

Sappiamo bene però che il linguaggio naturale è fonte di ambiguità e fraintendimenti. È molto importante quindi effettuare una profonda analisi del testo che descrive le specifiche per filtrare le eventuali inesattezze e i termini ambigui presenti. Per fissare alcune regole pratiche da seguire in questa attività faremo riferimento a un semplice esempio. Supponiamo di dover progettare una base di dati per una società di formazione e di aver raccolto, sulla base di alcune interviste fatte al personale di questa società, le specifiche dei dati espresse in linguaggio naturale riportate in figura 7.1. Si noti che abbiamo acquisito in questa fase anche informazioni sul carico previsto dei dati a regime.

Società di formazione	
1	Si vuole realizzare una base di dati per una società che eroga cor-
2	si, di cui vogliamo rappresentare i dati dei partecipanti ai corsi e
3	dei docenti. Per i partecipanti (circa 5000), identificati da un codi-
4	ce, si vuole memorizzare il codice fiscale, il cognome, l'età, il sesso,
5	il luogo di nascita, il nome dei loro attuali datori di lavoro, i posti
6	dove hanno lavorato in precedenza insieme al periodo, l'indirizzo e
7	il numero di telefono, i corsi che hanno frequentato (i corsi sono in
8	tutto circa 200) e il giudizio finale. Rappresentiamo anche i seminari
9	che stanno attualmente frequentando e, per ogni giorno, i luoghi e
10	le ore dove sono tenute le lezioni. I corsi hanno un codice, un titolo
11	e possono avere varie edizioni con date di inizio e fine e numero di
12	partecipanti. Se gli studenti sono liberi professionisti, vogliamo co-
13	nosocere l'area di interesse e, se lo possiedono, il titolo. Per quelli
14	che lavorano alle dipendenze di altri, vogliamo conoscere invece il
15	loro livello e la posizione ricoperta. Per gli insegnanti (circa 300),
16	rappresentiamo il cognome, l'età, il posto dove sono nati, il nome del
17	corso che insegnano, quelli che hanno insegnato nel passato e quel-
18	li che possono insegnare. Rappresentiamo anche tutti i loro recapiti
19	telefonici. I docenti possono essere dipendenti interni della società o
20	collaboratori esterni.

Figura 7.1 Esempio di requisiti espressi in linguaggio naturale

È facile rendersi conto che tale testo presenta un certo numero di ambiguità e imprecisioni. Per esempio si utilizzano i termini *partecipante* e *studente* per indicare lo stesso concetto. La stessa cosa accade per i termini *docente* e *professore* e per i termini *corso* e *seminario*.

Proviamo a fissare alcune regole generali per ottenere una specifica dei requisiti più precisa e senza ambiguità.

- **Scegliere il corretto livello di astrazione.** È bene evitare di utilizzare termini troppo generici o troppo specifici che rendono poco chiaro un concetto. Per esempio, nel nostro caso sono stati utilizzati i termini *titolo* (a riga 13), con riferimento ai partecipanti che sono liberi professionisti (che tra l'altro è utiliz-

risponde nessuna relazione (l'associazione CONTRATTO nell'esempio in questione).

Come esempio finale, in figura 8.29 viene riportata la rappresentazione dello schema relazionale ottenuto nel paragrafo 8.3.5. I legami logici tra le varie relazioni possono essere ora facilmente identificati.

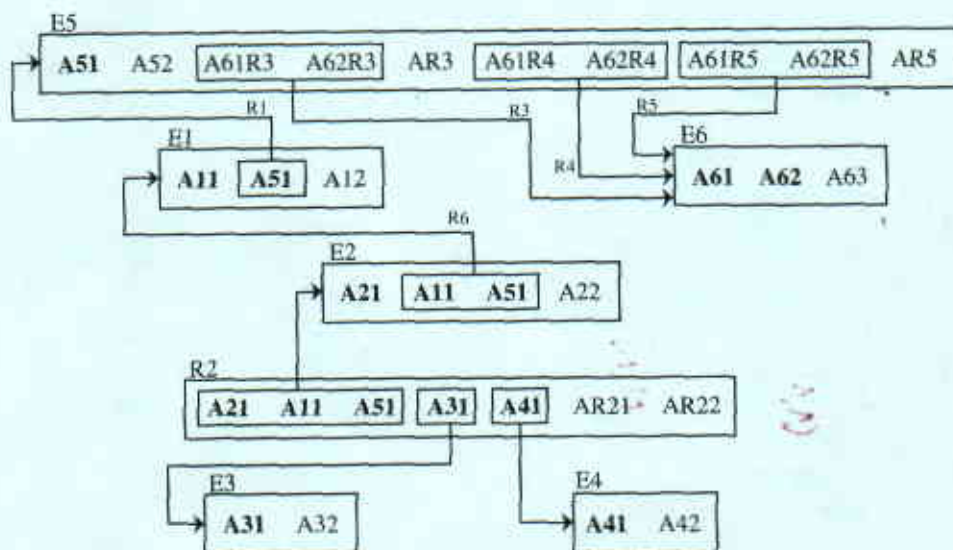


Figura 8.29 Rappresentazione grafica dello schema ottenuto nel paragrafo 8.3.5

Nell'appendice A verrà mostrato come una variante di questo formalismo grafico viene adottato da un sistema di gestione di basi di dati reale (Access), sia per rappresentare schemi di relazione che per esprimere graficamente operazioni di join.

8.4 Un esempio di progettazione logica

Riprendiamo l'esempio presentato nel capitolo precedente relativo alla base di dati della società di formazione, il cui schema concettuale viene riportato, per comodità, in figura 8.30. Le varie ristrutturazioni che discuteremo sono riportate nello schema finale in figura 8.33.

Sui dati descritti da questo schema erano state previste le seguenti operazioni.

Operazione 1: inserisci un nuovo partecipante indicando tutti i suoi dati.

Operazione 2: assegna un partecipante a una edizione di corso.

Operazione 3: inserisci un nuovo docente indicando tutti i suoi dati e i corsi che può insegnare.

Operazione 4: assegna un docente abilitato a una edizione di un corso.

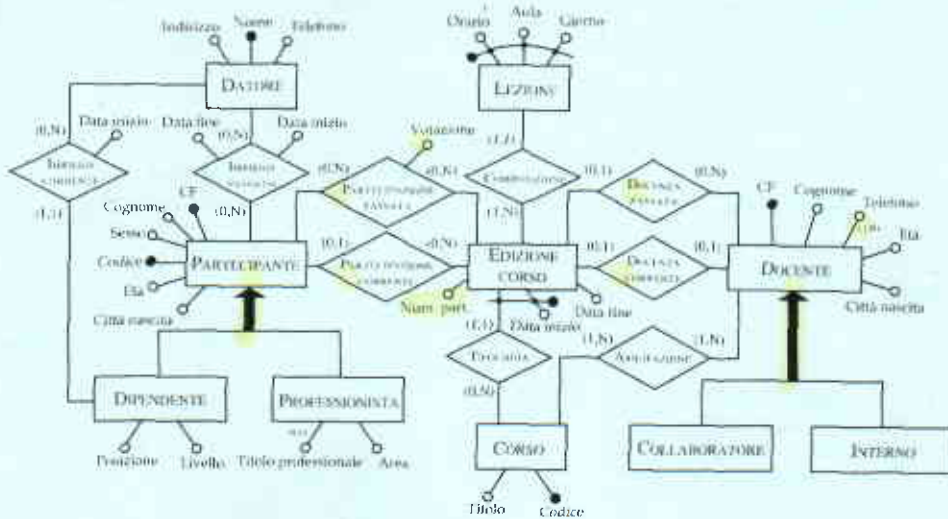


Figura 8.30 Lo schema E-R di una società di formazione

Operazione 5: stampa tutte le informazioni sulle edizioni passate di un corso con titolo, orari delle lezioni e numero dei partecipanti.

Operazione 6: stampa tutti i corsi offerti, con informazioni sui docenti che possono insegnarli.

Operazione 7: per ogni docente, trova i partecipanti a tutti i corsi da lui insegnati.

Operazione 8: effettua una statistica su tutti i partecipanti a un corso con tutte le informazioni su di essi, sulla edizione alla quale hanno partecipato e la rispettiva votazione.

8.4.1 Fase di ristrutturazione

Supponiamo che i dati di carico siano quelli riportati in figura 8.31. Eseguiamo, sulla base di questi dati, i vari passi della ristrutturazione.

Analisi delle ridondanze C'è un solo dato ridondante nello schema: l'attributo **Numero di partecipanti in EDIZIONE CORSO** che può essere derivato dalle associazioni **PARTECIPAZIONE CORRENTE** e **PARTECIPAZIONE PASSATA**. Questo dato richiede un quantitativo di memoria pari a $4 \times 1.000 = 4000$ bytes, avendo assunto che sono necessari 4 byte per ogni occorrenza di **EDIZIONE CORSO** per memorizzare il numero di partecipanti. Le operazioni coinvolte con questo dato sono la 2, la 5 e la 8. L'ultima di queste può essere trascurata perché si tratta di una operazione non frequente ed eseguita in modalità batch. Proviamo a valutare il costo delle operazioni 2 e 5 in caso di presenza e in assenza di dato ridondante. Possiamo dedurre dalla tavola dei volumi che ogni edizione di corso ha, in media,

Tavola dei volumi

Concetto	Tipo	Volume
Lezione	E	8000
Edizione corso	E	1000
Corso	E	200
Docente	E	300
Collaboratore	E	250
Interno	E	50
Partecipante	E	5000
Dipendente	E	4000
Professionista	E	1000
Datore	E	8000
Part. passata	R	10000
Part. corrente	R	500
Composizione	R	8000
Tipologia	R	1000
Doc. passata	R	900
Doc. corrente	R	100
Abilitazione	R	500
Impiego corrente	R	4000
Impiego passato	R	1000

Tavola delle operazioni

Operazione	Tipo	Frequenza
Op. 1	I	40/giorno
Op. 2	I	50/giorno
Op. 3	I	2/giorno
Op. 4	I	15/giorno
Op. 5	I	10/giorno
Op. 6	I	20/giorno
Op. 7	I	5/sett.
Op. 8	B	10/mese

Figura 8.31 Tavole dei volumi e delle operazioni per lo schema in figura 8.30

8 lezioni e 10 partecipanti. Da questi dati sono facilmente calcolabili le tavole degli accessi riportate in figura 8.32.

Tavole degli accessi in presenza di ridondanza

Operazione 2			
Concetto	Costr.	Acc.	Tipo
Partecipante	E	1	L
Par. corrente	R	1	S
Ediz. corso	E	1	L
Ediz. corso	E	1	S

Operazione 5			
Concetto	Costr.	Acc.	Tipo
Ed. corso	E	1	L
Tipologia	R	1	L
Corso	E	1	L
Composiz.	R	8	L
Lezione	E	8	L

Tavole degli accessi in assenza di ridondanza

Operazione 2			
Concetto	Costr.	Acc.	Tipo
Partecipante	E	1	L
Par. corrente	R	1	S

Operazione 5			
Concetto	Costr.	Acc.	Tipo
Ediz. corso	E	1	L
Tipologia	R	1	L
Corso	E	1	L
Composiz.	R	8	L
Lezione	E	8	L
Par. corrente	R	10	L

Figura 8.32 Tavole degli accessi per lo schema in figura 8.30

Da queste risulta:

- Dato ridondante presente: per l'operazione 2 abbiamo $2 \times 50 = 100$ accessi in lettura e altrettanti in scrittura al giorno mentre, per l'operazione 5, abbiamo $19 \times 10 = 190$ accessi in lettura al giorno, per un totale di 490 accessi giornalieri (avendo contato doppie le operazioni di scrittura);
- Dato ridondante assente: per l'operazione 2 abbiamo 50 accessi in lettura e altrettanti in scrittura al giorno, mentre, per l'operazione 5, abbiamo $29 \times 10 = 290$ accessi in lettura al giorno, per un totale di 440 accessi giornalieri (avendo contato doppie le operazioni di scrittura);

Abbiamo quindi, in presenza di ridondanza, degli svantaggi sia in termini di memoria che di efficienza. Decidiamo quindi di eliminare l'attributo ridondante Numero di partecipanti dalla relazione EDIZIONE CORSO.

Eliminazione delle gerarchie Nello schema sono presenti due gerarchie: quella relativa ai docenti e quella relativa ai partecipanti. Per i docenti si può notare che le operazioni che li riguardano, ovvero la 3, la 4, la 6 e la 7, non fanno distinzioni tra collaboratori esterni e dipendenti interni della società. Tra l'altro, le entità corrispondenti non hanno attributi specifici che li distinguono. Decidiamo

Due altre possibili ristrutturazioni che si può pensare di effettuare, proprio in conseguenza a quanto detto sulle edizioni dei corsi, sono l'accorpamento delle associazioni **DOCENZA PASSATA** e **DOCENZA CORRENTE** e delle associazioni analoghe **PARTECIPAZIONE PASSATA** e **PARTECIPAZIONE CORRENTE**. Si tratta infatti, in entrambi i casi, di due concetti simili (l'unica differenza è di carattere temporale) tra i quali alcune operazioni non fanno differenza (la 7 e la 8). Il loro accorpamento produrrebbe un altro beneficio: non sarebbe necessario trasferire occorrenze da una associazione a un'altra quando una edizione di corso termina. Per le partecipazioni ai corsi, un inconveniente è la presenza dell'attributo **Votazione** che non si applica alle partecipazioni correnti e quindi provocherebbe la presenza di valori nulli. Del resto, la tavola dei volumi ci dice che il numero medio di occorrenze dell'entità **PARTECIPAZIONE CORRENTE** è 500 e quindi, supponendo di aver bisogno di 4 byte per memorizzare la votazione, lo spreco di memoria sarebbe di soli due Kbyte. Decidiamo quindi di accorpare le due coppie di relazioni come descritto in figura 8.33. Va aggiunto il vincolo non esprimibile dallo schema che un docente non può insegnare più di una edizione di corso nello stesso periodo e, analogamente, il vincolo che un partecipante non può seguire più di un corso nello stesso periodo.

Infine, bisogna eliminare l'attributo multivalore **Telefono** associato all'entità **DOCENTE**. Per far questo, introduciamo una nuova entità **TELEFONO** legata da una associazione uno a molti con l'entità **DOCENTE**, che viene privata del relativo attributo.

È interessante osservare che le decisioni prese in questa fase ribaltano, in qualche maniera, decisioni prese in fase di progettazione concettuale. Questo però non deve sorprenderci: l'obiettivo della progettazione concettuale è solo quello rappresentare nella maniera migliore la realtà di interesse, mentre nella progettazione logica dobbiamo cercare di ottimizzare le prestazioni ed è quasi inevitabile dover rivedere le decisioni prese.

Scelta degli identificatori principali Solo l'entità **PARTECIPANTE** presenta due identificatori: il codice fiscale e il codice interno. Tra i due è certamente preferibile scegliere il secondo. Infatti, un codice fiscale richiede 16 byte di memoria mentre un codice interno, che serve a distinguere al più 5000 occorrenze (vedi tavola dei volumi), richiede non più di 2 byte.

C'è in effetti un'altra considerazione di carattere pragmatico da fare sugli identificatori e che riguarda l'entità **EDIZIONE CORSO**. Questa entità è identificata dall'attributo **Data inizio** e dall'entità **CORSO**. Ne risulta un identificatore piuttosto pesante che, in una rappresentazione relazionale, deve essere usato per rappresentare due associazioni (**PARTECIPAZIONE** e **DOCENZA**) con molte occorrenze. Si può osservare però che ogni corso ha un codice e che, in media, il numero di edizioni di un corso è pari a cinque. Questo significa che è sufficiente aggiungere un intero di una cifra al codice di un corso per avere un identificatore delle edizioni dei corsi, operazione che può essere fatta durante la creazione di una nuova edizione in maniera piuttosto efficiente e sicura. Da questa discussione risulta che è conveniente definire un nuovo identificatore per le edizioni dei corsi

che rimpiazza l'identificatore esterno precedente. Questo è un esempio di analisi e ristrutturazione che non rientra in nessuna delle categorie generali viste ma che, nei casi pratici, capita di incontrare.

Abbiamo con questo terminato la fase di ristrutturazione dello schema E-R originale. Lo schema risultante è quello in figura 8.33.

8.4.2 Traduzione verso il relazionale

Seguendo la strategia di traduzione descritta in questo capitolo, lo schema E-R in figura 8.33 può essere tradotto nel seguente schema relazionale.

EDIZIONE CORSO(Codice, DataInizio, DataFine, Corso, Docente)
 LEZIONE(Ora, Aula, Giorno, EdizioneCorso)
 DOCENTE(CF, Cognome, Età, CittàNascita, Tipo)
 TELEFONO(Numero, Docente), CORSO(Codice, Nome)
 ABILITAZIONE(Corso, Docente)
 PARTECIPANTE(Codice, CF, Cognome, Età, CittàNascita, Sesso)
 PARTECIPAZIONE(Partecipante, EdizioneCorso, Votazione*)
 DATORE(Nome, Telefono, Indirizzo)
 IMPIEGOPASSATO(Partecipante, Datore, DataInizio, DataFine)
 PROFESSIONISTA(Partecipante, Area, Titolo*)
 DIPENDENTE(Partecipante, Livello, Posizione, Datore, DataInizio)

Lo schema logico ottenuto va naturalmente completato con una documentazione di supporto che descriva, tra l'altro, tutti i vincoli di integrità referenziale che sussistono tra le varie relazioni. Questo può essere fatto usando la notazione grafica introdotta nel paragrafo 8.3.7.

8.5 Progettazione logica con gli strumenti CASE

La fase di progettazione logica viene generalmente supportata da tutti gli strumenti CASE di ausilio allo sviluppo di basi di dati. In particolare, trattandosi di una operazione basata su criteri precisi, la fase di traduzione verso il modello relazionale viene effettuata da questi sistemi in maniera pressoché automatica. La fase di ristrutturazione dello schema che precede la traduzione vera e propria è invece difficilmente automatizzabile e i vari prodotti non la supportano o lo fanno solo parzialmente, ricorrendo a soluzioni semplificate. Per esempio, alcuni sistemi traducono automaticamente tutte le generalizzazioni secondo uno solo dei metodi descritti nel paragrafo 8.2.2. Abbiamo visto però che la ristrutturazione di schemi E-R è un momento importante della progettazione perché affronta alcune problematiche (analisi delle ridondanze e trasformazioni orientate all'ottimizzazione) che è possibile risolvere prima di effettuare la traduzione e che non sono di