Modelli per la specifica dei requisiti

- ***Specifica*** significa ***definizione***
- Definire le proprieta` che l'applicazione dovra` avere, evitando di descrivere una loro possibile realizzazione
- Vari modelli descrittivi possibili, per problemi o classi di applicazioni diverse
- I modelli sono, obbligatoriamente, ***astratti*** poiche` non possono descrivere tutti i dettagli sul sistema
  - Modelli orientati all'elaborazione dati (data-flow)
  - Modelli basati su composizione (modelli semantici dei dati)
  - Modelli basati su classificazione (modelli ad oggetti)
  - Modelli basati su risposte a stimoli (real-time)
  - Modelli orientati a processi (reti di Petri)

- Modelli semantici dei dati
- Modelli ad oggetti
- Dizionari dei dati
- Modelli data-flow

Modelli semantici dei dati: il modello Entità-Relazioni

- Strumento per descrivere la struttura concettuale (o *schema concettuale*) dei dati (Chen, 1972)

- Applicazioni di tipo gestionale

- Impiegato nelle fasi iniziali della progettazione di sistemi di basi di dati

- Definisce le entita` del sistema, le associazioni tra esse e gli attributi delle entita`

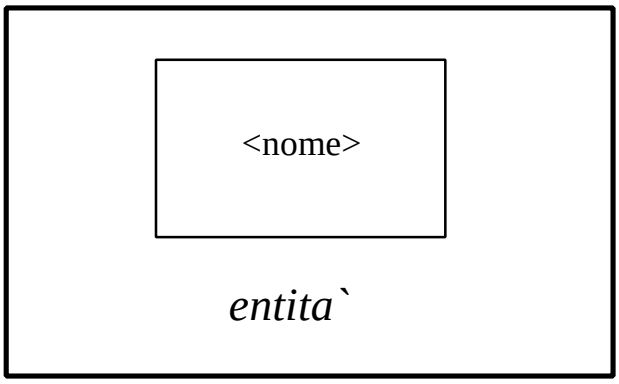
  Entità, oggetti necessari e sufficienti per il corretto funzionamento di un sistema informativo automatizzato

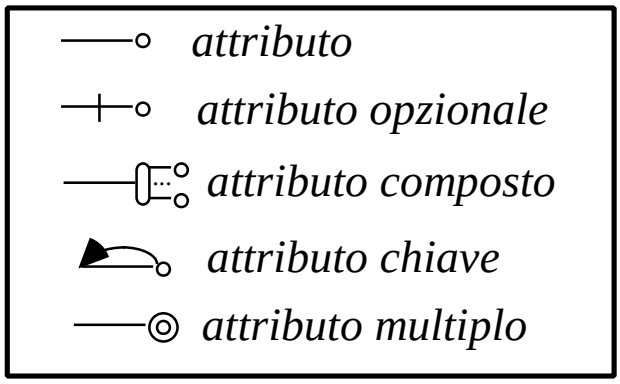
  Proprietà o *attributi*, servono a descrivere le entità (ad esempio, TITOLO, AUTORE, etc. per l'entità LIBRI)

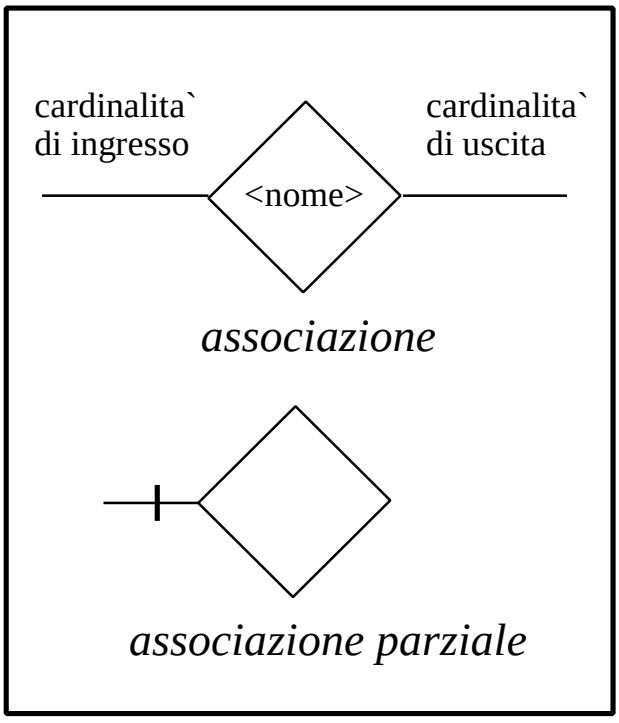
  Associazioni (o relazioni), stabiliscono una associazione o un legame logico tra entità

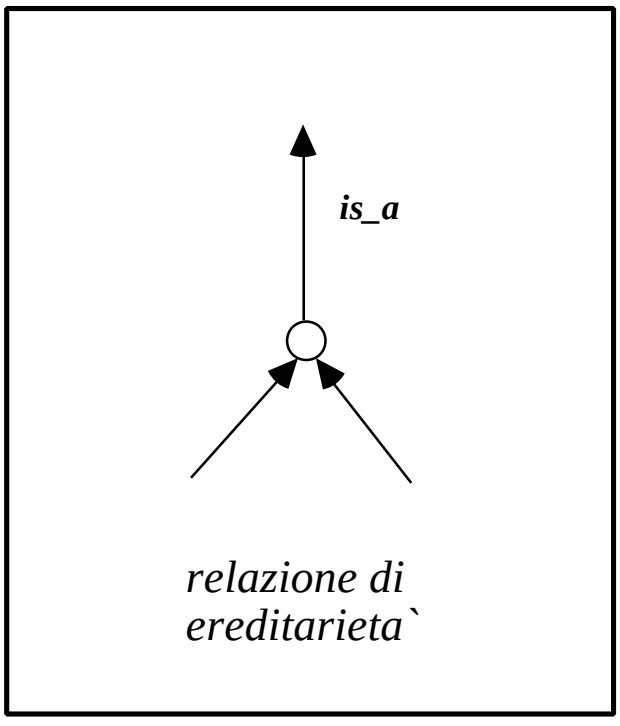
- Realizzabile facilmente in basi dati relazionali

Notazione del modello ER









- Formalismo grafico di descrizione sufficientemente astratto da poter servire come *strumento di comunicazione* tra progettisti e committenti

Meccanismi di astrazione: classificazione

- Oggetti diversi vengono riconosciuti come omogenei, appartenenti alla stessa categoria o classe

- Nel modellare una realtà bibliotecaria, ad esempio, occorrerà individuare una classe (tipo entità) DOCUMENTI (ad esempio per effettuare degli ordini)

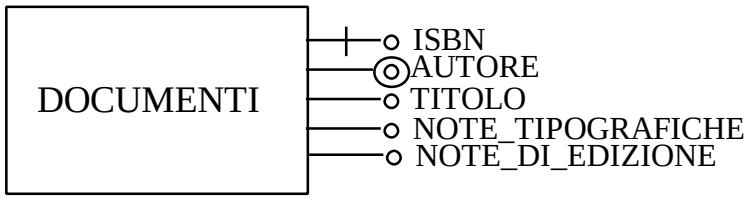
- Se si vuole costruire un sistema informativo per le opere d'arte di proprietà di un ente, occorrerà individuare certamente una classe OPERE_D'ARTE

Meccanismi di astrazione: aggregazione

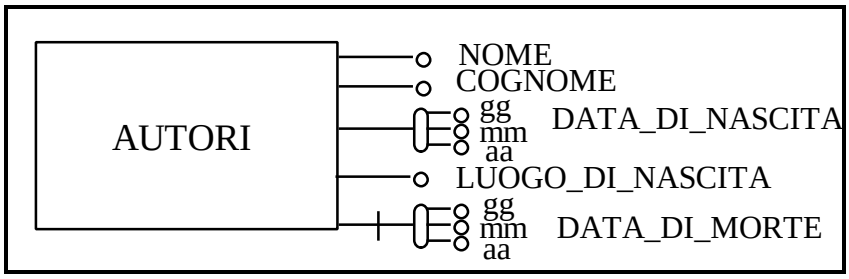
- Una classe (tipo entità) è caratterizzata attraverso un aggregato di *attributi*.

- La struttura è un insieme finito di proprietà (o attributi) comuni a tutti gli elementi

- Ad esempio, per i DOCUMENTI:
  TITOLO
  AUTORE
  NOTE_DI_EDIZIONE
  NOTE_TIPOGRAFICHE
  etc. (*Int. Standard Bibliographic Description*)
  ISBN (*Attributo opzionale*)



- Identifichiamo una entità AUTORI, i cui attributi sono per esempio:
  NOME
  COGNOME
  DATA_DI_NASCITA
  LUOGO_DI_NASCITA (stato)
  DATA_DI_MORTE (*opzionale*)

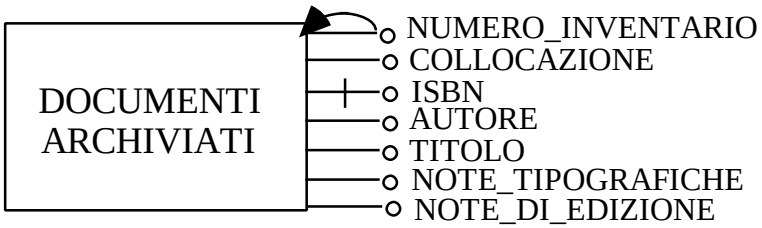

AUTORI
NOME
COGNOME
gg
mm
aa
DATA_DI_NASCITA
LUOGO_DI_NASCITA
gg
mm
aa
DATA_DI_MORTE

Chiave primaria di un'entita`

- Tra gli attributi di una entità (tipo entità) può esserci un attributo (o un insieme di attributi) che costituiscono una *chiave primaria*

- L'attributo o l'insieme di attributi che identificano in modo univoco ogni elemento di una entità

- Ad esempio, NUMERO_INVENTARIO per l'entità DOCUMENTI_ARCHIVIATI



attributo chiave

- La chiave primaria può essere utilizzata in sostituzione dell'elemento che individua

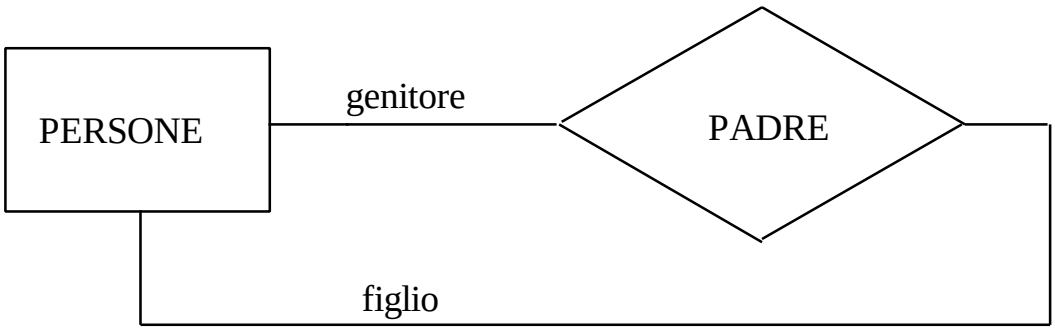
- Spesso e` creata appositamente (ad esempio, codice fiscale di un contribuente, numero di matricola di uno studente)

Modello ER: associazioni

- Una *relazione* (o *associazione*) costituisce, nel modello E-R, una connessione tra entità

- A ogni relazione viene associata la lista ordinata delle entità che vi partecipano

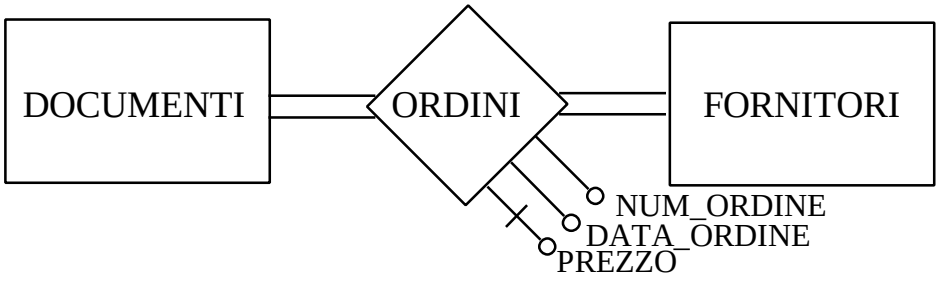
- In caso di ambiguita`, si specifica il *ruolo* delle diverse presenze di una stessa entità in una relazione



- Cardinalita` 1:1, 1:n, n:m

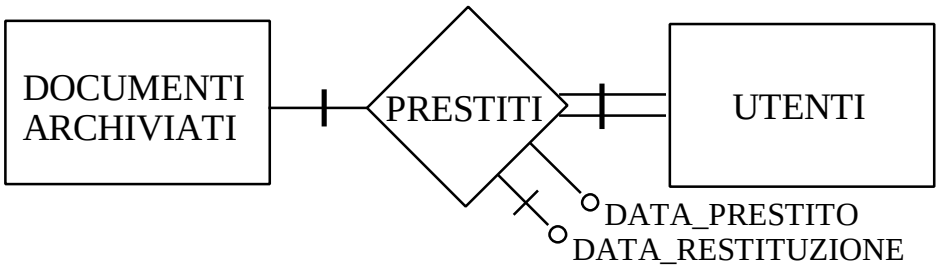
- Le associazioni possono presentare attributi

- Ad esempio, ordine di un documento: coinvolge l'entità DOCUMENTI ed i FORNITORI



Associazione n:m, ciascun documento può essere ordinato a più fornitori, ogni fornitore può ricevere l'ordine di più libri

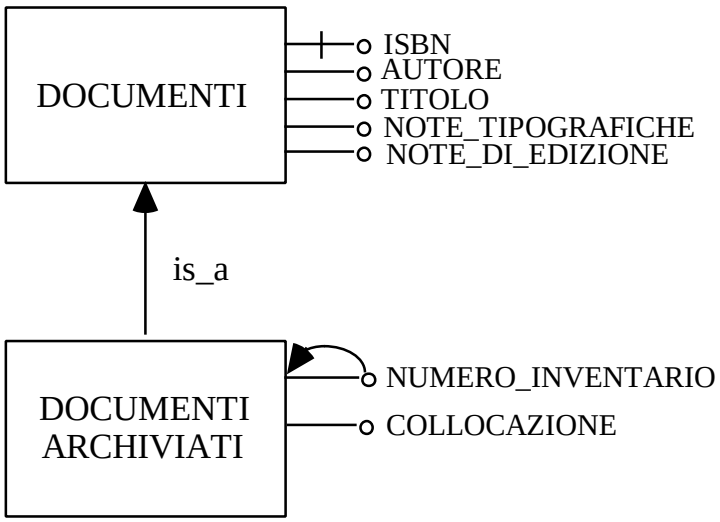
- Ad esempio, prestito di un documento: coinvolge l'entità DOCUMENTI_ARCHIVIATI e gli UTENTI



Meccanismi di astrazione: generalizzazione

- Meccanismo di astrazione che porta a riconoscere che una classe è più generale di un'altra o viceversa

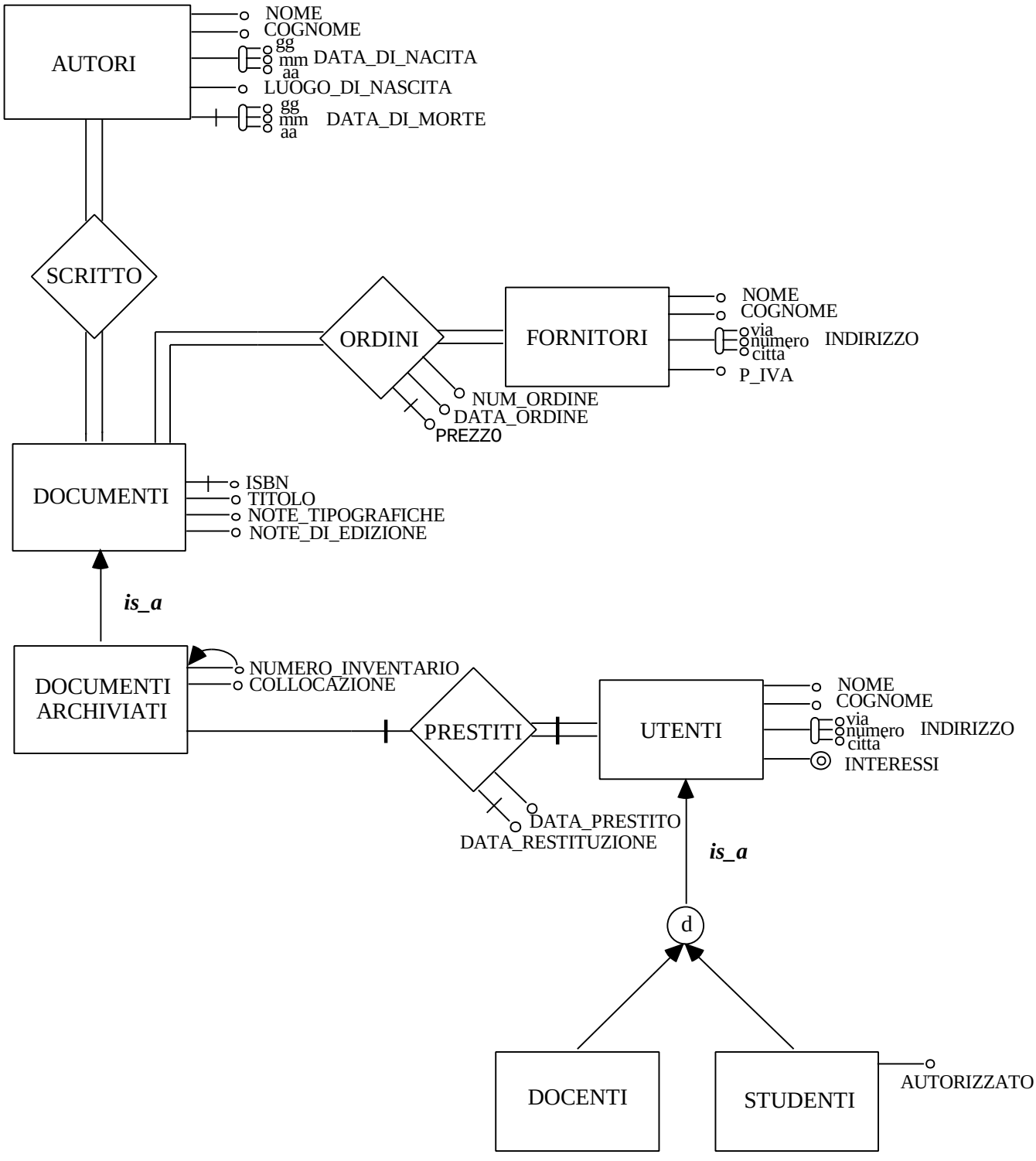
- Ad esempio:



Ogni elemento dell'entità DOCUMENTI_ARCHIVIATI è anche un elemento di DOCUMENTI

Gli attributi che descrivono i DOCUMENTI sono "ereditati" da DOCUMENTI_ARCHIVIATI

Schema concettuale: un esempio

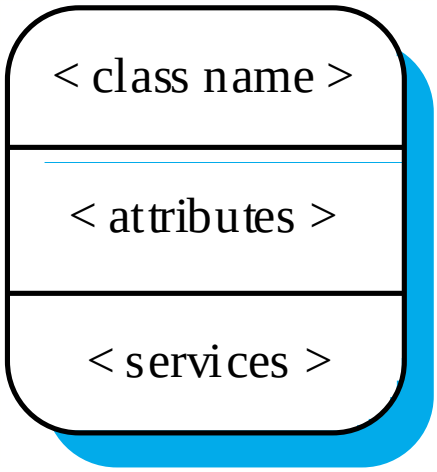


Lo schema deve essere:

- *completo* (tutte e sole le proprietà dei concetti della realtà di interesse nei requisiti);
- *corretto* (strutture del modello utilizzate in modo conforme nello schema);
- *minimo* (non ridondante);
- *leggibile* (deve rappresentare la realtà di interesse in modo comprensibile ed autoesplicativo);
- *indipendente dalle applicazioni*;
- *estendibile*;
- *indipendente dallo strumento informatico scelto*.

Modelli ad oggetti

- Descrivono il sistema in termini di classi di oggetti
- Una classe di oggetti e` un'astrazione su un insieme di oggetti con attributi e servizi comuni

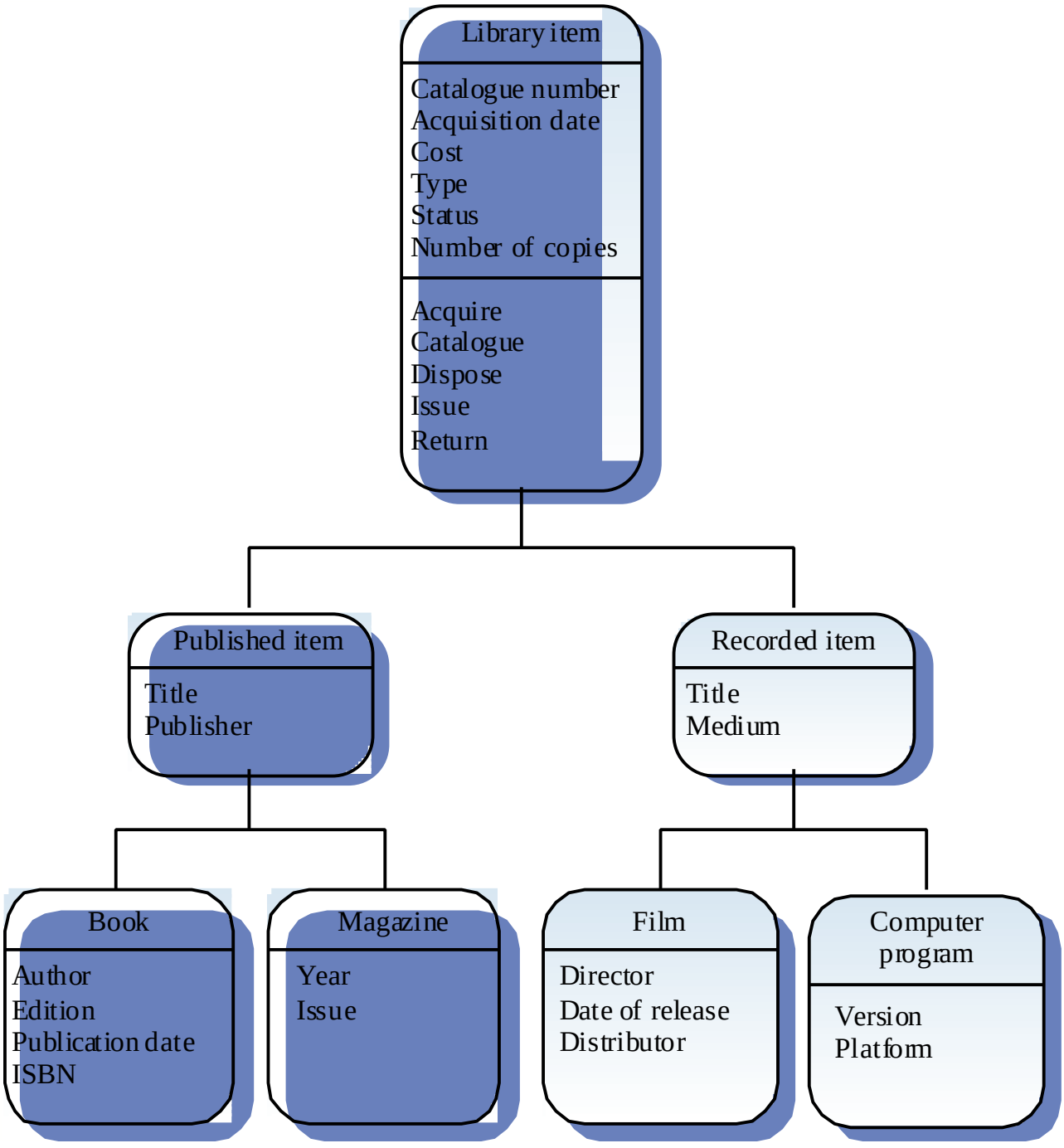


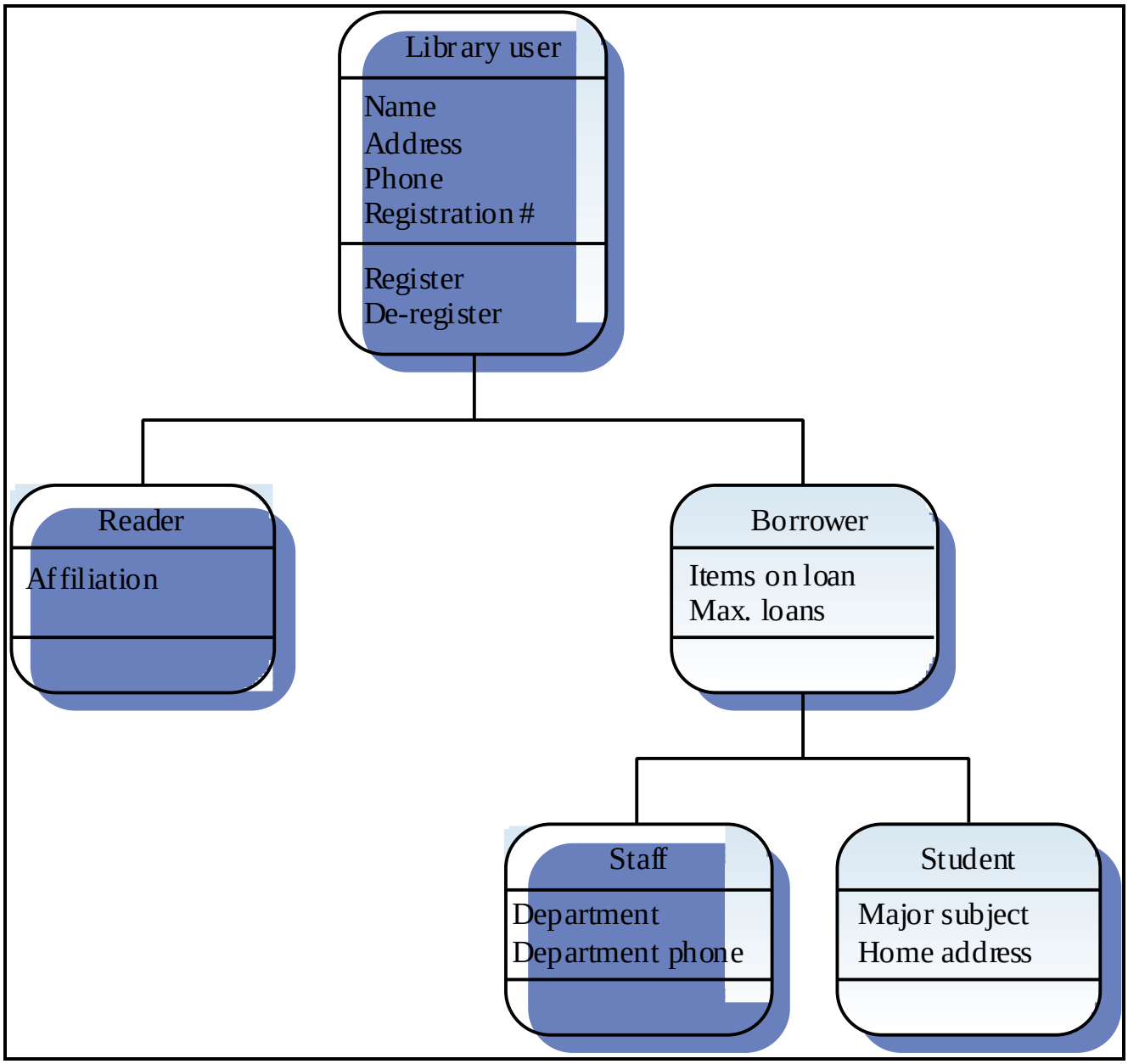
- Diverse tipologie di modelli:
  - a ereditarieta`
  - ad aggregazione
  - di servizio
- Identificazione delle classi di oggetti, processo complesso che richiede una profonda comprensione del dominio applicativo

Ereditarieta`

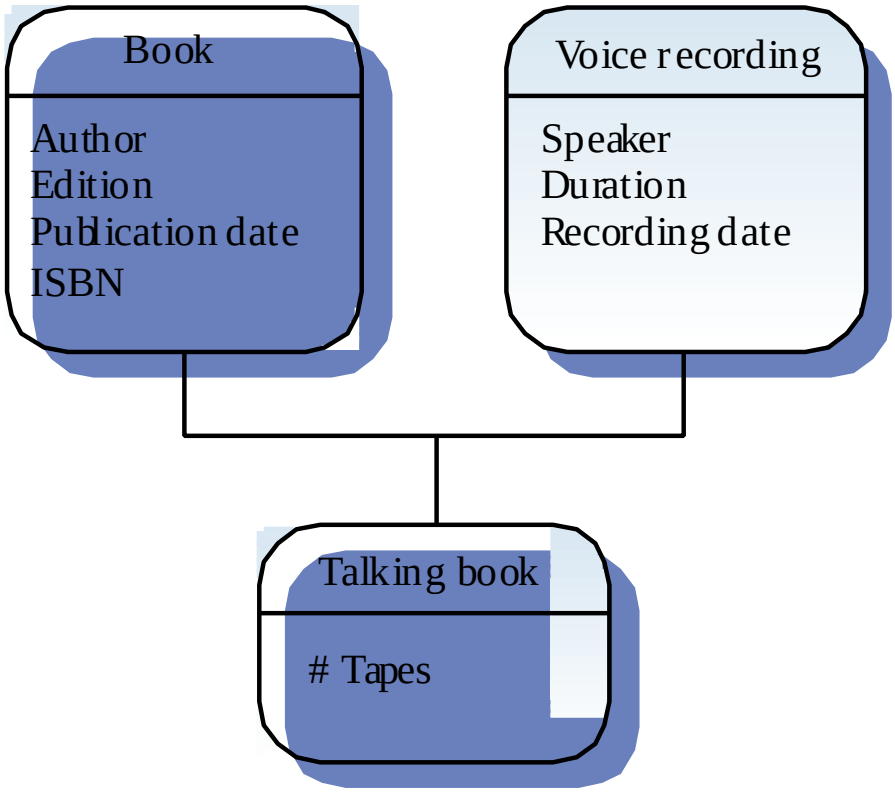
- Si organizzano le classi in gerarchia
- Le classi ai livelli piu` alti della gerarchia riflettono le caratteristiche comuni di tutte le classi
- Le classi ereditano i propri attributi e servizi dalle super-classi
- Il progetto della gerarchia e` la parte piu` complessa

Esempio biblioteca




Library user
Name
Address
Phone
Registration #
Register
De-register
Reader
Affiliation
Borrower
Items on loan
Max. loans
Staff
Department
Department phone
Student
Major subject
Home address

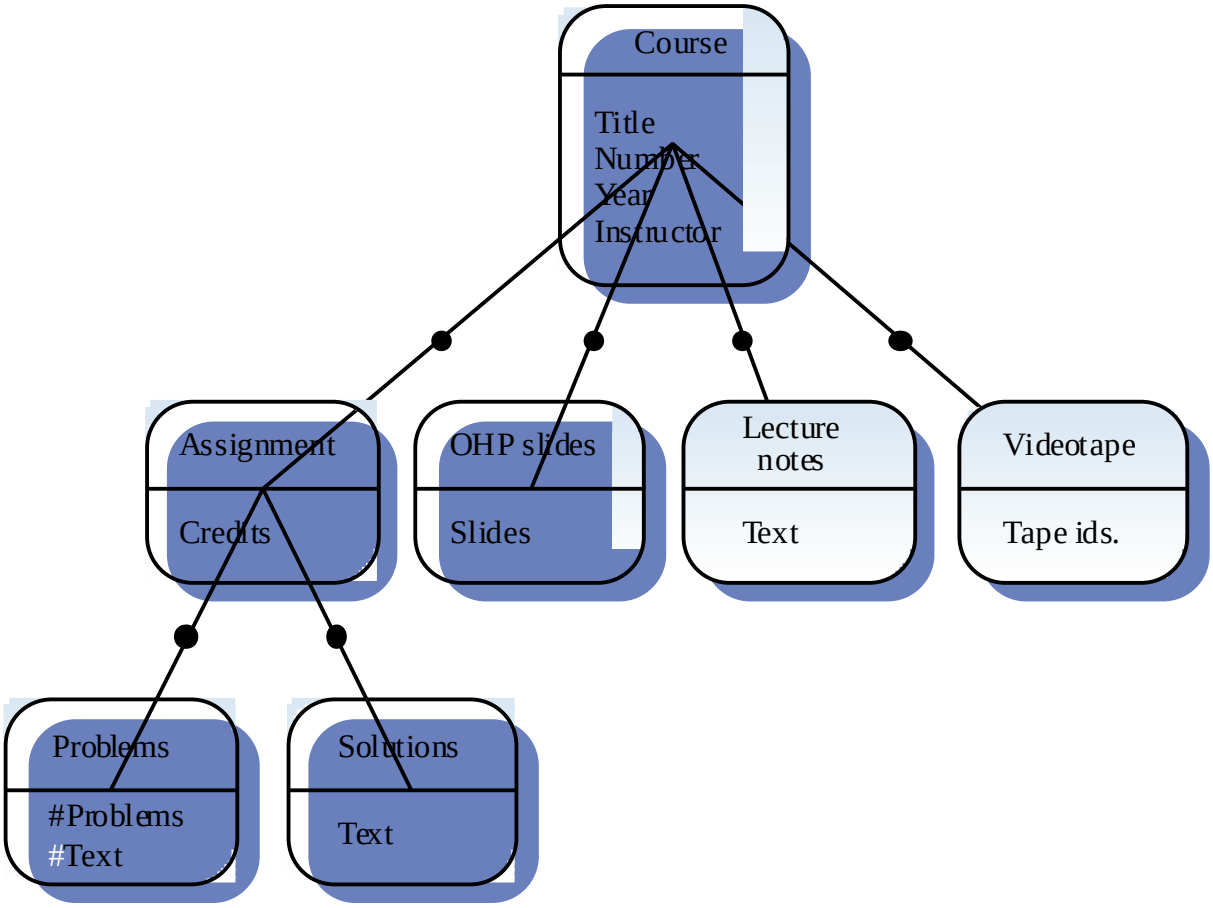
Ereditarieta` multipla



- Le classi possono ereditare attributi o servizi da piu` super-classi
- Si possono verificare conflitti se sono definiti attributi/servizi con lo stesso nome in piu` super-classi
- Rende piu` complessa la riorganizzazione della gerarchia

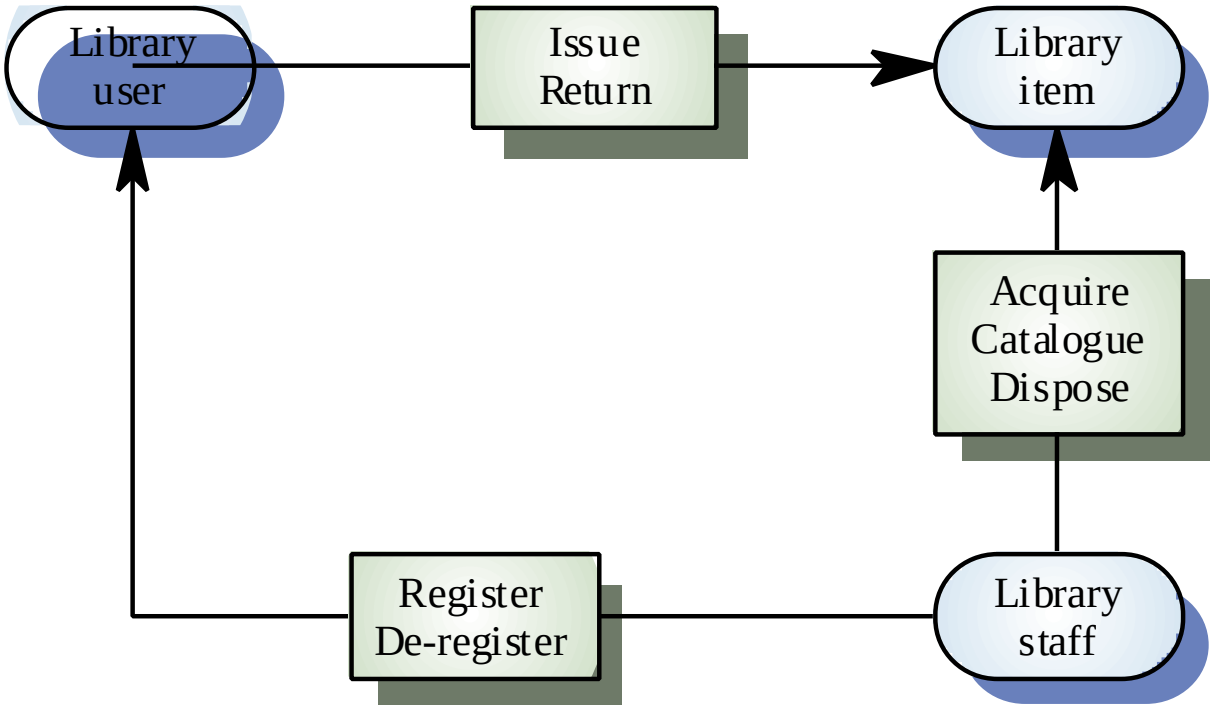
Aggregazione

- Alcuni oggetti sono aggregazione di altri oggetti



Servizi

- Evidenziano come i servizi forniti da un oggetto possano essere usati da altri oggetti



Diagrammi di Flusso dei Dati

- Complementari al formalismo E-R: infatti non si concentrano sulla struttura e sul significato dei dati, bensì sulle operazioni, o ***funzioni***, applicate ad essi

- Mostrano come i dati vengono elaborati nel sistema (De Marco, 1978)

- I Diagrammi di Flusso dei Dati (DFD) descrivono le operazioni effettuate sui dati e le dipendenze funzionali che si creano in virtù dei flussi di informazione esistenti tra i diversi processi

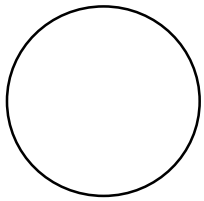
- Utilizzano una notazione grafica che si concentra sull'elaborazione funzionale, la memorizzazione dei dati e il passaggio di dati tra funzioni

- Applicazione sistematica del concetto di ***raffinamento della specifica*** mediante scomposizione gerarchica di un'operazione in un insieme di operazioni più semplici

- Costituiscono un metodo di specifica del software molto intuitivo e generale

DFD: rappresentazione grafica

Agente esterno



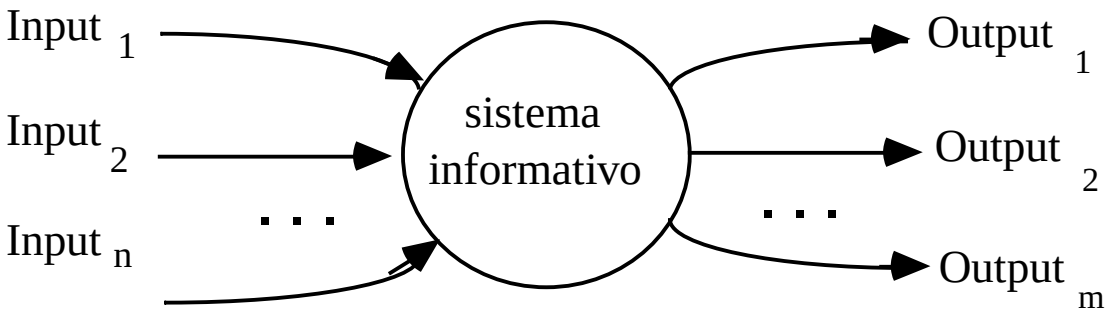
Funzione o processo

Flusso di dati

Deposito permanente di dati

- Gli *agenti esterni* al sistema, rappresentati mediante scatole rettangolari e non sono modellati ulterior-mente
- Funzioni, raffinabili
- Depositi di dati da usare come sorgente o destinazione di informazione permanente
- Flussi di dati, scambiati tra funzioni oppure tra una funzione e una componente di diverso tipo

Diagramma iniziale di un sistema informativo:



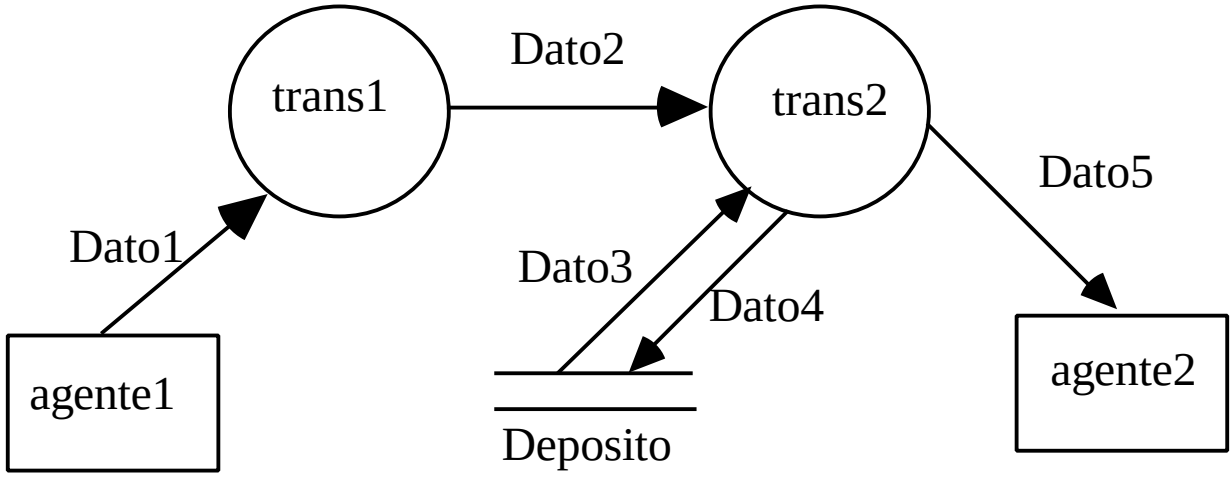
- Specifica del sistema prodotta per successivi raffinamenti e specializzazioni a partire dal diagramma iniziale

- Raffinamento di una funzione, facendole corrispondere un intero DFD (*esplosione della funzione*), che specifica in maggior dettaglio il suo significato in termini di altre funzioni più semplici

- Vincolo di *continuità del flusso informativo*, nel DFD corrispondente ad una funzione devono essere presenti gli stessi flussi netti di informazione in entrata e in uscita dalla funzione che si sta raffinando

- E` possibile introdurre, nel corso di un raffinamento, nuovi agenti o depositi di dati, a patto che non esistano flussi informativi tra questi e le funzioni esterne al diagramma locale

DFD: Esempio



DFD che modella un sistema in cui un agente esterno (persona o dispositivo automatico) chiamato *agente1* fornisce al sistema un dato *Dato1* che viene accettato ed elaborato dalla transizione *trans1*, che produce come risultato *Dato2*, a sua volta ingresso, insieme a *Dato3* proveniente dal deposito di dati *Deposito*, alla funzione *trans2* che produce le informazioni *Dato4* e *Dato5*, rispettivamente memorizzate nel deposito *Deposito* e comunicate all'agente esterno *agente2*

Raffinamenti di un DFD

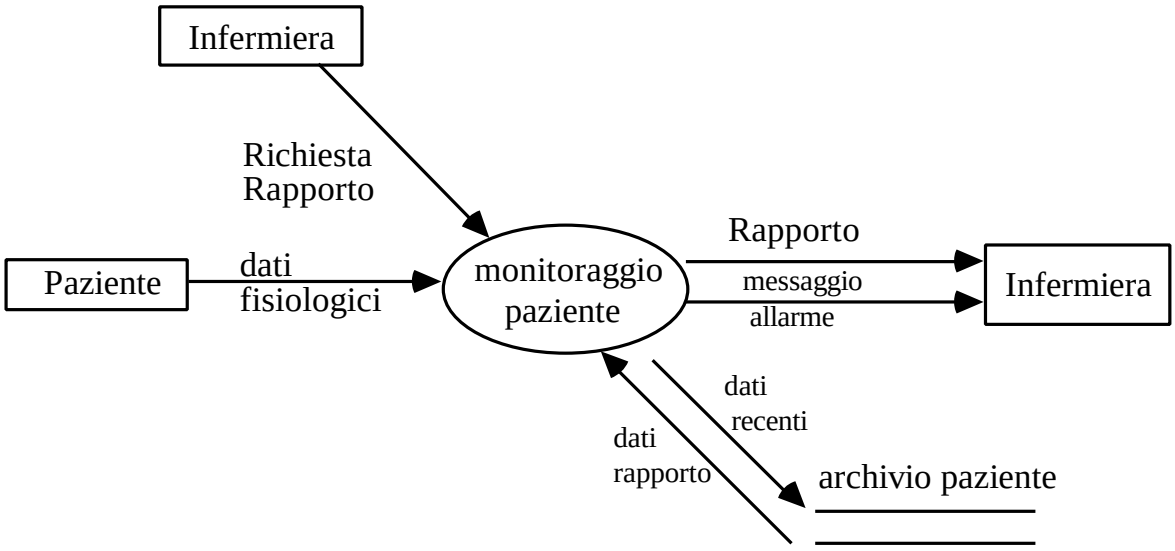
- Il diagramma corrispondente alla funzione *A* mantiene i flussi *I* e *O*, mentre quello per *A2* mantiene in entrata il solo flusso *K* e in uscita i soli flussi *M* ed *N* (vincolo di continuita` rispettato)

DFD: esempio applicativo

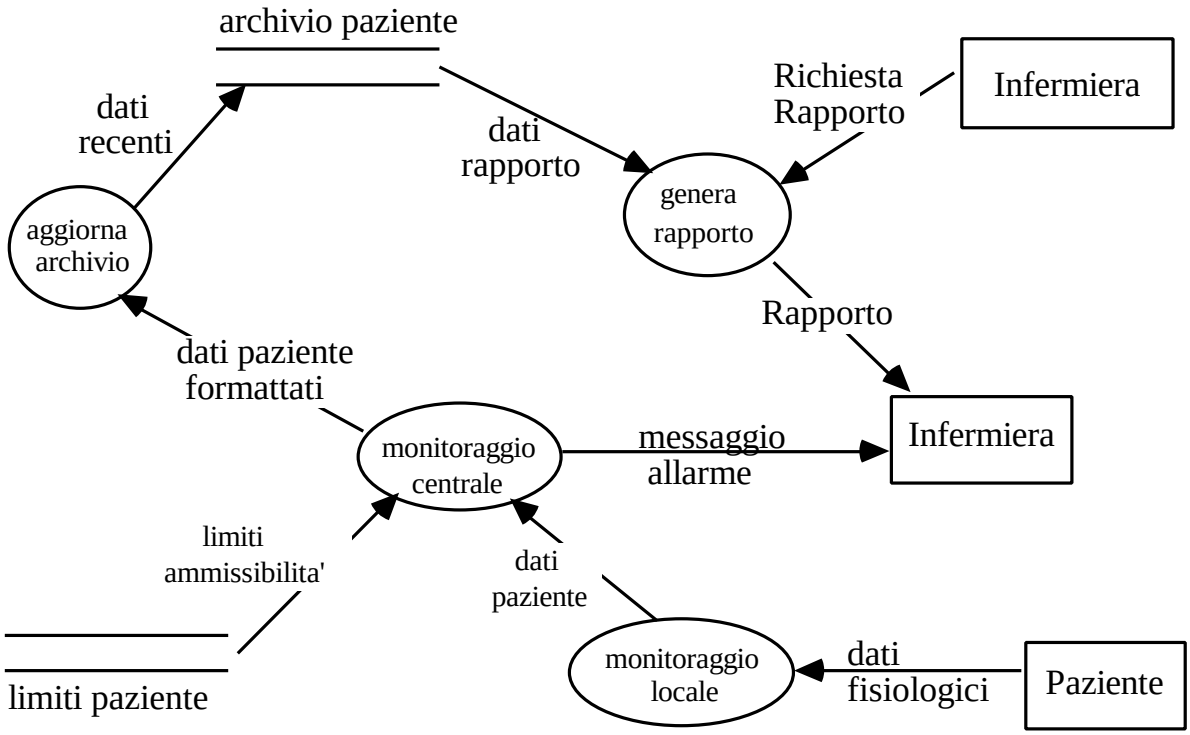
- Specifica di un sistema di monitoraggio di paziente in un ospedale

  I segnali vitali del paziente (temperatura, pressione, polso) sono misurati periodicamente e convertiti in forma manipolabile da programma; essi vengono confrontati coi dati ammissibili per il paziente, memorizzati in forma permanente su di un file, formattati e memorizzati in un file storico.
  Se i dati del paziente sono al di fuori dei limiti prestabiliti viene generato un messaggio di allarme che richiama l'attenzione del personale sanitario.
  Inoltre, un infermiere può occasionalmente, cioè in modo asincrono, richiedere un rapporto sulle condizioni del paziente: in questo caso il rapporto viene generato a partire dall'archivio storico.

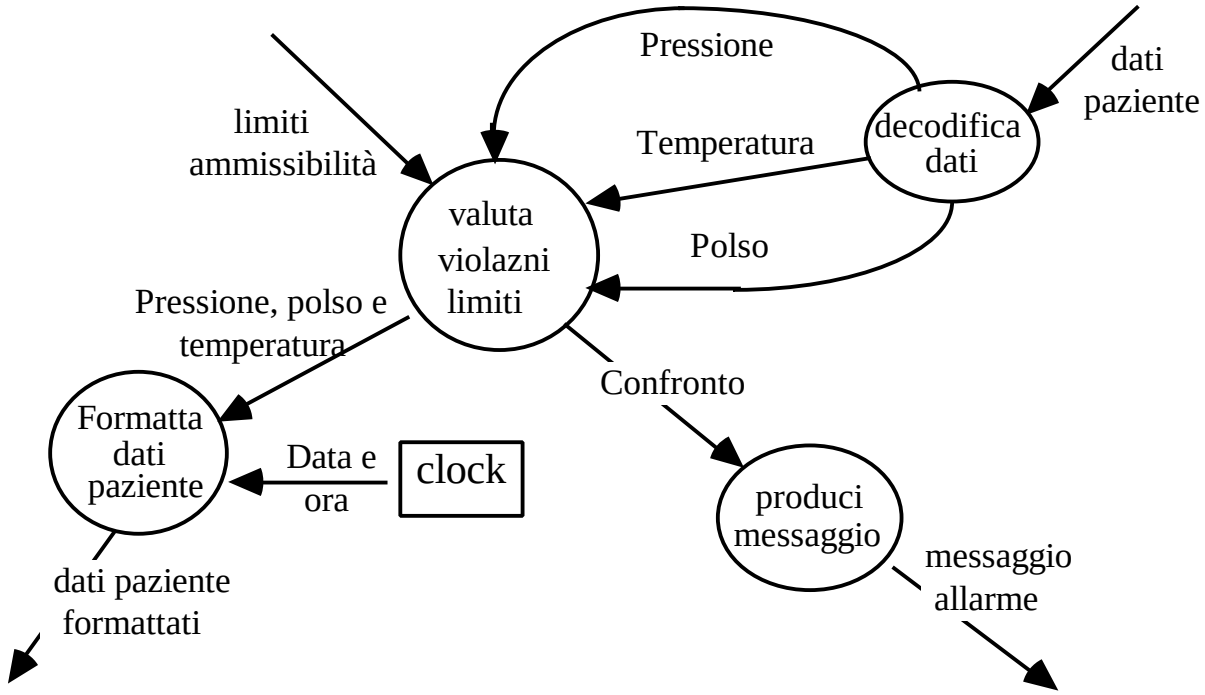
- DFD iniziale, in cui la funzione *monitoraggio paziente* individua l'intero sistema da specificare:



- Primo raffinamento, nel quale vengono enucleate le funzioni *monitoraggio centrale*, *monitoraggio locale*, *aggiorna archivio* e *genera rapporto*



- Ulteriore dettaglio della funzione *monitoraggio centrale.*:



Viene aggiunto l'agente *clock* che fornisce la data e l'ora corrente da associare a un dato insieme di misurazioni da formattare e memorizzare (non sarebbe stato opportuno introdurre tale agente ad un livello di dettaglio più alto)

Alcune regole e suggerimenti per una stesura di DFD:

Nella descrizione di un sistema mediante DFD esso deve essere immaginato in un ipotetico stato stabile e invariante in cui i dati in uscita vengono prodotti a partire da quelli in ingresso (si ignorano le condizioni e le operazioni per l'inizializzazione del sistema, per la sua terminazione e per la gestione di errori e situazioni eccezionali);

Ignorare il flusso di controllo e la sincronizzazione tra i processi (informazione non rilevante in questa fase della specifica);

Individuare entrate e uscite nette del sistema o della porzione che si sta descrivendo: esse vanno evidenziate, per esempio, disegnandole più esterne;

Assegnare ai flussi ed alle funzioni nomi significativi;

Per verificare la correttezza e la consistenza di un DFD, percorrere la sequenza dei flussi informativi sia a partire dagli ingressi verso le uscite sia risalendo dalle uscite fino agli ingressi dai quali esse dipendono e che vengono usati per calcolarle.

Note

- I DFD permettono di strutturare il sistema sia orizzontalmente, effettuando una scomposizione funzionale del sistema nelle sue componenti immediate, sia verticalmente, dal momento che il processo di decomposizione funzionale può essere iterato a qualsiasi livello su ogni attività introdotta, creando così una struttura gerarchica, ad albero, di diagrammi

- I DFD forniscono soltanto informazioni sulle dipendenze funzionali tra i dati, ma sono del tutto imprecisi e ambigui per quel che riguarda gli aspetti di sincronizzazione e controllo

- Non adatti per specifica di interfacce utente

- Diagrammi E-R forniscono un modello concettuale dei dati, con maggiore informazione sul loro significato e sul loro uso

- Integrazione dei DFD con i modelli semantici dei dati

Dizionari dei dati

- Elenco di nomi e descrizioni associate per le entita` usate nel sistema

- Se il nome rappresenta un oggetto composito, si specifica anche una descrizione della composizione

- Rappresenta un meccanismo per la gestione di nomi ed un collegamento tra analisi, progetto e realizzazione

- Deve comprendere tutti i nomi usati nel modello del sistema, nel progetto e nella realizzazione

- Esempio:

  dizionario che descrive la struttura di un dato mediante operatori di selezione, opzionalità, ripetizione e concatenamento tra le parti componenti:

Costrutto	**Notazione**	**Significato**
Definizione	A + B	A è definito come B
Sequenza	A **+** B	A concatenato con B
Selezione	**[** A | B **]**	A oppure B
Ripetizione	**{** A **}**$^{j}_{i}$	A ripetuto min i, max j volte
Opzionalità	**[** A **]**	A oppure nulla

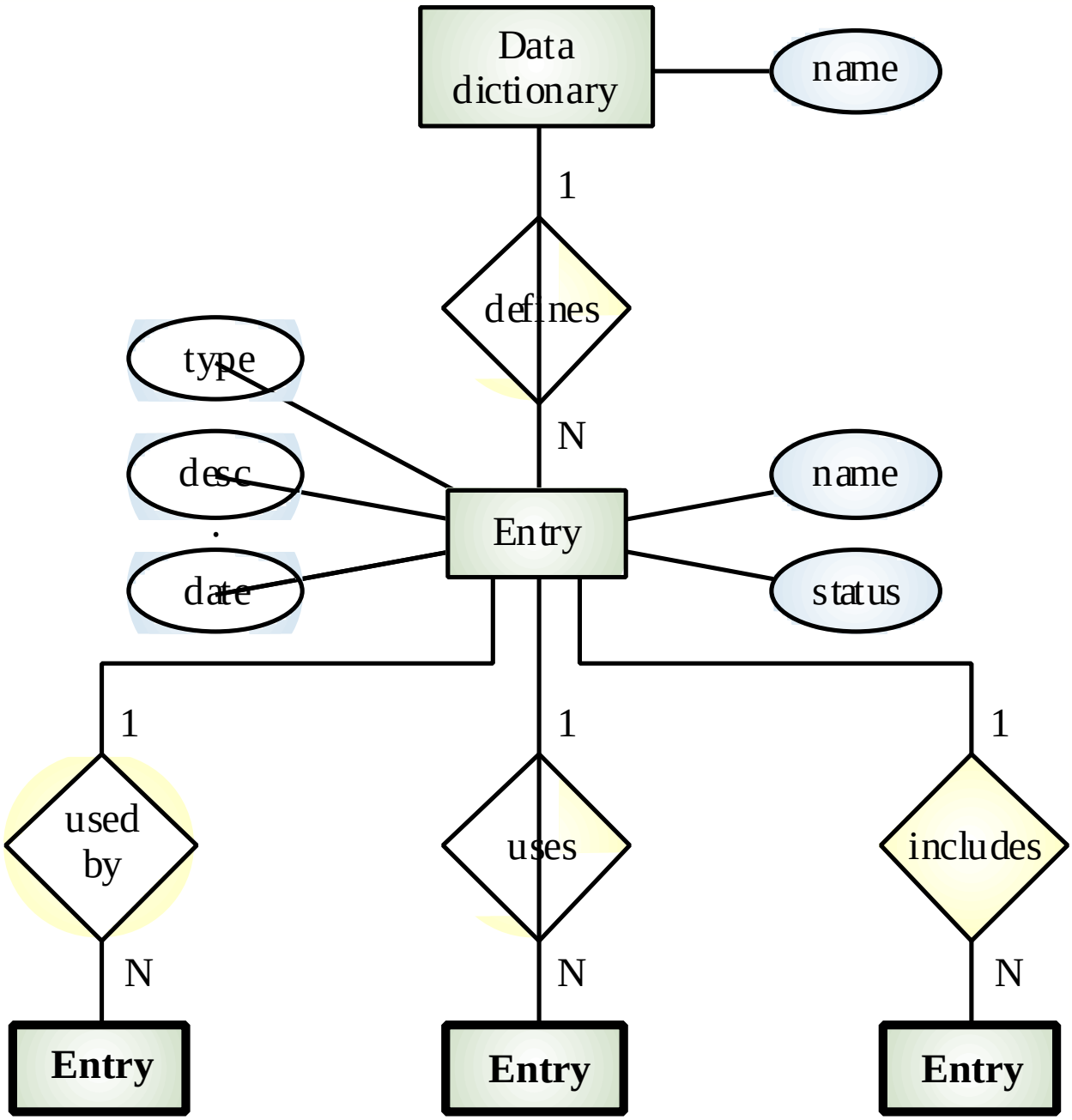
Dizionari dei dati: esempi

- Definizione della struttura e della composizione del numero telefonico in una organizzazione che dispone di un centralino automatico

  Un generico utente del servizio telefonico ha la possibilità di effettuare chiamate locali, chiamate esterne o di parlare col centralinista. Le chiamate locali possono essere in conferenza, con un massimo di cinque partecipanti; le chiamate esterne possono essere urbane o interurbane. La struttura delle possibili opzioni per il numero telefonico è definita allora dalle seguenti dichiarazioni

  numero telefonico + **[** numero locale **|** numero esterno **|** centralino **]**
  centralino + 0
  numero locale + [interno **|** conferenza **]**
  interno + **{** digit **}**$^{4}_{4}$
  digit + [0 | 1 | ... | 9 **]**
  conferenza + **{#** + interno**}**$^{5}_{2}$**+** # **+** #
  numero esterno + 9 **+ [** prefisso **]** + esterno
  esterno + **{** digit **}**$^{8}_{5}$
  prefisso + 0 **+ {** digit **}**$^{2}_{1}$

  In questo caso, espressioni regolari

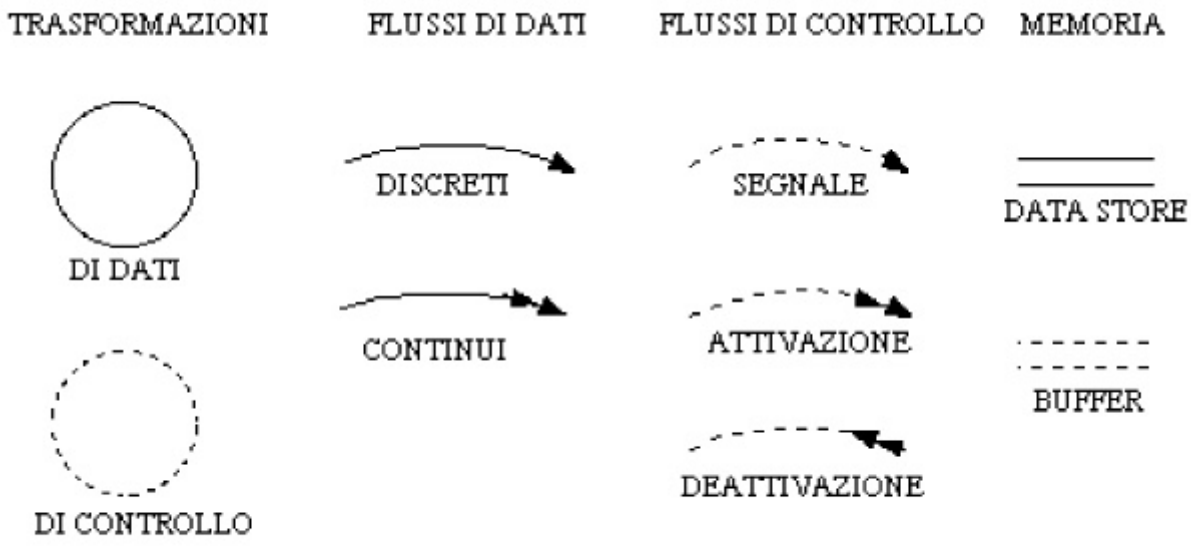

Data dictionary
name
1
defines
N
type
desc
date
Entry
name
status
1
1
1
used by
uses
includes
N
N
N
Entry
Entry
Entry

Name	Description	Type	Date
has_labels	1:N relation between entities of type Node or Link and entities of type Label.	Relation	5.10.93
Label	Holds structured or unstructured information about nodes or links. Labels can be text or can be an icon.	Entity	8.12.93
Link	Represents a relation between design entities represented as nodes. Links are typed and may be named.	Relation	8.12.93
name (label)	Each label has a name which identifies the type of label. The name must be unique within the set of label types used in a design.	Attribute	8.12.93
name (node)	Each node has a name whichmust be unique within a design. The name maybe up to 64 characters long.	Attribute	15.11.93

Transformation Schema

- Estensione dei DFD per rappresentare gli aspetti temporali e di sincronizzazione dei sistemi specificati

- Aggiungono alla descrizione della trasformazione dei dati dei DFD tradizionali la descrizione della gestione del controllo e della sincronizzazione



- Aggiungono funzioni di controllo, flussi di dati di controllo e memorie temporanee (elementi tratteggiati)

Elementi dei TS

- Funzioni di controllo, rappresentate con cerchi dalla circonferenza tratteggiata

- Flussi di controllo, divisi in segnali semplici, che trasmettono il verificarsi di una condizione, e segnali di attivazione e disattivazione esplicita di funzioni

- Flussi di dati, divisi a loro volta in segnali discreti e continui

- Memoria, suddivisa in memoria permanente, chiamata data store, e memoria temporanea (buffer), che viene letta e scritta in modo distruttivo

- I flussi di controllo per l'attivazione e la disattivazione esplicita delle funzioni possono essere emessi solo dalle funzioni di controllo e determinano lo stato di attivazione delle funzioni (di trasformazione dei dati) che li ricevono

- Le funzioni di trasformazione dei dati possono essere attivate da un ingresso attivo (al piu` uno): questo può essere un flusso di controllo di tipo segnale, oppure un dato discreto

- Funzioni di controllo specificate come automi a stati finiti (regole di evoluzione analoghe a Reti di Petri)