

Progettazione logica: criteri di ottimizzazione

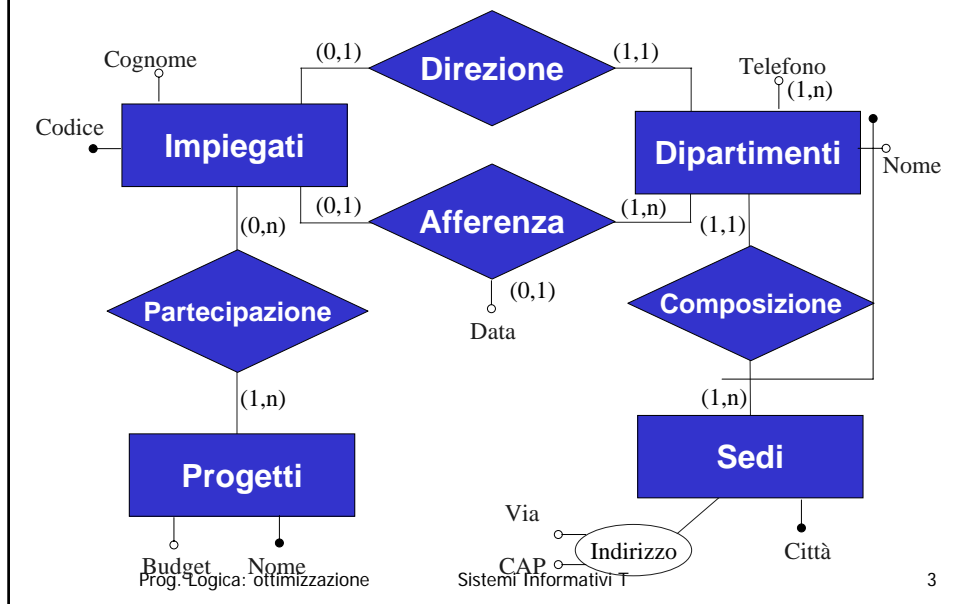
Sistemi Informativi T

Versione elettronica: [08.2.progLogica.ottimizzazione.pdf](#)

Criteri di ottimizzazione

- Per confrontare tra loro diverse alternative di traduzione bisogna conoscere, almeno in maniera approssimativa, il cosiddetto “**carico di lavoro**” del DB, ovvero:
 - I “**volumi**” dei dati in gioco
 - Le principali **operazioni** che il DB dovrà supportare
- Gli **indicatori di costo** che si utilizzano considerano quindi due aspetti
 - **spazio**: numero di istanze previste nel DB
 - **tempo**: numero di istanze (di entità e associazioni) visitate durante un'operazione

Schema di riferimento



3

Tavola dei volumi

- Specifica il numero stimato di istanze per ogni entità (E) e associazione (R) dello schema
- I valori sono necessariamente approssimati, ma indicativi
 - I valori indicati con * sono derivati utilizzando i vincoli di cardinalità

Concetto	Tipo	Volume
Sedi	E	10
Dipartimenti	E	80
Impiegati	E	2000
Progetti	E	500
Composizione	R	80 *
Afferenza	R	1900
Direzione	R	80 *
Partecipazione	R	6000

Prog. Logica: ottimizzazione

Sistemi Informativi T

4

Esempio di valutazione di costo

Dato il codice di un impiegato, trova tutti i dati dell'impiegato, del dipartimento nel quale lavora e dei progetti ai quali partecipa

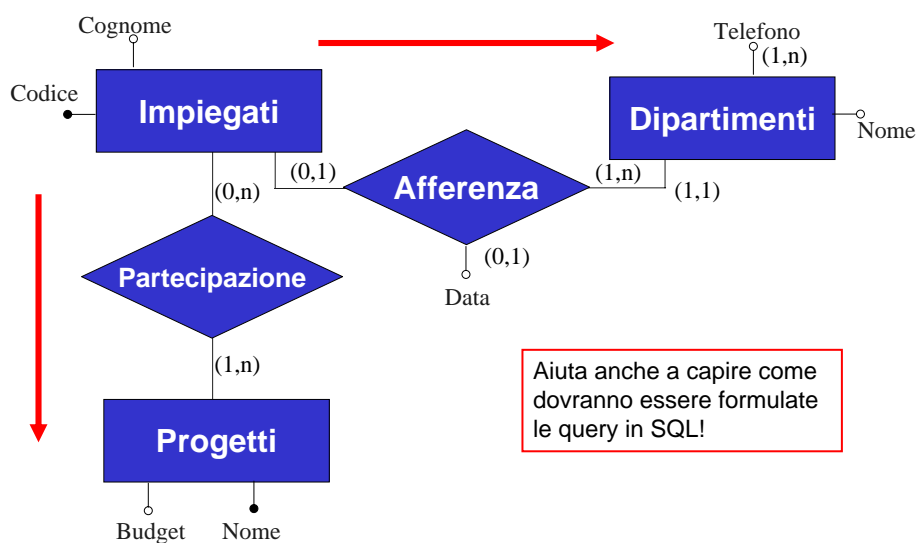
- Si costruisce una **tavola degli accessi** basata su uno **schema di navigazione**
- Lo **schema di navigazione** è la **parte dello schema E/R** interessata dall'operazione, estesa con delle frecce che indicano in che modo l'operazione "naviga" i dati
- Per ragionare su quali entità/associazioni sono interessate dall'operazione, si considera il principio di "duplicazione minima" di informazioni, ovvero:
 - entità identificate esternamente inglobano le relative associazioni (che quindi "non contano")
 - le associazioni inglobano gli identificatori delle entità partecipanti
 - entità figlie inglobano l'identificatore dell'entità genitore

Prog. Logica: ottimizzazione

Sistemi Informativi T

5

Esempio di schema di navigazione



Prog. Logica: ottimizzazione

Sistemi Informativi T

6

Esempio di tavola degli accessi

- Per ogni entità e associazione interessata dall'operazione, riporta il **numero di istanze interessate**, e il **tipo di accesso** (L: lettura; S: scrittura)
- Il numero delle istanze si ricava dalla tavola dei volumi mediante semplici operazioni
 - Ad es.: in **media** ogni impiegato partecipa a $6000/2000 = 3$ progetti)

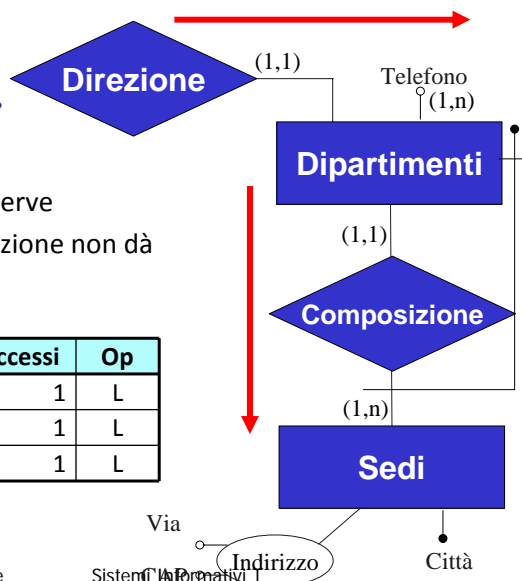
Concetto	Tipo	Accessi	Op
Impiegati	E	1	L
Afferenza	R	0.95	L
Dipartimenti	E	0.95	L
Partecipazione	R	3	L
Progetti	E	3	L

Un altro esempio

Dato il codice di un direttore, trova l'indirizzo della sede del suo dipartimento

- L'entità Impiegati non serve
- L'associazione Composizione non dà luogo ad accessi

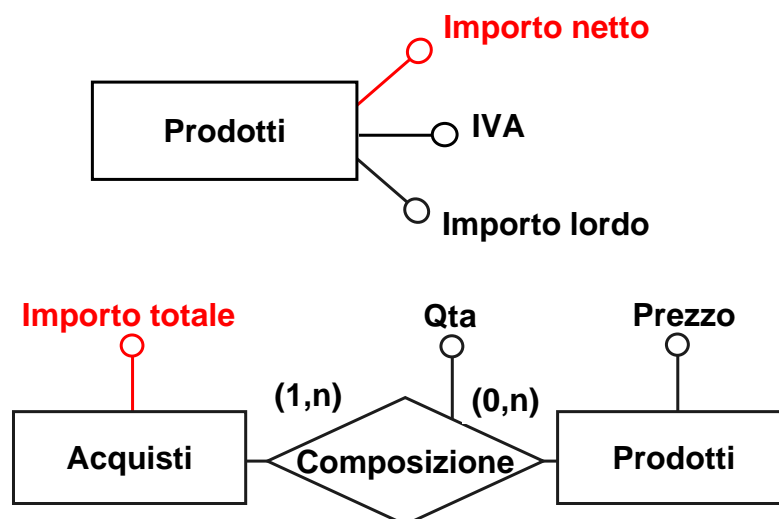
Concetto	Tipo	Accessi	Op
Direzione	R	1	L
Dipartimenti	E	1	L
Sedi	E	1	L



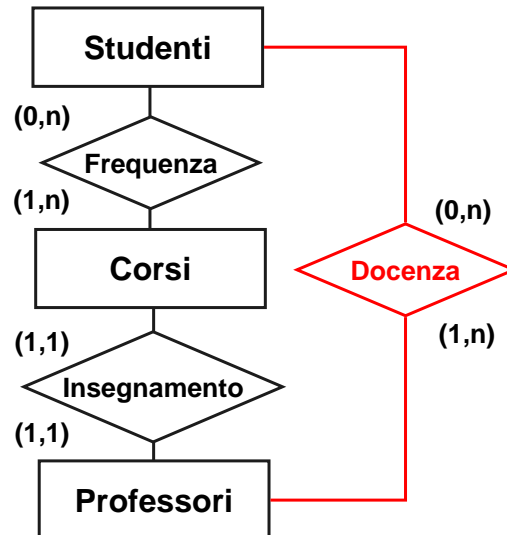
Analisi delle ridondanze

- Una **ridondanza** in uno schema E-R è una **informazione significativa ma derivabile da altre**
- E' importante capire se eliminare le ridondanze eventualmente presenti o mantenerle (**è quindi comunque importante averle individuate in fase di progettazione concettuale!**)
- **Se si mantiene una ridondanza**
 - **si semplificano alcune interrogazioni**, ma
 - **si appesantiscono gli aggiornamenti**
 - **si occupa maggior spazio**
- Le possibili ridondanze riguardano:
 - Attributi derivabili da altri attributi
 - Associazioni derivabili dalla composizione di altre associazioni (presenza di cicli)

Attributi derivabili



Associazioni ridondanti



Prog. Logica: ottimizzazione

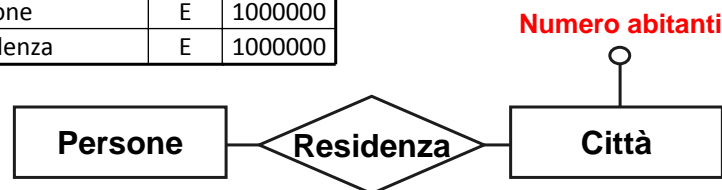
Sistemi Informativi T

11

Esempio di analisi di una ridondanza

- Si considerano innanzitutto le operazioni influenzate dalla ridondanza, considerando anche le loro frequenze di esecuzione...

Concetto	Tipo	Volume
Città	E	200
Persone	E	1000000
Residenza	E	1000000



- **Operazione 1:** memorizza una nuova persona con la relativa città di residenza (500 volte al giorno)
- **Operazione 2:** stampa tutti i dati di una città (incluso il numero di abitanti) (2 volte al giorno)

Prog. Logica: ottimizzazione

Sistemi Informativi T

12

I 2 scenari...

Con ridondanza

Concetto	Tipo	Accessi	Op
Persone	E	1	S
Residenza	R	1	S
Città	E	1	L
Città	E	1	S

Operazione 1

Senza ridondanza

Concetto	Tipo	Accessi	Op
Persone	E	1	S
Residenza	R	1	S

Operazione 2

Concetto	Tipo	Accessi	Op
Città	E	1	L

Concetto	Tipo	Accessi	Op
Città	E	1	L
Residenza	R	5000	L

Mantenere o no la ridondanza?

- È importante considerare la frequenza delle operazioni:
- **Con ridondanza:**
 - **Operazione 1:** 1500 accessi in scrittura e 500 accessi in lettura al giorno
 - **Operazione 2:** 2 accessi in lettura al giorno
 - Totale: **2002 accessi al giorno**
- **Senza ridondanza:**
 - **Operazione 1:** 1000 accessi in scrittura al giorno
 - **Operazione 2:** 10002 accessi in lettura al giorno
 - Totale: **11002 accessi al giorno**
- Si decide pertanto di mantenere la ridondanza
- In generale si possono anche fare considerazioni sullo spazio in più richiesto...

Gerarchie: cosa conviene fare?

- La scelta fra le alternative si può fare con **metodo simile a quello visto per l'analisi delle ridondanze**, considerando oltre al numero degli accessi anche l'occupazione di spazio
 - Questo in quanto, ad es., la soluzione (3) basata su traduzione indipendente risulta sempre svantaggiata in termini di accessi, ma ha il vantaggio di generare relazioni con schemi più compatti
- A livello qualitativo, è possibile seguire alcune **semplici regole generali** (basate sul principio: **mantieni insieme ciò che viene usato insieme**)

Collasso verso l'alto: conviene se gli accessi al genitore e alle figlie sono contestuali

Collasso verso il basso: conviene se gli accessi alle figlie sono distinti (ma è possibile solo con generalizzazioni totali)

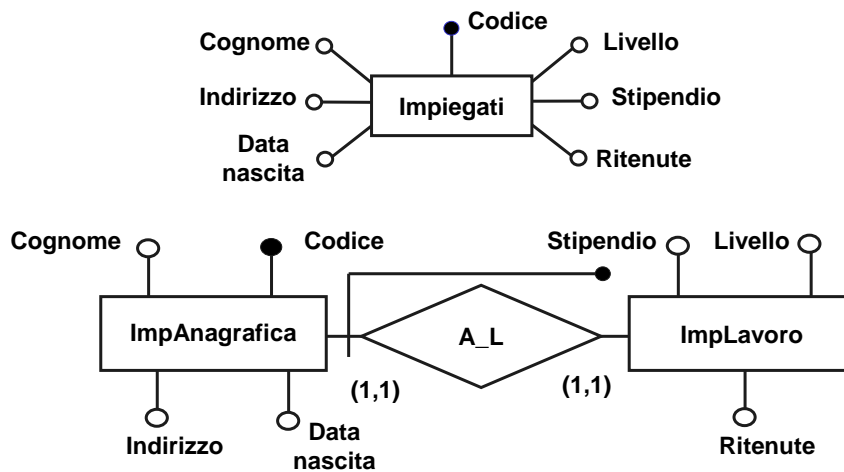
Traduzione indipendente: conviene se gli accessi alle entità figlie sono separati dagli accessi al genitore

Partizionamenti e accorpamenti

- È possibile ristrutturare lo schema accorpendo o partizionando entità e associazioni
- Tali ristrutturazioni vengono effettuate per rendere più efficienti le operazioni in base al principio già visto, ovvero:
 - separando attributi di un concetto che vengono acceduti separatamente
 - raggruppando attributi di concetti diversi acceduti insieme
- I casi principali sono:
 - **partizionamento "verticale" di entità**
 - verticale: si partizionano gli attributi
 - **partizionamento "orizzontale" di associazioni**
 - orizzontale: si partizionano le istanze
 - ad es.: storico e corrente (impieghi passati e impiego corrente)
 - possono cambiare i vincoli di cardinalità
 - **accorpamenti di entità e associazioni**
 - per le entità: caso particolare di traduzione di associazioni uno a uno

Partizionamento verticale di entità

- Si separano gli attributi in **gruppi omogenei dal punto di vista dell'uso**

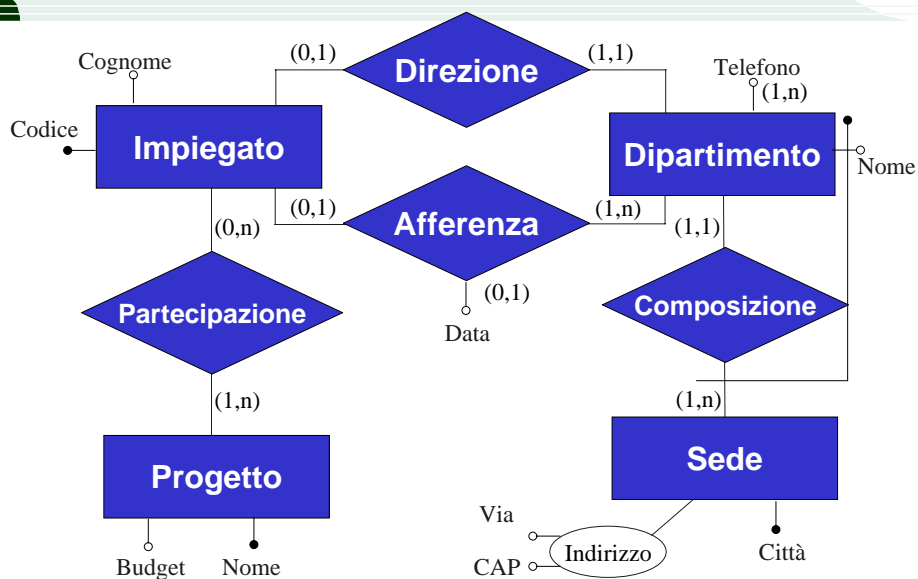


Prog. Logica: ottimizzazione

Sistemi Informativi T

17

Esempio di riferimento



Prog. Logica: ottimizzazione

Sistemi Informativi T

18

Schema logico relazionale

- Per le entità E che partecipano solo ad associazioni con $\text{max-card}(E,R) = n$ la traduzione è immediata:
Sede(Città, Via, CAP)
Progetto(Nome, Budget)
- Anche l'associazione Partecipazione si traduce immediatamente:
Partecipazione(Imiegato, Progetto)
- L'entità Dipartimento si può tradurre importando l'identificatore di Sede e inglobando l'associazione Direzione
Dipartimento(Nome, Città, Direttore)
Telefoni(Telefono, Dipartimento)
- Per tradurre l'associazione Afferenza, essendo pochi gli impiegati che non afferiscono a nessun dipartimento, si opta per una rappresentazione compatta
Impiegato(Codice, Cognome, Dipartimento*, Data*)

Osservazioni finali

- La progettazione logica, pur potendosi avvalere di strumenti software, non va svolta "alla cieca"; in presenza di diverse alternative occorre valutare:
 - La presenza o meno di valori nulli, e la loro incidenza, che dipende dal **volume dei dati**
 - Il modo con cui le **operazioni** navigano lo schema E/R, che poi si traduce in operazioni di join tra le relazioni che vengono create
- I casi visti non esauriscono l'argomento e lasciano sempre lo spazio per soluzioni specifiche "ad hoc"
- Ad esempio, associazioni uno a molti con $\text{max-card}(E_2,R) = K$, con K "piccolo", possono al limite essere tradotte con 1 sola relazione, prevedendo K repliche degli attributi di E2 (es. tipico: numeri di telefono)
- Va ricordato che i vincoli che non si riescono a rappresentare mediante il DDL di SQL devono comunque essere imposti, facendo uso di altri strumenti (trigger, transazioni, ecc.)