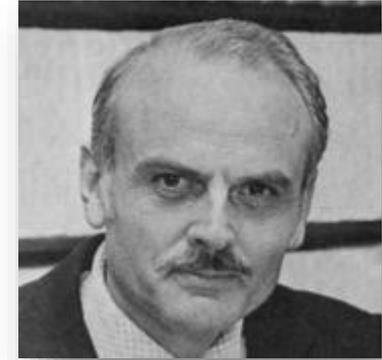


modello relazionale

informatica

- *progettazione logica relazionale:*
 - *conversione* di un diagramma *E/R* in un insieme di *relazioni (tabelle)*, che costituisce lo schema logico relazionale della base dati
- esistono altri modelli logici (*modello gerarchico*, *modello reticolare*) anche se oggi il modello relazionale è sicuramente il *più diffuso*

- introdotto nel **1970** dal matematico inglese ***Edgar Frank Codd***
- lavora in IMB e pubblica “*A Relational Model of Data for Large Shared Data Banks*”
(*un modello relazionale per i dati in grandi basi dati condivise*)
- prime ***implementazioni*** del modello intorno alla fine degli anni '70 (*ritardo dovuto alla **difficile** implementazione del modello matematico*)
- dagli anni '80 ampia ***diffusione*** di DBMS relazionali anche per sistemi di piccole dimensioni



- teoria basata sul concetto di *relazione tra insiemi*
- una relazione R su una sequenza di insiemi $D_1, D_2, D_3 \dots D_N$ (*non necessariamente distinti*) è un *sottoinsieme* finito del prodotto cartesiano $D_1 \times D_2 \times D_3 \dots \times D_N$ che è possibile esprimere con:

$$R \subseteq D_1 \times D_2 \times D_3 \dots \times D_N$$
- N ($N \geq 1$) è detto *grado* della relazione
- gli insiemi D sono detti domini della relazione e ciascuno di essi può essere un insieme composto da *dati di tipo elementare* (*carattere, booleano, numerico ...*)
- a ogni *dominio* è associato un *attributo*, che lo identifica univocamente all'interno della relazione

- il *nome* della relazione e la *lista* dei suoi *attributi* rappresenta il *significato intensionale* e costituisce lo *schema della relazione*

`<NomeRelazione>`

`(<Attributo1>:<Tipo1>, <Attributo2>:<Tipo2>,
<Attributo3>:<Tipo3> ... <AttributoN>:<TipoN>)`

`Computer (CodiceProgressivo: Intero, Marca: Stringa(40),
Modello: Stringa(40), NumeroDiSerie: Stringa(20))`

- gli elementi di R sono detti ennuple (*tuple*)

- esempio:
 - relazione di *nome computer* di *grado 4*
 - *cardinalità 3* (*numero di tuple*)
 - *attributi: CodiceProgressivo, Marca, Modello*
- *valori* attributi (*estensione*)

Rappresentazione come elenco

```
Computer(CodiceProgressivo, Marca, Modello, NumeroDiSerie) = {
  (1, HELL, A/11726, HLL199283772),
  (2, HELL, A/11799, HLL199283663),
  (5, Acid, ACID10, A1827676)
}
```



Rappresentazione come tabella

Grado 4

Attributo

Cardinalità 3

| CODICE_PROGRESSIVO | MARCA | MODELLO | NUMERO_DI_SERIE |
|---------------------------|--------------|----------------|------------------------|
| 1 | HELL | A/11726 | HLL199283772 |
| 2 | HELL | A/11799 | HLL199283773 |
| 5 | Acid | ACID10 | A1827676 |

Valore Attributo

Rappresentazione come insieme

(1, HELL, A/11726, HLL199283772)

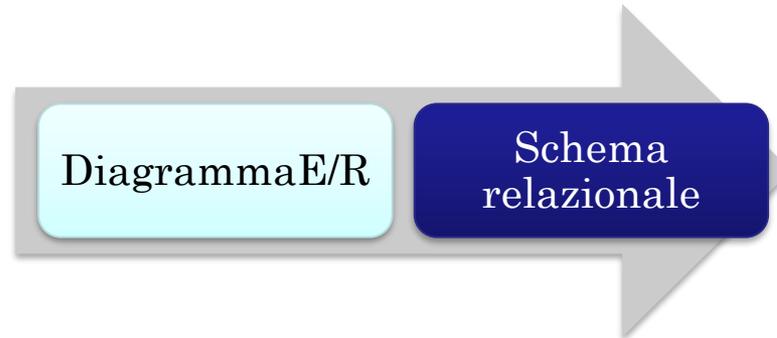
(1, HELL, A/11799, HLL199283773)

(5, Acid, ACID10, A1827676)

- *nome* relazione **Studente**
- *grado* della relazione **3**
(*Libretto, Nome, Data_Nascita*)
- *cardinalità* della relazione = **1200**
(*numero di tuple = numero di studenti*)
- Libretto è *campo chiave*

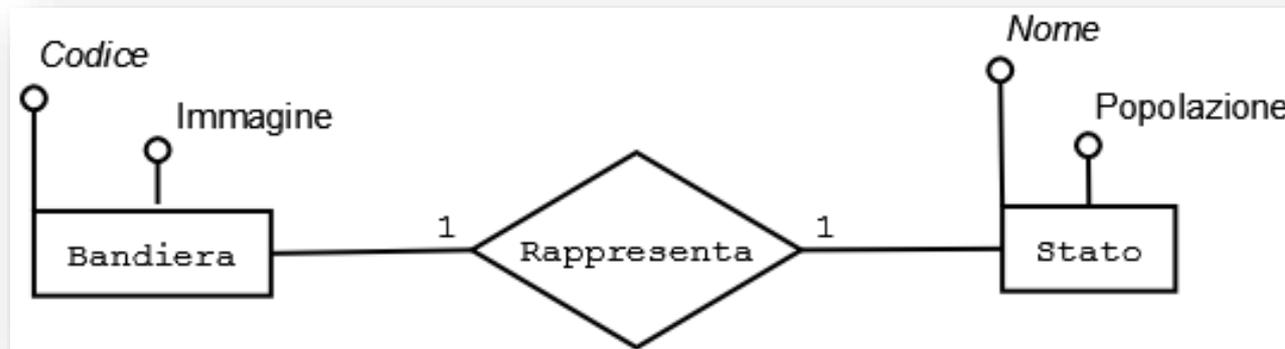
| Studente | | | |
|----------|-----------------|---------|--------------|
| | <u>Libretto</u> | Nome | Data_Nascita |
| | 34 | Verdi | 15/11/1990 |
| | 123 | Rossi | 10/02/1990 |
| | 210 | Bianchi | 27/04/1989 |
| | 45 | Neri | 12/12/1988 |

- *passaggio* da uno *schema concettuale* (diagramma E/R) a uno *schema relazionale* (insieme di relazioni) *equivalente*



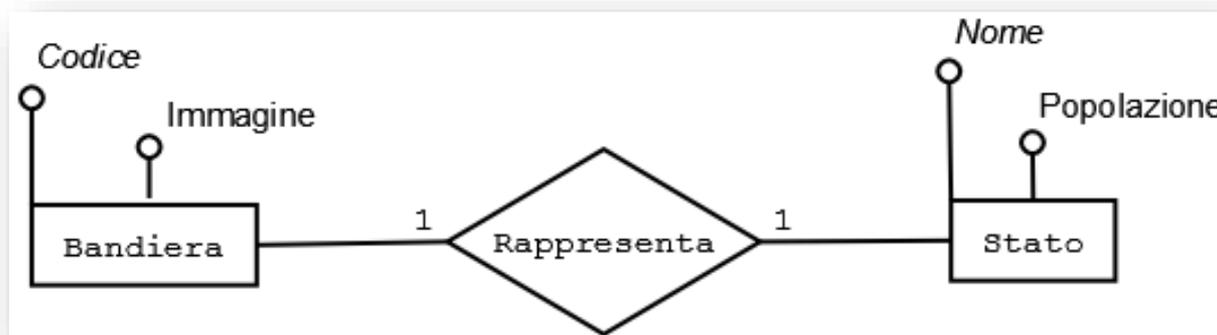
- ogni *entità* viene rappresentata da una *relazione*
- l'attributo *chiave primaria* dell'entità diventa attributo chiave primaria della relazione e viene rappresentato con carattere sottolineato
- gli attributi ~~composti~~ vengono svolti nella relazione, dove saranno presenti gli attributi costitutivi dell'attributo composto
- gli attributi ~~multipli~~ richiedono la *normalizzazione* (*discussa più avanti*)

- le *associazioni 1:1* vengono trasformate in un'*unica relazione*



| BandieraStato | | | | |
|---------------|--------|-------------|-----------|-------------|
| | Codice | Immagine | NomeStato | Popolazione |
| | 12 | Francia.jpg | Francia | 543210000 |
| | 20 | Italia.jpg | Italia | 564534000 |
| | 3 | Olanda.jpg | Olanda | 142100000 |

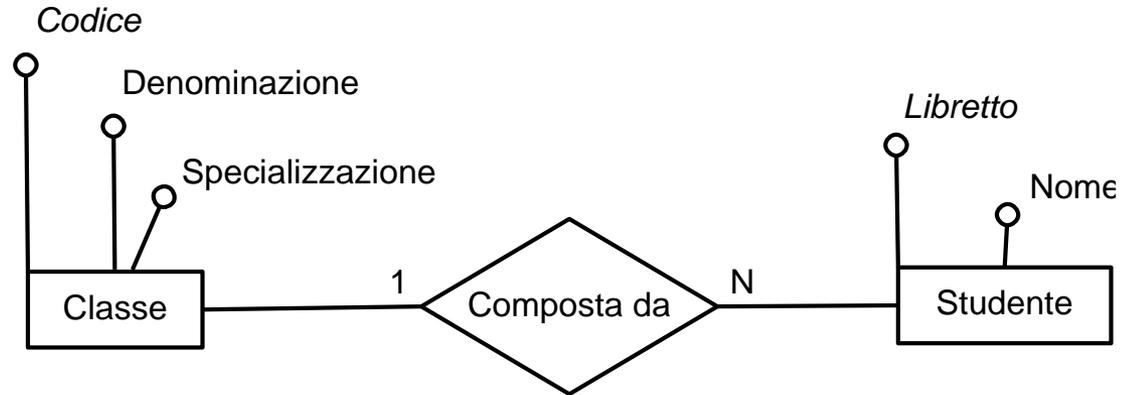
- si opera come nel caso di associazioni 1:n



| Bandiera | |
|---------------|-------------|
| <u>Codice</u> | Immagine |
| 3 | Olanda.jpg |
| 12 | Francia.jpg |
| 20 | Italia.jpg |

| Stato | | |
|------------------|-------------|----------------|
| <u>NomeStato</u> | Popolazione | CodiceBandiera |
| Olanda | 142100000 | 3 |
| Francia | 543210000 | 12 |
| Italia | 564534000 | 20 |

- le associazioni *binarie 1:N* tra le entità **A** e **B** sono tradotte in *due relazioni*:
 - relazione **R_a** che ha gli *attributi di A*
 - relazione **R_b** che ha gli *attributi di B* e gli attributi *chiave K_a di A (chiave esterna)*



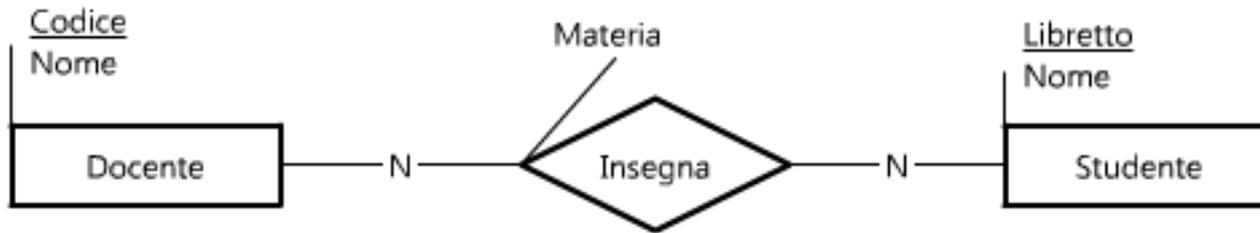
| Classe | | |
|---------------|-----------------|---------------------------|
| <u>Codice</u> | Denominazione | Specializzazione |
| 3AINF | 3 A Informatica | Informatica |
| 4CELE | 4 C Elettronica | Elettronica e Automazione |
| 5BINF | 5 B Informatica | Informatica |

| Studente | | |
|-----------------|---------|--------------|
| <u>Libretto</u> | Nome | CodiceClasse |
| 123 | Rossi | 5BINF |
| 234 | Bianchi | 3AINF |
| 21 | Verdi | 5BINF |

chiave esterna



- per tradurre nel modello relazionale una associazione ***N:N tra due entità*** è necessario introdurre ***tre relazioni***
- la relazione N:N tra le entità **A** e **B** si rappresenta con:
 - una relazione **R_A** che ha gli attributi di **A**
 - una relazione **R_B** che ha gli attributi di **B**
 - una relazione **R_S** con gli attributi chiave **K_A** di **R_A** e gli attributi chiave **K_B** di **R_B**

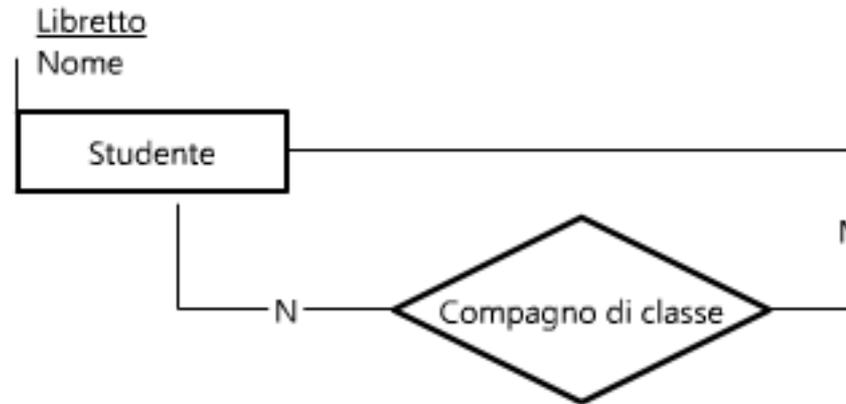


Made with lovelycharts.com

| <u>Codice</u> | Nome |
|---------------|---------|
| 001 | Rossi |
| 002 | Verdi |
| 003 | Bianchi |

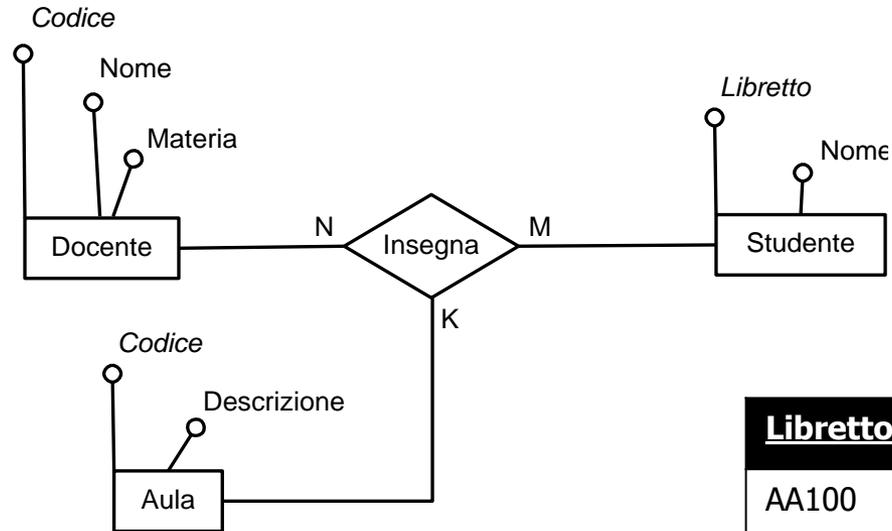
| <u>Kdoc</u> | <u>KStud</u> | Materia |
|-------------|--------------|-------------|
| 001 | AA100 | Informatica |
| 001 | BB100 | Informatica |
| 003 | AA100 | Storia |
| 002 | AA100 | Inglese |

| <u>Libretto</u> | Nome |
|-----------------|----------|
| AA100 | Aldo |
| AA101 | Giovanni |
| BB100 | Giacomo |



| <u>Libretto</u> | Nome |
|-----------------|----------|
| AA100 | Aldo |
| AA101 | Giovanni |
| BB100 | Giacomo |
| BB101 | Jack |

| KstudA | KstudB |
|--------|--------|
| AA100 | AA101 |
| BB100 | BB101 |



| <u>Codice</u> | Nome | Materia |
|---------------|---------|-----------------|
| 001 | Rossi | Informatic a |
| 002 | Verdi | Storia |
| 003 | Bianchi | Inglese |

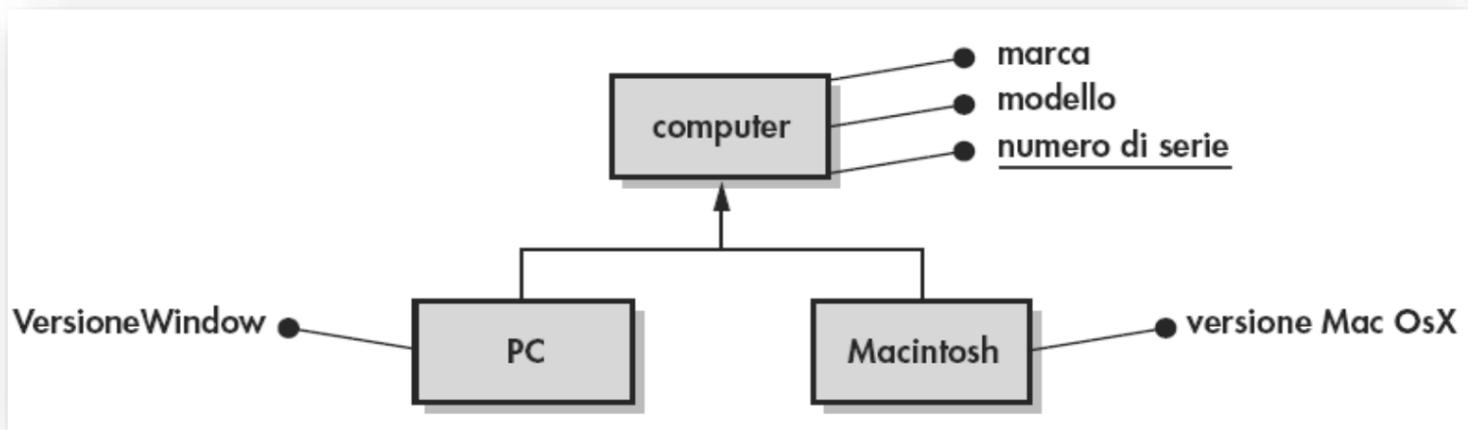
| <u>Libretto</u> | Nome |
|-----------------|----------|
| AA100 | Aldo |
| AA101 | Giovanni |
| BB100 | Giacomo |

| <u>Kdoc</u> | <u>Kstud</u> | <u>Kaula</u> |
|-------------|--------------|--------------|
| 001 | AA100 | A01 |
| 001 | BB100 | LInf |
| 003 | AA100 | A02 |
| 002 | AA100 | A02 |

| <u>Codice</u> | Descrizione |
|---------------|-------------------|
| A01 | A1 nuovo |
| A02 | A2 nuovo |
| LInf | L. Informatica |

- gli attributi delle associazioni **1:1** vanno inseriti in **una qualunque** delle relazioni risultanti (o nell'unica relazione ottenuta)
- gli attributi delle associazioni **1:N** vanno inseriti nell'associazione che conterrà la **chiave esterna**
- gli attributi delle associazioni **N:N** vanno inseriti nella **relazione** che rappresenta l'**associazione**

- padre P e figlie $F_1, F_2, F_3 \dots F_N$
- possibili tre approcci:
 - *accorpamento delle figlie nel padre*
 - *inclusione del padre nelle figlie*
 - *sostituzione della generalizzazione con associazioni 1:1*
- nel primo caso le figlie vengono eliminate dal modello e i loro attributi e associazioni vengono riportati nell'entità padre
 - viene aggiunto un attributo per indicare il tipo di tupla che contiene, per capire a quale figlia fa riferimento



- si ottiene un'unica relazione:

Computer (**Marca**, **Modello**, **NumeroDiSerie**, **Tipo**,
VersioneWindows, **VersioneMacOSX**)

- aggiunto l'attributo **Tipo**, che consente di distinguere se la tupla è una istanze di **PC** o di **Macintosh**
- in **Computer** sono stati *aggiunti* gli *attributi* presenti nelle entità figlie

- *inclusione del padre nelle figlie:*
 - eliminazione del padre P e riposizionamento degli attributi e associazioni nelle entità figlie
 - è attuabile solo nel caso di *generalizzazioni totali*

PC (Marca, Modello, NumeroDiSerie, VersioneWindows)

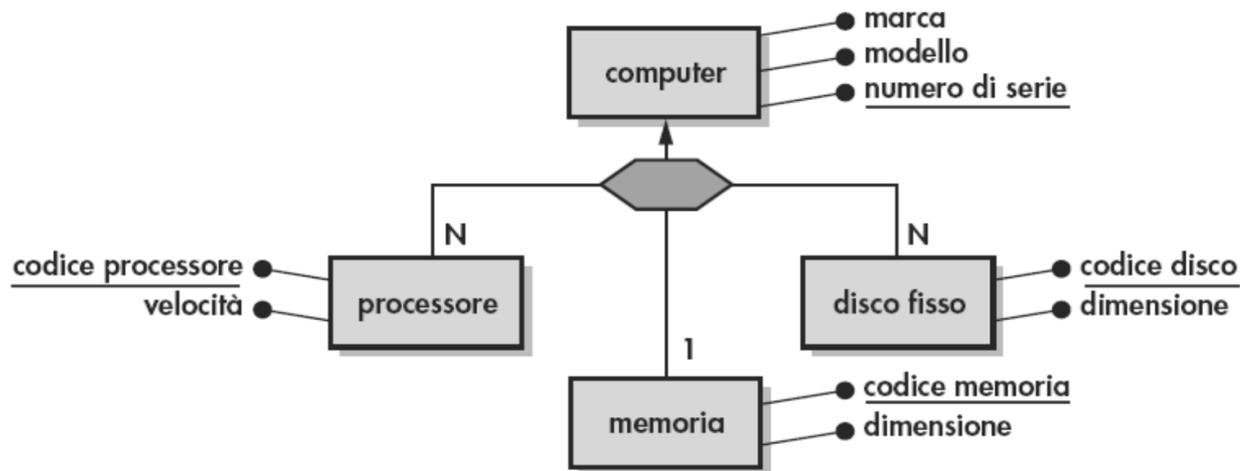
Macintosh (Marca, Modello, NumeroDiSerie, VersioneMacOSX)

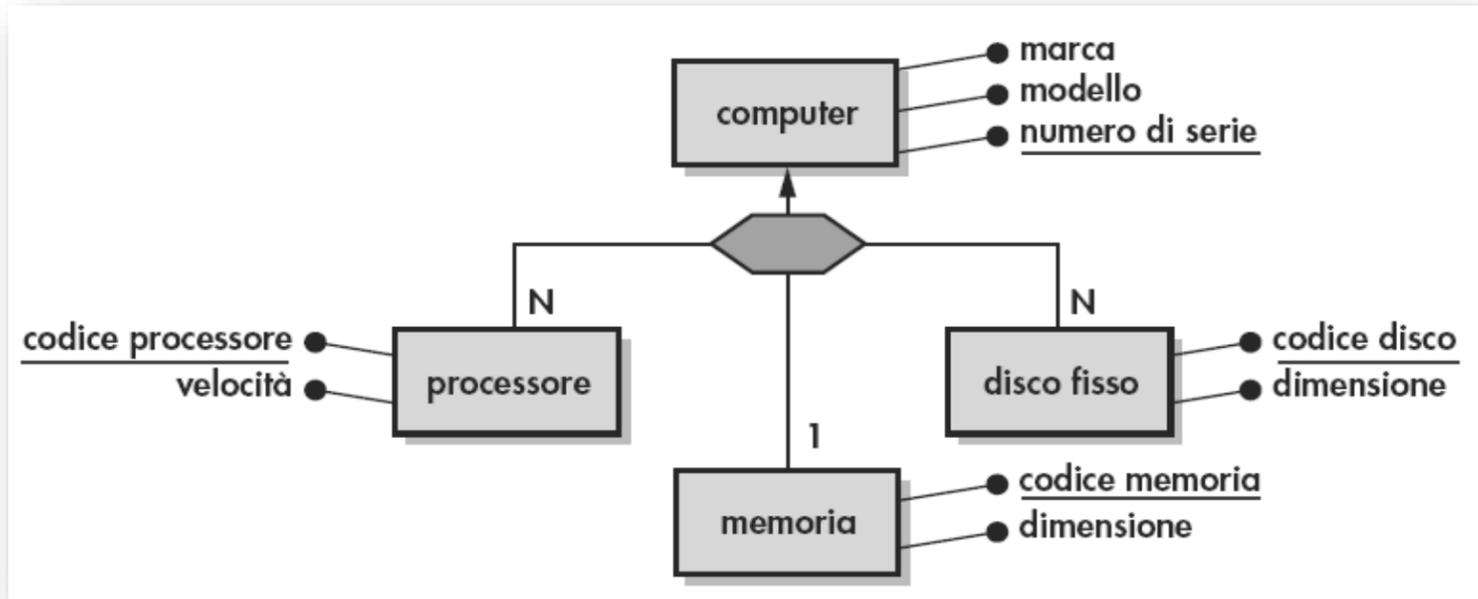
ISA

(diventa associazione 1:1)

- relazioni di tipo 1:1, una per ciascuna entità figlia
- non ci sono accorpamenti di entità
- le figlie vengono identificate utilizzando la **chiave dell'entità padre** come chiave **primaria** ed **esterna** delle figlie

- l'associazione per *aggregazione* si traduce utilizzando una relazione per ciascuna entità componente e una per l'entità aggregata





Computer (Marca, Modello, NumeroDiSerie)

Processore (CodiceProcessore, Velocità, NumeroSerieComputer)

Memoria (CodiceMemoria, Dimensione, NumeroSerieComputer)

DiscoFisso (CodiceDisco, Dimensione, NumeroSerieComputer)

