

# Ingegneria del Software

## 9. Macchine a stati

Dipartimento di Informatica  
Università di Pisa  
A.A. 2014/15

# so far...

- Modello del dominio
  - Modello statico: diagrammi delle classi
  - Modello dinamico : diagrammi di attività (business model)  
[diagrammi di macchina a stati]
- Requisiti
  - Modello statico: diagramma dei casi d'uso
  - Modello dinamico: narrazioni associate ai casi d'uso

# macchina a stati

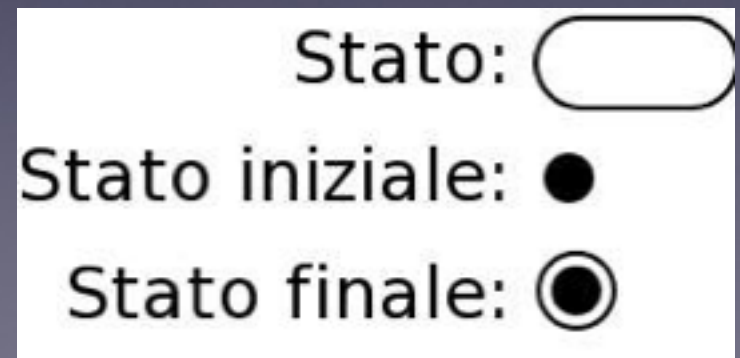
- Una macchina a stati è un grafo di stati e transizioni, associata a un classificatore (classe, sistema,...)
- Per ogni classe può esistere una macchina a stati che modella tutte le transizioni di stato di tutti gli oggetti di quella classe, in risposta a diversi tipi di evento
- Gli eventi sono tipicamente dei messaggi
  - inviati da altri oggetti
  - generati internamente
- La macchina a stati di una classe modella il comportamento degli oggetti della classe in modo trasversale per tutti i casi d'uso interessati

# uno stato

- Uno stato descrive un periodo di tempo nella vita di un oggetto
- Può essere caratterizzato come
  - Un insieme di valori qualitativamente simili
  - Un periodo di tempo durante il quale un oggetto attende il verificarsi di un evento
  - Un periodo di tempo durante il quale un oggetto svolge un'attività
- Uno stato ha un nome unico
- Uno stato può essere composito (più avanti)

# sintassi di base

- Gli stati sono rappresentati con rettangoli arrotondati
- Il disco nero marca l'inizio dell'esecuzione
  - Non è uno stato vero e proprio ma un marcatore che punta allo stato da cui partire
- Il disco nero bordato (nodo finale) indica che l'esecuzione è terminata
- Possono comparire in qualunque numero all'interno di un diagramma



# transizioni

- Una transizione collega tra loro due stati ed è rappresentata con una freccia
- L'uscita da uno stato definisce la risposta dell'oggetto all'occorrenza di un evento



eventi ::= evento | evento, eventi (disgiunz.)  
azioni ::= azione | azione, azioni (sequenza)

# un esempio



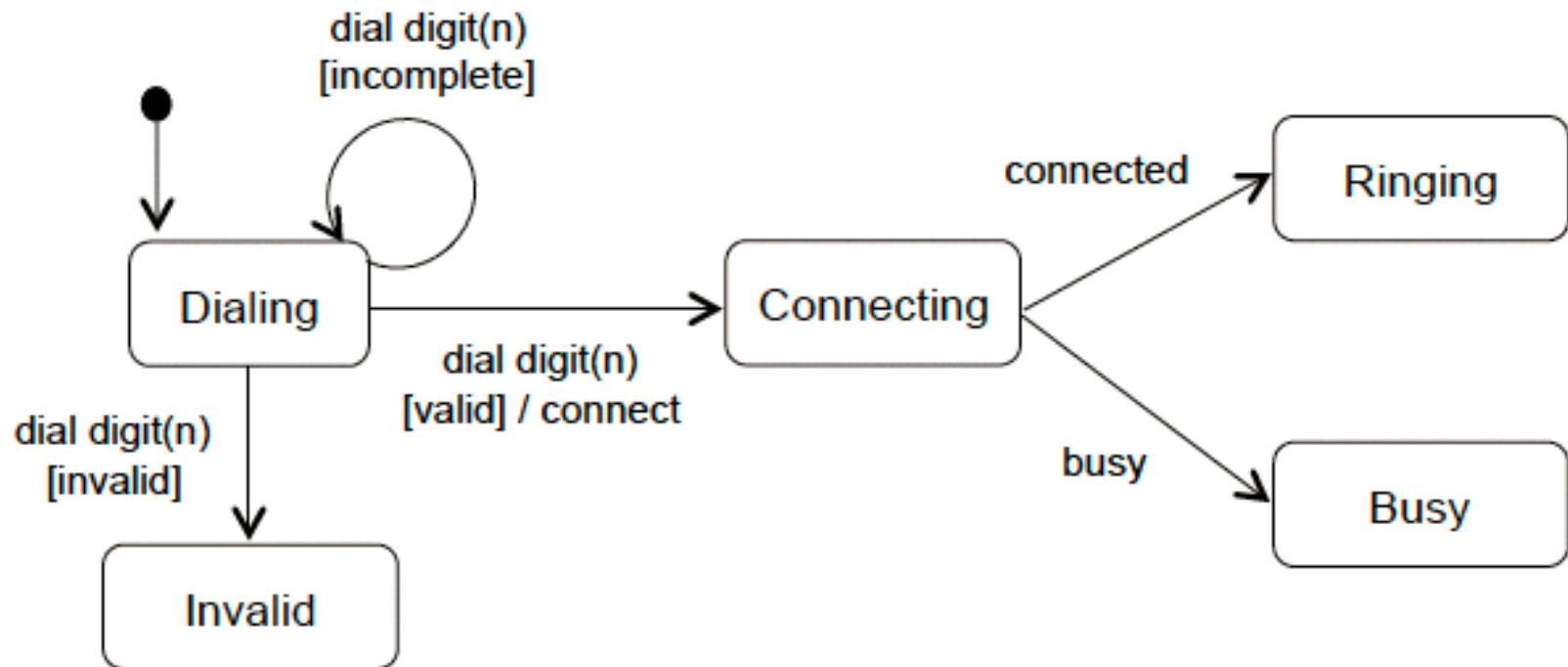
- L'evento è quasi sempre presente: non obbligatorio, ma in generale se non c'è il modello è mal costruito
  - Nelle transizioni di completamento (più avanti) si omette

# eventi

- Un evento è l'occorrenza di un fenomeno collocato nel tempo e nello spazio
  - che occorre istantaneamente
- Gli eventi che arrivano in uno stato per cui non è prevista alcuna transizione vengono ignorati
- È ammesso il non-determinismo e il broadcast
  - “A transition on a nested state takes precedence over a transition on one of its enclosing states. If two conflicting transitions are enabled at the same time, one of them fires non-deterministically”
  - “If multiple states are active, a single event occurrence may trigger a transition in each orthogonal region” [v. più avanti sugli stati compositi]



# un esempio: chiamata

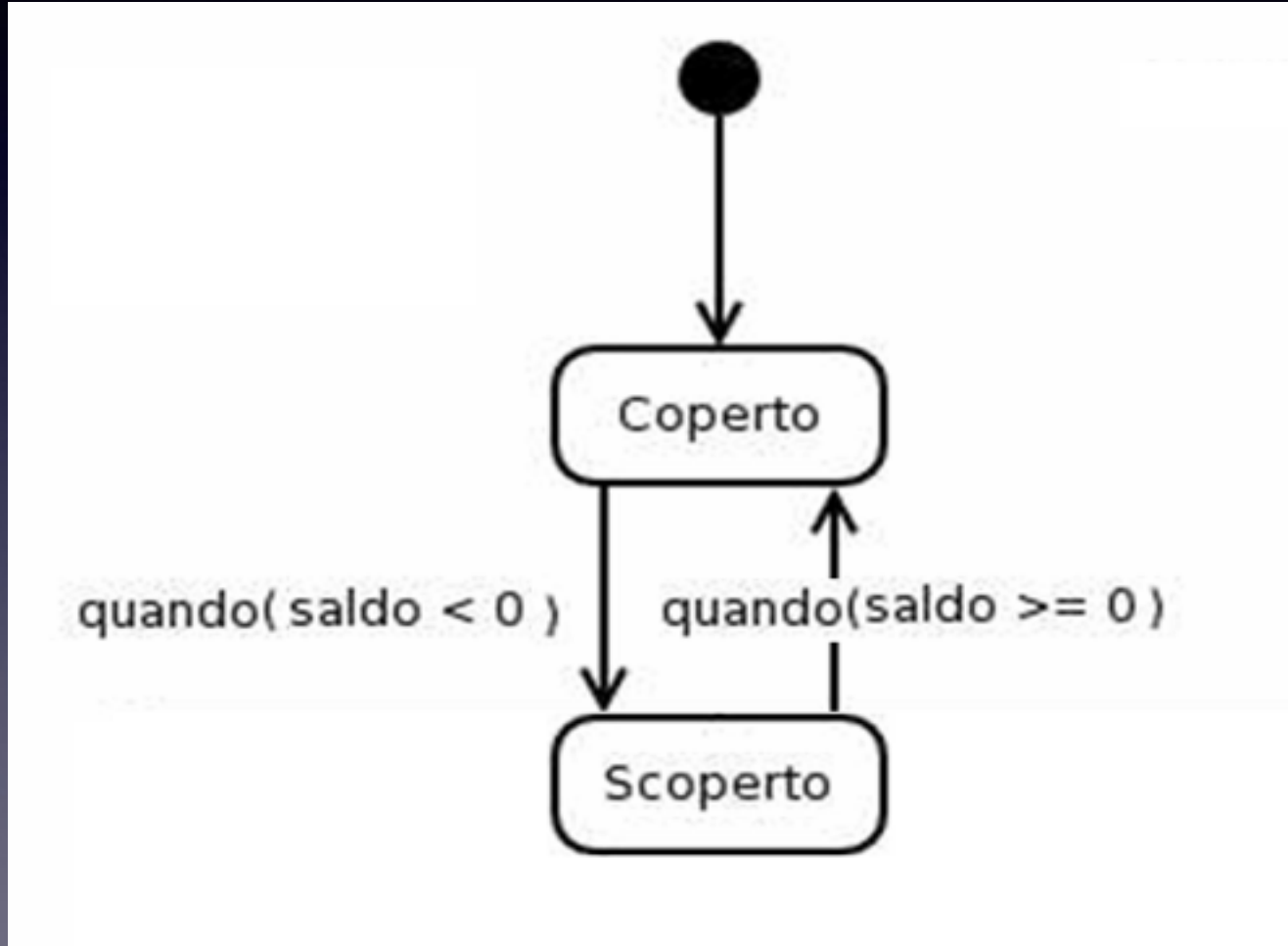


# tipi di evento

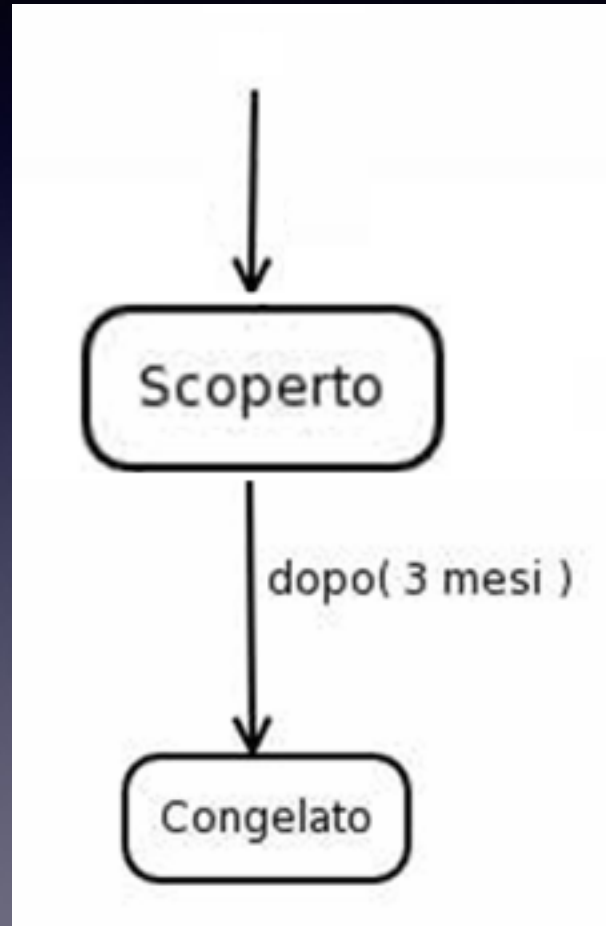
**Table 7-1: Kinds of Events**

<i>Event Type</i>	<i>Description</i>	<i>Syntax</i>
<b>call event</b>	Receipt of an explicit synchronous call request by an object	<b>op</b> (a:T)
<b>change event</b>	A change in value of a Boolean expression	<b>when</b> (exp)
<b>signal event</b>	Receipt of an explicit, named, asynchronous communication among objects	<b>sname</b> (a:T)
<b>time event</b>	The arrival of an absolute time or the passage of a relative amount of time	<b>after</b> (time)

# un evento di variazione



# un evento del tempo



# tipi di transizione

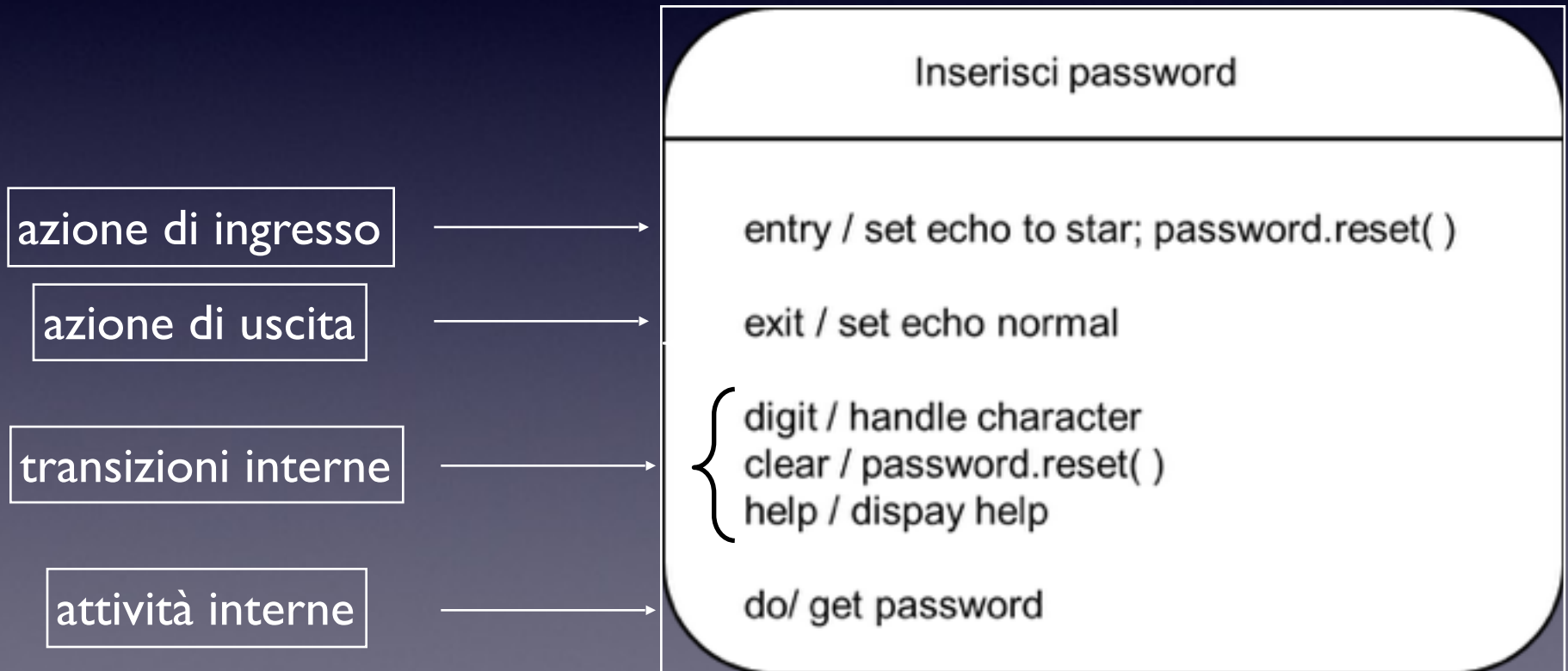
<i>Transition Kind</i>	<i>Description</i>	<i>Syntax</i>
entry transition	The specification of an <b>entry activity</b> that is executed when a state is entered	entry/ <b>activity</b>
exit transition	The specification of an <b>exit activity</b> that is executed when a state is exited	exit/ <b>activity</b>
external <b>transition</b>	A response to an <b>event</b> that causes a change of <b>state</b> or a self-transition, together with a specified <b>effect</b> . It may also cause the execution of exit and/ or entry activities for states that are exited or entered.	<b>e(a:T)[guard]/activity</b>
<b>internal transition</b>	A response to an event that causes the execution of an effect but does not cause a change of state or execution of exit or entry activities	<b>e(a:T)[guard]/activity</b>

# transizioni e attività interne

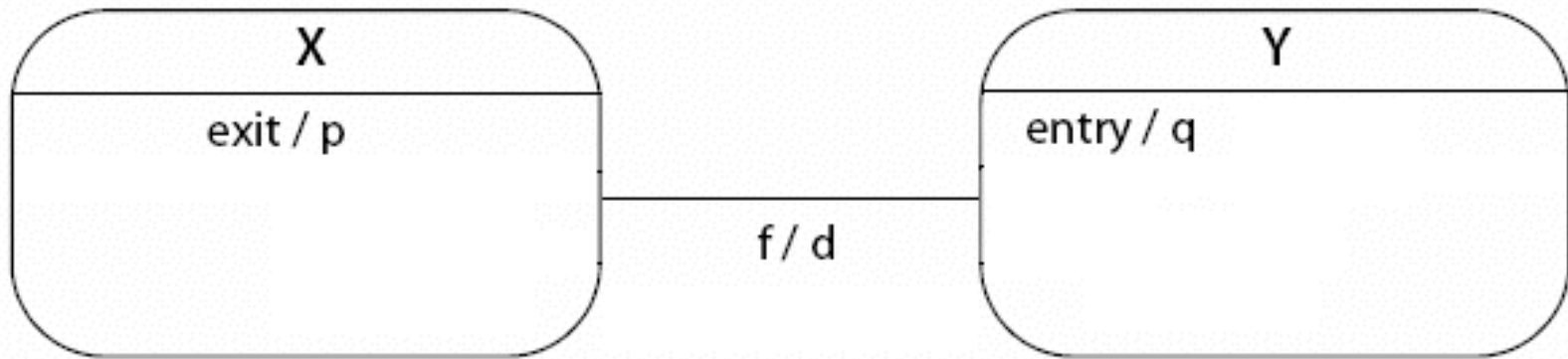
- Transizione interna: risposta a un evento che causa solo l'esecuzione di azioni
  - Azione di entrata: eseguita all'ingresso in uno stato
  - Azione di uscita: eseguita all'uscita di uno stato
  - Transizione interna: risposta ad un evento
- Attività interna (do-activity): eseguita in modo continuato (senza necessità di un evento scatenante), consuma del tempo e può essere interrotta

# un esempio

- Una barra separa il nome dello stato dalla regione



# un altro esempio



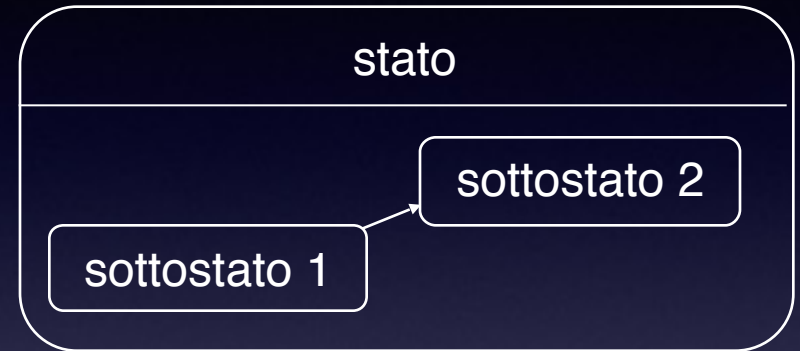
effective result: f / p; d; q





# stati compositi

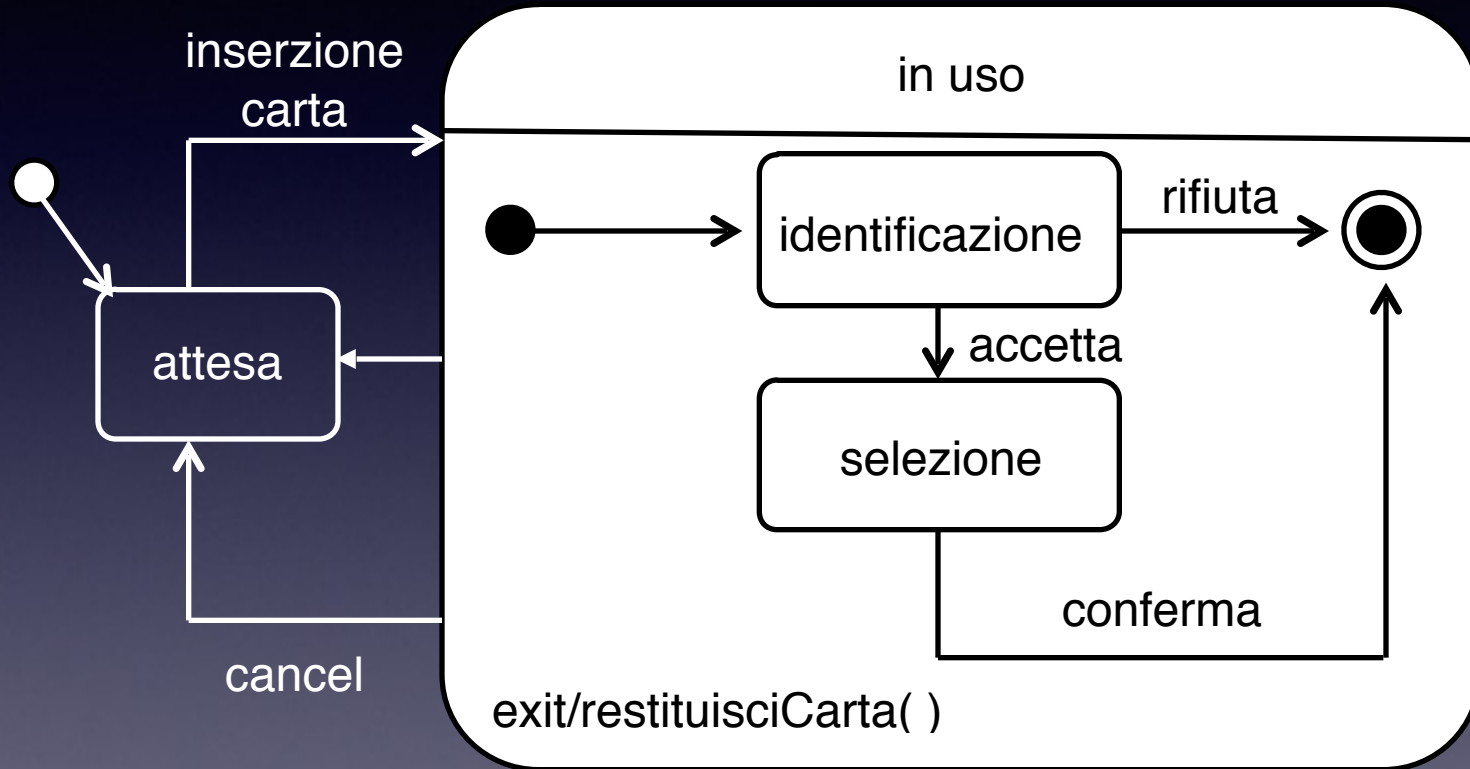
- sequenziale: uno stato attivo ad ogni istante



- parallelo: più sotto-stati attivi ad ogni istante

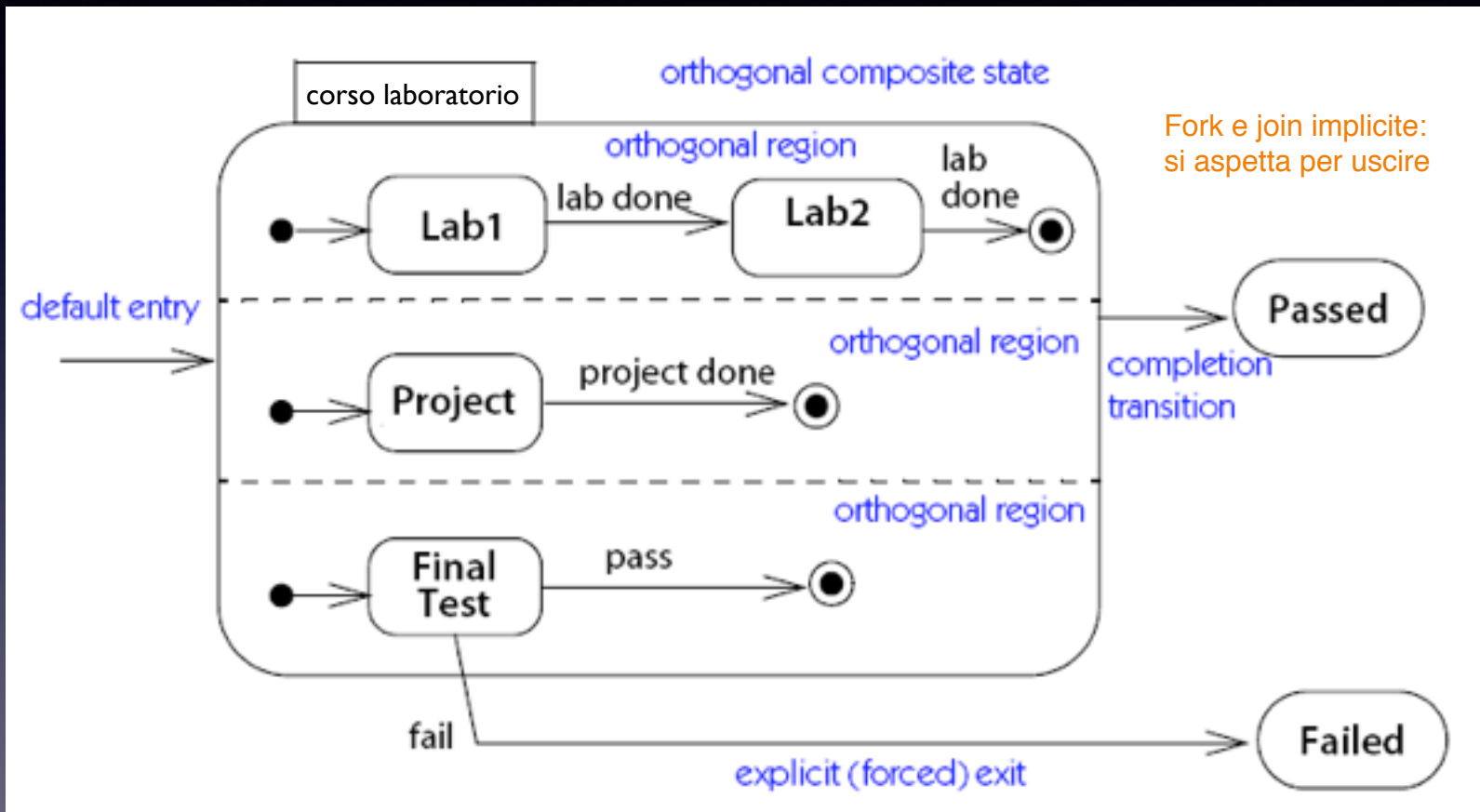


# esempio sequenziale



- Ogni regione può avere uno stato iniziale, e ogni transizione al limite esterno è diretta verso quello

# esempio parallelo



- Lo stesso per gli stati finali. Una transizione a uno di quelli fa scattare una transizione di completamento dello stato composito

# altri (pseudo-)stati

- Giunzione



- Storia



- Decisione



- Iniziale



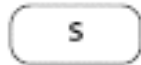
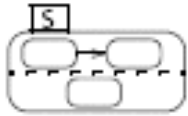
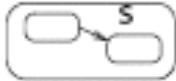






- Fork/Join



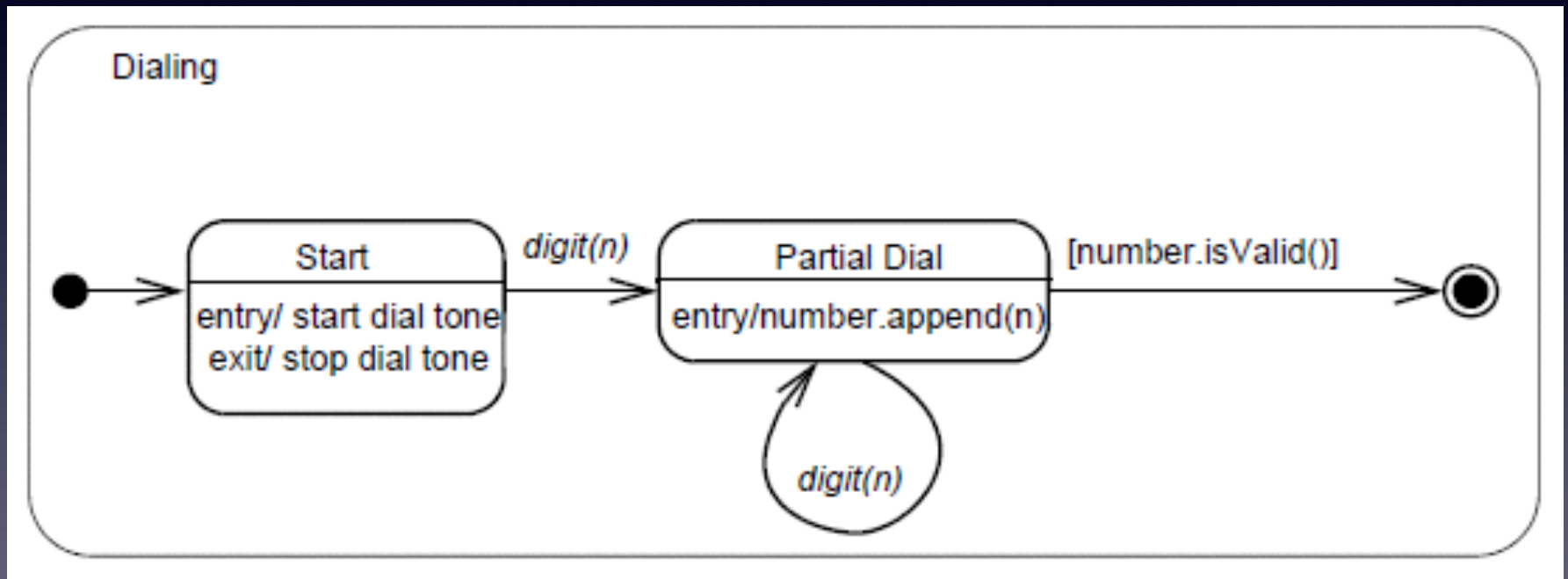
- Finale



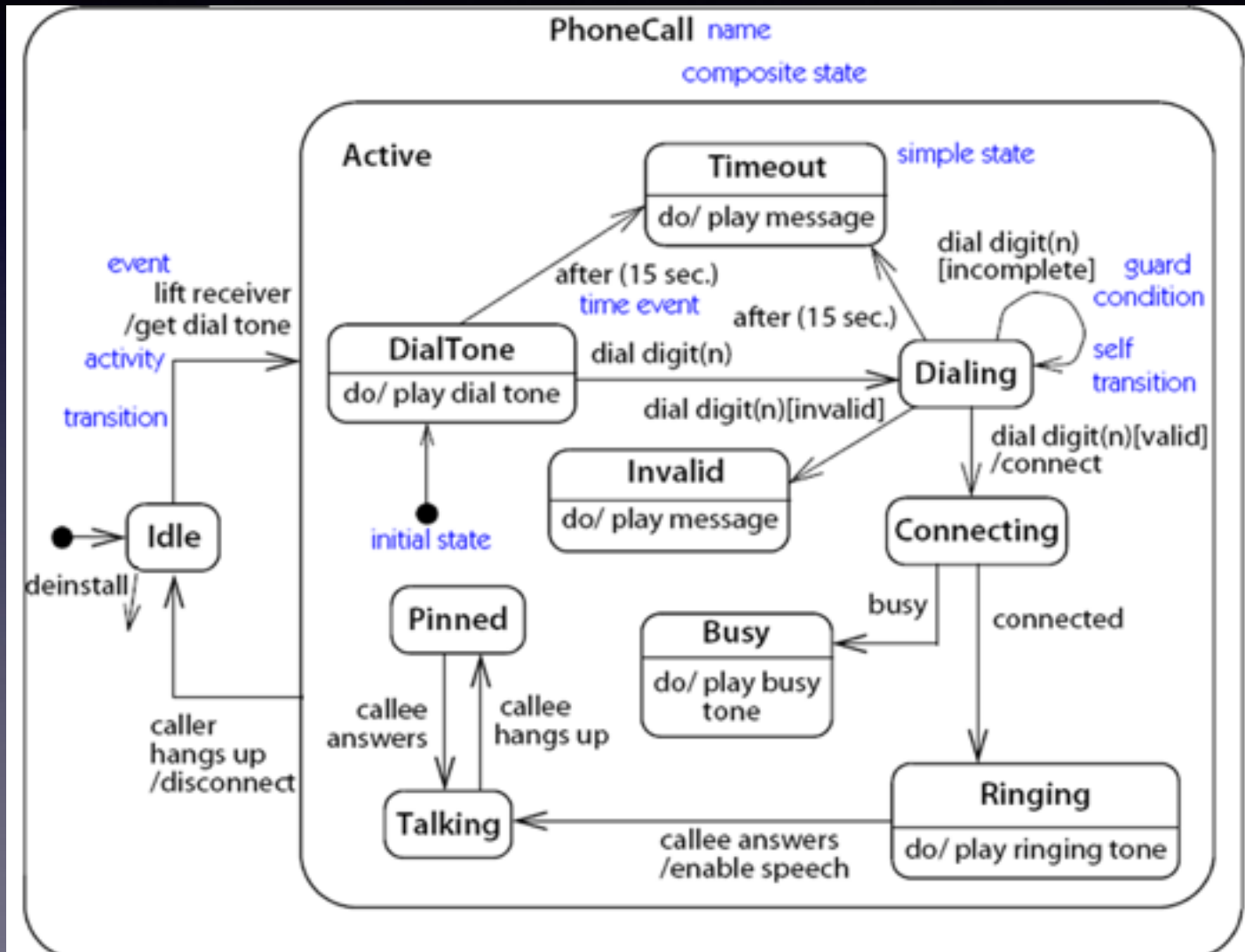
# tipi di stato

<i>State Kind</i>	<i>Description</i>	<i>Notation</i>
<b>simple state</b>	A <b>state</b> with no substructure	
<b>orthogonal state</b>	A state that is divided into two or more regions. One <b>direct substate</b> from each <b>region</b> is concurrently active when the composite state is active.	
<b>nonorthogonal state</b>	A composite state that contains one or more direct substates, exactly one of which is active at one time when the composite state is active	
<b>initial state</b>	A <b>pseudostate</b> that indicates the starting state when the enclosing state is invoked	
<b>final state</b>	A special state whose activation indicates the enclosing state has completed activity	
<b>terminate</b>	A special state whose activation terminates execution of the object owning the state machine	
<b>junction</b>	A pseudostate that chains <b>transition segments</b> into a single <b>run-to-completion</b> transition	
<b>choice</b>	A pseudostate that performs a dynamic branch within a single run-to-completion transition	
<b>history state</b>	A pseudostate whose activation restores the previously active state within a <b>composite state</b>	

# una chiamata migliorata

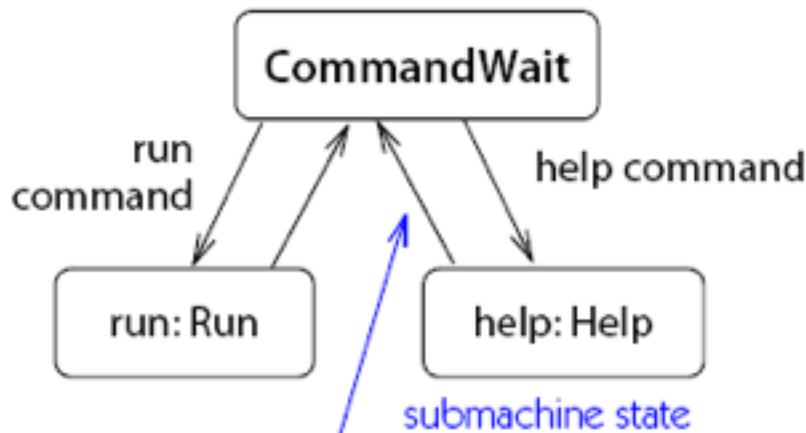


# un altro esempio



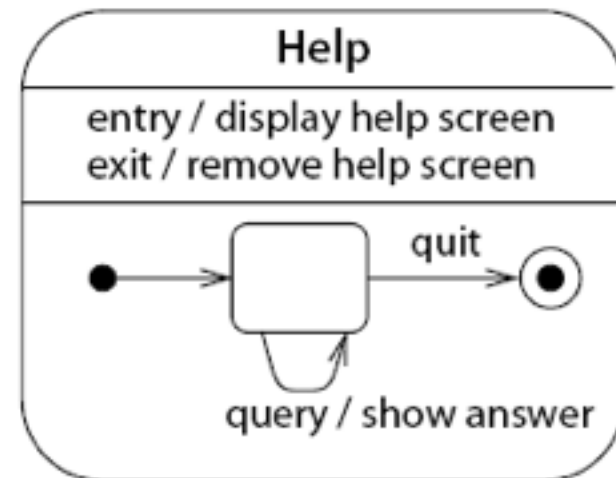
# sotto-macchine

- L'invocazione di una macchina annidata è rappresentato da nome-stato: nome-macchina



Completion transition fires when submachine completes.

submachine definition

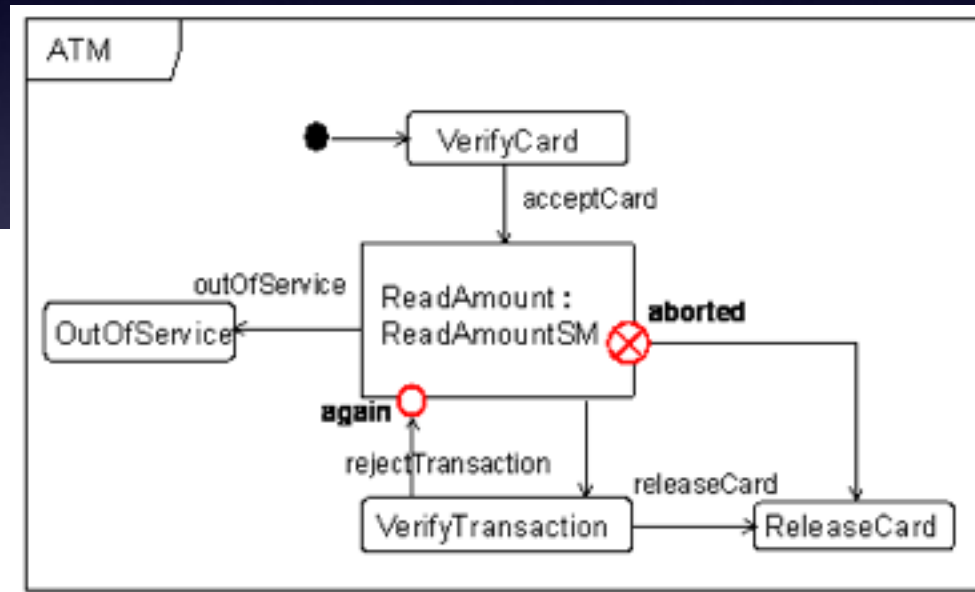
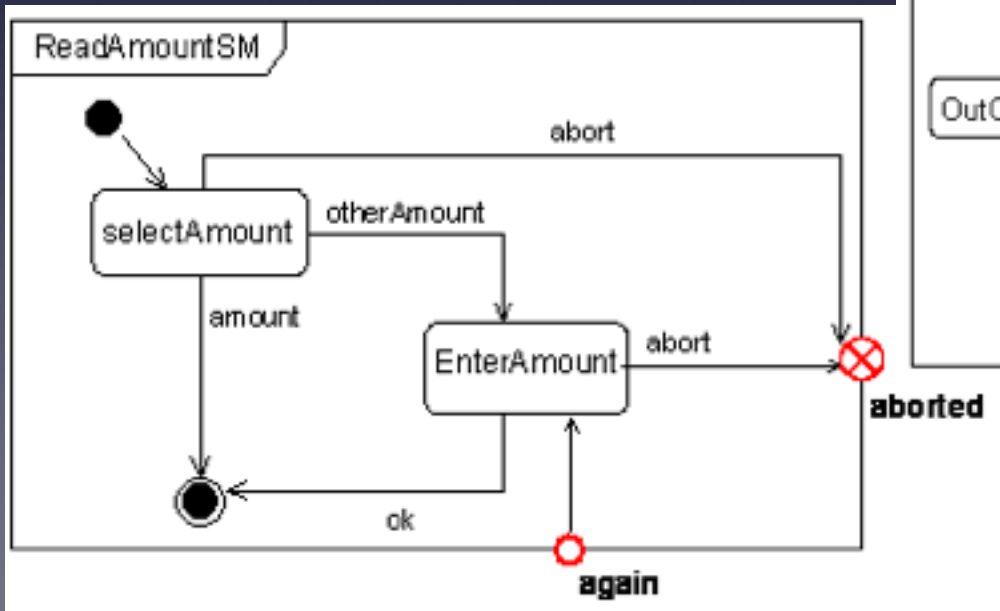


This submachine can be used many times.

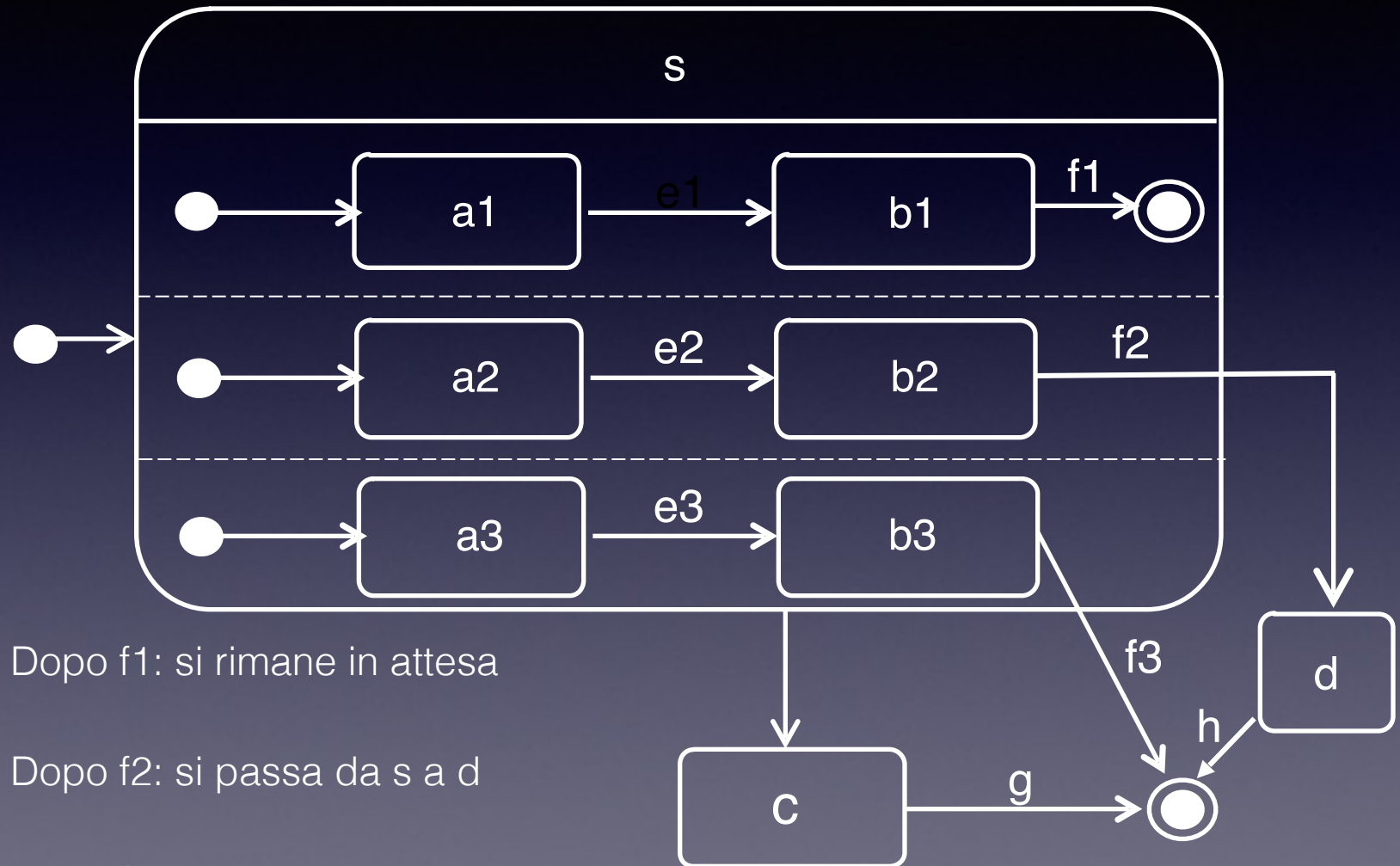


# punti di ingresso e uscita

- Una sotto-macchina può avere punti di ingresso e uscita che connettono direttamente a stati interni



# terminazione e stati composti



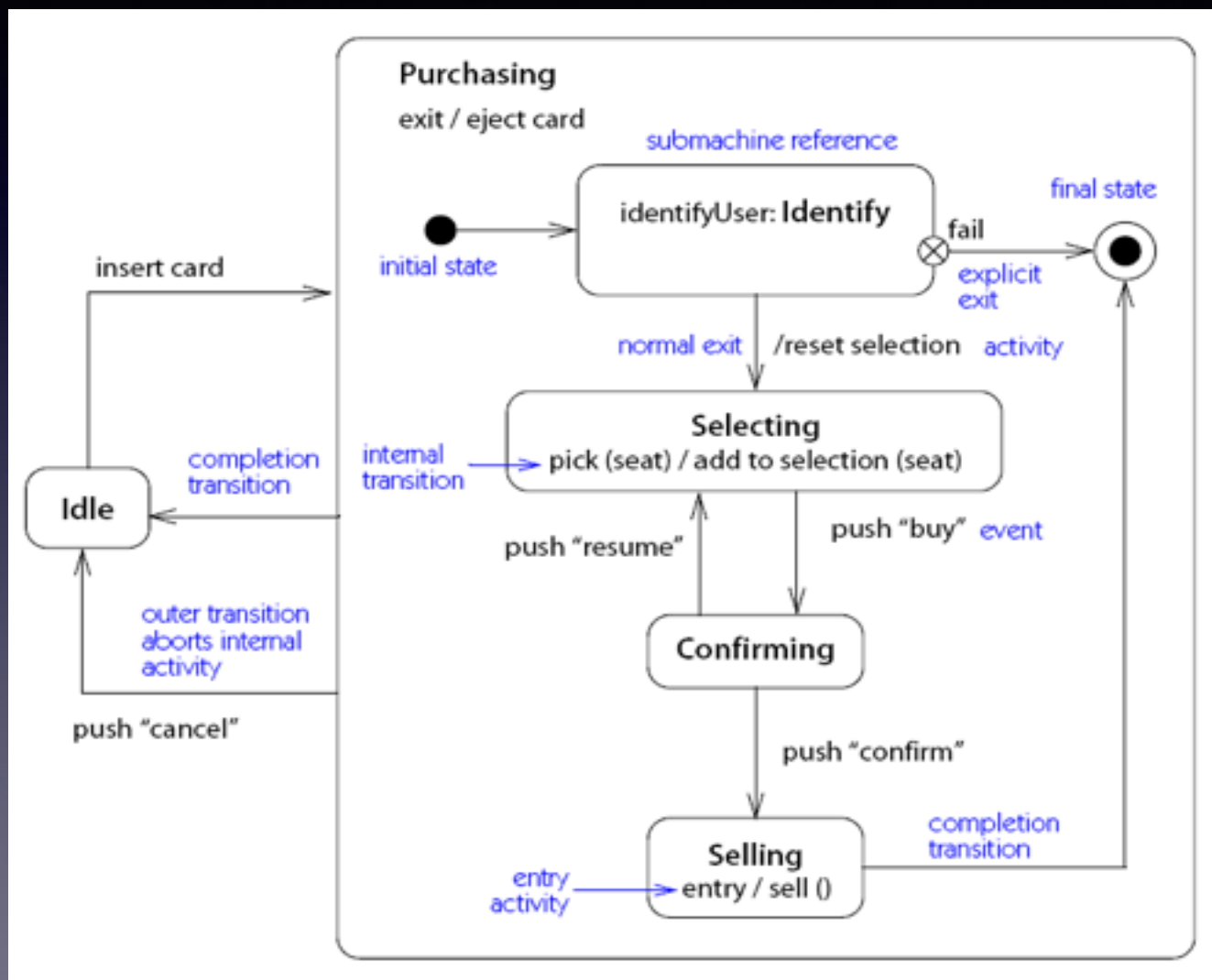
- Dopo  $f_1$ : si rimane in attesa
- Dopo  $f_2$ : si passa da  $s$  a  $d$
- Dopo  $f_3$ : si esce e si termina

- Uscendo da  $s$  si abbandonano tutti i sotto-stati

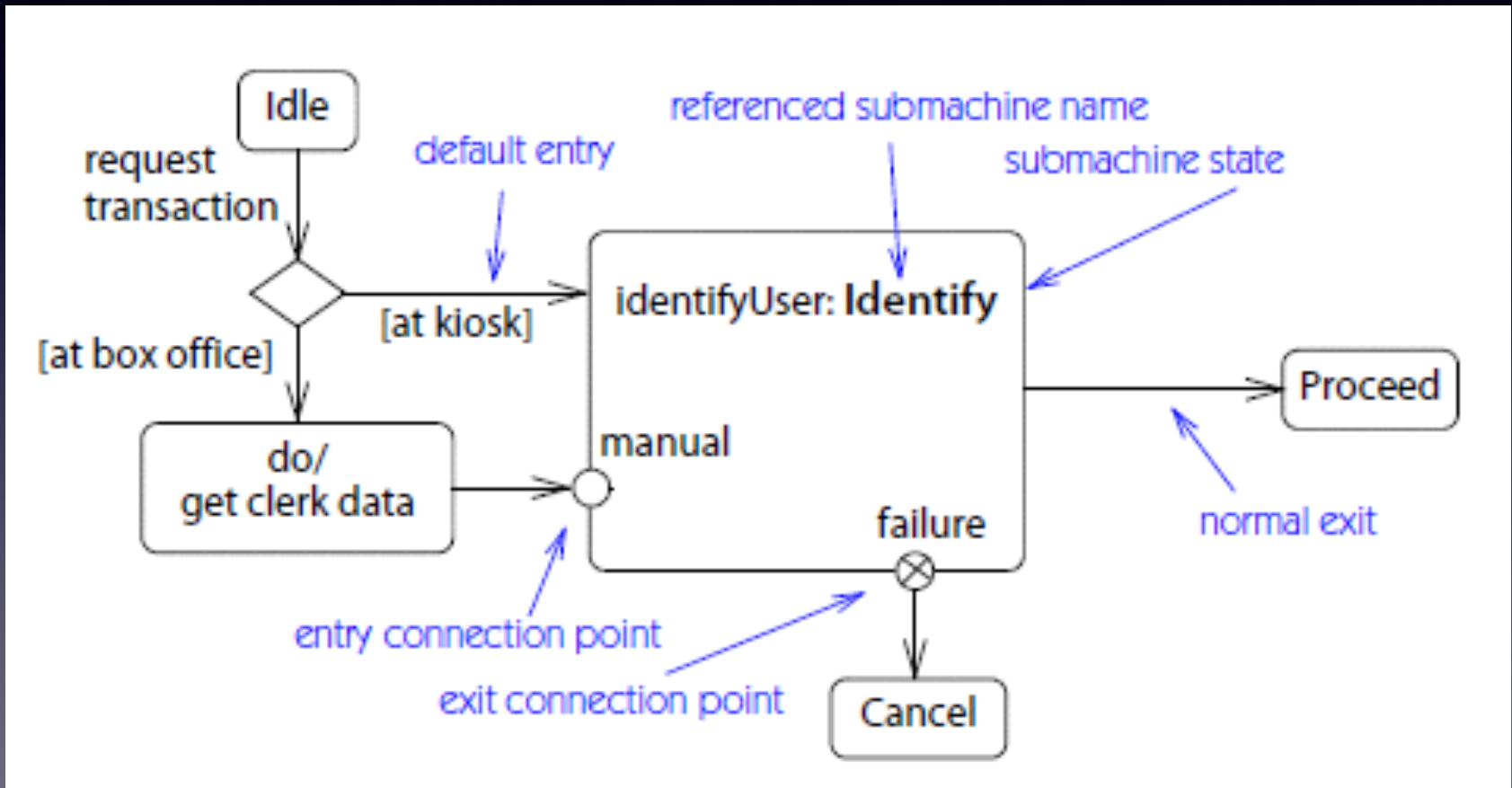
# transizioni di completamento

- Senza evento, scattano al raggiungimento
  - della terminazione di un'attività composita, ovvero, al raggiungimento
    - dello stato finale in un stato composito non-ortogonale
    - degli stati finali di tutte le regioni ortogonali di un stato composito
  - del pseudo-stato di terminazione di una regione di uno stato composito ortogonale
  - di un exit point
- Alla terminazione di entry e/o di do activity (la exit activity viene eseguita quando scatta la transizione di completamento)
- Di uno pseudo-stato giunzione
- Hanno priorità sugli eventi normali

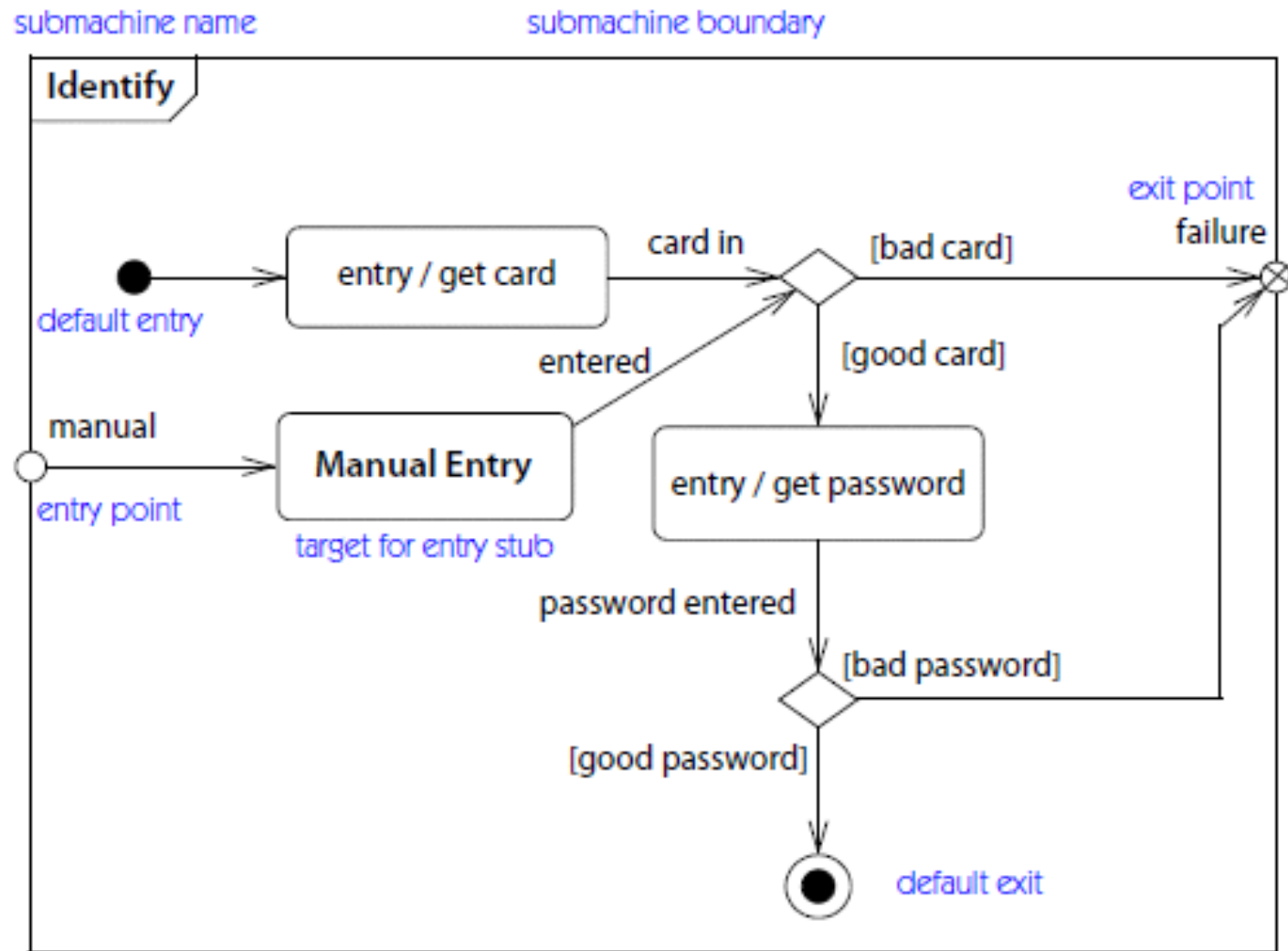
# esempio: acquisto biglietti



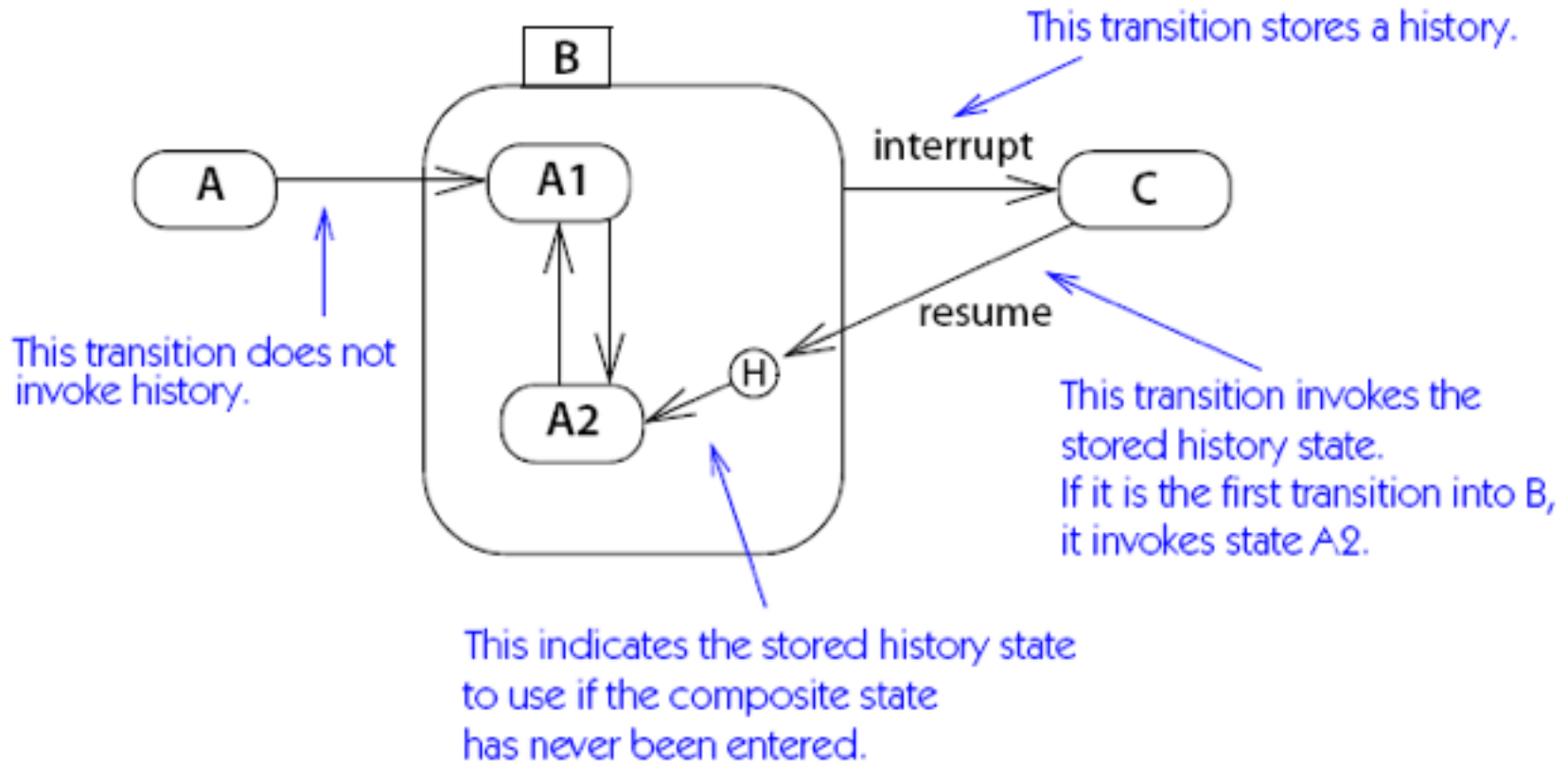
# variante: clienti con account



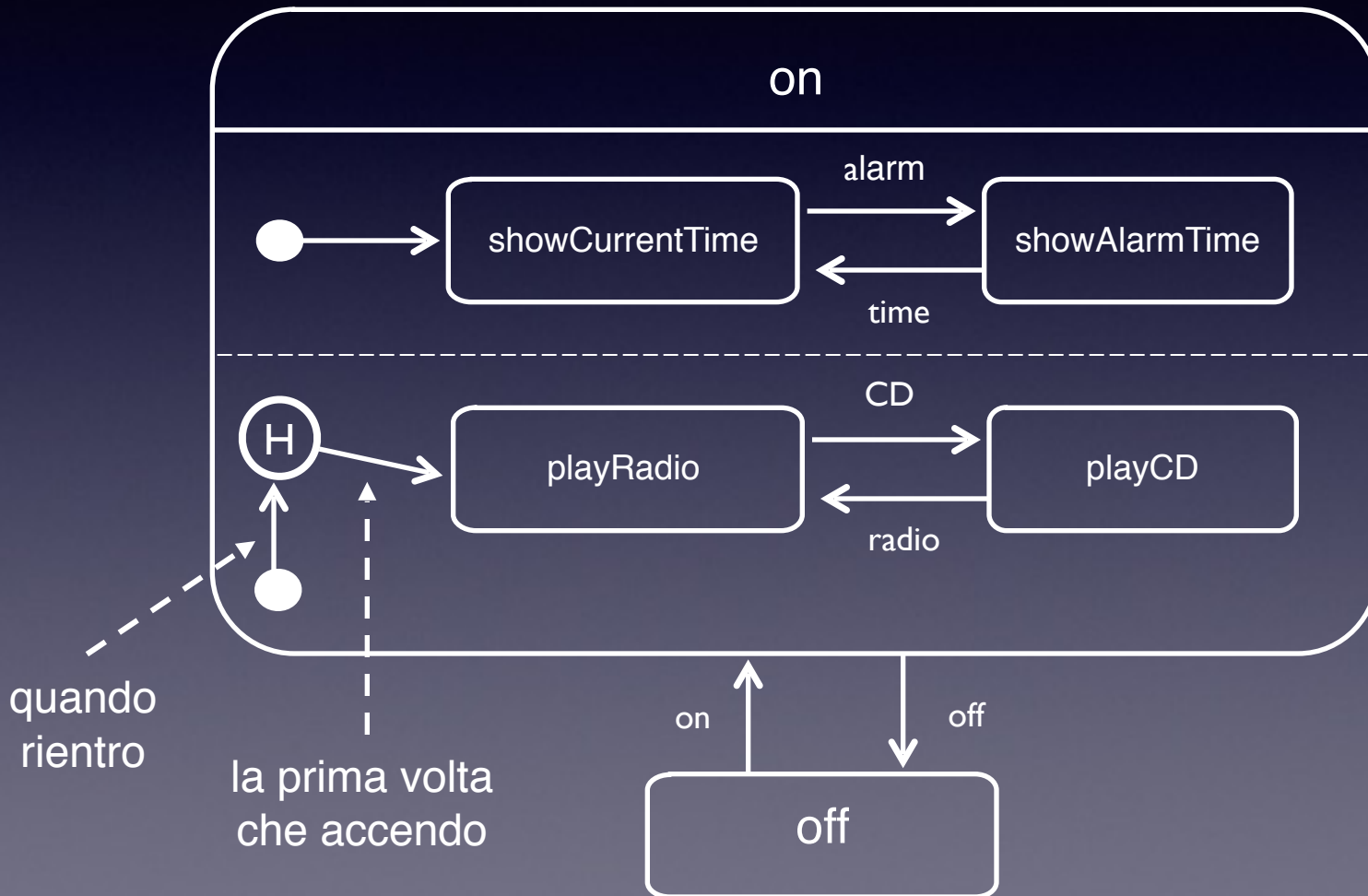
# e sotto-stati



# history state

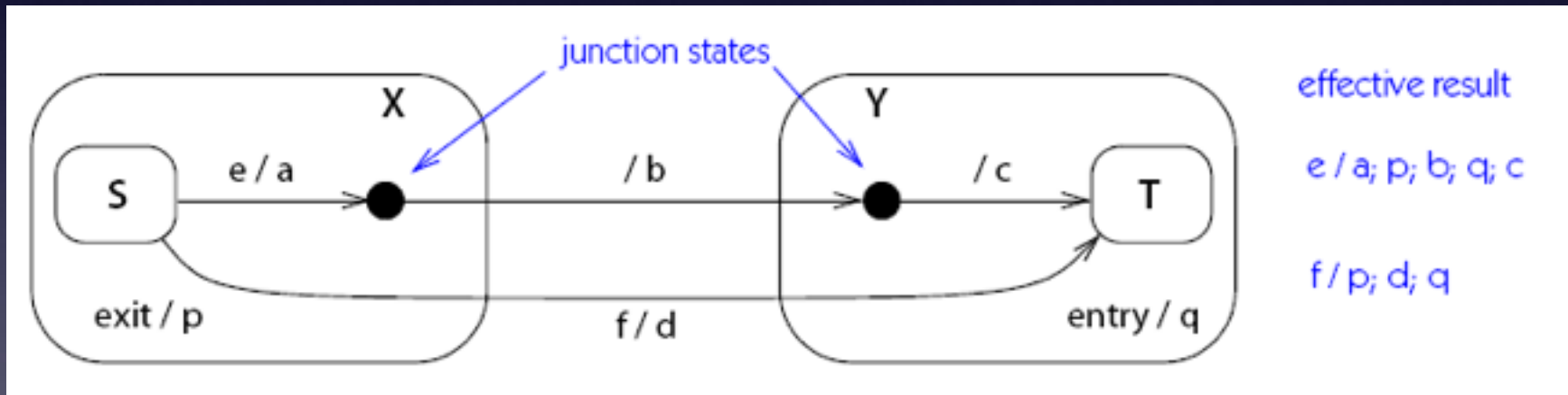


# un esempio di storia



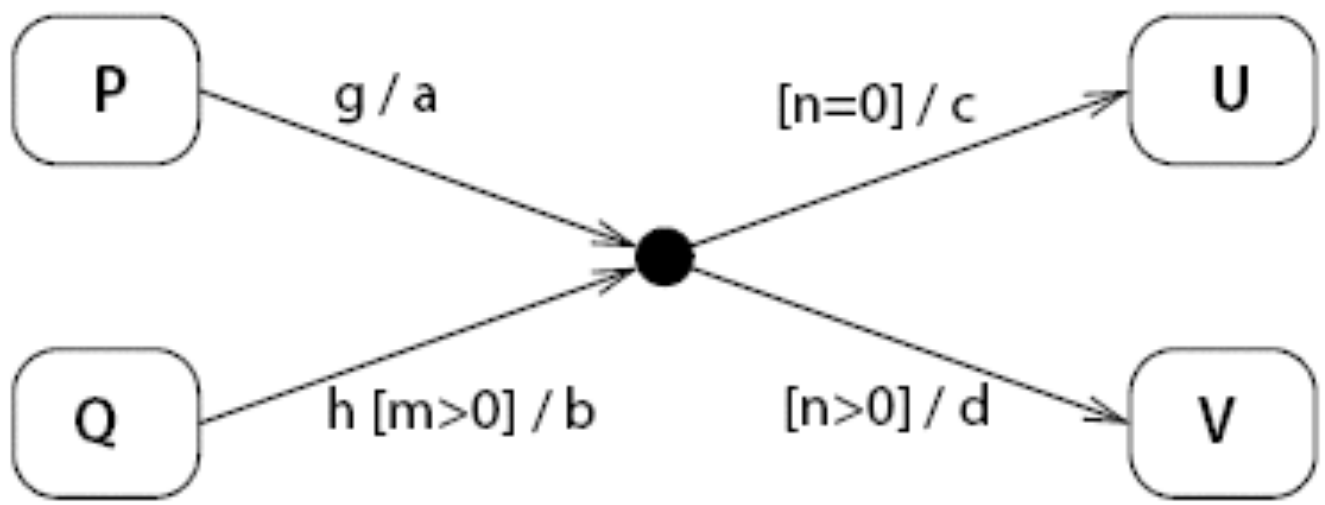


# uso dello stato giunzione



- La scelta è dinamica, la giunzione statica

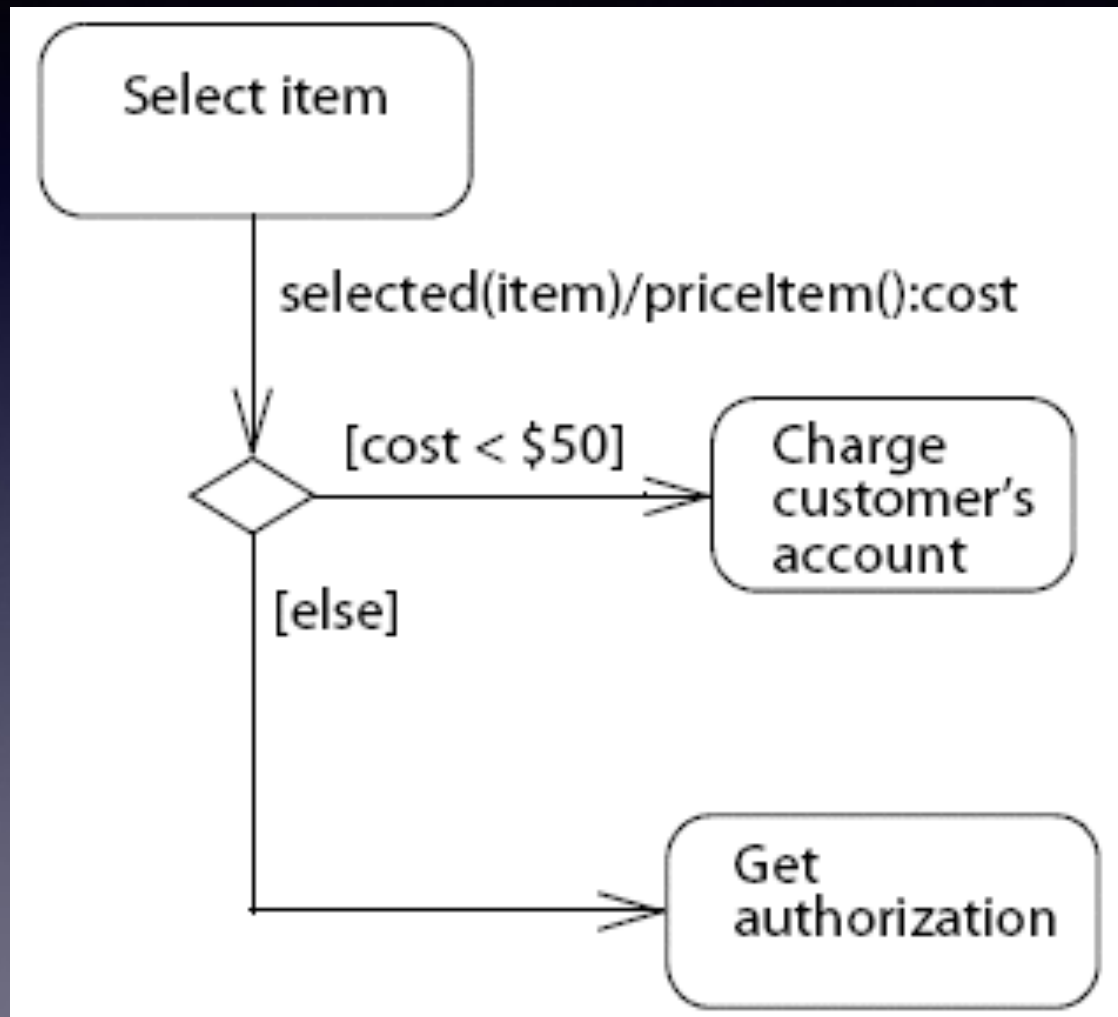
# esempio di giunzione



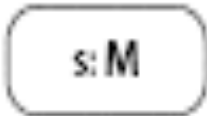


Equivalent transitions:

- $P g [n=0] / a; c U$
- $P g [n>0] / a; d V$
- $Q h [m>0 \text{ and } n=0] / b; c U$
- $Q h [m>0 \text{ and } n>0] / b; d V$

# esempio di scelta



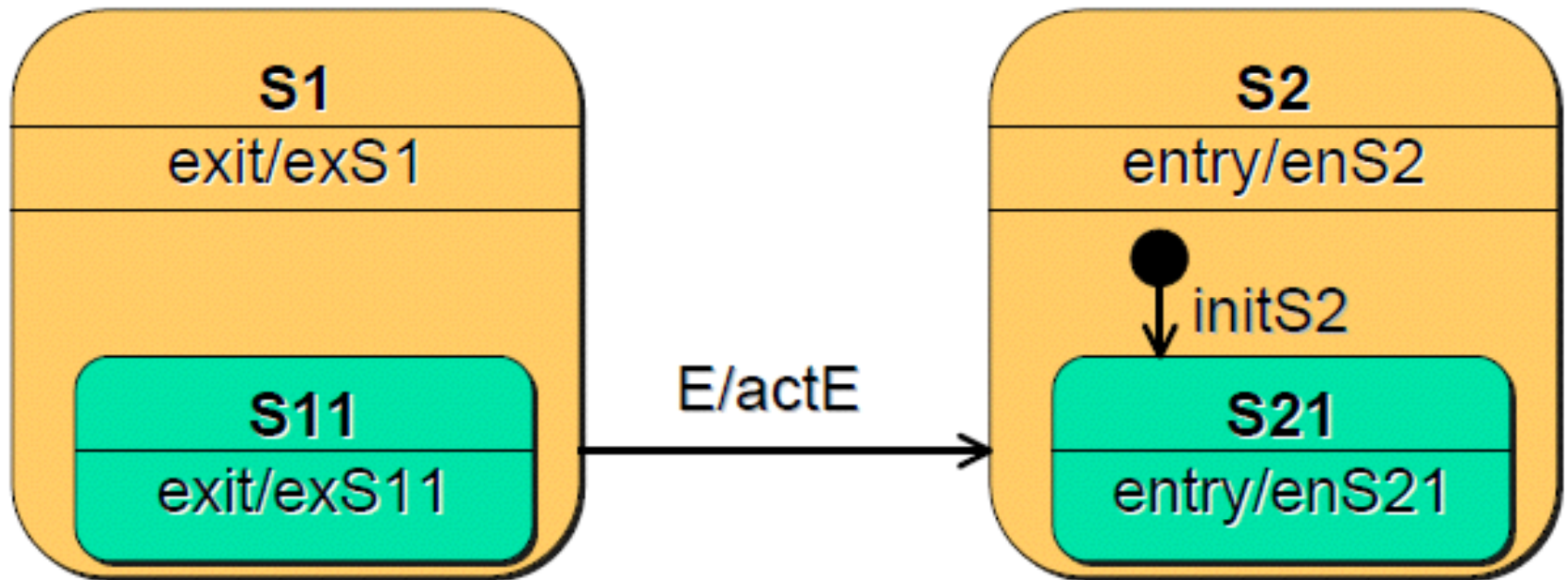
# sulle sotto-macchine

<i>State Kind</i>	<i>Description</i>	<i>Notation</i>
<b>submachine state</b>	A state that references a state machine definition, which conceptually replaces the submachine state	
<b>entry point</b>	A externally visible pseudostate within a state machine that identifies an internal state as a target	
<b>exit point</b>	A externally visible pseudostate within a state machine that identifies an internal state as a source	

# sullo stato di terminazione

- “Entering a *terminate* pseudo-state implies that the execution of this state machine by means of its context object is terminated. The state machine does not exit any states nor does it perform any exit actions other than those associated with the transition leading to the terminate pseudo-state. Entering a terminate pseudo-state is equivalent to invoking a DestroyObjectAction.”

# sull'ordine delle azioni



**Actions execution sequence:**

exS11  $\Rightarrow$  exS1  $\Rightarrow$  actE  $\Rightarrow$  enS2  $\Rightarrow$  initS2  $\Rightarrow$  enS21