

Progetto di Basi di Dati

Si intende sviluppare un'applicazione dedicata al servizio di delivery per un ristorante giapponese. I clienti, registrandosi all'applicazione mediante il proprio indirizzo e-mail e fornendo il proprio nome e cognome, acquisiscono la possibilità di effettuare ordini. Ogni ordine è identificato da un codice univoco e si caratterizza per il costo totale e la data di effettuazione.

L'applicazione prevede due tipologie di ordine:

1. Ordini con consegna a domicilio, per i quali è necessario indicare un indirizzo specifico, costituito da via, città e numero civico.
2. Ordini con ritiro presso il locale, per i quali è necessario indicare l'orario di ritiro prefissato.

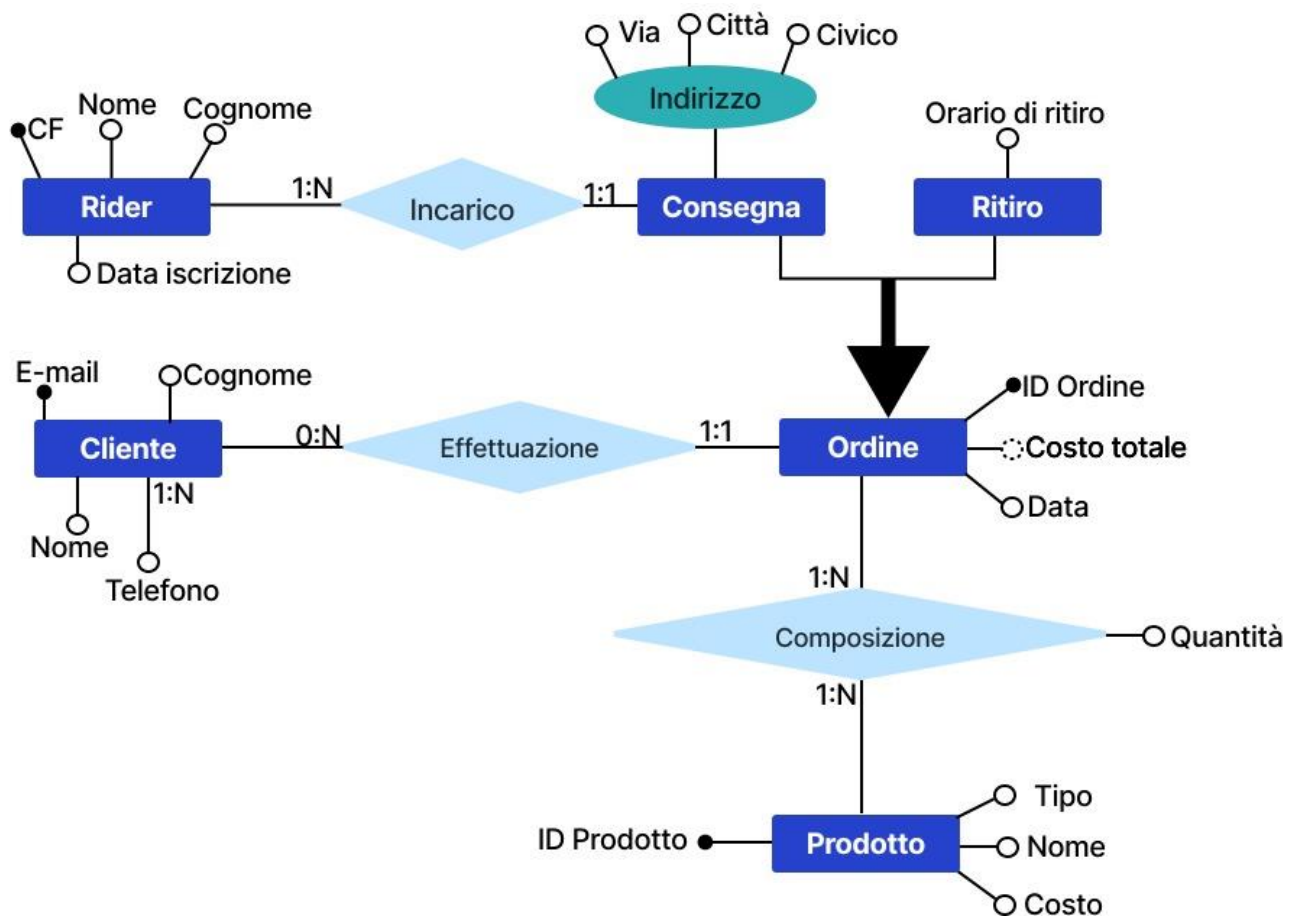
La responsabilità della consegna degli ordini è affidata a dei corrieri (riders), ciascuno dei quali è identificato da un codice fiscale e distinto per nome, cognome e data di iscrizione al servizio.

Per garantire una comunicazione efficace in caso di eventuali problematiche relative agli ordini, l'applicazione memorizza uno o più numeri di telefono per ogni cliente registrato.

Gli ordini sono costituiti da uno o più prodotti, ognuno con una specifica quantità per ordine. I prodotti sono categorizzati in base a nome, costo, tipologia e sono distinti tramite un identificativo unico.

Progettazione concettuale della base di dati

Schema ER



Definizione delle procedure per la gestione della base di dati

Tavola dei volumi

CONCETTO	TIPO	VOLUME
CLIENTE	Entità	50
RIDER	Entità	10
PRODOTTO	Entità	20
ORDINE	Entità	500
CONSEGNA	Entità	400
RITIRO	Entità	100
EFFETTUAZIONE	Relazione	500
COMPOSIZIONE	Relazione	10000
INCARICO	Relazione	400

Si è considerato:

- che al massimo ogni cliente effettua dieci ordini
- 80% degli ordini sono consegne.
- Effettuazione relazione 1-N, si inserisce volume degli ordini.
- Incarico relazione 1-N, si inserisce volume delle consegne
- Composizione relazione N-N, tutti posso partecipare con tutti: $20 * 500$ non è plausibile avere un paletto al numero di prodotti da vendere

Tavola delle operazioni

Considerando che in ordine compaiono 10 prodotti diversi, le operazioni che coinvolgono l'attributo ridondante sono le seguenti:

1. Visualizza il costo totale di un ordine (100/giorno).
2. Memorizza un nuovo ordine (200/giorno).
3. Visualizza i dati di un rider con i relativi ordini effettuati (30/giorno).

Operazione	Tipo	Frequenza
Op.1	I	100 al giorno
Op.2	I	200 al giorno
Op.3	I	30 al giorno

Progettazione logica

Analisi delle ridondanze

Il dato ridondante è costo totale dell'entità ordine.

Consideriamo che costo totale venga memorizzato come float a doppia precisione e che sulla macchina in questione utilizzi 8 bytes.

Tavola degli accessi

Operazione 1

Calcolo con ridondanza				Calcolo senza ridondanza			
Concetto	Costrutto	Accessi	Tipo Accessi	Concetto	Costrutto	Accessi	Tipo Accessi
Ordine	E	1	L	Ordine	E	1	L
				Composizione	R	10	L
				Prodotto	E	10	L
Totale	$100 * 1L + 4Kbyte = 100L + 4Kbyte$			$100 * 21L = 2100L$			

Operazione 2

Calcolo con ridondanza				Calcolo senza ridondanza			
Concetto	Costrutto	Accessi	Tipo Accessi	Concetto	Costrutto	Accessi	Tipo Accessi
Ordine	E	1	S	Ordine	E	1	S
Effettuazione	R	1	S	Effettuazione	R	1	S
Composizione	R	10	S	Composizione	R	10	S
Prodotto	E	10	L				
Totale	$200 * (2L+2L+20L+20L) + 4Kbyte = 8800L + 4Kbyte$			$200 * (2L+2L+20L) = 4800L$			

Operazione 3

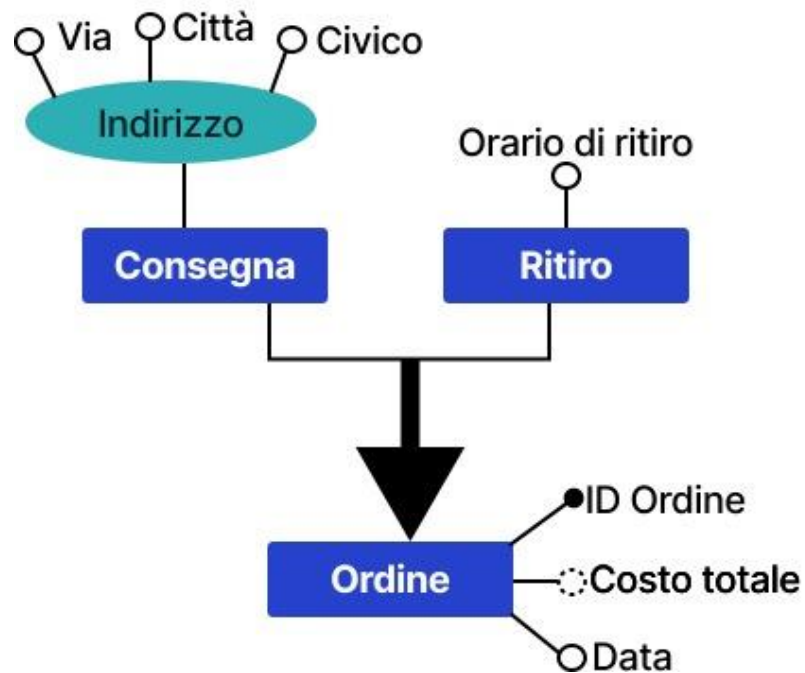
Calcolo con ridondanza				Calcolo senza ridondanza			
Concetto	Costrutto	Accessi	Tipo Accessi	Concetto	Tipo	Accessi	Tipo Accessi
Rider	E	1	L	Rider	E	1	L
Incarico	R	40	L	Incarico	R	40	L
Consegna	E	40	L	Consegna	E	40	L
Ordine	E	40	L	Ordine	E	40	L
				Composizione	R	400	L
				Prodotto	E	400	L
Totale	$30 * (1L+40L+40L+40L) = 3630L + 4Kbyte$			$30 * (1L+40L+40L+40L+400L+400L) = 27630L$			

Totale accessi con ridondanza	Totale accessi senza ridondanza
12530L + 4Kbyte	34530L

Dato il minor numero di accessi, è più efficace scegliere di mantenere il dato ridondante “costo totale”.

Eliminazione delle gerarchie

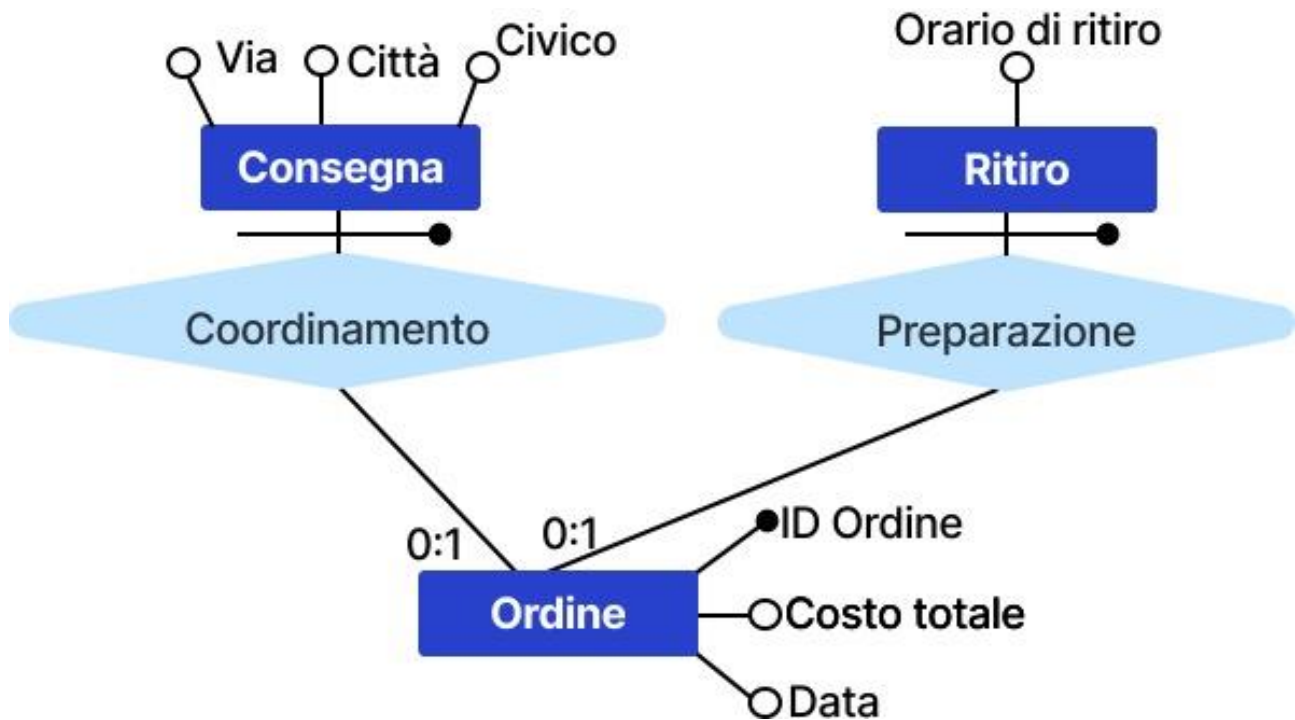
Nello schema inizialmente elaborato, è presente la seguente specializzazione dell'entità "Ordine":



In questa fase di progettazione logica, è necessario individuare un metodo efficace di ristrutturazione che permetta l'eliminazione di questa specializzazione.

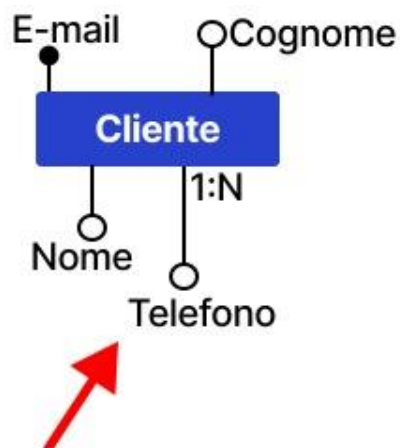
Tra i tre diversi metodi studiati per l'eliminazione delle generalizzazioni si è scelto di trasformare i figli in entità deboli rispetto al padre.

Ristrutturiamo quindi come segue:



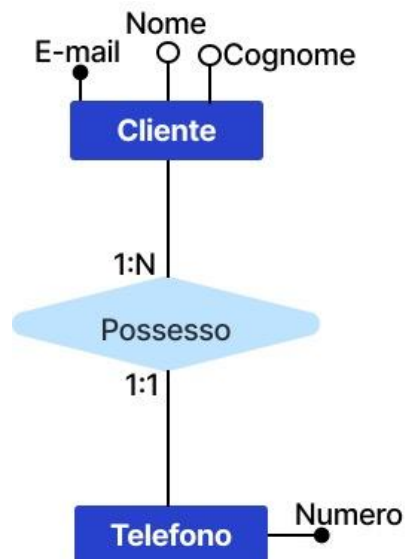
Eliminazione dell'attributo multi-valore

Nello schema inizialmente elaborato compare l'attributo multi-valore "numero telefono".



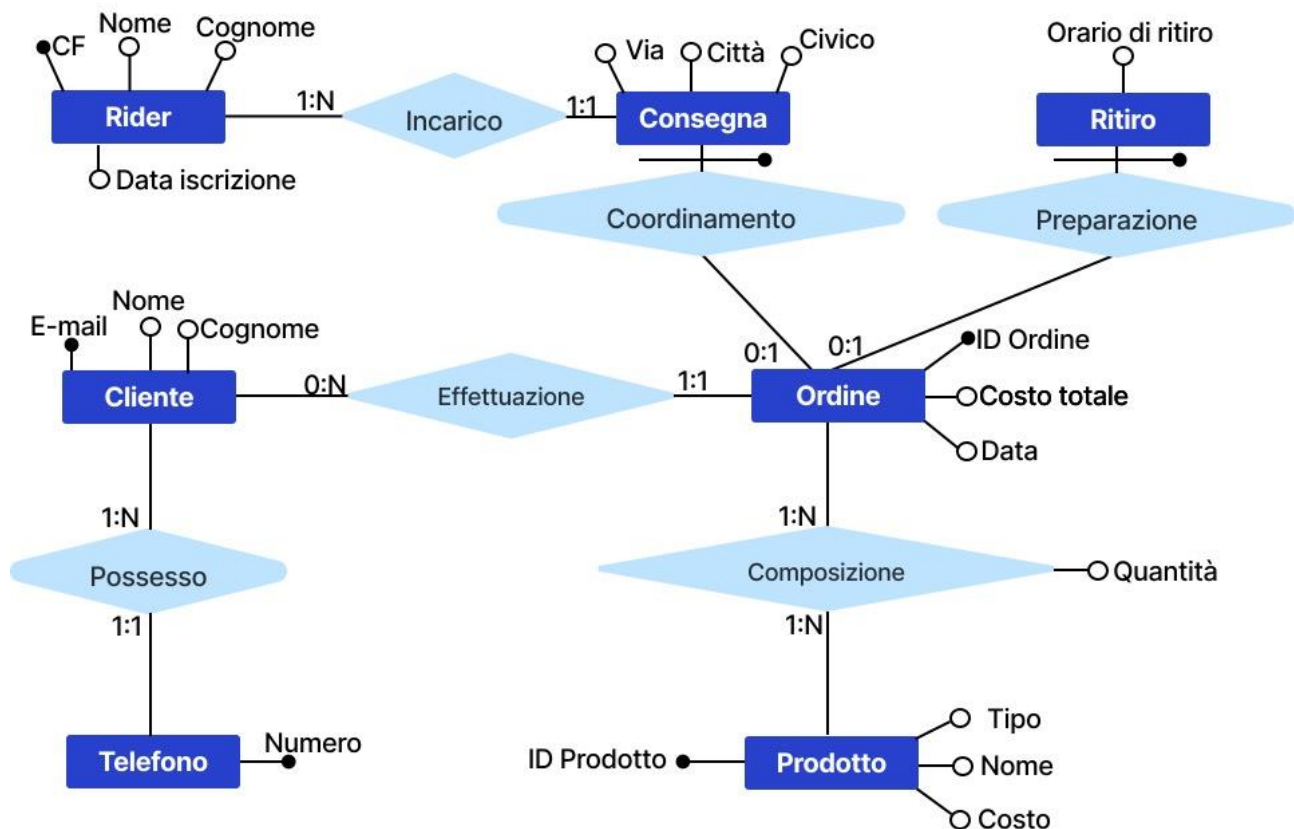
Tale forma di attributo va risolto in maniera differente in fase di progettazione logica.

Si sceglie quindi di definire una nuova entità: Telefono



Schema ER ristrutturato

Al termine della fase di ristrutturazione, lo schema ER completo che ne deriva è il seguente:



Schema relazionale

Si procede al mapping della base di dati:

Cliente (E-mail, Nome, Cognome)

Telefono (Numero, *Cliente.E-mail*)

Rider (CF, Nome, Cognome, Data iscrizione)

Ordine (ID_Ordine, Costo Totale, *Cliente.E-mail*)

Consegna (ID_Ordine, Via, Città, Civico, *Rider.CF*)

Ritiro (ID_Ordine, Orario_Ritiro)

Composizione (ID_Ordine, ID_Prodotto, Quantità)

Prodotto (ID_Prodotto, Tipo, Costo, Nome)

Telefono (Cliente.E-mail) ha VIR con Cliente (E-mail)

Ordine (Cliente.E-mail) ha VIR con Cliente (E-mail)

Consegna (ID_Ordine) ha VIR con Ordine(ID_Ordine)

Ritiro (ID_Ordine) ha VIR con Ordine(ID_Ordine)

Consegna (Rider.CF) ha VIR con Rider(CF)

Composizione (ID_Ordine) ha VIR con Ordine(ID_Ordine)

Composizione (ID_Prodotto) ha VIR con Prodotto(ID_Prodotto)