

Esercitazione

Vincenzo Gervasi, Laura Semini
Ingegneria del Software
Dipartimento di Informatica
Università di Pisa

ATTENZIONE

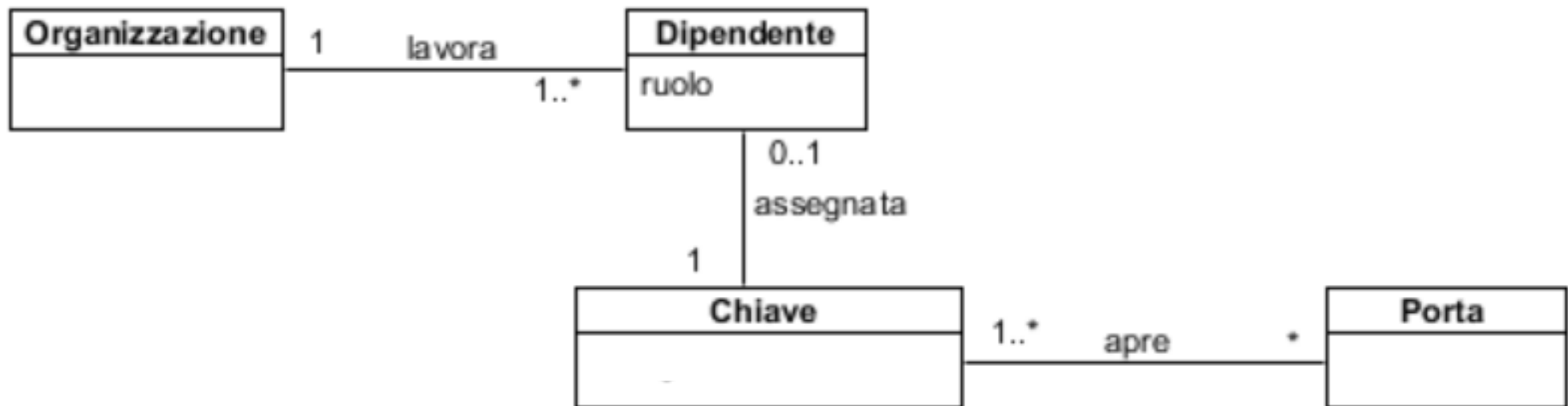
- Questi lucidi servono come base per discussione: non necessariamente i diagrammi riportati sono completi e/o corretti

Chiavi magnetiche

Chiavi magnetiche

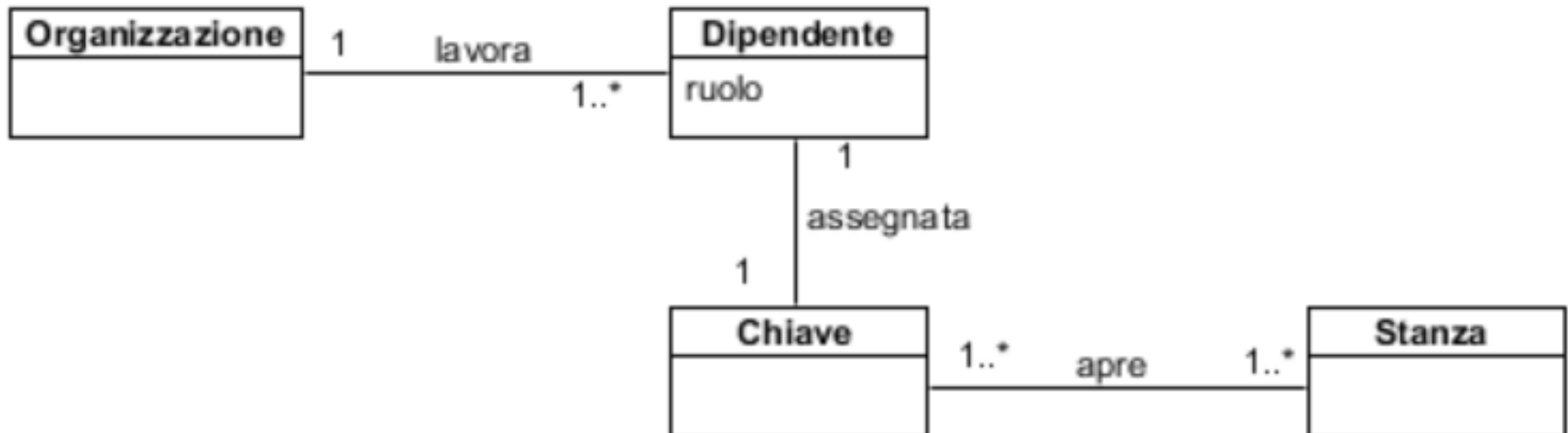
- Per motivi di sicurezza, un'organizzazione ha deciso di realizzare un sistema secondo il quale a ogni dipendente è assegnata una chiave magnetica per accedere (aprire) determinate stanze. I diritti di accesso dipenderanno in generale dalla posizione e dalle responsabilità del dipendente. Quindi sono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

Soluzione ragionevole (si considerano anche le chiavi non ancora assegnate)

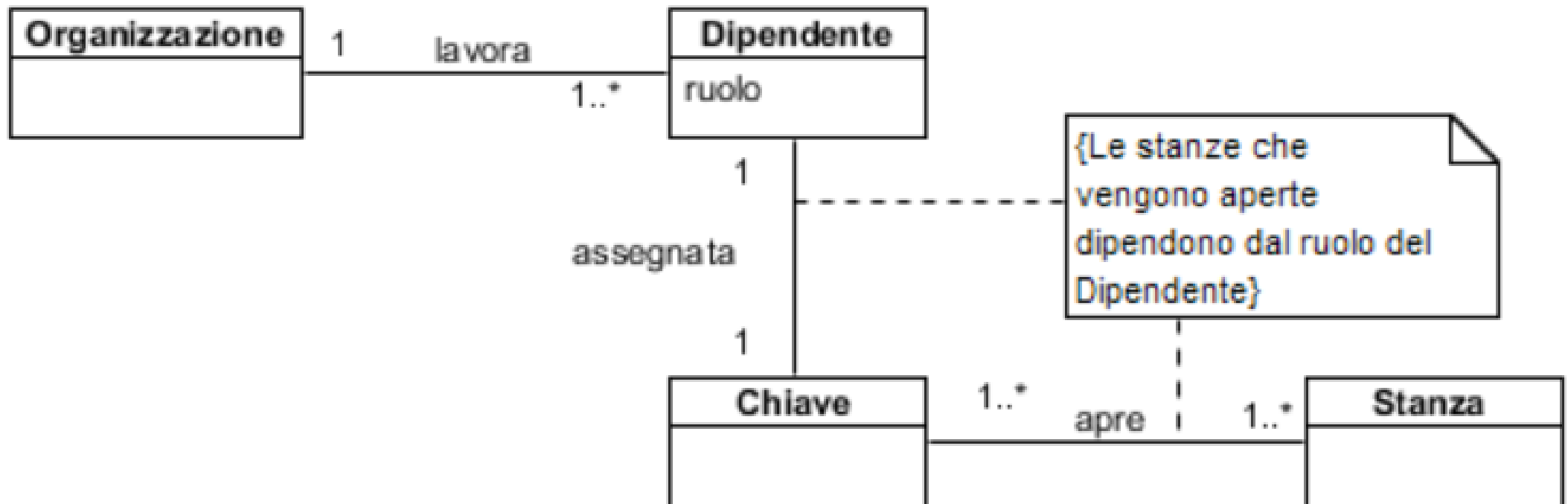


Soluzione ragionevole (si considerano solo le chiavi già assegnate)

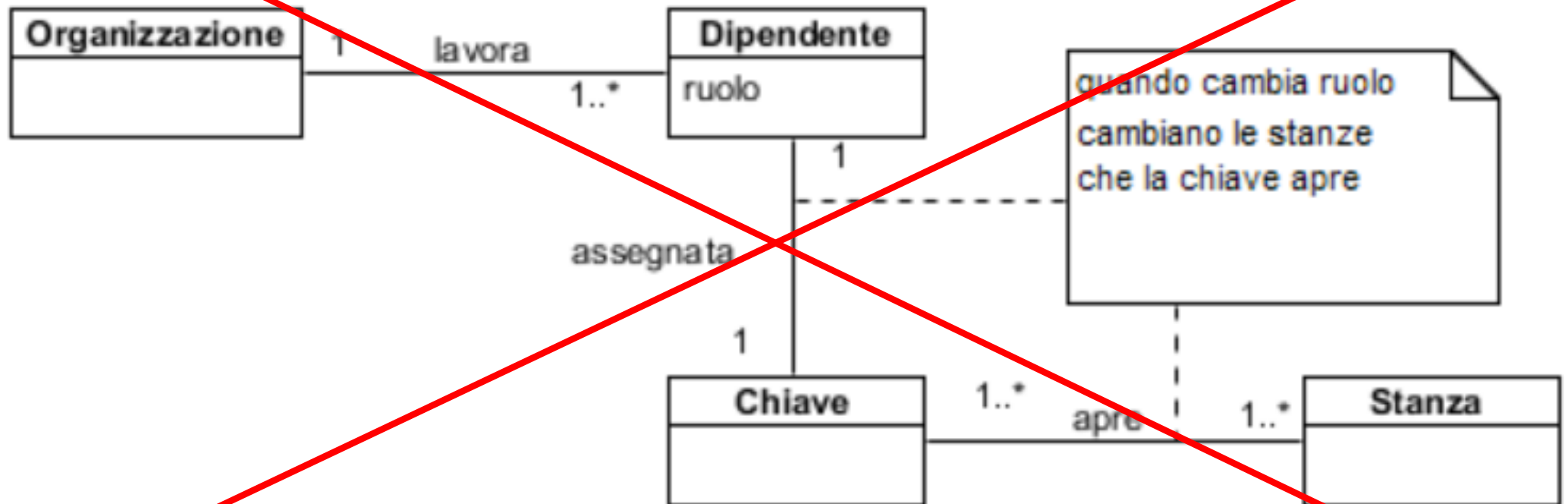
- Notare però come devono cambiare le molteplicità
- Ragionare su porta/stanza



Ruoli-Diritti



Non inserire aspetti del modello dinamico



Grande distribuzione

Ripristino scorte con trasporto tra i punti vendita della catena (classi)

- Si assuma che la catena sia formata da k punti vendita.
- Dare un diagramma delle classi, considerando come classi o attributi tutti e soli i termini seguenti: prodotto, nome, codice, scorta, quantitàDisponibile, quantitàInArrivo, PuntoVendita, Catena, Fornitore.



Grande Distribuzione

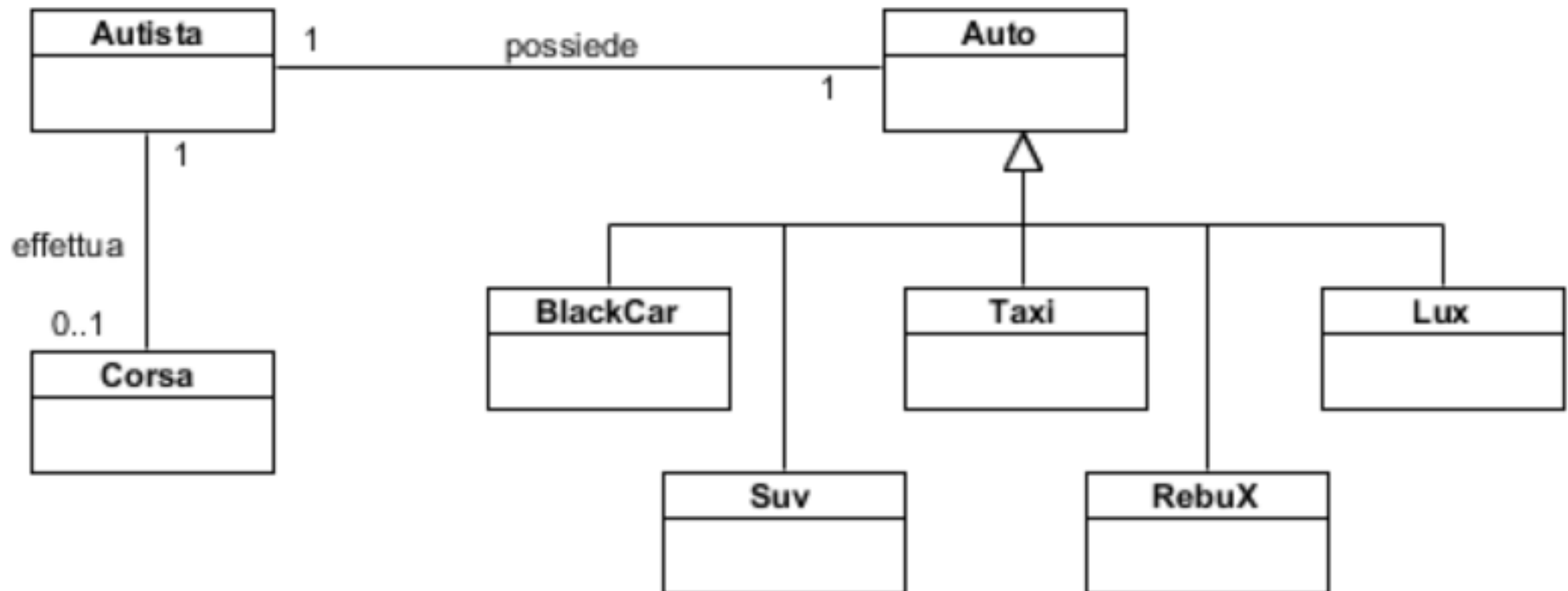
- Altri esercizi svolti nella dispensa omonima.

Rebu

Es 1

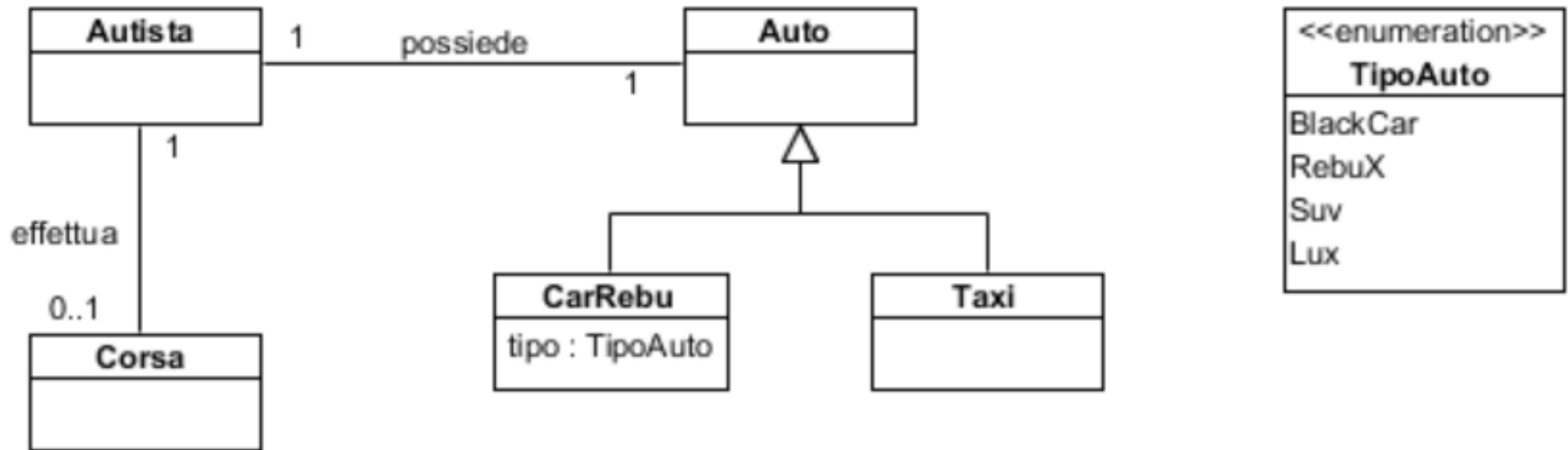
- Si assuma che un autista svolga servizio con la propria auto. Si fornisca il frammento del diagramma delle classi che descrive il dominio, restringendosi all'autista e alle classi che siano in diretta relazione con esso.

Es 1: soluzione accettabile



Es 1: soluzione preferibile

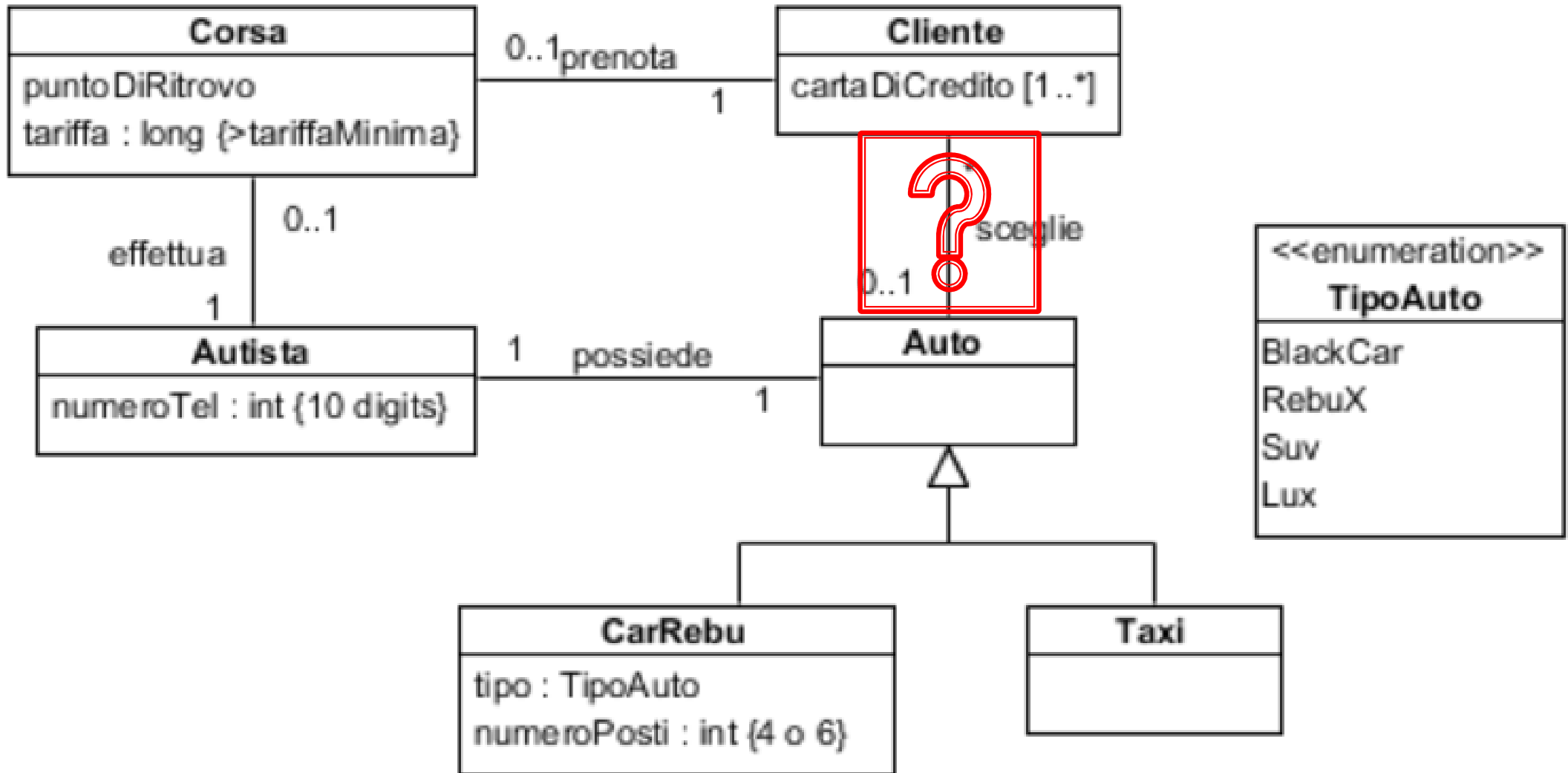
- Solo il taxi può avere proprietà e/o comportamenti sostanzialmente diversi



Es 1 modificato

- Si assuma che un autista svolga servizio con la propria auto. Si fornisca il frammento del diagramma delle classi che descrive il dominio, restringendosi all'autista, alla corsa e alle classi che siano in diretta relazione con queste.
- Permette di documentare le scelte del cliente

Alcuni dettagli in più, necessari in questo caso. (Discutere)

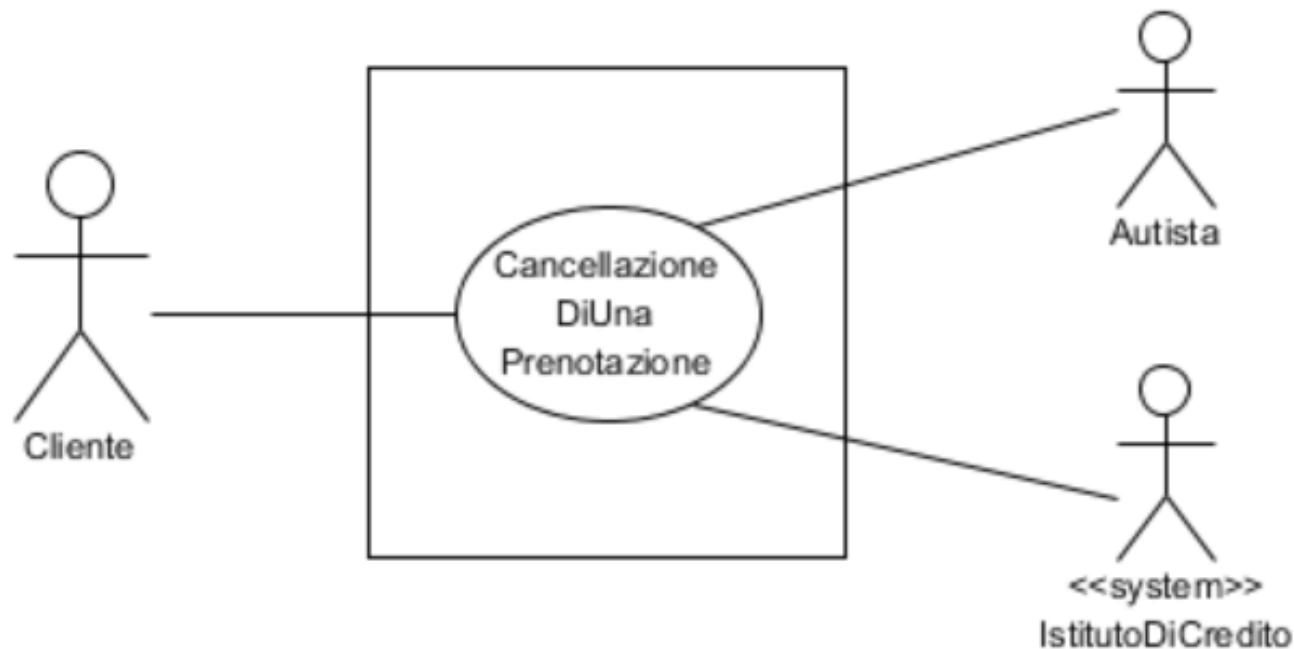


Occorre modellare: il cliente sceglie il tipo, il sistema l'istanza

Rebu (requisiti)

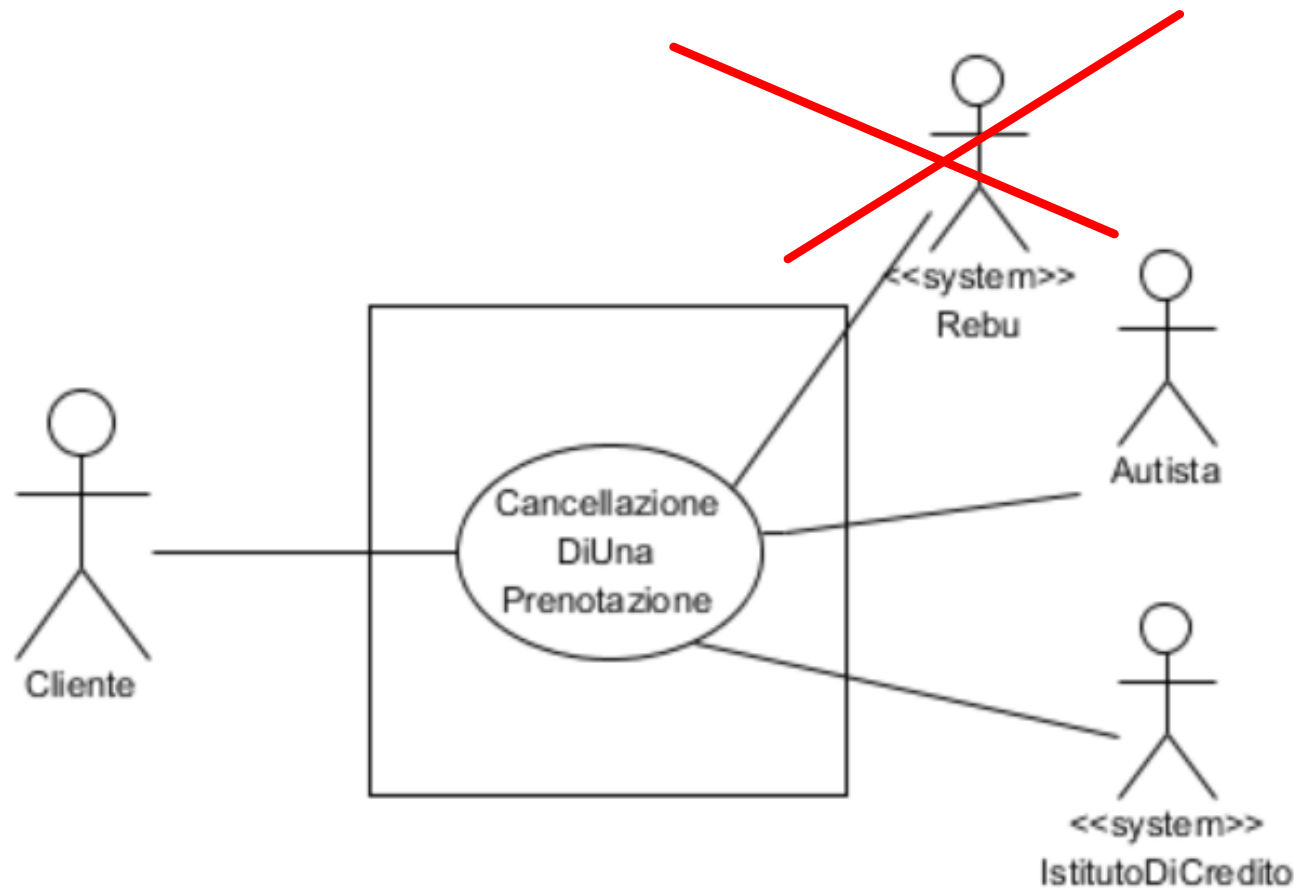
REBU: Esercizio 1

- *Requisiti.* Si forniscano diagramma dei casi d'uso e narrativa, considerando solo il caso: cancellazione di una prenotazione.



REBU: Esercizio 1 con ERRORE

- Il sistema che si sta descrivendo non è MAI un attore



Narrativa: cancellazione di una prenotazione

Caso d'uso: Cancellazione di una prenotazione

Attore primario: Cliente

Attori secondari: Autista, Istituto di credito

Precondizioni: Esiste una prenotazione fatta dal Cliente

Postcondizioni: Prenotazione cancellata, eventualmente penalità addebitata.

Sequenza principale degli eventi:

1. Il Cliente chiede, tramite l'app la cancellazione della prenotazione.
2. Il Sistema informa l'Autista.
3. Se (sono passati più di 5 minuti dalla prenotazione)
 1. Il Sistema addebita \$10 al Cliente.
4. Il sistema cancella la prenotazione.

Sequenze alternative degli eventi: Nessuna

OSSERVAZIONE: Nulla si dice nel testo se parte di (o tutti) quei \$10 vanno all'autista → chiedere al committente

Narrativa (analisi di una diversa soluzione): cancellazione di una prenotazione

Caso d'uso: Cancellazione di una prenotazione

Attore primario: Cliente

Attori secondari: Autista, Istituto di credito

Precondizioni: Esiste una prenotazione fatta dal Cliente

Postcondizioni: Prenotazione cancellata.

Sequenza principale degli eventi:

1. Il Cliente chiede, tramite l'app la cancellazione della prenotazione.
2. Il Sistema informa l'Autista.
3. Il sistema cancella la prenotazione.

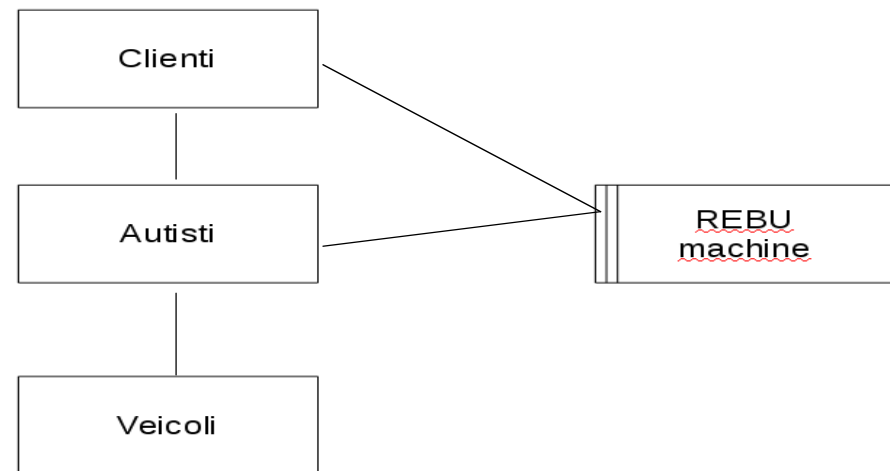
Sequenze alternative degli eventi: sono passati più di 5 minuti dalla prenotazione.

OSSERVAZIONE: Soluzione "sinatticamente" corretta: la sequenza principale porta alla postcondizione, ma non ha senso considerare come caso eccezionale il ritardo nella cancellazione.

REBU: Esercizio 2a

(a) Quali fenomeni condivisi vi sembrano necessari fra questi domini per descrivere i requisiti di REBU?

- A. Clienti – Machine: nome, telefono, carte di credito (ok perché decide su quale addebitare corsa)
- B. Clienti – Autisti: incontro
- C. Clienti – Veicoli: scelta di tipo
- D. Autisti – Veicoli: possesso
- E. Autisti – Machine: nome, **accredito** (meglio tramite banca)



REBU: Esercizio 2b

- (a) ..
- (b) si completi il diagramma in modo da poter esprimere il requisito che un cliente va informato delle tariffe applicate a una corsa.

Aggiungere Designed Domain Tariffe e collegare R a Clienti e Tariffe

REBU: Esercizio 3

Si consideri il seguente caso d'uso: PRENOTAZIONE TRAMITE APP

Attore primario: Cliente

Attori secondari: Autista, Istituto di credito

Precondizioni: Cliente autenticato tramite l'app

Postcondizioni: Prenotazione effettuata oppure richiesta al Cliente di riprovare più tardi.

Sequenza principale degli eventi:

1. Il Cliente chiede, tramite l'app, l'invio di un'auto, specificando tipo, indirizzo, orario e carta di credito con cui pagare la corsa.
2. Il Sistema trasmette la richiesta a tutti gli autisti in servizio, dotati di un'auto del tipo richiesto, e che non siano già assegnati ad altre corse nell'orario indicato
3. Il sistema raccoglie le segnalazioni di disponibilità degli autisti nell'arco di 1 minuto
4. **Se** (l'insieme degli autisti disponibili non è vuoto), il sistema sceglie uno degli autisti disponibili (in base a diverse euristiche) e gli assegna la corsa, informandolo.
5. **Altrimenti**, se possibile, si ripete dal punto 2. con il tipo di auto successivo, in ordine di prezzo crescente (ma specificando nella richiesta che per la corsa si offre il prezzo corrispondente al tipo di auto della richiesta originale).
6. **Se** (è stato trovato un autista)
 1. Il Sistema informa il Cliente sui tempi di attesa
 2. Il Cliente accetta la corsa
 3. Il Sistema assegna la corsa all'autista e lo informa
 4. Il Sistema richiede all'Istituto di credito la pre-autorizzazione ad addebitare sulla carta di credito l'importo della corsa.
 5. Il sistema conferma la corsa all'utente, fornendo i dettagli di contatto dell'autista a cui è stata assegnata.

Altrimenti il sistema chiede al cliente di riprovare dopo qualche minuto

Sequenze alternative degli eventi: Non ci sono autisti disponibili. L'istituto di credito rifiuta la pre-autorizzazione.

- **Domanda 3.** Si diano i diagrammi di contesto secondo il metodo Jackson con i domini rilevanti per il caso d'uso, fornendo diagrammi separati per il caso in cui si sia chiamati a sviluppare soltanto:
 - la app in dotazione agli utenti
 - il Sistema REBU
 - il sistema di interfaccia (verso REBU) dell'Istituto di credito
 - In particolare, si indichi per ciascun caso quali domini sono dati (given), quali progettati (designed), e quali costituiscono la macchina (machine).

REBU: Esercizio 3

- Si diano i diagrammi di contesto secondo il metodo Jackson con i domini rilevanti per il caso d'uso, fornendo diagrammi separati per il caso in cui si sia chiamati a sviluppare soltanto:
 1. la app in dotazione agli utenti
 2. il Sistema REBU
 3. il sistema di interfaccia (verso REBU) dell'Istituto di credito
- In particolare, si indichi per ciascun caso quali domini sono dati (given), quali progettati (designed), e quali costituiscono la macchina (machine).

SOLUZIONE: costruire a lezione

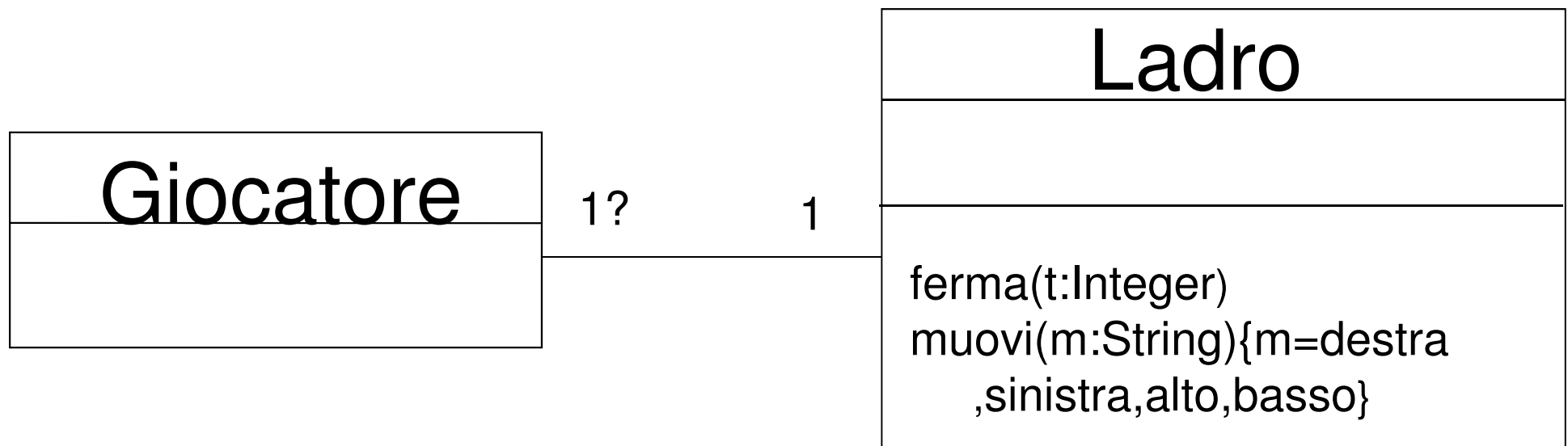
Cops and Robbers

Cops and Robbers (1)

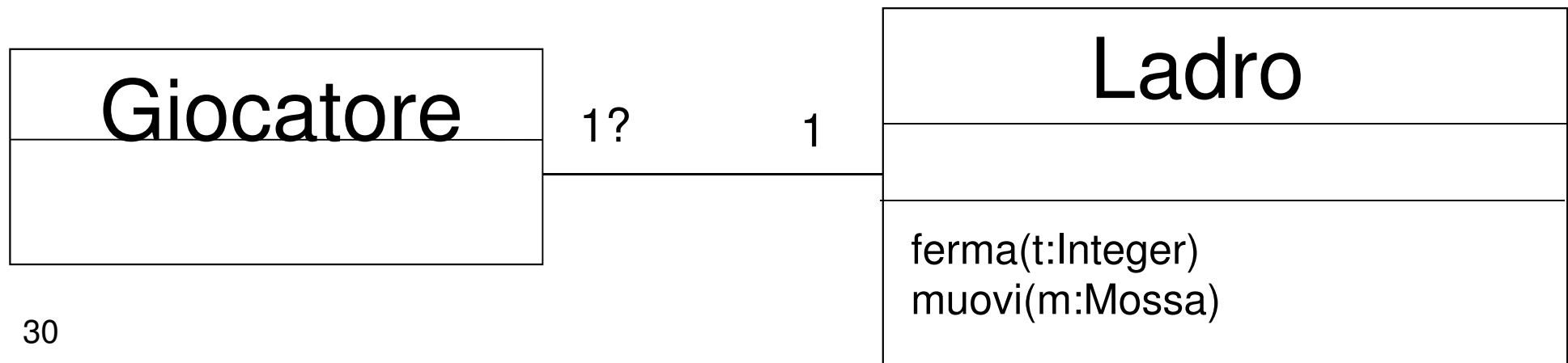
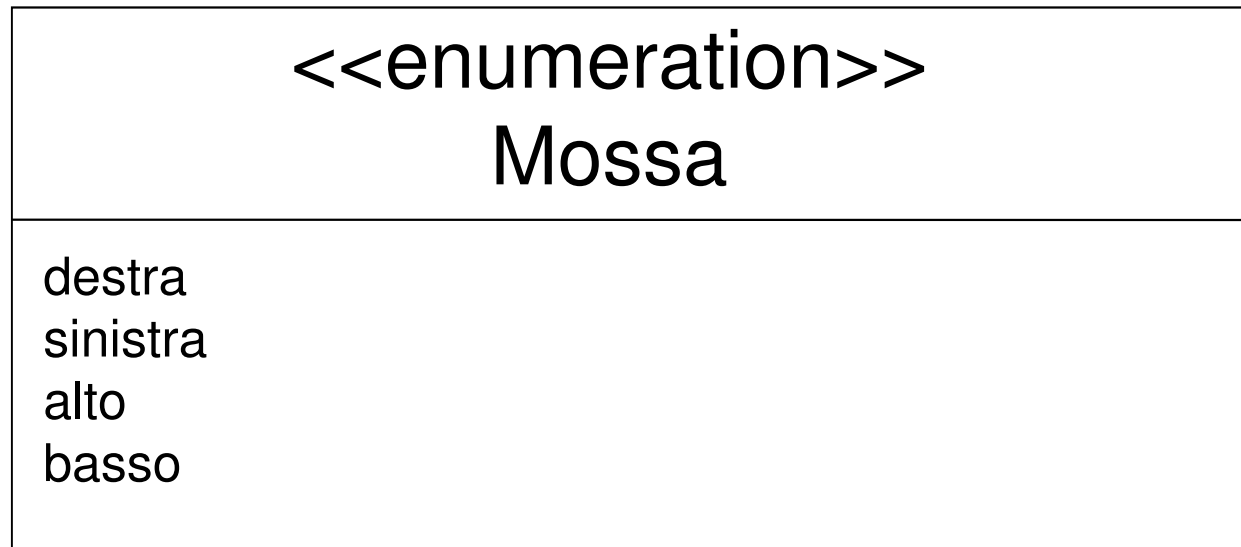
- Scopo del progetto e quello di realizzare Cops And Robbers (Guardie e Ladri) un semplice gioco multiplayer.
- In Cops And Robbers il mondo virtuale è popolato da un certo numero di Ladri e da un insieme di Guardie.

Cops and Robbers (2)

- Ogni giocatore controlla un singolo ladro, e lo può muovere nelle quattro direzioni destra, sinistra, alto, basso, oppure fermarlo per un intervallo di tempo arbitrario.



Tipo enumerazione

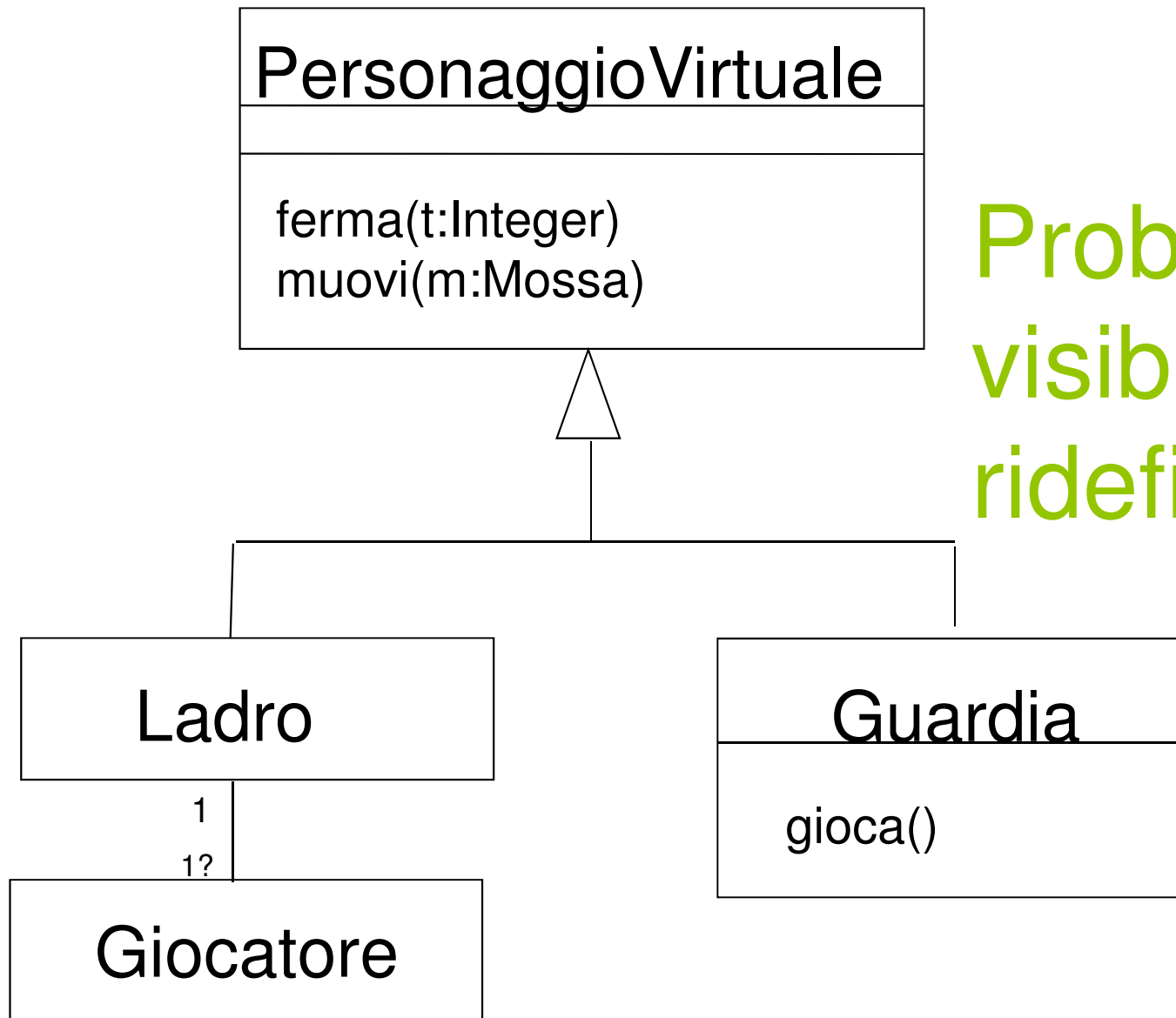


Cops and Robbers (3)

- Nel mondo virtuale si muovono inoltre alcune Guardie, dotate di intelligenza artificiale il cui movimento (anche esso nelle quattro direzioni) è determinato in base ad un semplice algoritmo.

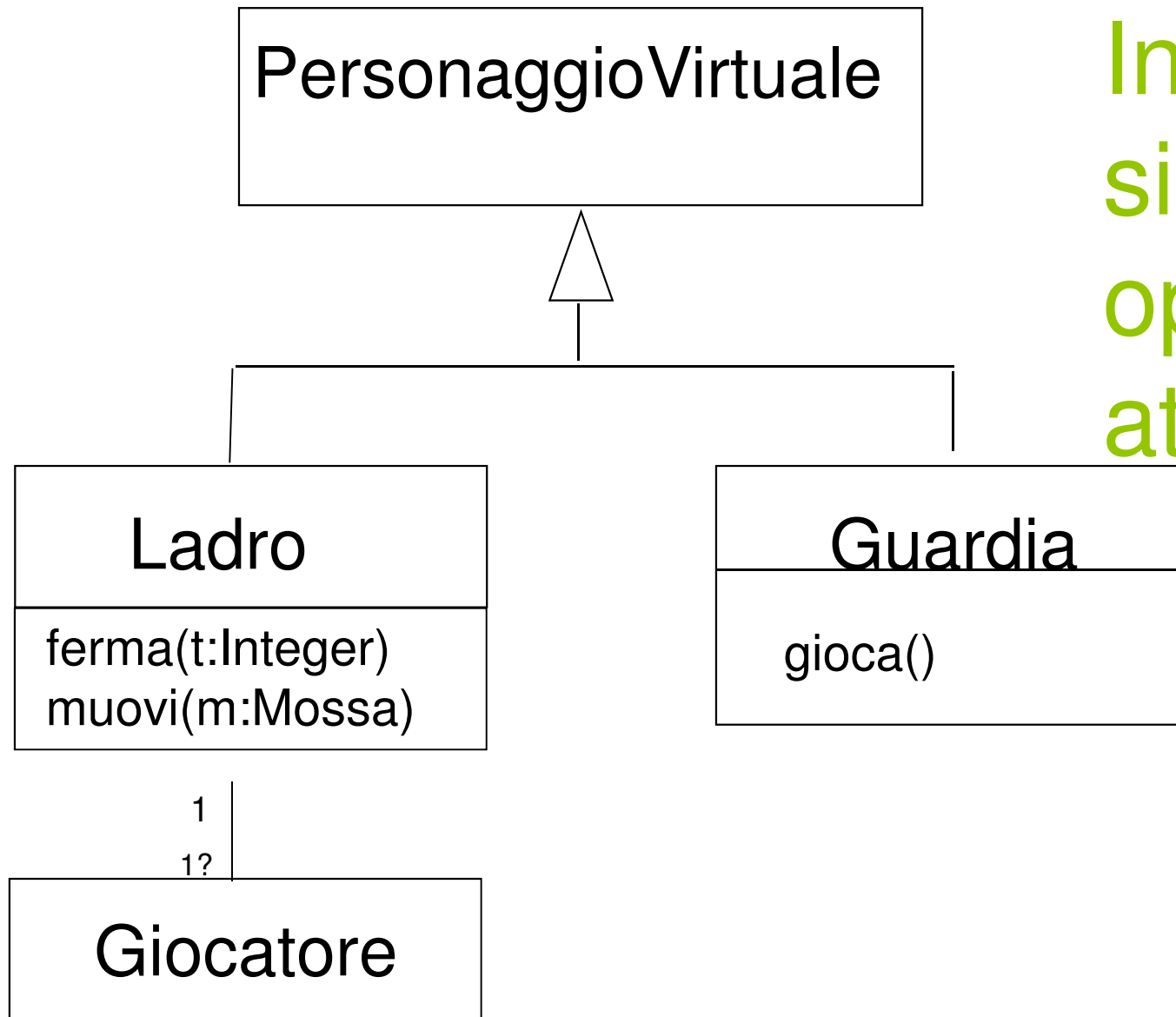
Guardia
- ferma(t:Integer) ?? - muovi(m:Mossa) gioca()

Cops and Robbers (4)

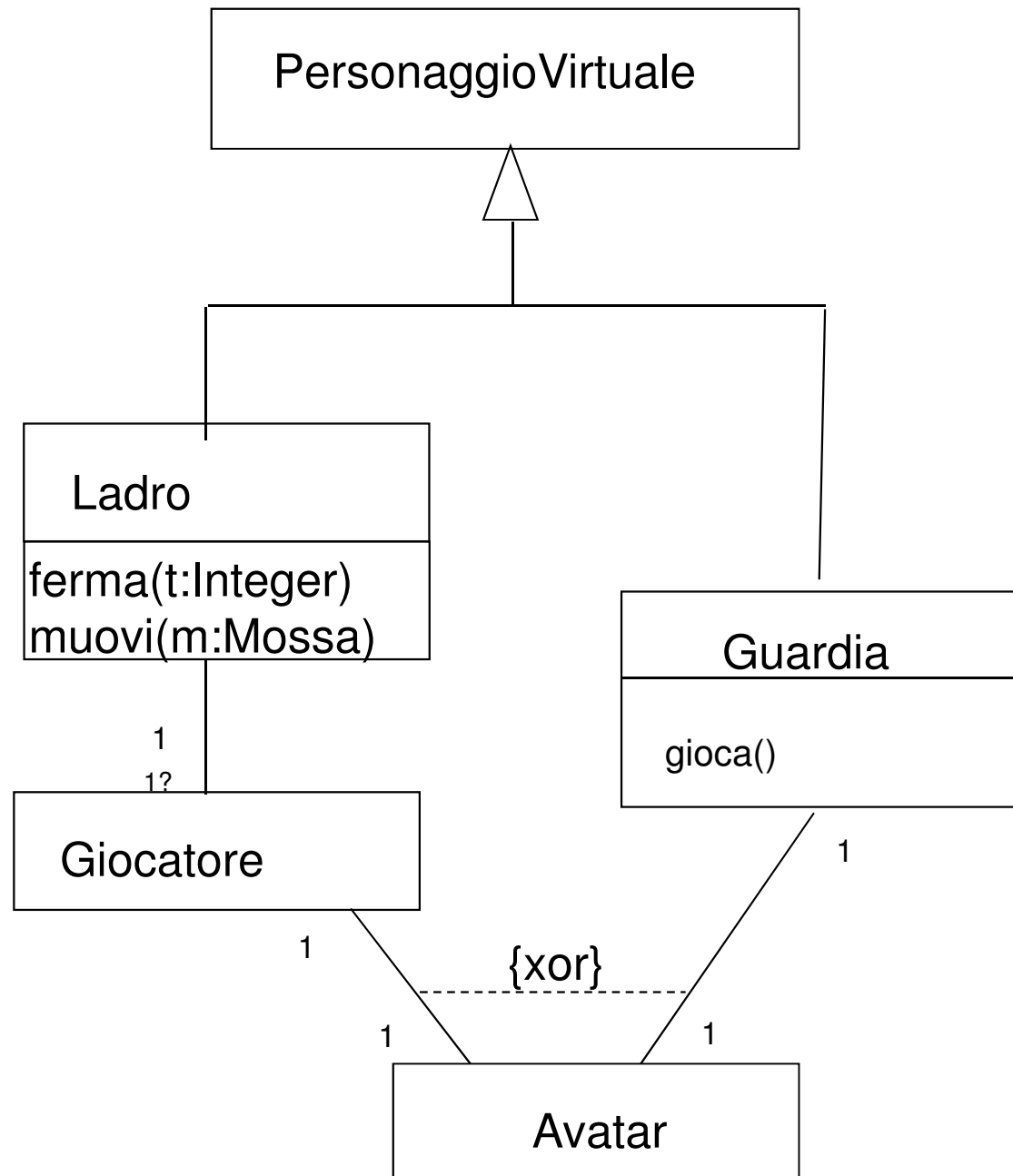


Problema con
visibilità (non
ridefinibile)

Cops and Robbers (4b)



In analisi
si ignorano
operazioni e
attributi privati



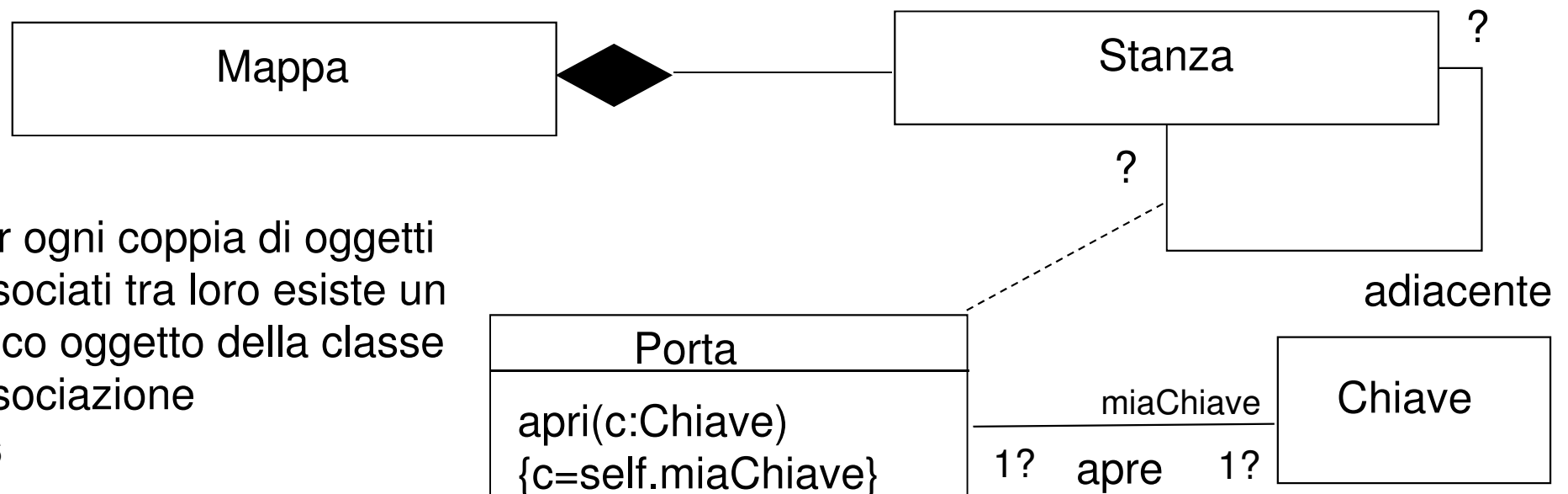
- Ad ogni guardia ed ad ogni giocatore è associato un avatar diverso, che lo identifica univocamente nel gioco.

Ha senso?

- Ladro vs Giocatore
 - Ridiscutere descrizione del dominio

Cops and Robbers (6)

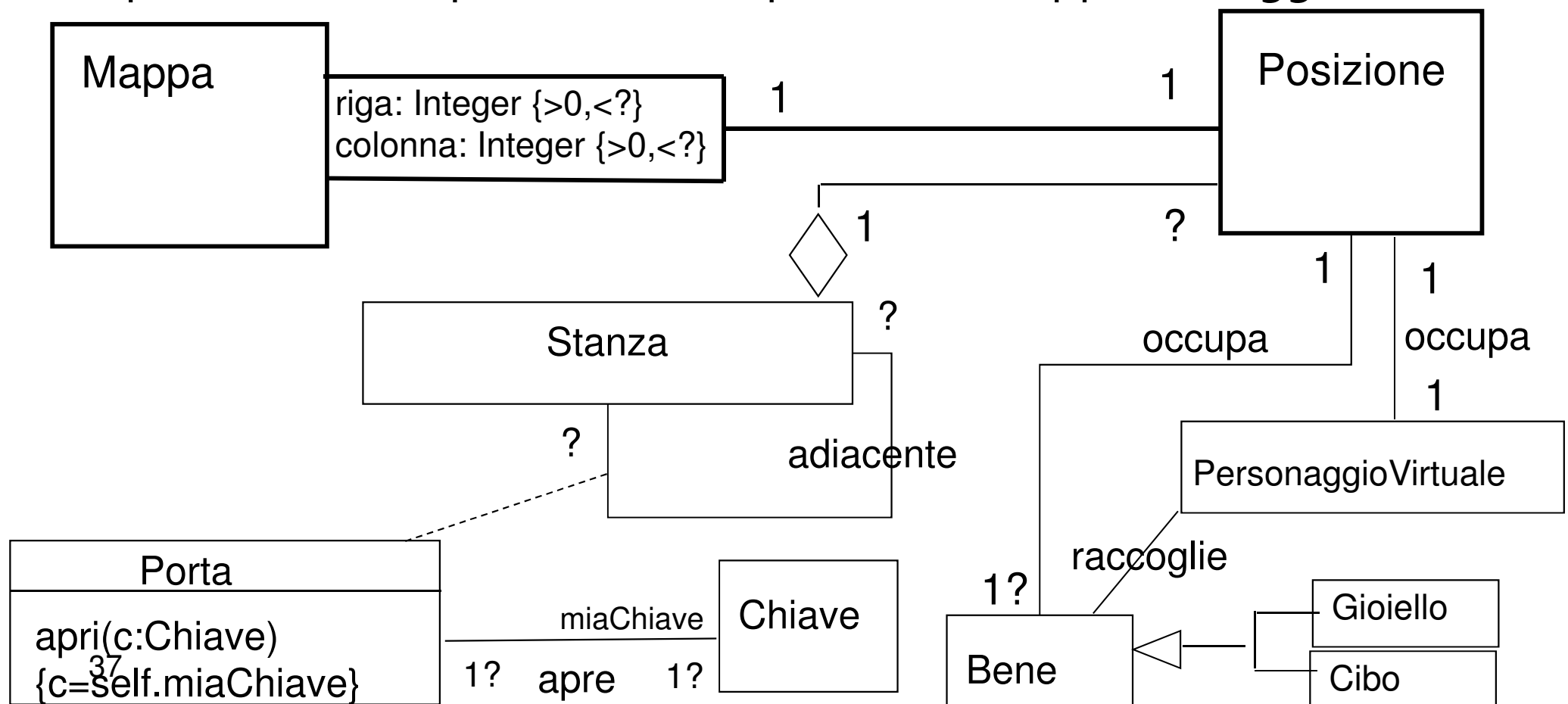
- La mappa di gioco è divisa in un certo numero di stanze, divise da porte. I ladri e le guardie possono spostarsi da una stanza ad una adiacente mediante la porta di accesso. Quando un giocatore passa in un'altra stanza, si ferma per un breve intervallo di tempo davanti alla porta di ingresso per cercare la chiave ed aprirla.



- Per ogni coppia di oggetti associati tra loro esiste un unico oggetto della classe associazione

Cops and Robbers (9)

- In ogni stanza sono presenti un insieme di cibi ed un insieme di gioielli, che i giocatori (personaggio virtuale?) possono raccogliere passando sulla posizione occupata sulla mappa dall'oggetto.

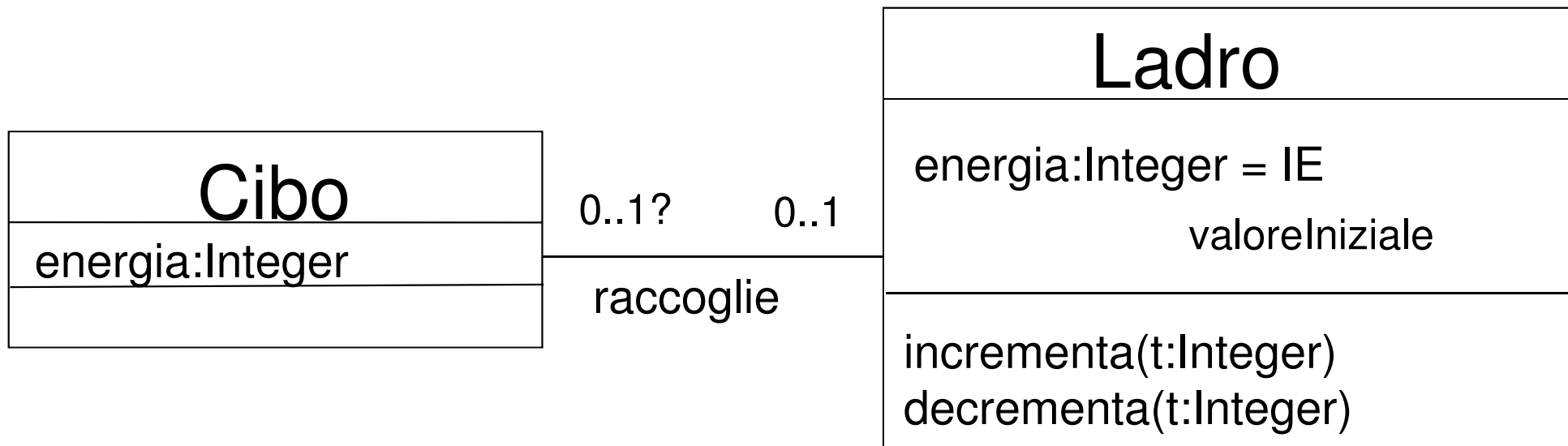


Cops and Robbers (10)

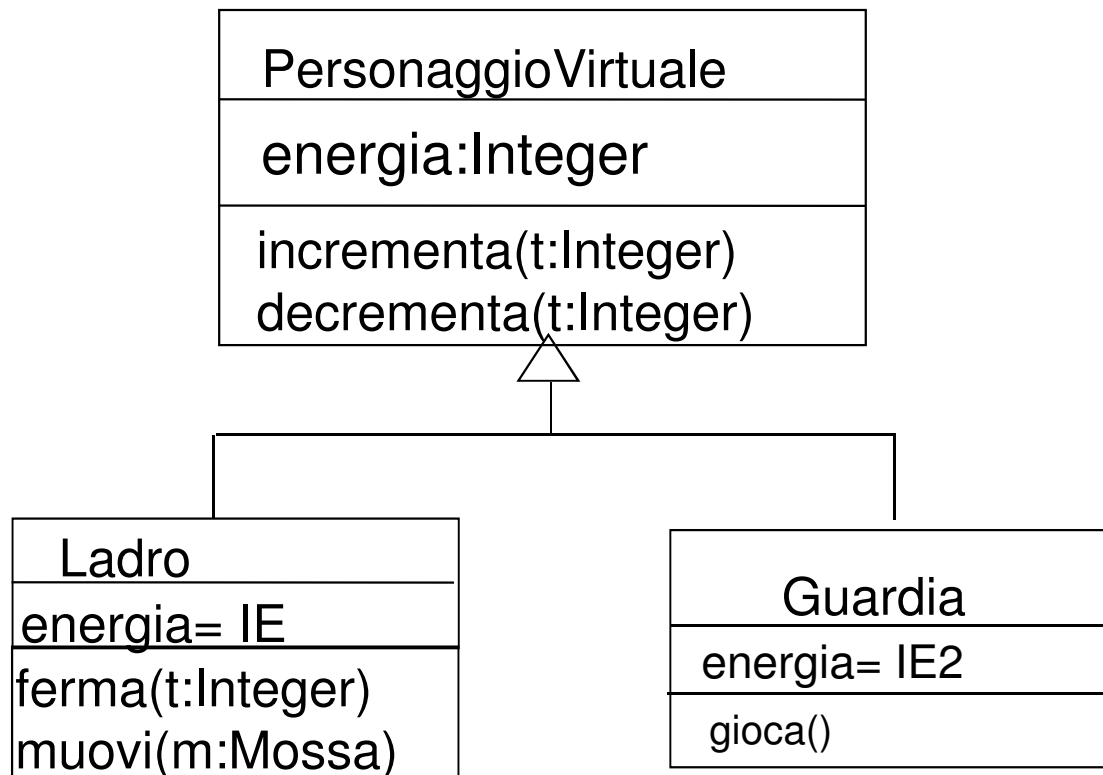
- Ogni ladro è caratterizzato da un livello di energia. Ogni cibo è caratterizzato da un valore intero, che indica l'energia che può fornire al giocatore che lo raccoglie. Il livello di energia di ogni ladro viene incrementato o decrementato durante il gioco, secondo le seguenti regole:
 - ad ogni ladro viene attribuito un livello IE di energia, all'inizio del gioco, uguale per tutti i ladri.
 - ogni ladro può incrementare il suo livello di energia raccogliendo cibo.
 - il livello di energia di un ladro viene decrementato se e quando si muove.
 - Lo spostamento di un ladro è determinato dal suo livello di energia. Se un ladro raggiunge il livello 0 di energia muore e viene eliminato dal gioco.

Cops and Robbers (11) modello statico

- Ogni ladro è caratterizzato da un livello di energia. Ogni cibo è caratterizzato da un valore intero, che indica l'energia che può fornire al giocatore che lo raccoglie. Il livello di energia di ogni ladro viene incrementato o decrementato ...:
 - ad ogni ladro viene attribuito un livello IE di energia, all'inizio del gioco, uguale per tutti i ladri.



Cops and Robbers (15)



- Anche le guardie sono caratterizzate da un livello di energia, che viene loro attribuito all'inizio del gioco. L'energia di un guardia si incrementa ogni volta che essa cattura un ladro e si decrementa quando essa si muove.

Soluzione comunque ragionevole: discutere

Visual Paradigm for UML Standard Edition (University of Pisa)

Gioco

