

Unified Modeling Language State Diagram

Sandro Morasca

Università degli Studi dell'Insubria

Dipartimento di Scienze Teoriche e Applicate

Via Ottorino Rossi 9 – Padiglione Rossi

21100 Varese, Italy

sandro.morasca@uninsubria.it



- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

- Rappresentano il comportamento di un qualunque sistema (ad esempio un oggetto di una classe) in termini di
 - eventi a cui il sistema è sensibile
 - azioni prodotte
 - transizioni di stato
 - identificazione degli stati interni del sistema
- Possibilità di descrivere evoluzioni parallele
- Sintassi mutuata da StateChart (D. Harel)



Elementi grafici

UML – State Diagram

➤ Introduzione

Diagrammi
semplici
Diagrammi
strutturati
Flussi di
esecuzione



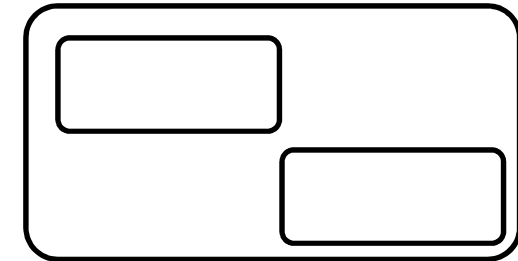
Stato



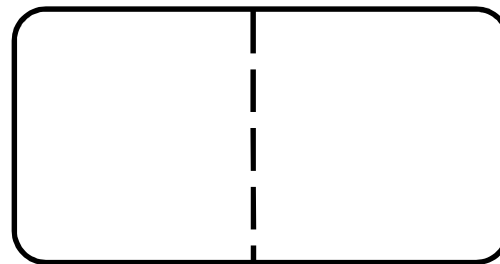
Stato iniziale



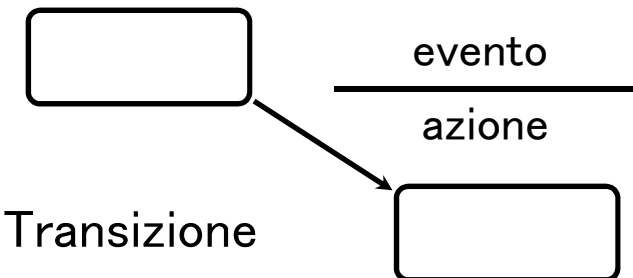
Stato finale



Decomposizione OR

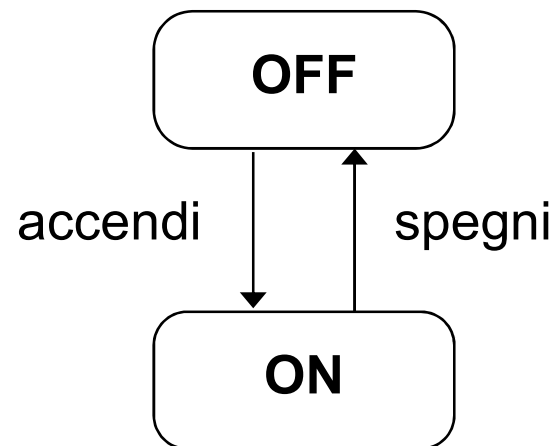


Decomposizione AND





- Uno stato rappresenta una situazione durante la quale una condizione (solitamente implicita) è verificata
- La condizione può rappresentare una situazione statica, come l'attesa di un evento esterno, o semplicemente una situazione stabile che può essere modificata solo da un intervento esterno

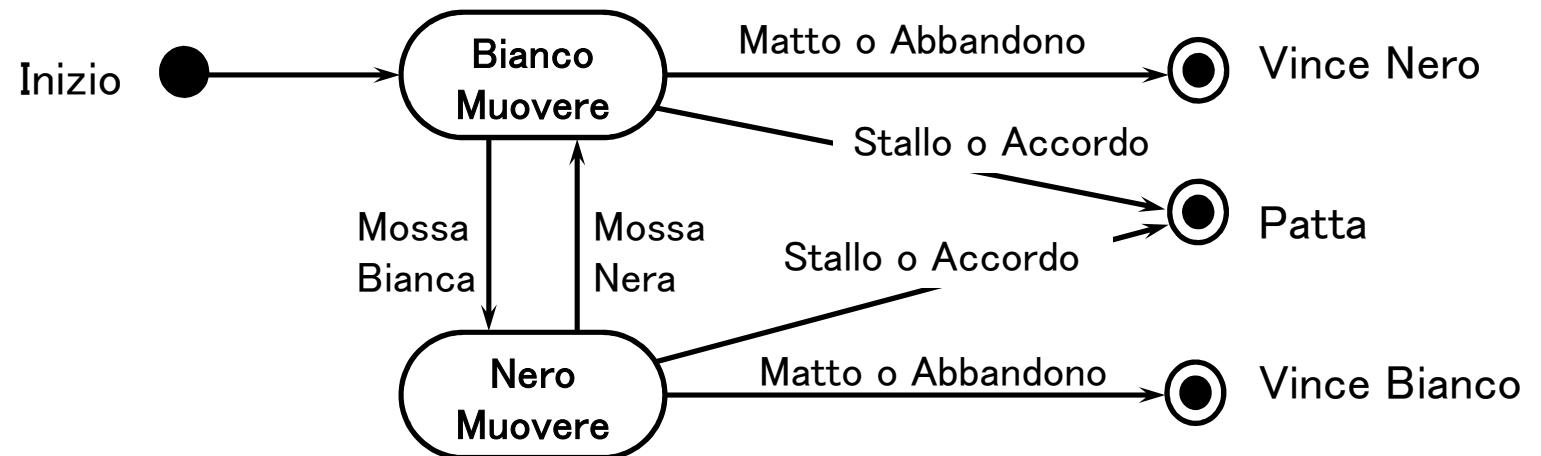




Esempio

UML – State Diagram

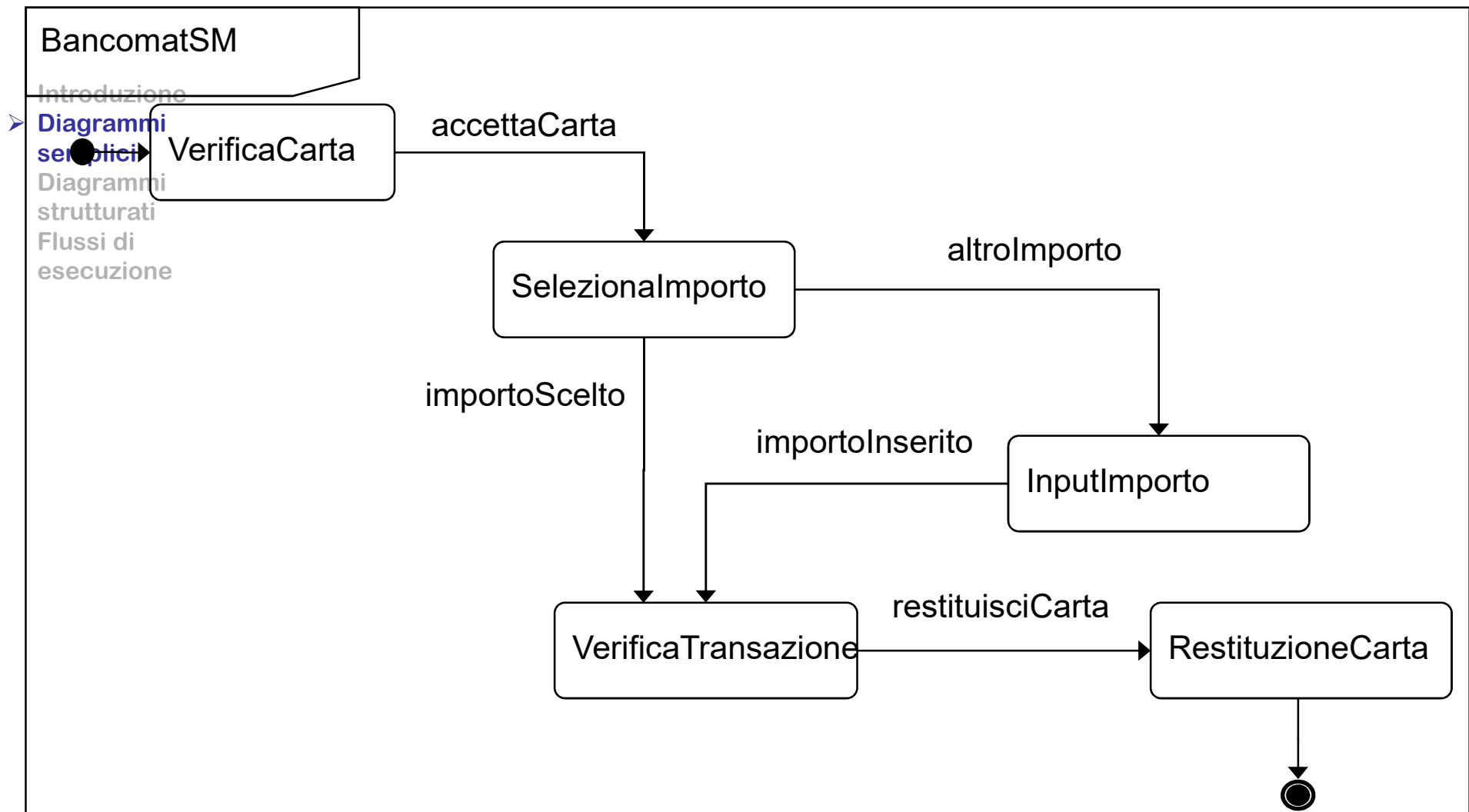
- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione





Esempio

UML – State Diagram





Stato determinato da condizione statica

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

- È l'insieme dei valori degli attributi e dei link posseduti da un oggetto in un certo istante
- Astratto
 - es. motore acceso/spento (non interessa il numero di giri/min.)
 - può corrispondere a diverse - anche infinite - combinazioni di valori degli attributi



Stato determinato da condizione statica

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

- Lo stato influenza il comportamento
 - l'oggetto reagisce in modo qualitativamente diverso agli eventi esterni, in funzione dello stato in cui si trova
 - es. se si può fare o no un'operazione di pop dipende dallo stato della pila (vuota o no)
 - il comportamento quantitativo è influenzato dal valore degli attributi, dei link e dei parametri delle operazioni
 - es. il valore restituito da pop dipende dal valore dell'attributo della pila che rappresenta il contenuto



Durata degli stati

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

- Uno stato perdura nel tempo
 - finché un evento non fa cambiare stato all'oggetto (es. un versamento fa passare un conto corrente da saldo negativo a saldo positivo)



Stati “dinamici”

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

- La condizione verificata in uno stato può anche rappresentare la dinamica dell'oggetto
 - es. l'elemento entra nello stato quando inizia una certa attività e lo lascia appena l'attività viene completata



● Evento = stimolo esterno

- può causare nell'oggetto destinatario un cambio di valore, di stato o la produzione di ulteriori eventi
- si intende che un evento ha una individualità ben definita
 - raggruppabili in classi di eventi (con attributi caratterizzanti)
- la risposta ad uno stimolo dipende dallo stato, e può implicare una transizione di stato
 - esempio: versamento su CC



Identificare gli stati

UML – State Diagram

Introduzione
➤ Diagrammi
semplici
Diagrammi
strutturati
Flussi di
esecuzione

- Per ciascun oggetto occorre identificare tutti gli stati in cui può trovarsi
- Suggerimenti
 - trascurare gli attributi ininfluenti
 - alcuni attributi non modificano in modo qualitativo il comportamento di un oggetto, ma al più i valori degli eventi che l'oggetto produce
 - individuare le condizioni limite
 - trovare tutti i confini e i limiti dello stato
 - definire un corretto livello di astrazione
 - per un sistema di prenotazione la partenza di un aereo è un evento, ma per il software di controllo dell'aereo sono centinaia di eventi distinti



Lo scenario

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

- Uno scenario è una sequenza di eventi e azioni che descrive una specifica esecuzione del sistema
- Dà un'idea dell'ordine degli eventi
 - talune coppie indicano un ordine casuale

Tracciato degli eventi

Il chiamante alza la cornetta
Inizia il tono pronto
Il chiamante compone uno
Smette il tono pronto
Il chiamante compone quattro
Il chiamante compone quattro
Il chiamante compone uno
Il chiamante compone uno
...
Il telefono chiamato inizia a squillare
Inizia il tono di libero
Il telefono chiamato viene sganciato
Il telefono chiamato smette di squillare
Smette il tono di libero
...



Ricavare il diagramma degli stati

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

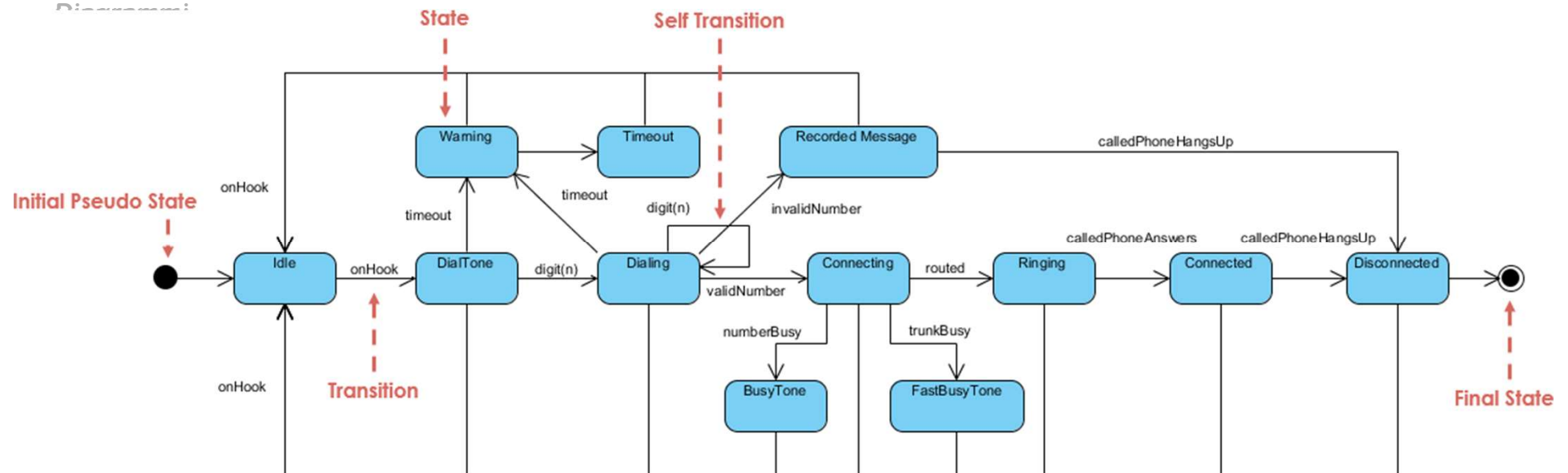
- Una collezione sufficientemente ricca di sequence diagram descrive tutte le possibili evoluzioni dell'oggetto considerato
 - il fatto di poterle descrivere tutte deriva dalla capacità di astrazione
- Ad esempio, in uno scenario il chiamante riaggancia per primo, in un altro scenario riaggancia il ricevente: occorre rappresentare la reazione dell'oggetto ad entrambi questi eventi



Esempio

UML – State Diagram

- Introduzione
- Diagrammi semplici



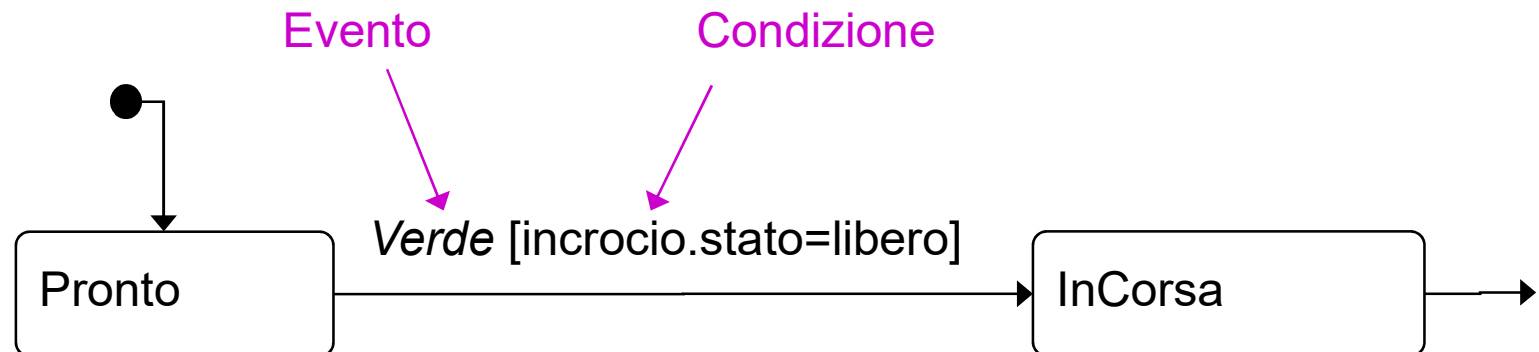


Condizioni

UML – State Diagram

Introduzione
➤ Diagrammi
semplici
Diagrammi
strutturati
Flussi di
esecuzione

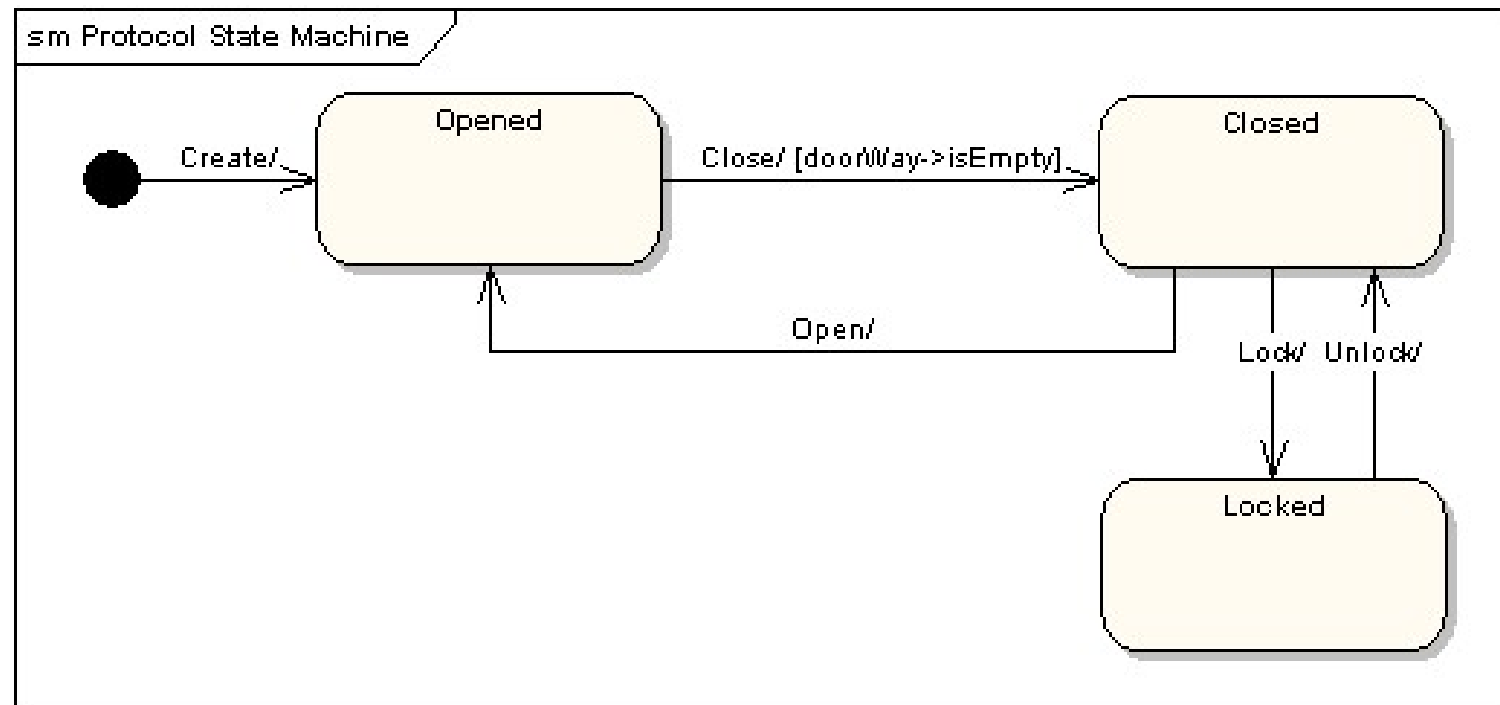
- Sono funzioni booleane sui valori degli oggetti
- Sono valide in un intervallo di tempo
- Sono utili come guardie delle transizioni di stato (non basta l'evento, deve essere verificata la condizione)





- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

● Comportamento di una porta



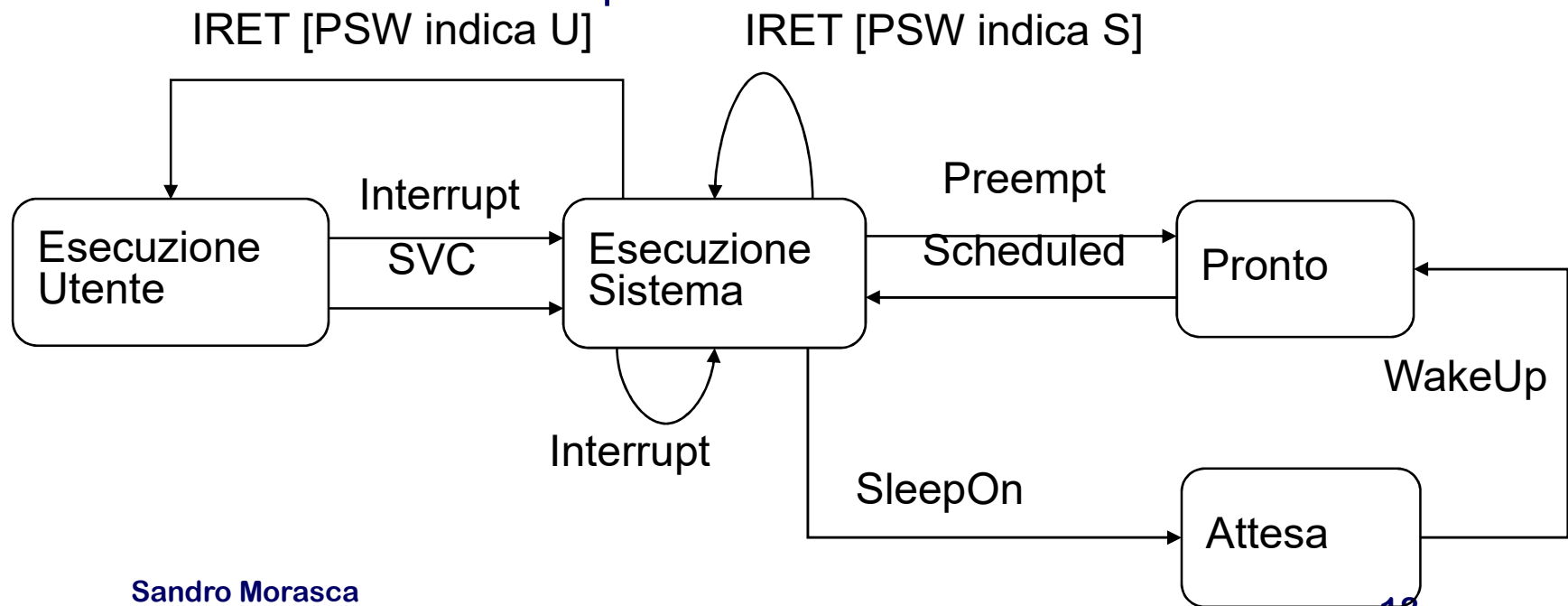


Condizioni: esempio

UML – State Diagram

Introduzione
➤ Diagrammi
semplici
Diagrammi
strutturati
Flussi di
esecuzione

- Lo statechart descrive gli stati in cui si può trovare un processo in un elaboratore multi tasking
 - Il risultato di una IRET (ritorno da interrupt) dipende dall'indicazione del modo di esecuzione (Utente o Supervisore) contenuta nella Program Status Word scaricata dalla pila





Operazioni

UML – State Diagram

Introduzione
➤ Diagrammi
semplici
Diagrammi
strutturati
Flussi di
esecuzione

- Durante la loro vita gli oggetti eseguono operazioni
 - associate allo stato (attività)
 - associate alle transizioni (azioni)
- Le attività hanno una durata
 - continue
 - sequenziali
- Le azioni sono istantanee



● Azioni

- sono operazioni che hanno durata istantanea (rispetto alla granularità del tempo): tipicamente produzione di eventi
- sono associate alle transizioni di stato oppure all'ingresso o all'uscita da uno stato

● Attività

- sono operazioni con durata significativa
- sono associate ad uno stato
 - continue o sequenziali

● Transizioni automatiche

- se uno stato ha una attività associata e una freccia senza eventi esce da questo, la freccia indica la transizione svolta automaticamente al completamento della attività



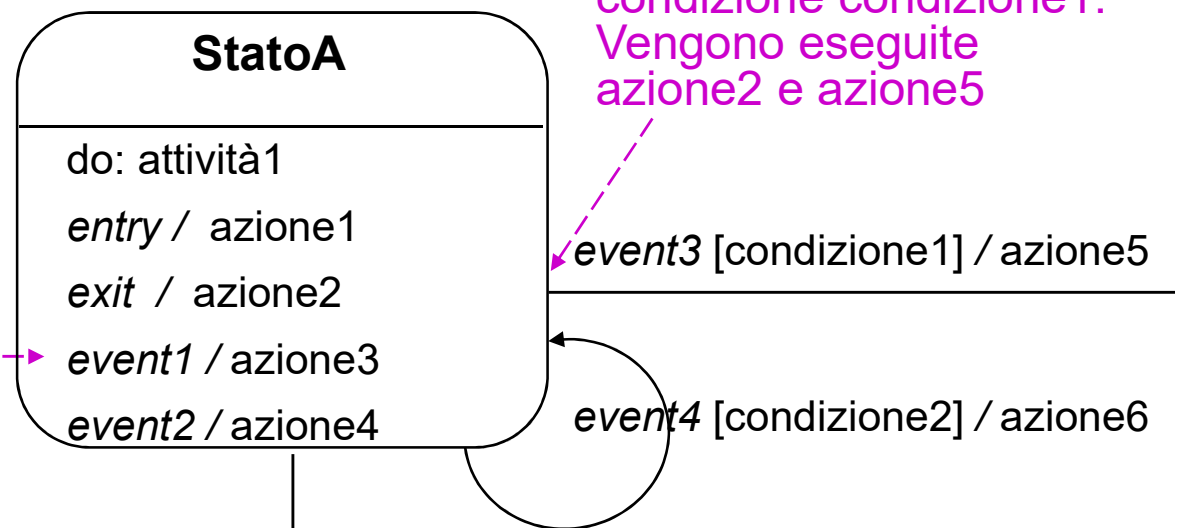
Notazione generale

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

In caso di ricezione dell'evento1 nello stato A viene eseguito azione3 senza cambio di stato (e quindi senza azioni di entry ed exit)

Transizione automatica (evento scatenante fine della attività1)



Transizione causata dall'evento evento3 sotto condizione condizione1. Vengono eseguite azione2 e azione5

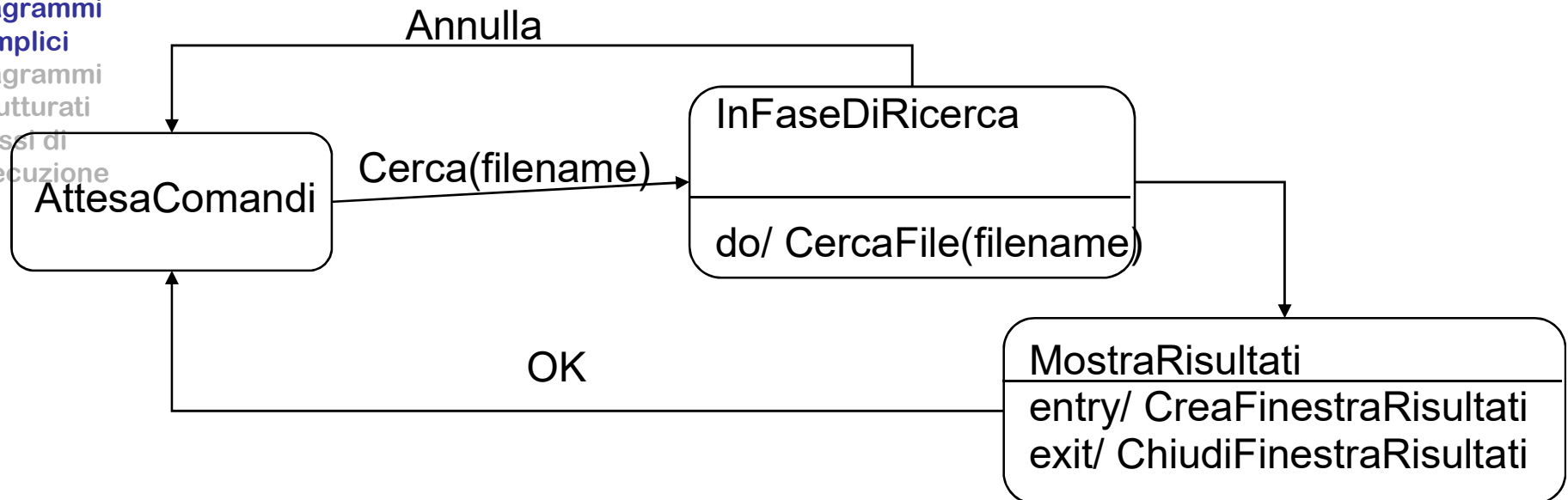
Pseudo transizione causata dall'evento evento4; nell'ordine vengono eseguiti: azione2, azione6, azione1



Esempio

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione



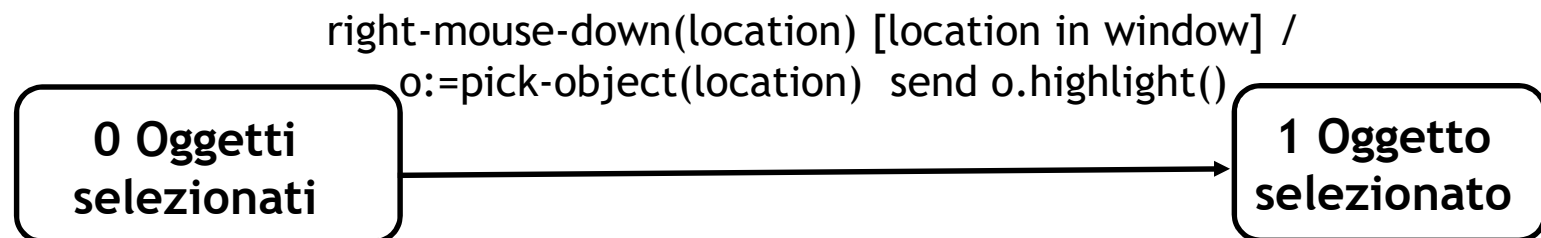
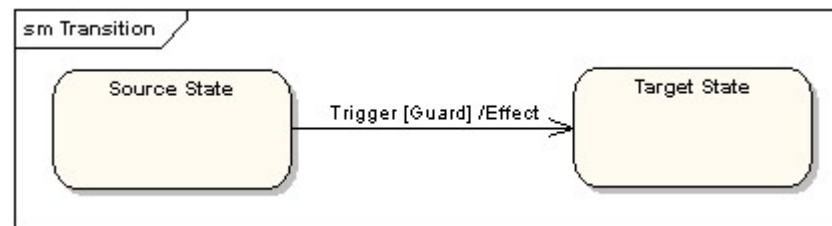


Eventi (generati da azioni)

UML – State Diagram

Introduzione
➤ Diagrammi
semplici
Diagrammi
strutturati
Flussi di
esecuzione

- Spesso le azioni consistono nell'inviare un evento ad un altro oggetto
 - evento(arg) [cond] / azione send t.azione(arg)





Invio di eventi

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

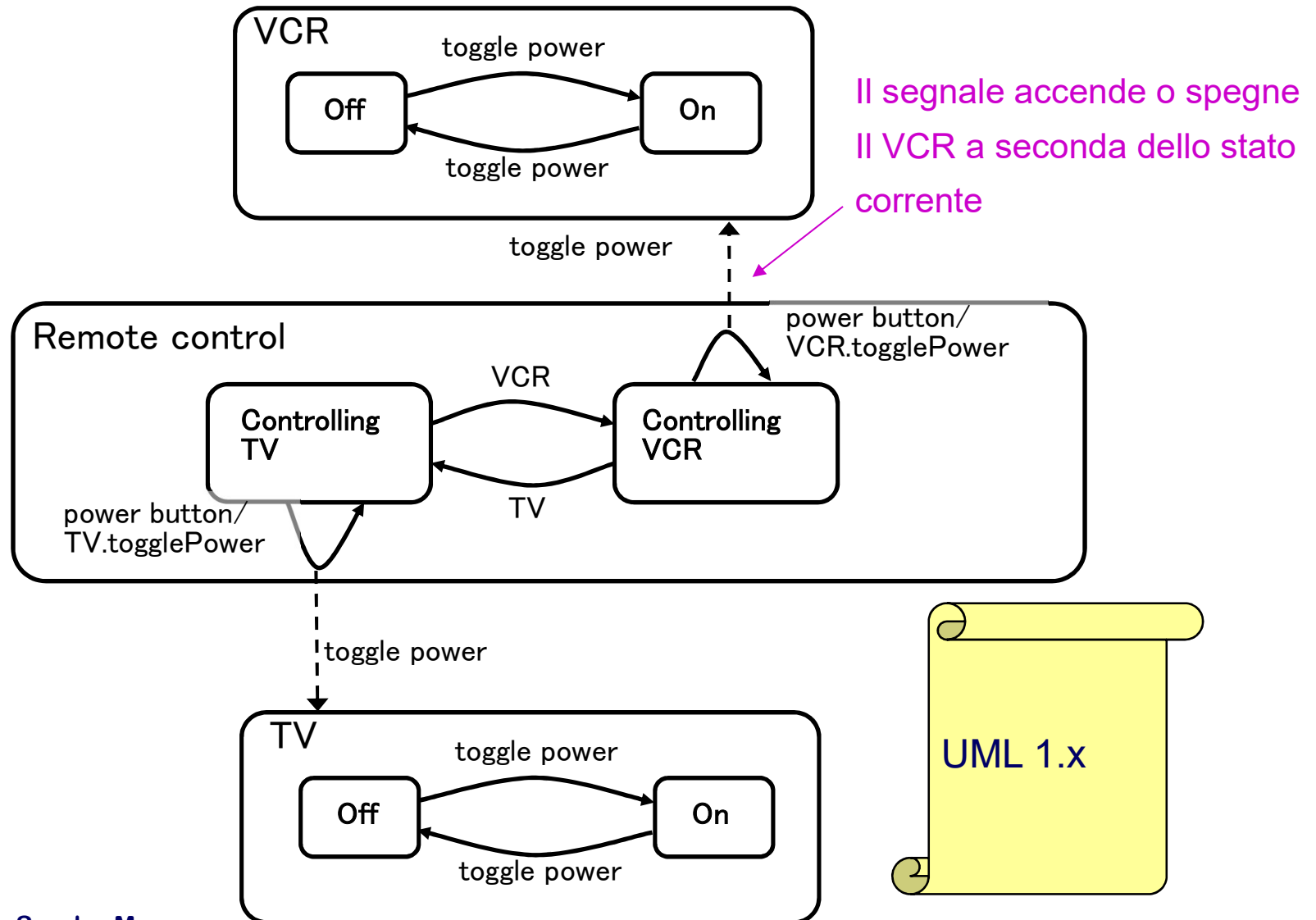
- Gli eventi possono avere attributi
- Il destinatario può essere unico o un intero set di oggetti



Invio di eventi

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione





Problemi dei diagrammi a stati piatti

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Diventano poco espressivi e troppo ingarbugliati al crescere delle dimensioni del problema
 - es. il numero di collegamenti possibili tra stati è quadratico nel numero di stati
- Soluzione: diagrammi strutturati
 - la strutturazione favorisce la descrizione sintetica di sistemi complessi
 - l'attività corrispondente ad uno stato può essere espansa in un diagramma a stati di più basso livello, dove ogni stato rappresenta una fase dell'attività.
 - generalizzazione (specializzazione delle attività, gerarchie di ereditarietà, ...)
 - aggregazione (stati concorrenti)



Tipi di stati

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Semplici
 - non hanno sottostati (cioè non ha regioni né sottomacchine)
- Compositi
- Sotto-macchine



Stati composti

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

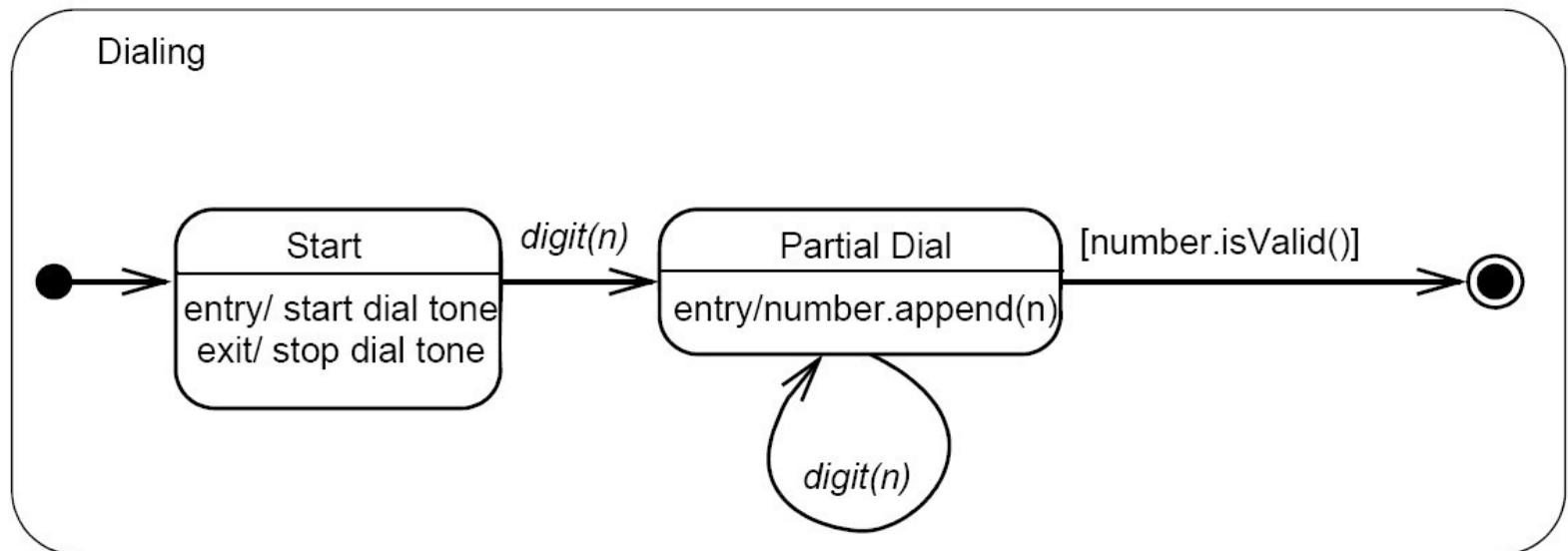
- Uno stato composto può essere
 - uno stato composto semplice, con una sola regione
 - uno stato ortogonale, con più regioni
 - ogni regione ha un insieme di stati e transizioni



Stato composito

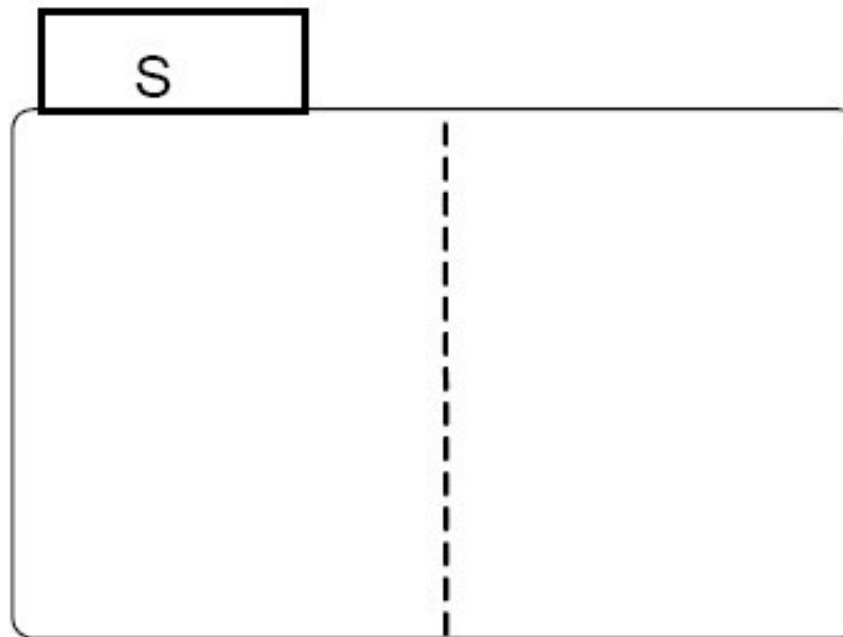
UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione





- Una regione è una parte ortogonale di uno stato composito o di una state machine
- Contiene stati e transizioni





Sottostati concorrenti

UML – State Diagram

FrequenzaCorso:
Stato composito con
una singola regione

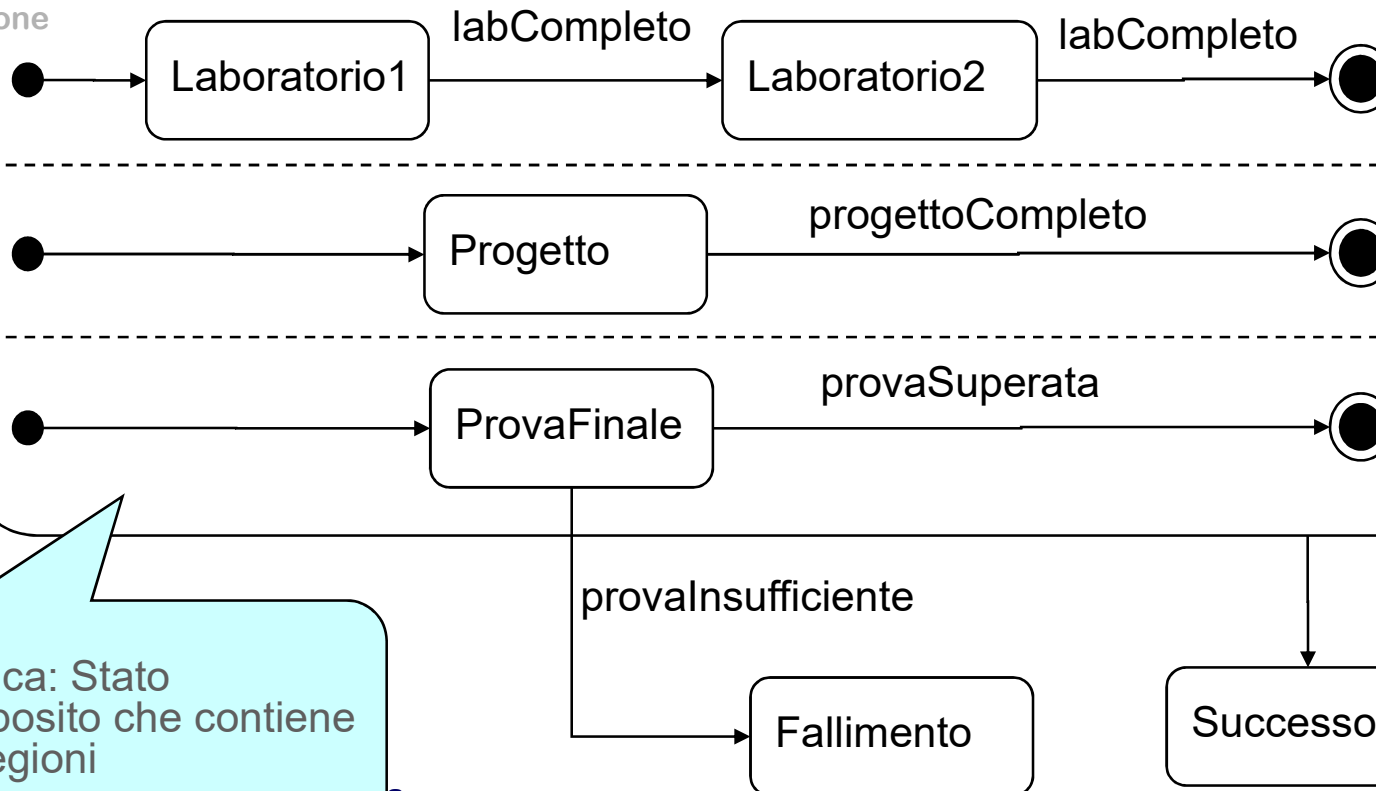
FrequenzaCorso

Introduzione
Diagrammi
semplici

➤ Diagrammi
strutturati

Flussi di
esecuzione

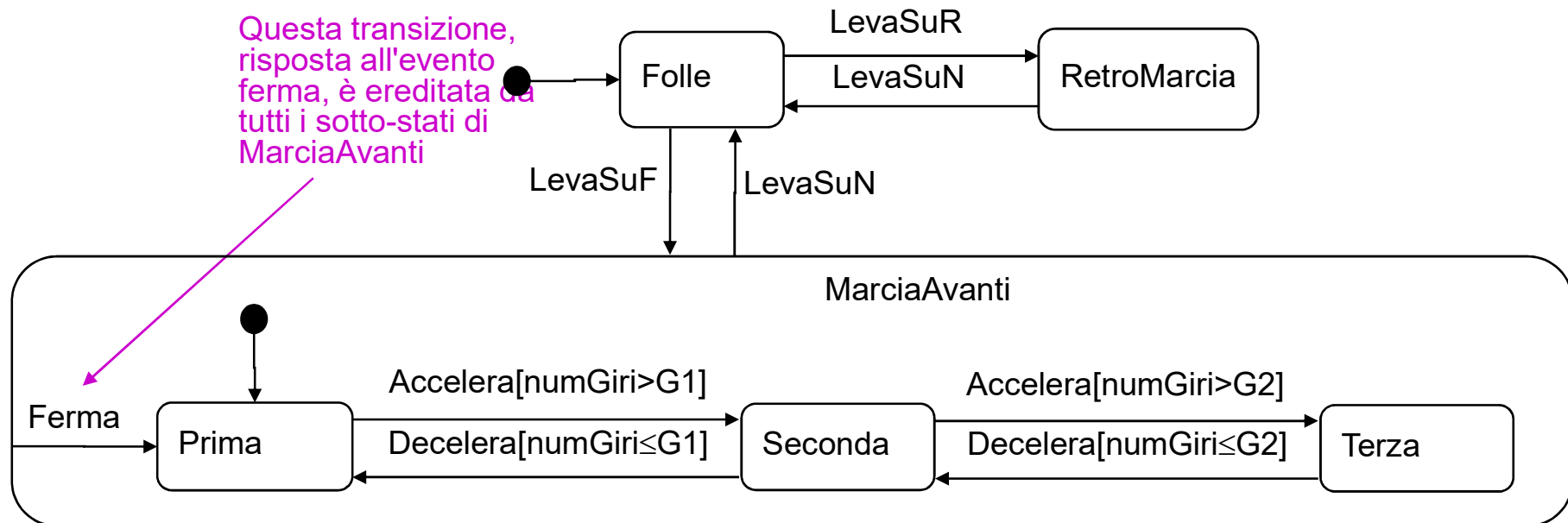
Verifica



Verifica: Stato
composito che contiene
tre regioni



- Uno stato strutturato equivale ad una scomposizione OR degli stati: l'oggetto si trova, all'interno di uno stato più generale, in un qualunque sotto-stato
- I sottostati ereditano le transizioni dei loro superstati (a meno di overriding)

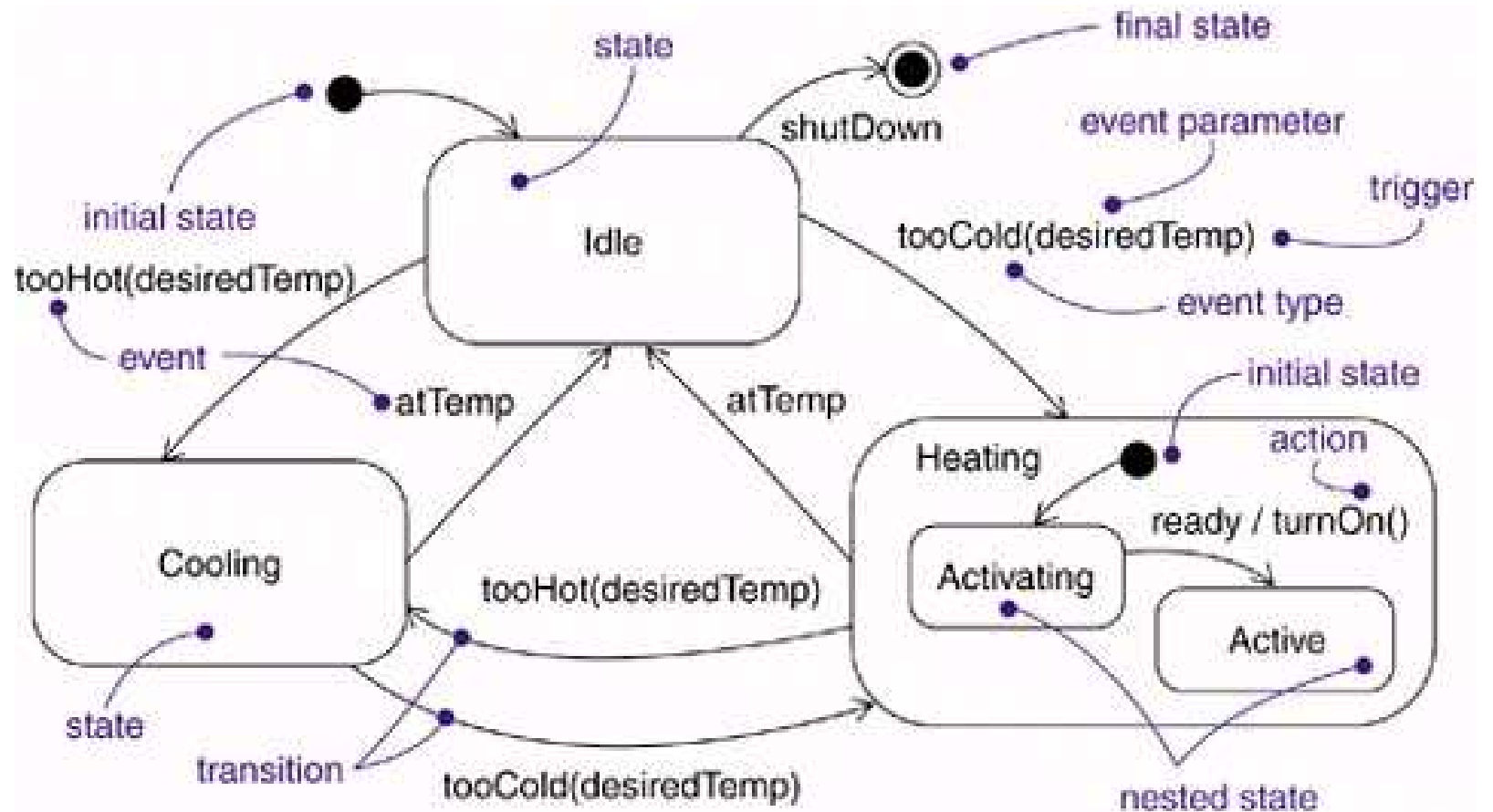




Esempio

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

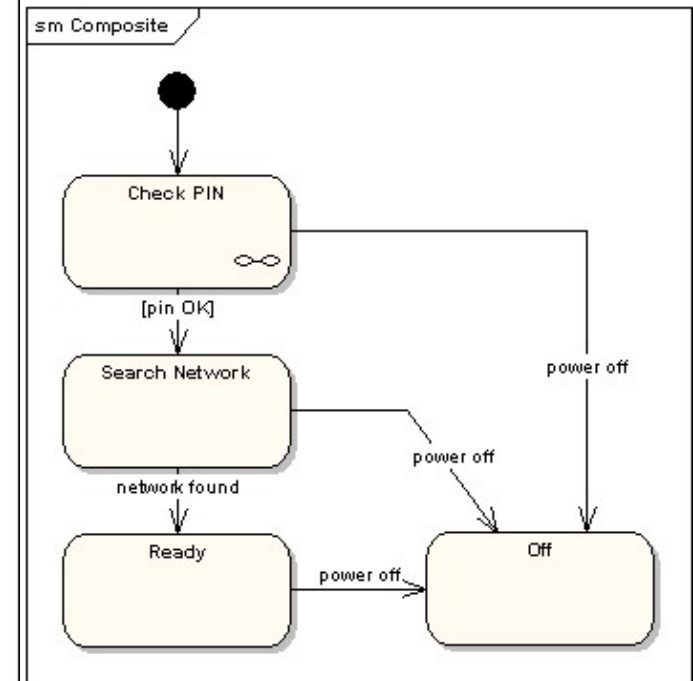
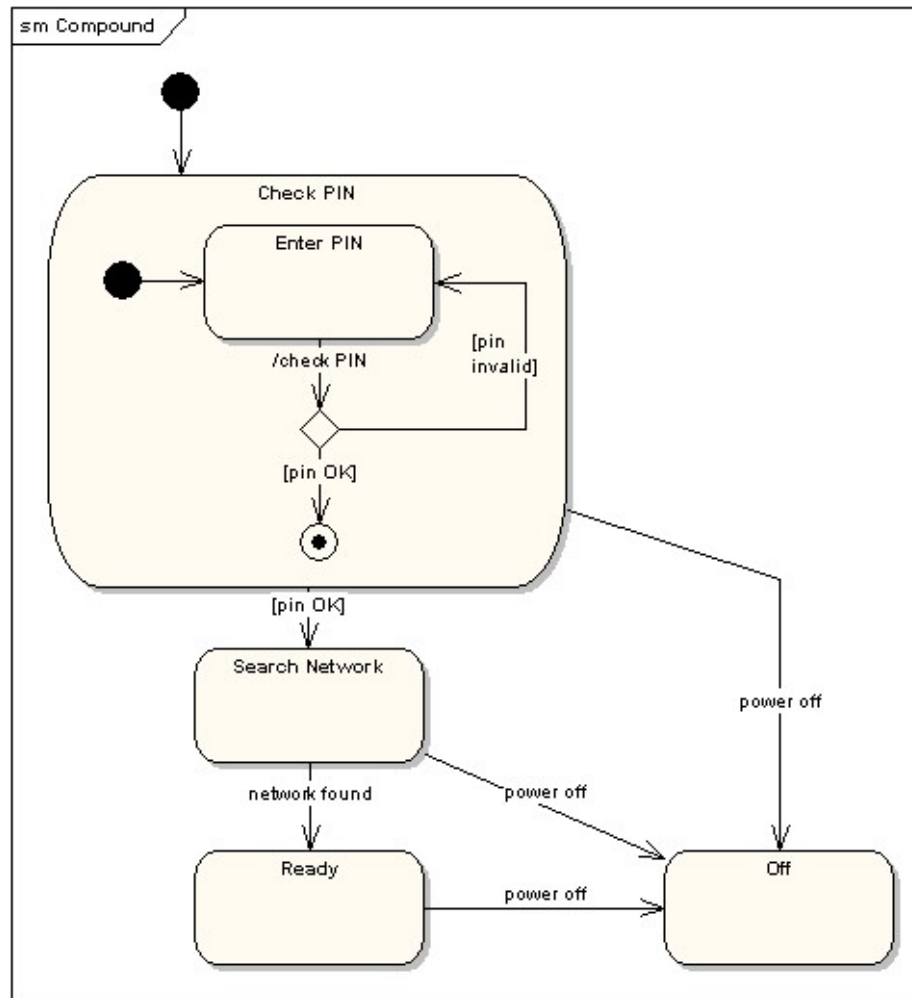




Esempio: due automi equivalenti

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ **Diagrammi
strutturati**
Flussi di
esecuzione



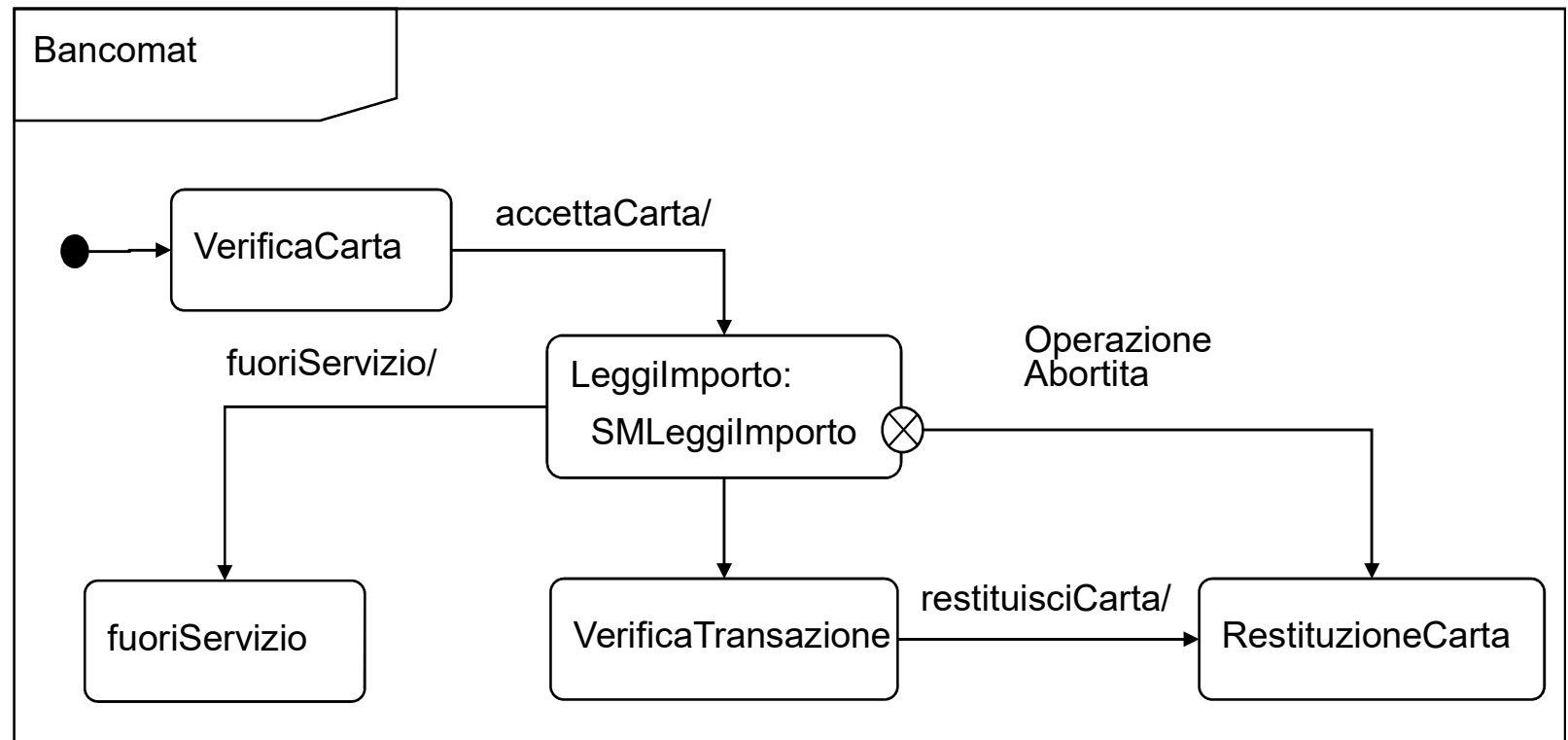


Esempio

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Una macchina a stati che utilizza la sotto-macchina LeggilImporto



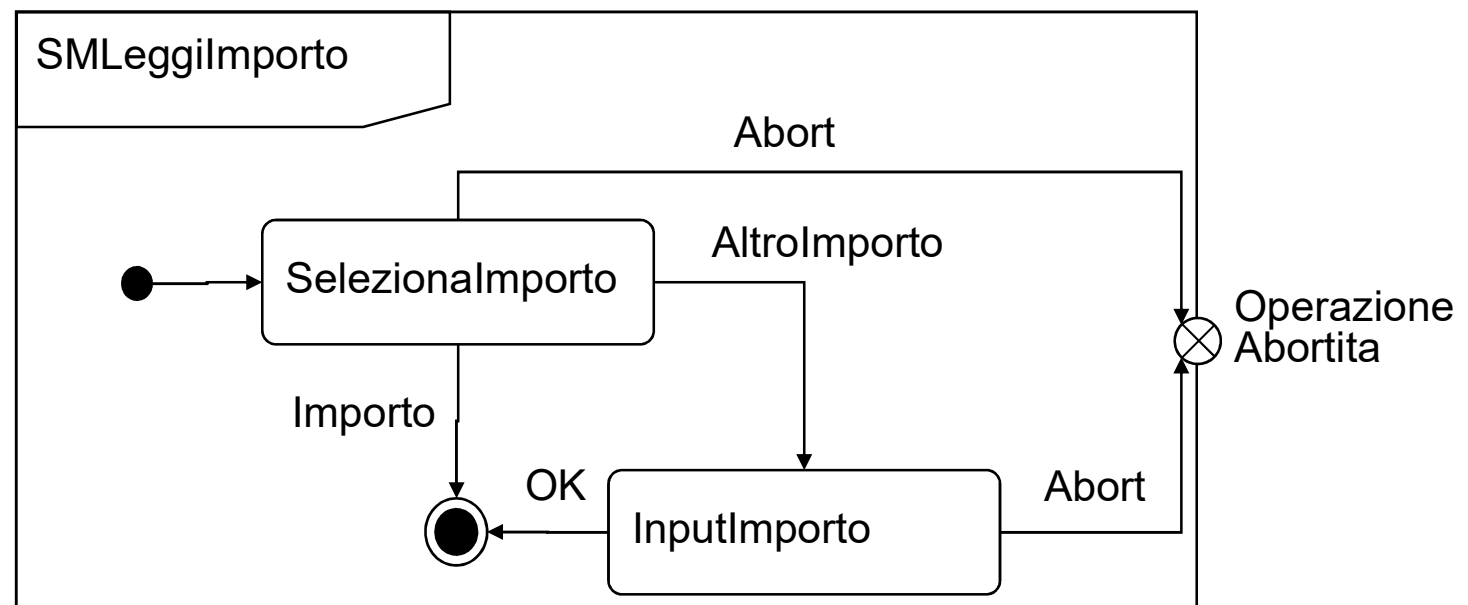


Esempio

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

● Sottomacchina con punto d'uscita



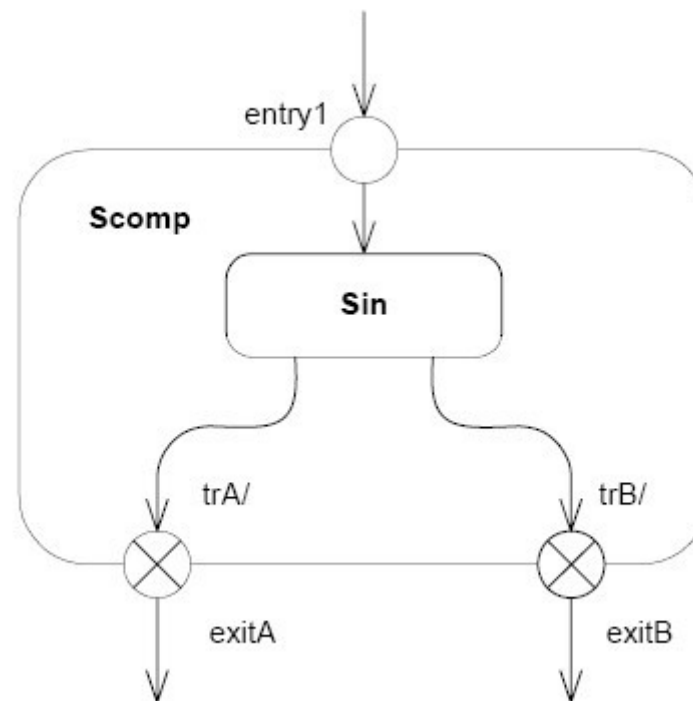


Punti d'entrata e uscita

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

● Punti d'entrata e uscita per uno stato complesso



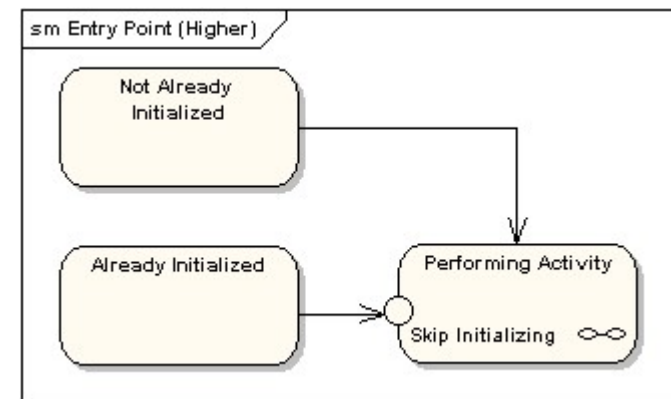
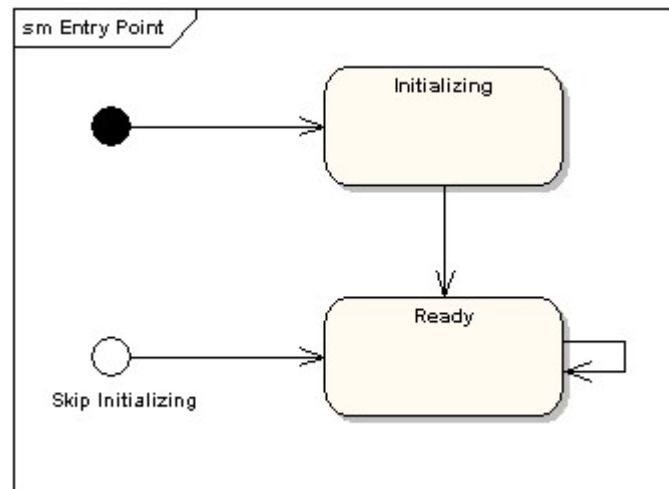


Entry point

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Punti di ingresso in un automa, diversi dallo stato iniziale



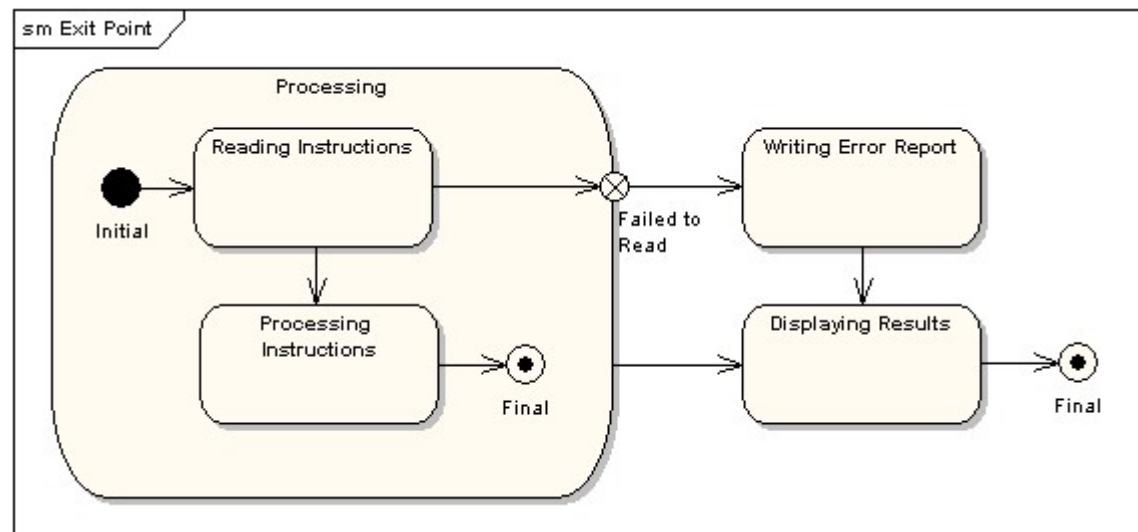


Exit point

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Si possono avere più punti di uscita



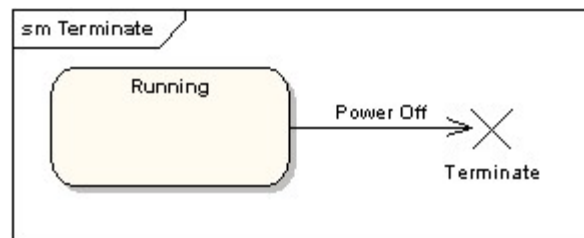


Terminazione

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Mostra la fine della lifeline della macchina a stati

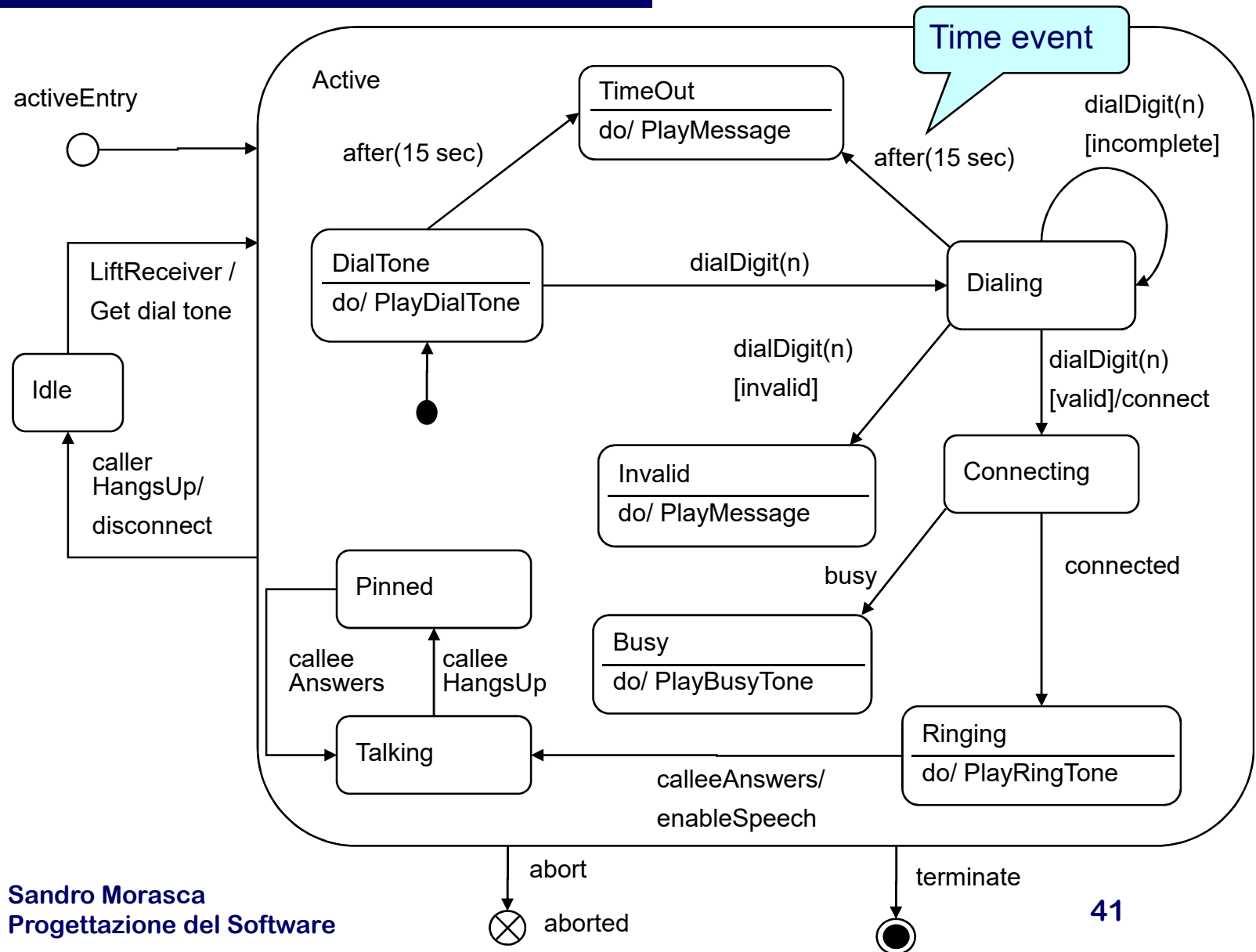




Esempio di macchina a stati

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione





Sottomacchine a stati

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Talvolta una “sotto-macchina a stati” è troppo complessa per essere rappresentata direttamente nella macchina di cui fa parte
 - se ne può dare una rappresentazione semplificata, che dovrà poi essere specificata adeguatamente separatamente



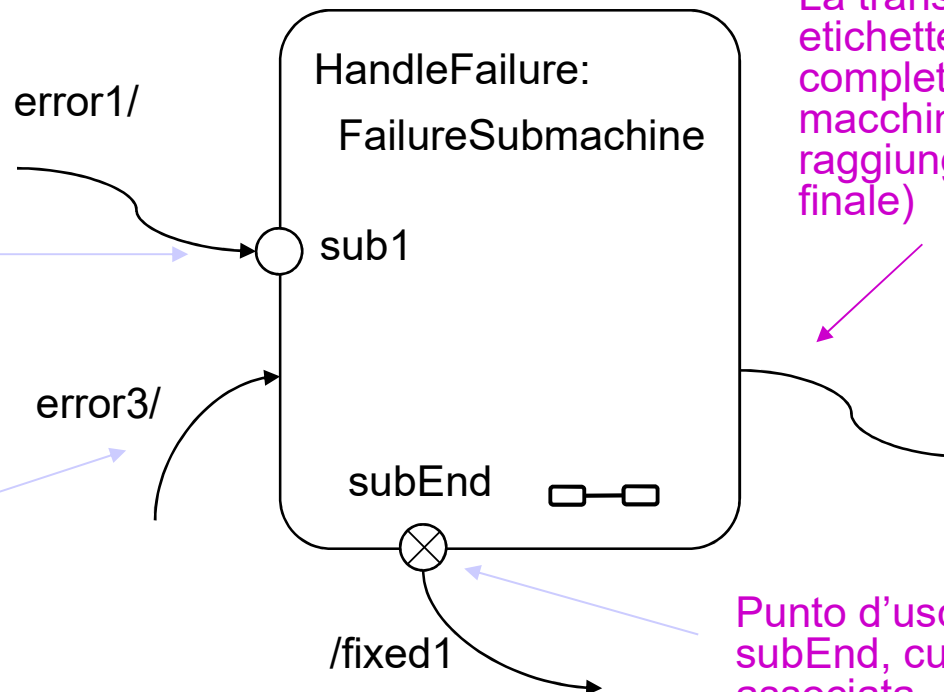
Sottomacchine a stati

UML – State Diagram

- Introduzione
- Diagrammi semplici
- **Diagrammi strutturati**
- Flussi di esecuzione

Punto d'ingresso sub1: indica l'entrata nella sotto-macchina come conseguenza a dell'evento error1.

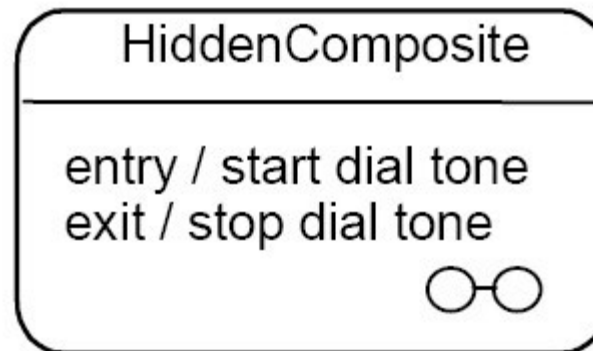
La transizione causata dall'evento error3 porterà al sotto-stato iniziale della sotto-macchina



Punto d'uscita subEnd, cui è associata l'azione fixed1.



- La scomposizione interna può essere nascosta





Aggregazione e concorrenza

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Un modello descrive un insieme di oggetti concorrenti, ciascuno col suo stato e col suo diagramma a stati
- Lo stato dell'intero sistema è dato dal complesso degli stati degli oggetti componenti (il cui numero può anche cambiare dinamicamente)
- Il diagrammi a stati di un oggetto complesso è l'aggregazione dei diagrammi di ciascun componente
- L'aggregazione è la "and-relationship": lo stato complessivo (aggregato) è dato dall'unione degli stati dei vari diagrammi

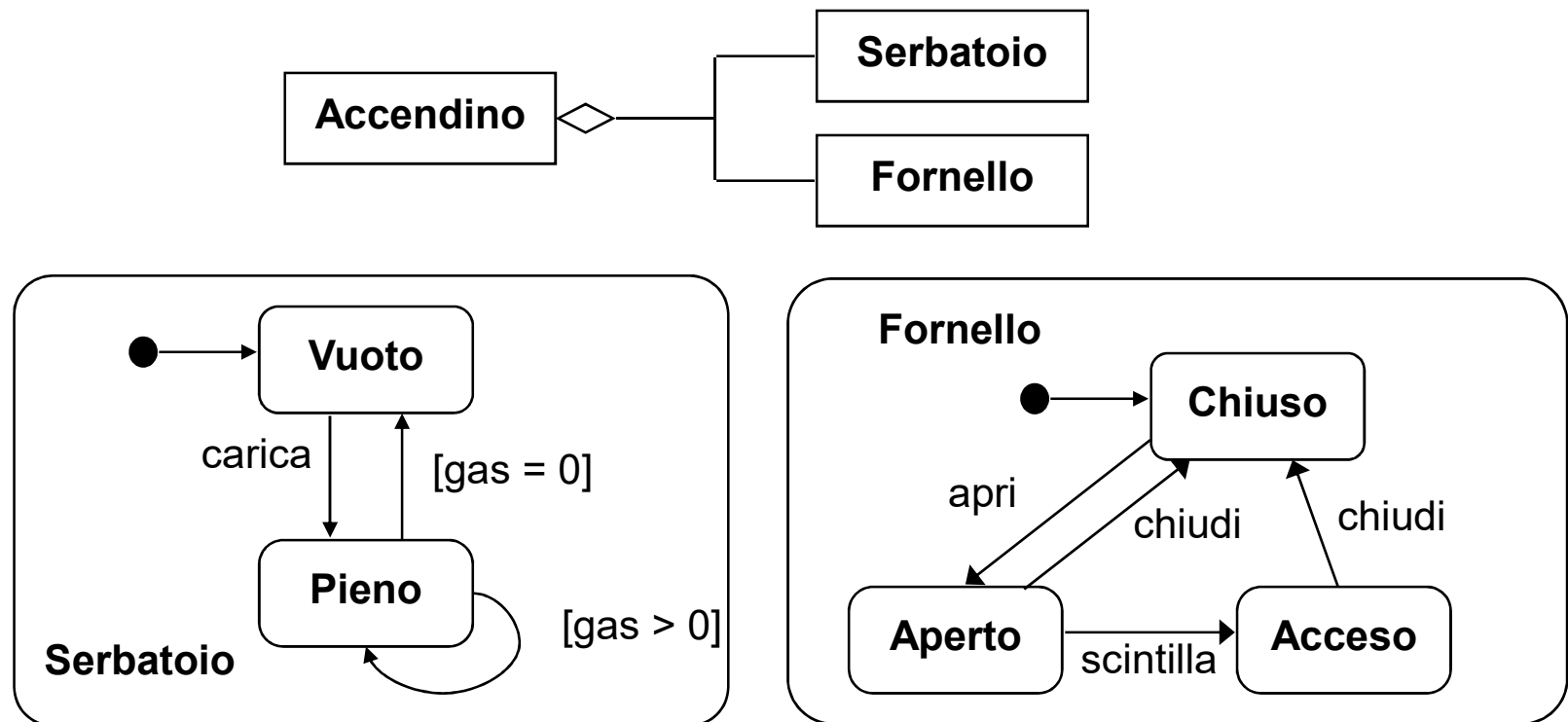


Concorrenza in oggetti compositi

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- Oggetti diversi hanno dinamica intrinsecamente concorrente
 - si ha concorrenza anche in un oggetto composto





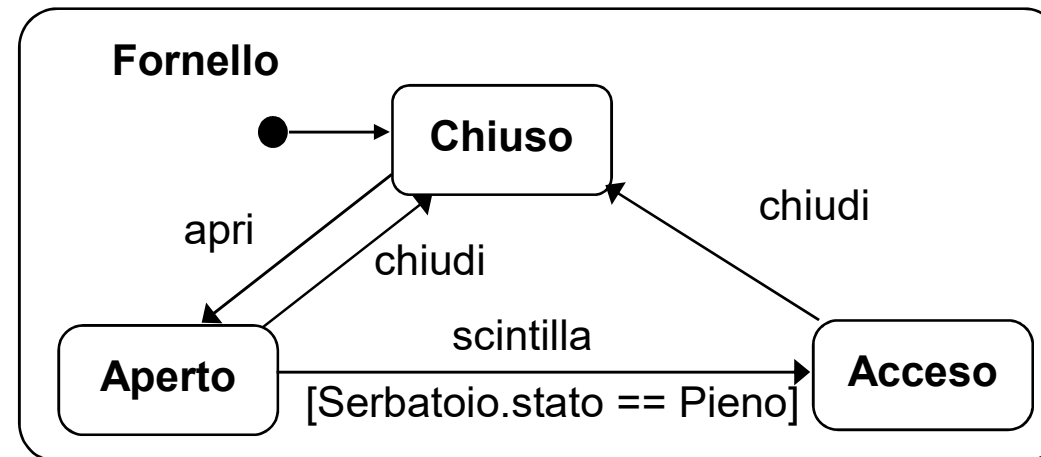
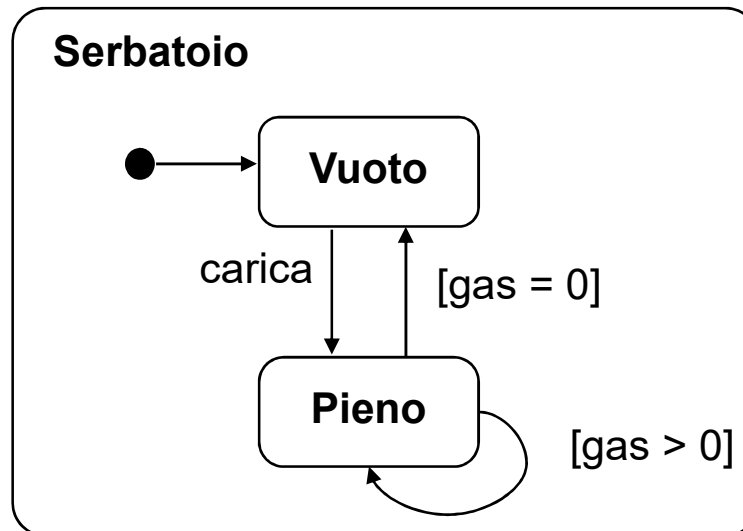
- Interazioni tra oggetti
 - Gli stati dei componenti possono interagire (la guardia che regola le transizioni di un oggetto dipende dallo stato di un altro oggetto)



Aggregazione e concorrenza

UML – State Diagram

- Introduzione
- Diagrammi semplici
- **Diagrammi strutturati**
- Flussi di esecuzione



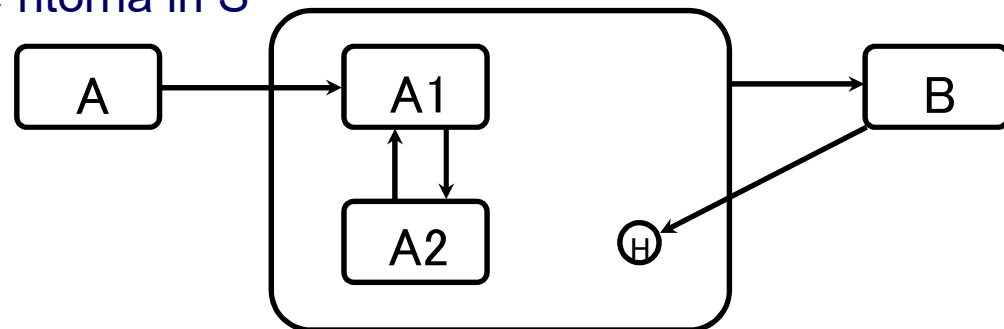


History

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

- History può essere associata a stati non foglia
- Quando l'esecuzione lascia uno stato S con history
 - si salva l'ultimo stato visitato S_1 in S
- Quando l'esecuzione ritorna in S
 - si riparte da S_1



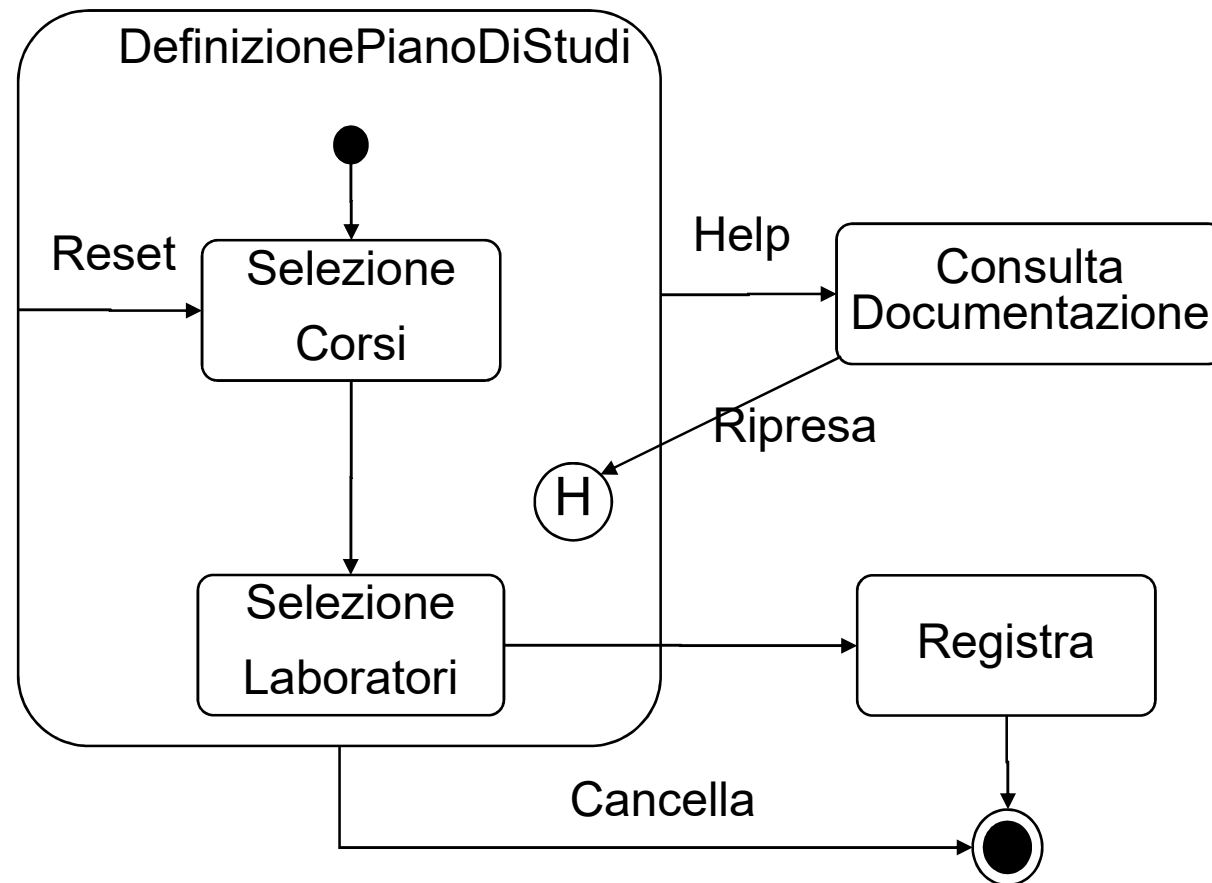
- Shallow history \textcircled{H}
 - si torna all'ultimo stato solo in quel livello
- Deep history $\textcircled{H*}$
 - si torna all'ultimo stato in tutti i livelli



Esempio: Selezione corsi

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

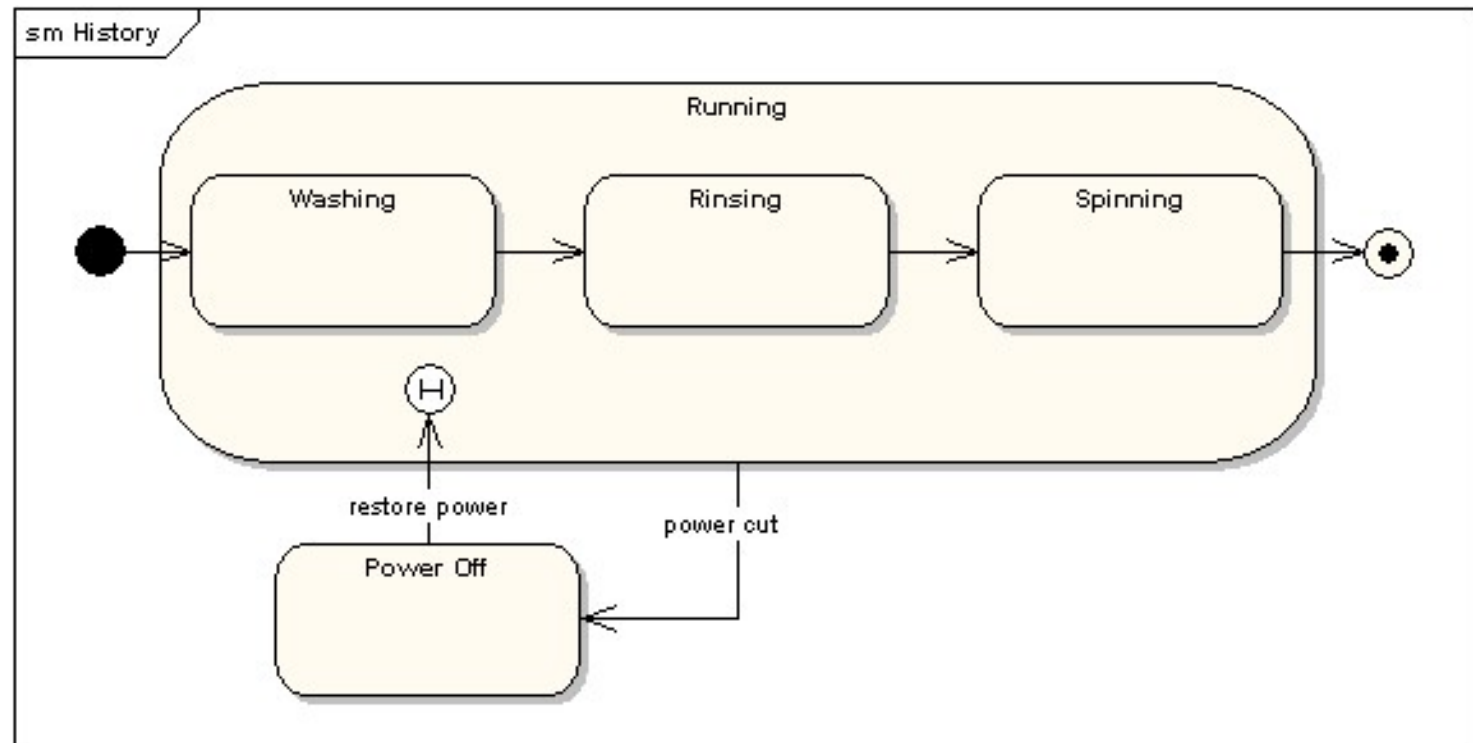




Esempio

UML – State Diagram

- Introduzione
- Diagrammi semplici
- **Diagrammi strutturati**
- Flussi di esecuzione



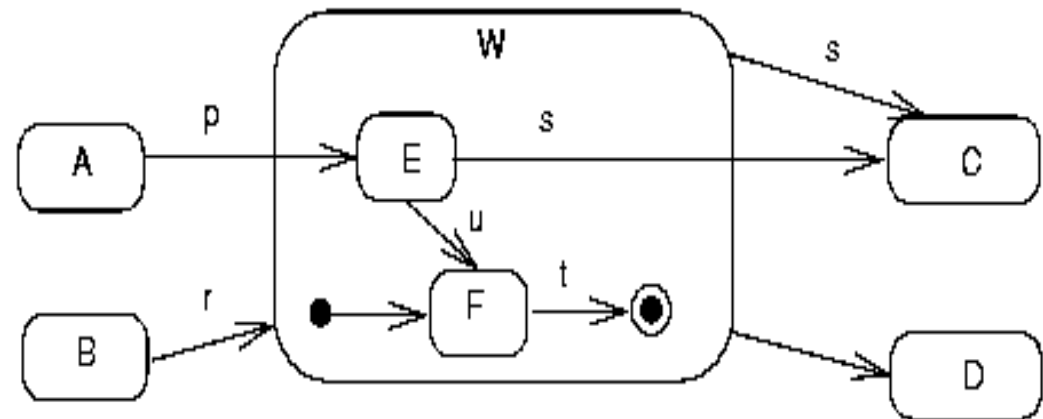


Stubbed transitions

UML – State Diagram

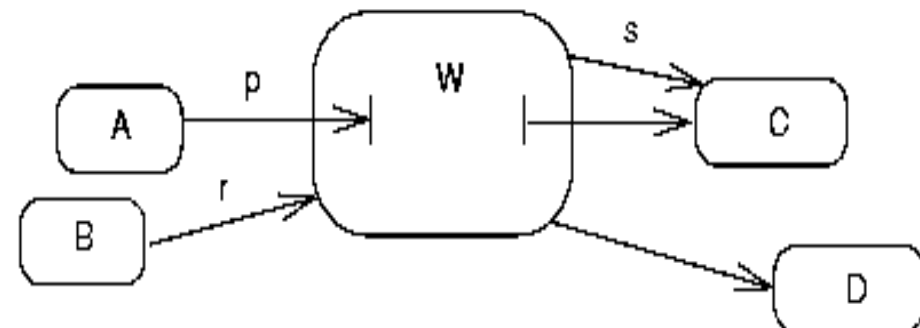
Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione

● Forma completa



may be abstracted as

● Forma astratta

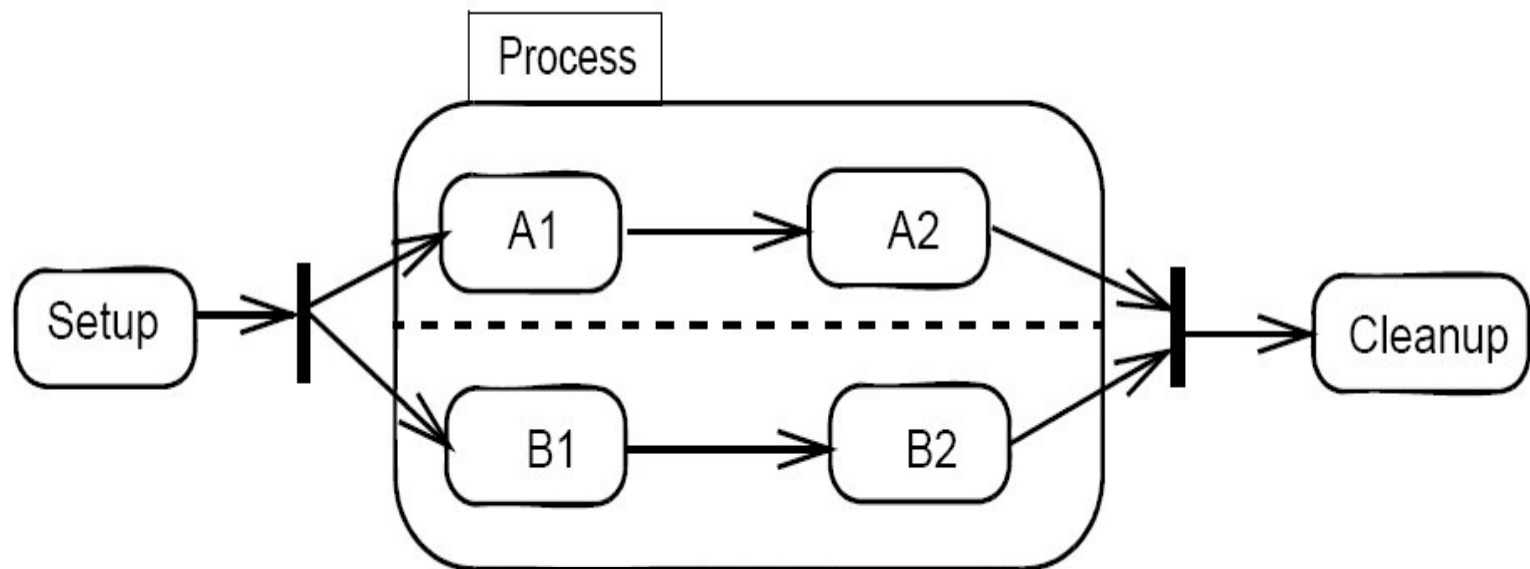




Fork e join

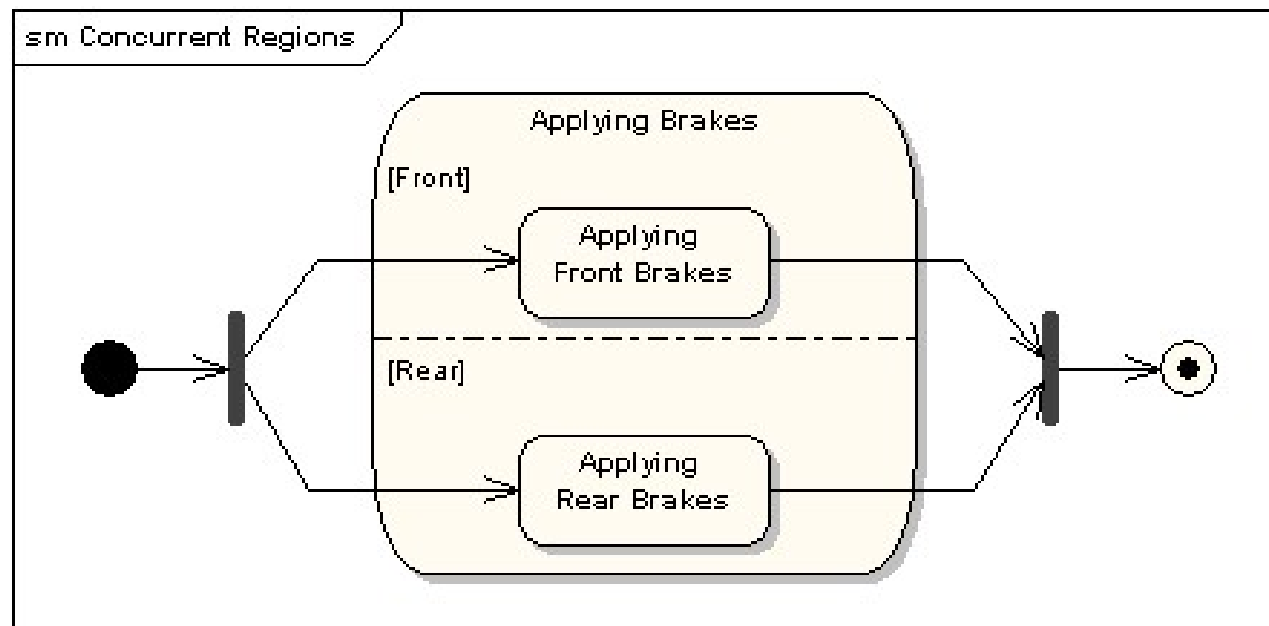
UML – State Diagram

- Introduzione
- Diagrammi semplici
- **Diagrammi strutturati**
- Flussi di esecuzione





● Flussi di esecuzione paralleli

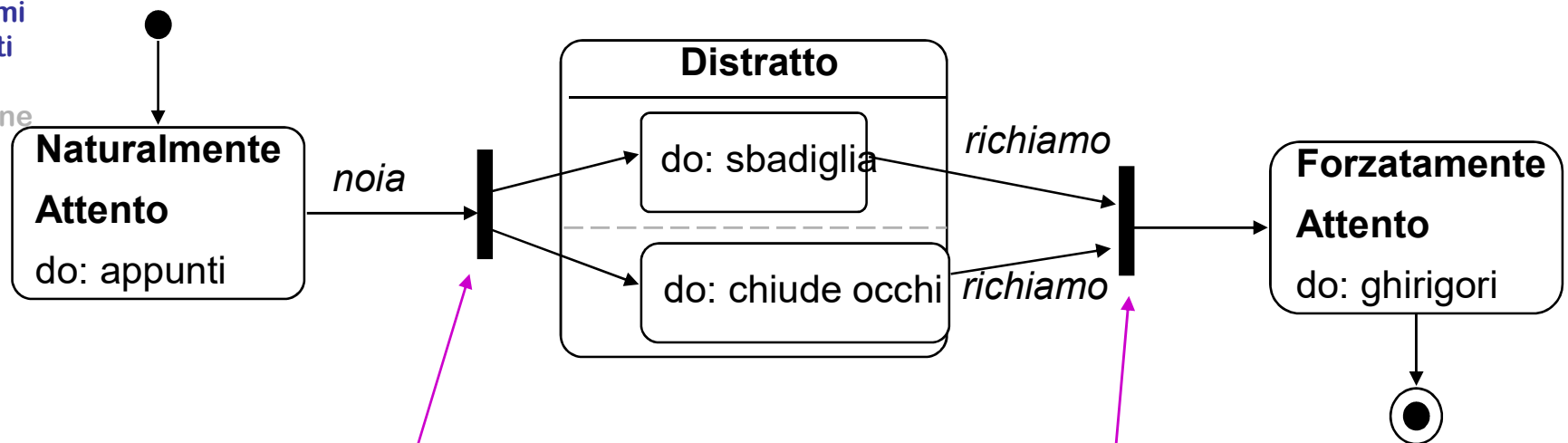




Concorrenza di due attività

UML – State Diagram

Introduzione
Diagrammi
semplici
➤ Diagrammi
strutturati
Flussi di
esecuzione



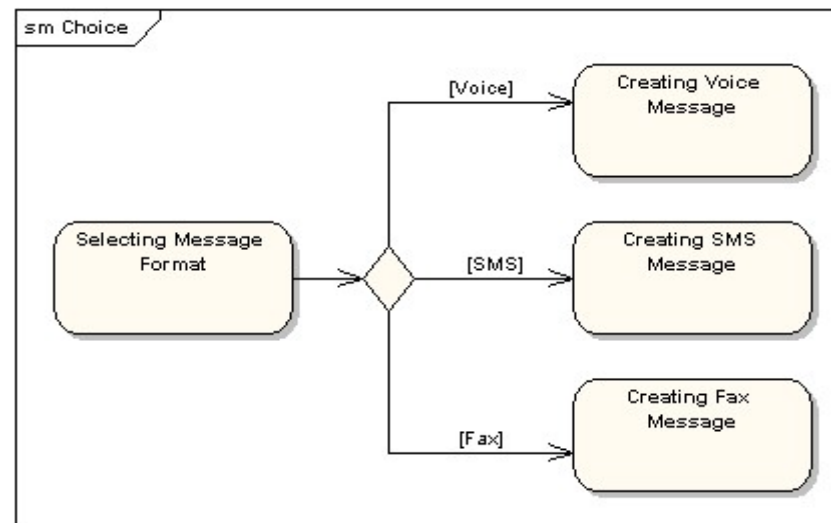
Transizione complessa
(fork)

Sincronizzazione
(join)



● Meccanismo di selezione

- in dipendenza dal risultato dell'elaborazione precedente, inclusa l'elaborazione sulla transizione in ingresso allo stato
 - scelta “dinamica”
- rappresentato da un rombo

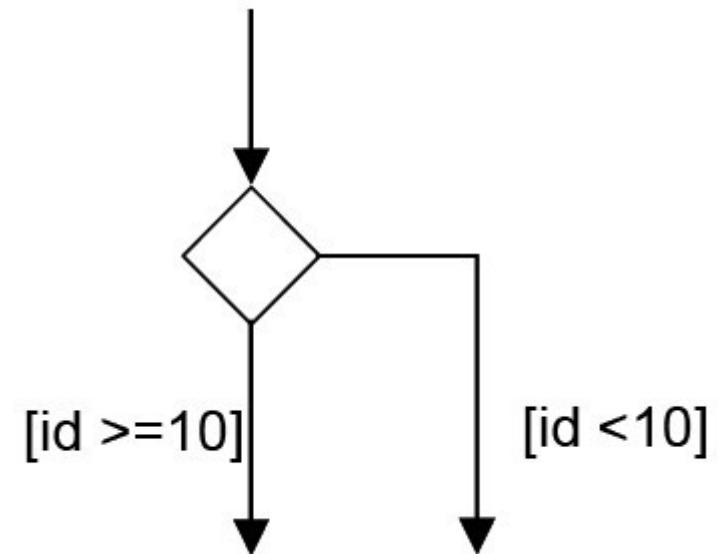
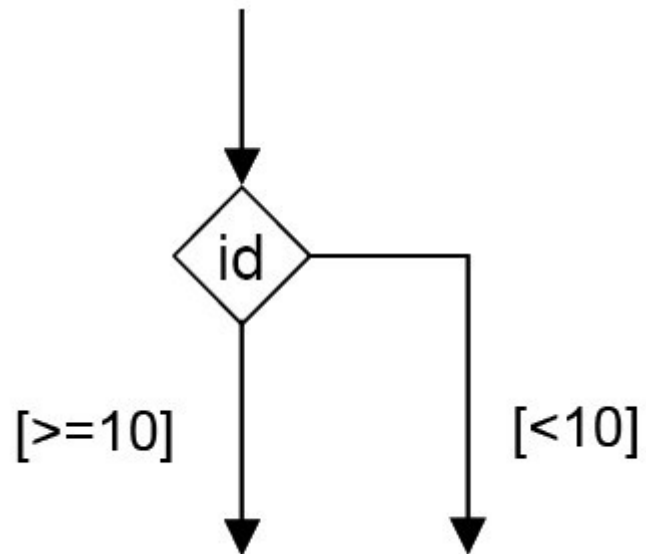




Pseudo-stato di scelta

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione

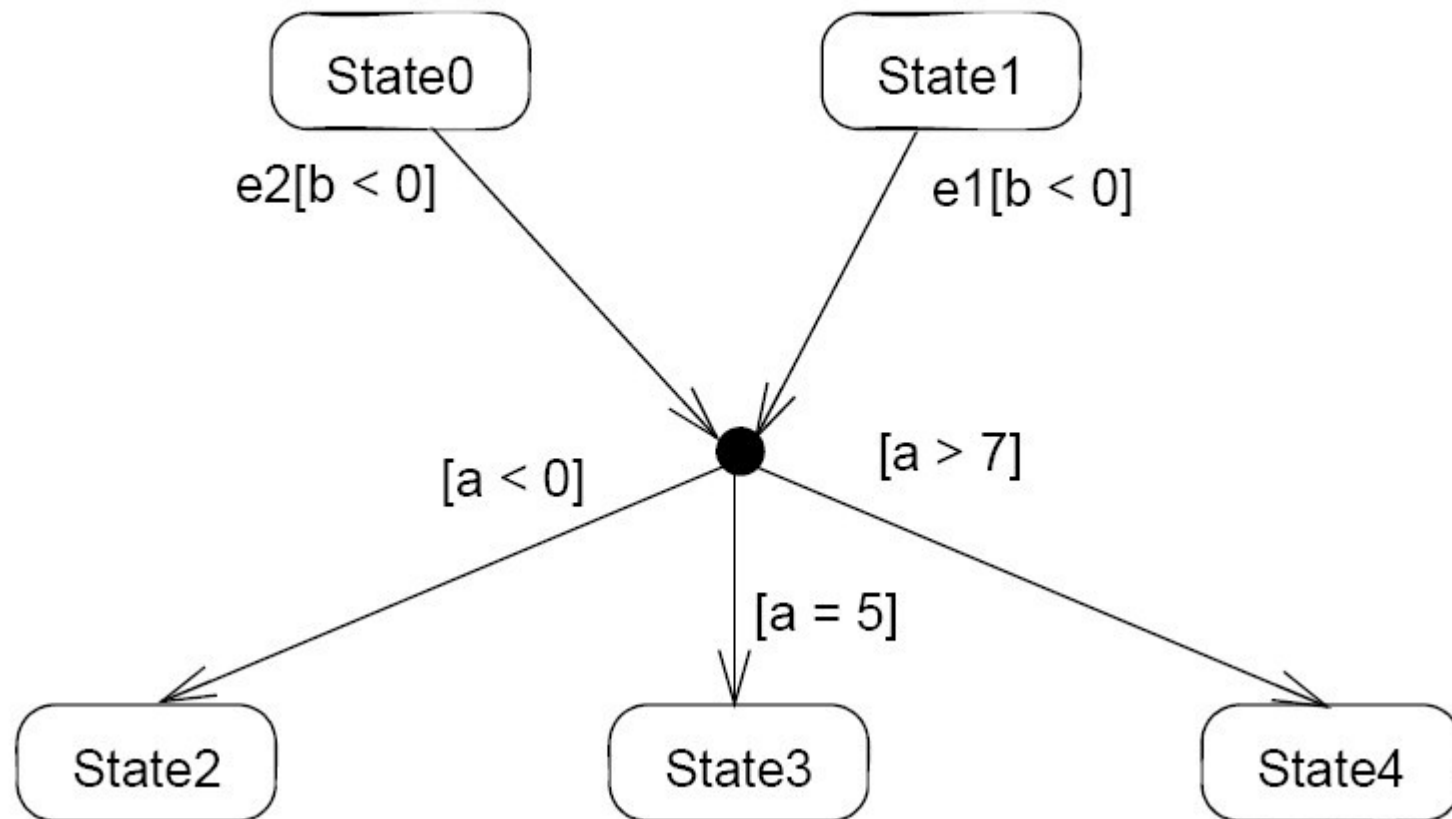




Giunzione (junction)

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione



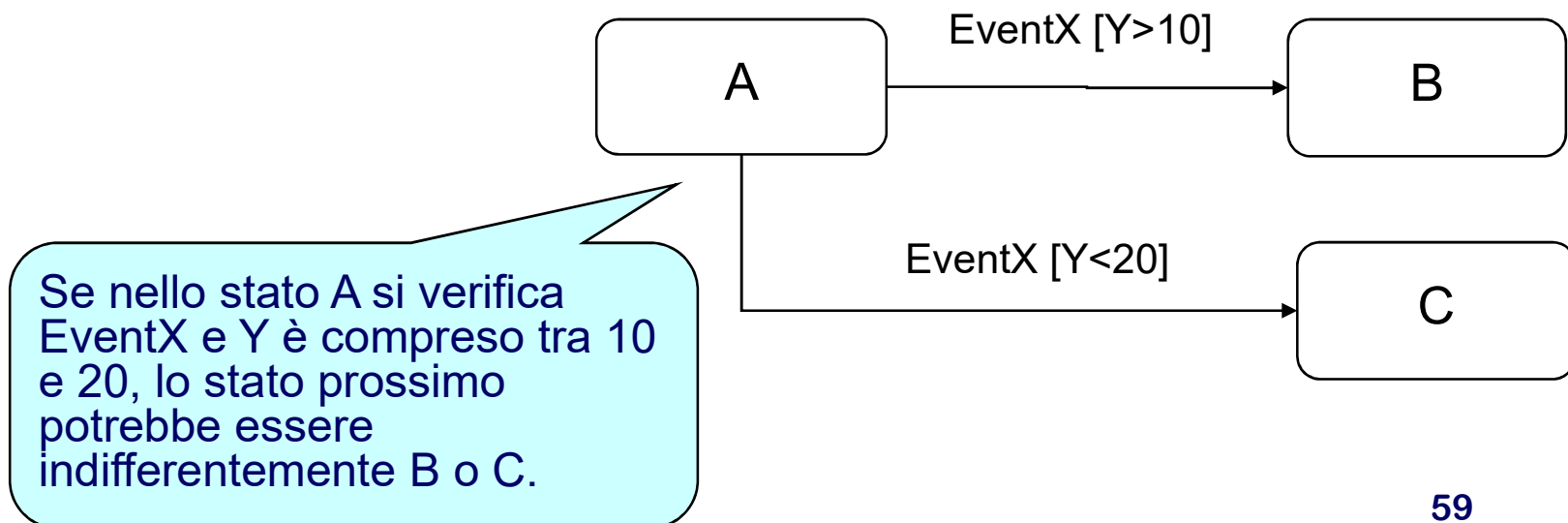


Conflitto tra transizioni

UML – State Diagram

Introduzione
Diagrammi
semplici
Diagrammi
strutturati
➤ Flussi di
esecuzione

- Più transizioni abilitate possono essere in conflitto tra loro.
 - es. quando due transizioni hanno origine dallo stesso stato, sono sollecitate dallo stesso evento, ma con guardie diverse e non mutuamente esclusive possono essere in conflitto.
- In questo caso solo una transizione scatterà





Priorità tra transizioni

UML – State Diagram

Introduzione
Diagrammi
semplici
Diagrammi
strutturati
➤ Flussi di
esecuzione

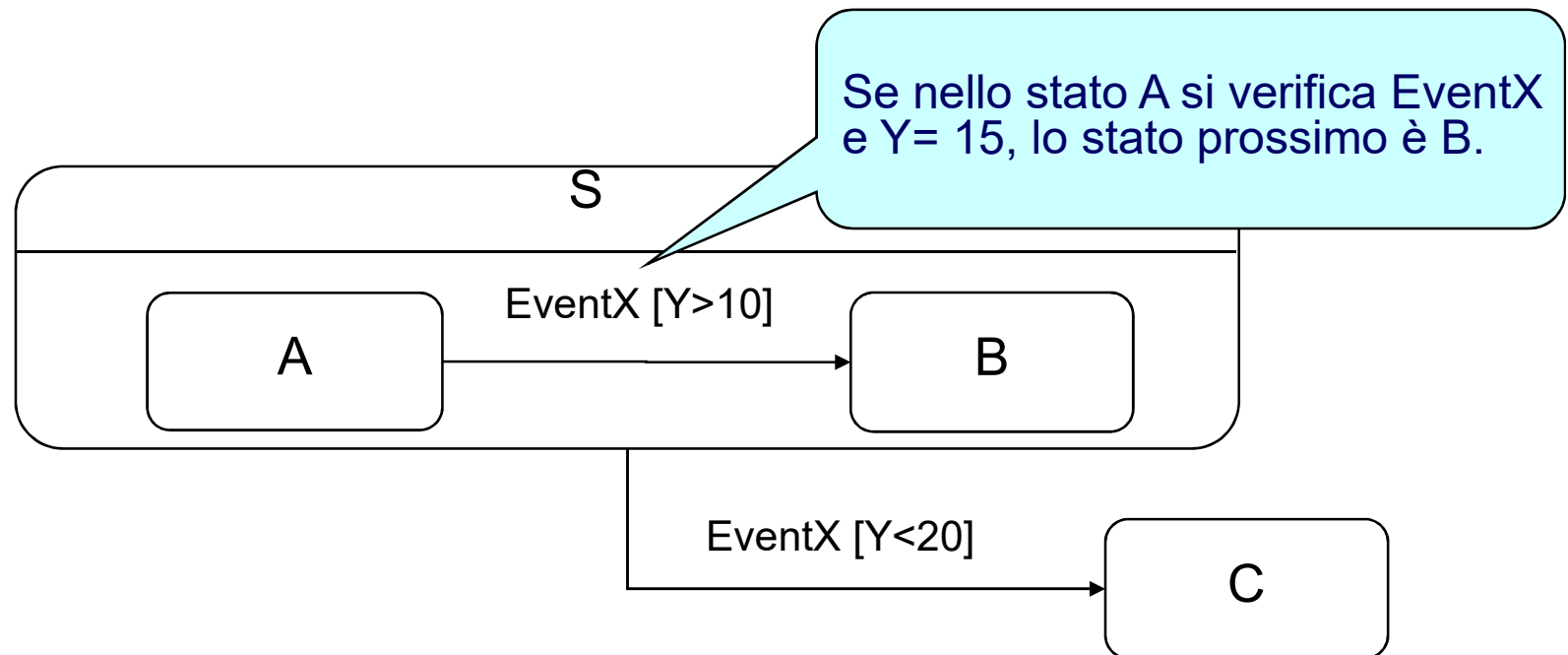
- In situazioni di conflitto, la scelta della transizione da sparare è parzialmente basata su priorità implicite
 - che risolvono alcuni conflitti, ma non tutti
- Le priorità sono basate sulla posizione relativa nella gerarchia degli stati
 - la transizione che ha origine da un sotto-stato ha priorità più alta di una transizione che origina da uno stato contenente



Priorità tra transizioni

UML – State Diagram

- Introduzione
- Diagrammi semplici
- Diagrammi strutturati
- Flussi di esecuzione





State machine: suggerimenti metodologici

UML 2

- Rappresentare solo classi con dinamica significativa
- Usare "scenari" e "tracce degli eventi"
- Solo attributi ed eventi rilevanti
- Livello di astrazione adeguato
 - granularità di eventi e stati adatta all'applicazione
- Distinguere tra attività ed azioni
- Usare diagrammi strutturati dove appropriato
- Verificare la consistenza incrociata
 - attenzione agli eventi condivisi
- Diagrammi delle sottoclassi indipendenti da quelli delle superclassi