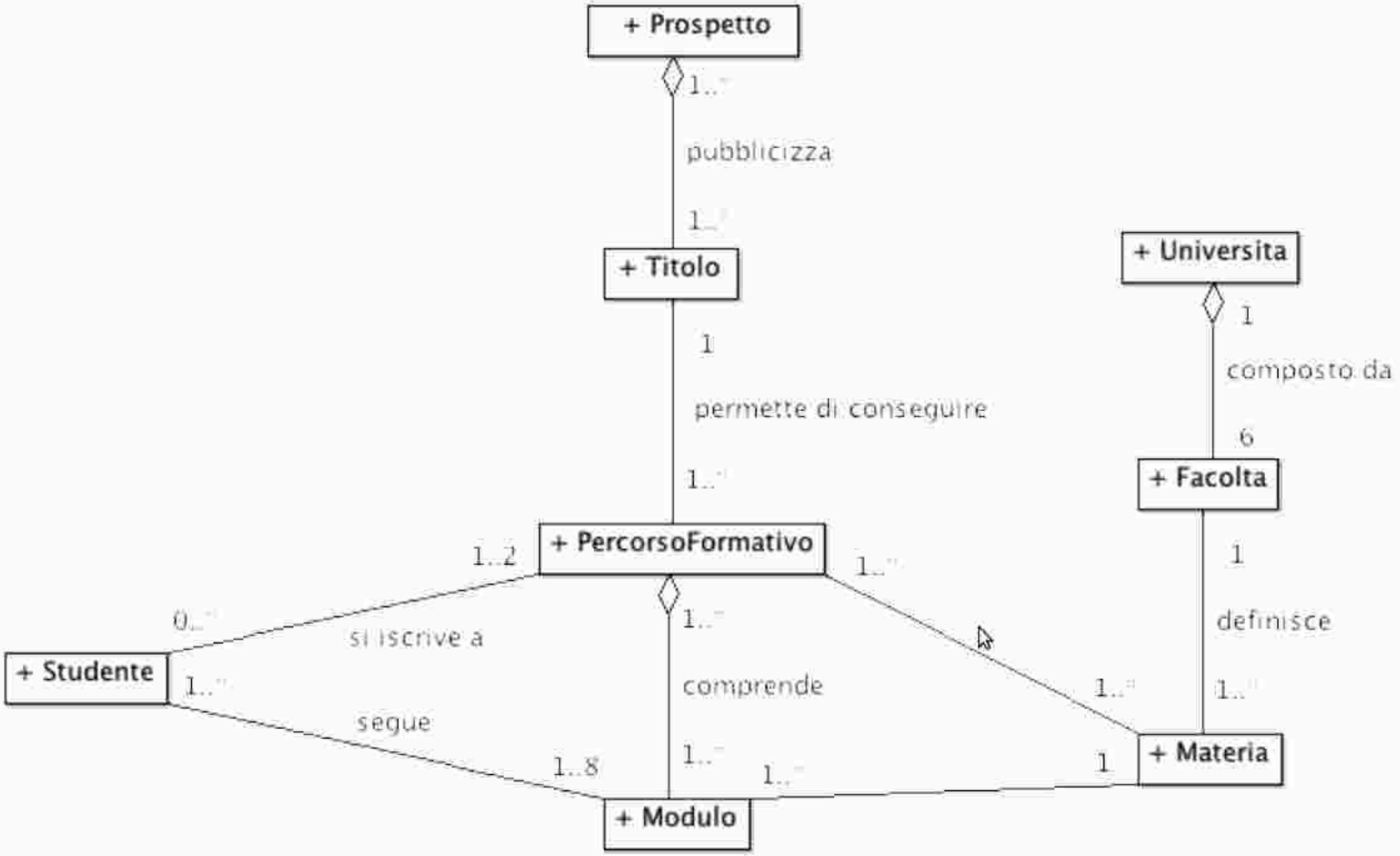
Completare il diagramma per rappresentare (parte 1):

- La DMU **è composta** di 6 **Facoltà**
- Ogni facoltà **definisce** un numero di **materie** (‘contabilità’, ‘ragioneria’, etc.) di cui si occupano moduli differenti e che sono insegnate in percorsi differenti.

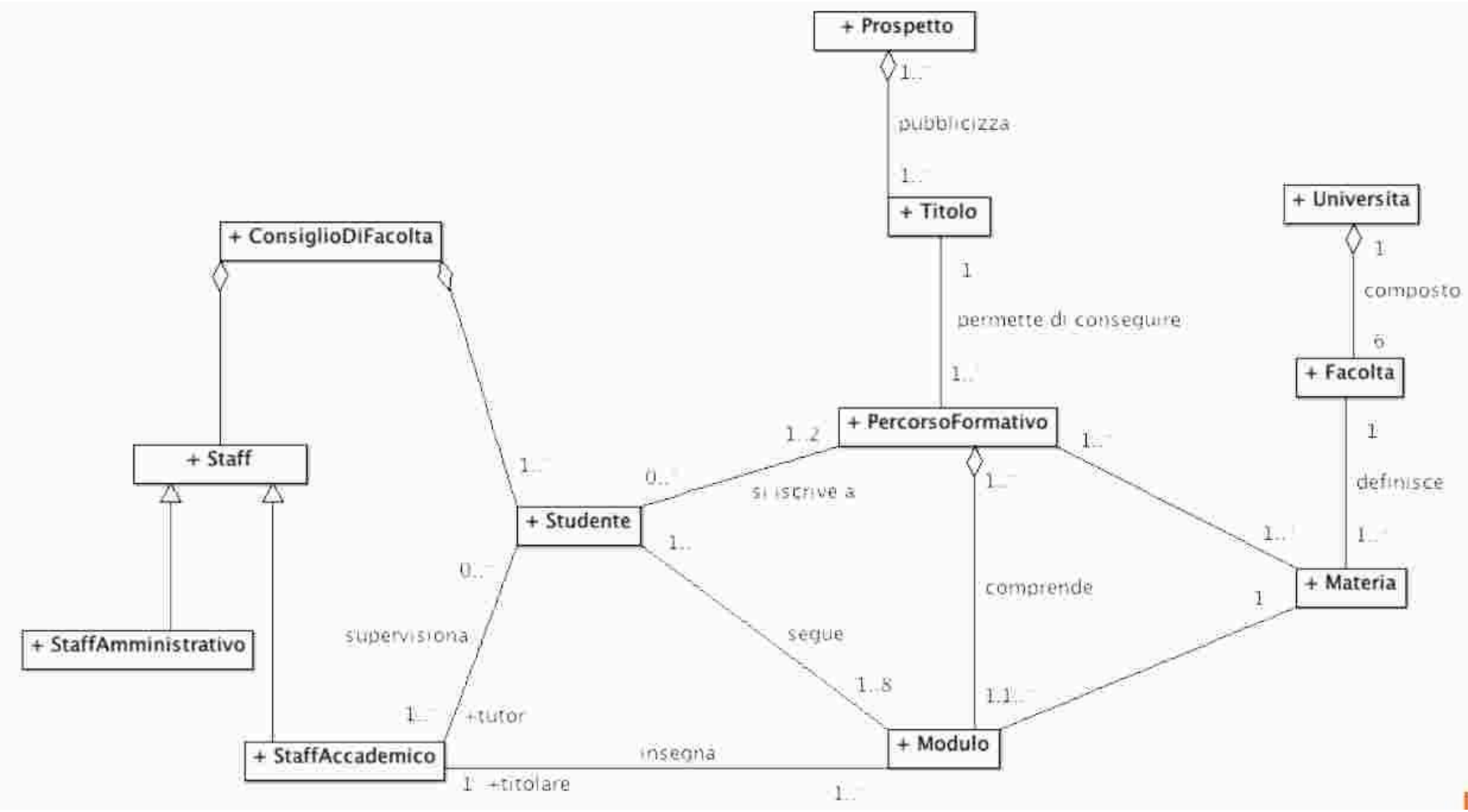
E successivamente (parte 2):

- Il **consiglio di Facoltà è composto** da **studenti** e da **staff accademico** o **ammistrativo**
- Lo staff accademico **insegna** un numero arbitrario di moduli
- Lo staff accademico **supervisiona** diversi studenti, ciascuno dei quali segue un percorso formativo
- Alcuni rappresentanti dello staff amministrativo sono consiglieri ma non insegnano

Esercizio


+ Prospetto
1..*
pubblicizza
1..*
+ Titolo
1
permette di conseguire
1..*
+ Universita
1
composto da
6
+ Facolta
1
definisce
1..*
1..2
+ PercorsoFormativo
1..*
0..*
si iscrive a
1..*
+ Studente
1..*
segue
1..8
1..*
comprende
1..*
1..*
1
+ Materia
1..*
+ Modulo

Esercizio


+ Prospetto
1..
pubblicizza
1..
+ Titolo
1
permette di conseguire
1..
+ Universita
1
composto
6
+ Facolta
1
definisce
1..
+ ConsiglioDiFacolta
+ Staff
1..
0..
si iscrive a
1..2
+ PercorsoFormativo
1..
1..
+ Studente
1..
0..
supervisiona
segue
comprende
1..8
1..1..
1
+ Materia
+ StaffAmministrativo
1..
+tutor
+ StaffAccademico
insegna
1 +titolare
1..
+ Modulo

Esercizio

- Un sistema di gestione di una discoteca permette di organizzare feste private. Una festa ha un ospitante (cioè la persona che fa gli inviti), tanti invitati, un disk-jockey, alcuni camerieri, un buttafuori. Il sistema deve permettere di invitare persone, ricevere le risposte (di accettazione o rifiuto), definire un menù-bar di cocktail e bevande, una playlist. I camerieri servono le bevande. Il disk-jockey gestisce la playlist, e accetta richieste anche durante la festa. Il buttafuori controlla che alla festa entrino solo persone con invito.

Disegnare un diagramma delle classi che modella il precedente dominio

Esercizio

- La macchina del pane permette di preparare deliziose pagnotte, tonde o “a bauletto”. Usare la macchina è semplice: basta estrarre il cestello dalla macchina, aggiungere gli ingredienti (facendo attenzione a mettere prima i liquidi e poi gli altri ingredienti, ed evitando che il lievito entri a contatto con il sale), riposizionare il cestello, selezionare il programma e avviare. A fine cottura la macchina mantiene il pane caldo per un’ora. Entro quell’ora bisogna estrarlo e farlo asciugare, altrimenti si indurisce. Prima di affettarlo è necessario aspettare un’altra mezzora.
- Tutte le pagnotte sono preparate con gli stessi ingredienti di base: liquido (acqua o latte), farina (bianca o integrale), sale, zucchero e lievito (fresco o in polvere). Inoltre le pagnotte possono essere arricchite con ingredienti extra, da aggiungere dopo il segnale acustico emesso dalla macchina: olive, semi o pomodori secchi.
- Tutte le pagnotte inoltre sono preparate con lo stesso ciclo, ad eccezione del pane integrale che prevede un pre-riscaldamento della macchina che impasta, si ferma per due fasi di lievitazione e cuoce.

Disegnare un diagramma delle classi che modella il precedente dominio