

Associazione Banca del Tempo

Analisi del problema

Le entità che si possono individuare nel problema sono:

- *Categoria*, per le diverse tipologie di prestazione
- *Prestazione*, per il dettaglio delle singole prestazioni di ciascuna categoria
- *Socio*, per le persone che partecipano all'associazione
- *Zona*, per le aree geografiche in cui è suddiviso il territorio di riferimento della BdT

Gli attributi di *Categoria* sono il codice e la descrizione.

Gli attributi di *Prestazione* sono il codice e la descrizione.

Gli attributi di *Socio* sono codice, cognome, nome, indirizzo, telefono.

Gli attributi di *Zona* sono codice, descrizione, mappa grafica.

Tra l'entità *Categoria* e l'entità *Prestazione* esiste un'associazione uno a molti, perché ogni categoria comprende diverse prestazioni, mentre ogni prestazione è classificabile in una sola categoria.

Tra l'entità *Prestazione* e l'entità *Socio* si possono stabilire due associazioni molti a molti:

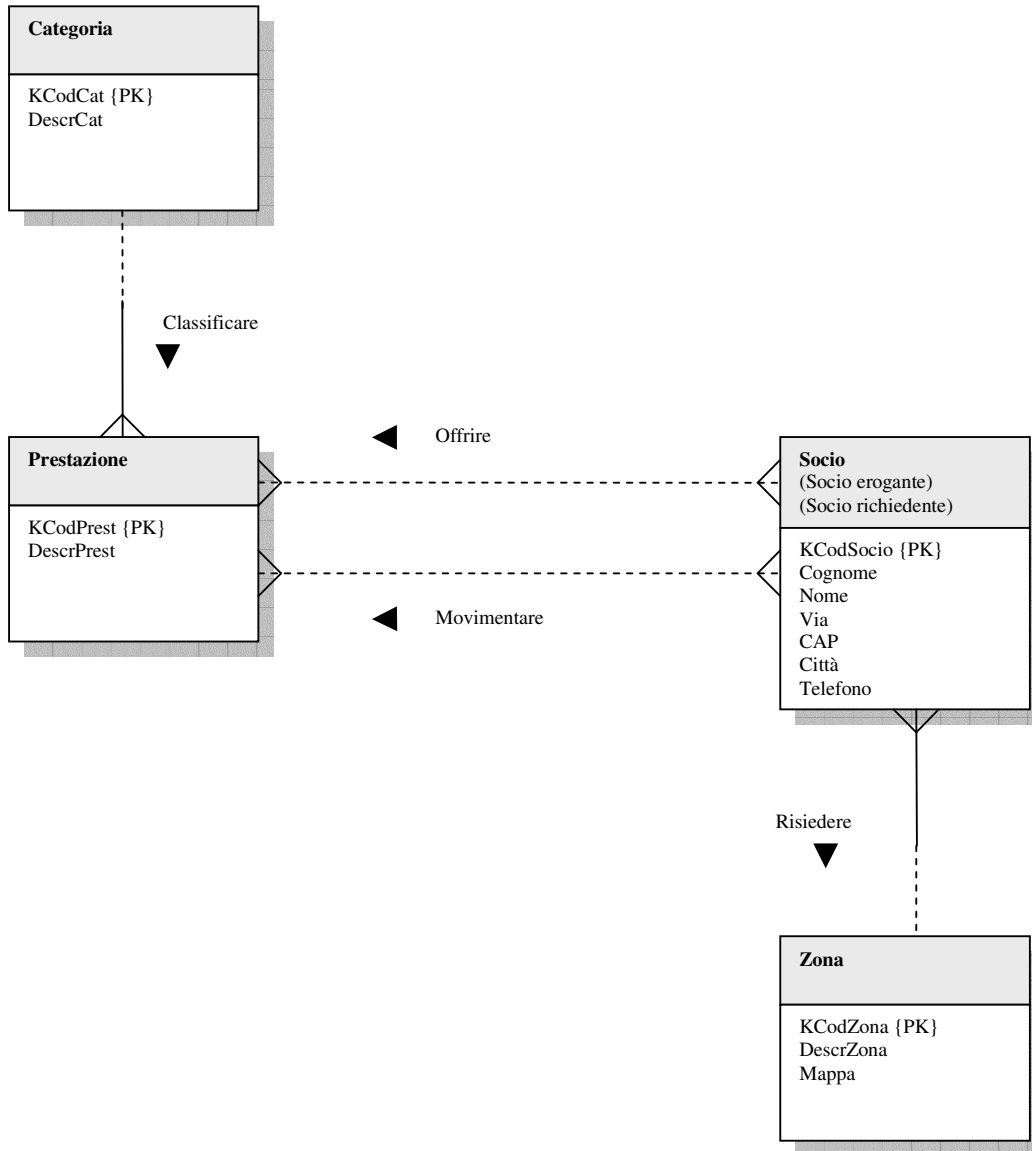
- la prima per rappresentare la disponibilità dei soci a svolgere le prestazioni
- la seconda per le prestazioni effettivamente erogate e ricevute (movimenti).

Nulla impedisce infatti a un socio di mettere a disposizione il proprio tempo per diverse prestazioni; le prestazioni erogate e ricevute inoltre possono riguardare diversi tipi di servizi in giornate diverse.

Tra l'entità *Zona* e l'entità *Socio* esiste un'associazione uno a molti perché ogni zona comprende diversi soci e un socio abita in una sola zona.

Sulla base dell'analisi effettuata si può disegnare il **modello E/R**.

Si osservi che l'entità *Socio* entra come socio generico nell'associazione *Offrire* e in modo doppio (erogante e richiedente) nell'associazione *Movimentare*. Infatti ogni movimento deve contenere il riferimento al socio che eroga la prestazione e al socio che riceve la prestazione. La parola "movimentare" sta ad indicare che l'associazione rappresenta sia l'azione dell'erogare che quella del ricevere.



Il modello E/R viene verificato utilizzando le **regole di lettura**:

Ogni categoria può classificare una o più prestazioni, ogni prestazione deve essere classificata in una sola categoria.

Ogni socio può offrire una o più prestazioni, ogni prestazione può essere offerta da uno o più soci.

Ogni socio può movimentare (come erogante/richiedente) una o più prestazioni in giorni diversi, ogni prestazione può essere movimentata (con erogazioni/ricevute) da uno o più soci.

Ogni zona può essere la residenza di uno o più soci, ogni socio deve risiedere in una sola zona.

Utilizzando le regole di derivazione, si passa dal modello concettuale al modello logico (chiavi primarie sottolineate, chiavi esterne in corsivo).

Tabelle

Categorie (KCodCat, DescrCat)

Zone (KCodZona, DescrZona, Mappa)

Prestazioni (KCodPrest, DescrPrest, *CodCat*)

Soci (KCodSocio, Cognome, Nome, Via, CAP, Città, Telefono, *CodZona*)

Offerte (ID, *CodSocio*, *CodPrest*)

Movimenti (ID, DataMov, Ore, *CSE*, *CSR*, *CodPrest*)

(CSE indica il codice socio erogante, CSR il codice socio ricevente)

Interrogazioni

a.

```
Select CSE As SocioErogante, Sum(Ore) As OreErogate Into Temp1
From Movimenti
Group by CSE;
```

```
Select CSR As SocioRicevente, Sum(Ore) As OreRicevute Into Temp2
From Movimenti
Group by CSR;
```

```
Select Cognome, Nome, Telefono
From Soci, Temp1, Temp2
Where Soci.KCodSocio = Temp1.SocioErogante
And Soci.KCodSocio = Temp2.SocioRicevente
And OreRicevute > OreErogate;
```

b.

```
Select Mappa
From Zone, Soci
Where Zone.KCodZona = Soci.CodZona
And Soci.KCodSocio = [quale codice del socio richiedente];
```

```
Select Cognome, Nome, Via, CAP, Città, Telefono
From Soci, Zone, Offerte
Where Soci.KCodSocio = Offerte.CodSocio
And Zone.KCodZona = Soci.CodZona
And Offerte.CodPrest = [quale codice prestazione]
And KCodSocio <> [quale codice del socio richiedente]
And Soci.CodZona = (Select CodZona
                    From Soci
                    Where Soci.KCodSocio = [quale codice del socio richiedente]);
```

c.

```
Select Soci.*
From Soci, Offerte, Prestazioni
Where Soci.KCodSocio = Offerte.CodSocio
And Prestazioni.KCodPrest = Offerte.CodPrest
And Prestazioni.DescrPrest = 'Segreteria'
And KCodSocio IN (Select CodSocio
                  From Offerte, Prestazioni
                  Where Offerte.CodPrest = Prestazioni.KCodPrest
                  And Prestazioni.DescrPrest <> 'Segreteria');
```

d.

```
SELECT DescrPrest, Sum(Ore) AS OreErogate
FROM Prestazioni, Movimenti
WHERE Prestazioni.KCodPrest = Movimenti.CodPrest
GROUP BY DescrPrest
Order By Sum(Ore) DESC;
```