UNIVERSITÀ PARTHENOPE
DiST
Dipartimento di Scienze e Tecnologie

Programmazione 3
e
Laboratorio di Programmazione 3

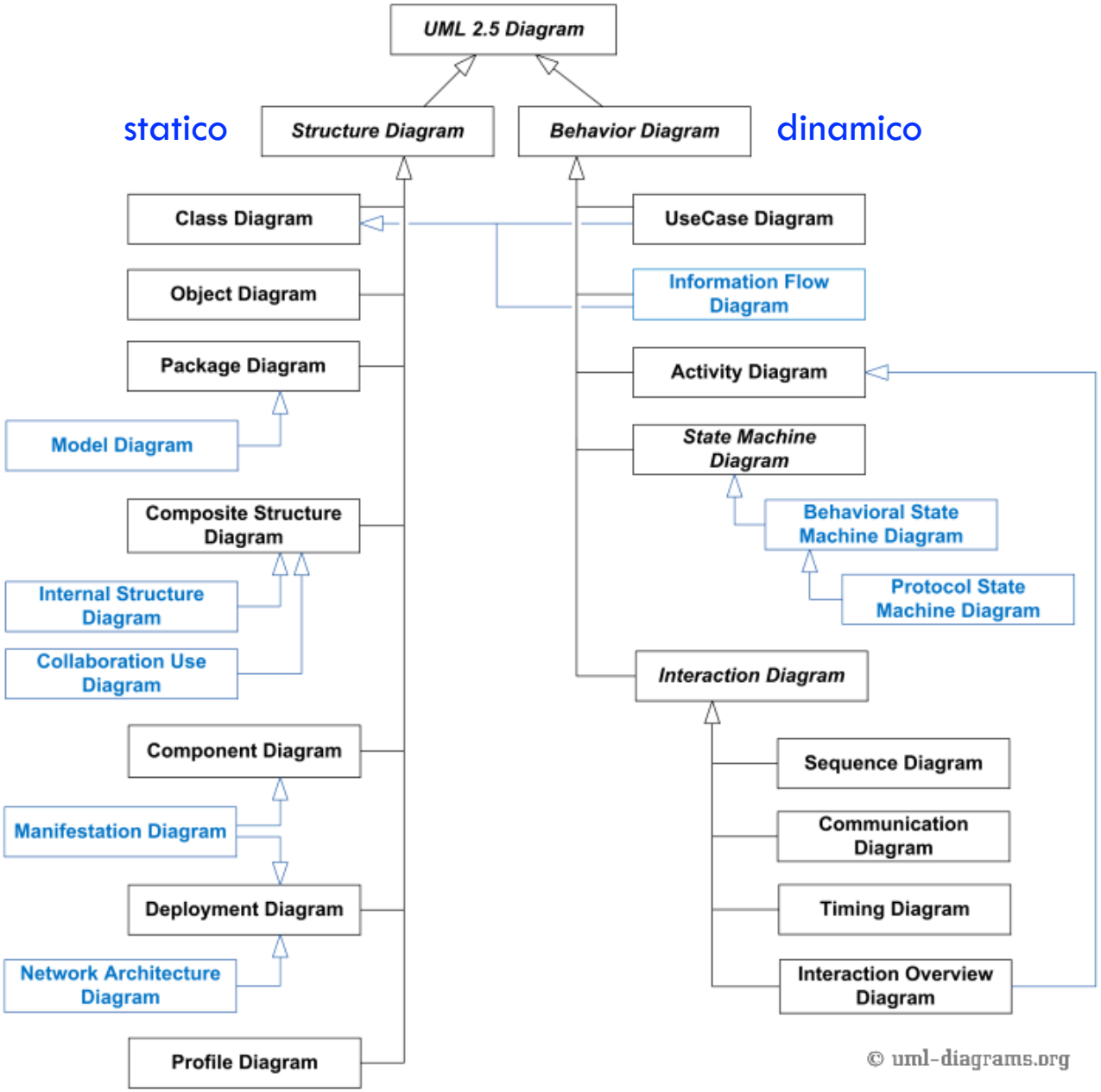
Unified Modeling Language (UML)

Angelo Ciaramella

Unified Modeling Language

- Unified Modeling Language (UML)
  - insieme di diagrammi formali per la descrizione di un problema o di una soluzione
  - prima versione sviluppata nel 1994
  - l'ultima versione è la 2.5
  - contiene elementi grafici (simboli) connessi

UML 2.5



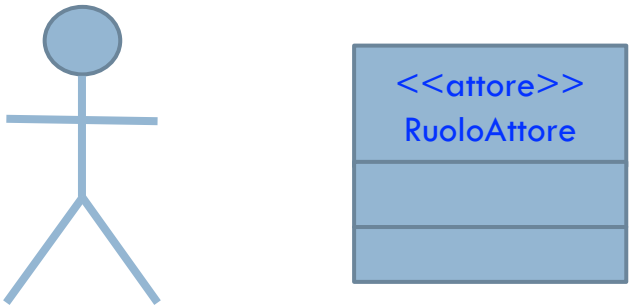
non sono ufficialmente inseriti

Use Case Diagram

- Use Case Diagram
  - interazioni tra l'ambito della descrizione e le entità a esso esterne
- Elementi principali
  - attore
  - caso d'uso (use case)

Attore

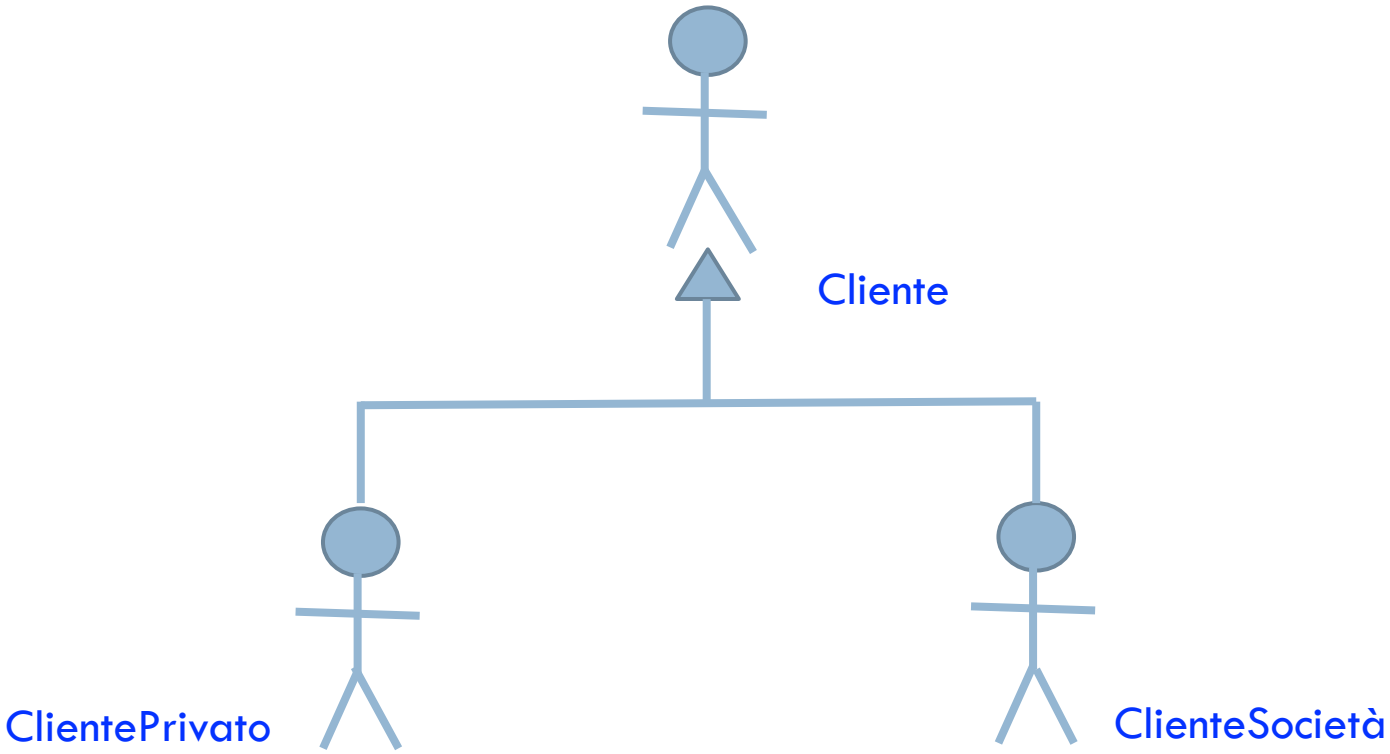
- Attore
  - una persona
  - componente di un altro sistema



Rappresentazione entità `Attore`

Generalizzazione

- Relazione di generalizzazione
  - Derivazione di entità figlie da un padre



Generalizzazione di Attori

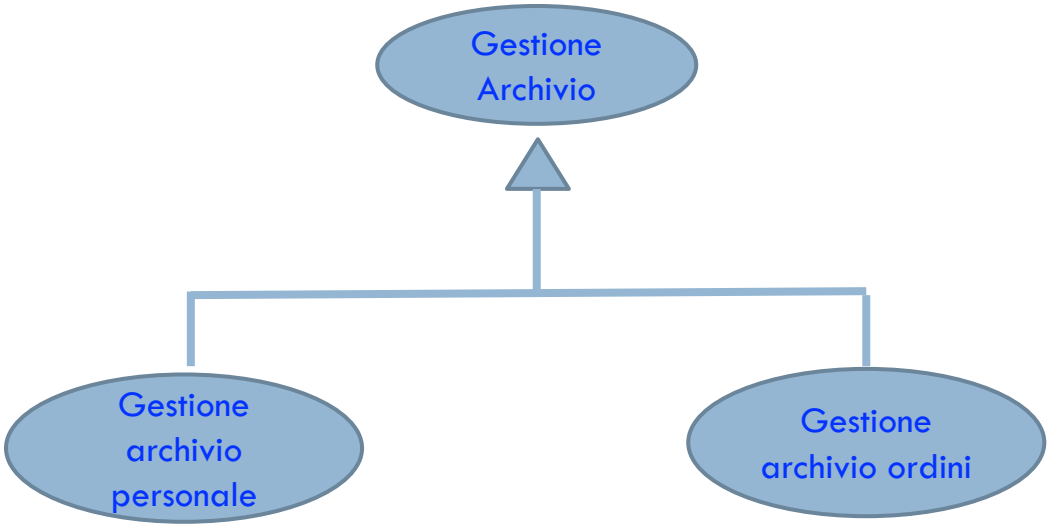
Caso d'uso

- Caso d'uso
  - Descrive una macro-funzionalità

Caso d'uso

Generalizzazione

- Generalizzazione tra use case
  - l'use case figlio eredita tutte le caratteristiche dello use case padre
  - l'use case figlio è presente in tutti gli scenari dello use case padre

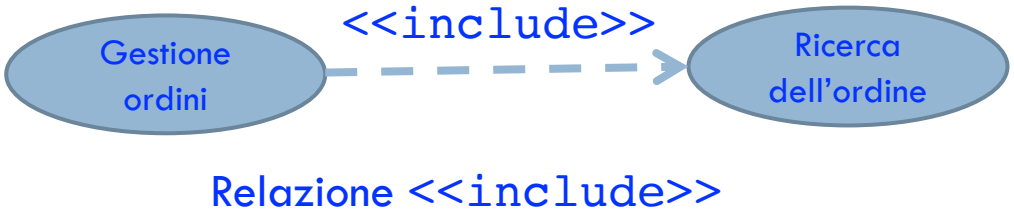


Generalizzazione tra use case

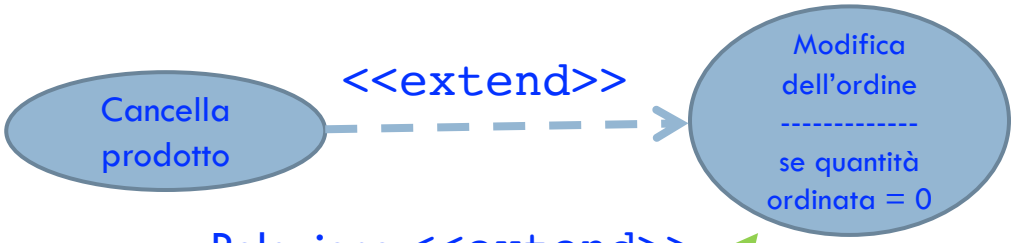
Relazioni di inclusione ed estensione

- Le relazioni di inclusione ed estensione
  - stereotipi della relazione di dipendenza
    - elementi di UML per evidenziare proprietà peculiari
  - inclusione
    - per la realizzazione dello use case che include, è necessario che sia realizzato lo use case incluso
  - estensione
    - uno use case estende un secondo use case (use case base) quando descrive in modo più ampio e dettagliato una variante dello use case base

Relazioni di inclusione ed estensione



Relazione <<include>>

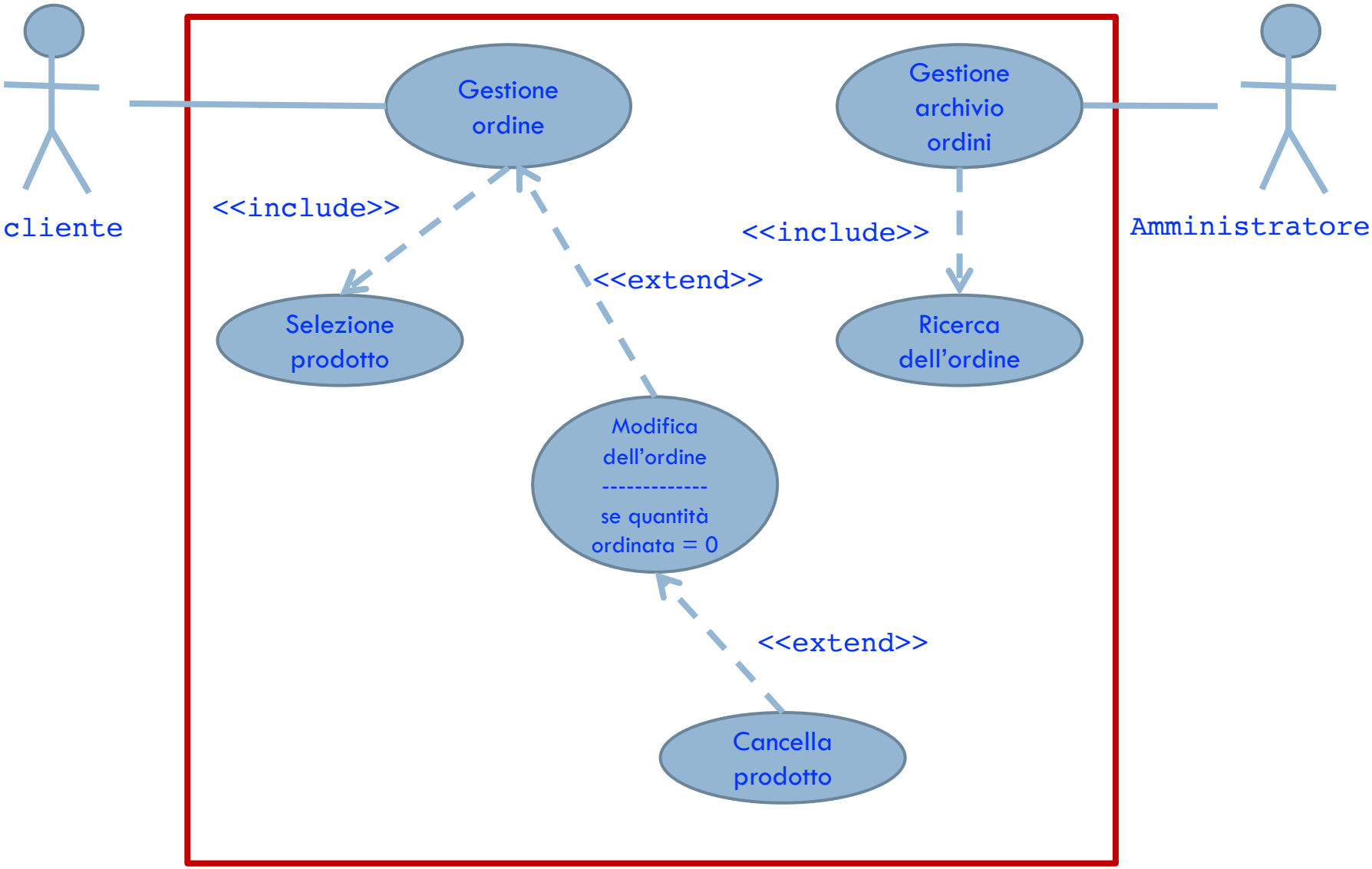


Relazione <<extend>>

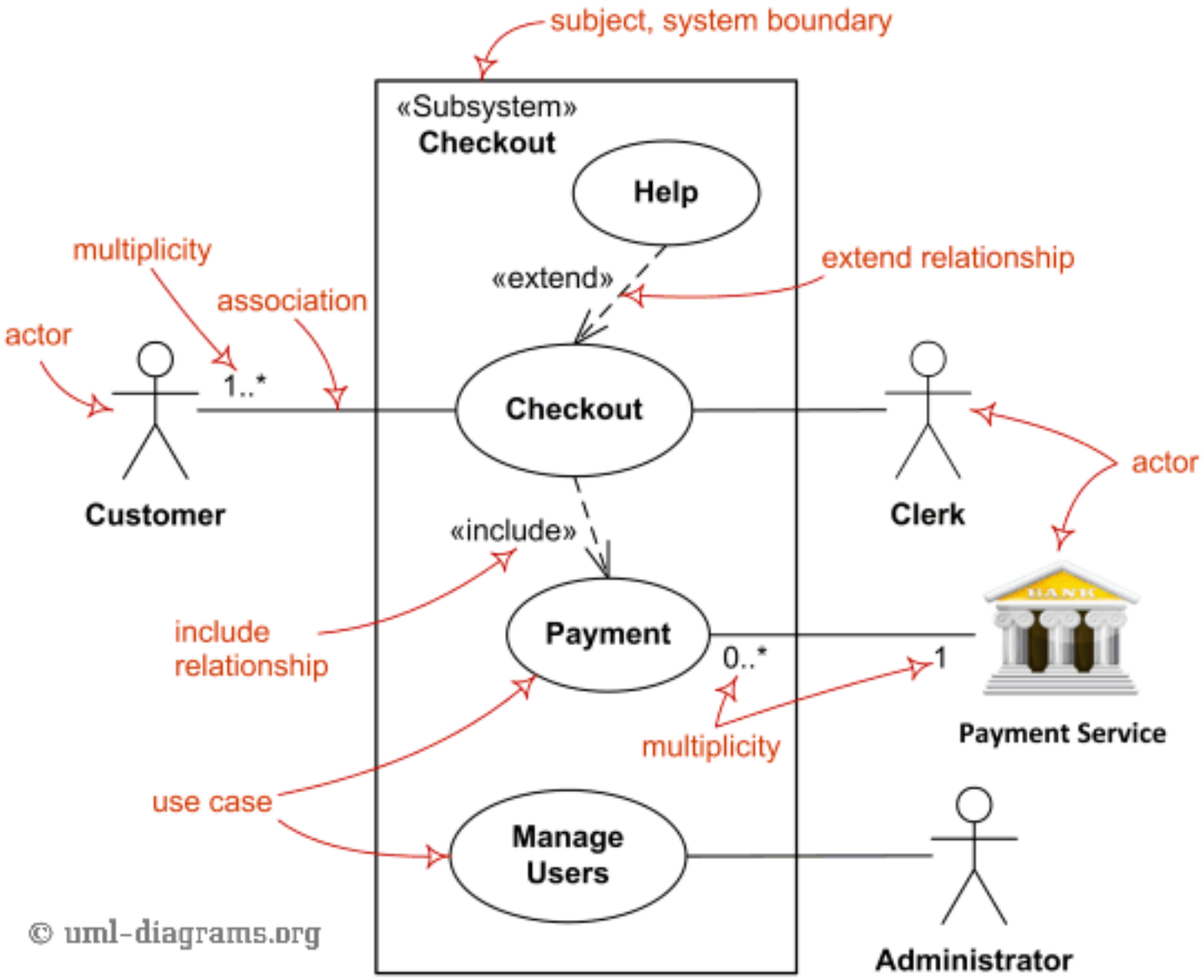
use case base

extension point, vincoli di esecuzione

Esempio di use case diagram



Esempio di use case diagram

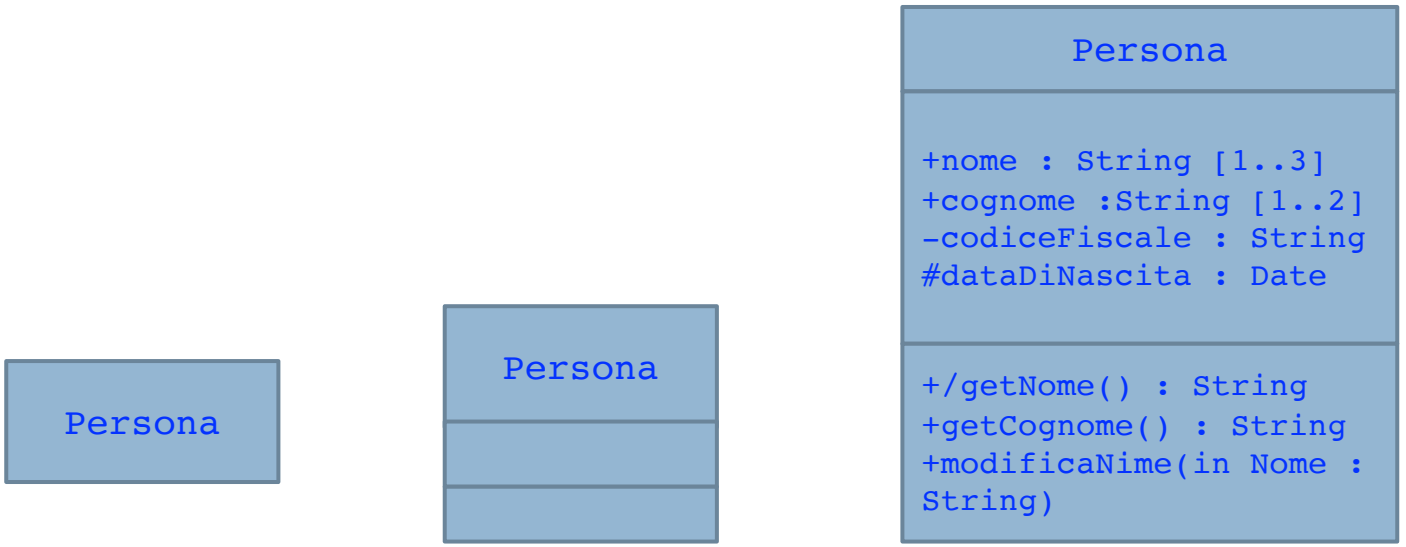


Class diagram

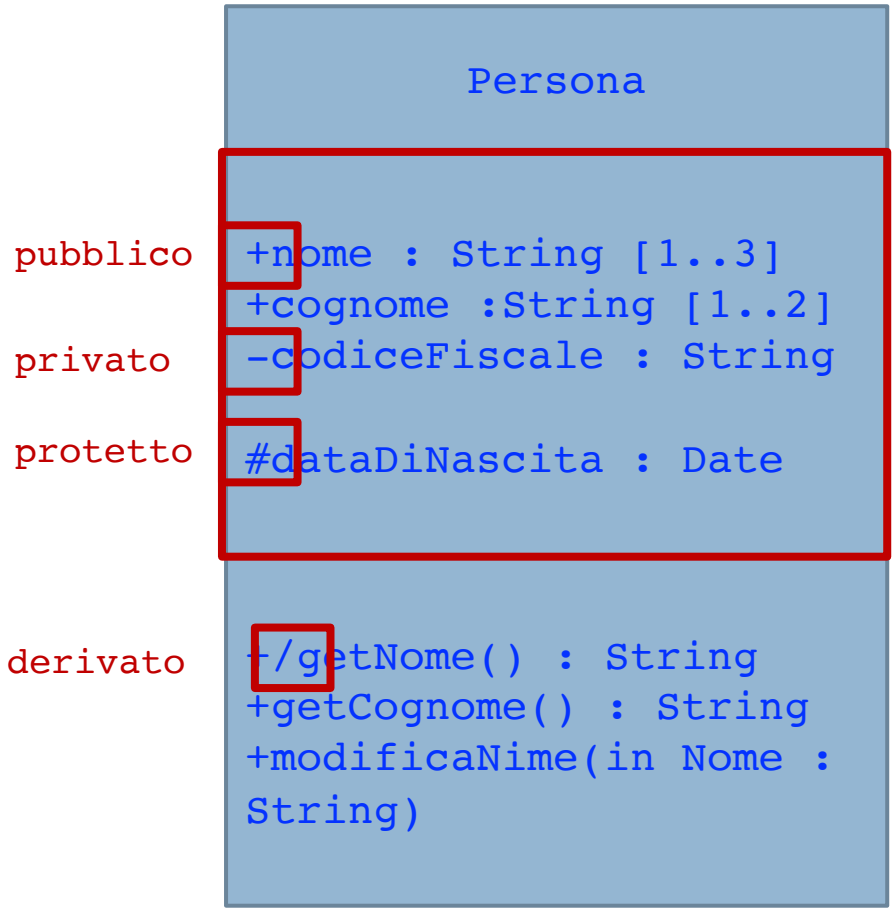
- Il class diagram
  - illustra l'ambito di descrizione da un punto di vista statico
  - evidenzia caratteristiche e mutue relazioni
    - classi
    - relazioni

Classe

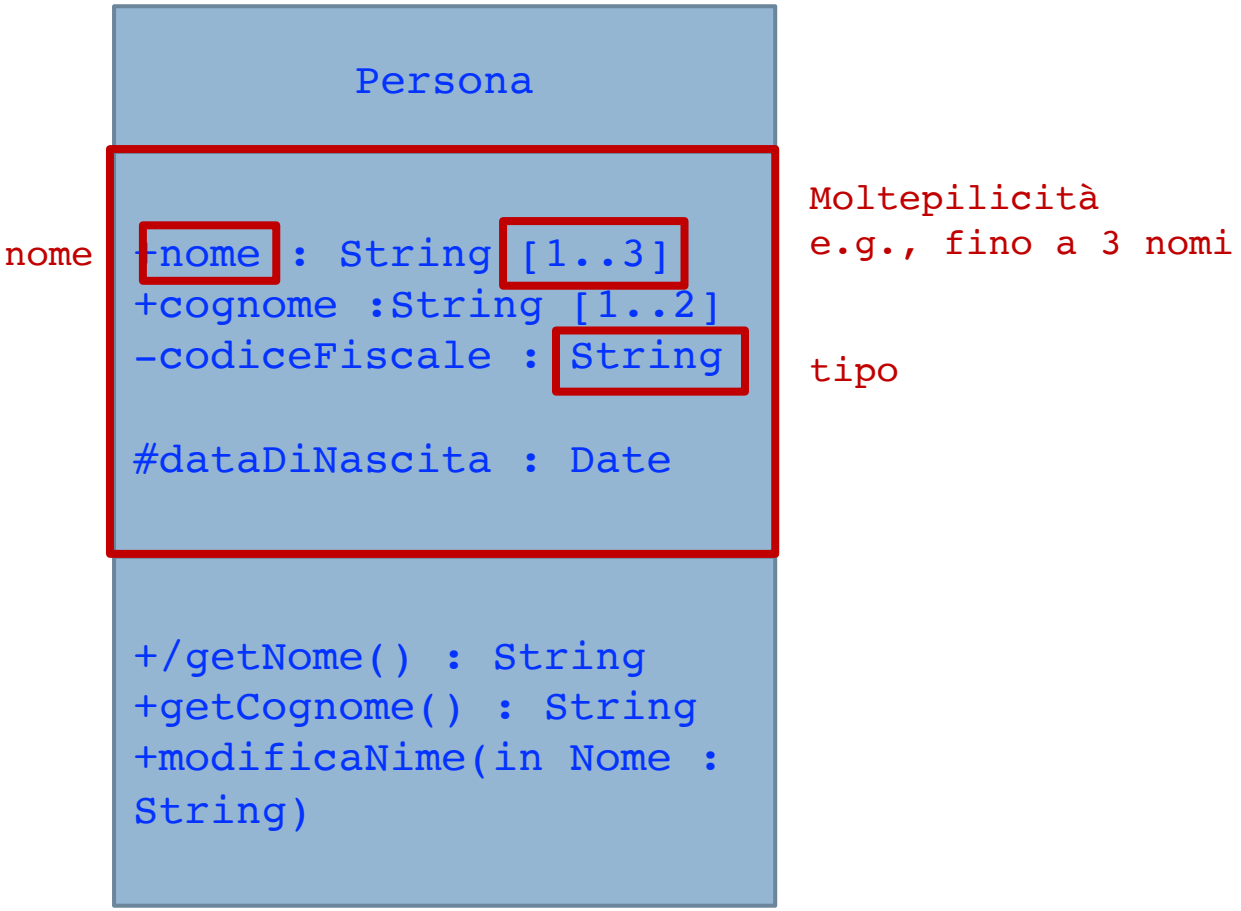
- Classe
  - descrive un insieme di entità dotate delle stesse caratteristiche e proprietà
    - oggetti



Attributi

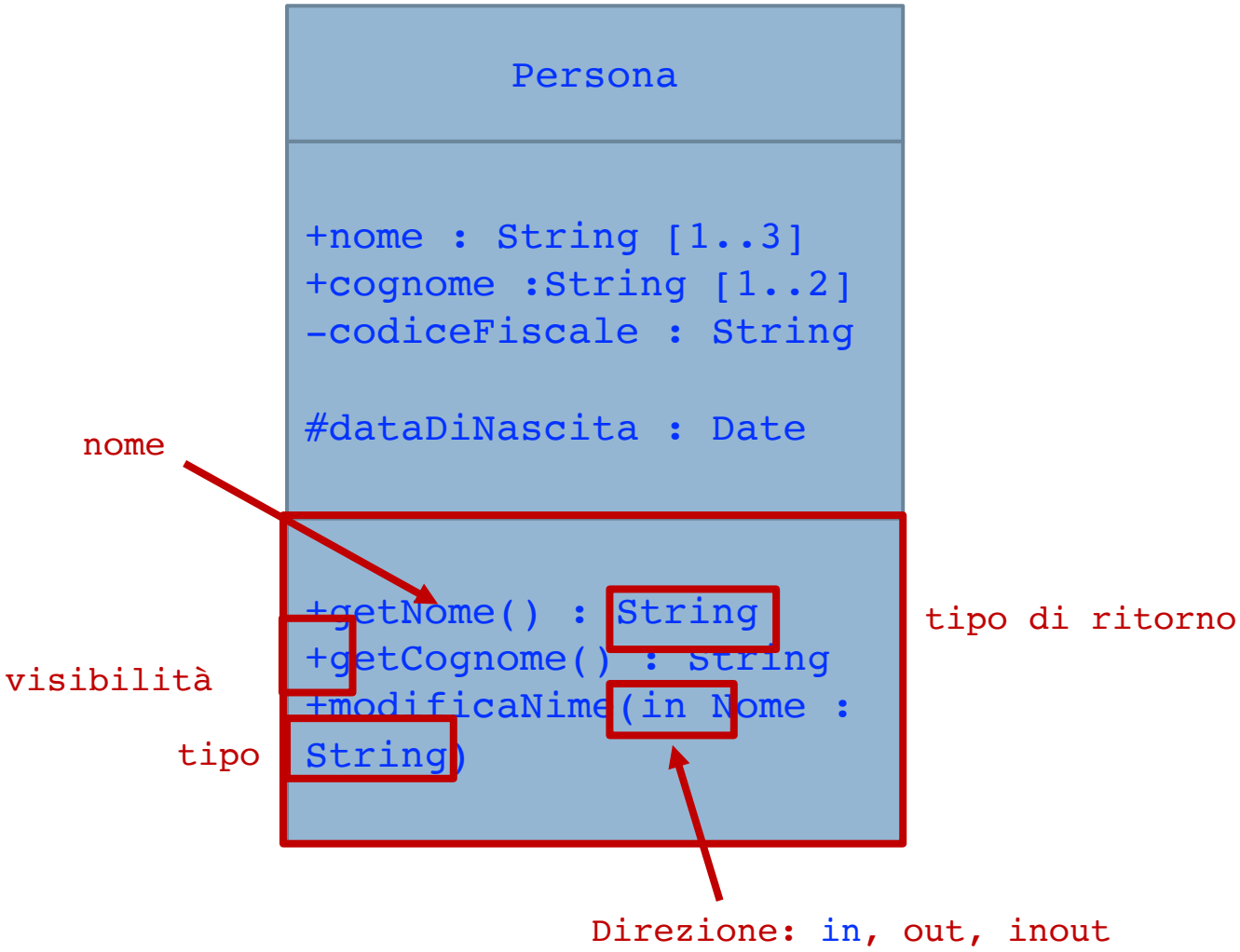

Persona
pubblico
privato
protetto
derivato
+nome : String [1..3]
+cognome :String [1..2]
-codiceFiscale : String
#dataDiNascita : Date
+/getNome() : String
+getCognome() : String
+modificaNime(in Nome : String)

Attributi



E' possibile avere anche un valore di Default per gli attributi

Operazioni


Persona
+nome : String [1..3]
+cognome :String [1..2]
-codiceFiscale : String
#dataDiNascita : Date
+getNome() : String
+getCognome() : String
+modificaNime(in Nome : String)
nome
visibilità
tipo
tipo di ritorno
Direzione: in, out, inout

Classe astratte

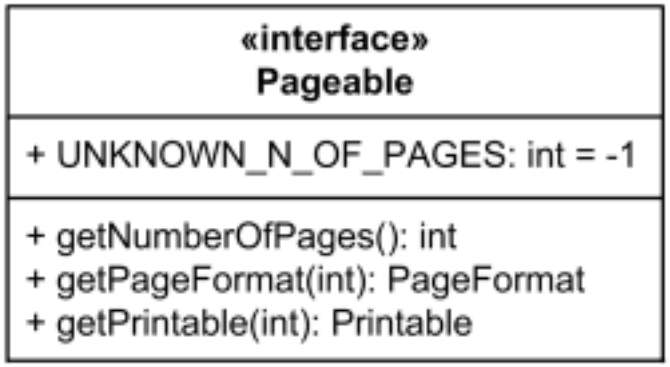
- Classe astratta
  - Contiene attributi e dichiarazioni di operazioni
  - Non può essere istanziata direttamente
  - L'implementazione delle classi che estendono la classe

Interfaccia

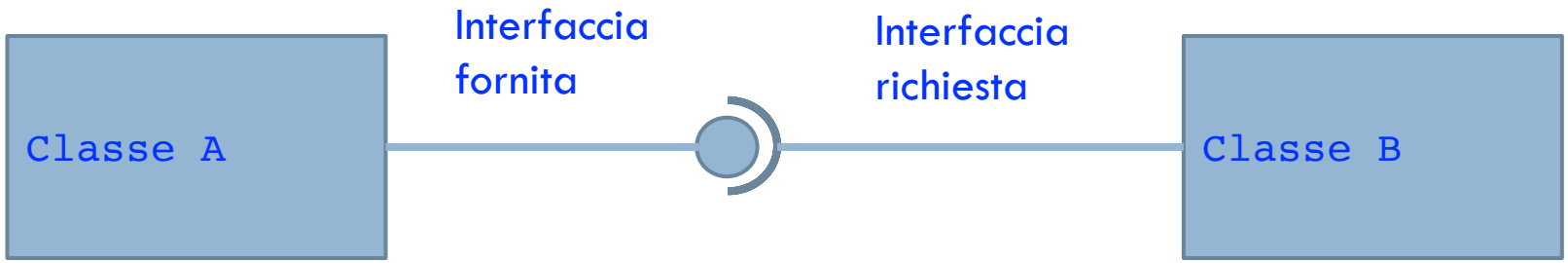
- Interfaccia
  - è una classe che non ha implementazione
  - presenta solo dichiarazioni di operazioni

- Le interfacce sono di due tipi
  - Interfaccia fornita
    - La classe fornisce le operazioni che vi sono dichiarate
  - Interfaccia richiesta
    - La classe ha bisogno delle operazioni in essa
    - dichiarate per poter svolgere le elaborazioni

Interfaccia

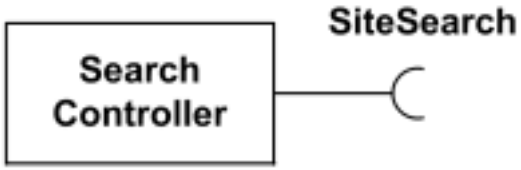


Esempio di interfaccia

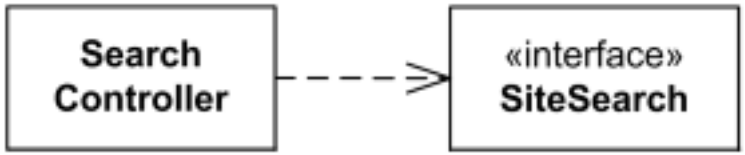


Esempio di interfacce

Interfaccia

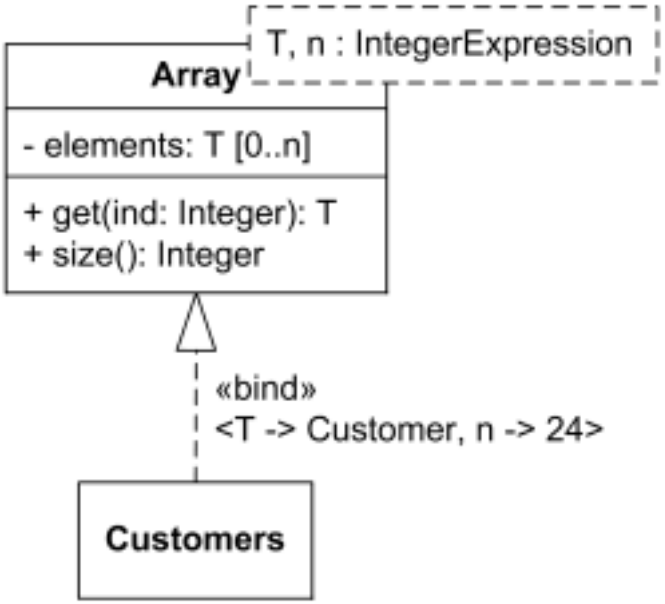


Esempio di interfaccia richiesta

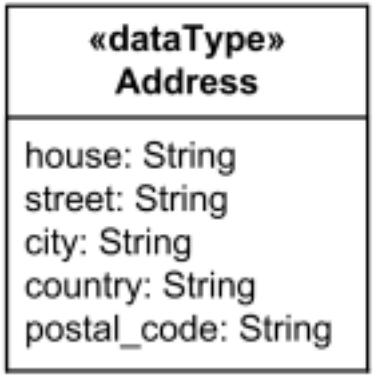


Esempio di interfaccia richiesta con l'interfaccia rappresentata da un rettangolo

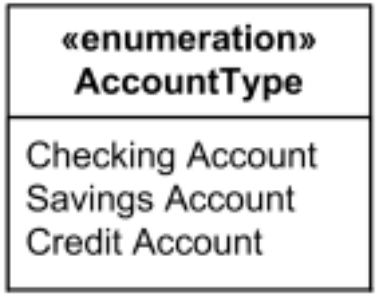
Template, Enumerazioni e Dato



Classe Template



Tipo dato



Enumerazione

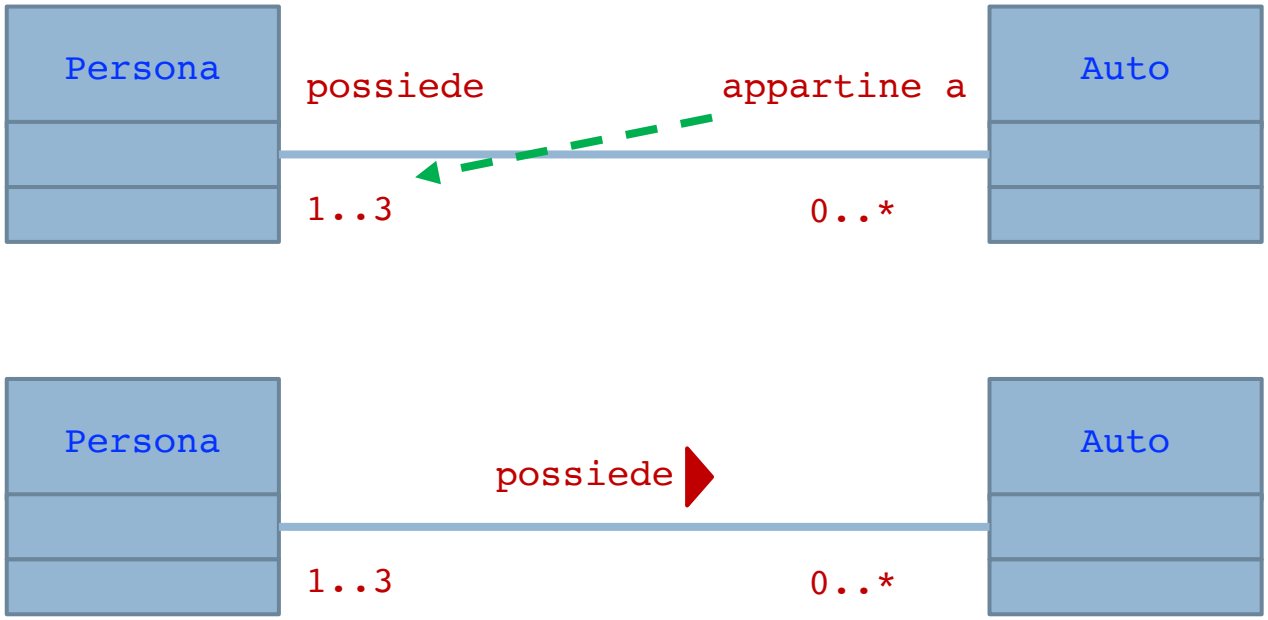
Dati primitivi

Stereotipi di classe

- Stereotipi di classe standard
  - Focus
    - classe principale per gestire un flusso di controllo di più classi
  - Auxiliary
    - classe ausiliare
  - Type
    - specifica un dominio per gli oggetti e operazioni applicabili
  - Utility
    - classe con attributi e operazioni statiche

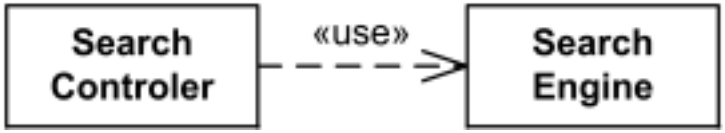
Associazioni

- Associazione
  - relazione statica che lega le classi tra di loro

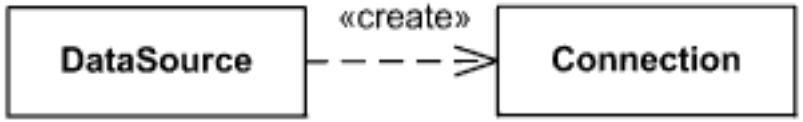


* numero massimo non definito

Dipendenza Usage

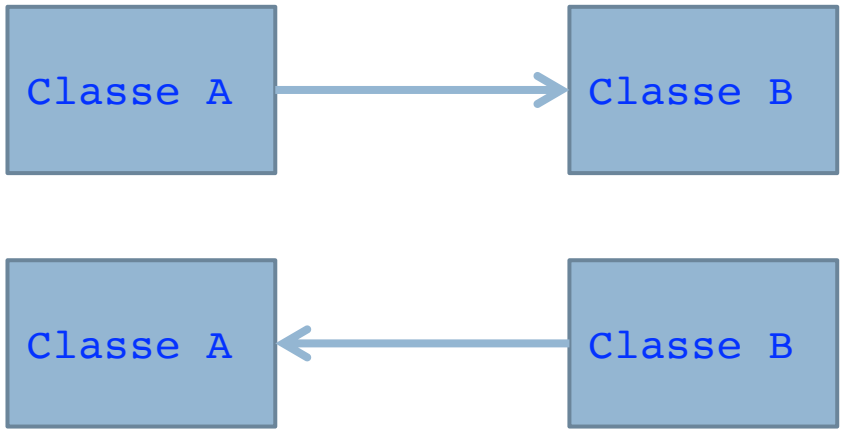


Relazione di uso



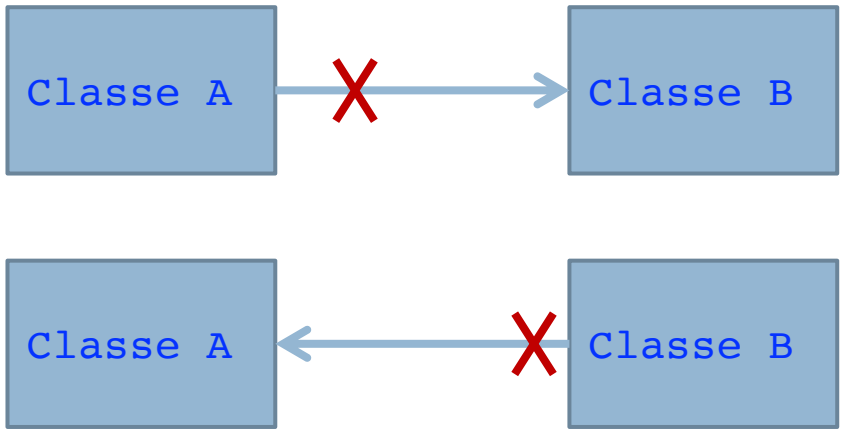
Relazione di creazione

Navigabilità



Navigabilità unidirezionale

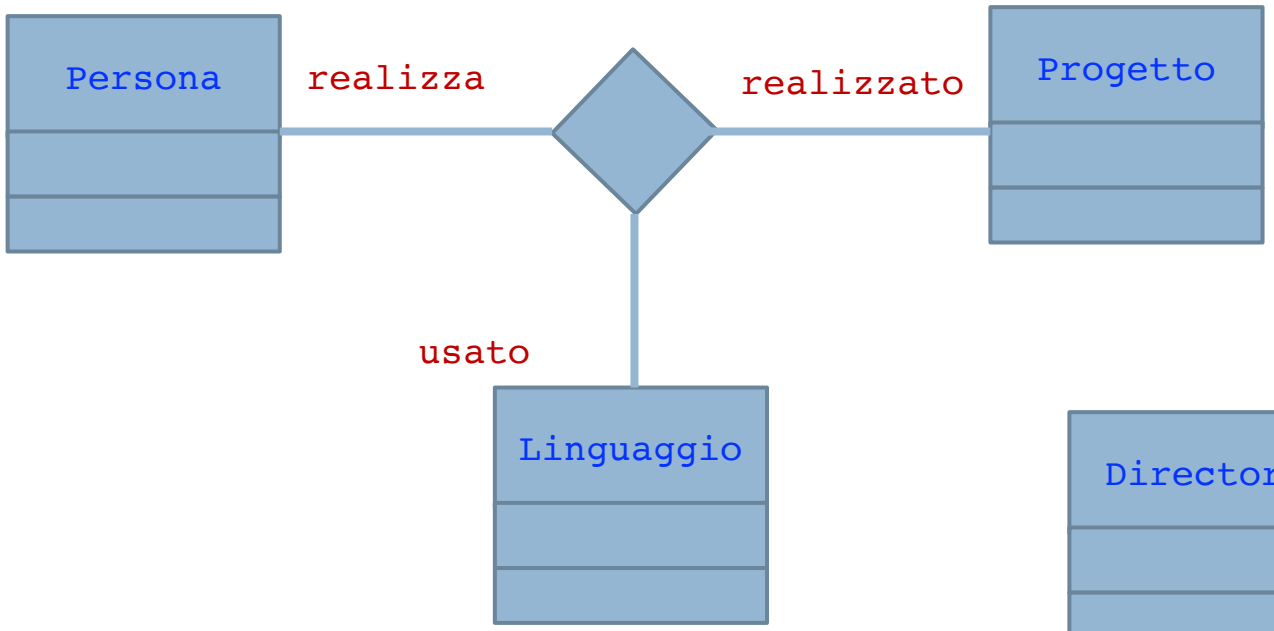
Navigabilità didirezionale



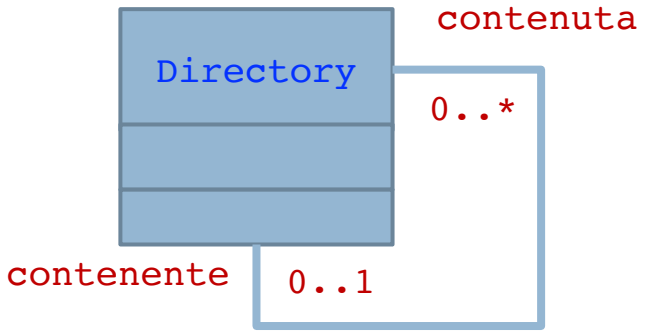
Non navigabilità

Associazioni

Associazione binaria

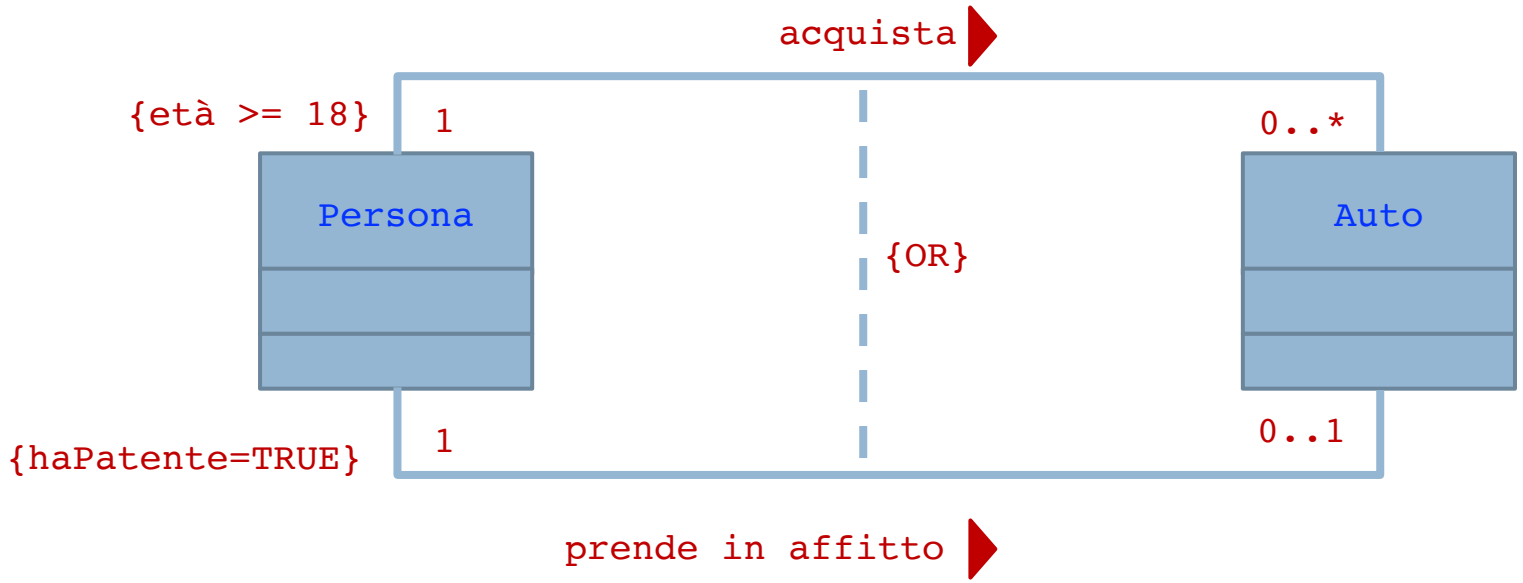


Associazione n-aria



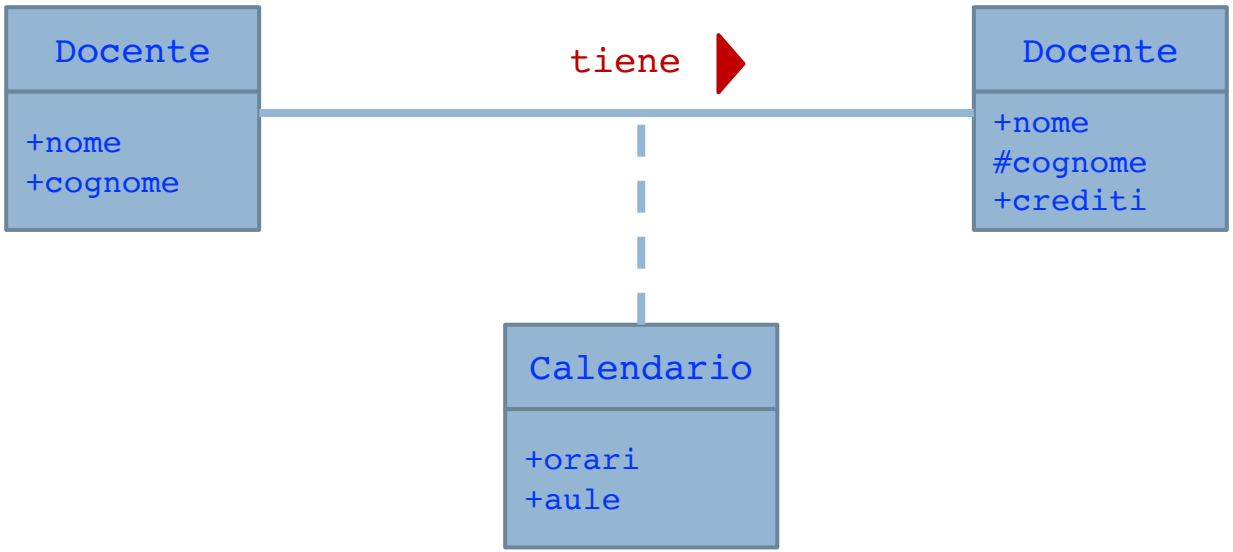
Associazione riflessiva

Vincoli



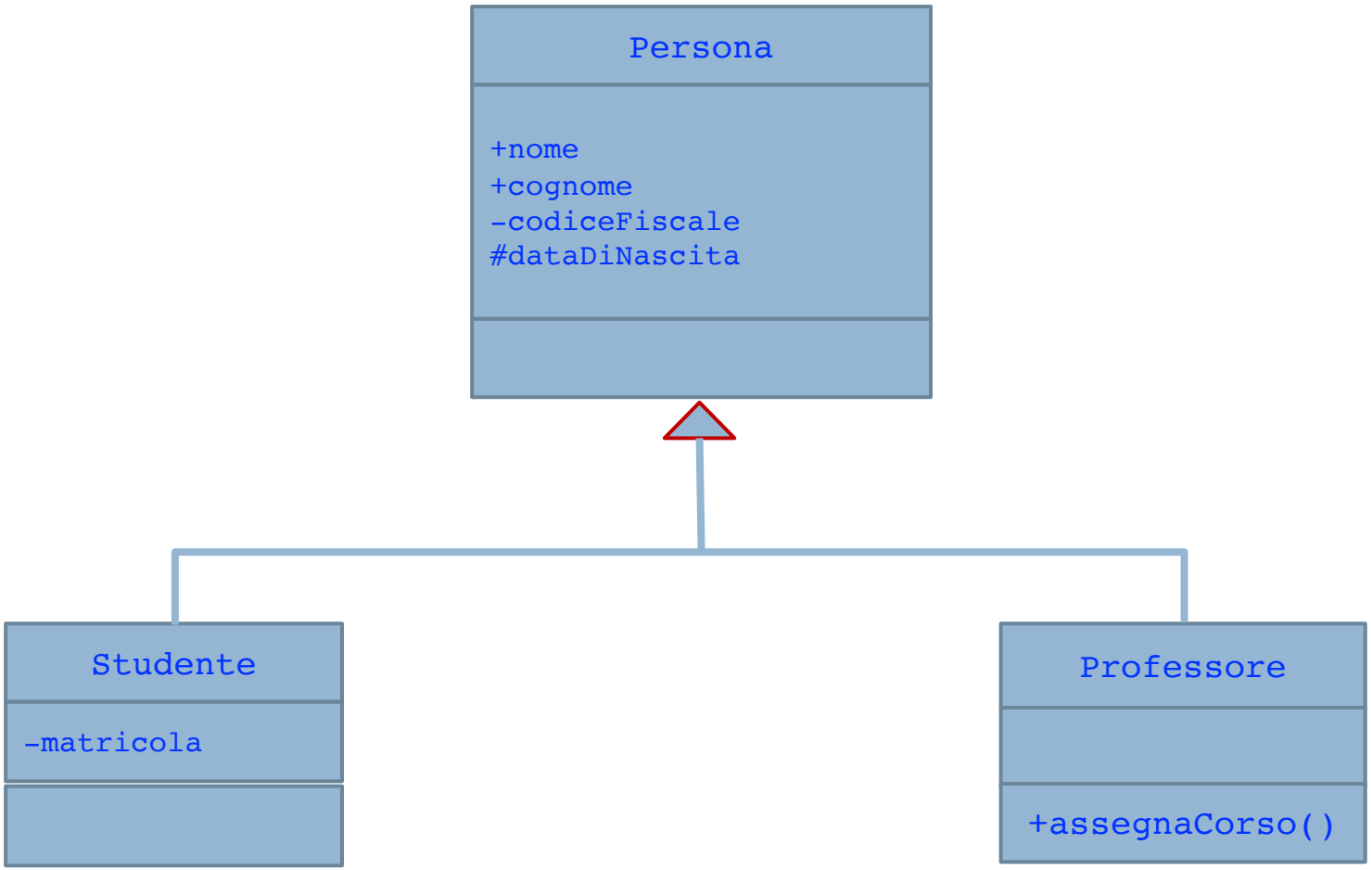
Vincoli sulle associazioni

Association class



Esempio di `association class Calendario`

Relazioni



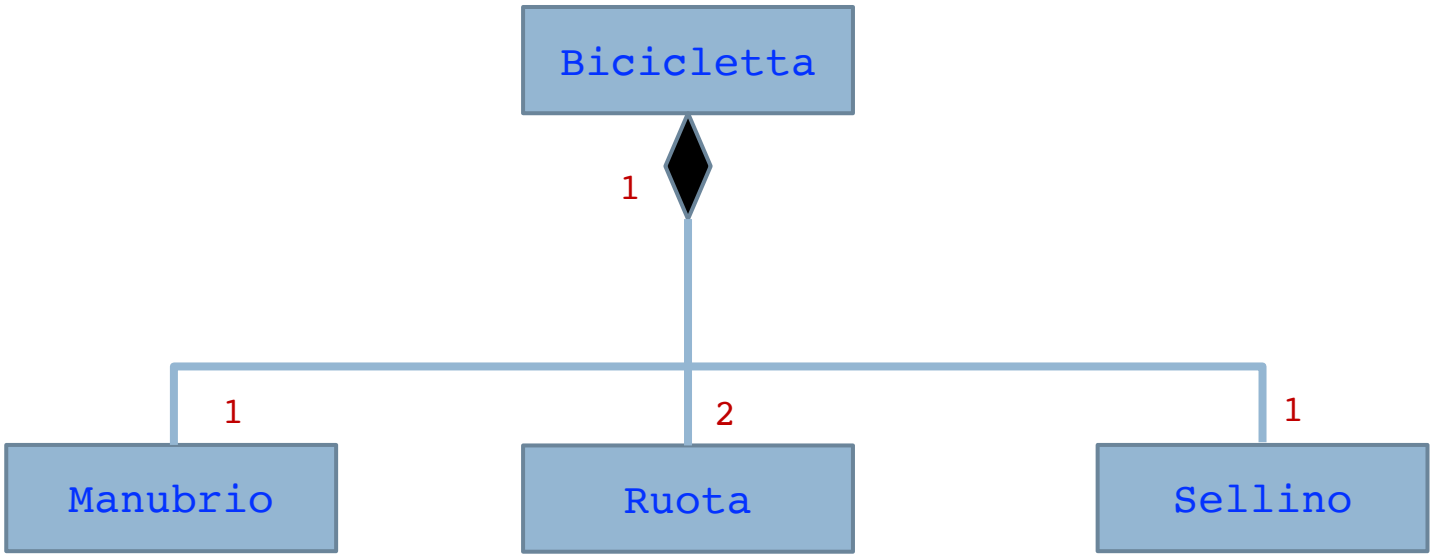
Esempio di generalizzazione

Generalizzazione e vincoli

- Vincoli
  - Overlapping
    - un'istanza appartiene a due sottoclassi di due set distinti
  - Disgiunta
    - Un'istanza appartiene ad un'unica sottoclasse
  - Completa
    - Tutte le possibili istanze della superclasse appartengono a una delle sottoclassi definite
  - Incompleta
    - Almeno un'istanza della superclasse non appartenga alle sottoclassi definite nella generalizzazione

Composizione e aggregazione

Relazione di aggregazione

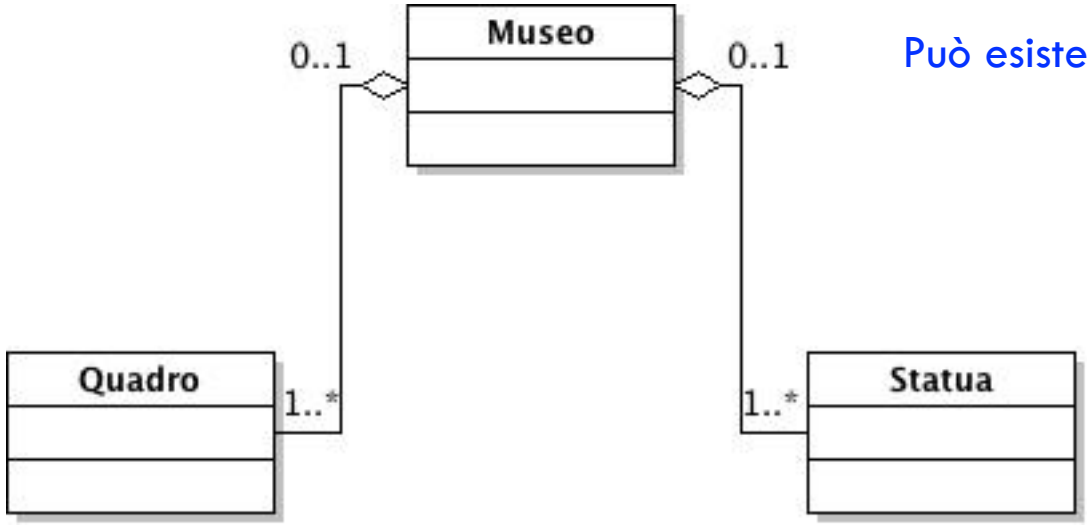


Relazione di composizione (solo parti intere)

Composizione vs Aggregazione

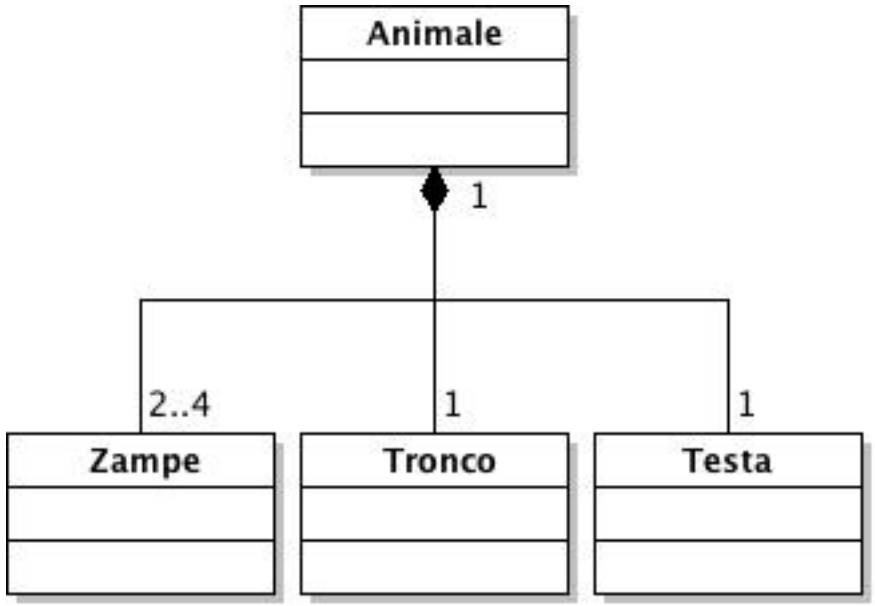
- aggregazione
  - relazione non forte
    - relazione nella quale le classi parte hanno un significato anche senza che sia presente la classe tutto
- composizione
  - relazione forte
    - relazione nella quale le classi parte hanno un reale significato solo se sono legate alla classe tutto

Composizione vs aggregazione



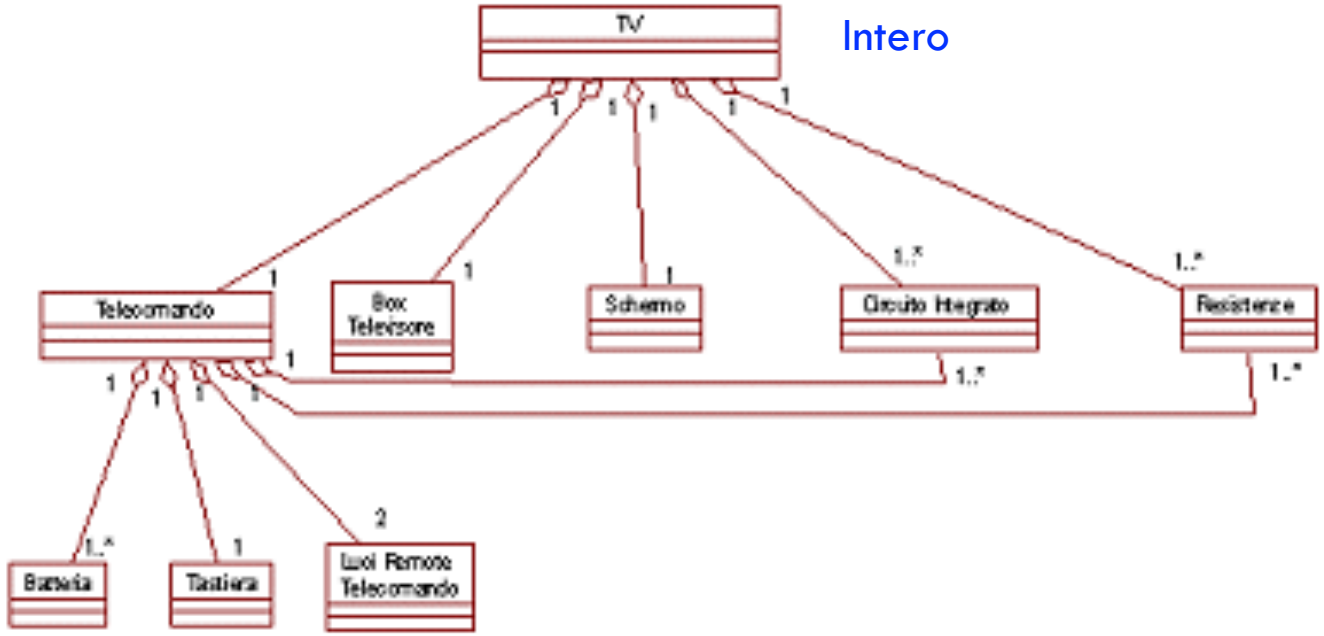
Può esistere senza alcune classi parte

Relazione di aggregazione

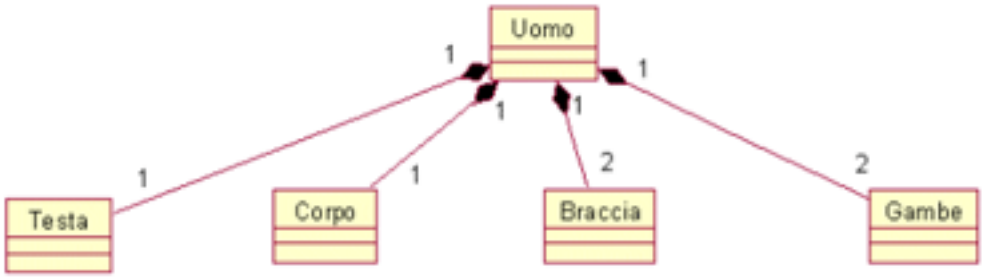


Relazione di composizione
(appartenenza solo ad un intero)

Composizione e aggregazione

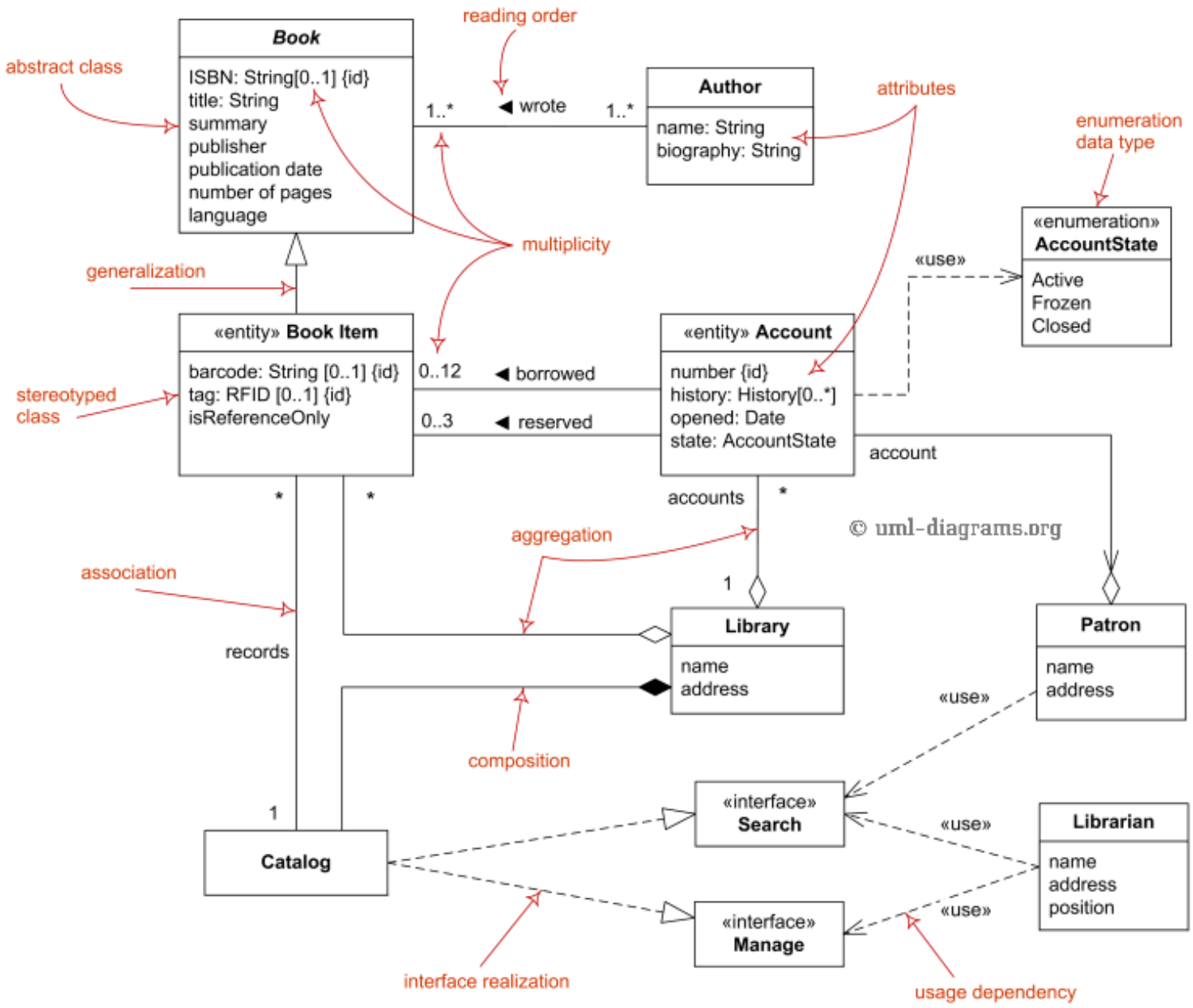


Relazione di aggregazione

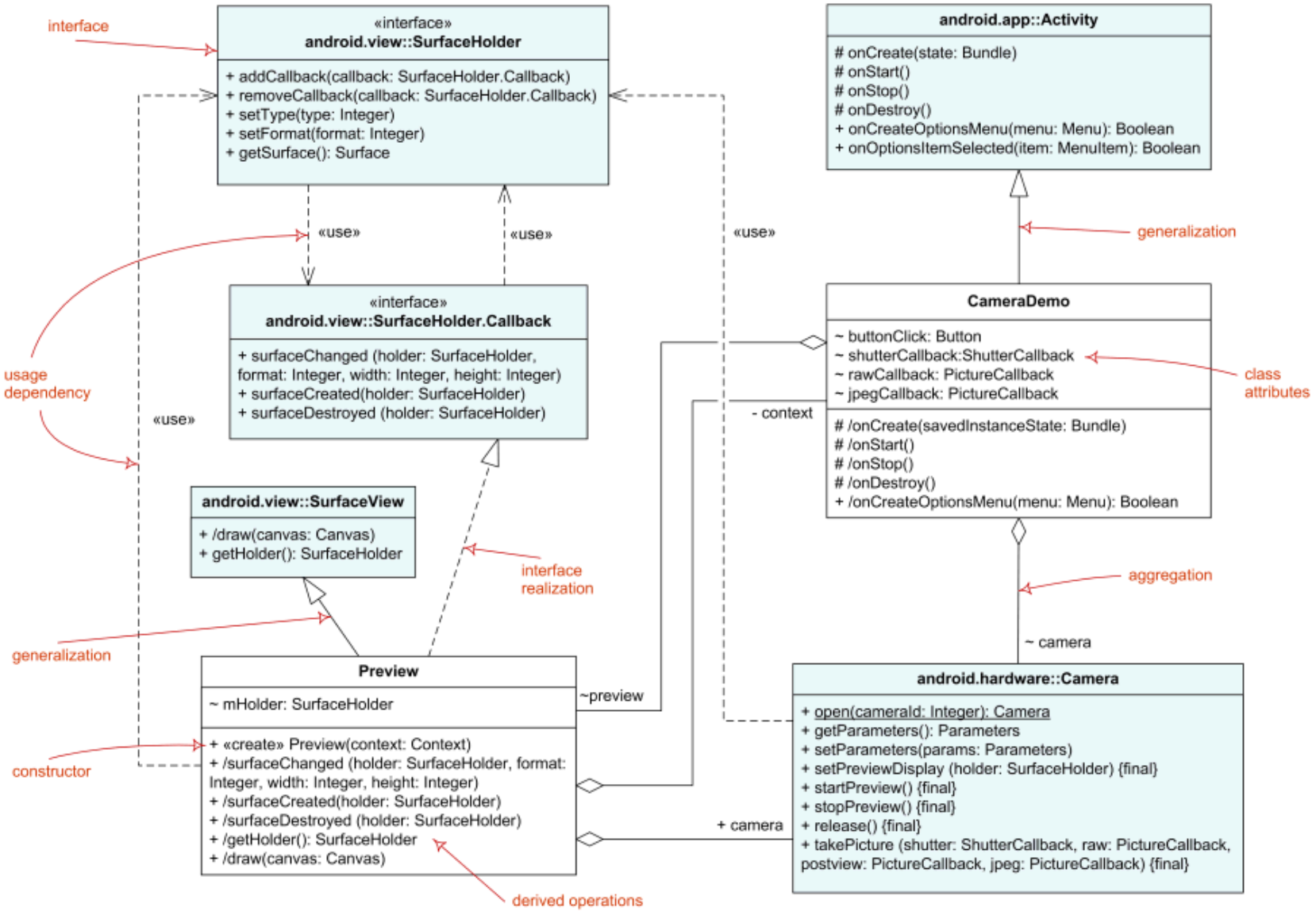


Relazione di composizione (appartenenza solo ad un intero)

Domain diagram

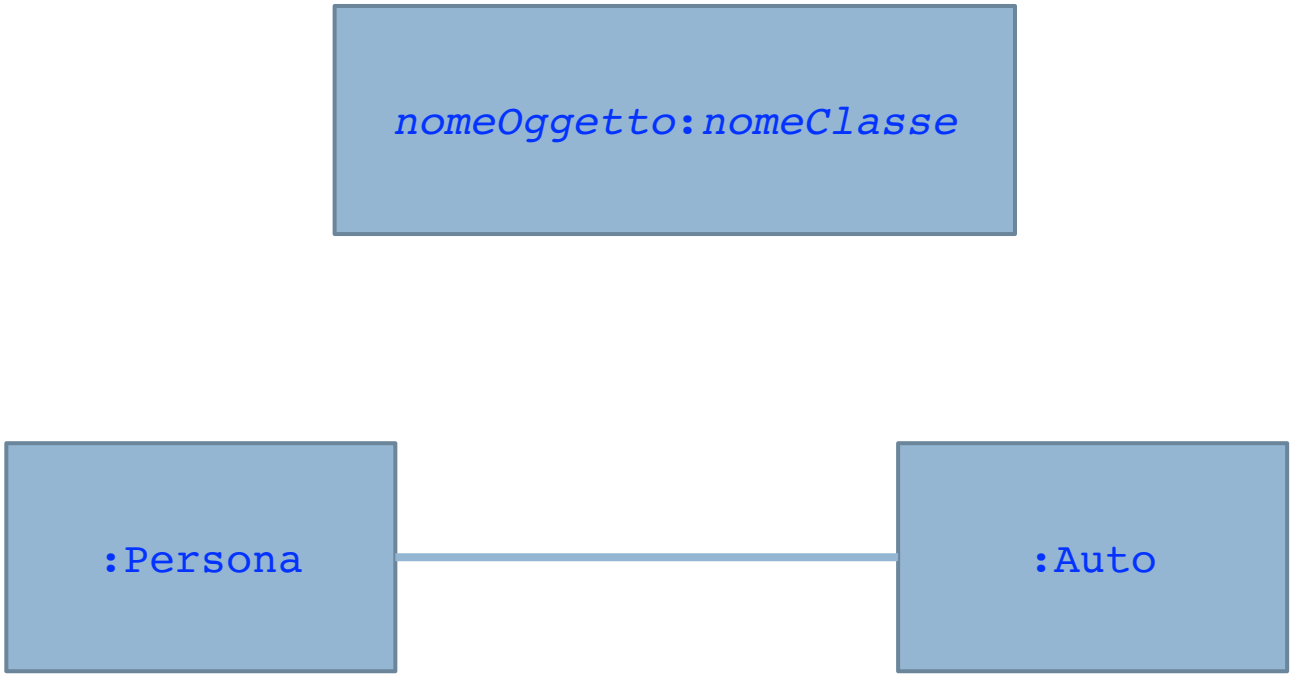


Class diagram



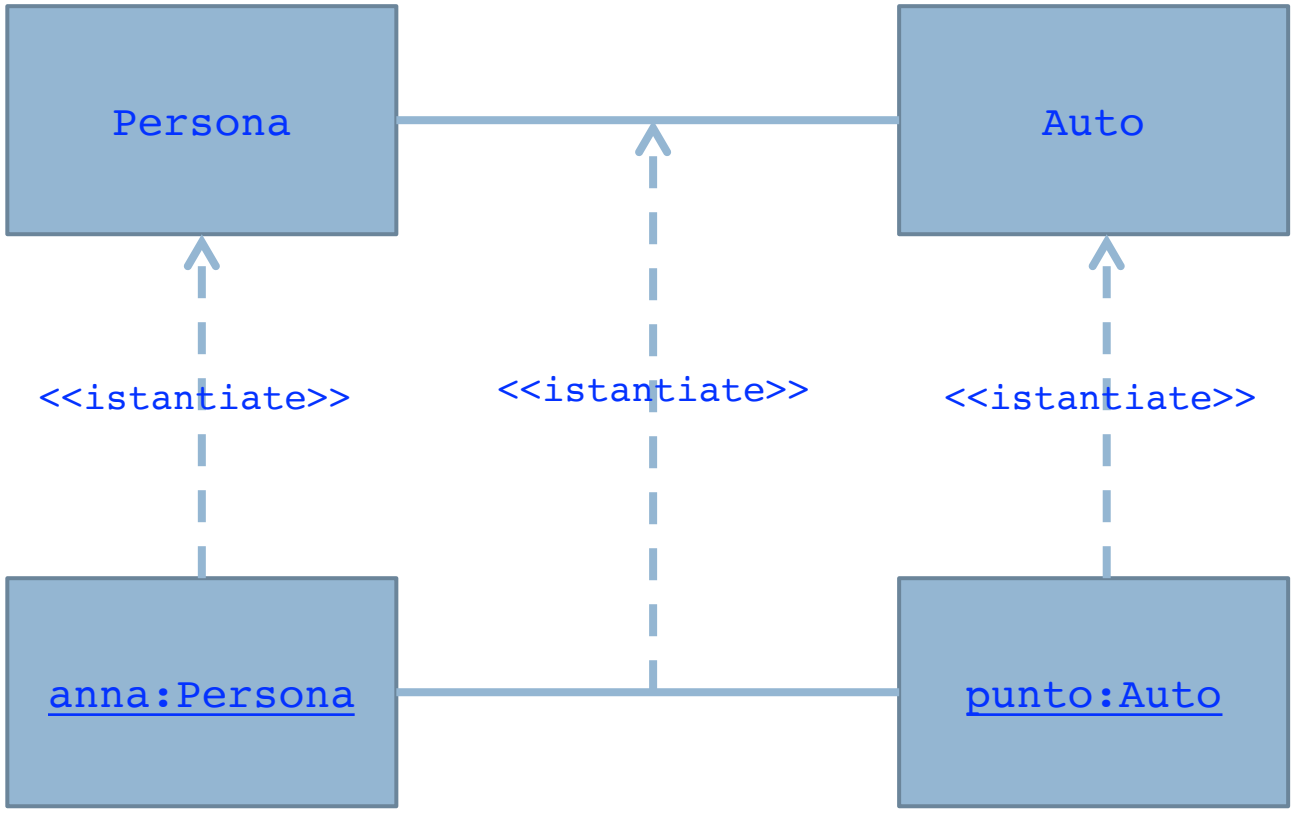
Oggetti e link

- Un oggetto è un'istanza di una classe



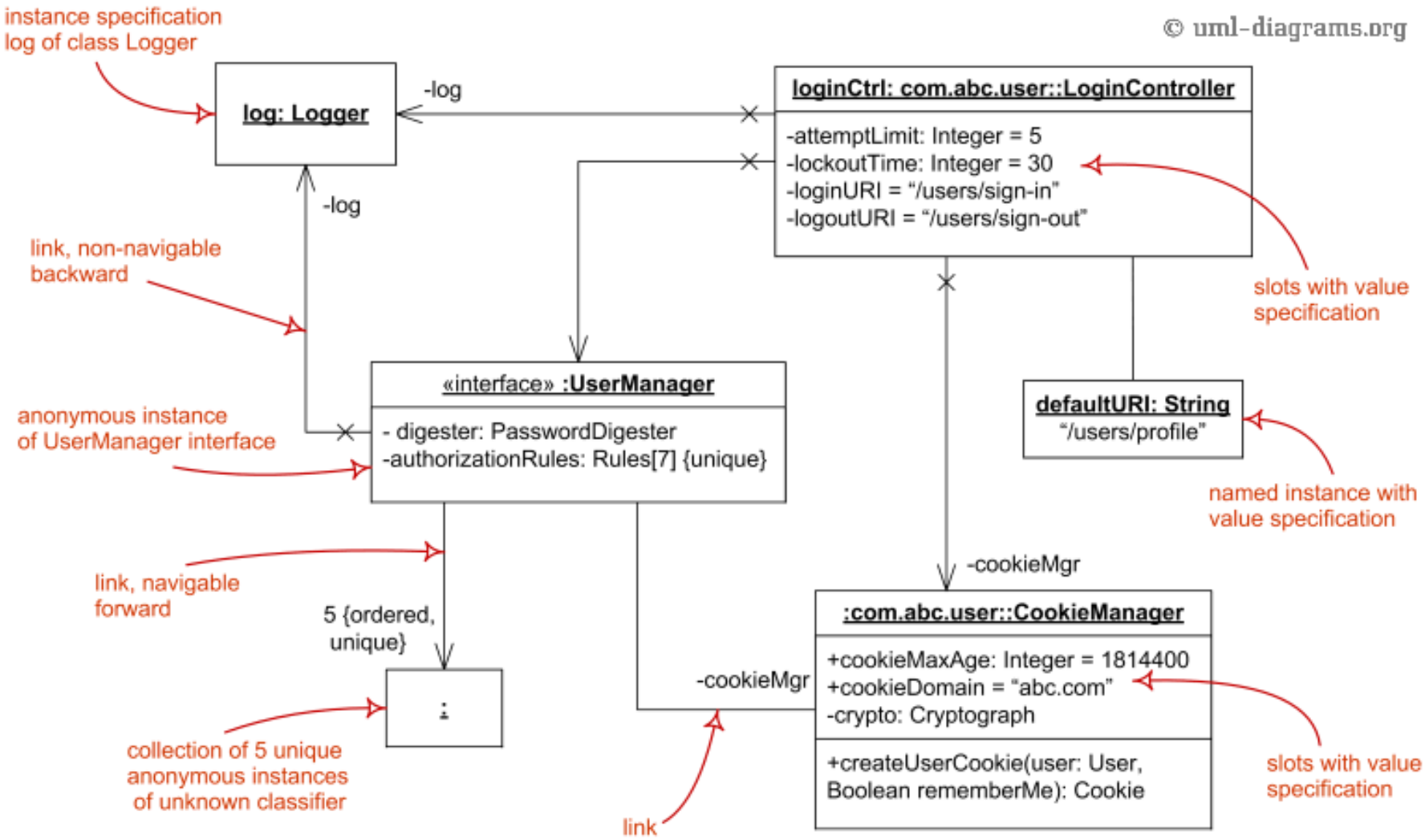
Oggetti e link

Object diagram



Esempio di Object Diagram

Esempio di Object diagram

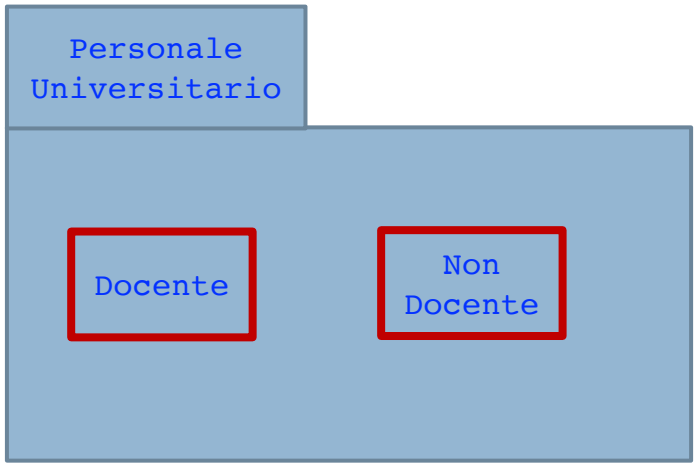


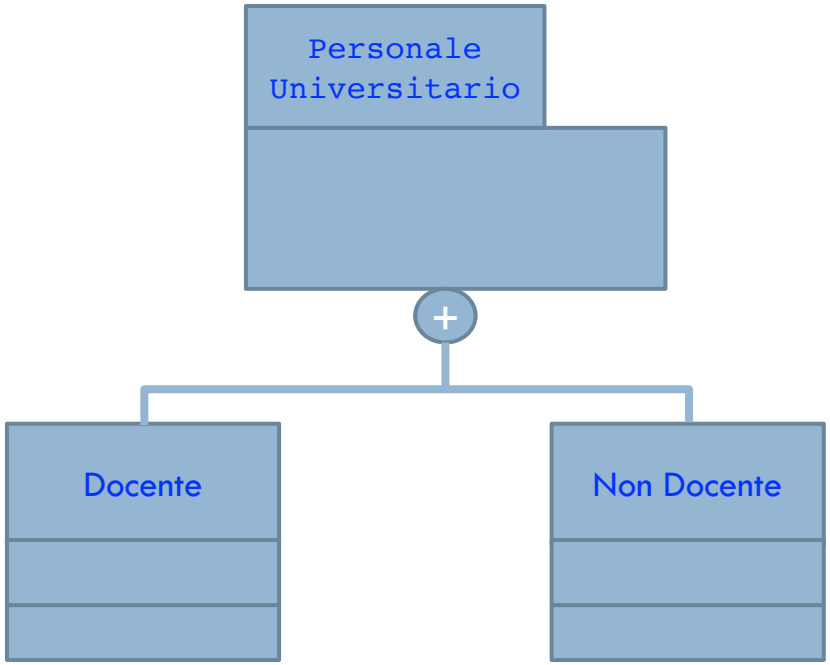
Package diagram

- Package Diagram
  - Permette di raggruppare più entità mostrandone le mutue relazioni
  - Permette di operare a un livello di astrazione superiore
    - Spesso usato per raggruppare le classi definite in un class diagram
  - L'elemento principale è il package

Rappresentazione di un package

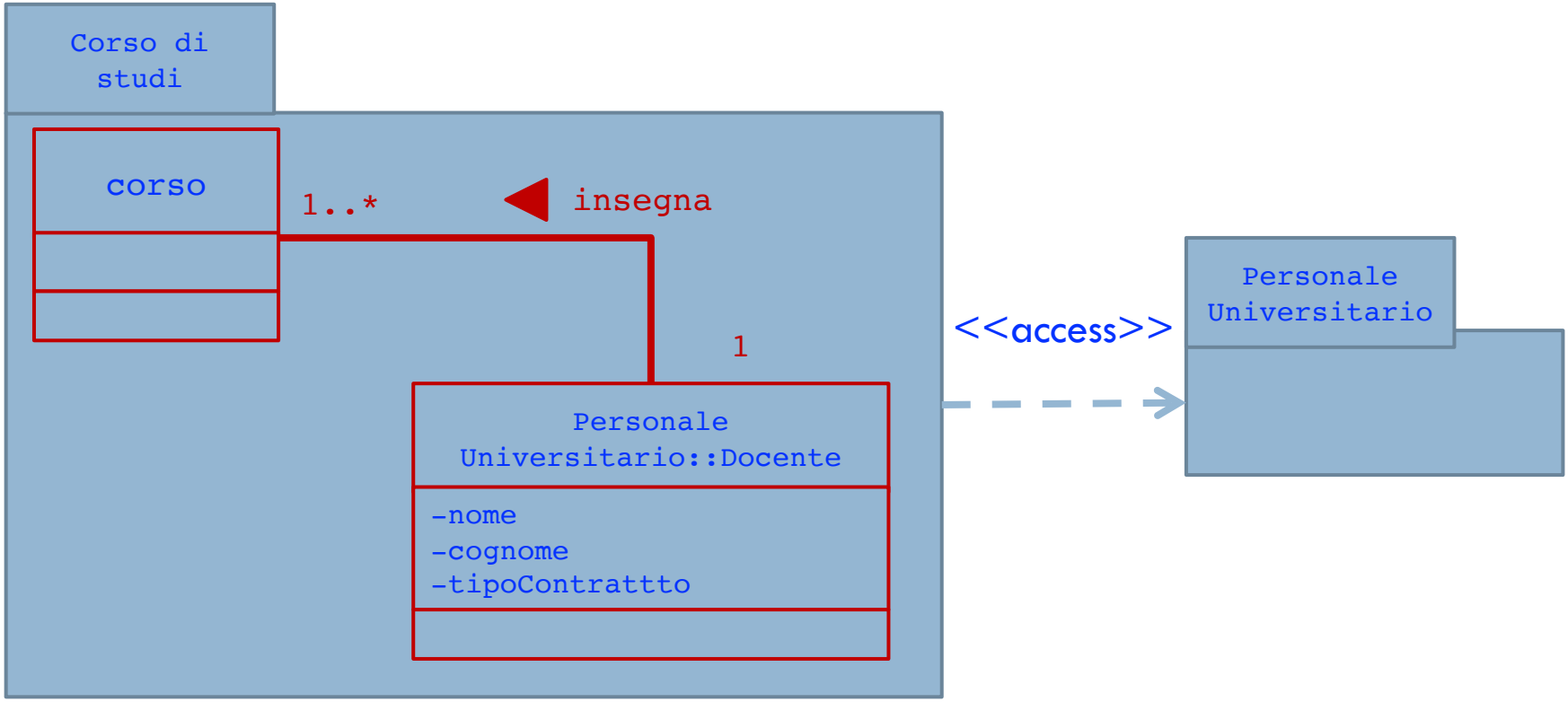
Package con classi





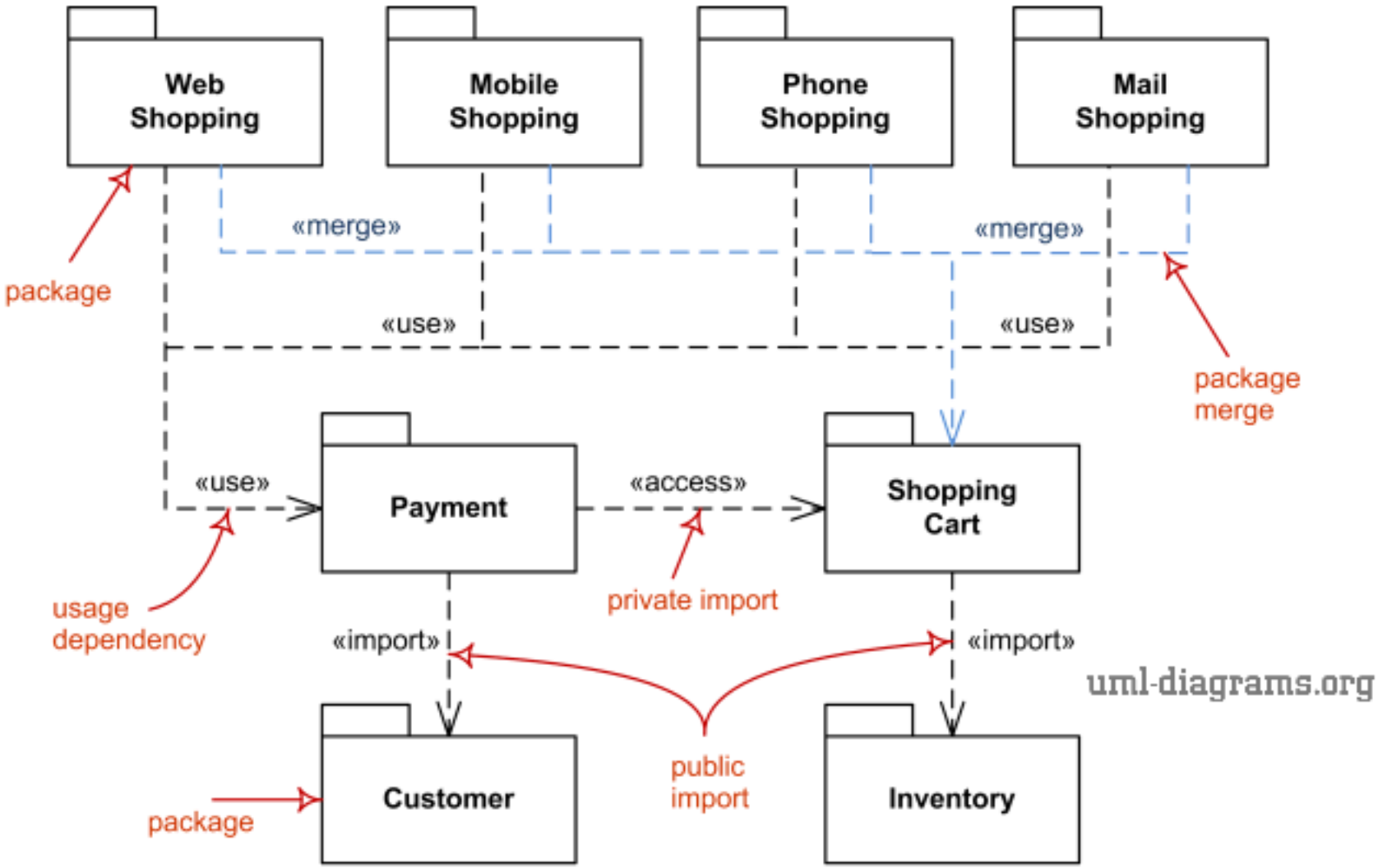
Package contenente classi

Dipendenza <<access>>

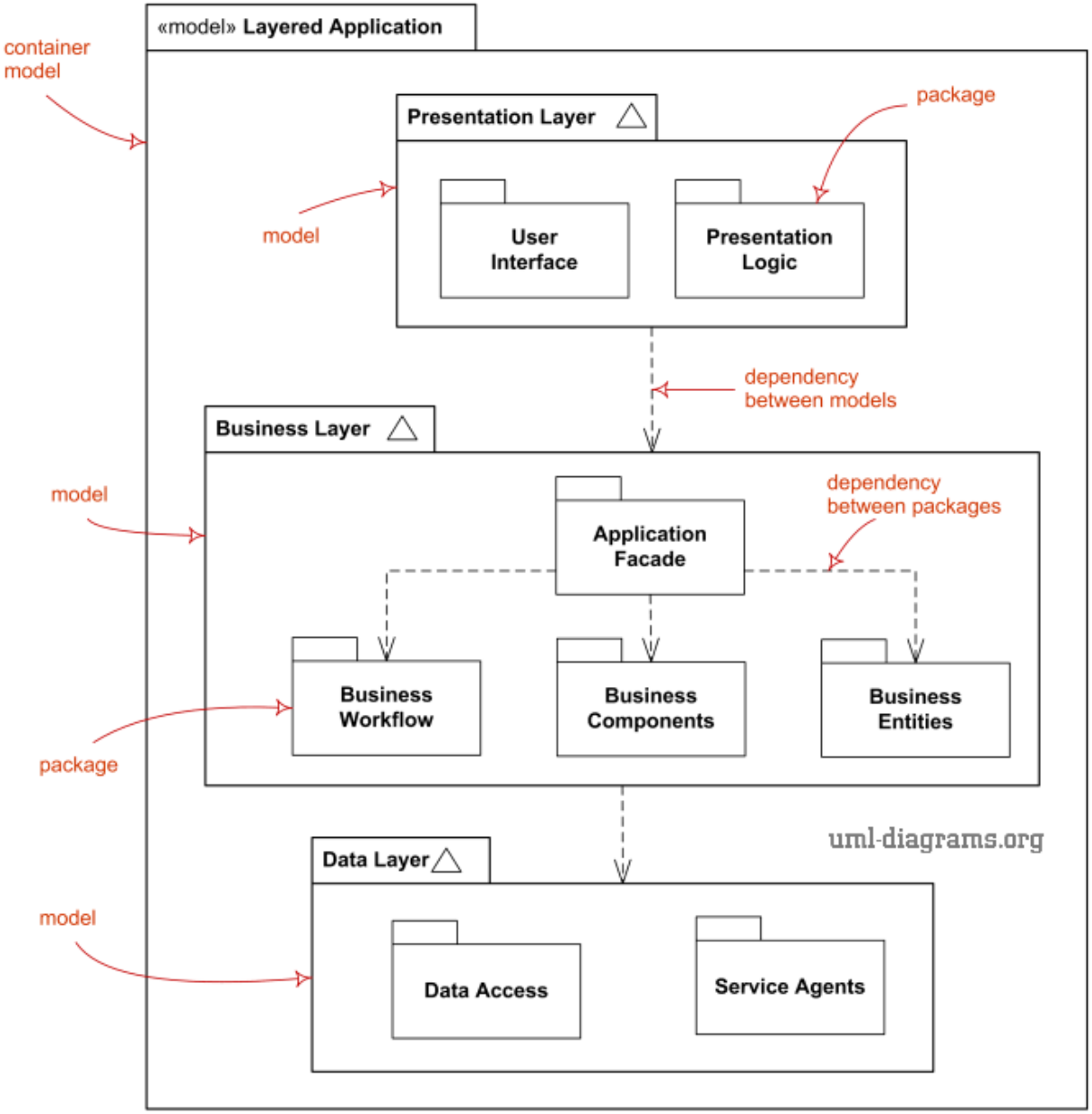


Dipendenza <<access>>: si usano attributi e operazioni di un altro package

Package Diagram



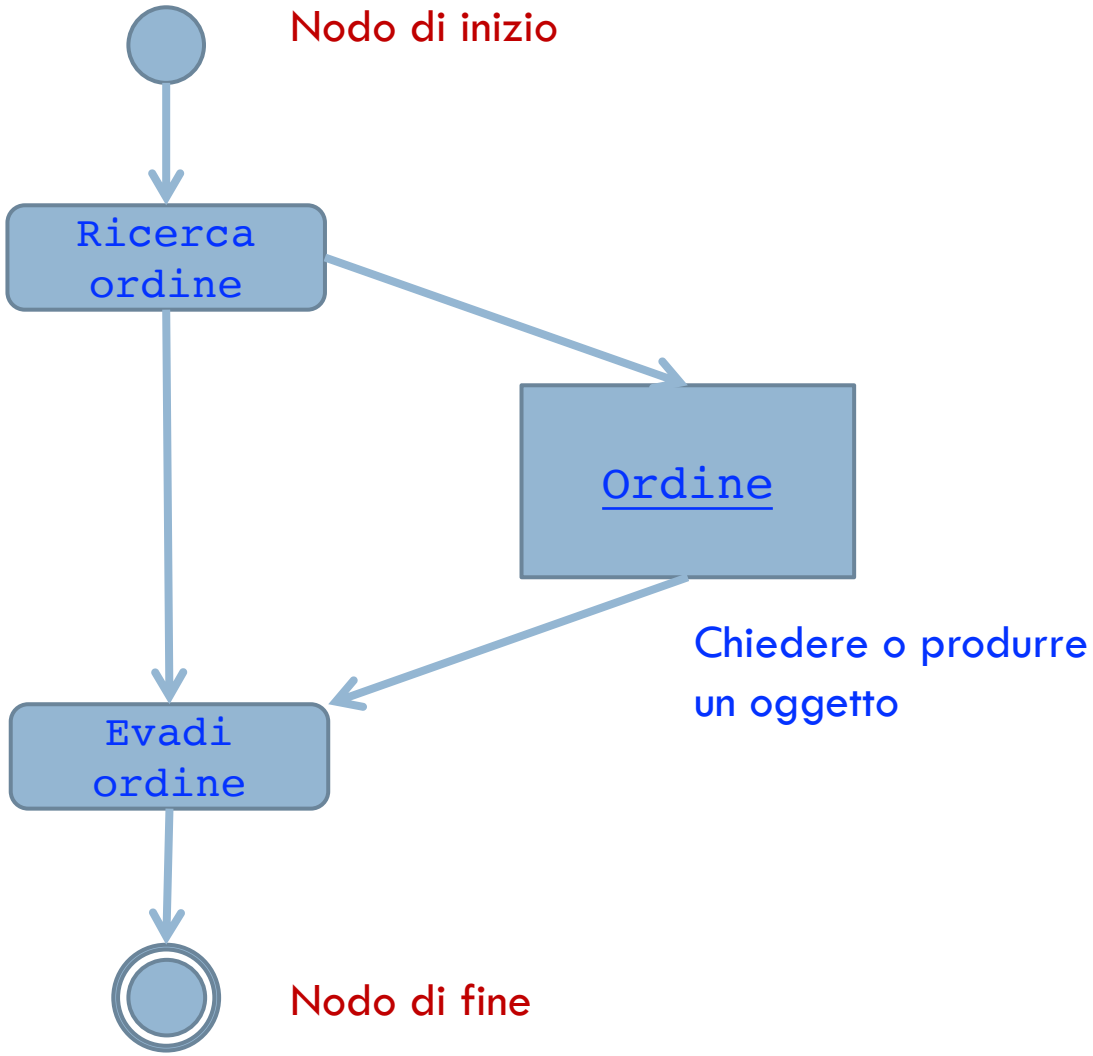
Model Diagram



Activity diagram

- Activity Diagram
  - descrive il flusso di azioni necessarie per eseguire una particolare procedura
- Elementi principali
  - nodi azioni
  - archi
  - nodi di inizio e fine
  - nodi di controllo
  - nodi oggetto
  - segnali
  - partizione

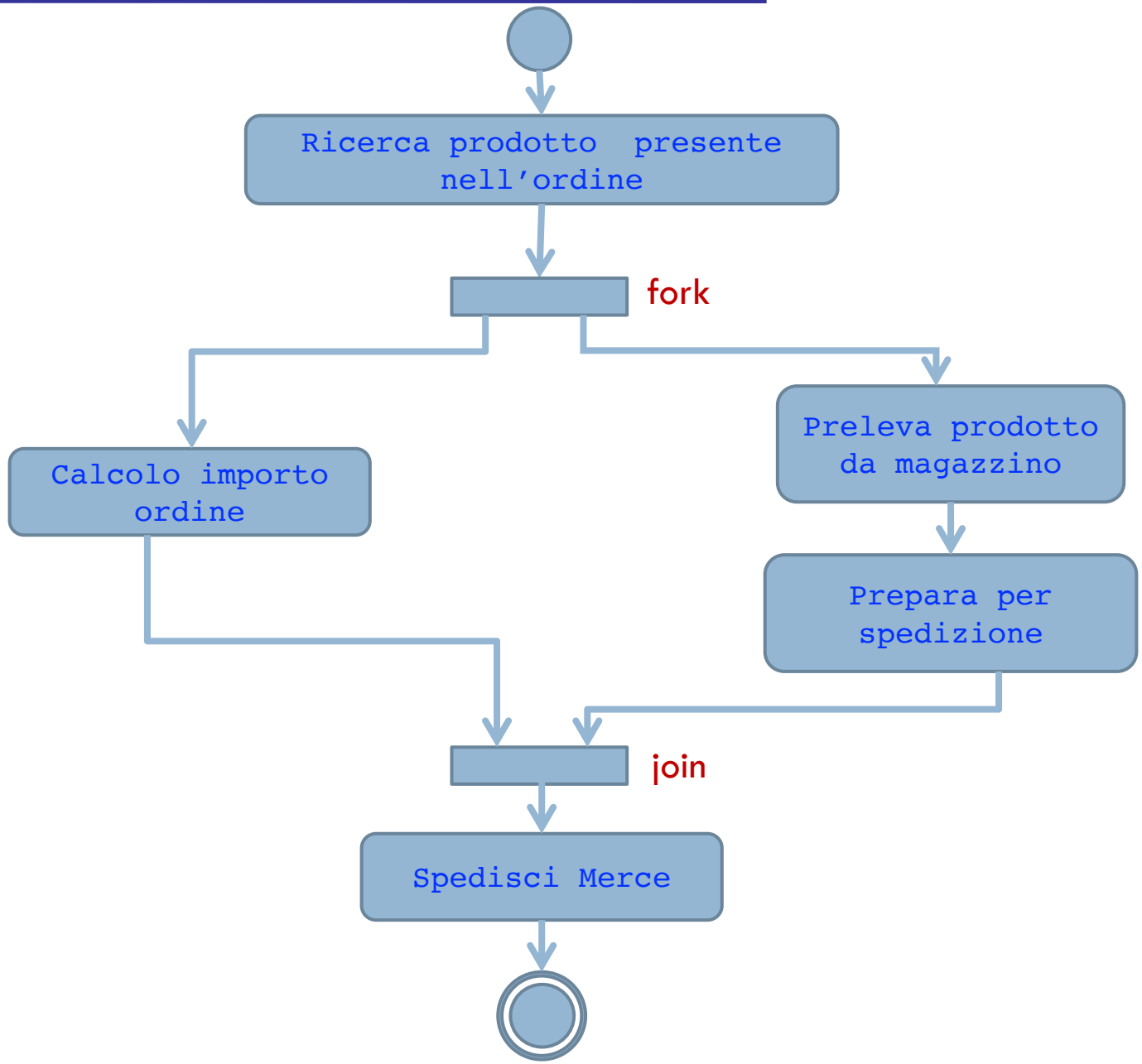
Semplice Activity diagram



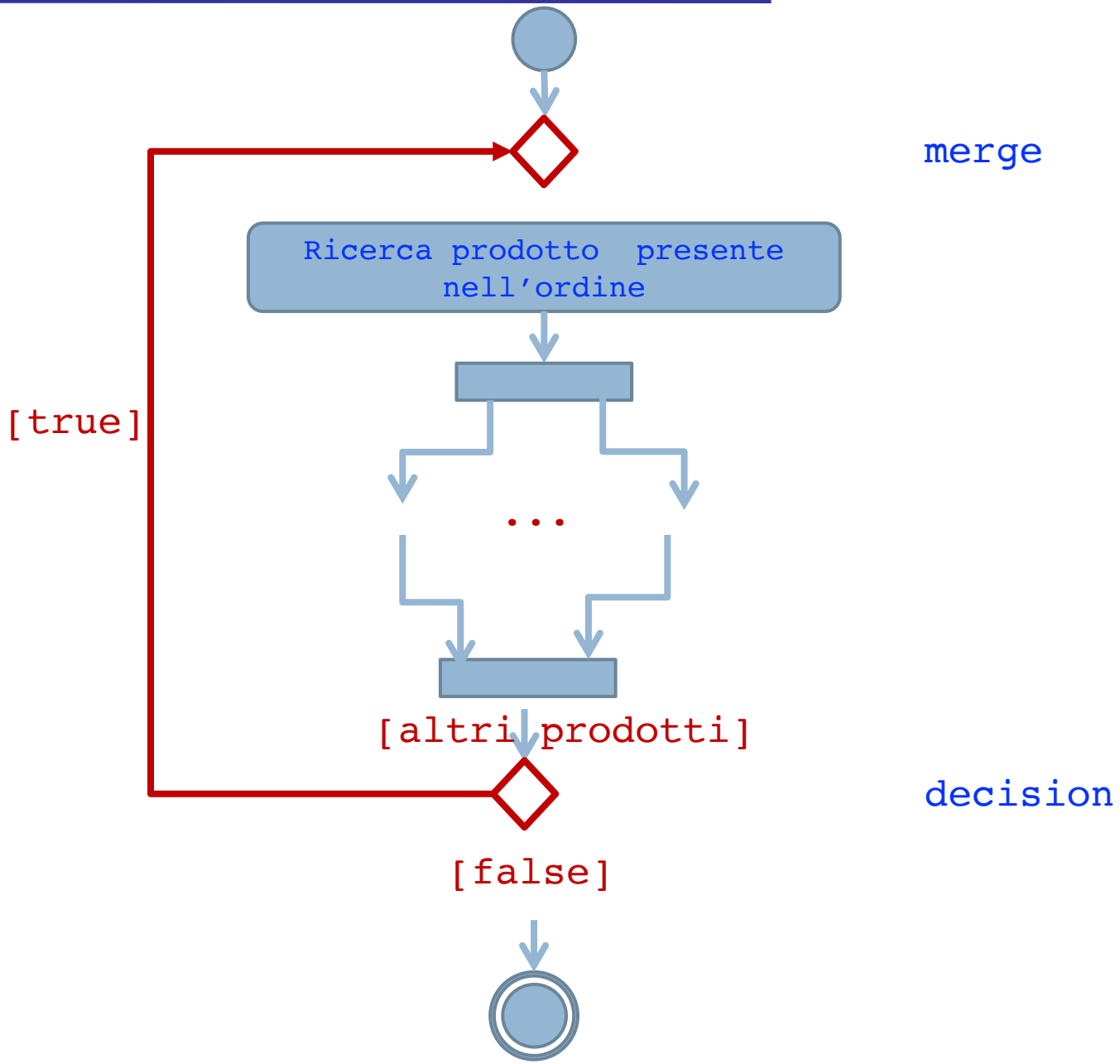
Nodi di controllo

- Nodi di controllo
  - Fork
  - Join
  - Decision
  - Merge

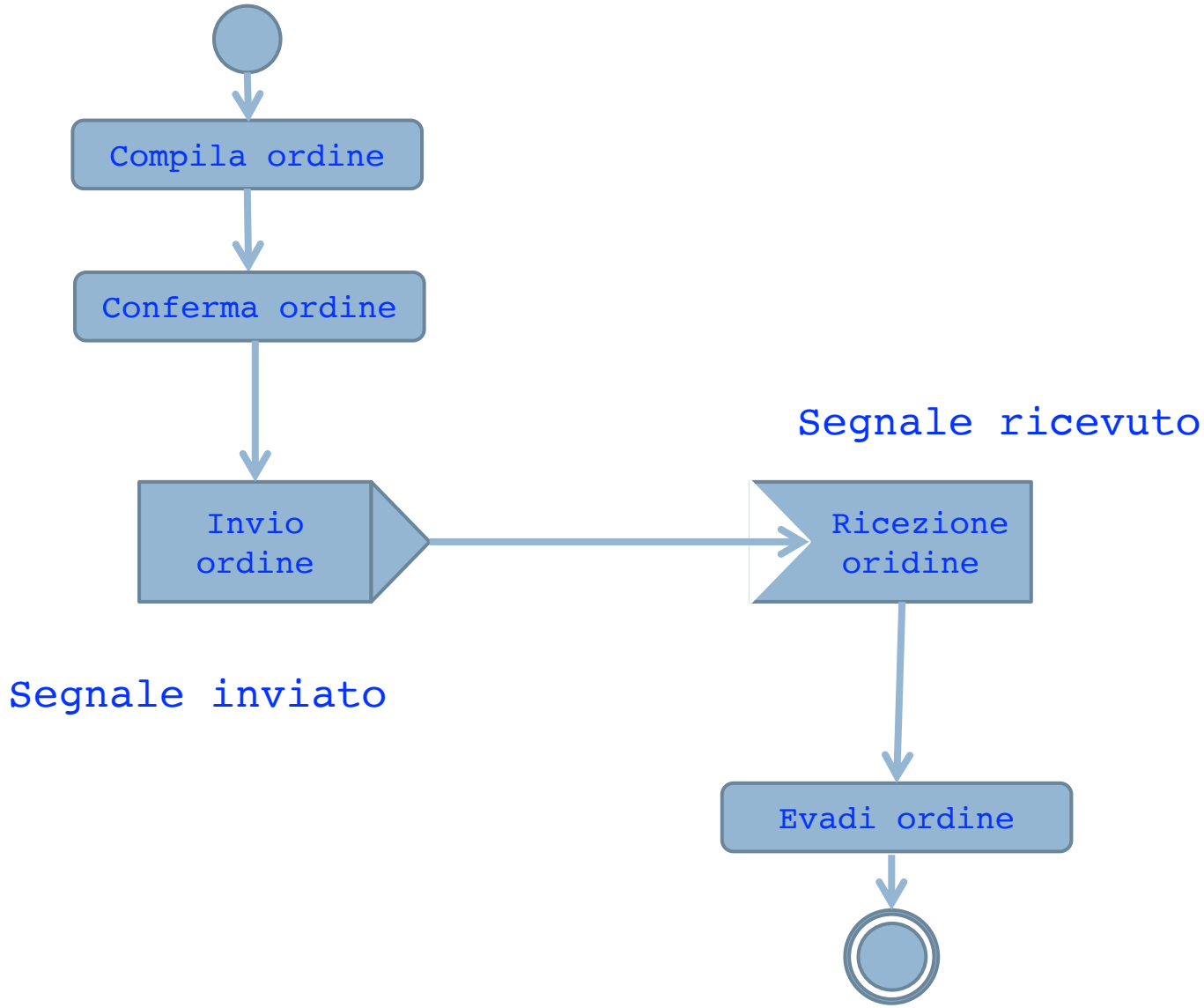
Fork e Join


Ricerca prodotto presente nell'ordine
fork
Calcolo importo ordine
Preleva prodotto da magazzino
Prepara per spedizione
join
Spedisci Merce

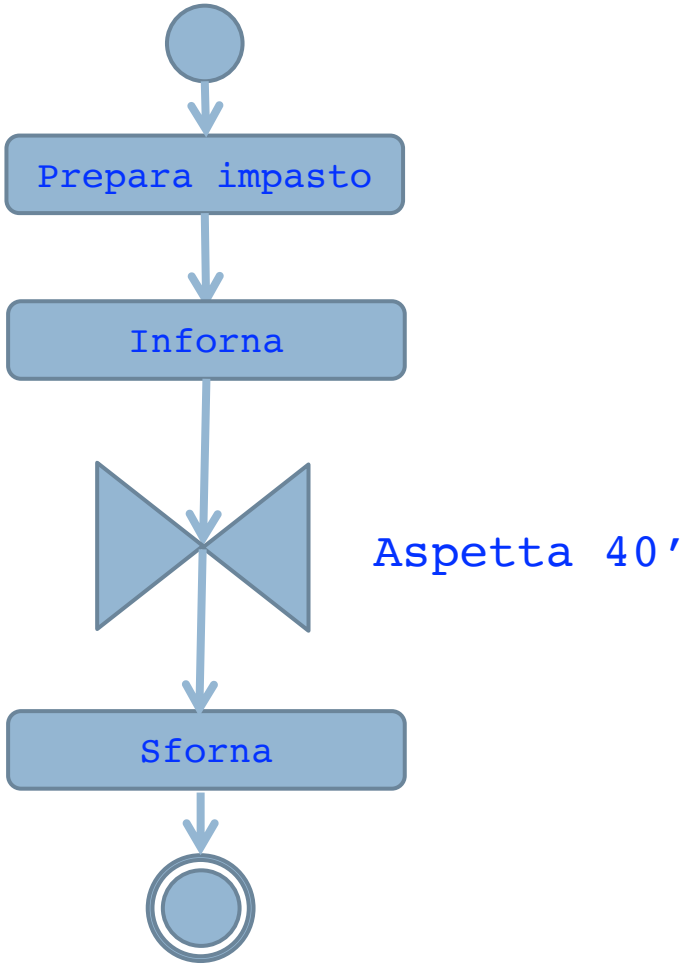
Decision e Merge



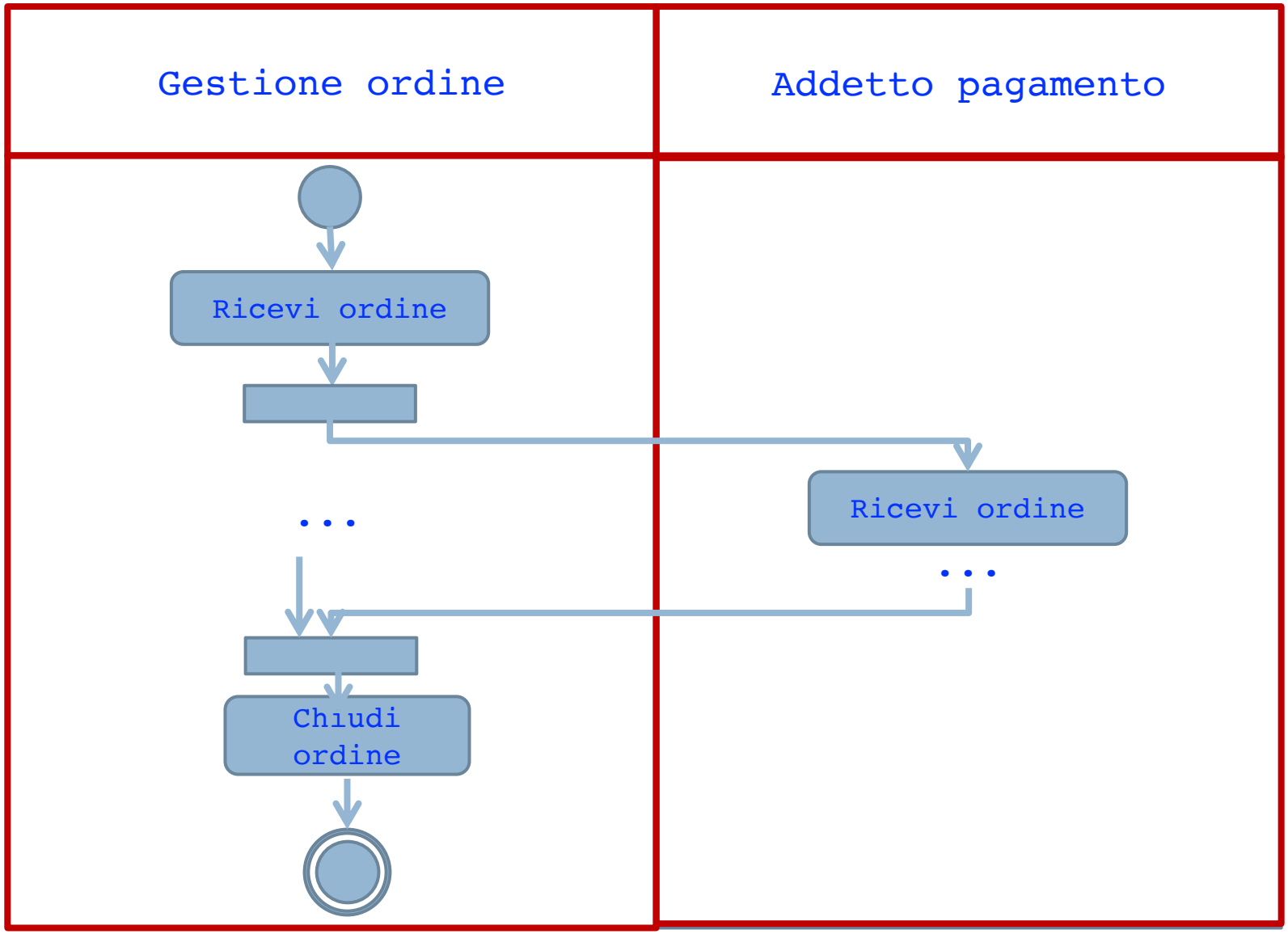
Segnali inviati e ricevuti



Segnale temporale

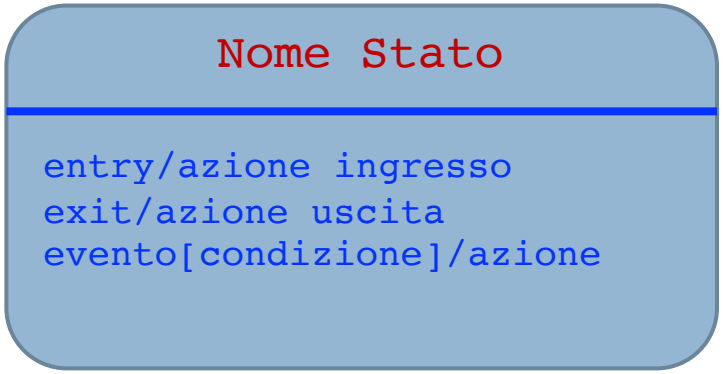


Partizioni


Gestione ordine
Addetto pagamento
Ricevi ordine
Ricevi ordine
Chiudi ordine

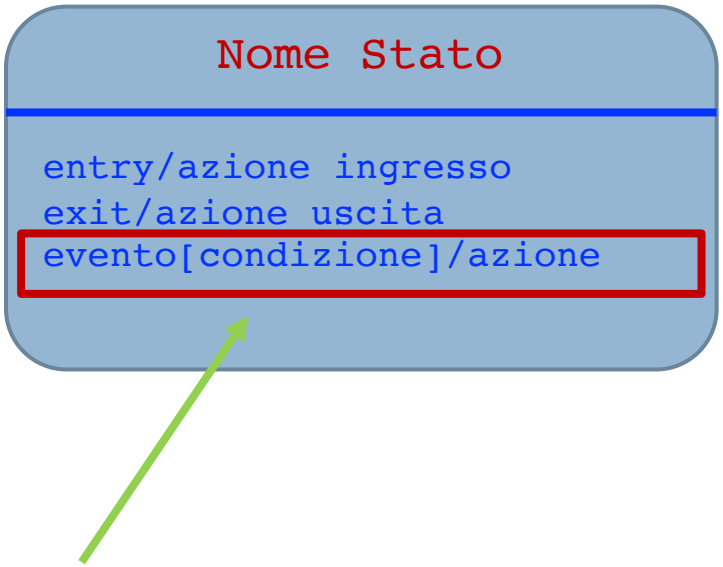
State Machine diagram

- State Machine diagram
  - rappresentano le caratteristiche di una qualunque entità attraverso la descrizione degli stati e passaggi di stato



Rappresentazione di uno stato

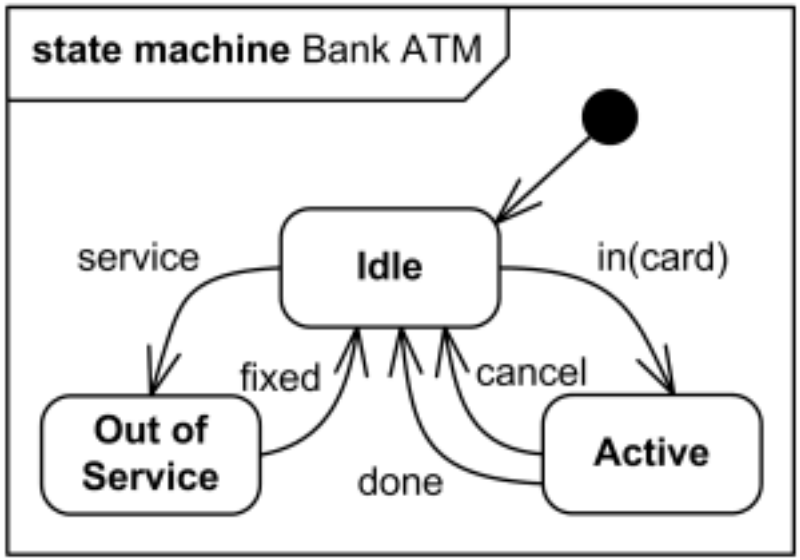
State Machine diagram



Attività interne

Anche le transizioni sono rappresentate in questa forma

State Machine Diagram

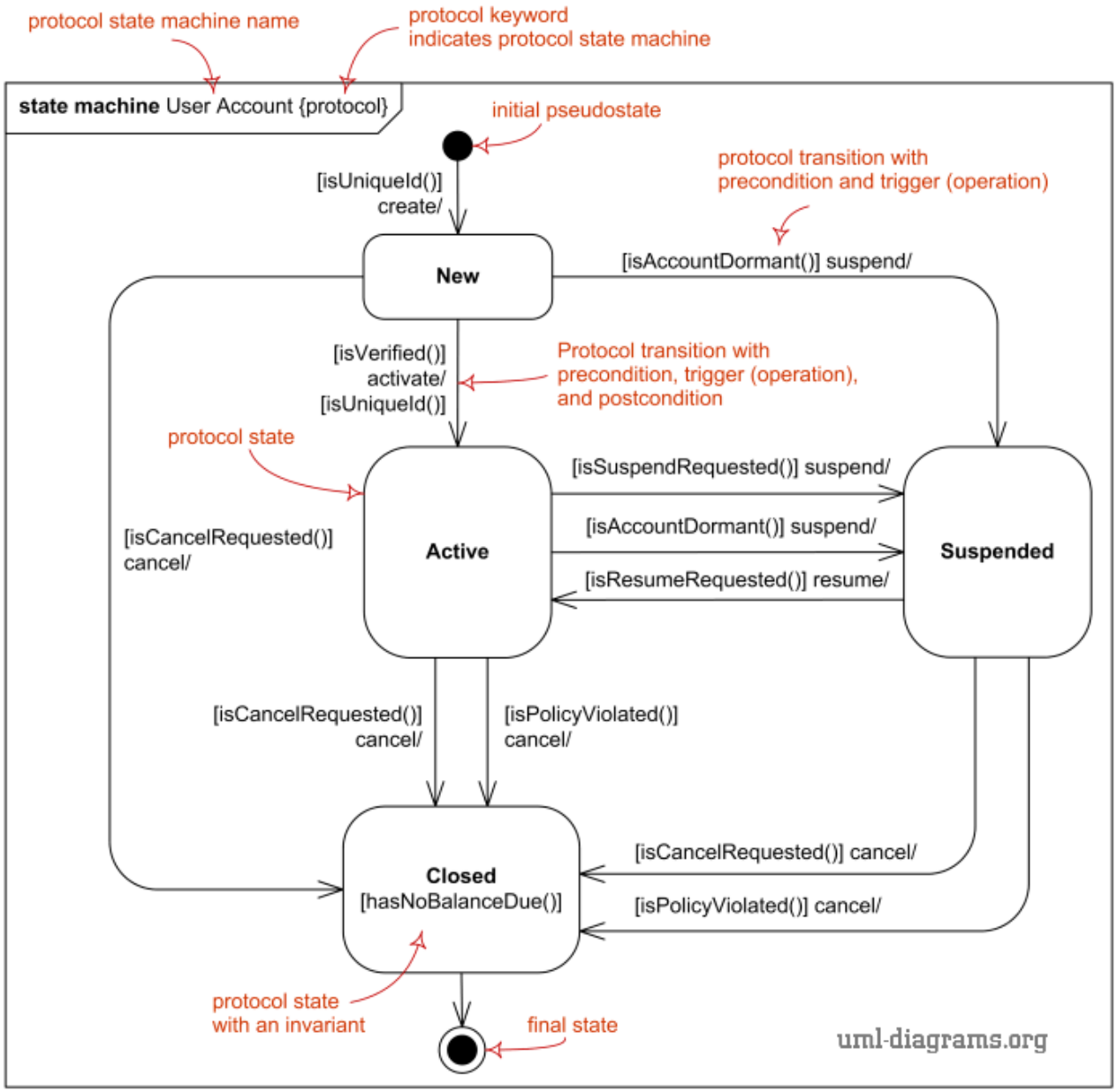


Uno State Machine Diagram per la gestione ad alto livello di un ATM

Stati

- Altri tipi di stato
  - Superstato
    - Contiene altri stati
  - Stato concorrente
    - Contiene altri stati validi contemporaneamente in modo concorrente
  - Pseudostati
    - Non hanno attività interna
    - Stati di inizio e fine

Protocol State Machine Diagram



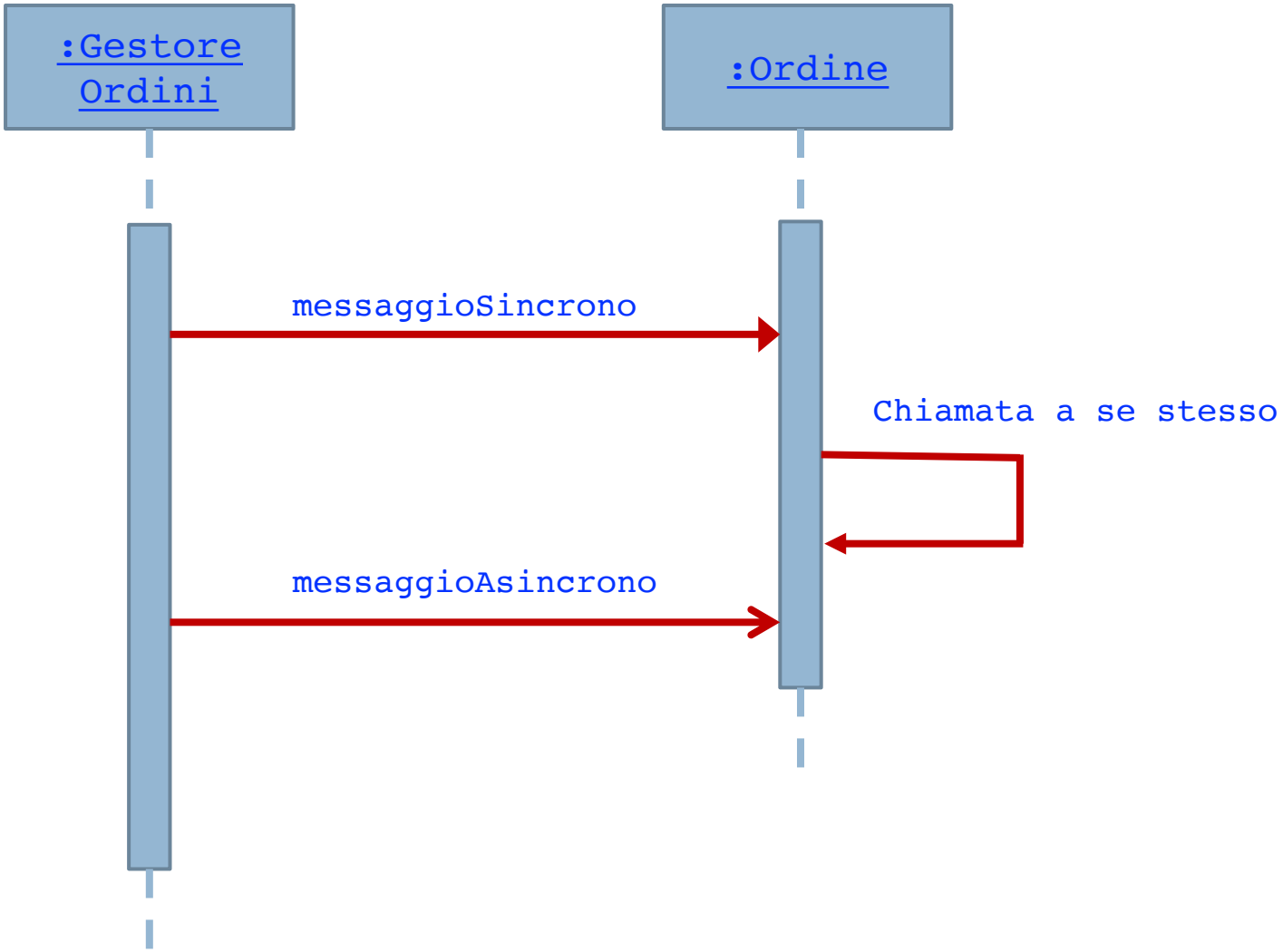
Sequence diagram

- Sequence diagram
  - Descrive l'interazione tra elementi come una sequenza temporale di
    - azioni
    - eventi
  - Gli elementi principali sono gli oggetti
    - ogni oggetto ha la sua linea di vita
    - per indicare un oggetto attivo si usa la barra di attivazione

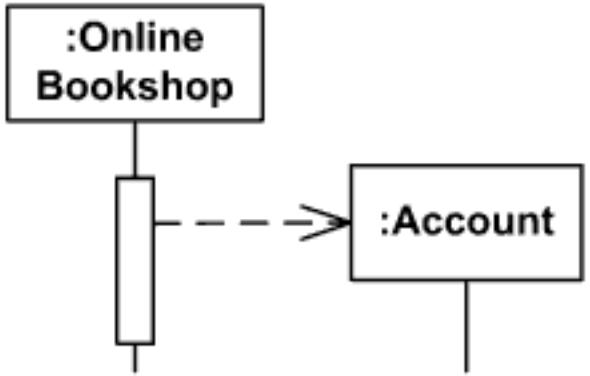
Messaggi

- La sequenza di interazioni tra oggetti è realizzata tramite messaggi
  - Sincroni
    - l'oggetto che invia il messaggio attende la risposta dell'oggetto ricevente e interrompe la propria esecuzione
  - Asincroni
    - l'oggetto che invia il messaggio non attende la risposta prima di continuare

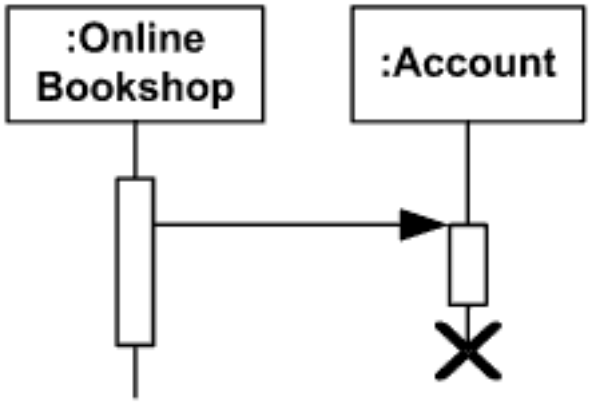
Messaggi


:Gestore Ordini
:Ordine
messaggioSincrono
Chiamata a se stesso
messaggioAsincrono

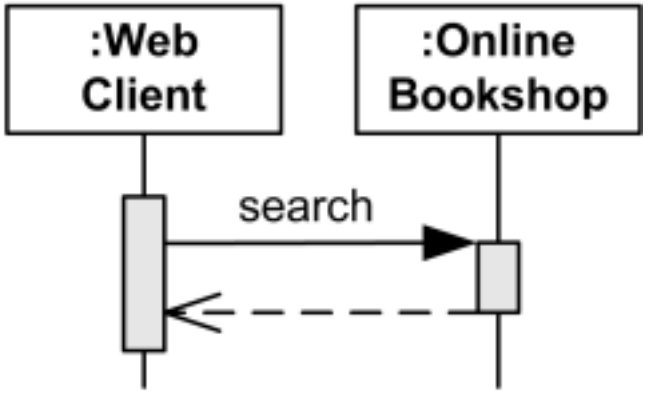
Messaggi



Creazione messaggio

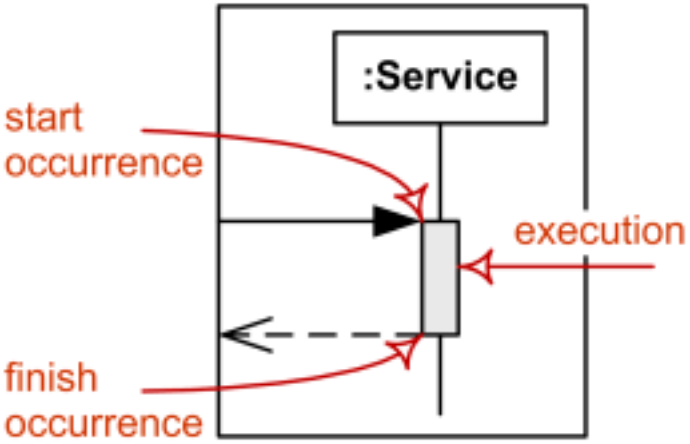


Distruzione messaggio

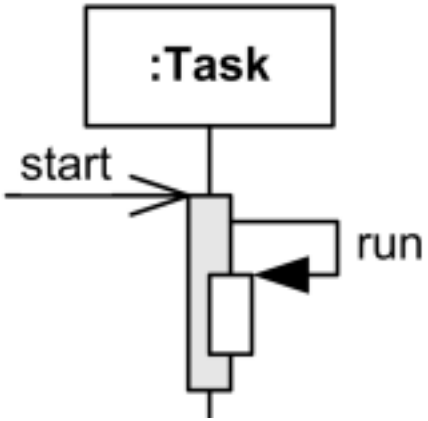


Replica al messaggio

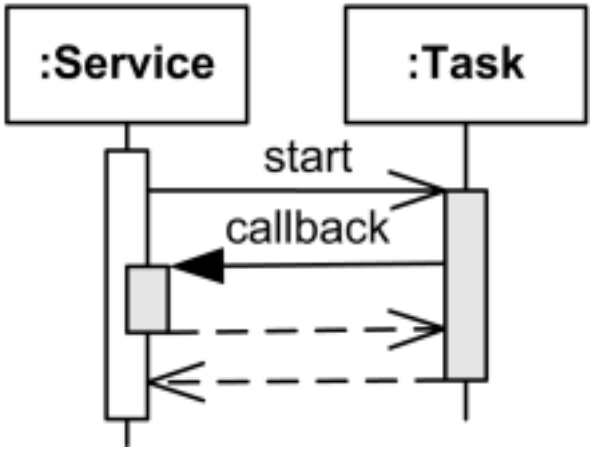
Esecuzione



Esecuzione di un'occorrenza

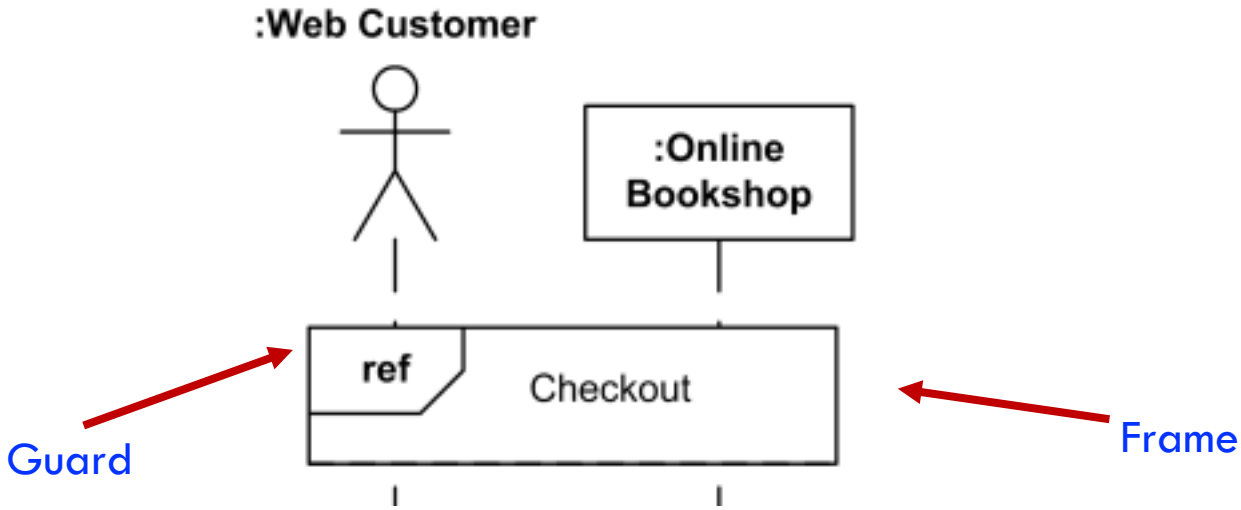


Messaggio a se stesso



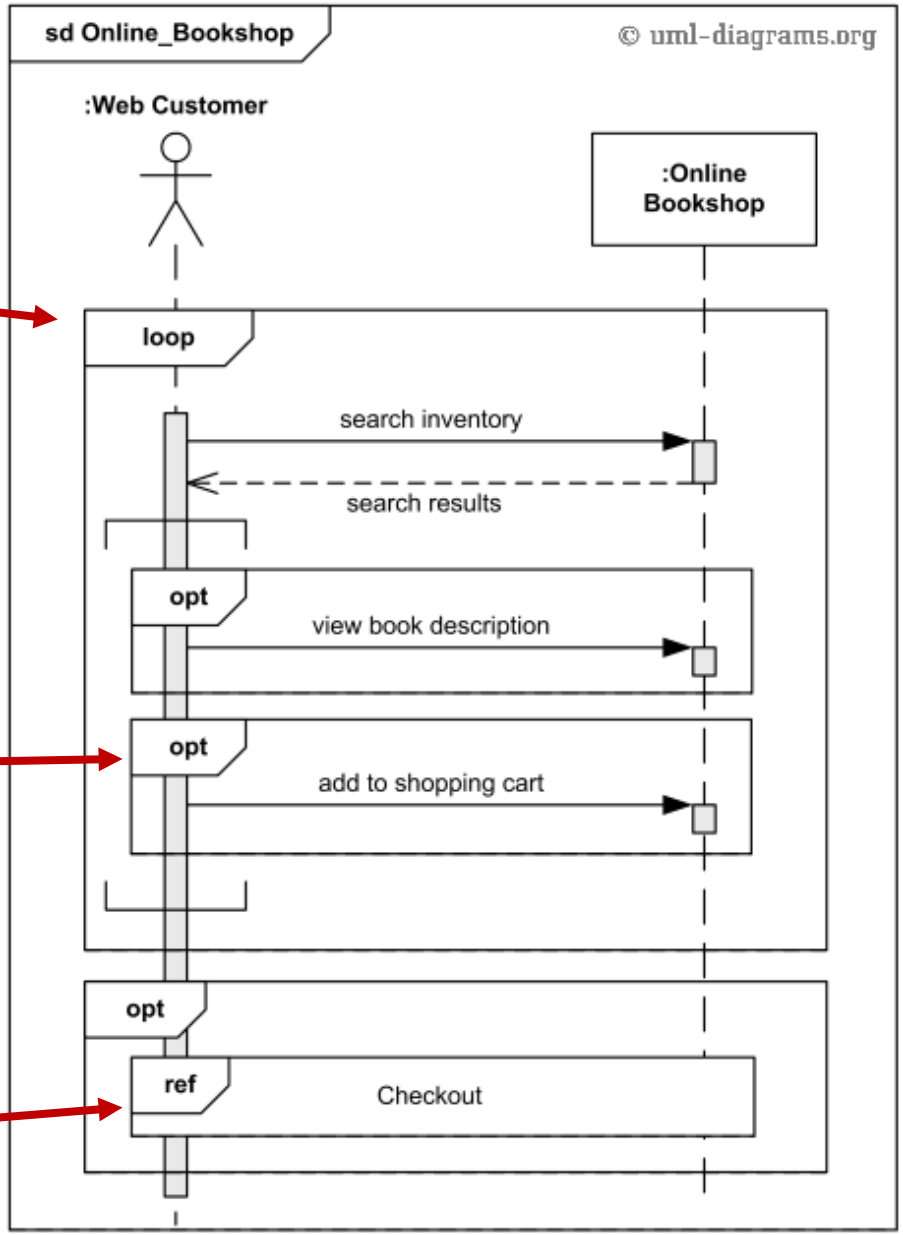
Messaggio callback

Interazione



Esempio di interazione

Online Bookshop



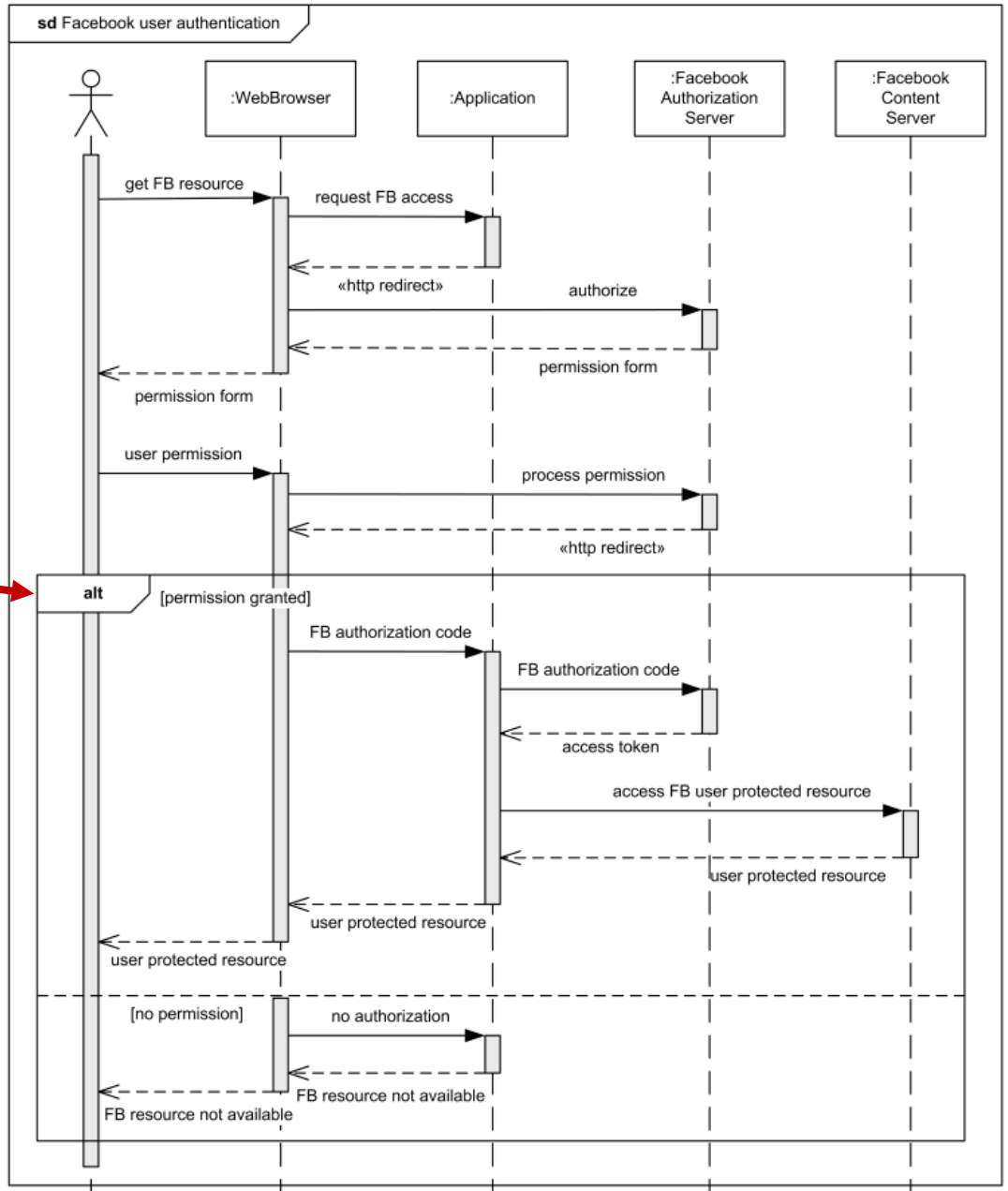
Ciclo

Opzionali

Richiamare altri SD

Autenticazione Facebook

Messaggi alternativi



Component Diagram

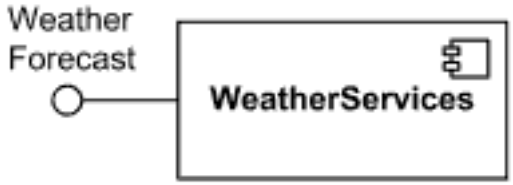
- Component Diagram
  - descrive un'entità complessa scomponendola in parti distinte in modo da evidenziare struttura e legami

Rappresentazione di componenti

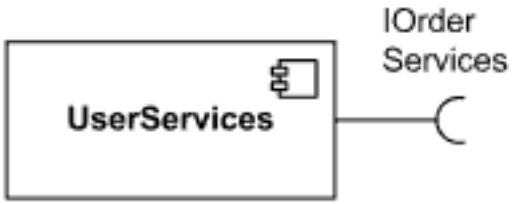
Stereotipi standard

- Stereotipi
  - «subsystem»
  - «process»
  - «service»
  - «specification»
  - «realization»
  - «implement»
  - «database»
  - «thread»
  - «infrastucture»

Interfacce

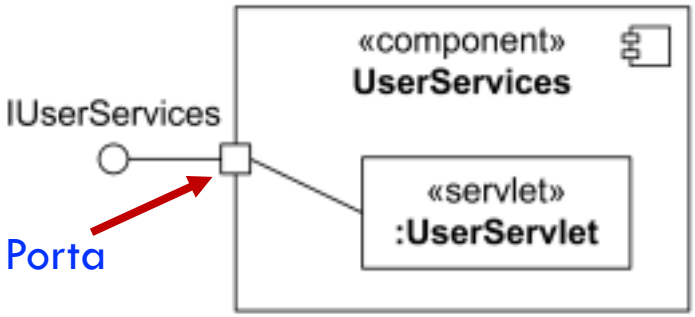


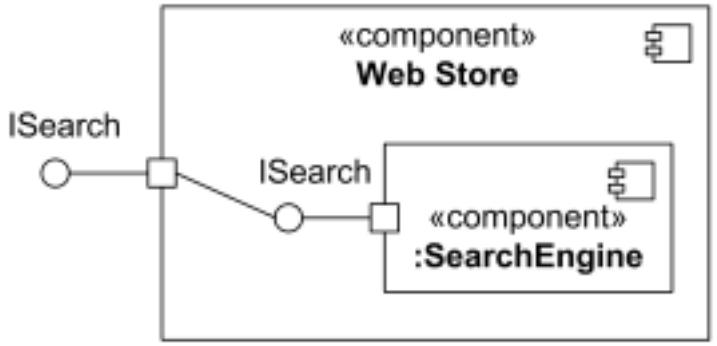
Interfaccia fornita

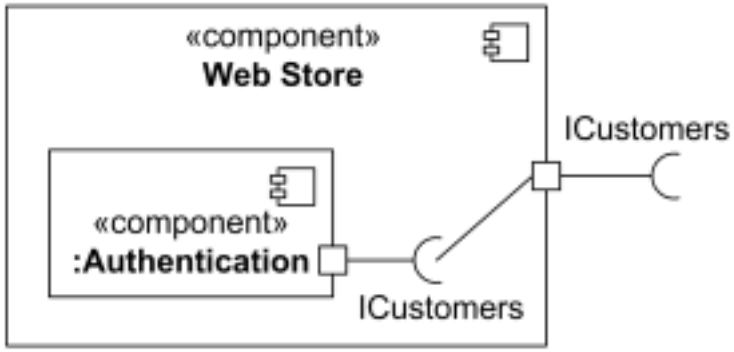


Interfaccia richiesta

Connettore delegazione

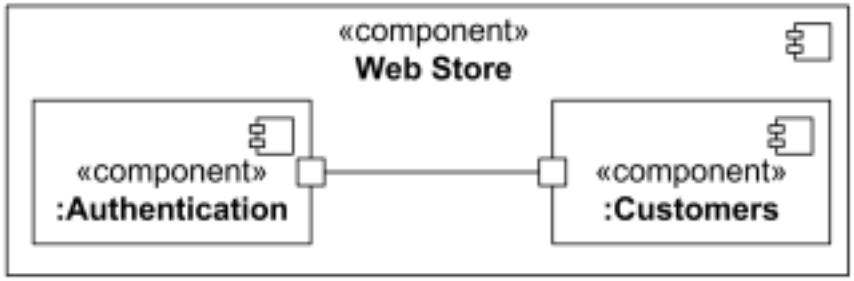


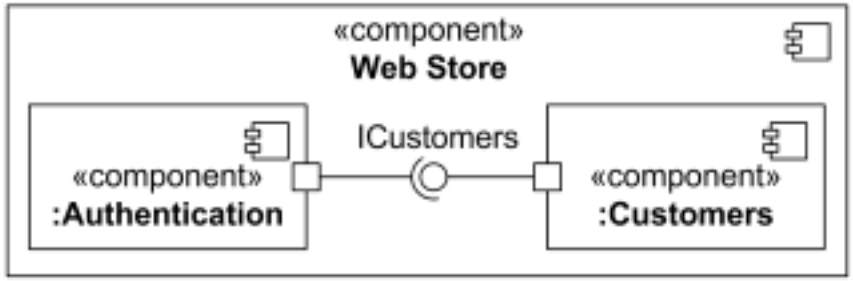


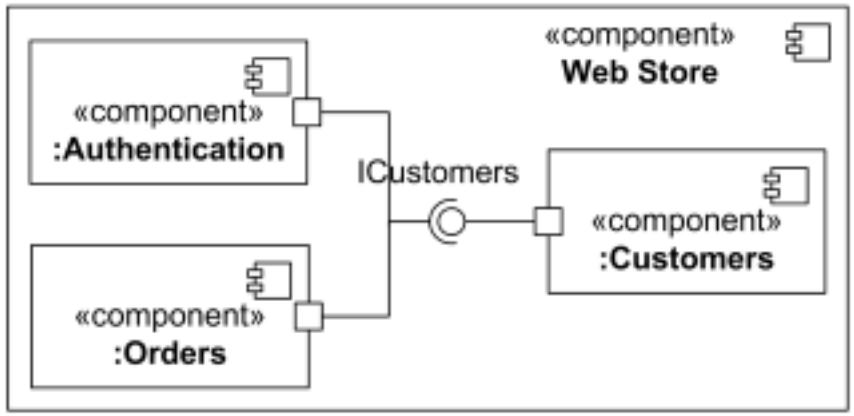


Esempi di connettori di delegazione

Connettore assemblaggio

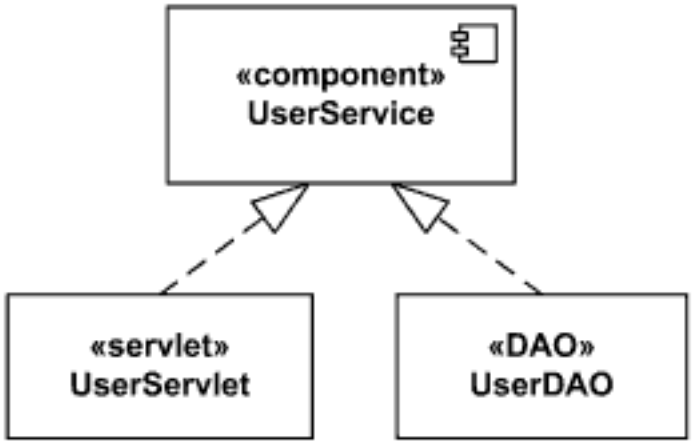






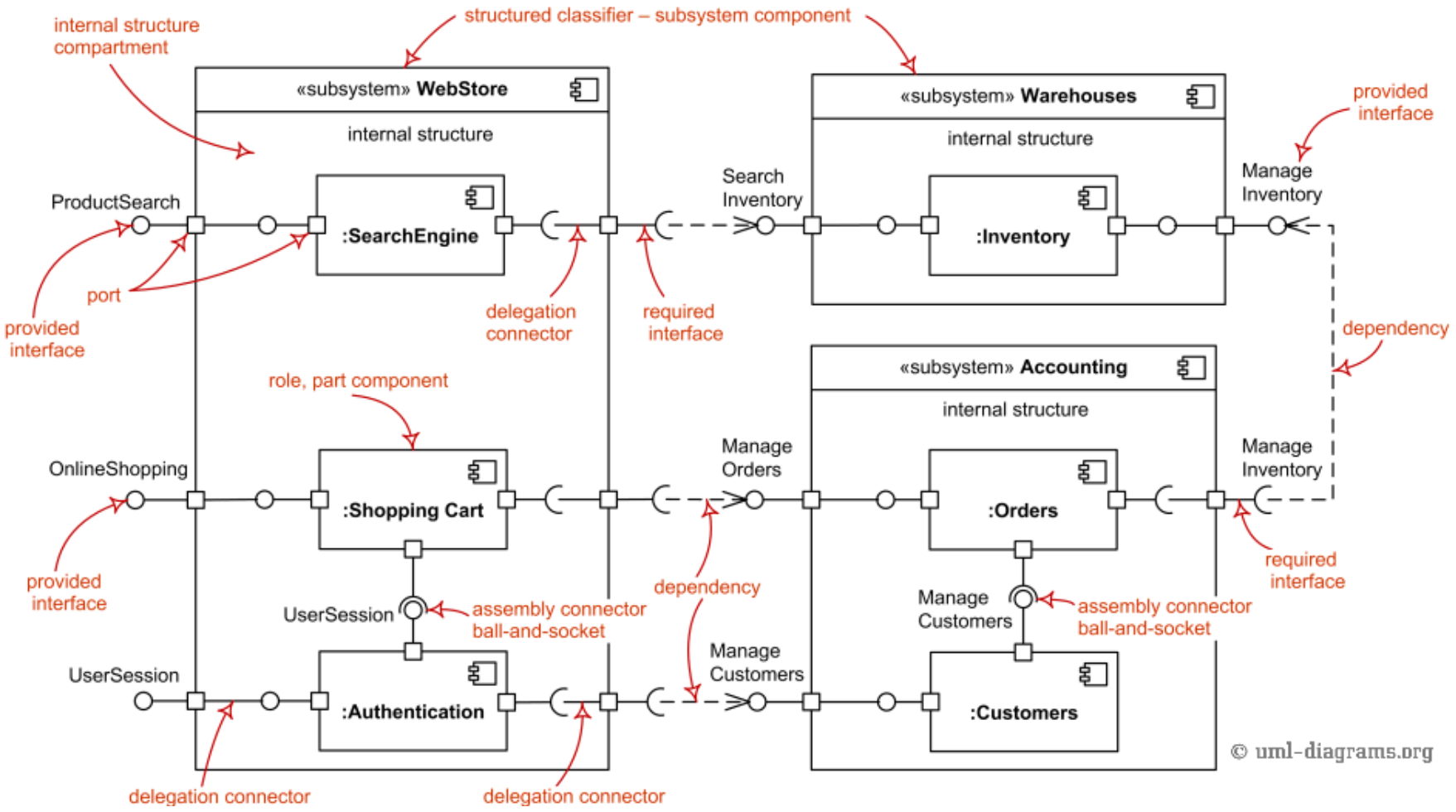
Esempi di connettori di assemblaggio

Componente realizzazione



Esempio di componenti di realizzazione

Esempio di component diagram

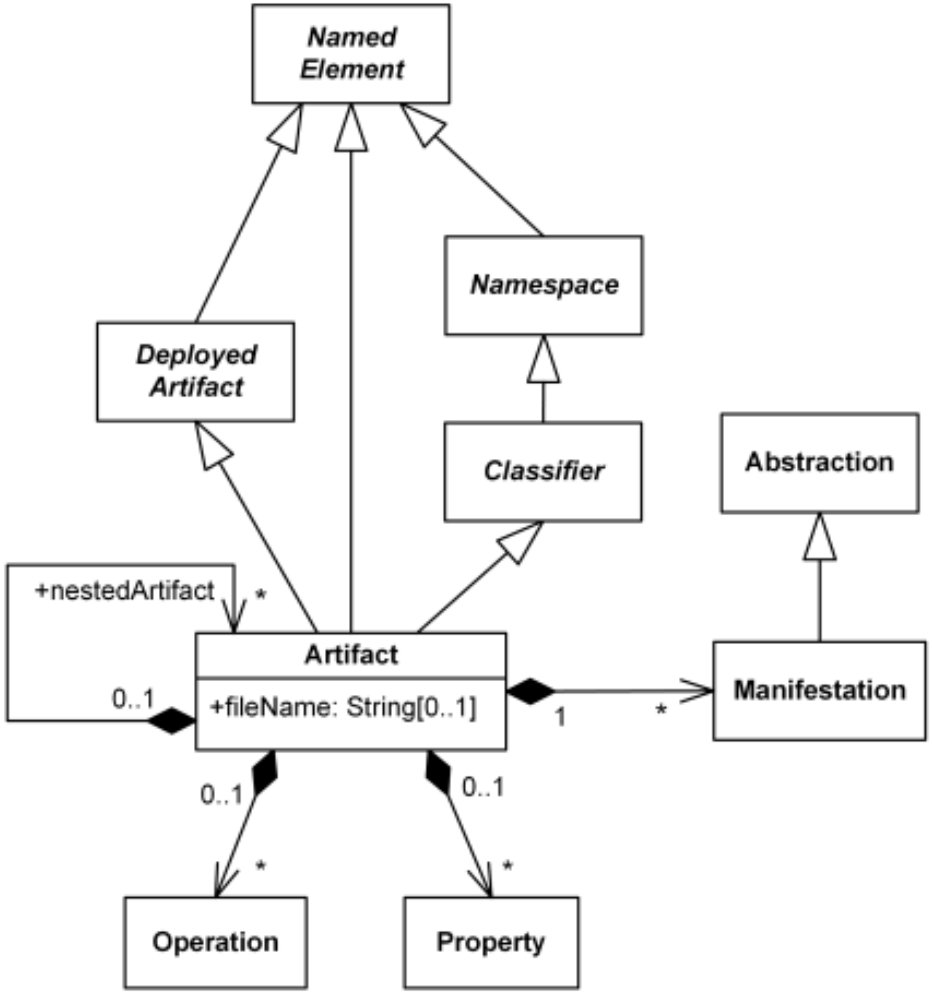


Artifact

web-tools-lib.jar

Esempio di artifact – rappresentazione completa di qualche elemento

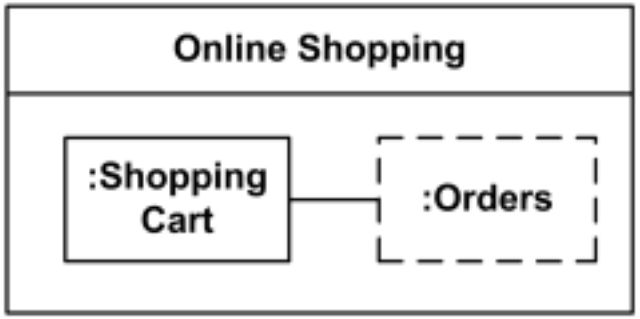
Sintassi di Artifact



Esempio di utilizzo di artifact

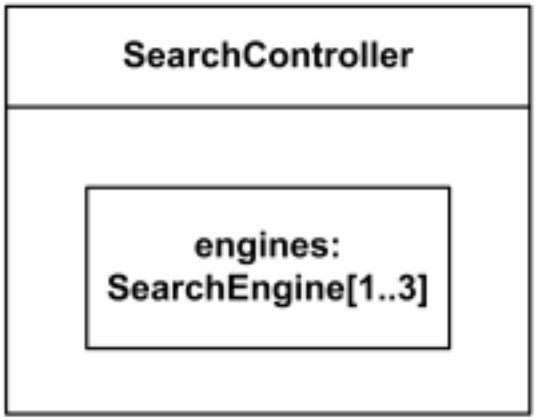
Composite structure diagram

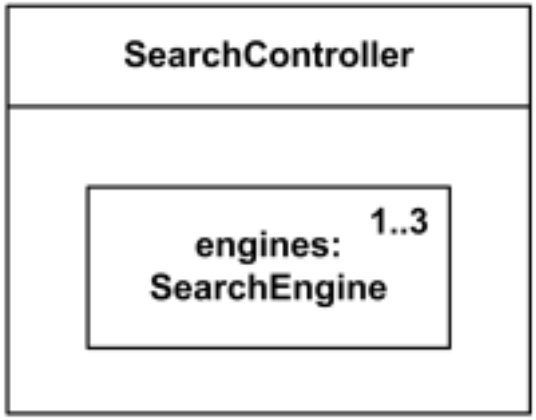
- Composite Structure diagram
  - struttura interna di elementi complessi



Struttura interna di una classe

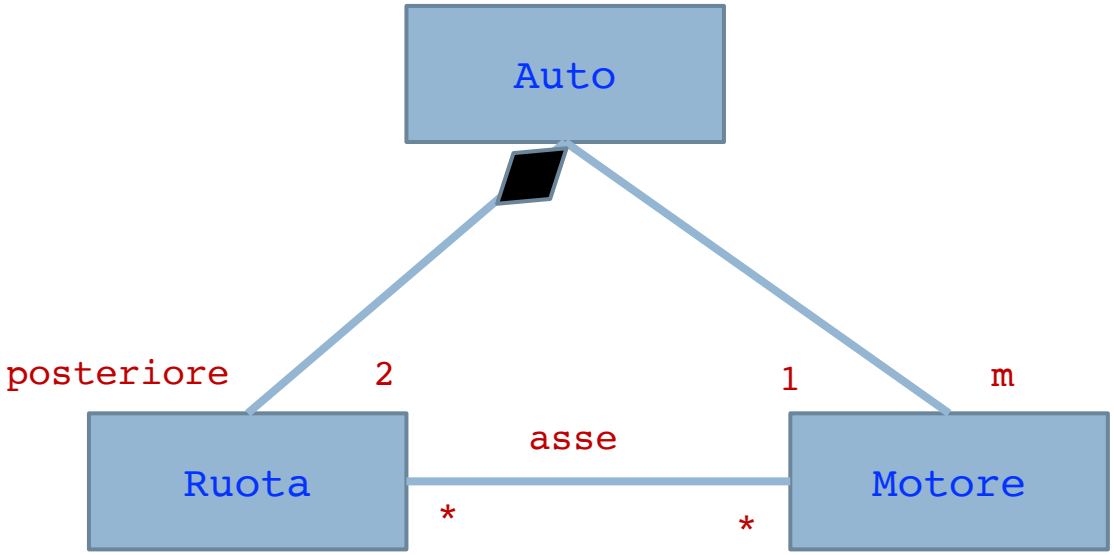
Parti

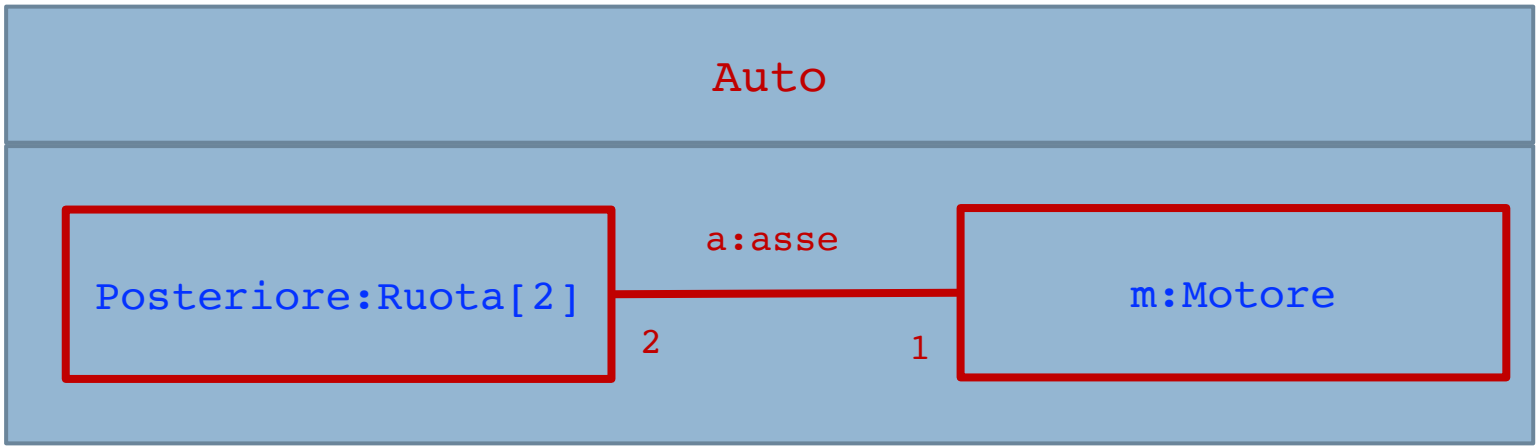




Parti e molteplicità

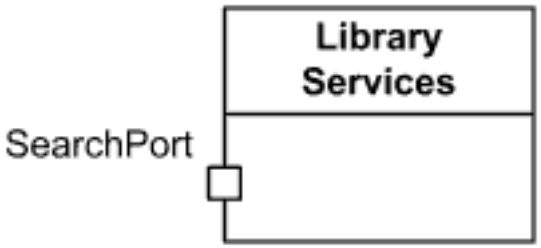
Parti

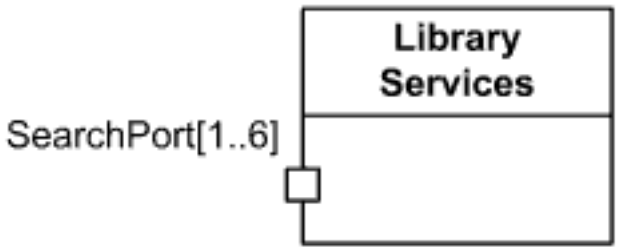


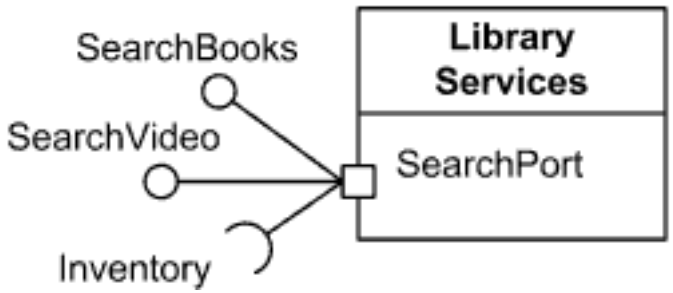


Parti e molteplicità

Porte

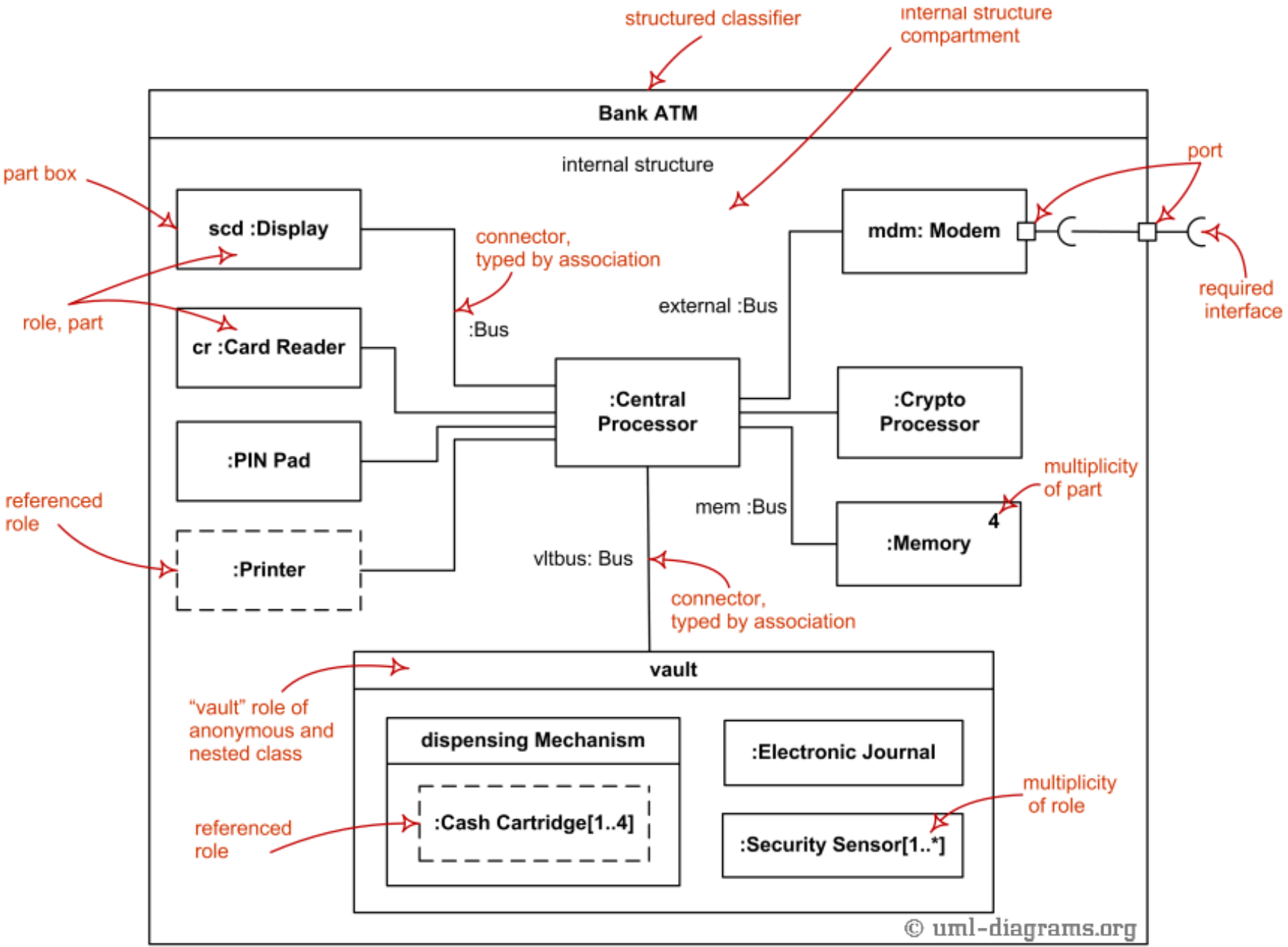




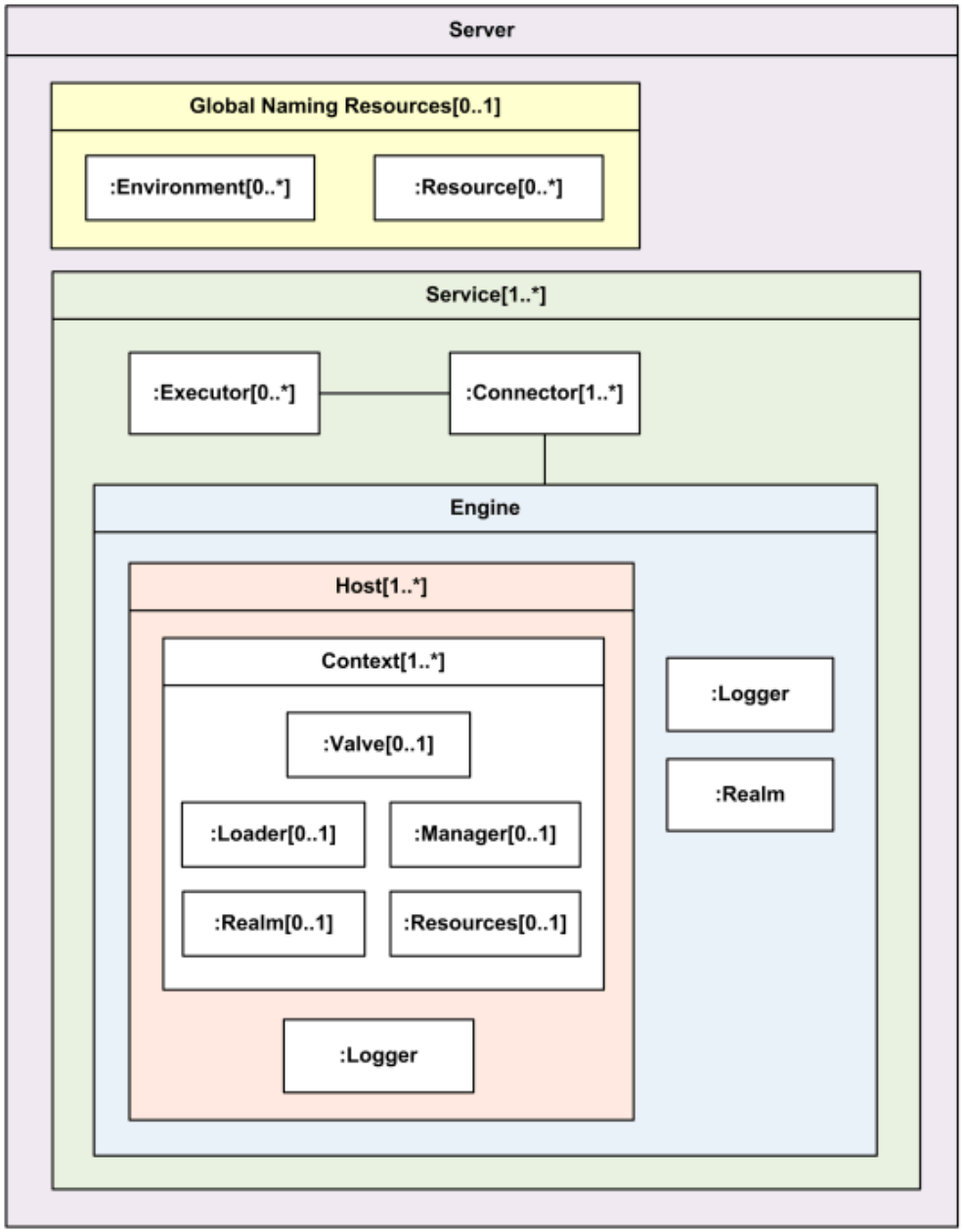


Le porte consentono di provvedere a dei servizi

BankAccount

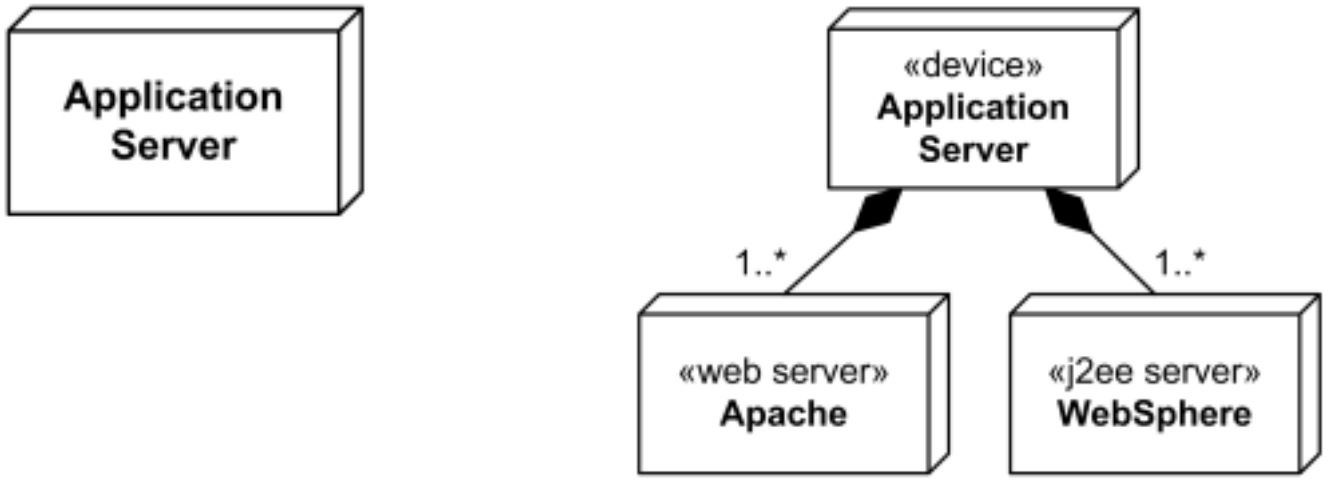


Apache Tomacat 7 Server



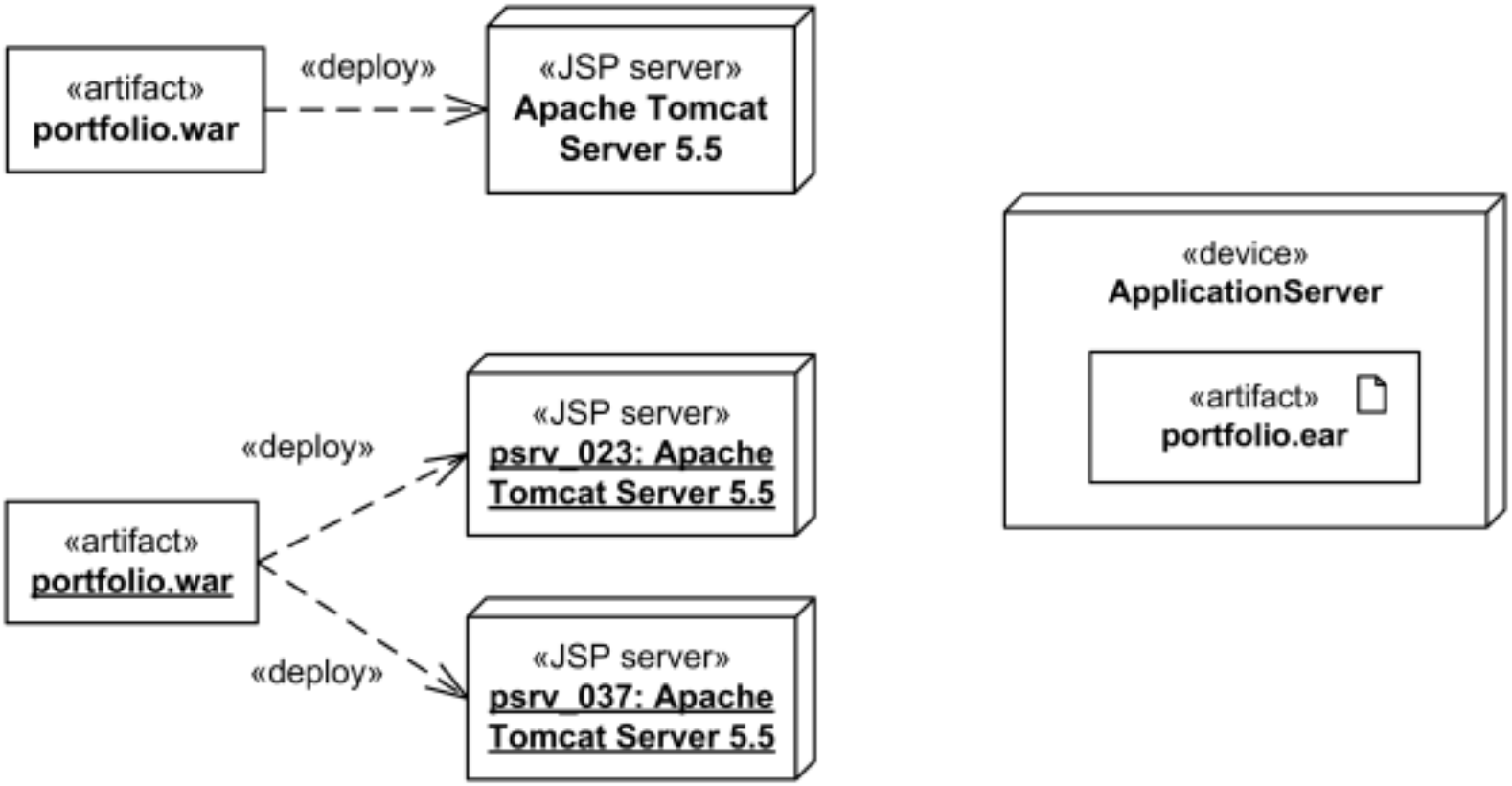
Deployment diagram

- Deployment diagram
  - Organizzazione e strutturazione degli elementi di un sistema informatico dal punto di vista fisico
  - e.g., sistemi informatici distribuiti



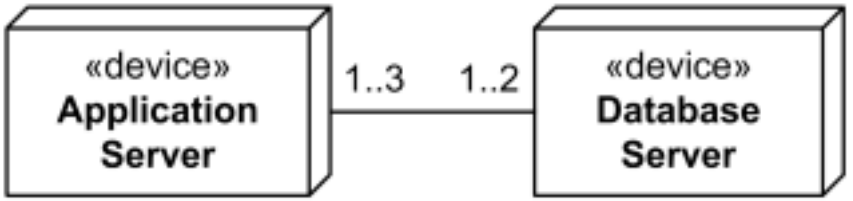
Il componente fondamentale è il nodo

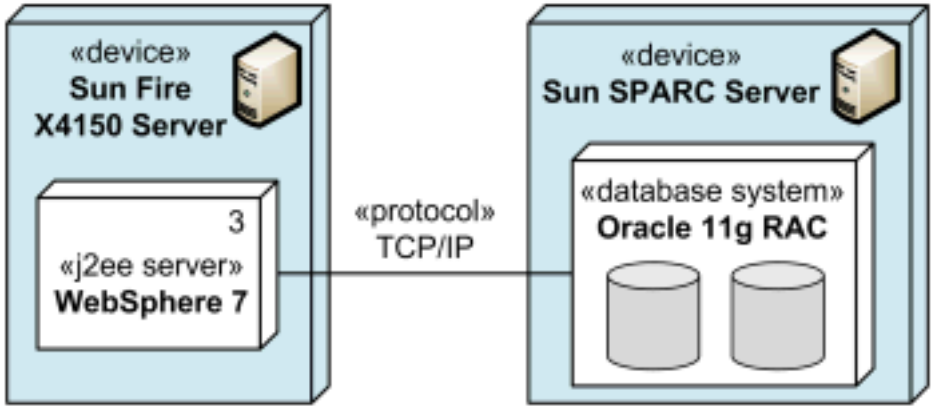
Deployment



Esempi di Deployment di un'Apllicazione Web

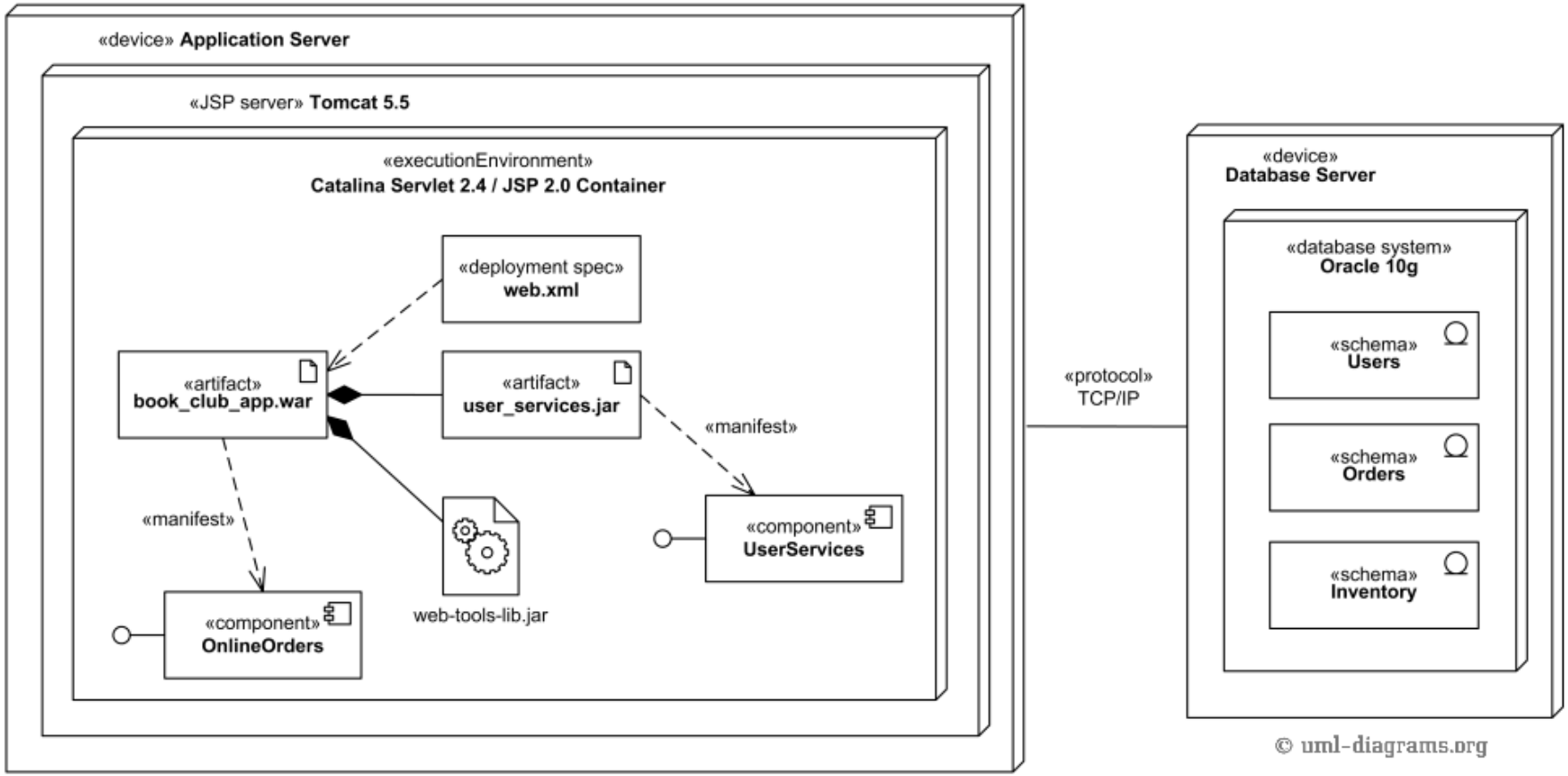
Communication path





Esempi di path di comunicazione tra device

J2EE Web Application



Apple iTunes