

# Progetto per il Corso di BASI DI DATI

## Laboratorio di diagnosi mediche

Versione del 26 Ottobre 2001

Questo documento presenta un esempio sostanzialmente completo di progettazione di base di dati, relativamente ad un laboratorio di diagnosi medica.

Il punto di partenza è costituito da un insieme di specifiche, che saranno preliminarmente analizzate e filtrate. La progettazione concettuale è effettuata sulla base di queste nuove specifiche. Vengono analizzate le operazioni previste, e studiato il carico della base di dati. Viene poi effettuata la progettazione logica nelle sue due fasi; nella prima, indipendente dal modello logico, vengono analizzate le ridondanze presenti, eliminate le gerarchie, effettuati partizionamenti/agggregazioni di concetti e scelti gli identificatori principali; nella seconda, viene effettuata la traduzione verso il modello relazionale. Sono state infine realizzate alcune delle operazioni previste in SQL.

L'esempio è tratto dal libro: "Progetti ed Esercizi di Basi di Dati" di C. Batini, L. Cabibbo, R. Torlone.

# 1 Descrizione e specifiche

Si vuole sviluppare il progetto della base di dati di un laboratorio di diagnosi medica, partendo da un insieme di requisiti. Le fasi da svolgere vanno dall'analisi dei requisiti, alle varie fasi dell'analisi, fino all'implementazione delle operazioni previste. Durante il progetto è necessario produrre un insieme di documenti, che costituiscono appunto la *documentazione* del progetto:

- Analisi dei requisiti.
- Lo schema concettuale, tramite il modello E-R, presentato a diversi gradi di raffinamento
- Una descrizione delle operazioni previste, e le tavole di carico.
- Lo schema ottenuto per ristrutturazione dalla prima fase della progettazione logica. Lo schema logico finale.
- Un listato delle interrogazioni e delle istruzioni (aggiornamenti, inserimenti, cancellazioni) SQL relative alle operazioni previste.
- contenuto di test della base di dati e nella stampa dei risultati delle interrogazioni su tali dati.

Nei prossimi paragrafi vengono descritte le specifiche ottenute mediante intervista ai responsabili del laboratorio.

Nelle sezioni successive vengono svolte le varie attività del progetto, talvolta interamente, altre volte fornendo solo una parte della documentazione richiesta.

## Specifiche sui dati

Si vuole progettare il sistema informativo di un laboratorio di diagnosi medica.

Diversi tipi di persone sono coinvolte nel laboratorio: medici, assistenti, pazienti. Per i pazienti, rappresentiamo alcuni dati anagrafici, quali il nome, il cognome, l'età, l'indirizzo, il telefono ed il codice fiscale (che li identifica). Per i medici e gli assistenti, oltre ai dati anagrafici, abbiamo un codice interno che li identifica.

I clienti del laboratorio (circa 100.000) hanno bisogno di visite mediche e/o analisi che vanno riservate in anticipo, fissando data ed ora. La storia delle analisi e delle visite degli ultimi dodici mesi deve essere memorizzata nel sistema.

Le prestazioni offerte dal laboratorio appartengono a varie tipologie, identificate da un codice e caratterizzate da una descrizione. Anche i pazienti vengono classificati rispetto a diverse tipologie identificate da un codice e caratterizzate da una descrizione. Ogni tipo di prestazione ha un costo che dipende dal tipo di paziente.

Ogni dottore (circa 150) può effettuare solo determinati tipi di analisi e visite. Ogni assistente (circa 300) può effettuare solo determinati tipi di analisi. Le analisi (circa 200 al giorno) e le visite (circa 100 al giorno) sono effettuate in apposite stanze.

Ogni prestazione offerta ha un esito, caratterizzato da una descrizione, una data ed un prezzo. L'esito di ogni analisi va approvato con il nome di un dottore. Gli esiti delle analisi e delle visite devono essere memorizzati in una cartella del paziente, che registra la storia delle ultime 30 visite e/o analisi. Di ogni cartella va memorizzata la data di apertura. Le prestazioni possono essere effettuate o come esito di altre prestazioni o indipendentemente.

Per gli assistenti, che sono dipendenti del laboratorio, vogliamo rappresentare il loro livello e lo stipendio. Per i dottori, che sono invece considerati consulenti del laboratorio, rappresentiamo un valore percentuale per il calcolo delle parcelle, la specializzazione, l'ente di appartenenza e la disponibilità settimanale.

Il laboratorio rilascia delle fatture per gli esiti di analisi e visite. Una fattura può riferirsi a diverse prestazioni di uno stesso cliente.

## Specifiche sulle operazioni

Per il laboratorio di diagnosi medica sono previste alcune operazioni, di cui riportiamo una breve descrizione ed il carico previsto.

- O1 Introduci un nuovo paziente (frequenza: 50 al giorno).
- O2 Modifica i dati di un paziente già memorizzato (frequenza: 10 al giorno).
- O3 Prenota una visita o un'analisi tenendo conto delle disponibilità (frequenza: 500 al giorno).
- O4 Modifica una prenotazione (orario e/o data) (frequenza: 100 al giorno).
- O5 Produci l'esito di una visita o di un'analisi con aggiornamento della cartella clinica del paziente (frequenza: 300 al giorno).
- O6 Stampa la cartella clinica di un paziente (frequenza: 100 al giorno).
- O7 Cambia i prezzi delle visite o delle analisi (frequenza: 10 al mese).
- O8 Genera una fattura per un cliente (frequenza: 300 al giorno).
- O9 Prepara una statistica sul numero di pazienti analizzati da ogni dottore ed ogni assistente in un mese (frequenza: 1 al mese).
- O10 Prepara un prospetto delle fatture emesse in un mese, con nome del paziente ed importo totale (frequenza: 1 al mese).

## 2 Analisi dei requisiti

In questo paragrafo ci occupiamo della fase di analisi e ristrutturazione dei requisiti raccolti, producendo un insieme omogeneo e non ambiguo di specifiche da utilizzare nelle fasi successive della progettazione.

*Viene omessa una descrizione più dettagliata di questa attività.*

Nella fase di analisi, vogliamo individuare i termini più rilevanti, eventuali sinonimi e/o omonimi, le correlazioni tra i vari termini e la presenza di eventuali ambiguità. Per fare questo è utile la costruzione di un *glossario dei termini* nel quale, ad ogni termine rilevante individuato nelle specifiche, associamo una breve descrizione, eventuali sinonimi e/o omonimi ed i termini ad essi collegati logicamente. Una porzione del glossario dei termini per l'applicazione in esame è riportato in Figura 1.

A questo punto è utile riorganizzare le specifiche, partizionandole in gruppi di frasi che fanno riferimento ad informazioni omogenee. In questa fase cerchiamo di filtrare le ambiguità presenti ed utilizzare una terminologia più accurata facendo uso del glossario (ad esempio, decidiamo di utilizzare solo il termine *medico* e non più *dottore*).

Otteniamo, al termine di questa fase, le specifiche che seguono.

### Dati di carattere generale

Si vuole progettare il sistema informativo di un laboratorio di diagnosi medica. Diversi tipi di persone sono coinvolte nel laboratorio: pazienti e personale del laboratorio.

Termine	Descrizione	Sinonimi	Termini collegati
Paziente	Cliente del laboratorio di diagnosi medica.	Cliente	Tipologia del Paziente
Personale Laboratorio	Le persone che lavorano per il laboratorio, ovvero medici e assistenti.		Medico, Assistente
Medico	Medico che lavora nel laboratorio.	Dottore	Analisi, Visita
Assistente	Assistente che lavora nel laboratorio.		Analisi
Prestazione	Visita medica e/o analisi effettuata nel laboratorio. Va riservata in anticipo.		Esito
Esito	Esito di una prestazione effettuata. Va approvata da un medico.		Prestazione, Medico, Fattura
Fattura	Fattura rilasciata per una o più prestazioni.		Paziente, Esito

Figura 1: Porzione del glossario dei termini

### Dati sui pazienti

Per i **pazienti** (circa 100000), rappresentiamo alcuni dati anagrafici, quali il nome, il cognome, l'età, l'indirizzo, il telefono ed il codice fiscale (che li identifica). I pazienti vengono classificati rispetto a diverse tipologie, identificate da un codice e caratterizzate da una descrizione.

### Dati sul personale del laboratorio

Il **personale del laboratorio** è composto da medici ed assistenti. Per i componenti del personale, oltre ai dati anagrafici, abbiamo un codice interno che li identifica.

### Dati sugli assistenti

Per gli **assistenti** (circa 300), che sono dipendenti del laboratorio, vogliamo rappresentare il loro livello e lo stipendio. Ogni assistente può effettuare solo determinati tipi di analisi.

### Dati sui medici

Per i **medici** (circa 150), che sono considerati consulenti del laboratorio, rappresentiamo un valore percentuale per il calcolo delle parcelle, la specializzazione, l'ente di appartenenza e la disponibilità settimanale. Ogni medico può effettuare solo determinati tipi di analisi e/o visite.

### Dati sulle prestazioni

I pazienti hanno bisogno di **prestazioni** ovvero visite mediche (circa 200 al giorno) e/o analisi (circa 100 al giorno). Le prestazioni vanno riservate in anticipo, fissando data ed ora. La storia delle prestazioni offerte degli ultimi dodici mesi deve essere memorizzata nel sistema. Le varie prestazioni appartengono a varie tipologie, identificate da un codice e caratterizzate da una descrizione. Ogni tipo di prestazione ha un prezzo che dipende dal tipo di paziente. Le prestazioni sono effettuate in apposite stanze. Le prestazioni possono essere effettuate o come esito di altre prestazioni o indipendentemente.

### Dati sugli esiti

Ogni prestazione ha un **esito**, caratterizzato da una descrizione, una data ed un prezzo. L'esito di ogni analisi va approvato con un nome di un medico. Gli esiti delle prestazioni devono essere memorizzate in una cartella del paziente, che registra la storia delle ultime 30 visite e/o analisi. Di ogni cartella va memorizzata la data di apertura.

### Dati sulle fatture

Il laboratorio rilascia delle **fatture** per gli esiti di analisi e/o visite. Una fattura può riferirsi a diverse prestazioni di uno stesso paziente.

## 3 Progettazione concettuale

Sulla base dell'analisi fatta nel paragrafo precedente è quasi immediato il disegno di un primo schema scheletro, che riportiamo in Figura 2.



Figura 2: Schema scheletro

Procedendo a macchia d'olio si arriva prima ad uno schema intermedio, riportato in Figura 3. Si è scelto di distinguere le prestazioni effettuate, che sono un sottoinsieme delle prestazioni (nel glossario il termine *Prestazione* è associato sia a quelle effettuate che a quelle prenotate e non ancora effettuate). Il personale di laboratorio è stato distinto in assistenti e medici, tramite una gerarchia di generalizzazione di tipo totale ed esclusivo. Abbiamo scelto di rappresentare le tipologie di pazienti e quelle di prestazioni tramite entità piuttosto che per mezzo di attributi.

Lo schema finale è riportato in Figura 4. In questo schema sono stati introdotti i concetti relativi alle fatture. Si è fatta una distinzione tra tipi di prestazioni (visite ed analisi), sia a livello di tipologie di prestazioni che a livello di prestazioni effettuate. (Tale distinzione non si è ritenuta importante per le prestazioni prenotate ma non effettuate.) Notiamo che è così possibile rappresentare il fatto che gli assistenti possono essere abilitati (e quindi possono effettuare) solo a certi tipi di analisi (ma a nessun tipo di visita) mentre i medici possono essere abilitati a qualsiasi tipo di prestazione. Infine, è stato rappresentato il concetto dell'approvazione degli esiti tramite una relazione tra ESITO e MEDICO.

Alcuni vincoli richiesti dalle specifiche non possono essere espressi dallo schema E-R; le annotazioni riportate in Figura 5 completano la documentazione relativa alla progettazione concettuale. Va osservato che molti di questi vincoli sono stati introdotti in corrispondenza a cicli di relazioni, per richiedere che l'informazione ottenuta seguendo due diversi cammini che congiungono le medesime entità sia non contraddittoria. (Ad esempio, il primo vincolo richiede che la navigazione da ESITO verso PAZIENTE tramite PRESTAZIONE porti alle medesime informazioni che non la navigazione tramite CARTELLA CLINICA.)

Il *dizionario dei dati* completa la documentazione della progettazione concettuale. Esso è composto da una tabella che descrive le entità dello schema E-R, con la sua descrizione, la descrizione degli attributi e degli identificatori, e da una tabella che descrive le relazioni dello schema E-R, con

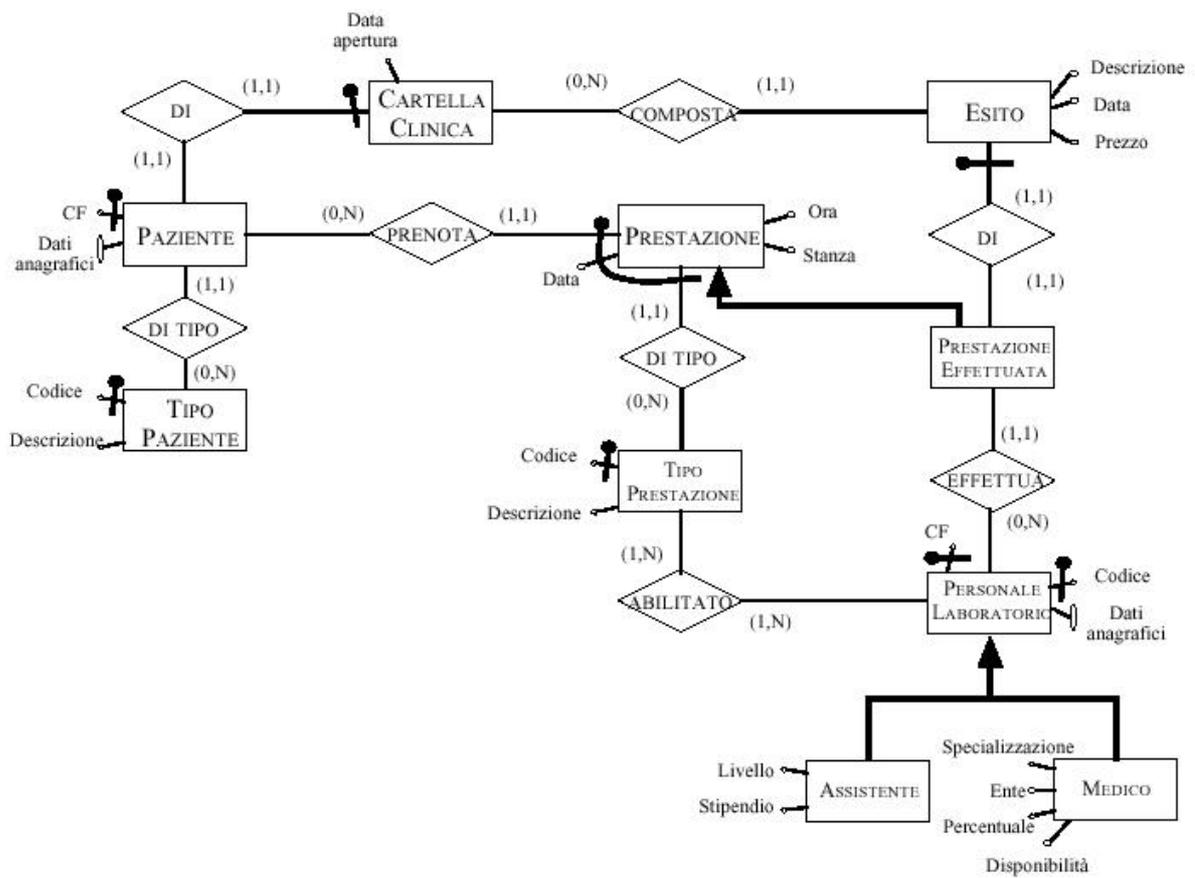


Figura 3: Schema intermedio

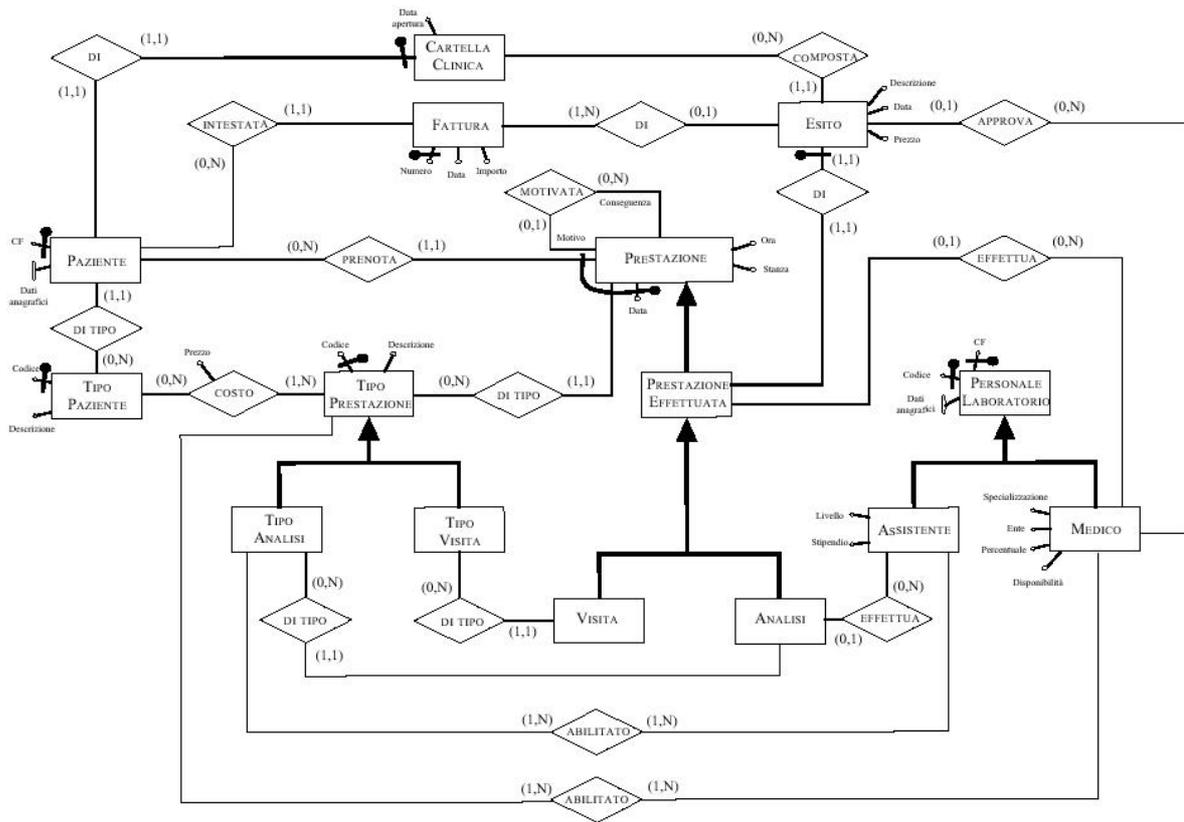


Figura 4: Schema E-R finale

**Esiti nelle Cartelle cliniche:** tutti gli esiti che compongono una cartella clinica sono relativi al paziente intestatario di tale cartella clinica.

**Esiti nelle Fatture:** tutti gli esiti che compongono una fattura sono relativi al paziente intestatario di tale fattura.

**Tipo di una prestazione effettuata:** l'informazione sul tipo di una prestazione che si raggiunge navigando da PRESTAZIONE verso TIPO PRESTAZIONE tramite la relazione DI TIPO è la stessa, per una prestazione effettuata — sia essa una visita o analisi — che si ottiene navigando da VISITA verso TIPO VISITA tramite la relazione DI TIPO, ovvero la stessa che si ottiene navigando da ANALISI verso TIPO ANALISI tramite la relazione DI TIPO.

**Prestazioni effettuate da un personale abilitato:** se un medico o un assistente ha effettuato una prestazione, tale medico o assistente deve essere abilitato ad effettuare il tipo di quella prestazione.

**Prestazioni effettuate da una sola persona:** una Prestazione effettuata è stata effettuata esattamente da una sola persona, sia esso medico o assistente.

Figura 5: Vincoli non espressi dallo schema E-R

Entità	Descrizione	Attributi	Identificatore
Paziente	Cliente del laboratorio di diagnosi medica	Codice fiscale, Nome, Cognome, Età, Indirizzo, Telefono	Codice fiscale
Tipo paziente	Tipologie dei clienti	Codice, Descrizione	Codice
Prestazione	Visita o analisi prenotata e/o effettuata	Data, Ora, Stanza	Data, Paziente (tramite la relazione Prenota), Tipo Prestazione (tramite la relazione Di Tipo)
Prestazione Effettuata	Sottoinsieme delle prestazioni	<i>vedi</i> Prestazione	<i>vedi</i> Prestazione
Fattura	Fattura rilasciata ai pazienti	Numero, Data, Importo	Numero
Cartella clinica	Cartella clinica di un paziente	Data apertura	Paziente (tramite la relazione Di)
Esito	Esito di una visita o di una analisi	Descrizione, Data, Prezzo	Prestazione (tramite la relazione Di)
Tipo Prestazione	Tipologia prestazione medica	Codice, Descrizione	Codice
Personale Laboratorio	Medici ed assistenti del laboratorio	Codice, Codice Fiscale, Dati anagrafici	Codice e Codice Fiscale
Medico	Medici del laboratorio	Specializzazione, Ente, Percentuale, Disponibilità	<i>vedi</i> Personale
Assistente	Assistenti del laboratorio	Livello, Stipendio	<i>vedi</i> Personale

Figura 6: Porzione del dizionario dati — Entità

le entità partecipanti e gli eventuali attributi. Porzioni del dizionario dei dati per il laboratorio di diagnosi medica sono riportate nelle Figure 6 e 7.

## 4 Analisi delle funzioni

Prima di procedere con la progettazione logica della base di dati, è necessario valutare il carico della applicazione, caratterizzando sia il volume atteso dei dati memorizzati che il profilo delle operazioni che devono essere realizzate, ovvero come e con che frequenza esse accedono i dati memorizzati.

Per quanto riguarda il primo aspetto (legato solo ai dati) facciamo riferimento alla *tavola dei volumi*, di cui una porzione è riportata in Figura 8.

Questa tavola riporta per ciascun concetto dello schema E-R (entità, relazioni, attributi) il numero atteso di istanze che quel concetto avrà in un funzionamento di regime dell'applicazione. (Il *tipo* di un *concetto* può assumere il valore E, R oppure A, indicando rispettivamente una entità, una relazione oppure un attributo.)

La caratterizzazione delle operazioni è più complessa. La *tavola delle frequenze* (riportata in Figura 9) descrive la frequenza attesa con cui le varie operazioni saranno eseguite a regime. (Qui il *tipo* di una operazione può assumere il valore OL oppure B, a secondo che l'operazione vada eseguita interattivamente — ovvero On Line — oppure fuori linea — ovvero Batch.)

Per ciascuna operazione prevista è necessario descriverne lo *schema di operazione*, ovvero l'insieme dei concetti dello schema E-R utilizzati dall'operazione (che viene descritto ancora come uno schema E-R). Un esempio di schema di operazione è riportato in Figura 10, in riferimento all'operazione **O8**.

Relazione	Entità Partecipanti	Descrizione	Attributi
Costo	Tipo Prestazione, Tipo Paziente	Il prezzo di una prestazione dipende dalla tipologia della prestazione e da quella del paziente	Prezzo
Di Tipo (Paziente)	Paziente, Tipo Paziente	Associa ogni paziente con la tipologia attuale	
Prenota	Paziente, Prestazione	Ogni prestazione è stata prenotata da un paziente	
Intestata	Paziente, Fattura	Associa ogni fattura con l'intestatario	
(Cartella clinica) Di	Paziente, Cartella Clinica	Ogni paziente ha una cartella clinica	
Composta	Cartella Clinica, Esito	La cartella clinica di un paziente è composta dagli esiti delle prestazioni effettuate da quel paziente	
Approva	Medico, Esito	Ogni esito deve essere approvato da un medico	

Figura 7: Porzione del dizionario dati — Relazioni

Concetto	Tipo	Volume
Paziente	E	100.000
Cartella Clinica	E	100.000
Tipo Paziente	E	20
Prestazione	E	400.000
Tipo Prestazione	E	200
Fattura	E	150.000
Esito	E	320.000
Di (Esito)	R	300.000
Di (Cartella Clinica)	R	100.000
Intestata	R	150.000

Figura 8: Porzione della tavola dei volumi

Operazione	Descrizione	Frequenza	Tipo
O1	Introduci nuovo paziente	50/giorno	OL
O2	Modifica dati paziente	10/giorno	OL
O3	Prenota prestazione	500/giorno	OL
O4	Modifica prenotazione	100/giorno	OL
O5	Esito prestazione	300/giorno	OL
O6	Stampa cartella clinica	100/giorno	OL
O7	Modifica prezzi	10/mese	OL
O8	Genera/Stampa fattura	300/giorno	OL
O9	Statistica pazienti	1/mese	B
O10	Stampa prospetto fatture	1/mese	B

Figura 9: Tavola delle frequenze

Sarebbe poi utile la descrizione degli algoritmi che implementano le operazioni, e che indicano il modo con cui i dati dello schema di operazione vengono visitati ed i controlli di integrità che devono essere effettuati.

Una ulteriore caratterizzazione delle operazioni è data dalla *tavola degli accessi*, che descrive per ciascuna operazione il numero di istanze dei vari concetti acceduti (su base giornaliera, e specificando se l'accesso sia di lettura o di scrittura).<sup>1</sup>

## 5 Progettazione logica indipendente dal modello

Effettuiamo ora la progettazione logica della base di dati. In questo paragrafo ci occupiamo della fase alta della progettazione logica, ovvero quella indipendente dal modello dei dati scelto, suddivisa in analisi delle ridondanze, eliminazione delle gerarchie, partizionamento/riaccorpamento di entità/relazioni e scelta degli identificatori principali. Questa fase conduce alla stesura di uno schema E-R ristrutturato della realtà di interesse. Nel prossimo paragrafo ci occuperemo della traduzione di questo schema E-R in uno schema relazionale.

### 5.1 Analisi delle ridondanze

In questa fase vanno individuate eventuali ridondanze nello schema concettuale. Le ridondanze possono essere costituite da relazioni (in presenza di cicli di relazioni) oppure da attributi (chiamati anche dati derivati). Utilizzando le informazioni sul carico, dobbiamo poi decidere se mantenere (introdurre, in alcuni casi) oppure eliminare queste ridondanze.

La relazione DI TIPO, che collega PRESTAZIONE con TIPO PRESTAZIONE, rappresenta la medesima informazione delle relazioni omonime che collegano VISITA e ANALISI con TIPO VISITA e TIPO ANALISI, rispettivamente. Osserviamo che non è importante decidere ora quale delle relazioni eliminare, in quanto tale decisione potrà essere meglio affrontata dopo aver deciso come tradurre le gerarchie coinvolte.

Osserviamo che gli attributi Prezzo in ESITO e COSTO sono tra loro indipendenti (ovvero, non costituiscono ridondanza). Infatti, un paziente potrebbe cambiare tipologia dopo aver avuto un esito. In tale caso (supponendo di credere una ridondanza Prezzo in ESITO ed eliminandola) si perderebbe il prezzo associato alla prestazione effettuata.

L'attributo Importo in FATTURA è ridondante, in quanto può essere calcolato come la somma dei prezzi degli esiti che compongono la fattura. Allo stesso modo, la relazione INTESTATA, che collega FATTURA con PAZIENTE, è ridondante perché si può arrivare alla stessa informazione navigando attraverso ESITO (poi si può arrivare a PAZIENTE attraverso una relazione, in quanto ESITO è identificata esternamente anche attraverso PAZIENTE).

Consideriamo la ridondanza costituita dalla relazione INTESTATA; supponiamo di mantenere tale ridondanza. Come addizionale occupazione di spazio abbiamo bisogno di rappresentare, per ogni istanza di FATTURA, 16 byte per il codice fiscale. Quindi utilizziamo  $16 \times 150.000 = 2.400.000 = 2.4M$ byte di memoria extra.

Le operazioni interessate alla ridondanza sono **O8** ed **O10**; tra queste, solo l'operazione **O8** deve essere eseguita in linea, quindi terremo principalmente conto del suo costo di esecuzione.

L'operazione **O8** deve generare e poi stampare le fatture da emettere; consideriamo la sua esecuzione quindi suddivisa in una fase di generazione ed una di stampa. Per determinare gli accessi effettuati dall'esecuzione di una operazione, è utile avere il suo schema di operazione, ovvero la porzione dello schema concettuale utilizzata dall'operazione. Lo schema di operazione di **O8** è riportato in Figura 10.

Nella fase di generazione, deve essere acceduto un PAZIENTE, poi vanno determinate le istanze di ESITO relativamente alle voci da registrare in una fattura (dalla tavola dei volumi mostrata in

---

<sup>1</sup>Per un esempio di tavola degli accessi, si può fare riferimento al Capitolo 7 del libro.

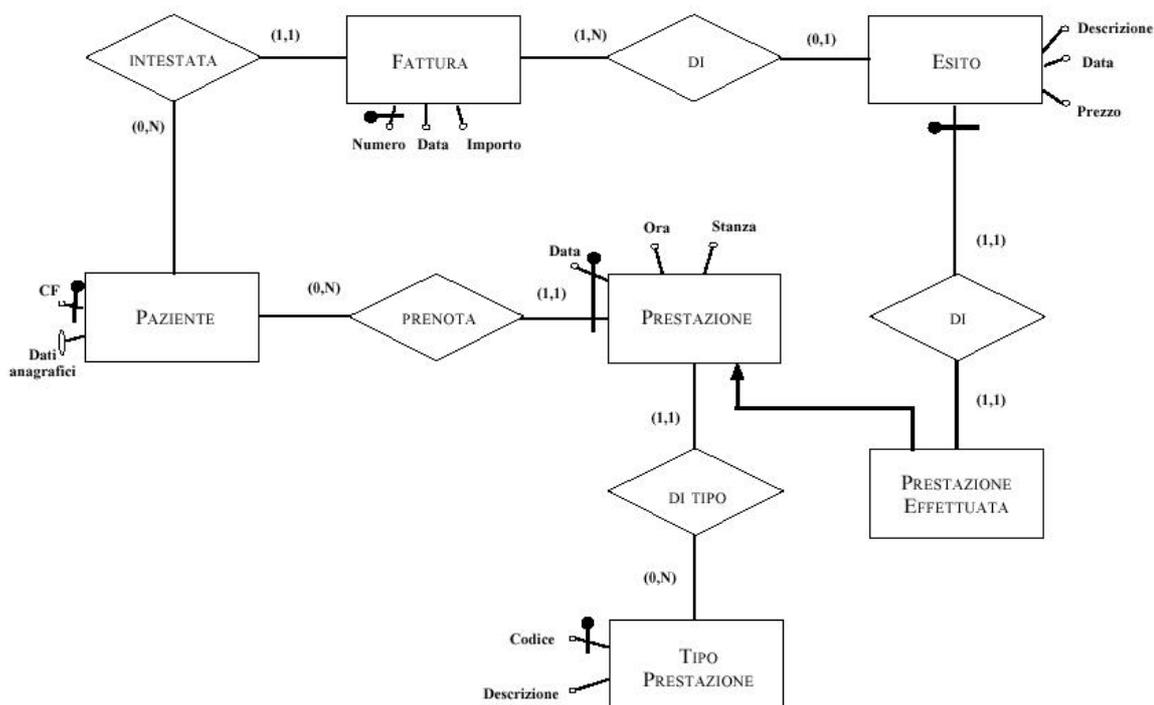


Figura 10: Schema di operazione di **O8**

Figura 8 è possibile dedurre che ogni fattura è composta mediamente da 2 esiti) quindi dobbiamo contare doppi tali accessi; infine va registrata la corrispondente istanza di FATTURA, 2 istanze di DI ed una istanza di INTEGERSATA. Questo per 300 volte al giorno. Nella fase di stampa, accediamo ad una istanza di FATTURA, quindi ad una istanza di PAZIENTE tramite la relazione INTEGERSATA (questo, ovviamente, qualora anche la ridondanza sull'attributo Importo di FATTURA fosse mantenuta).

Supponiamo ora di non mantenere la ridondanza. Nella fase di generazione, gli accessi in lettura ai dati sono gli stessi; possiamo però risparmiare un accesso in scrittura relativo alla relazione INTEGERSATA. Nella fase di stampa, l'accesso a PAZIENTE non avviene più tramite INTEGERSATA ma deve avvenire tramite DI ed ESITO (osserviamo che, da una istanza di ESITO, che contiene informazioni sul cofice fiscale del paziente che è intestatario della fattura, possiamo poi accedere a PAZIENTE). Sostanzialmente, per ogni esecuzione di **O8**, abbiamo bisogno di un accesso di scrittura se la ridondanza è presente e di 2 accessi in lettura se essa è assente.

Sembrerebbe che, dal punto di vista del numero di accessi, dovremmo decidere di mantenere la ridondanza. In realtà, tipicamente gli accessi in scrittura hanno costo maggiore di quelli in lettura; assumendo doppio il costo di un accesso in scrittura rispetto ad uno in lettura, dal punto di vista del numero di accessi la situazione è di parità. Decidiamo comunque di eliminare la ridondanza, visto l'elevata occupazione di memoria necessaria per mantenerla.

In maniera analoga, decidiamo di eliminare anche la ridondanza di Importo in FATTURA.

## 5.2 Eliminazione delle gerarchie

Nello schema sono presenti 4 gerarchie. In particolare, 2 di essere fanno capo alla entità PRESTAZIONE.

Per quanto riguarda le *prestazioni*, le operazioni non fanno grosse distinzioni né tra *visite* ed

*analisi*, né tra quelle *effettuate* o quelle solo *prenotate*. Questo ci porta a concludere che questa gerarchia può essere ristrutturata fondendo tutte le entità nella loro entità padre PRESTAZIONE, alla quale vanno però aggiunti due attributi di tipo booleano, Tipo Prestazione (che assumerà valore V oppure A, per *visita* oppure *analisi*) ed Effettuata (S oppure N, per *si* oppure *no*). Bisogna prestare attenzione alla cardinalità minima della relazione DI che collega PRESTAZIONE con ESITO.

Un discorso simile vale anche per la gerarchie che descrive i *tipi delle prestazioni*, dove le entità possono essere fuse nella entità padre TIPO PRESTAZIONE, introducendo un attributo Tipo (con valore V oppure A). (In seguito a questa scelta, decidiamo di non introdurre l'attributo Tipo Prestazione in PRESTAZIONE, in quanto derivabile da Tipo nell'istanza di TIPO PRESTAZIONE associata.) Bisogna prestare attenzione alla cardinalità minima delle relazioni ABILITATO.

L'ultima gerarchia da analizzare è quella relativa a PERSONALE LABORATORIO. La gerarchia è di tipo esclusivo e totale. Per entrambe le entità figlie sono presenti diversi attributi che li distinguono (Livello e Stipendio per ASSISTENTE, Specializzazione, Ente, Percentuale e Disponibilità per MEDICO). Inoltre, tutte le relazioni di interesse per la gerarchia incidono sui figli, e nessuna sul padre. Decidiamo quindi di mantenere le sole entità figlie, trasferendo in esse gli attributi della entità PERSONALE LABORATORIO.

### 5.3 Partizionamento/accorpamento entità/relazioni

Osserviamo che le entità PAZIENTE e CARTELLA CLINICA sono legate da una relazione di tipo *uno a uno*; inoltre, CARTELLA CLINICA è identificata solo attraverso la partecipazione alla relazione stessa. Visto l'esigua dimensione degli attributi in CARTELLA CLINICA, decidiamo di accorpare le due entità in PAZIENTE (sostanzialmente, è necessario introdurre l'attributo Data Apertura Cartella Clinica in PAZIENTE). Inoltre modifichiamo il nome della relazione COMPOSTA cambiandolo in CARTELLA.

### 5.4 Scelta degli identificatori primari

Le entità che hanno più di un identificatore sono MEDICO ed ASSISTENTE (identificate entrambe da Codice Fiscale oppure da un CODICE interno). Utilizziamo come identificatore di tali entità il codice interno, sia perché esso deriva da un concetto proprio del Laboratorio, che è quindi probabilmente significativo, sia per motivi di spazio (la dimensione dell'attributo Codice, valutabile in 4 byte, è minore di quella di Codice Fiscale, 16 byte).

In Figura 11 è riportato lo schema E-R ottenuto ristrutturando lo schema concettuale in base alle considerazioni fatte in questo paragrafo.

## 6 Progettazione logica nel modello relazionale

### 6.1 Traduzione entità

La prima fase di questa attività individua un insieme di nomi di R-relazione (parliamo di R-relazioni per indicare relazioni del modello relazionale, da non confondere con le relazioni del modello Entità-Relazione) ottenuti dalla traduzione delle entità presenti nello schema E-R ristrutturato. Questo passo individua insiemi di attributi che costituiscono lo schema "iniziale" di tali R-relazioni (vedremo che questi insiemi possono essere modificati nelle fasi successive). Prima di procedere, osserviamo che le specifiche richiedono informazioni circa l'età di pazienti e del personale del laboratorio. Decidiamo di rappresentare tale informazione indirettamente con un attributo Anno Nascita anziché con Età, essendo evidentemente questa una scelta più stabile.

Individuiamo i seguenti schemi di R-relazione:

- PAZIENTE (Codice Fiscale, Cognome, Nome, Anno Nascita, Indirizzo, Telefono, DataCC)
- TIPO PAZIENTE (Codice, Descrizione)

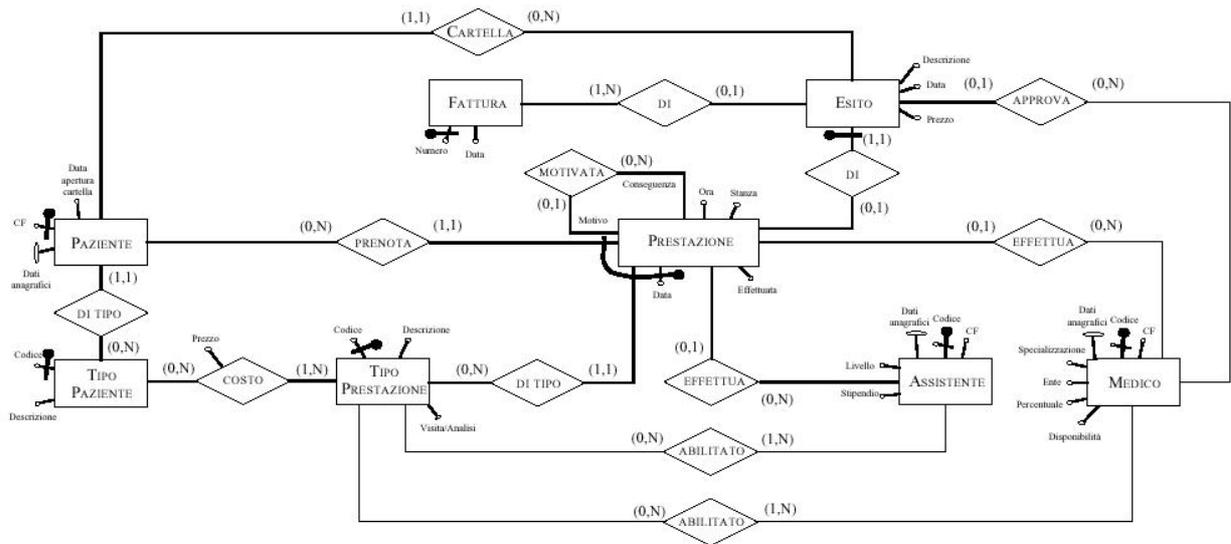


Figura 11: Schema E-R ristrutturato

- TIPO PRESTAZIONE (Codice, Descrizione, Visita/Analisi)
- FATTURA (Numero, Data)
- MEDICO (Codice, Codice Fiscale, Cognome, Nome, Anno Nascita, Indirizzo, Telefono, Specializzazione, Ente, Percentuale, Disponibilità)
- ASSISTENTE (Codice, Codice Fiscale, Cognome, Nome, Anno Nascita, Indirizzo, Telefono, Livello, Stipendio)
- PRESTAZIONE (CFPaziente, CTipo Prestazione, Data, Ora, Stanza, Effettuata)
- ESITO (CFPaziente, CTipoPrestazione, Data Esito, Descrizione, Prezzo)

La traduzione delle entità identificate esternamente (ovvero PRESTAZIONE ed ESITO) comporta anche la traduzione delle relazioni attraverso le quali tali entità sono identificate. Nella successiva fase tali relazioni non devono quindi essere ulteriormente prese in considerazione.

## 6.2 Traduzione di relazioni

Consideriamo dapprima le relazioni di tipo *uno a uno*, poi quelle di tipo *uno a molti*, infine quelle di tipo *molti a molti*.

Osserviamo che l'unica relazione di tipo *uno a uno* presente nello schema ristrutturato è quella collegante ESITO con PRESTAZIONE. Non è però necessario considerare ulteriormente questa relazione, in quanto è stata già tradotta insieme all'entità ESITO (per la sua identificazione esterna).

Molte sono invece le relazioni di tipo *uno a molti*. Consideriamole una per una. La relazione DI TIPO che collega PAZIENTE con TIPO PAZIENTE è a partecipazione obbligatoria dal lato PAZIENTE. Quindi possiamo tradurre questa relazione introducendo un attributo DI TIPO in PAZIENTE. Similmente:

- Per tradurre la relazione APPROVA, introduciamo un attributo Approva nella relazione ESITO.

- Per tradurre la relazione CARTELLA CLINICA, introduciamo un attributo chiamato CFPaziente nella relazione ESITO.

Consideriamo ora le relazioni di tipo *uno a molti* con partecipazione non obbligatoria. Per quanto riguarda le due relazioni EFFETTUA da ASSISTENTE e MEDICO verso PRESTAZIONE, decidiamo di introdurre due attributi di nome CAssistente e CMedico in PRESTAZIONE. Similmente, per la relazione DI che collega ESITO con FATTURA, introduciamo un attributo NFattura in ESITO. Ovviamente tale scelta implica la possibilità di avere valori nulli nei corrispondenti attributi di istanze legali.

L'ultima relazione di tipo *uno a molti* da considerare è MOTIVATA. Potremmo anche in questo caso introdurre un insieme di attributi per MOTIVATA nella R-relazione PRESTAZIONE, indicando da quale altra prestazione una prestazione è motivata. Evitiamo tale scelta per due motivi: il primo è che solo una piccola parte delle prestazioni è motivata da un'altra prestazione. Inoltre, dovremmo reintrodurre l'insieme di attributi che identifica l'entità PRESTAZIONE, che ha una dimensione non piccola. La scelta che facciamo è dunque quella di tradurre tale relazione con la R-relazione

- MOTIVATA (CFPazienteC, CTipo PrestazioneC, DataC, CFPazienteM, CTipo PrestazioneM, DataM)

Il suffisso C oppure M indica il ruolo Conseguenza o Motivo, rispettivamente.

Infine, consideriamo le relazioni di tipo *molti a molti*, che vanno tradotte con R-relazioni. Otteniamo i seguenti schemi di R-relazione:

- COSTO (CTPaziente, CTPrestazione, Prezzo)
- ABILITA MEDICO (CMedico, CTPrestazione)
- ABILITA ASSISTENTE (CAssistente, CTPrestazione)

In Figura 12 viene riportato lo schema logico completo per il laboratorio di diagnosi medica.

## 7 Implementazione di alcune operazioni

In questo paragrafo viene proposta l'implementazione di alcune delle operazioni che sono definite per il laboratorio di diagnosi medica.

### 7.1 Script di installazione

La seguente porzione di codice è una sequenza di comandi SQL che possiamo utilizzare per definire lo schema associato alla nostra applicazione medica. Il nome associato a tale schema è LDM, come dichiarato all'inizio dello script.

Vengono poi create le varie tabelle. Ad ogni tabella è associato almeno un indice, utilizzato per rafforzare il relativo vincolo di chiave; un eventuale secondo indice associato ad una tabella viene utilizzato per rendere più efficiente l'accesso ai dati (ad esempio, abbiamo un indice per PAZIENTE sugli attributi Cognome e Nome).

```
##### # File: INSTALLA.SQL
# Script di installazione
#####

create database LDM;

create table Paziente
(
```

```

    PAZIENTE (Codice Fiscale, Cognome, Nome, Anno Nascita, Indirizzo,
              Telefono, DataCC, Di Tipo)
    TIPO PAZIENTE (Codice, Descrizione)
    TIPO PRESTAZIONE (Codice, Descrizione, Tipo)
    FATTURA (Numero, Data)
MEDICO (Codice, Codice Fiscale, Cognome, Nome, Anno Nascita, Indirizzo,
        Telefono, Specializzazione, Ente, Percentuale, Disponibilità)
ASSISTENTE (Codice, Codice Fiscale, Cognome, Nome, Anno Nascita,
            Indirizzo, Telefono, Livello, Stipendio)
PRESTAZIONE (CFPaziente, CTipo Prestazione, Data, Ora, Stanza,
            Effettuata, Tipo, CMedico, CAssistente)
ESITO (CFPaziente, CTipo Prestazione, Data, Data Esito,
        Descrizione, Prezzo, Approva, NFattura)
MOTIVATA (CFPazienteC, CTipo PrestazioneC, DataC,
          CFPazienteM, CTipo PrestazioneM, DataM)
COSTO (CTPaziente, CTPrestazione, Prezzo)
ABILITA MEDICO (CMedico, CTPrestazione)
ABILITA ASSISTENTE (CAssistente, CTPrestazione)

```

Figura 12: Schema logico

```

cognome          char(40) NOT NULL,
nome             char(20) NOT NULL,
CF              char(16) NOT NULL,
annonascita     smallint,
indirizzo       char(40),
telefono        char(13),
dataCC          date,
tipo            char(4)
);

create unique index Paz_ix on Paziente (CF); create index PazCN_ix on Paziente
(cognome,nome);

create table TipoPaziente
(
  codice          char(4) NOT NULL,
  descrizione     char(40)
);

create unique index TPaz_ix on TipoPaziente (codice);

create table TipoPrestazione
(
  codice          char(4) NOT NULL,
  descrizione     char(40),
  AV             char(1)
);

create unique index TPres_ix on TipoPrestazione (codice);

create table Fattura
(
  numero         integer NOT NULL,
  data           date

```

```

);

create unique index Fat_ix on Fattura (numero);

create table Medico
(
    codice                char(8) NOT NULL,
    CF                    char(16) NOT NULL,
    cognome               char(30) NOT NULL,
    nome                  char(20) NOT NULL,
    annonascita           smallint,
    indirizzo              char(40),
    telefono              char(13),
    specializzazione      char(40),
    ente                   char(40),
    percentuale            decimal(2,0),
    disponibilita         char(7)
);

create unique index Med_ix on Medico (codice);

create table Assistente
(
    codice                char(8) NOT NULL,
    CF                    char(16) NOT NULL,
    cognome               char(30) NOT NULL,
    nome                  char(20) NOT NULL,
    annonascita           smallint,
    indirizzo              char(40),
    telefono              char(13),
    livello               char(4),
    stipendio              integer
);

create unique index Ass_ix on Assistente (codice);

create table Prestazione
(
    paziente              char(16) NOT NULL,
    tipoPrestazione      char(4) NOT NULL,
    data                  date NOT NULL,
    ora                   char(5),
    stanza                char(4),
    effettuata            char(1),
    medico                char(8),
    assistente            char(8)
);

create unique index Pres_ix on Prestazione (paziente, tipoPrestazione, data);

create table Esito
(
    paziente              char(16) NOT NULL,
    tipoPrestazione      char(4) NOT NULL,
    data                  date NOT NULL,
    dataesito             date,
    descrizione1          char(60),
    descrizione2          char(60),
    descrizione3          char(60),
    prezzo                integer,
    approva               char(8),
    fattura               integer
);

```

```

);

create unique index Esa_ix on Esito (paziente, tipoPrestazione, data);

create table Motivata
(
    pazienteC          char(16) NOT NULL,
    tipoPrestazioneC  char(4) NOT NULL,
    dataC              date NOT NULL,
    pazienteM          char(16),
    tipoPrestazioneM  char(4),
    dataM              date
);

create unique index Mot_ix on Motivata (pazienteC, tipoPrestazioneC, dataC);

create table Costo
(
    tipoPaziente      char(4) NOT NULL,
    tipoPrestazione   char(4) NOT NULL,
    prezzo            integer
);

create unique index Cos_ix on Costo (tipoPaziente, tipoPrestazione);

create table AbilitaMedico
(
    medico            char(8) NOT NULL,
    tipoPrestazione   char(4) NOT NULL
);

create unique index AM_ix on AbilitaMedico (medico, tipoPrestazione);

create table AbilitaAssistente
(
    assistente        char(8) NOT NULL,
    tipoPrestazione   char(4) NOT NULL
);

create unique index AA_ix on AbilitaAssistente (assistente, tipoPrestazione);

update statistics;

grant resource to public;

```

## 7.2 Operazione O1

```
INSERT INTO ...;
```

## 7.3 Operazione O2

...

## 8 Dati di Test

In questa sezione devono essere mostrate le schermate di alcune tabelle popolate con dati adeguati a testare la correttezza delle operazioni implementate.