

Indice

10 – Diagrammi di attività.....	2
10.1 – Nodi azione.....	3
10.2 – Nodi controllo	4
10.3 – Nodi oggetto	6
10.4 – Esempio di attività.....	8
10.5 – Regioni interrompibili.....	9
10.6 – Conclusioni	10

10 – Diagrammi di attività

Il **diagramma di attività** modella un'attività relativa ad un qualsiasi oggetto, ad esempio: classi, casi d'uso, interfacce, componenti e operazioni di classe.

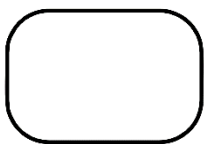
Solitamente, i diagrammi di attività si **utilizzano** per:

1. Modellare il flusso di un caso d'uso (analisi);
2. Modellare il funzionamento di un'operazione di classe (progettazione);
3. Modellare un algoritmo (progettazione).

Le **attività** sono **specificate** da vari nodi:

- Nodi **azione**, che specificano unità di comportamento;
- Nodi **oggetto**, che specificano gli oggetti utilizzati come input e output di azioni;
- Nodi **controllo**, che specificano il flusso dell'attività tramite:
 - Nodi iniziali e finali;
 - Nodi decisione e fusione;
 - Nodi fork/join;
 - Nodi finali di flusso;

Qui di seguito un'introduzione visiva dei componenti:



Action node



Object node



————— *Control nodes* —————

10.1 – Nodi: azione

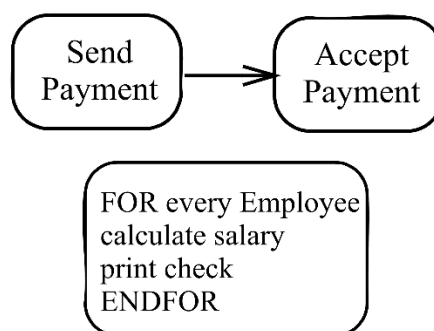
Un'azione può invocare un'attività, un **comportamento** o un'operazione.

Come gli altri elementi di UML, anche le azioni accettano livelli di dettaglio e linguaggi differenti.

Al contrario dei messaggi nei diagrammi di interazione, le azioni non costringono il modellatore a definire tutte le entità in gioco.

Le transazioni (freccie) tra azioni possono avere una guardia.

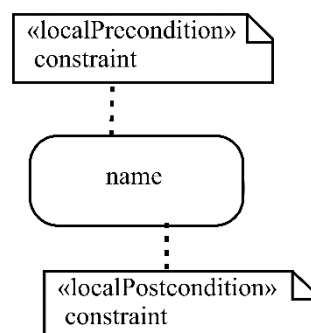
Un **esempio** di nodo d'azione:



Per capire la semantica dei diagrammi di attività, bisogna immaginare delle entità, dette **token**, che viaggiano lungo il diagramma. Il **flusso dei token definisce il flusso dell'attività**.

I token possono rimanere fermi in un nodo azione/oggetto in attesa che si avveri una condizione su una freccia, oppure una preconditione o postcondizione su un nodo.

Una preconditione, o una postcondizione, è una condizione espressa in qualunque modo che deve essere soddisfatta per far iniziare o terminare l'azione (consente ad un token di entrare o uscire). Per esempio:



Il **movimento** di un token è **atomico**.

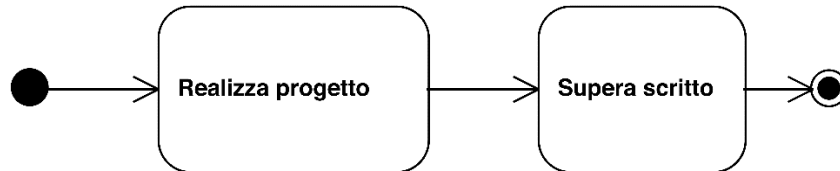
Un **nodo azione viene eseguito quando sono presenti token su tutti gli archi in entrata**, e tutte le preconditioni sono soddisfatte.

Al termine di un'azione, sono generati **control token** su tutti gli archi di uscita.

10.2 – Nodi: controllo

Nodi iniziali e finali

Come si vede dalla seguente immagine, il disco nero marca l'**inizio** dell'attività, quindi genera un token.



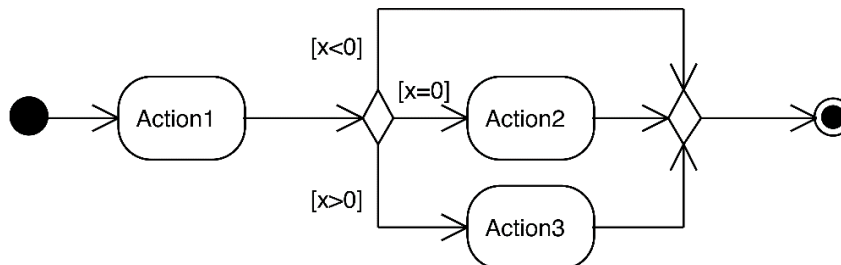
Quando un token raggiunge un disco nero bordato, nodo finale, l'attività ha **terminato**.

Possono comparire in qualunque numero all'interno di un'attività (ogni nodo iniziale fa partire un flusso di esecuzione, il primo nodo finale raggiunto ferma tutti i flussi).

Nodi decisione e fusione

I nodi **decisione** hanno un input e vari output **mutuamente esclusivi**: copiano i token in entrata su uno degli output (nell'immagine è il rombo a sinistra).

Invece, i nodi **fusione** hanno vari input e un solo output, sul quale vengono indirizzati tutti i token in ingresso (nell'immagine è il rombo a destra).

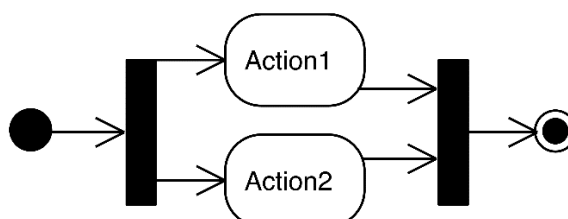


Nodi fork/join

I nodi **fork** hanno un ingresso e varie uscite: i token in ingresso sono duplicati su tutte le uscite.

I nodi **join** hanno vari ingressi e una sola uscita: quando sono presenti token su tutti gli ingressi, viene prodotto almeno un token in uscita.

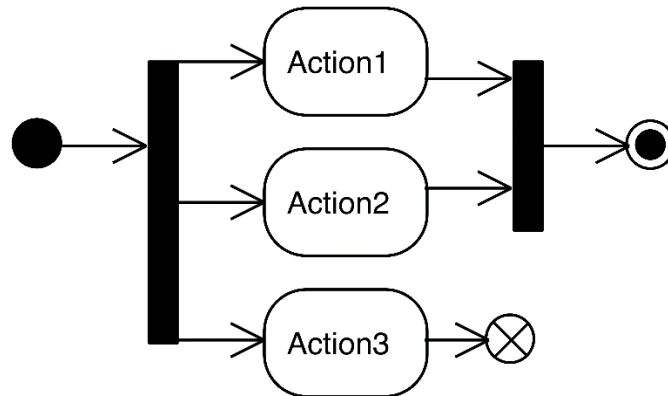
I nodi **fork** **dividono un'esecuzione in più flussi concorrenti**, i nodi **join** **sincronizzano e riuniscono i flussi**.



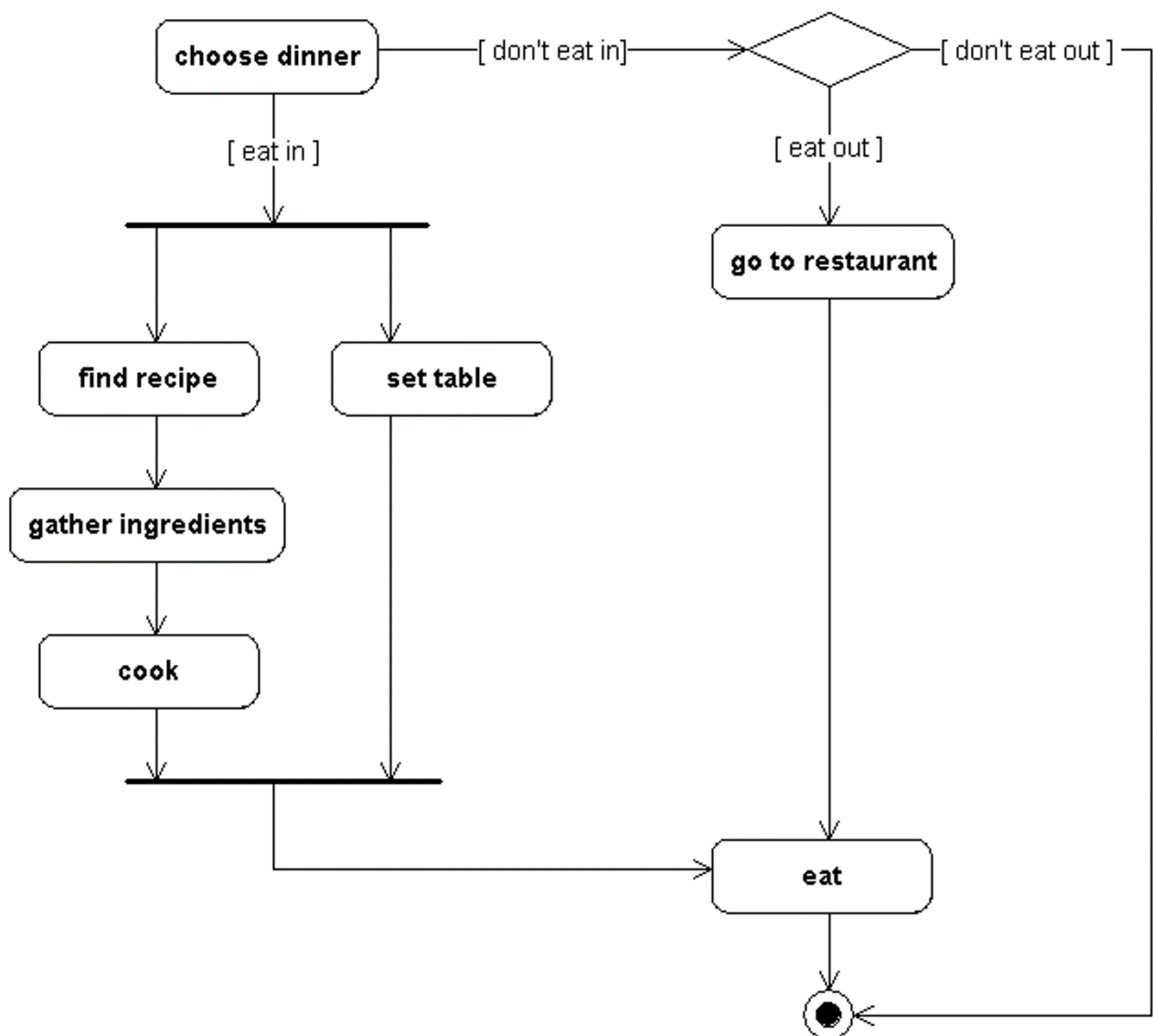
Nodi finali di flusso

Quando raggiunti da un token, causano la **terminazione** solo del flusso che li ha toccati.

Il raggiungimento di un nodo finale di attività causa comunque la terminazione di tutti i flussi.



Esempio completo

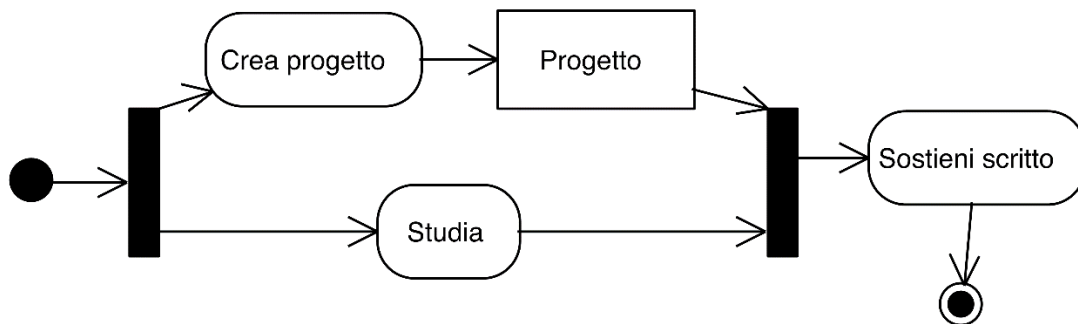


10.3 – Nodi: oggetto

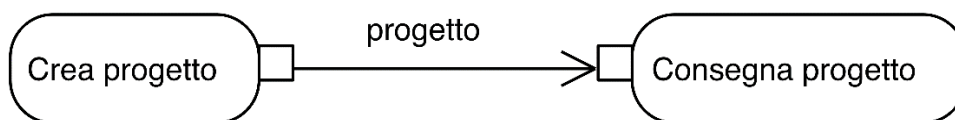
I **nodi oggetto** servono per modellare gli oggetti in input e output delle azioni.

I token di uscita da questi nodi sono **object token**, e sono diversi dai **control token** prodotti dai nodi azione poiché rappresentano veri e propri oggetti (gli object token).

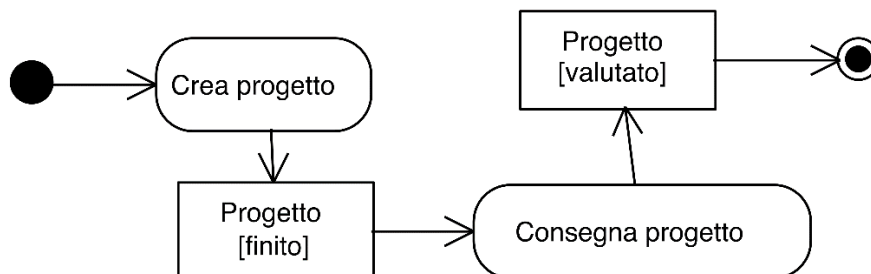
Gli archi in entrata e uscita dai nodi oggetto sono **object flow** anziché **control flow**, e ci sono regole che limitano il loro uso (esempio gli archi che entrano ed escono dai nodi decisione e fusione devono essere o tutti object o tutti control).



I **pin** si agganciano ai nodi azione per definire un input oppure un output di quell'azione. Questa notazione è **equivalente** a quella di un nodo oggetto tra i due nodi azione. I pin aiutano a mostrare i parametri e valori di ritorno di un'azione.

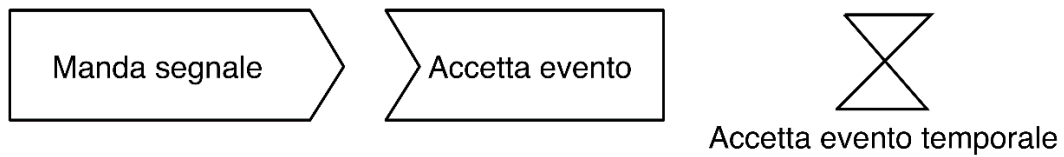


Spesso risulta utile aggiungere lo **stato di un oggetto** per mostrare l'evoluzione durante l'attività. Gli stati devono essere coerenti con la macchina a stati associata all'oggetto. Questo è l'**anello di congiunzione tra diagrammi di attività e diagrammi di stato!**



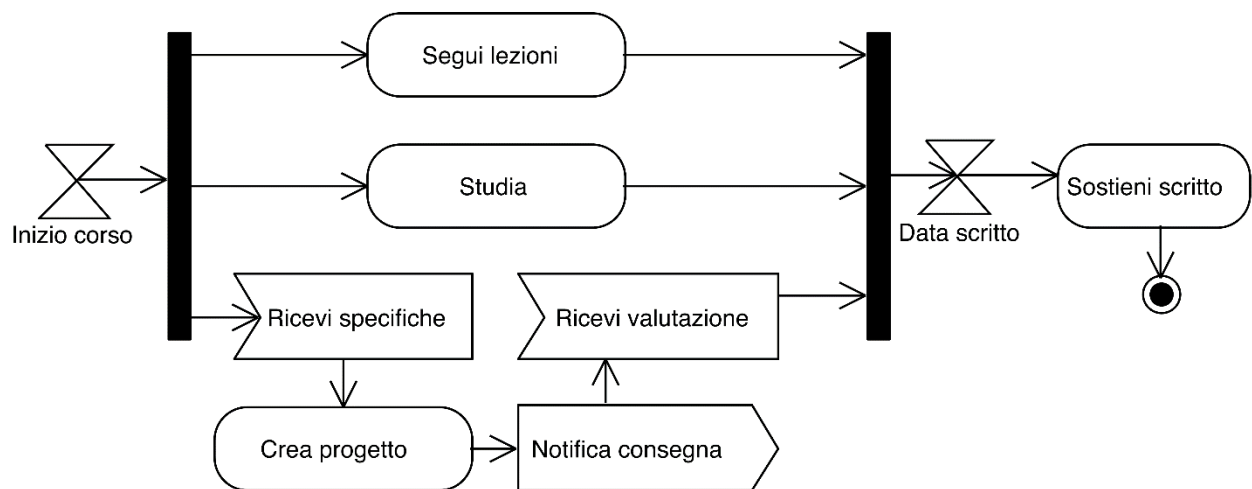
Esiste un collegamento stretto da segnali ed eventi.

Esistono alcuni nodi azione specializzati che gestiscono l'invio e la ricezione di segnali. L'**invio dei segnali** è **asincrono** e **non blocca l'attività**.



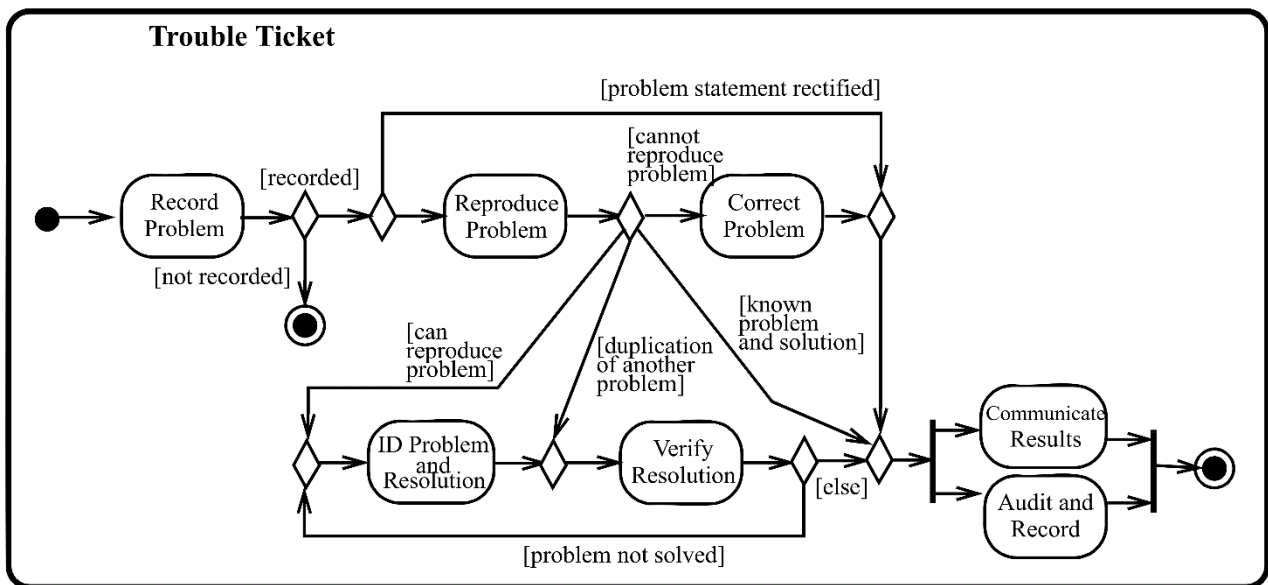
I nodi ricezione sono attivi quando hanno token su tutti gli archi in entrata (se ne hanno) oppure durante l'intera vita dell'attività (se non ne hanno). Inoltre, generano token alla ricezione.

La ricezione di eventi temporali funziona nello stesso modo, i token sono generati in base ad un'espressione temporale.

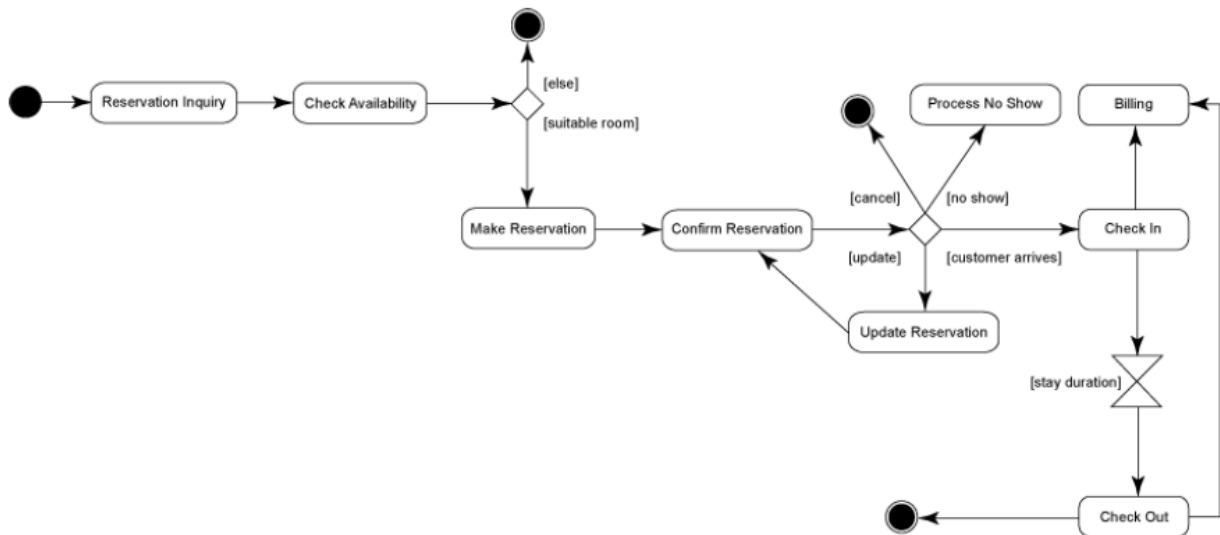


10.4 – Esempio di attività

Un'attività è costituita da un flusso di azioni che possono essere considerate come mattoni. Inoltre, un'azione può invocare un'altra attività.



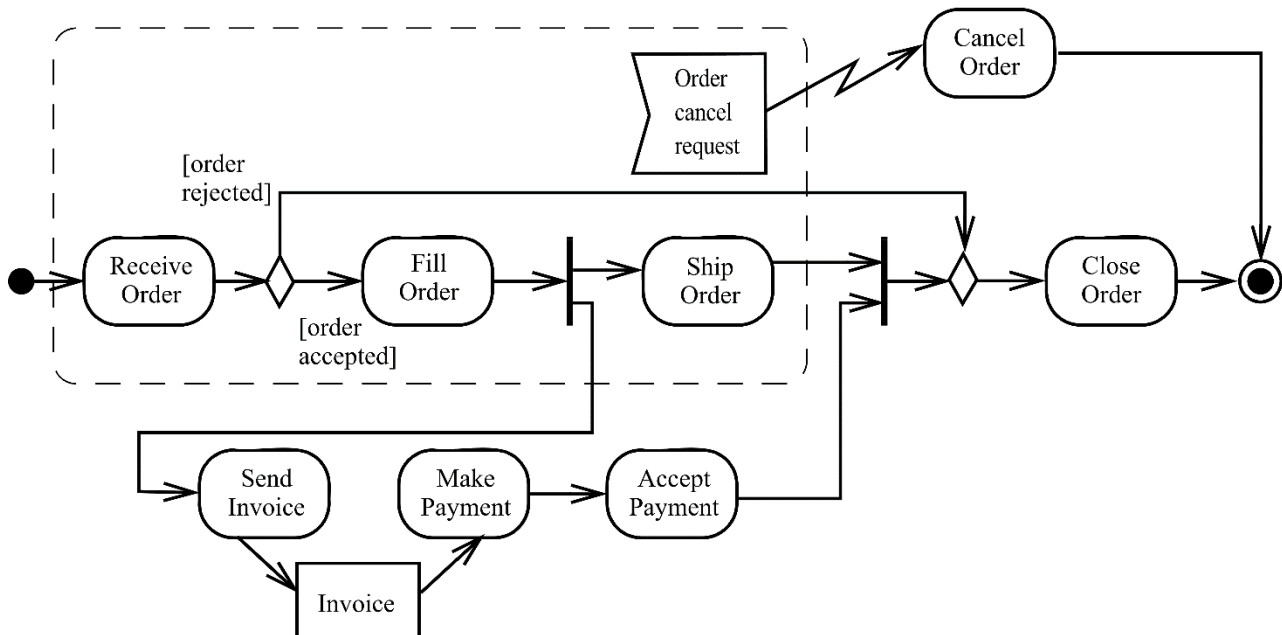
Un altro esempio:



10.5 – Regioni interrompibili

Si possono definire delle **regioni interrompibili** che si usano per **specificare una reazione che può avvenire in qualunque momento** e comporta l'interruzione dell'attività. Degli esempi sono: le eccezioni, i segnali, le situazioni di errore dall'esterno.

La **notazione** impiegata è quella di un'attività con i bordi tratteggiati. Uno o più archi di interrupt (a zig-zag) partono da nodi interni e puntano verso nodi esterni (come da immagine).



L'**interrupt** è **generato quando un arco di interrupt è attraversato da un token**: tutti gli altri token e comportamenti nella regione sono terminati.

La ricezione di eventi all'interno della regione funziona **solo** se ci sono token al suo interno.

Nell'**esempio** in figura, l'ordine è cancellato solo se un token si trova all'interno della regione al momento della ricezione del segnale.

10.6 – Conclusioni

I diagrammi di attività descrivono un flusso di azioni che realizzano un certo comportamento specifico. L'enfasi non è sullo scambio di messaggi ma sui blocchi di comportamento.

Come tutti i diagrammi UML, possono essere usati sia a livello di analisi che di progettazione.