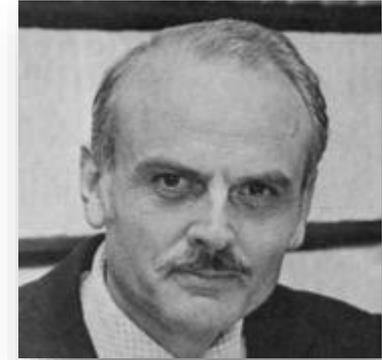


# modello relazionale

programmazione per basi di dati

- *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*

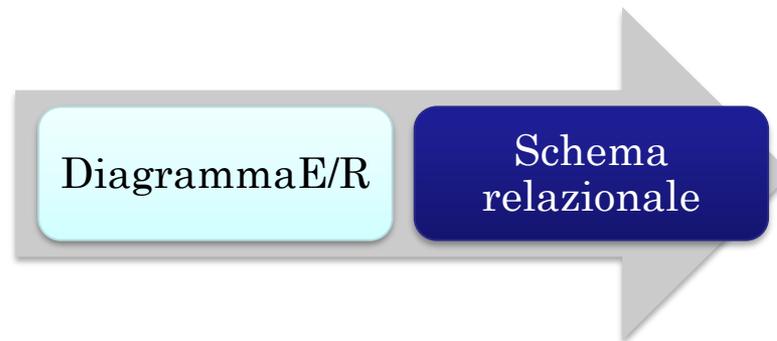
- introdotto nel **1970** dal matematico inglese ***Edgar Frank Codd***
- lavora in IMB e pubblica *“A Relational Model of Data for Large Shared Data Banks”* (un modello relazionale per i dati in grandi basi dati condivise)
- prime ***implementazioni*** del modello intorno alla fine degli anni '70 (ritardo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



- **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**

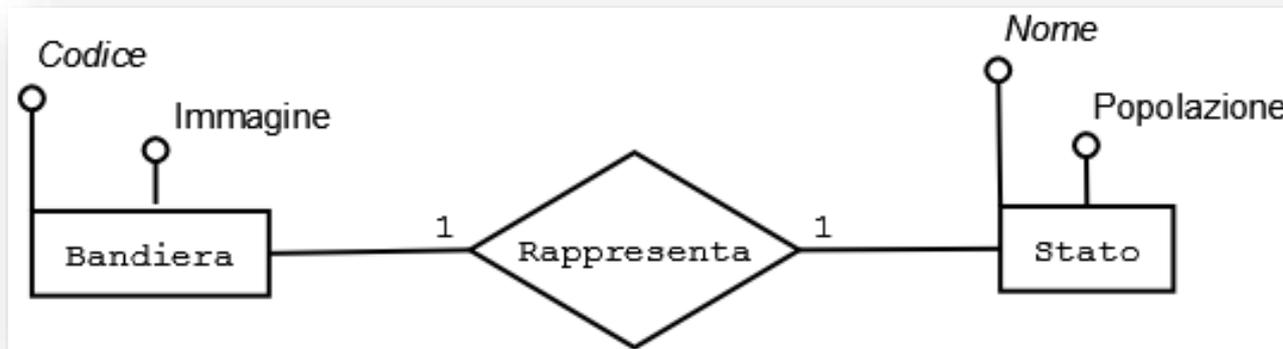
Studente			
	<u>Libretto</u>	Nome	Data_Nascita
	34	Verdi	15/11/1990
	123	Rossi	10/02/1990
	210	Bianchi	27/04/1989
	45	Neri	12/12/1988

- *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*

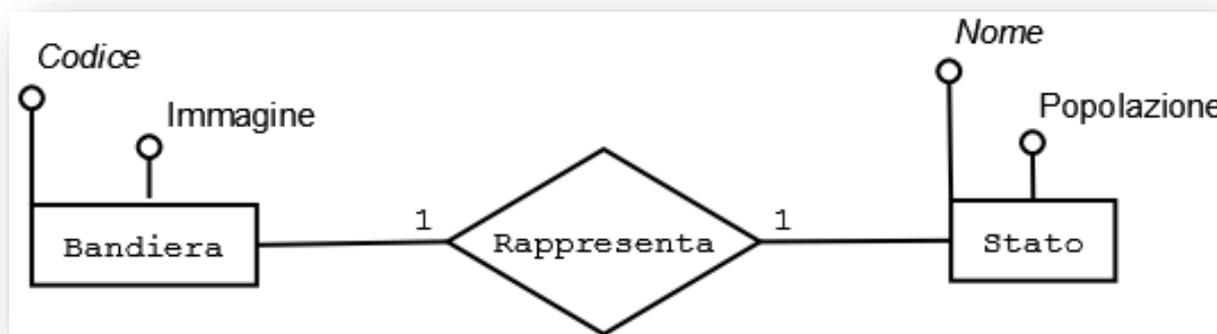


BandieraStato				
	<u>Codice</u>	Immagine	NomeStato	Popolazione
	12	Francia.jpg	Francia	543210000
	20	Italia.jpg	Italia	564534000
	3	Olanda.jpg	Olanda	142100000

# rappresentazione alternativa di associazioni 1:1

A. Ferrari

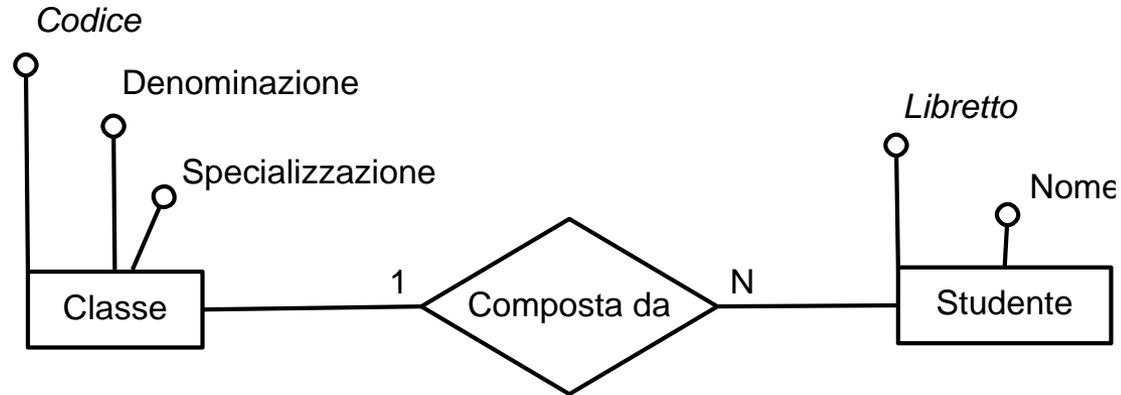
- si opera come nel caso di associazioni 1:n



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 **Ra** che ha gli *attributi di A*
  - relazione **Rb** che ha gli *attributi di B* e gli attributi *chiave Ka 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

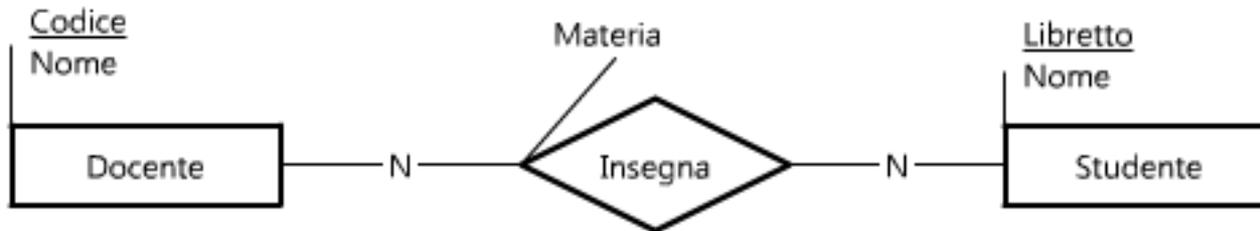
Studente		
<u>Libretto</u>	Nome	CodiceClasse
123	Rossi	5BINF
234	Bianchi	3AINF
21	Verdi	5BINF

*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  $R_B$

# Esempio associazione N:N

A. Ferrari

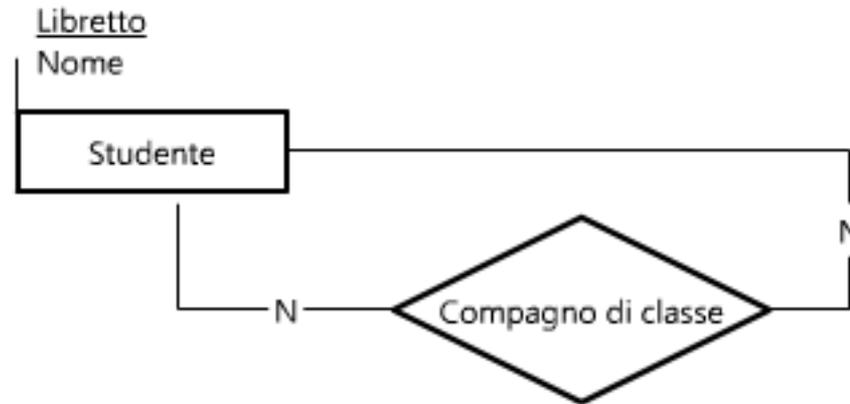


Made with [lovelycharts.com](https://lovelycharts.com)

<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

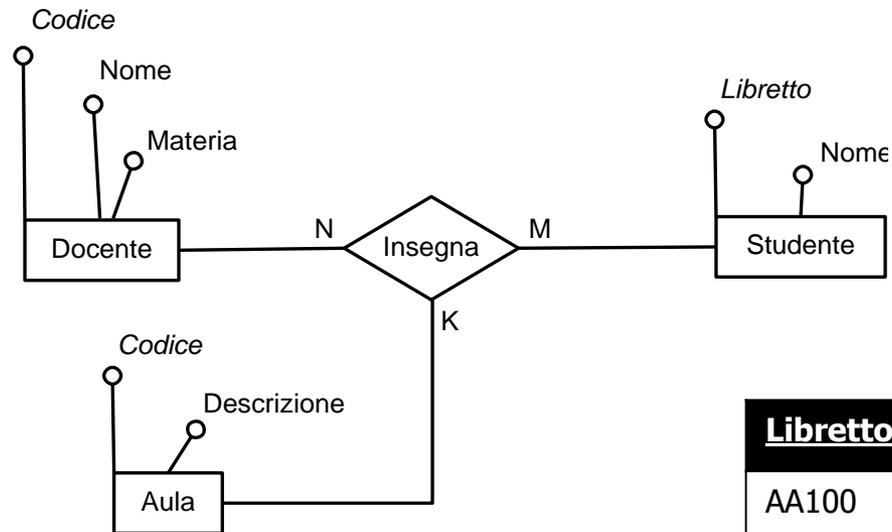


<u>Libretto</u>	Nome
AA100	Aldo
AA101	Giovanni
BB100	Giacomo
BB101	Jack

KstudA	KstudB
AA100	AA101
BB100	BB101

# associazione ternaria

A. Ferrari



<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

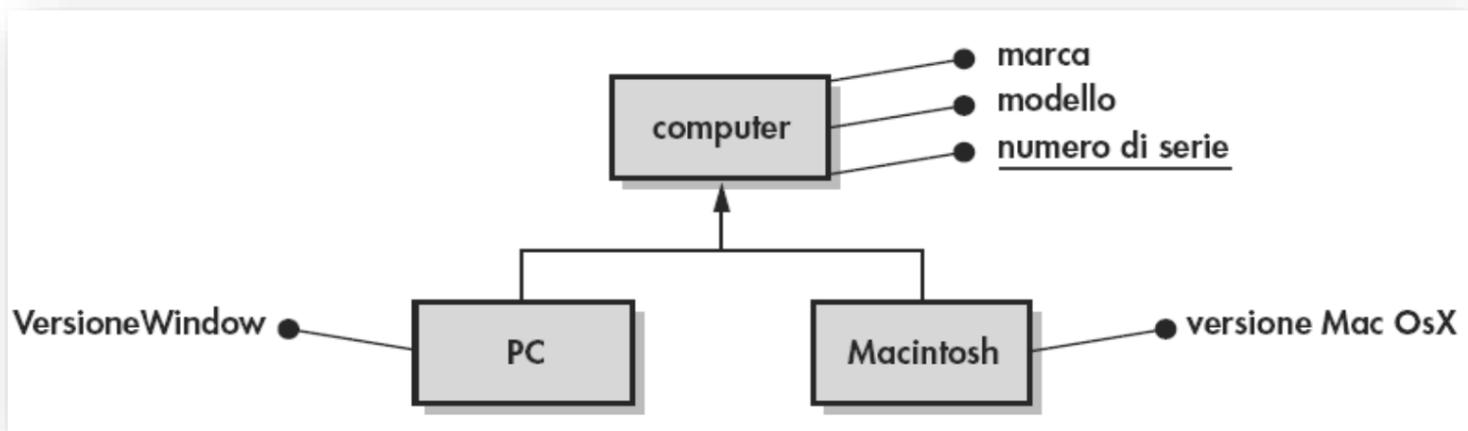
A. Ferrari

- 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***

# associazioni di generalizzazione (ISA)

A. Ferrari

- 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

# ISA (inclusione del padre nelle figlie)

A. Ferrari

- *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)

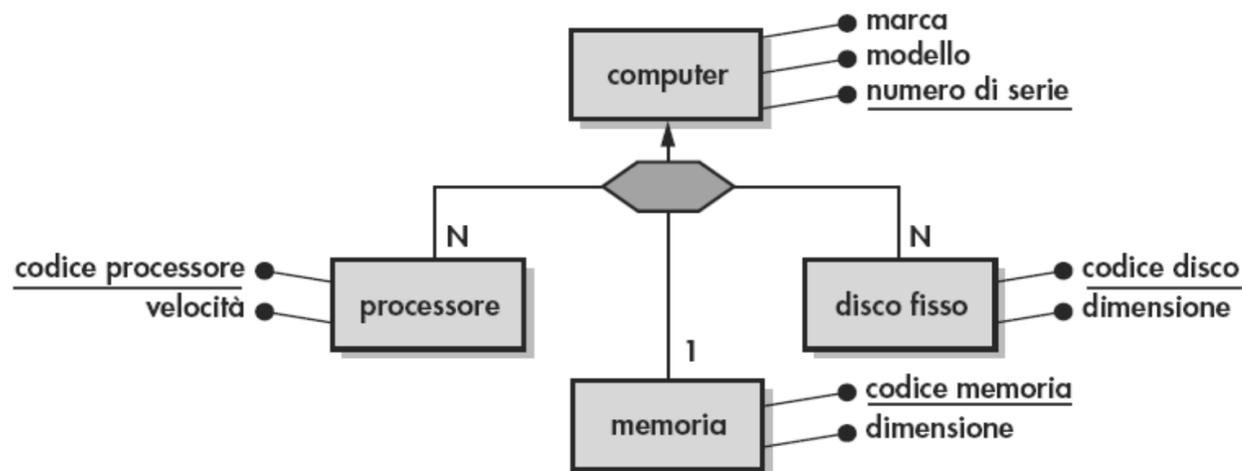
A. Ferrari

- 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

# associazioni di aggregazione (HASA)

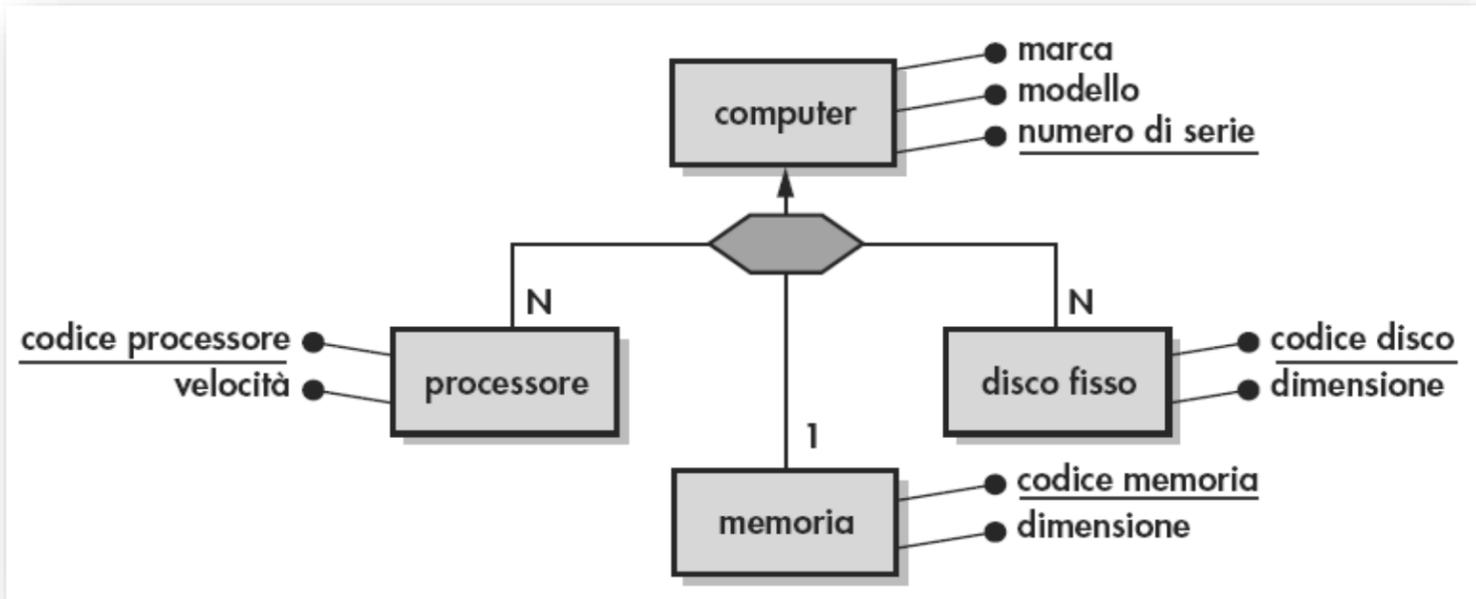
A. Ferrari

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



# HASA (esempio)

A. Ferrari



**Computer**(Marca, Modello, NumeroDiSerie)

**Processore**(CodiceProcessore, Velocità, NumeroSerieComputer)

**Memoria**(CodiceMemoria, Dimensione, NumeroSerieComputer)

**DiscoFisso**(CodiceDisco, Dimensione, NumeroSerieComputer)

# esempio (tennis)

A. Ferrari

