

# Fondamenti di Basi di Dati

## Esercitazione 5a - Progettazione logica

### Gioco di società

Eseguire la progettazione logica della base di dati dell'Esercitazione 4a, di cui riportiamo per comodità le specifiche e lo schema concettuale. Inoltre presentare un esempio di istanza di base di dati, popolando le tabelle con i dati di esempio richiesti in questo testo. Implementare la base di dati in SQLiteStudio.

Si vuole rappresentare lo stato attuale di gioco in un gioco di società, con le seguenti informazioni:

- I **Continenti**, ognuno con un nome e un numero di armate supplementari a cui esso dà diritto.
- I **Territori**, ognuno con un nome, e un Continente di appartenenza; ogni Territorio confina con altri Territori: si vogliono rappresentare tali confini.
- Le **carte**. Esistono due di due tipi di carte:
  - le **carte ordinarie**, ciascuna delle quali mostra un Territorio e un'Arma (quest'ultima può essere Fanteria, Artiglieria, o Cavalleria);
  - le **carte jolly**.
- I **giocatori**, con nome del giocatore, il colore delle armate, l'ordine di gioco nell'ambito di un turno, le carte che possiede e i Territori che occupa. Ogni Territorio può essere occupato da un solo giocatore, con un certo numero di armate: si vuole rappresentare, per ogni Territorio, il giocatore che lo occupa e il numero di armate con cui lo occupa.

Implementare la base di dati in SQLite, e popolarla con i dati che rappresentino lo stato del gioco (semplificato) raffigurato di seguito. A tal fine, si tenga presente che:

- nel tabellone è presente un unico continente, chiamato "Italia centrale", il cui possesso dà diritto a 2 armate supplementari;
- i jolly sono le carte che raffigurano un fante, un cannone e un carro armato;
- l'ordine del gioco è: Gianni, Giulia.

**Carte di Gianni  
(armate nere):**



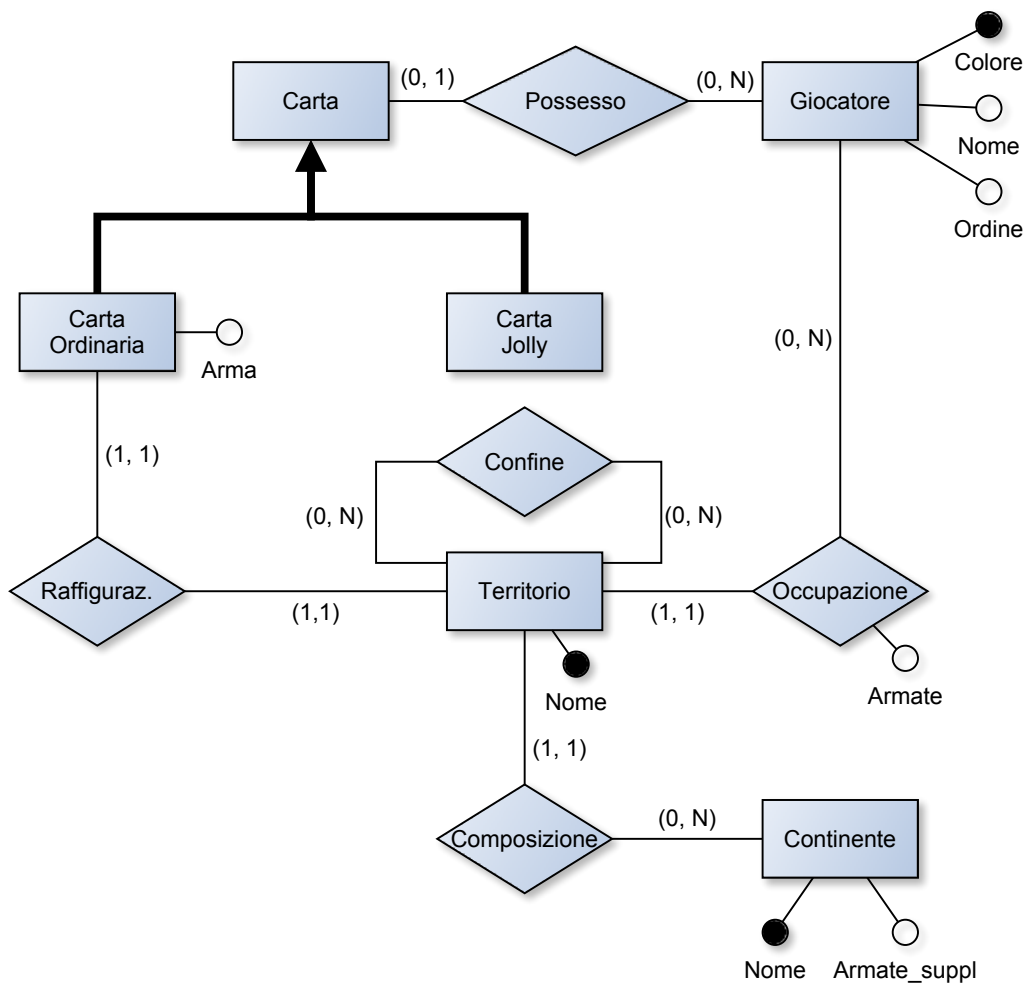
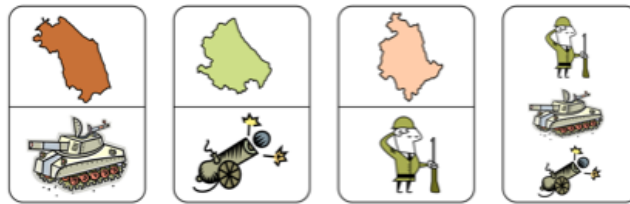
**Carte di Giulia  
(armate rosse):**



**Tabellone:**



**Carte  
del  
mazzo:**



## Comprensorio sciistico

Eseguire la progettazione logica della base “Comprensorio sciistico” di dati dell'Esercitazione 4a, di cui riportiamo per comodità le specifiche e lo schema concettuale. Inoltre presentare un esempio di istanza di base di dati, popolando le tabelle con i dati di esempio richiesti in questo testo.

Ogni impianto parte da una **piazzola**, avente una certa quota, e arriva a un'altra piazzola posta a una quota più elevata. Ogni pista parte da una piazzola e arriva a una piazzola collocata a una quota inferiore.

Per gli **impianti** si vogliono rappresentare il nome, la lunghezza e la portata, espressa in passeggeri/ora. Gli impianti possono essere **cabinovie**, **seggiovie** o **sciovie**:

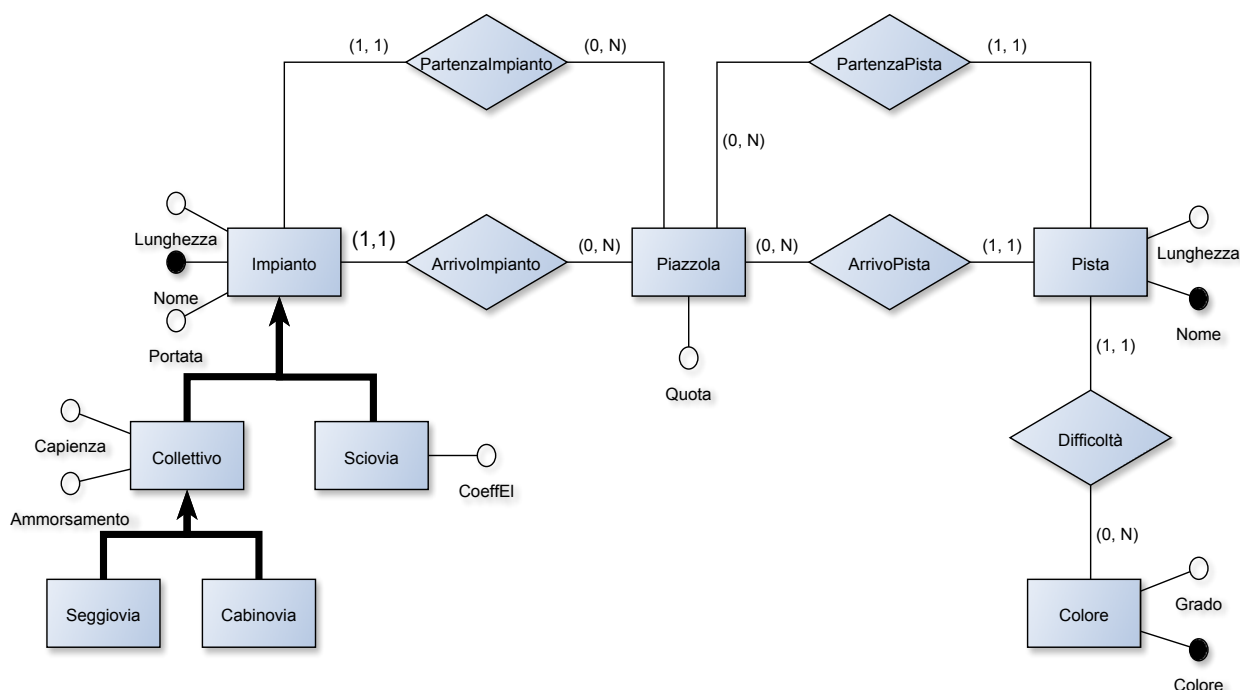
- per le cabinovie e le sciovie si vogliono rappresentare anche la capienza del singolo veicolo, e il tipo di ammortamento (fisso o automatico);
- per le sciovie si vuole rappresentare il coefficiente di elasticità della molla di recupero del piattello.

Per le **piste** si vogliono rappresentare il nome, la lunghezza e il colore della pista.

Il colore di una pista ne esprime il grado di difficoltà. Si dovrà poter definire un insieme di **colori** possibili, ordinato per grado di difficoltà.

Popolare un'istanza di esempio con i seguenti dati (nell'esempio, ogni piazzola è a una quota diversa):

- Pista **Orso** (rossa) e **Cocodrillo** (nera): entrambe partono da quota 1500 m e arrivano a quota 1200 m, condividendo partenza e arrivo.
- Pista **Stambecco** (rossa): parte da quota 1800 m e arriva a quota 1200 m.
- Seggiovia triposto **Poltrona**: parte da quota 1200 m e arriva a quota 1500 m, ammortamento manuale



- Skilift **Sgabello**: parte da quota 1500 m e arriva a quota 1800 m.
- Cabinovia esaposto **Divano**: parte da quote 1200 m e arriva a quota 1800 m, ammortamento automatico.

## Comprensorio sciistico - svolgimento della progettazione logica

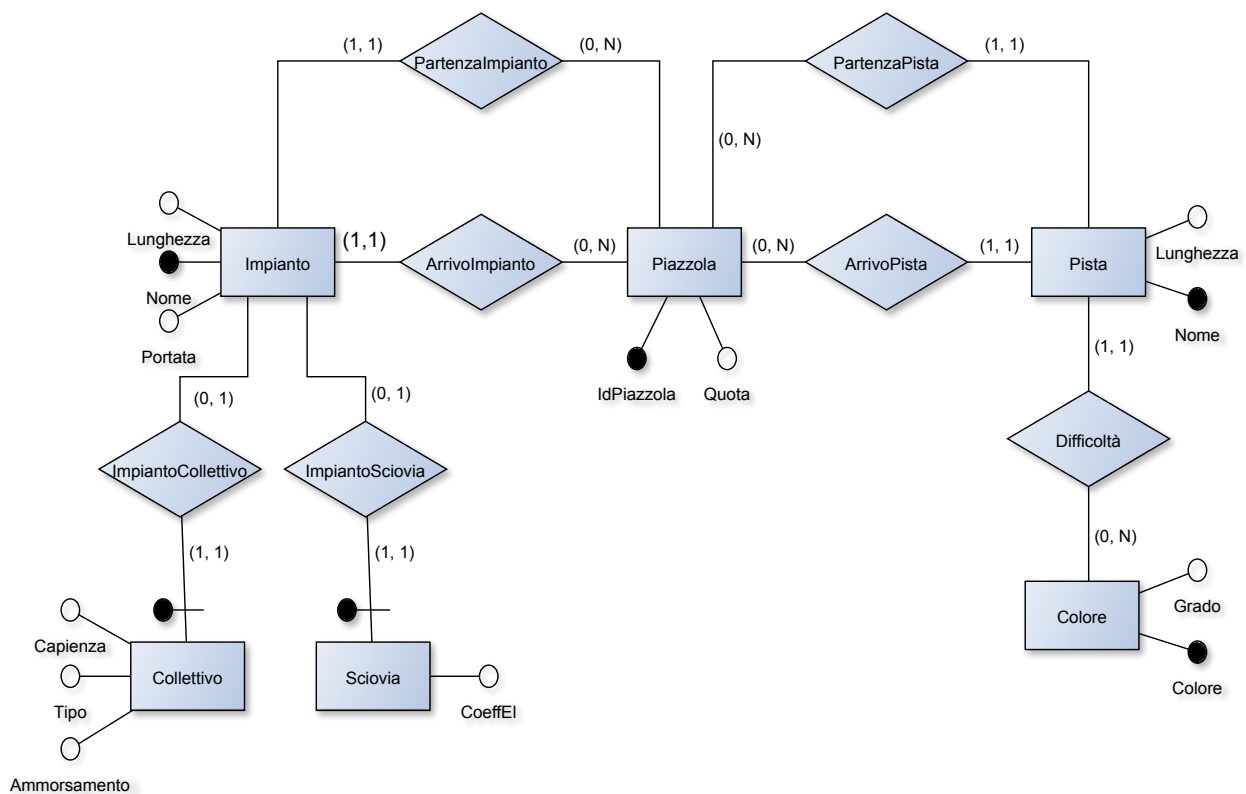
### Ristrutturazione dello schema Entità-Relazione

Eliminiamo la generalizzazione **Collettivo** accorpando le entità figlie **Seggiovia** e **Cabinovia** nell'entità padre **Collettivo**, poiché le entità figlie sono del tutto prive di attributi. Introduciamo un attributo aggiuntivo **tipo** che può assumere, come valori, "Seggiovia" oppure "Cabinovia".

Successivamente eliminiamo anche la generalizzazione **Impianto**. Poiché tanto l'entità padre quanto le entità figlie hanno parecchi attributi significativi, scegliamo di eliminare la generalizzazione introducendo, per ciascuna entità figlia, una relazione uno-a-uno con l'entità padre. (Ribadiamo che la generalizzazione si sarebbe potuta eliminare anche usando uno degli altri metodi presentati nel corso.)

L'entità **Piazzola** non ha identificatori tra gli attributi (più piazzole possono avere la stessa quota). Introduciamo dunque un identificatore ad-hoc **IdPiazzola**.

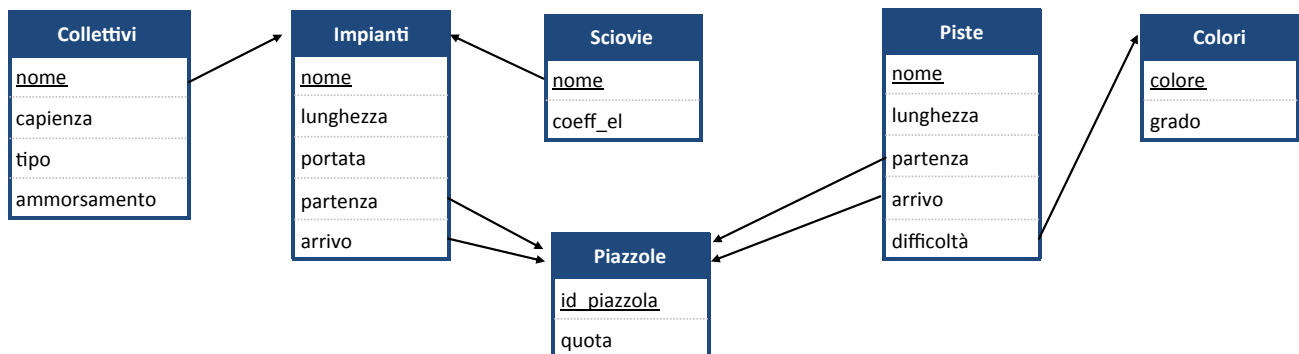
Ne risulta il seguente schema E-R ristrutturato:



Osserviamo che l'identificatore delle entità **Collettivo** e **Sciovia** è completamente esterno; in altri termini, esso è lo stesso identificatore di **Impianto**.

## Traduzione verso il modello relazionale

La traduzione verso il modello relazionale è particolarmente semplice perché tutte le relazioni sono uno-a-molti (o uno-a-uno): esse si realizzano introducendo una foreign key nella tabella che discende dall'entità che partecipa alla relazione con cardinalità massima pari a 1:



## Implementazione istanza

### Piazzole

id_piazzola	quota
P1	1200
P2	1500
P3	1800

### Colori

colore	grado
Blu	1
Rossa	2
Nera	3

### Piste

nome	lunghezza	partenza	arrivo	difficoltà
Orso	900	P2	P1	Rossa
Cocodrillo	700	P2	P1	Nera
Stambecco	680	P3	P2	Rossa

### Impianti

nome	lunghezza	portata	partenza	arrivo
Poltrona	500	1000	P1	P2
Sgabello	400	600	P2	P3
Divano	900	2000	P1	P3

### Collettivi

nome	capienza	tipo	ammorsamento
Poltrona	3	S	M
Divano	6	C	A

### Skilift

nome	Coeff_el
Sgabello	6,53

## DB Riparto scout

Si vuole costruire una base di dati per memorizzare i dati relativi alla composizione di un riparto scout.

Un riparto è costituito da un certo numero di ragazzi, detti **scout**, di cui vogliamo rappresentare i dati anagrafici (nome, cognome, data di nascita), il numero di tessera e il numero di telefono.

Gli scout del riparto sono suddivisi in gruppetti, detti **squadriglie**. Ogni **squadriglia** ha un nome (il nome di un animale), e un “motto”, chiamato urlo di squadriglia. Ciascuno scout occupa una posizione gerarchica all’interno della propria squadriglia: ci sono un capo, un vice, un terzo di squadriglia, e così via.

Durante la loro vita, gli scout compiono un percorso, raggiungendo una serie di tappe in un ordine prefissato. Ogni **tappa** ha un nome, e un ordine rispetto alle altre tappe. Vogliamo rappresentare la tappa raggiunta da ciascuno scout (per ultima).

Infine, l’abilità tecnica di uno scout è “certificata” mediante il conseguimento di brevetti di specialità. Ogni **specialità** ha un nome e una tappa minima richiesta (il ragazzo non può conseguire la specialità se non ha raggiunto almeno quella tappa). Uno scout può conseguire tutti i brevetti di specialità che desidera, purché abbia raggiunto la tappa minima richiesta.

Eseguire la progettazione concettuale e logica della base di dati del riparto scout.

Preparare un’istanza con i seguenti dati:

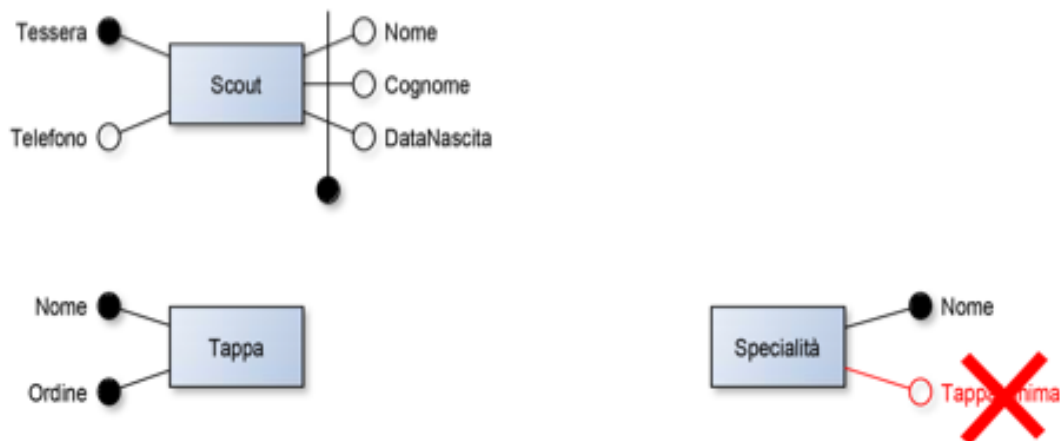
- Tappe: Novizio, Piede Tenero, Seconda Classe, Prima Classe, Esploratore Scelto.
- Specialità: Segnalatore, Ambulanziere, Campeggiatore (richiedono la Seconda Classe); Pioniere, Fotografo (richiedono la Prima Classe).
- Squadriglie: Aquile (urlo: “Aquile sulle vette”), Castori, Pantere.
- Scout: Luca (Capo Aquile), Maurizio (Vice Aquile), Pierpaolo (Terzo Aquile), Andrea (Capo Pantere).
- Luca e Andrea sono scout di Seconda Classe, Maurizio e Pierpaolo sono Piedi Teneri.
- Luca ha le specialità di Segnalatore, Andrea ha le specialità di Campeggiatore e Segnalatore.

### DB Riparto scout – Svolgimento progettazione concettuale

#### Entità e loro attributi, identificatori

Iniziamo a individuare e rappresentare le entità e i loro attributi. Le entità sono **Scout**, **Squadriglia**, **Tappa**, e **Specialità**:





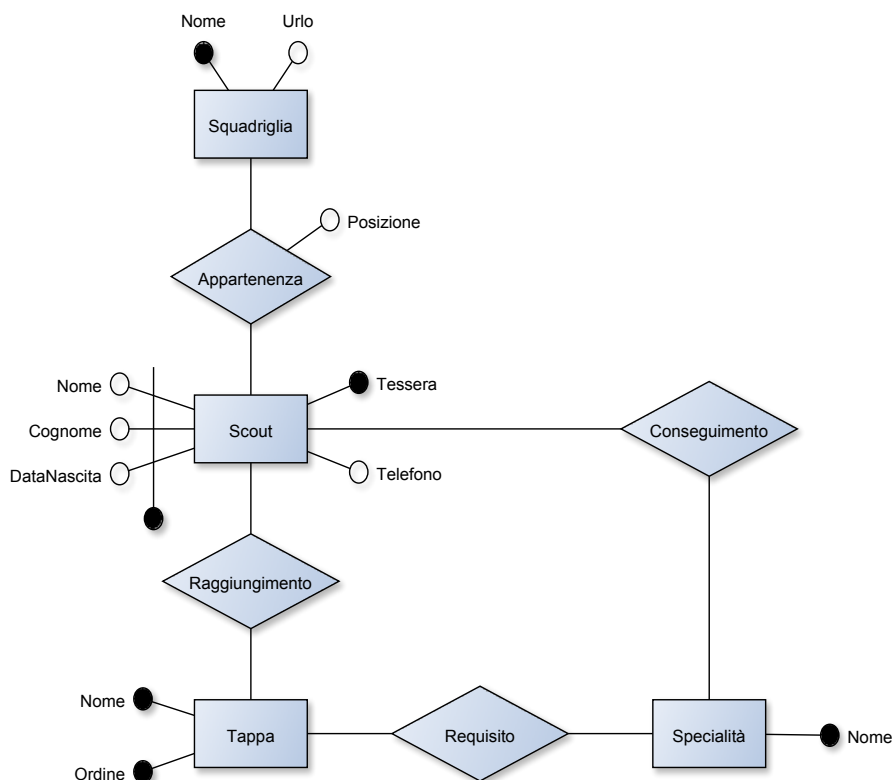
Per ciascuna entità abbiamo individuato uno o più possibili identificatori.

Osserviamo che è **errato** definire **TappaMinima** come attributo dell'entità **Specialità**. Infatti, da un punto di vista concettuale, la tappa minima richiesta per conseguire una specialità è una tappa; e, dal momento che nello schema è presente un'entità **Tappa**, per rappresentare il concetto di “tappa minima richiesta” occorre introdurre una relazione tra l'entità **Specialità** e l'entità **Tappa**. Chiameremo tale relazione **Requisito**.

## Relazioni e loro attributi

Introduciamo ora le relazioni.

**Appartenenza:** denota l'appartenenza degli scout alle squadriglie. Poiché vogliamo rappresentare l'ordine gerarchico degli scout all'interno della squadriglia, definiamo un attributo **Posizione** della relazione. Il caposquadriglia apparterrà alla squadriglia con posizione 1, il vicecapo apparterrà alla squadriglia con posizione 2, e così via.



**Raggiungimento:** denota la tappa raggiunta da ciascuno scout.

**Conseguimento:** denota le specialità conseguite da ciascuno scout.

**Requisito:** denota la tappa richiesta per poter conseguire una specialità.

Ne risulta lo schema concettuale (parziale) precedente.

### Cardinalità delle relazioni

Riesaminiamo una a una le relazioni, e stabiliamo le loro cardinalità.

#### Appartenenza

- Scout - A quante squadriglie appartiene uno scout? Minimo 1, massimo 1: cardinalità (1, 1).
- Squadriglia – Quanti scout ha una squadriglia? Minimo 0, massimo senza vincoli: cardinalità (0, N).

Le cardinalità massime sono 1 e N: la relazione è quindi di tipo uno-a-molti.

#### Raggiungimento

- Scout – In quante tappe si trova uno scout? Minimo 1, massimo 1 (ricordiamo che le tappe funzionano come i gradi militari, ogni scout occupa una tappa): cardinalità (1, 1).
- Tappa – Quanti scout occupano una data tappa? Minimo 0 (per esempio le tappe più difficili potrebbero ancora non essere state raggiunte da alcuno scout), massimo indeterminato: cardinalità (0, N).

Le cardinalità massime sono 1 e N: la relazione è quindi di tipo uno-a-molti.

#### Conseguimento

- Scout – Quante specialità può conseguire uno scout? Quante ne vuole, cioè minimo 0, massimo senza vincoli: cardinalità (0, N).
- Specialità – Quanti scout possono conseguire una specialità? Minimo 0, massimo senza vincoli: cardinalità (0, N).

Le cardinalità massime sono N e N: la relazione è quindi di tipo molti-a-molti.

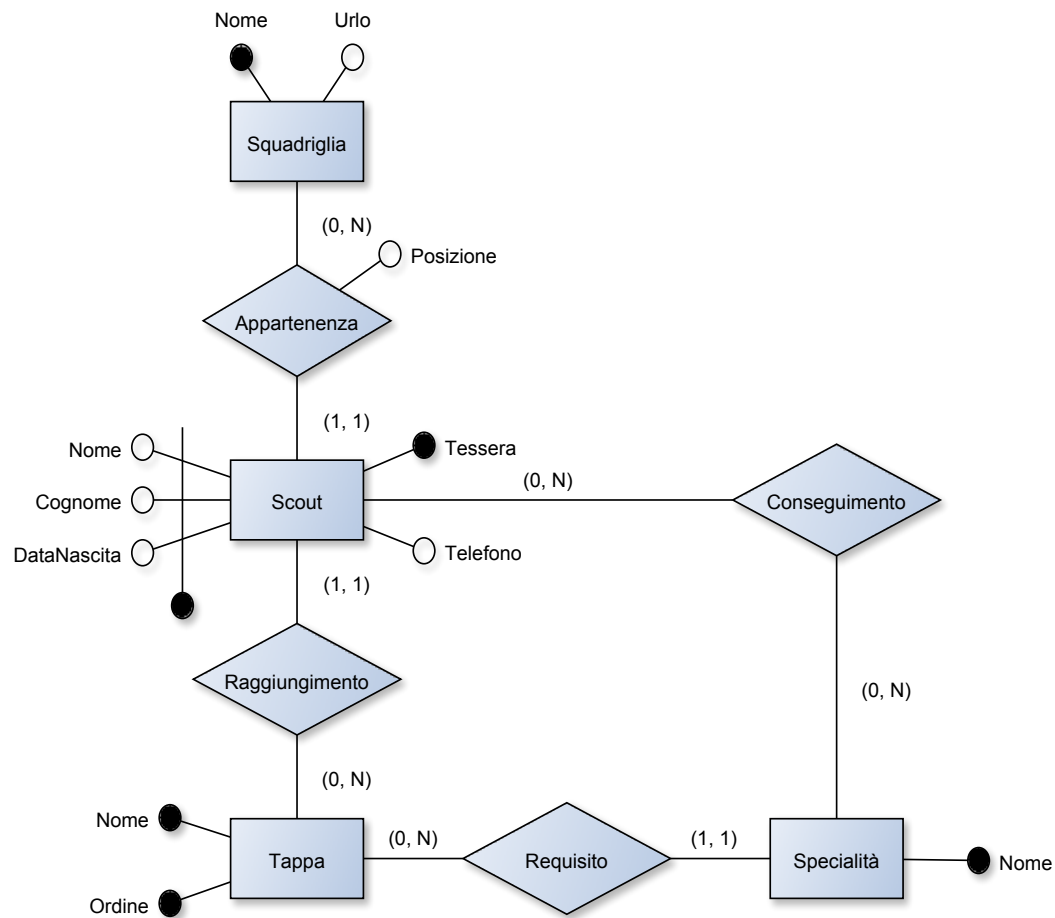
#### Requisito

- Specialità – Quante tappe sono il requisito per conseguire una specialità? Secondo le specifiche, ogni specialità ha come requisito una data tappa, quindi ogni specialità ha minimo 1 e massimo 1 requisito: cardinalità (1, 1).
- Tappa – Data una tappa, quante specialità hanno tale tappa come requisito? Non lo sappiamo, quindi minimo 0, massimo senza vincoli: cardinalità (0, N).

Le cardinalità massime sono 1 e N: la relazione è quindi di tipo uno-a-molti.



Lo schema Entità-Relazione completo è dunque il seguente:



**La progettazione logica e l'implementazione dell'istanza richiesta sono lasciati come esercizio.**