

# Università Degli Studi Di Salerno

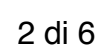


## Progetto di Basi Dati: RooManager

Gargiulo Emanuele - 0512102244

Izzo Giandomenico - 0512102292

Ristrutturazione ER, schema relazionale, normalizzazione



Si era originariamente pensato di inserire un attributo in "Dipendente" per salvare il numero di task svolti da ognuno.

Le operazioni che andrebbero a coinvolgere tale attributo sono:

- Creazione di task quotidiano
- Lettura del numero di task svolti da un certo dipendente nell'ultima settimana.

La prima operazione viene svolta 200 volte al giorno, quindi 1400 volte alla settimana. La seconda viene invece svolta due volte alla settimana.

Per la prima operazione, con la presenza dell'attributo ridondante, abbiamo

Task            E 1 S

Dipendente   E 1 S

= 2 S = 4 accessi x 1400 volte = 5600 accessi alla settimana

Per la seconda operazione, con la presenza dell'attributo ridondante, abbiamo

Dipendente E 1 L = 1 L x 2 volte = 2 accessi.

**Totale accessi con ridondanza: 5600.**

La quantità di memoria utilizzata, ipotizzando 4 byte per dipendente e tenendo presente che avremo circa 20 dipendenti, è trascurabile.

Per la prima operazione, senza ridondanza, abbiamo

Task E 1 S = 2 accessi x 1400 volte = 2800 accessi

Per la seconda operazione, senza ridondanza, abbiamo

Dipendente   E 1   L

Task            E 70 L

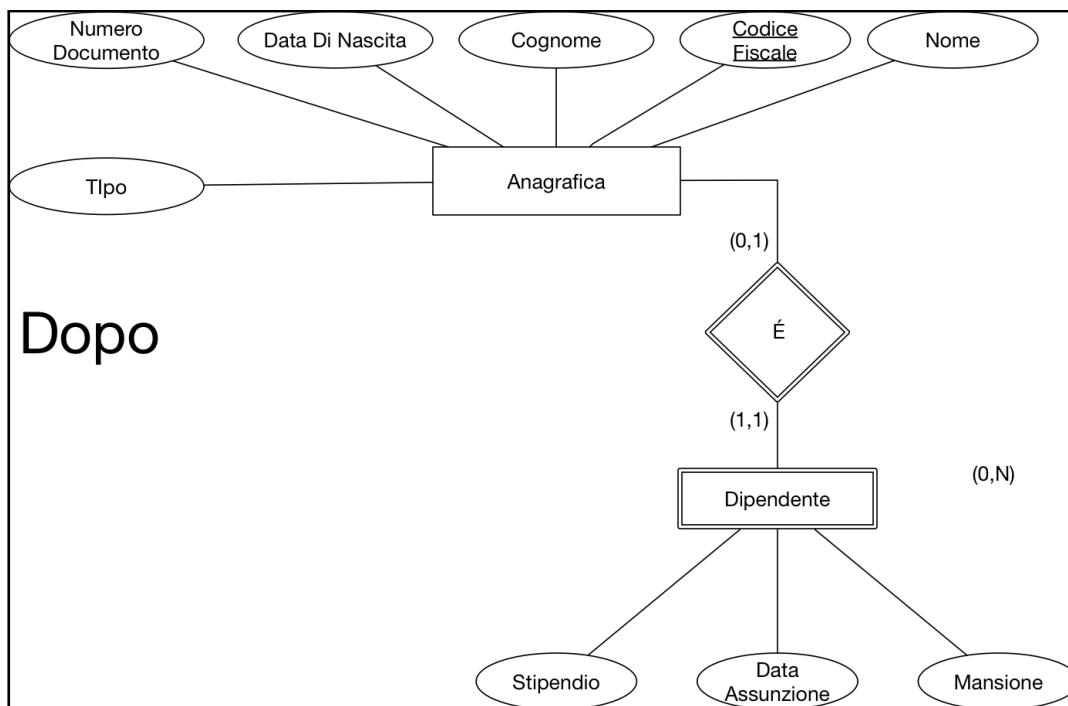
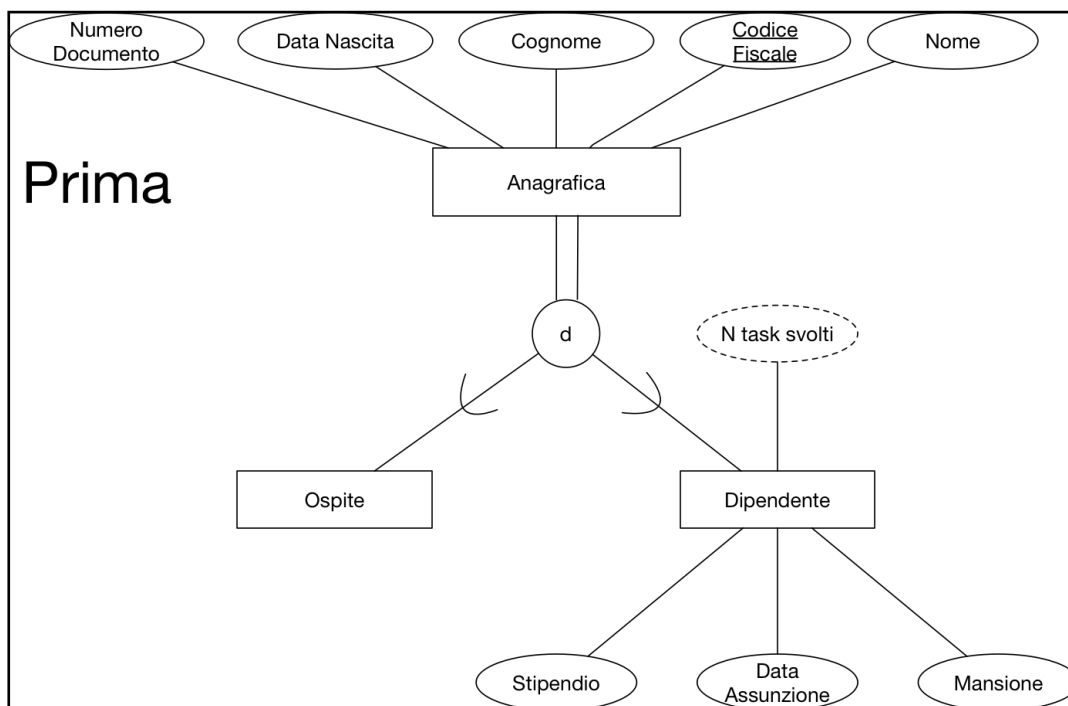
Vengono considerati 70 task perché se abbiamo 20 dipendenti e vengono svolti in media 1400 task alla settimana,  $1400 / 20 = 70$

= 71 accessi x 2 volte = 142 accessi.

**Totale accessi senza ridondanza: 2942**

Dunque senza ridondanza avremo un numero di accessi inferiore quasi della metà, oltre a risparmiare memoria. Preferiremo dunque derivare il numero di task quando necessario piuttosto che memorizzarlo come attributo dell'entità dipendente.

Erano presenti diverse generalizzazioni, che sono state opportunamente modificate. L'entità Anagrafica, che era specializzata in Ospite o Dipendente, è stata modificata accorparendo il tipo "Ospite" all'interno di Anagrafica stessa, dato che non aveva attributi aggiuntivi, e collegando l'entità "Dipendente" all'entità "Anagrafica" con una relazione di tipo *è*, cioè sostituendo la generalizzazione con un'associazione, soluzione sembrata migliore in quanto erano presenti operazioni che facevano distinzioni tra Anagrafica in generale (e quindi Ospite) e Dipendente. E' stato inoltre aggiunto un attributo di "Tipo" all'interno di Anagrafica per indicare se si tratta di un'entità di tipo "base" (Anagrafica) o di un'entità di tipo Dipendente.

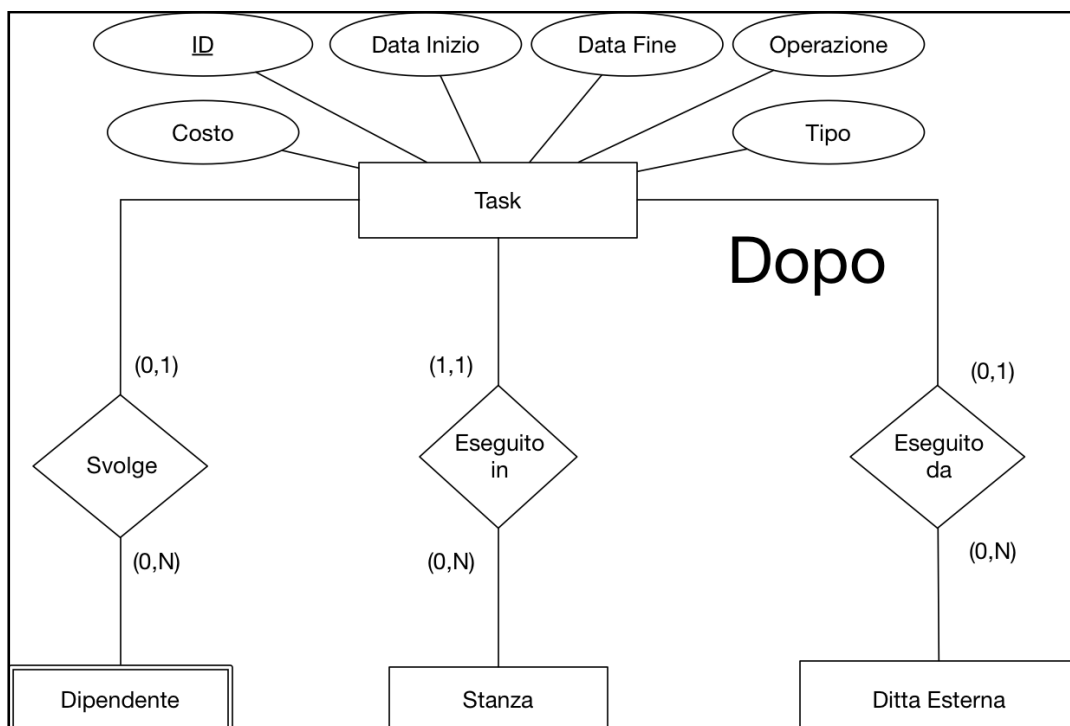
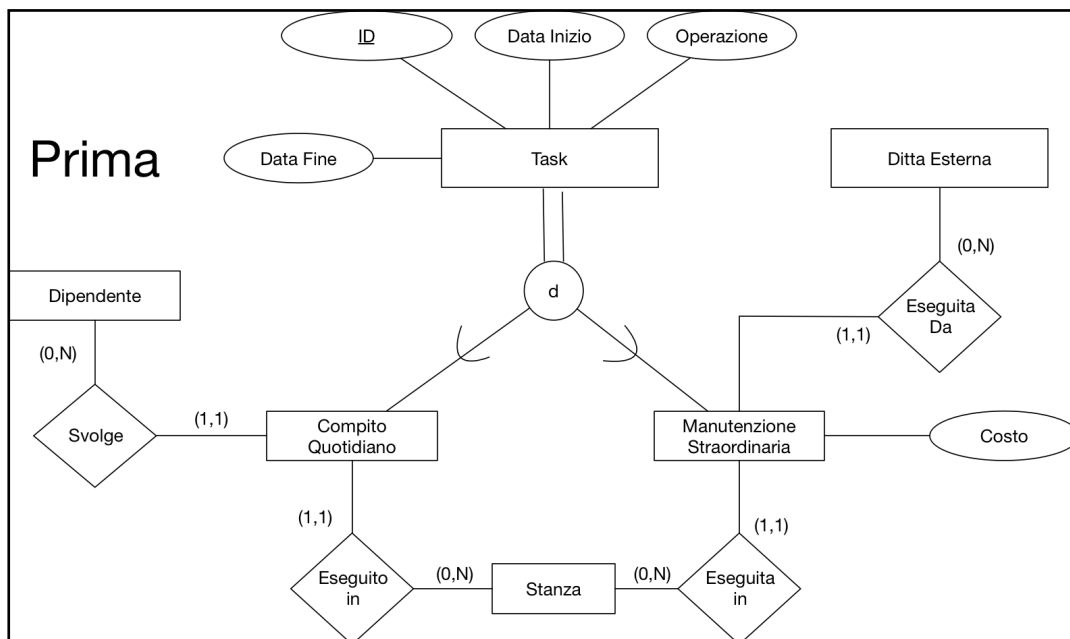


Per quanto riguarda l'entità Task, essa era originariamente suddivisa in entità Compito Quotidiano ed in entità Manutenzione Straordinaria.

Come attributi, l'unica differenza era che per la Manutenzione Straordinaria veniva memorizzato un costo. E' stato ritenuto che portando tutto nell'entità padre task, la quantità di memoria utilizzata dai campi *null* inevitabilmente presenti è assolutamente accettabile se messa nel contesto della semplificazione che questo porta allo schema concettuale.

E' stato dunque naturale portare le entità Compito Quotidiano e Manutenzione Straordinaria all'interno dell'entità padre Task.

E' stato poi aggiunto un attributo *tipo* che permette di distinguere i task quotidiani da quelli di manutenzione straordinaria.



## Mapping:

Anagrafica (CodiceFiscale, nome, cognome, dataNascita, numeroDocumento, tipo)

Dipendente (Anagrafica.CodiceFiscale↑, Stipendio, DataAssunzione, Mansione)

Stanza (Numero, tipologia, costoGiornaliero)

Permanenza (Anagrafica.CodiceFiscale↑, Stanza.Numero↑, DataFine, DataInizio)

Task (ID, Operazione, Tipo, Stanza.Numero↑, Costo, DataInizio, DataFineDittaEsterna.PIVA↑\*, Dipendente.CodiceFiscale↑\*)

DittaEsterna (PIVA, nome, sede, recapito)

Fattura (ID, Causale, Importo, Data, Permanenza.CodiceFiscale↑, Permanenza.Stanza↑)

### //RIVEDERE NORMALIZZAZIONE

Dopo aver esaminato lo schema relazionale finale, è risultato che esso rispetta già la terza forma normale, in quanto non è presente alcun dato che dipenda parzialmente dalla chiave o che abbia dipendenze transitive.

Si giunge dunque alla conclusione che non rimangono relazioni da normalizzare.