

Macchine a stati

Macchina a stati

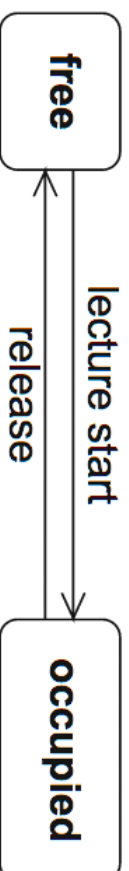
- Una macchina a stati describe il comportamento dinamico delle istanze di un classificatore (per esempio degli oggetti istanza di una classe).
- Per costruire una macchina a stati dobbiamo individuare gli stati significativi in cui si puo' trovare un oggetto durante la sua vita.
- Inoltre dobbiamo descrivere come da ciascuno di questi stati l'oggetto puo' passare (transire) in un altro.
- Le transizioni avvengono in risposta al verificarsi di un evento. Gli eventi sono tipicamente;
 - messaggi inviati da altri oggetti
 - eventi generati internamente
- Una macchina a stati è rappresentata con un grafo di stati e transizioni, associata a un classificatore

Stato

- Uno stato è un insieme di valori (di alcune variabili significative) di un oggetto:
 - È un'astrazione dello stato concreto dell'oggetto (caratterizzato dai valori di tutte le variabili)
 - Rappresenta uno stato significativo
 - È caratterizzato dal dare la stessa risposta qualitativa ad eventi che possono accadere.
- Uno stato ha un nome unico
- Uno stato può essere composito (più avanti)

Importante il livello di dettaglio

Per modellare un'aula



che verra' poi specificato a livello di implementazione



LectureHall	
- free: boolean	
+ occupy()	
+ release()	

```
class LectureHall {
    private boolean free;
    public void occupy() {
        free=false;
    }
    public void release() {
        free=true;
    }
}
```

Sintassi di base

Gli stati sono rappresentati con rettangoli arrotondati

- Il disco nero marca l'inizio. Non è uno stato vero e proprio ma un marcatore che punta allo stato da cui partire.
- Il disco nero bordato (nodo finale), indica la terminazione.
- Possono comparire in qualunque numero all'interno di un diagramma

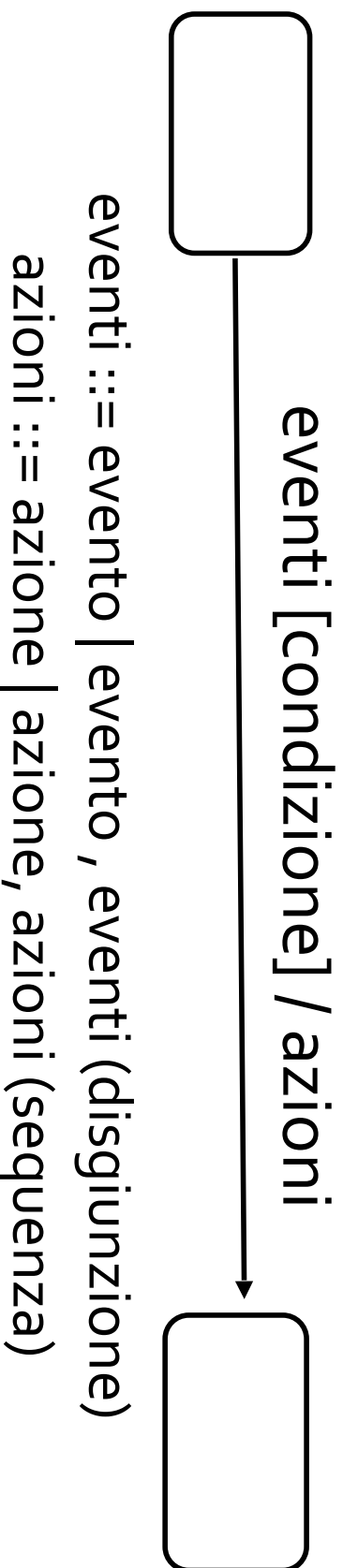
Stato: 

Stato iniziale: ●

Stato finale: 

Transizione

- Una transizione collega tra loro due stati, è rappresentata con una freccia
- L'uscita da uno stato definisce la risposta dell'oggetto all'occorrenza di un evento, viene presa solo se la condizione è vera, e comporta l'esecuzione delle azioni specificate

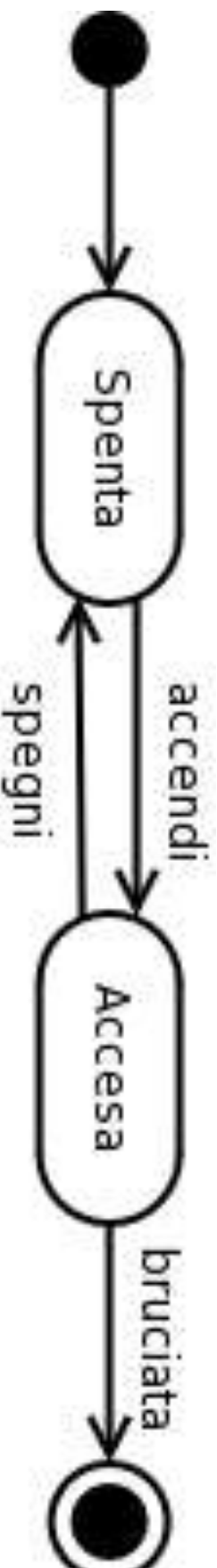


Tutti opzionali anche se l'evento è bene che ci sia.

Solo nelle transizioni di completamento (più avanti) l'evento non serve.

Esempio stati di una lampadina

Descriviamo la vita di una lampadina



Lampadina
accesa : boolean = false
accendi()
spegni()

OSSERVAZIONE: gli eventi (anche se non tutti) corrispondono alle operazioni della classe

Evento

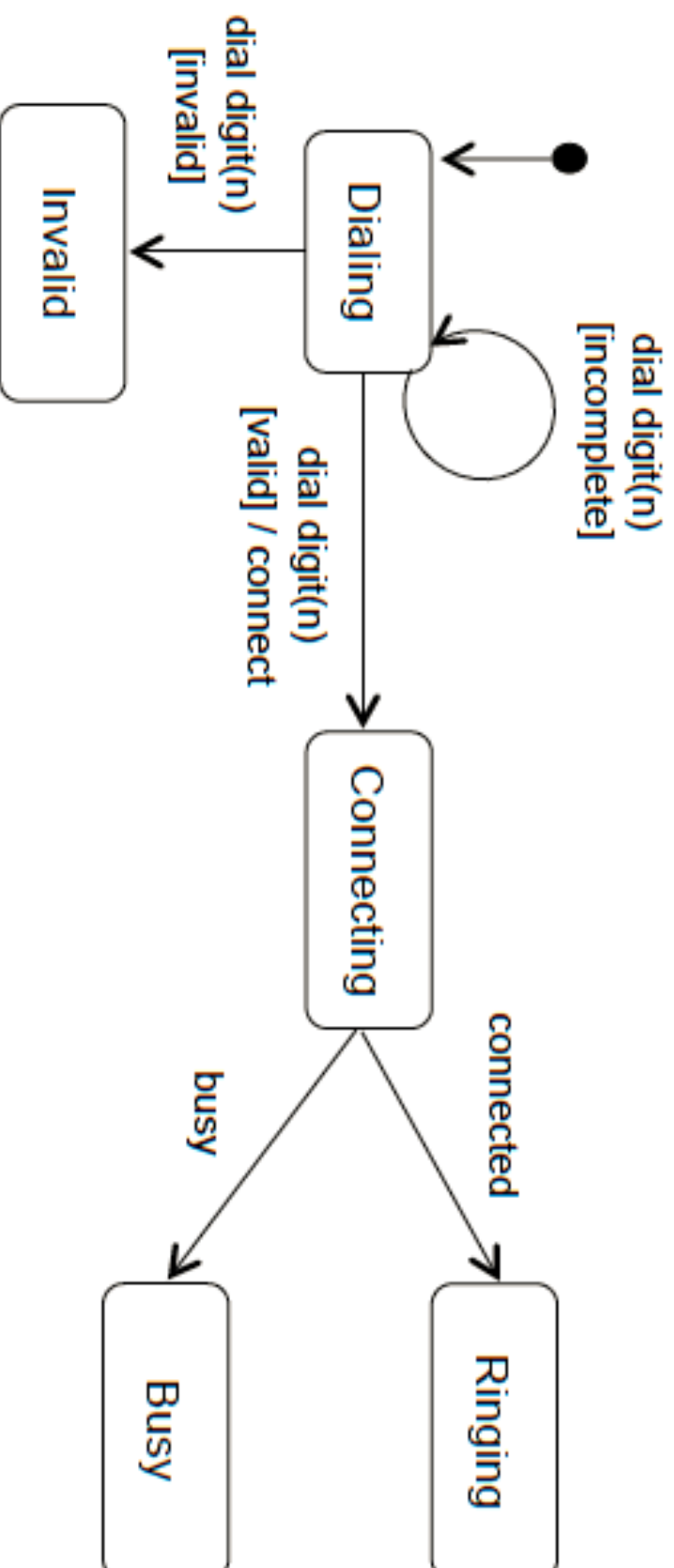
- Un evento è l'occorrenza di un fenomeno collocato nel tempo e nello spazio
- Un evento occorre istantaneamente
- Modellate qualcosa come un evento se ha delle conseguenze
- Gli eventi che arrivano in uno stato per cui non è prevista alcuna transizione vengono ignorati
- È ammesso il non-determinismo: un evento può fare da trigger a più transizioni:
 - Se le due transizioni escono dallo stesso stato, ne viene scelta una non-deterministicamente

Tipi di evento

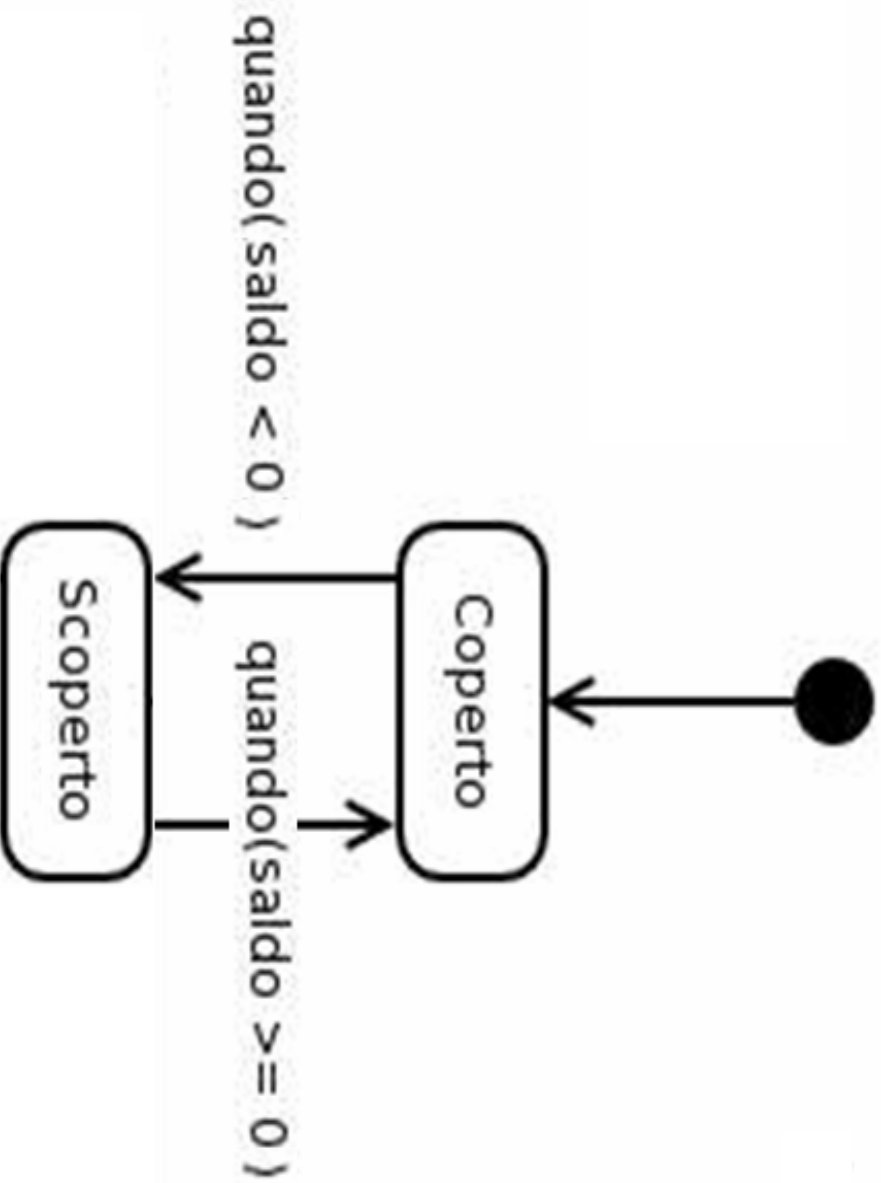
- **Operazione o segnale** **op(a:T)**
 - la transizione è abilitata quando l'oggetto (in quello stato) riceve una chiamata di metodo / un segnale con parametri (a) e tipo (T) (i parametri sono opzionali)
- **Evento di variazione** **quando(exp)**
 - la transizione è abilitata appena l'espressione diventa vera
 - l'espressione può indicare un tempo assoluto o una condizione su variabili
 - spesso in inglese: **when(exp)**
- **Evento temporale** **dopo(time)**
 - la transizione è abilitata dopo che l'oggetto è stato fermo "time" in quello stato
 - spesso in inglese: **after(time)**

Evento operazione o segnale

- Operazioni della classe telefono (dial digit(n)) o segnali (busy e connected)

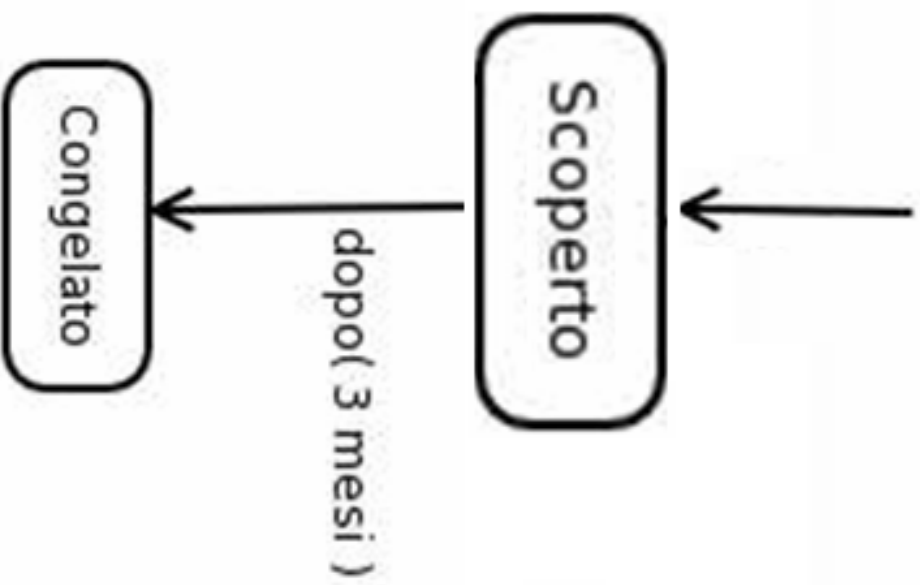


Eventi di variazione (esempio)



- Un evento occorre in modo istantaneo
- una condizione non è istantanea
- è istantaneo il momento in cui diventa vera

Eventi temporali (esempio)



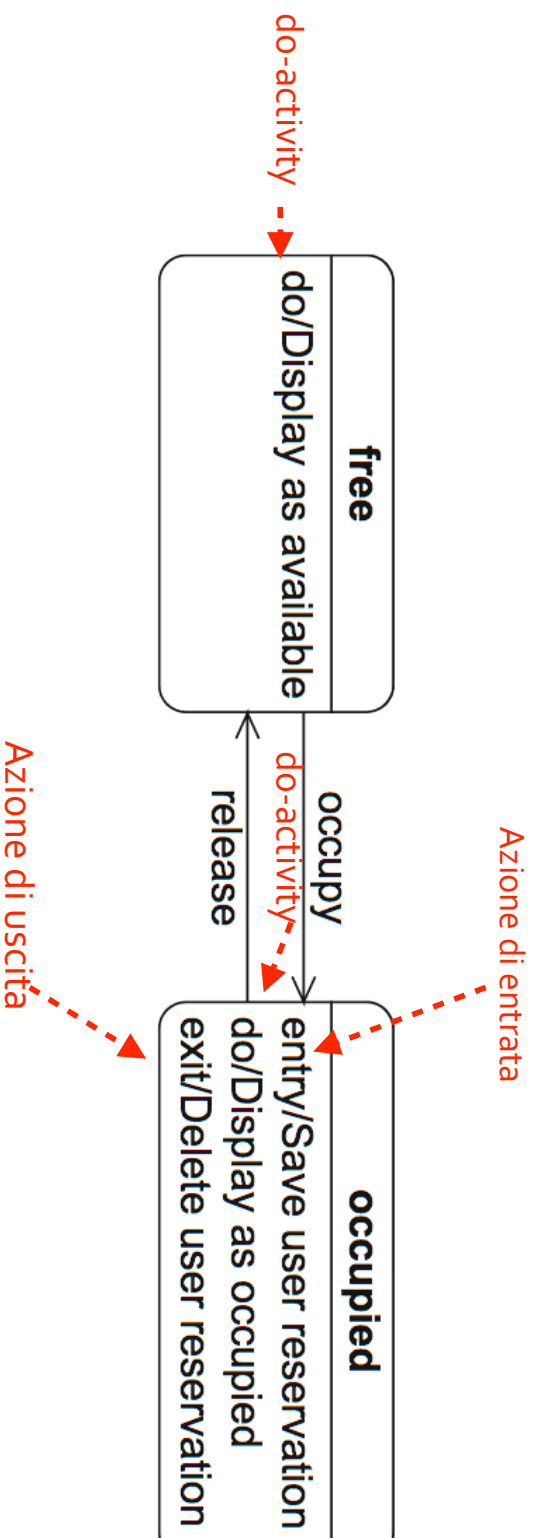
Dopo che l'oggetto è stato
3 mesi nello stato Scoperto,
transisce nello stato
Congelato

Transizioni e attività interne

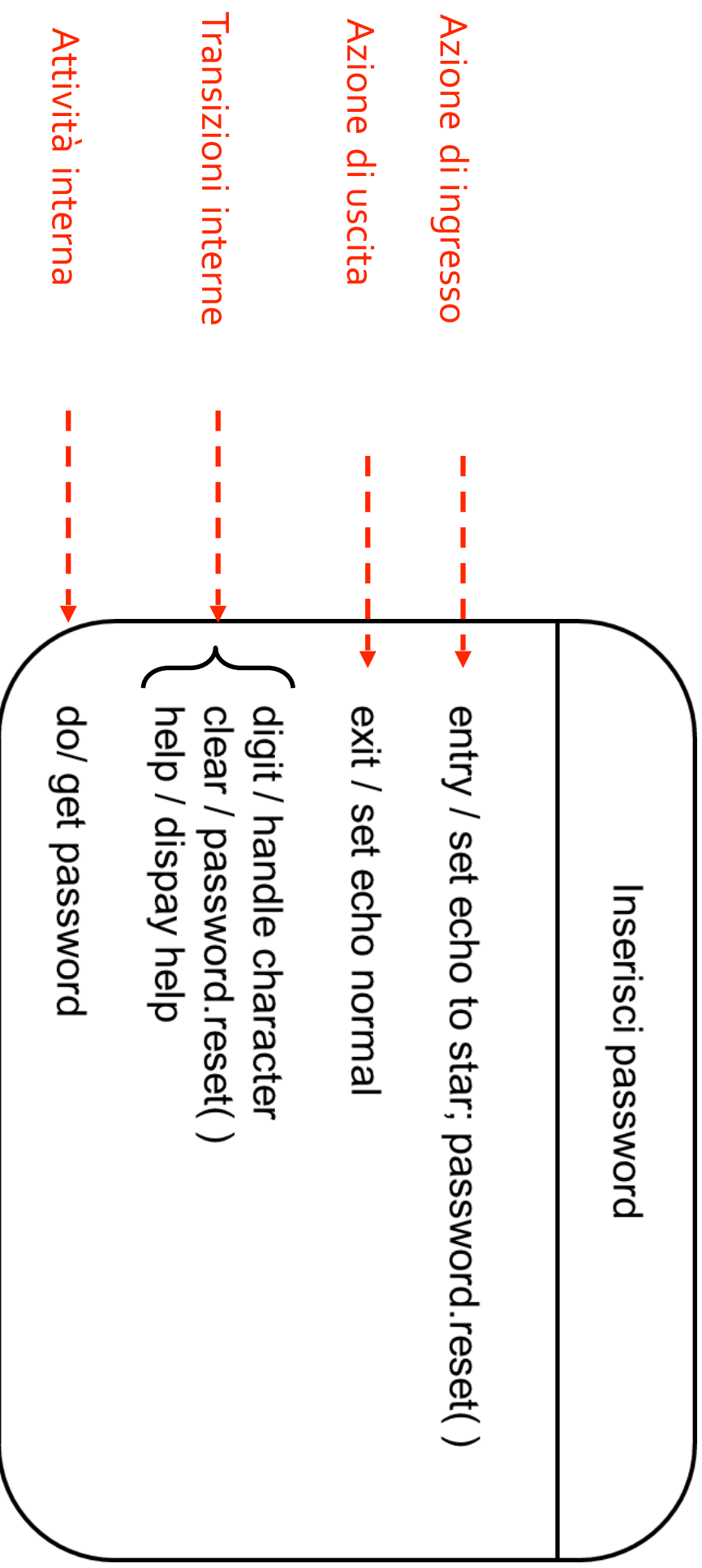
- Transizione interna: risposta a un evento che causa solo l'esecuzione di azioni. Esempi:
 - **Azione di entrata**: eseguita all'ingresso in uno stato
 - **Azione di uscita**: eseguita all'uscita di uno stato
 - **Transizione interna**: risposta ad un evento
- **Attività interna (Do-activity)**: eseguita in modo continuato mentre l'oggetto si trova in quello stato (senza necessità di un evento scatenante), al contrario di tutte le altre azioni che sono atomiche:
 - consuma del tempo
 - può essere interrotta (quando un evento fa uscire dallo stato)

Esempio

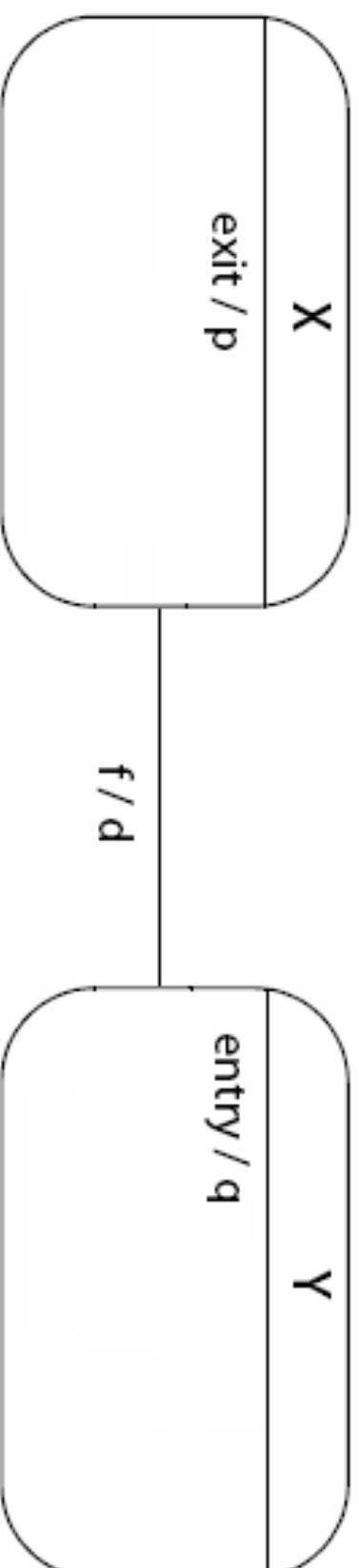
Ritorniamo sull'esempio dell'aula:



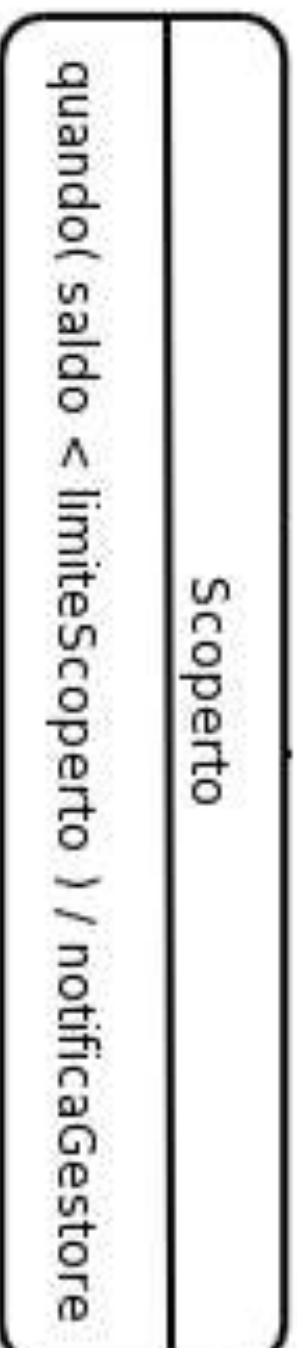
Sintassi



Esempi

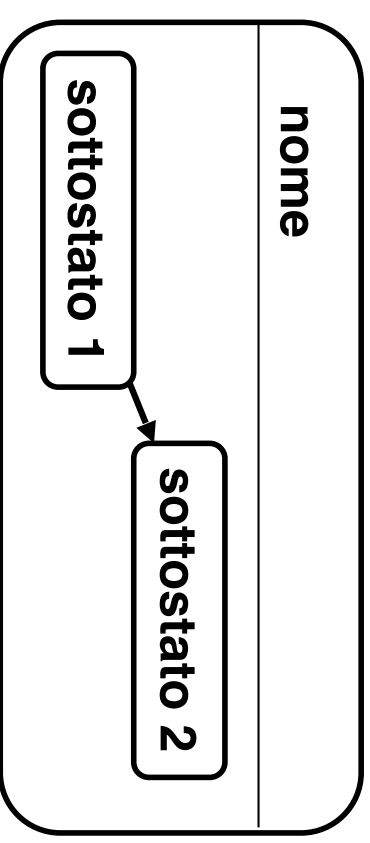


effective result: f / p; d; q

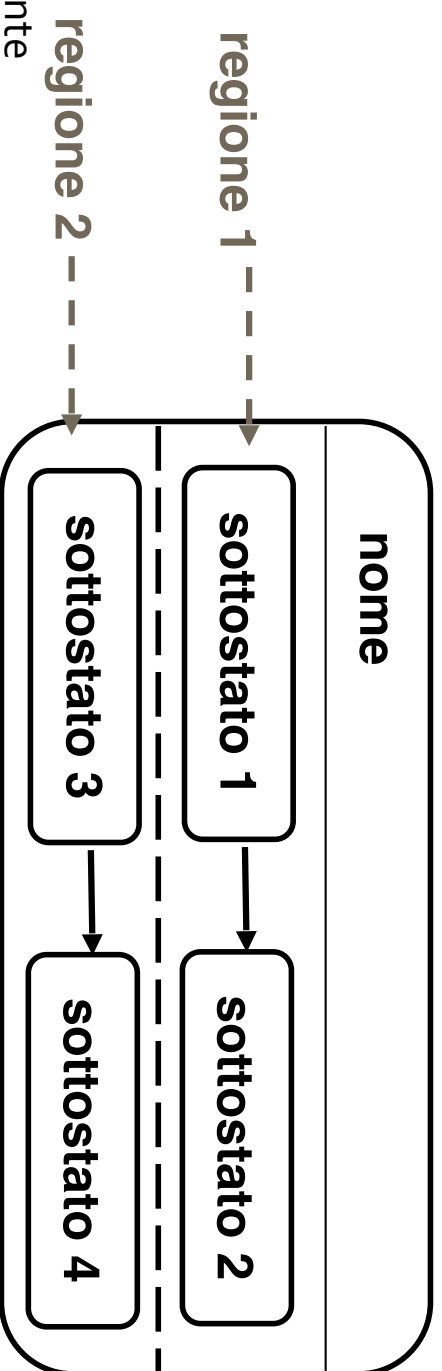


Stati compositi

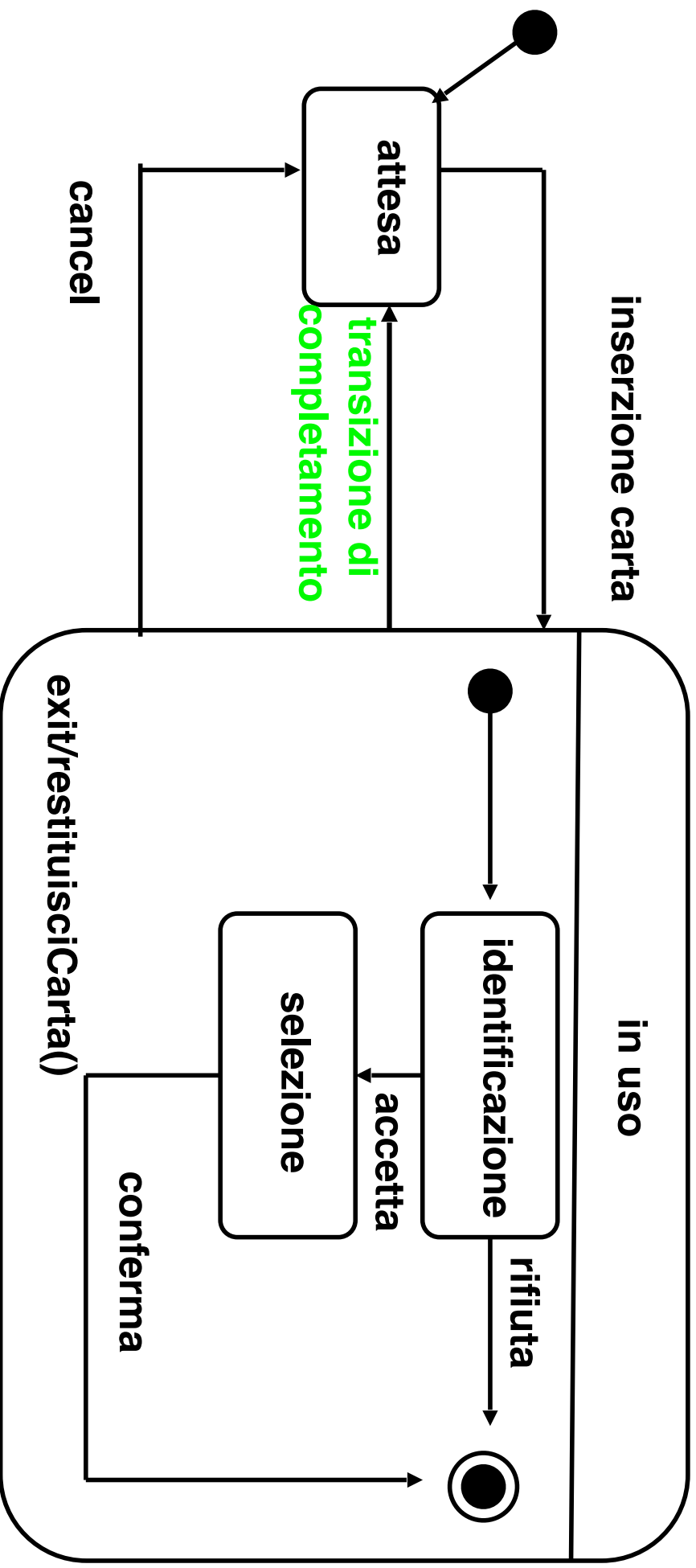
- **Composito**
- **sequenziale:**
- Un sottostato attivo in ogni istante



- **Composito**
- **parallelo:**
- sottostati attivi contemporaneamente
- uno per regione

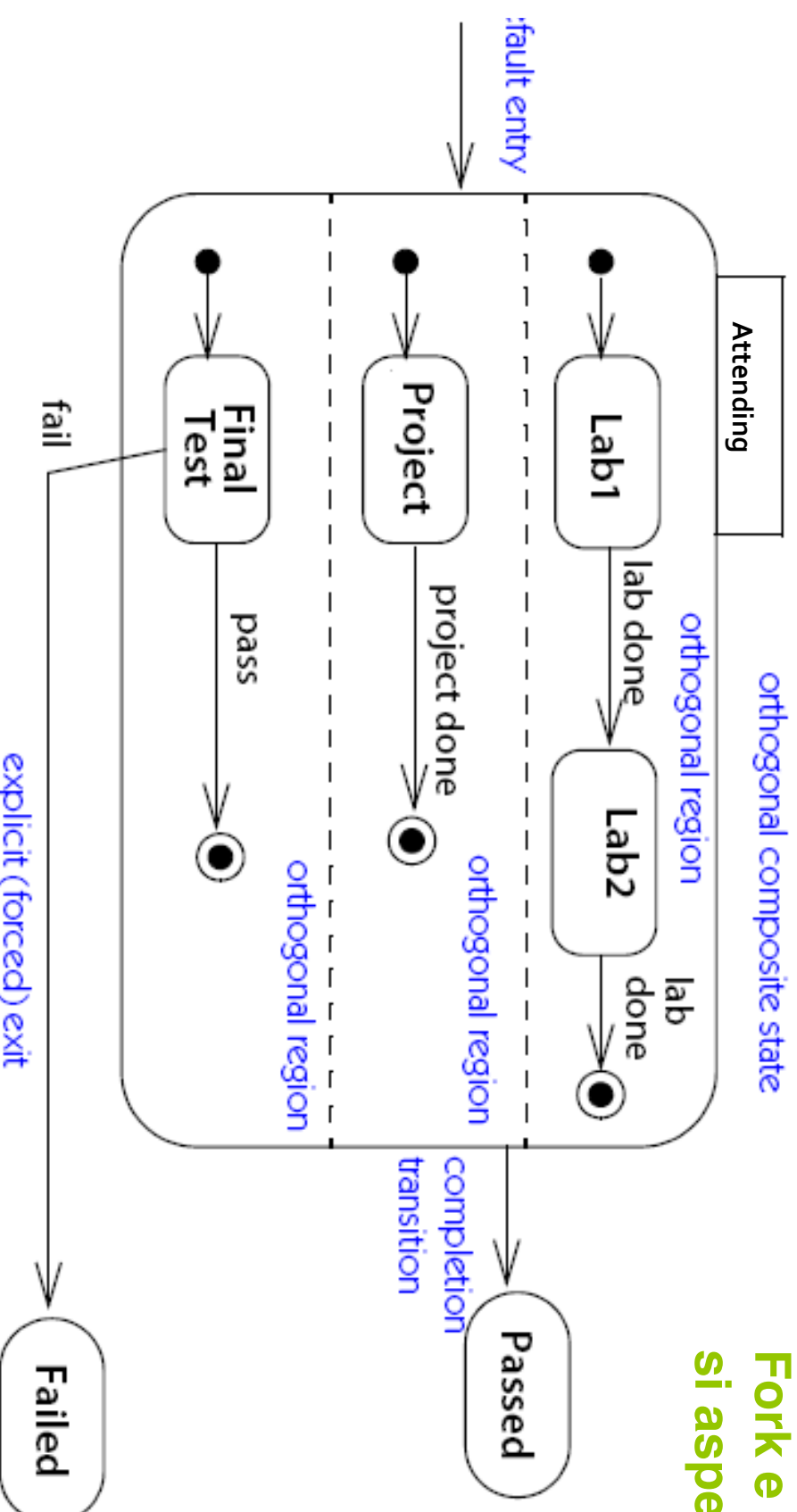


Esempio stato composito sequenziale



- Ogni transizione che arriva sul bordo prosegue nello stato iniziale
- Dallo stato finale (dopo le exit) si prosegue nella **transizione di completamento**
- Ogni transizione (non di completamento) che parte dal bordo si intende possibile **da un qualsiasi stato interno**

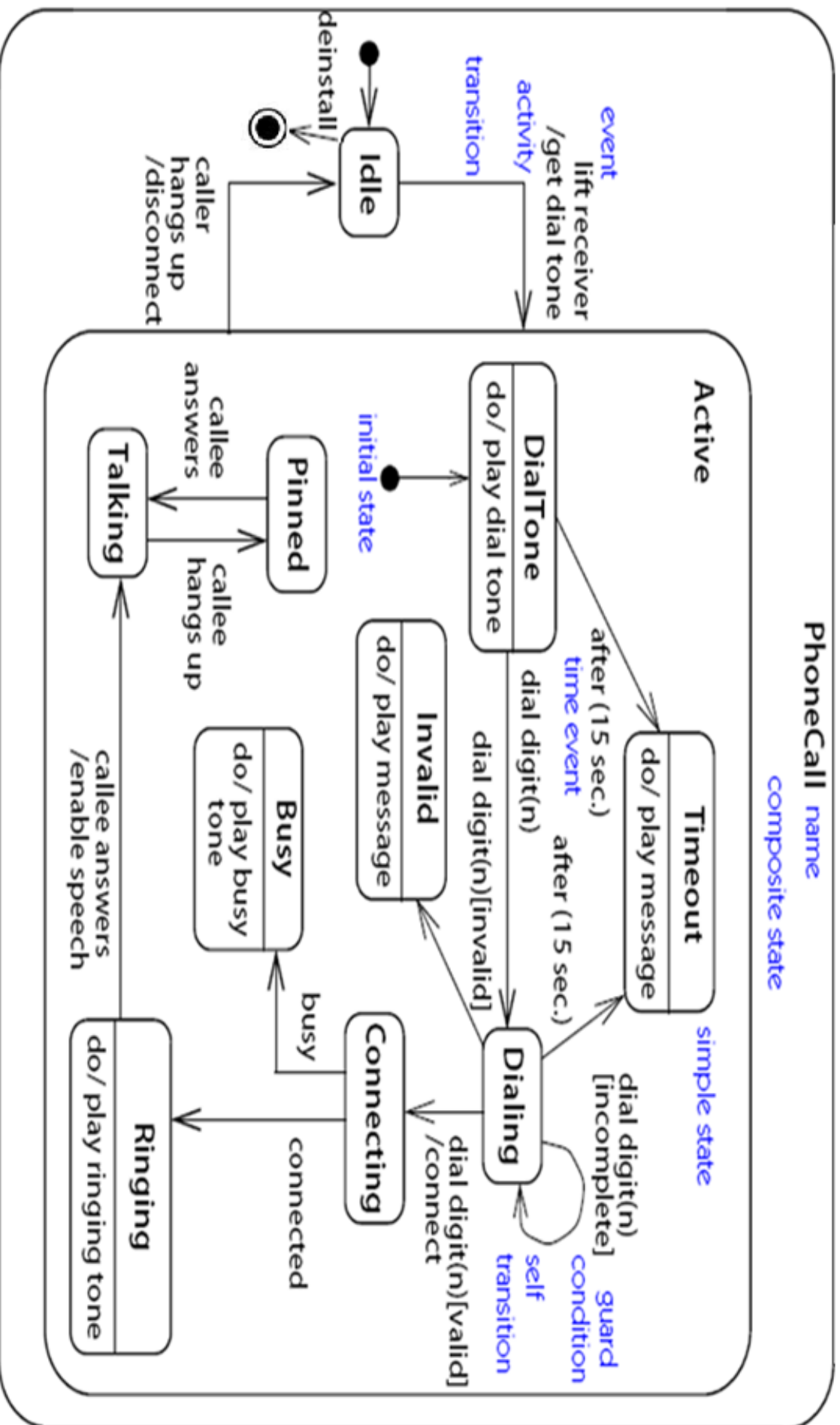
stato composito parallelo un esempio: corso di laboratorio



Fork e join implicite:
si aspetta per uscire

- Ogni transizione che arriva sul bordo prosegue **in tutti gli stati iniziali** **explicit (forced) exit**
- Una volta raggiunti **tutti gli stati finali** si prosegue nella transizione di completamento
- Possono esserci transizioni che bucano il bordo e si intendono possibili **dal solo stato interno** a cui sono collegate

Esempio: stato composito sequenziale senza stato finale



Pinned=trattenuto