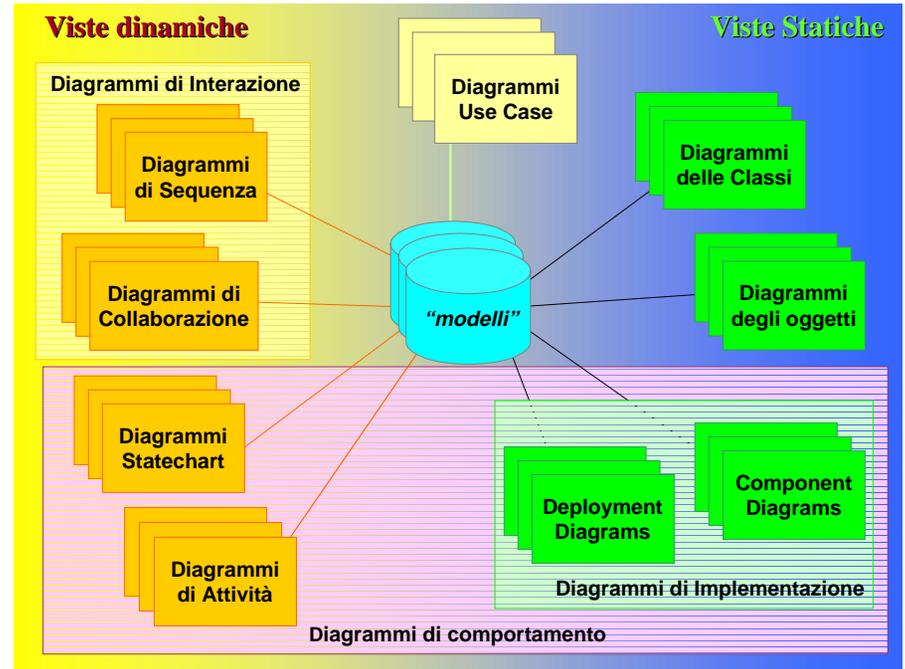


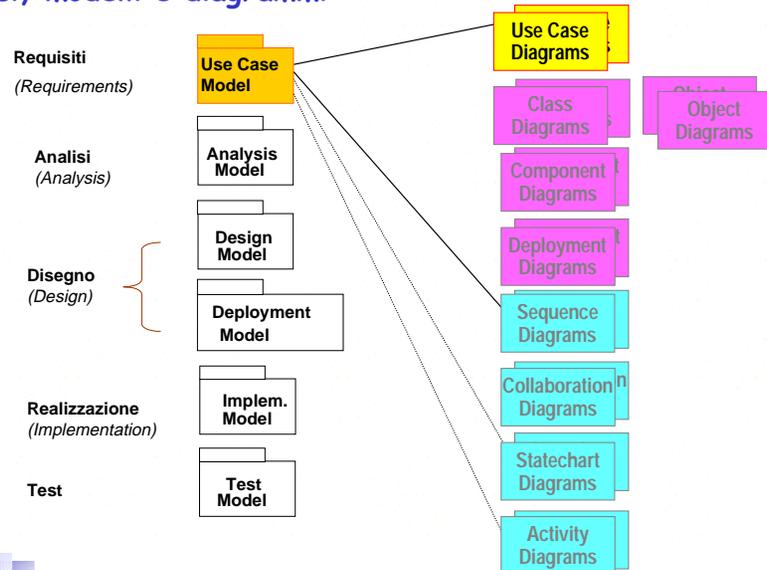
# UML

## Diagrammi casi d'uso

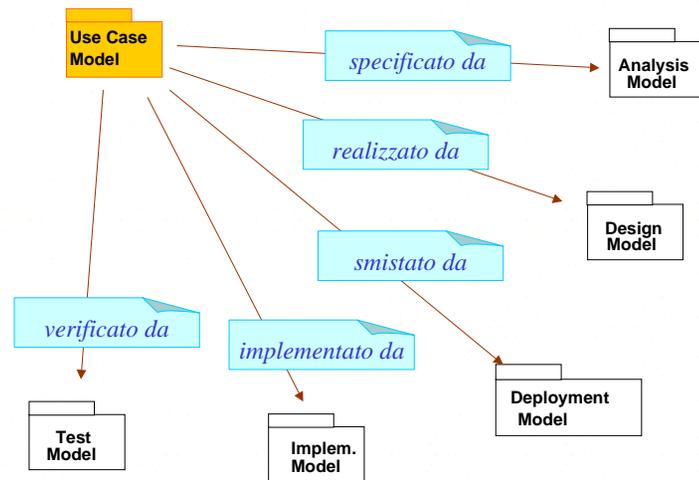
### Use case diagrams



### Fasi, modelli e diagrammi

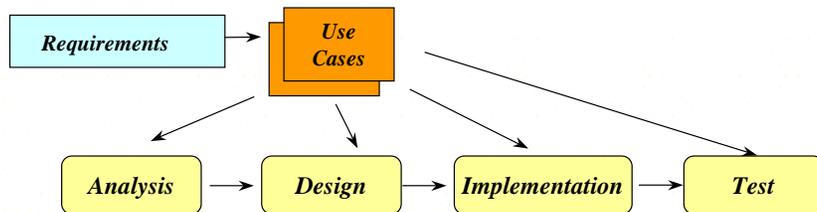


### Use case model e dipendenze



## Use-case driven design

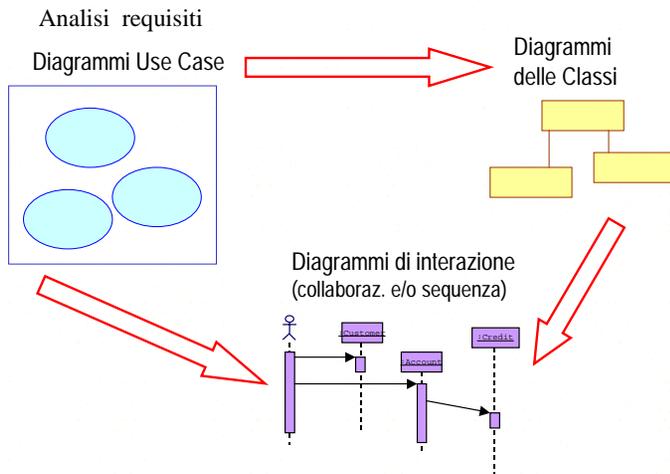
Lo **use case** definisce la funzionalità che è utilizzata come base per lo sviluppo delle fasi di analisi, di progetto (*disegno*), di implementazione e di test.



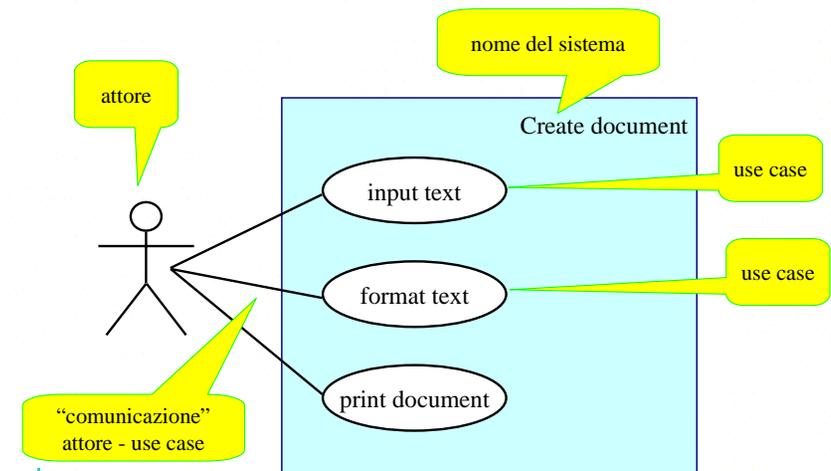
## Diagramma use case

- Un diagramma *use case* è un grafo composto da attori, un insieme di *use case* opzionalmente racchiusi da un rettangolo, associazioni tra attori e use case, relazioni tra use case e generalizzazioni tra attori.
- Un *diagramma use case* è un dispositivo di modellazione usato per descrivere i requisiti funzionali di un sottosistema o di una semplice classe, ma anche dell'intero sistema da analizzare.
- Nel "Unified Process" un "Use Case Model" descrive completamente i requisiti funzionali dell'intero sistema da analizzare raccogliendo tutti gli use case relativi ai sottosistemi in cui è stato scomposto il sistema.

## Prime fasi di utilizzo dei diagrammi Use Case



## Esempio di diagramma use case

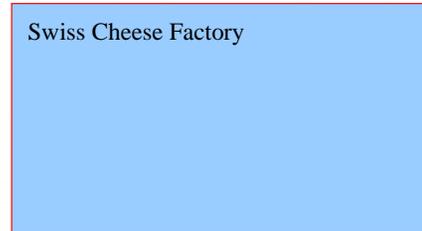




## USE CASE: raccomandazioni

- Gli identificatori degli use case devono utilizzare il nome di un verbo  $\Rightarrow$  e' un processo
- Le funzionalita' del sistema devono essere tracciabili all'interno degli use case
- Gli use case devono essere tracciabili all'interno del codice implementato
- I limiti (*boundary*) di un sistema possono essere:
  - limiti delle componenti HW e/o SW;
  - limiti dell'organizzazione in cui opera;
  - limiti del dipartimento in cui opera.

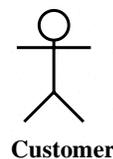
## A system boundary



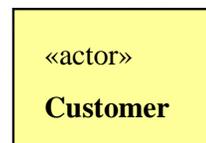
- *Determining the boundary can be a design activity.*
- *Note that the system may not be a software system; it could be a business or a hardware device for example.*

## Attore (*actor*)

Un attore è qualcuno o qualcosa che interagisce col sistema. Si rappresenta con un omino stilizzato (*stickman*), ed ha solitamente un nome



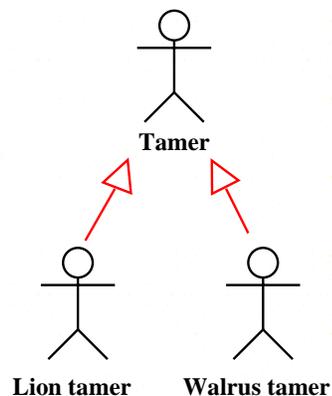
o come una classe con la specifica « actor »



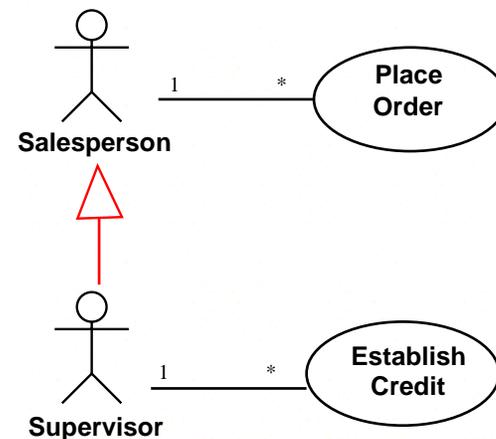
## attore (2)

- Col termine "qualcosa" vogliamo intendere che un attore può essere una persona, un dispositivo od un altro sistema a se stante.
- Gli attori possono essere attivi o passivi.
- Un attore *attivo* avvia un use case.
- Un attore *passivo* semplicemente riceve informazioni da un use case.

## Generalizzazione tra attori



## Generalizzazione tra attori (2)

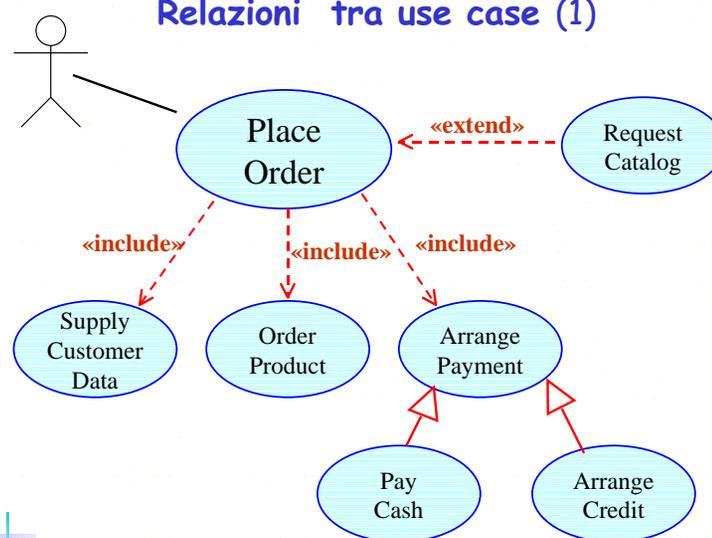


## Relazione attore - use case

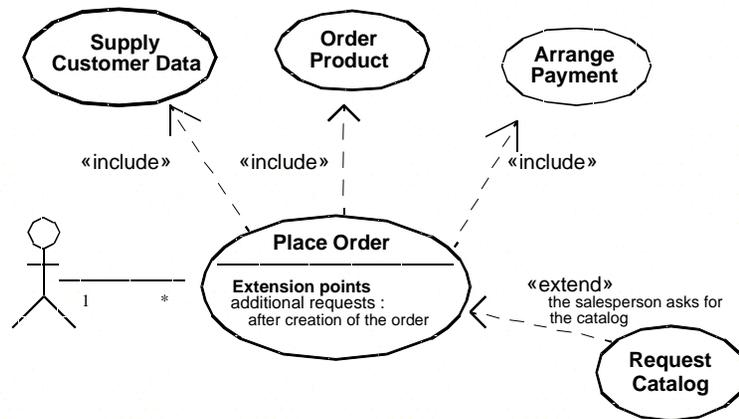


- Un attore interagisce con un use case mediante una **associazione di comunicazione** (*communication association*).
- Questa è **normalmente** una relazione uno a uno, priva di direzione.
- Le cose possono essere più complesse...

## Relazioni tra use case (1)



## Relazioni tra use case (2)



## Relazioni tra use case (3)



- **extend** - a dashed arrow indicating an addition to functionality of the base case.



- **include** - a dashed arrow indicating a calling relationship like a function call.

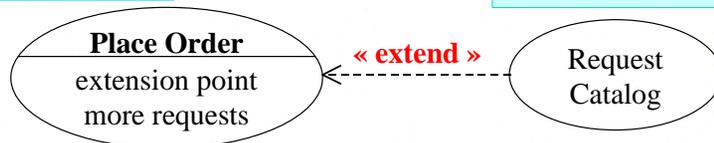


- **generalization** - a hollow arrowhead indicating inheritance.

“The UML User Guide”, 1999. ( The Three Amigos )

### « extend »

base use case



Il *client use case* aggiunge funzionalità al *base use case*, inserendo sequenze di azioni nelle azioni di base.

Il *base use case* ad un certo punto valuterà una condizione che, se risulta vera, l'azione “estesa” verrà inserita.

Il *client use case* ha senso solo nel contesto del *base use case*.

Il *base use case* implica l'estensione.

### « include »

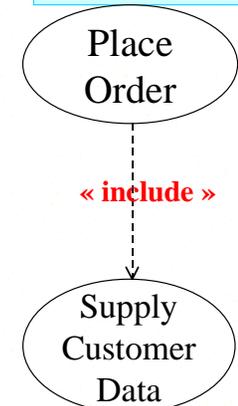
base or client case

La relazione *include* è come una chiamata a funzione.

Si svolge sotto il controllo del *case base* (o *client*).

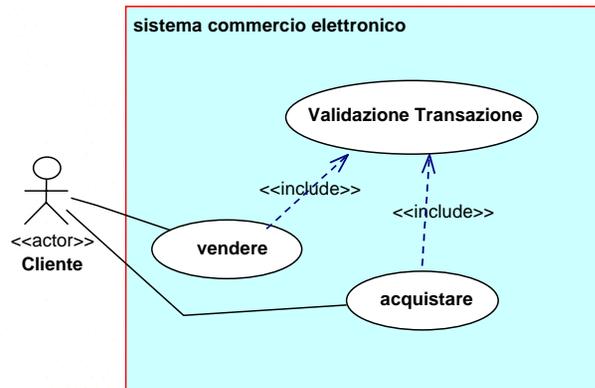
Il *base case* è in esecuzione e quando richiesto è richiamato il “fornitore” (*supplier case*).

Il *supplier use case* è un use case a se stante.



supplier case

## esempio include

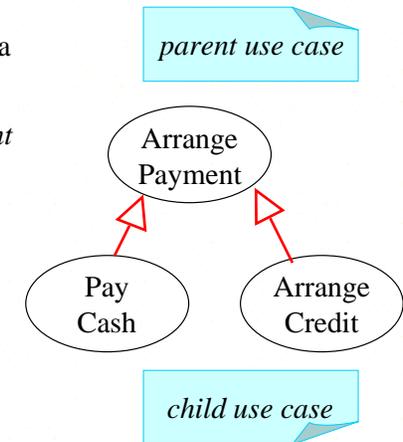


## Generalizzazione

L'eredità tra use case implica la sostituibilità.

Ogniqualvolta *ArrangePayment* è utilizzabile, *PayCash* potrebbe essere accettabile.

Il *child use case* può "estendere" il comportamento di *parent use case*, tuttavia potrebbe essere presa in considerazione la sostituibilità col *parent use case*.



## Usare gli use case

- *Establish the context of the system by identifying the actors that surround it.*
- *For each actor, consider the behaviour that each expects or requires the system to provide.*
- *Name these common behaviours as use cases.*
- *Factor common behaviour into new use cases that are used by others; factor variant behaviour into new use cases that extend more main line flows.*
- *Model these use cases, actors, and their relationships in a use case diagram.*
- *Adorn these use cases with notes that assert non functional requirements; you may have to attach some of these to the whole system.*

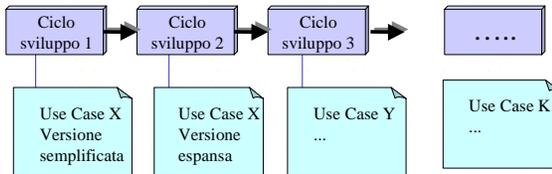
"The UML User Guide", 1999. ( The Three Amigos )

## Fasi per la redazione degli Use Case

1. Dalle funzionalità del sistema definire:
  - gli attori;
  - gli use case.
2. Scrivere gli use case ad alto livello
  - deciderne le prioritá (primario, secondario, opzionale)
3. Disegnare lo Use Case Diagram
4. Relazionare gli use case tra di loro
  - descrivere tali relazioni
5. Espandere gli use case piú importanti (use case espansi)
6. Rank use case

## Rank Use Case

- Schedulare i cicli di sviluppo in base agli Use Case



esempio tabella di ranking

Use Case	a	b	c	d	Somma
X	4	5	2	0	11
Y	2	1	2	1	6
K	1	1	1	1	4

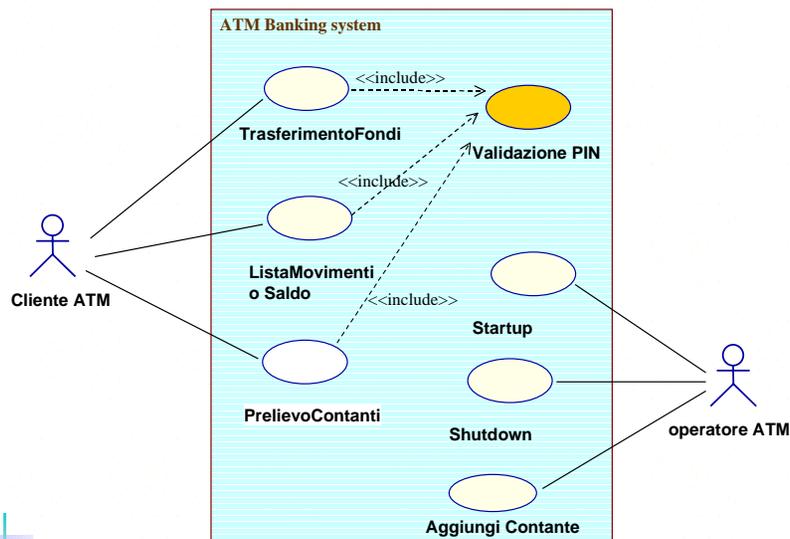
Criteria di ranking

- Importanza architeturale
- Maggior numero di funzionalita' di base
- Alto rischio
- Difficolta' di realizzazione

## USE CASE in descrizione narrativa

- Documento testuale che descrive la sequenza di eventi di un attore che utilizza il sistema.
- Serve a migliorare la comprensione dei requisiti
- Passo preliminare per descrivere i requisiti del sistema
- Accompagna la descrizione grafica (diagramma use case)

## Esempio "Diagramma use case sistema bancomat" alto livello



## es. USE CASE "Validazione PIN" (1)

In glossario

**Use case:** validazione PIN (*Personal Identification Number*)

**Sommario:** il sistema controlla e convalida il PIN del cliente

**Attore:** cliente ATM

**Precondizioni:** il sistema è in attesa e mostra sul display un messaggio di benvenuto.

**Descrizione** (funzionalità):

In glossario

- Il cliente inserisce l' **ATM Card** (*bancomat / carta di credito*);
- Il sistema riconosce l' ATM Card e ne legge il numero;
- Il sistema chiede il PIN;
- Il cliente inserisce il PIN;
- Il sistema controlla la data di scadenza e se la Card risulta in stato di smarrita o rubata;
- Se la Card è valida, il sistema controlla se il PIN è corretto (PIN mantenuto dal sistema);

## es. USE CASE "Validazione PIN" (2)

### Descrizione (cont.):

- 7) Se il PIN è corretto, il sistema controlla quali conti correnti sono accessibili con quella Card;
- 8) Il sistema mostra al cliente le possibili transazioni (menu): prelievo, saldo, lista movimenti o trasferimento;

### Alternative:

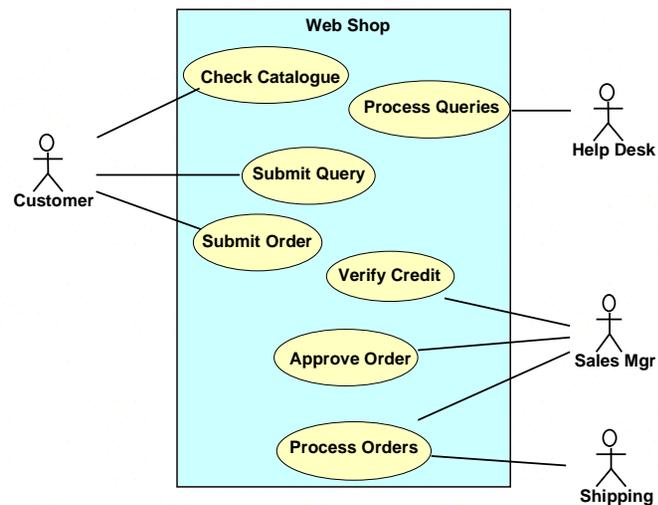
- 1) Il sistema non riconosce l'ATM Card, la Card viene espulsa;
- 2) Il sistema determina che la Card è scaduta: la Card viene confiscata;
- 3) Il sistema determina che la Card risulta smarrita o rubata: la Card viene confiscata;
- 4) Il cliente digita un PIN non corretto...
- 5) ...

### Postcondizioni:

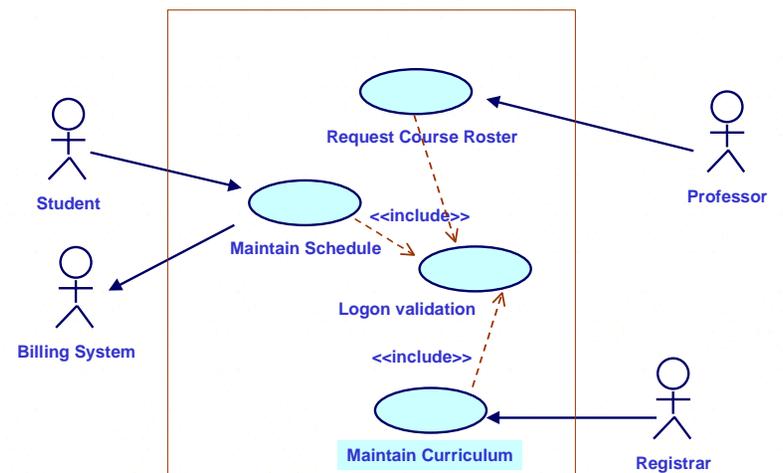
Il PIN è stato validato.

## Esempi di diagrammi use case

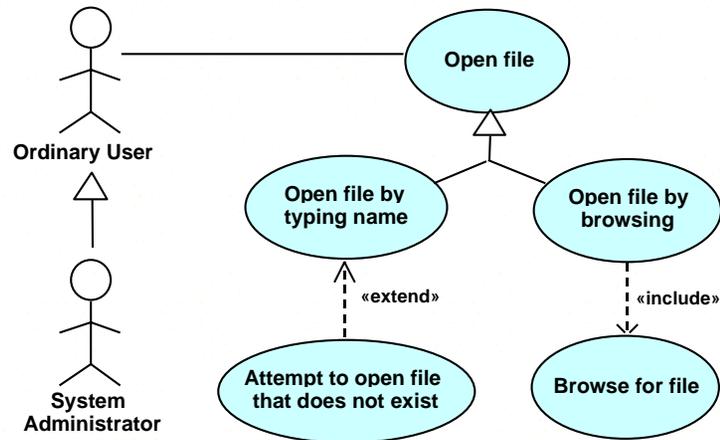
### Commercio elettronico



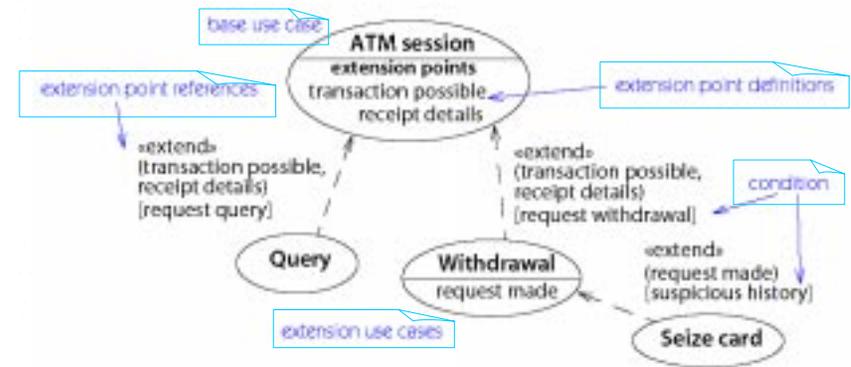
### University registration system



## Apertura file



## ATM



## Bibliografia

Grady Booch, James Rumbaugh, Ivar Jacobson.  
The Unified Modeling Language User Guide, Addison  
Wesley , (1999).

Grady Booch, James Rumbaugh, Ivar Jacobson  
The Unified Modeling Language Reference Manual,  
Addison Wesley, (1999).

Ivar Jacobson, Grady Booch, James Rumbaugh  
The Unified Software Development Process,  
Addison Wesley, (1999).

R.S Pressman "Principi di Ingegneria del software" 4° ed  
McGraw-Hill, (2004).

## Riferimenti nel Web

**OMG UML - Reference manual UML 1.5**  
[www.omg.org/uml/](http://www.omg.org/uml/)

**Materiali e riferimenti su CASI D'USO**  
[alastair.cockburn.us/usecases/usecases.html](http://alastair.cockburn.us/usecases/usecases.html)

**UML: tool, demo, doc**  
[www.rational.com](http://www.rational.com)

**UML: Tutorial e link**  
[www.kobryn.com](http://www.kobryn.com)