

Modello comportamentale

- Rappresenta gli aspetti funzionali del sistema da un **punto di vista operativo**, evidenziando come gli oggetti collaborano ed interagiscono al fine di offrire i servizi che il sistema mette a disposizione
- Fa uso di vari formalismi:
 - **Use case diagram** (per descrivere scenari di funzionamento)
 - **Activity diagram** (per descrivere il flusso di elaborazione)
 - **Sequence diagram** (per descrivere l'interazione tra gli oggetti)
 - **Collaboration diagram** (per descrivere l'interazione tra gli oggetti)
- Viene costruito in modo **iterativo** ed **incrementale**, usando le informazioni del **modello dei dati**, che a sua volta fa uso del modello comportamentale per identificare *operazioni* e classi aggiuntive (*control classes* e *boundary classes*)

Use Case Diagram

- Può essere sviluppato a **differenti livelli di astrazione** (sia in fase di OOA che OOD)
- Durante la fase di OOA, si concentra su **COSA** il sistema deve fare (*scenari di funzionamento*)
- Un caso d'uso rappresenta:
 - **una funzionalità completa** (flusso principale, sottoflussi e alternative)
 - **una funzionalità visibile dall'esterno**
 - **un comportamento ortogonale** (ogni use case viene eseguito in modo indipendente dagli altri)
 - **una funzionalità originata da un attore del sistema** (una volta originato, il caso d'uso può interagire con altri attori)
 - **una funzionalità che produce un risultato significativo per un attore**

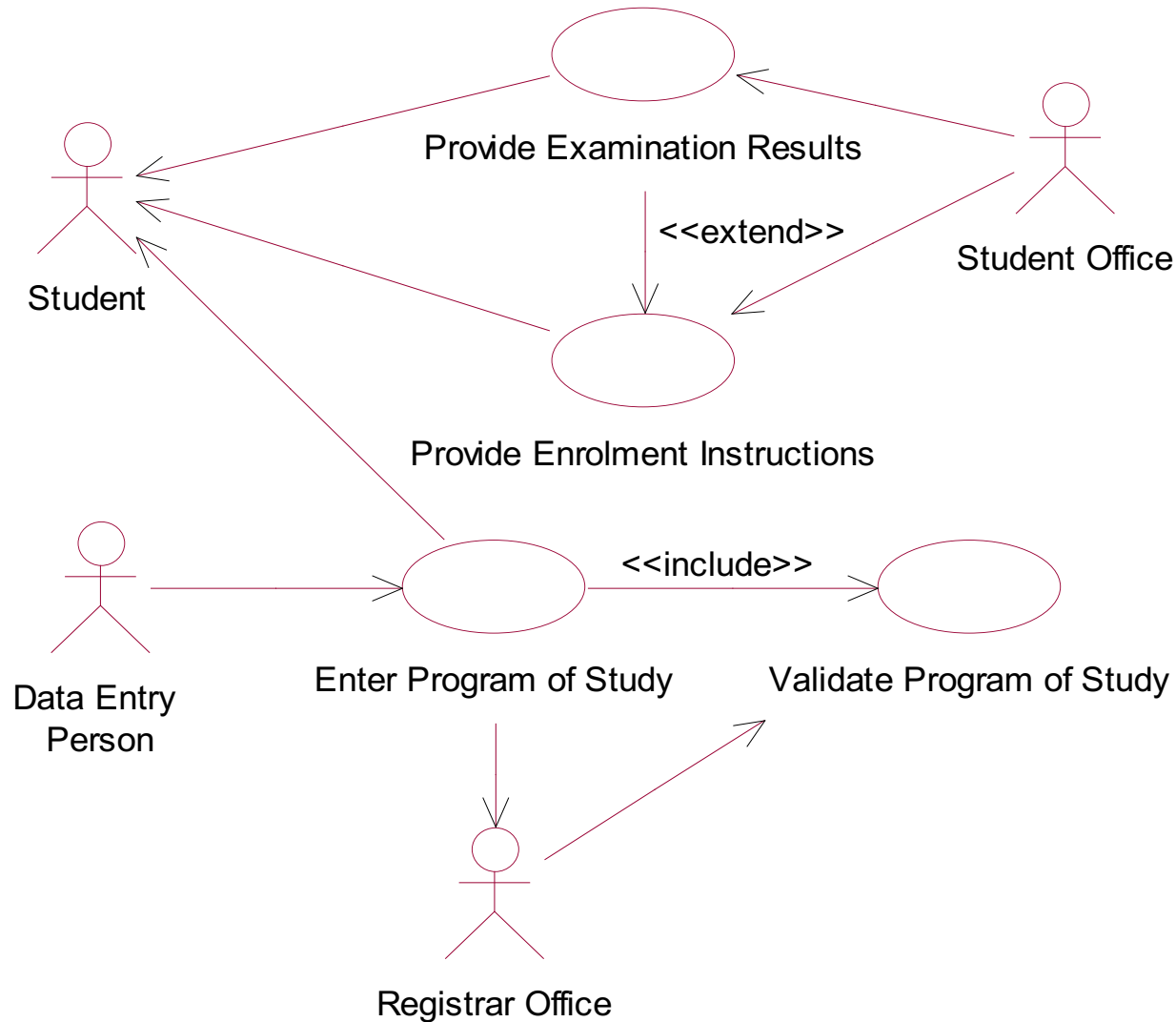
Identificazione dei casi d'uso

- A partire da:
 - Insieme dei **requisiti utente**
 - **Attori** del sistema (insieme ai relativi obiettivi)
- L'identificazione può essere facilitata facendosi guidare dalle seguenti domande:
 - Quali sono i compiti principali svolti da ciascun attore?
 - Un attore accede o modifica l'informazione nel sistema?
 - L'attore rappresenta il tramite mediante cui il sistema viene informato di modifiche apportate in altri sistemi?
 - L'attore deve essere informato di eventuali cambiamenti avvenuti nel sistema?
- Durante la fase di OOA, i casi d'uso identificano le **necessità degli attori** del sistema

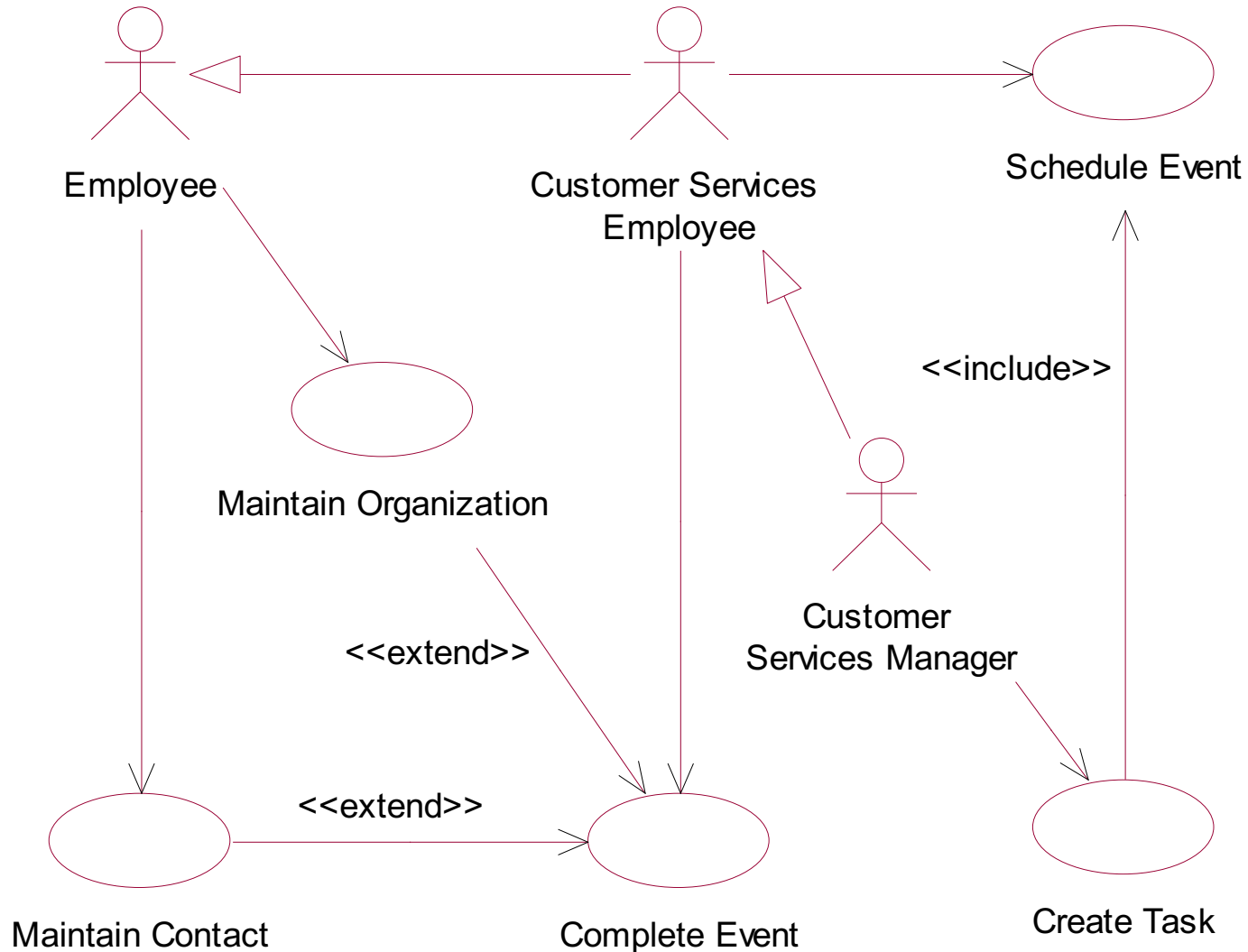
Specifica di *use case diagram*

- **Rappresentazione grafica** di attori, casi d'uso e relazioni
- Si possono rappresentare quattro tipi di relazioni:
 - **Associazione** (tra attore e caso d'uso)
 - **Include** (identificato dallo *stereotype*: «include»)
 - Un caso d'uso *included* è sempre necessario per completare il caso d'uso con il quale è messo in relazione
 - **Extend** (identificato dallo *stereotype* : «extend»)
 - Un caso d'uso *extended* può attivare il caso d'uso dal quale viene esteso (ma tale attivazione non è necessaria per completare il caso d'uso *extended*)
 - **Generalizzazione**

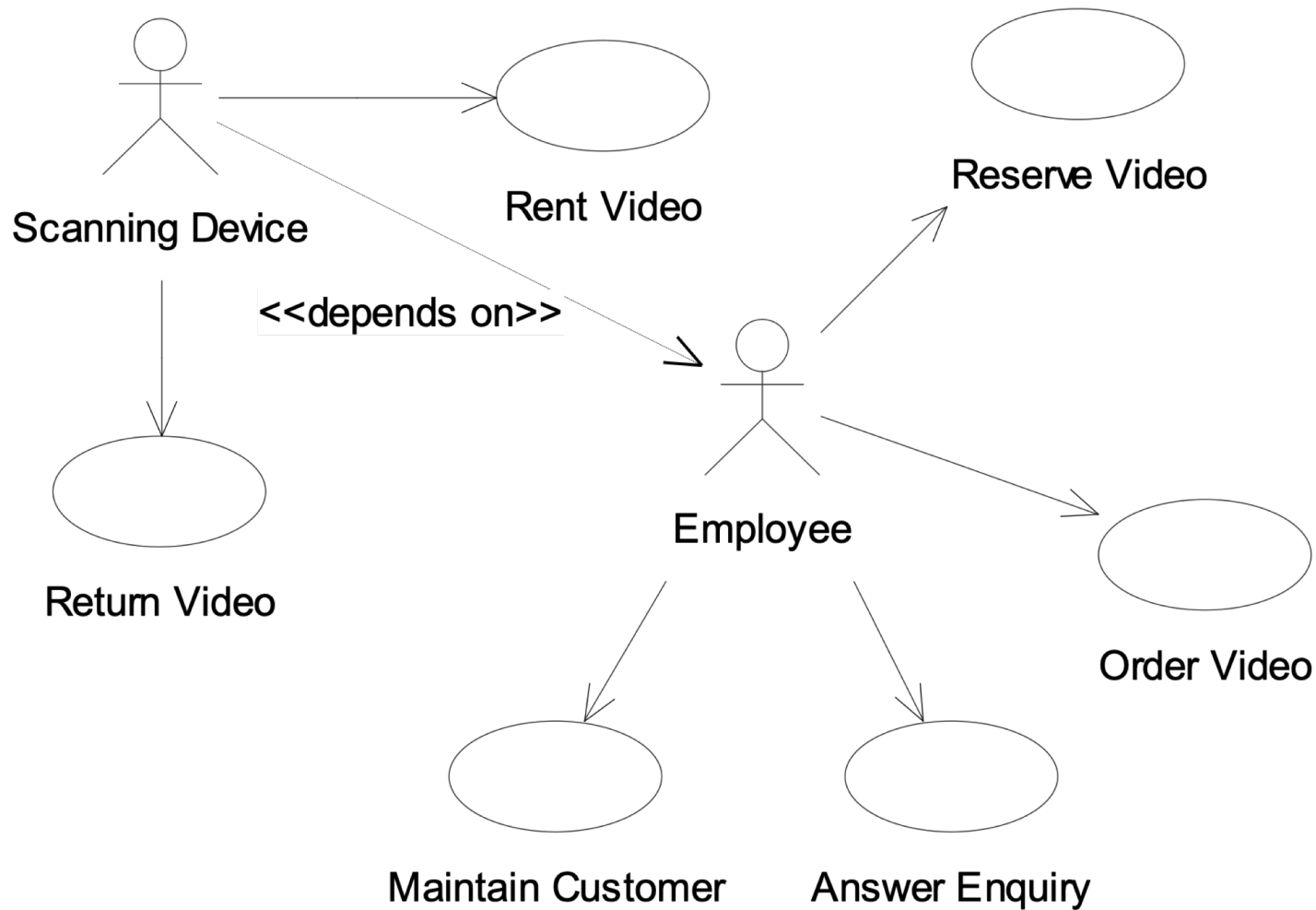
Example A.5 – University Enrolment



Example C.3 – Contact Management



Example B.4 – Video Store



Example B.4 – Video Store (Rent Video - 1)

Brief Description	<p><i>A customer wishes to rent a video tape or disk that is picked from the store's shelves or that has been previously reserved by the customer.</i></p> <p><i>Provided the customer has a non-delinquent account, the tape is rented out once the payment has been received.</i></p> <p><i>If the tape is not returned in a timely fashion, an overdue notice is mailed to the customer.</i></p>
Actors	<p><i>Employee, Scanning Device</i></p>
Preconditions	<p><i>Video tape or disk is available to be hired.</i></p> <p><i>Customer has a membership card. Scanner devices work correctly.</i></p> <p><i>Employee at the front desk knows how to use the system.</i></p>

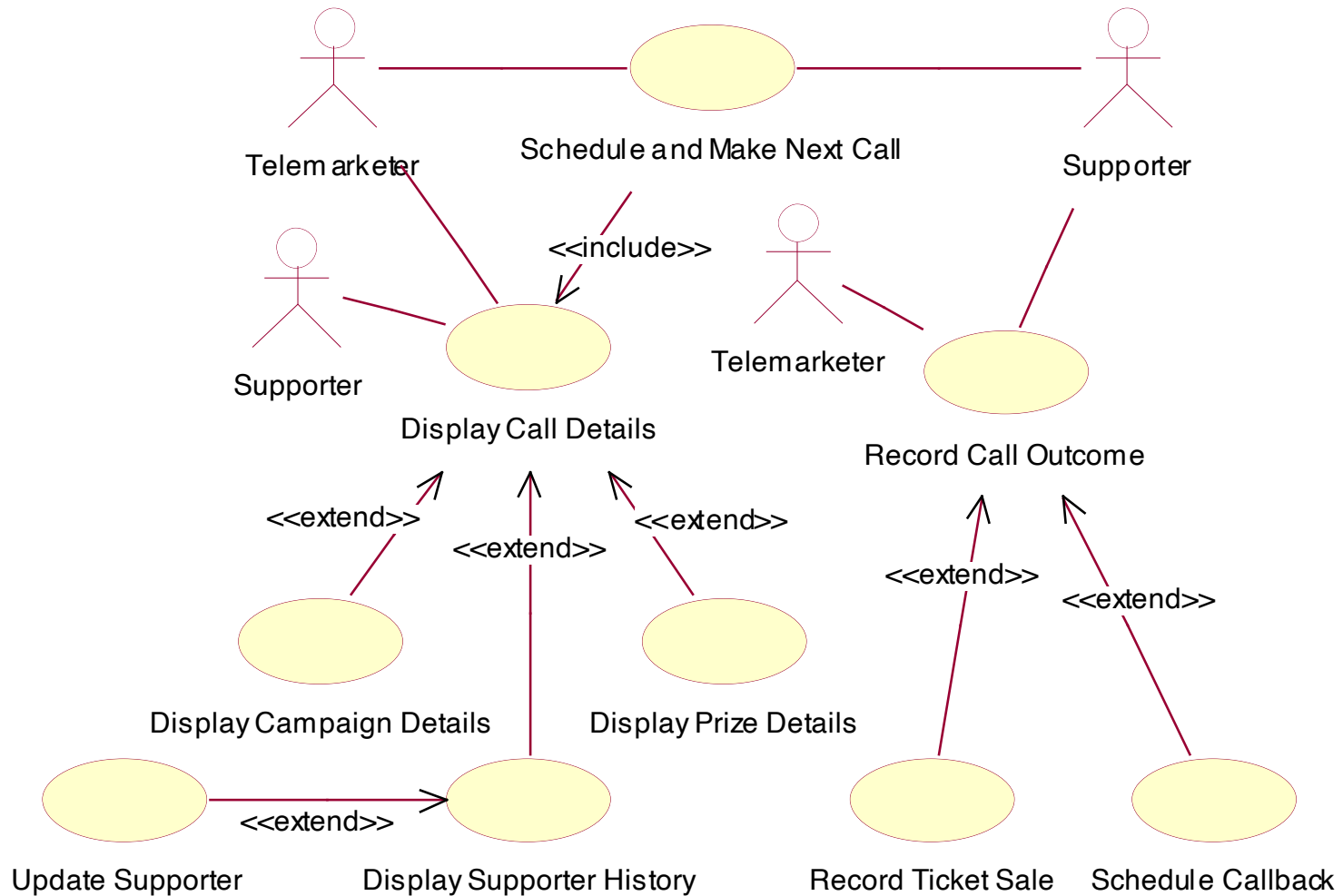
Example B.4 – Video Store (Rent Video - 2)

Main Flow	<p><i>A customer may inquire an employee about video availability (including a reserved video) or may pick a tape or disk from the shelves. The video and membership card are scanned and any delinquent or overdue details are brought up for the employee attention. If the customer does not have a delinquent rating, then he/she can hire up to a maximum of eight videos. However, if the rating of the customer is "unreliable" then a deposit of one rental period for each tape or disk is requested. Once the amount payable is received, the stock is updated and the tapes and disks are handed out to the customer together with the rental receipt. The customer pays by cash, credit card or electronic transfer. Each rental record stores the check-out and due-in dates together with the identification of the employee. A separate rental record is created for each video hired.</i></p> <p><i>The use case will generate an overdue notice to the customer if a video has not been returned within two days of the due date, and a second notice after another two days (and at that time the customer is noted as "delinquent").</i></p>
------------------	---

Example B.4 – Video Store (Rent Video - 3)

Alternative Flows	<p><i>A customer does not have a membership card. In this case, the Maintain Customer use case may be activated to issue a new card.</i></p> <p><i>An attempt to rent too many videos.</i></p> <p><i>No videos can be rented because of the customer's delinquent rating.</i></p> <p><i>The video medium or membership card cannot be scanned because they are damaged</i></p> <p><i>The electronic transfer or credit card payment is refused.</i></p>
Postconditions	<p><i>Videos are rented out and the database is updated accordingly.</i></p>

Example D.3 – Telemarketing



Activity Diagram

- Rappresenta a vari livelli di astrazione il **flusso di esecuzione**, sia *sequenziale* che *concorrente*, in una applicazione object-oriented
- E' una variante degli *state diagram*, in cui gli *stati* rappresentano l'esecuzione di *azioni* e le *transizioni* sono attivate dal completamento di tali azioni
- Usato principalmente in fase di OOD per rappresentare il flusso di esecuzione delle operazioni definite nel *class diagram*
- In fase di OOA, viene usato per rappresentare il **flusso delle attività nella esecuzione di un caso d'uso** (un caso d'uso può essere associato ad uno o più *activity diagram*)
- Poichè non vengono mostrati gli oggetti che eseguono le attività, può essere costruito anche in **assenza** del *class diagram*
- In presenza del class diagram, ogni **attività** può essere associata ad **una o più operazioni** appartenenti ad una o più classi

Specifica di activity diagram

- Un evento (*esterno*) che origina un caso d'uso viene modellato come un **evento** che causa l'esecuzione di un *activity diagram*
- Gli **action state** vengono **identificati** a partire dalla descrizione testuale dello scenario di funzionamento di un caso d'uso
- Gli *action state* vengono quindi associati mediante **transition lines** (che possono essere controllate da **guard conditions**)
- Le transizioni in uscita da un *action state* vengono percorse quando l'*action state* viene completato (l'esecuzione procede da un *action state* al successivo)
- Un *action state* viene **completato** quando la sua elaborazione termina
- **Flussi concorrenti** vengono modellati con **barre di sincronizzazione** (*barre fork-join*)
- **Flussi alternativi** vengono modellati con **nodi decisionali** (*branch/merge diamonds*)
- Eventi esterni non sono generalmente modellati

Example B.5 – Video Store

Rent Video use case

