

Progetto Basi di Dati

Eros Fabrici, Lorenzo Pittia, Giovanni Dreas

August 27, 2019

Indice

1	Analisi dei requisiti	2
1.1	Testo esercizio	2
1.2	Analisi requisiti	2
2	Modello E/R, vincoli di integrità e regole di derivazione	3
2.1	Vincoli d'integrità	3
2.2	Regole di derivazione	3
3	Progettazione Logica	5
3.1	Analisi delle Ridondanze	5
3.1.1	Tabella dei Volumi	5
3.1.2	Tabella delle operazioni	5
3.1.3	Analisi	5
3.2	Eliminazione delle generalizzazioni	8
3.3	Eliminazione degli attributi multi-valore	8
3.4	Modello ER ristrutturato	8
3.5	Modello relazionale	9
3.5.1	Normalizzazione	9

1 Analisi dei requisiti

1.1 Testo esercizio

Si voglia modellare il seguente insieme di informazioni riguardanti i pazienti ricoverati in un dato ospedale italiano.

- Ogni reparto sia identificato univocamente dal nome, occupi uno e un solo piano di un determinato edificio e abbia un unico primario e uno o più altri medici. Si tenga traccia del numero di camere, del numero di letti disponibili e del numero di letti occupati di ogni reparto.
- Ogni medico, primari compresi, sia identificato univocamente dal codice fiscale e sia caratterizzato da un nome, un cognome, dall'anno di nascita e da una o più specializzazioni. Si assuma che ogni medico, primari compresi, possa lavorare per un solo reparto.
- Ogni camera di un dato reparto sia identificata da un numero (camera 5 del reparto Oculistica) e sia caratterizzata dal numero di letti disponibili e dal numero di letti occupati.
- Ogni letto di un dato reparto sia identificato da un numero (letto 21 del reparto Ginecologia) e sia caratterizzato dalla camera ove correntemente si trova. Si assuma che un letto possa essere nel tempo spostato da una camera ad un'altra di uno stesso reparto, ma non possa cambiare reparto.
- Ogni paziente sia identificato univocamente dal codice fiscale e sia caratterizzato dal nome, dal cognome, dall'anno di nascita e dal letto occupato. Si tenga presente che fra i pazienti vi possono essere anche medici dell'ospedale, primari compresi.

1.2 Analisi requisiti

Dall'analisi del testo è possibile capire che lo scopo principale del database è quello di conservare le informazioni dei pazienti di un ospedale italiano.

Dunque il database risulta essere indirizzato all'uso da parte del personale interno dell'ospedale, che si occupa di inserire i nuovi pazienti, aggiornare tutte le informazioni relative (che letto occupa, quanti letti disponibili, etc.), aggiungere le diagnosi effettuate dai medici, stampare le informazioni relative a i pazienti, visualizzare il numero di letti disponibili nei reparti, etc.

E' stato ritenuto necessario aggiungere (dato lo scopo del database), oltre a tutte le informazioni indicate nel testo, la possibilità di tenere traccia dei ricoveri passati di ogni paziente, quindi quando un paziente viene dimesso, esso non viene cancellato. Inoltre si è voluto tenere traccia delle diagnosi effettuate dai medici relative ai pazienti (ovviamente, siccome un medico può essere anche un paziente, non è possibile eseguire un auto-diagnosi).

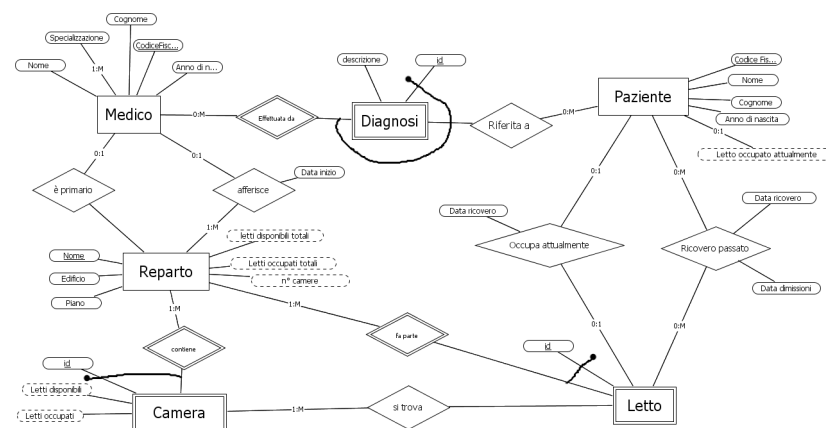
Dunque avremo le seguenti entità: *Medico*, *Paziente*, *Reparto*, *Camera* e *Letto*. Inoltre, dal testo è possibile estrarre che l'entità *Paziente* e l'entità *Medico* hanno gli stessi attributi ad eccezione per il secondo che ha un attributo in più. Ciò può portare a fare uso di una generalizzazione, ma non è stato ritenuto utile, perché le due entità in questione non hanno relazioni in comune con le altre e quindi non sarebbe di alcuna utilità.

E' stato deciso di modellare l'edificio e il piano come attributi di reparto, e non come un'entità separata, dato che in un edificio ci possono essere più reparti, così come in un piano.

2 Modello E/R, vincoli di integrità e regole di derivazione

Abbiamo usato un approccio inside-out, dato che è quello su cui abbiamo fatto più pratica a lezione. Siamo partiti dall'entità Medico, per poi espanderci. Ci siamo permessi di aggiungere un'entità, le relazioni che ne conseguivano e una relazione in più per tenere traccia dei ricoveri passati. Vi sono due assunzioni che abbiamo preso in considerazione, come accennato precedentemente nell'analisi:

1. Un paziente può non avere una diagnosi (solo pazienti appena ricoverati, pazienti passati hanno sicuramente almeno una diagnosi)
2. Un medico può partecipare a solo una delle relazioni "è primario" e "afferisce" (chiarimenti sulla scelta nella analisi delle ridondanze)



2.1 Vincoli d'integrità

1. Un Letto partecipa alla relazione "si trova" con una Camera la quale sta nella stesso Reparto del Letto (ergo: un letto facente parte di un certo Reparto può trovarsi soltanto in una Camera che fa parte del reparto in questione).
2. Un Paziente può essere anche Medico (e anche primario) e viceversa.
3. Un Paziente deve partecipare ad almeno una delle due seguenti relazioni: "occupa attualmente" e "ricovero passato".
4. Un medico non può effettuarsi l'autodiagnosi.
5. Un medico può effettuare la diagnosi solo a pazienti facenti parte del reparto in cui lavora.

2.2 Regole di derivazione

1. L'attributo derivato "n° camere" dell'entità Reparto viene derivato dalle partecipazioni con la relazione "contiene".
2. Gli attributi "letti disponibili" e "letti occupati" di Camera sono derivati dalla relazione "si trova".
3. Gli attributi "letti disponibili" e "letti occupati" dell'entità Reparto viene derivato dalle partecipazioni con la relazione "contiene".
4. L'attributo derivato "letto occupato attualmente" di Paziente viene derivato dalla parteci-

pazione con la relazione "occupa attualmente"

3 Progettazione Logica

In questa sezione si procede con vari raffinamenti al modello ER di modo da poterne trarre un modello logico equivalente.

3.1 Analisi delle Ridondanze

3.1.1 Tabella dei Volumi

Concetto	Tipo	Volume
Medico	E	500
Reparto	E	15
Camera	E	250
Letto	E	1000
Paziente	E	20000
Diagnosi	E	19500
E' primario	R	15
Afferisce	R	485
Contiene	R	250
Effettuata da	R	19500
Fa parte	R	1000
Si trova	R	1000
Occupa attualmente	R	700
Ricovero passato	R	19600
Riferita a	R	19500

Table 1: Tabella dei volumi

3.1.2 Tabella delle operazioni

Operazioni	Frequenza
Inserimento nuovo paziente	100 al giorno
Inserimento nuovo medico	2 al mese
Inserimento nuova diagnosi	10 al giorno
Modifica stato paziente (da occupa attualmente a ricovero passato)	70 al giorno
Stampa informazioni di un paziente	100 al giorno
Stampa letti disponibili di un reparto	100 al giorno
Stampa letti disponibili di una camera	50 al giorno
Stampa informazioni camera	50 al giorno

Table 2: Tabella delle operazioni

3.1.3 Analisi

A seguire alcune delle operazioni che coinvolgono tutti gli attributi ridondanti:

Operazione 1: stampa letti disponibili di un reparto.

Operazione 2: stampa informazioni paziente (compreso id del letto occupato).
Operazione 3: stampa informazioni camera (compreso n°letti occupati e disponibili).
Operazione 4: inserimento di un nuovo paziente.
Operazione 5: stampa il numero di di camere di un dato reparto.

CON RIDONDANZA

Concetto	Tipo	N°accessi	Tipo accesso
Reparto	E	1	R

Table 3: Tabella accessi Operazione 1

Concetto	Tipo	N°accessi	Tipo accesso
Paziente	E	1	R

Table 4: Tabella accessi Operazione 2

Concetto	Tipo	N°accessi	Tipo accesso
Camera	E	1	R

Table 5: Tabella accessi Operazione 3

Concetto	Tipo	N°accessi	Tipo accesso
Paziente	E	1	W
Occupa attualmente	R	1	R
Occupa attualmente	R	1	W
Paziente	E	1	W
Letto	E	1	R
Fa parte	R	1	R
Reparto	E	1	W
Si trova	R	1	R
Camera	E	1	W

Table 6: Tabella accessi Operazione 4

Concetto	Tipo	N°accessi	Tipo accesso
Reparto	E	1	R

Table 7: Tabella accessi Operazione 5

SENZA RIDONDANZA

Concetto	Tipo	N°accessi	Tipo accesso
Reparto	E	1	R
Fa parte	R	66	R
Letto	E	66	R
Occupa attualmente	R	66	R

Table 8: Tabella accessi Operazione 1

Concetto	Tipo	N°accessi	Tipo accesso
Paziente	E	1	R
Occupa attualmente	R	1	R
Letto	E	1	R

Table 9: Tabella accessi Operazione 2

Concetto	Tipo	N°accessi	Tipo accesso
Camera	E	1	R
Si trova	R	4	R
Letto	E	4	R
Occupa attualmente	R	4	R

Table 10: Tabella accessi Operazione 3

Concetto	Tipo	N°accessi	Tipo accesso
Paziente	E	1	W
Occupa attualmente	R	1	R
Occupa attualmente	R	1	W

Table 11: Tabella accessi Operazione 4

Concetto	Tipo	N°accessi	Tipo accesso
Reparto	E	1	R
Contiene	R	17	R

Table 12: Tabella accessi Operazione 5

E' possibile quindi osservare che la ridondanza riferita all'**operazione 1** sia necessaria per ridurre gli accessi. Per quanto riguarda l'**operazione 3**, in assenza di ridondanza non ha un costo considerevolmente maggiore rispetto al caso in cui vi sono presenti gli attributi ridondanti (una camera non avrà mai un numero potenzialmente grande di letti, al massimo una decina), dunque è stata decisa la loro rimozione.

Relativamente l'attributo "letto occupato attualmente" di Paziente, riteniamo che non sia opportuno tenerlo, dato che un paziente occuperà al più 1 solo letto. L'**operazione 4** risulta leggermente più

efficiente se assenti gli attributi ridondanti, ma ciò porterebbe a un maggiore costo, come detto in precedenza, nell'esecuzione dell'**operazione 1**.

In base ai risultati dell'**operazione 5** è stata decisa la rimozione dell'attributo "n°camere" dato che da requisiti, il numero camere di un reparto rimane fisso.

Ricapitolando:

1. Rimuovere l'attributo "letto occupato attualmente" dall'entità *Paziente*.
2. Conservare gli attributi "letti disponibili totali" e "letti occupati totali" dell'entità *Reparto* e rimuoverne l'attributo "n°camere".
3. Rimuovere gli attributi "letti disponibili" e "letti occupati" dell'entità *Camera*

E' stato scelto inoltre il fatto che un medico può partecipare solo a una delle due relazioni tra "afferisce" e "è primario". Questo perché se un medico è primario di un certo reparto allora ne è anche afferente e quindi abbiamo constatato che non è necessario che esso compaia anche nella relazione "afferisce" dato che risulterebbe come un'informazione ridondante che non vi è di grande utilità da un punto di vista degli accessi (se volessimo stampare tutti gli afferenti di un reparto, accediamo ad entrambe le relazioni effettuando una lettura ciascuna, con la ridondanza si accederebbe solo alla relazione "afferisce").

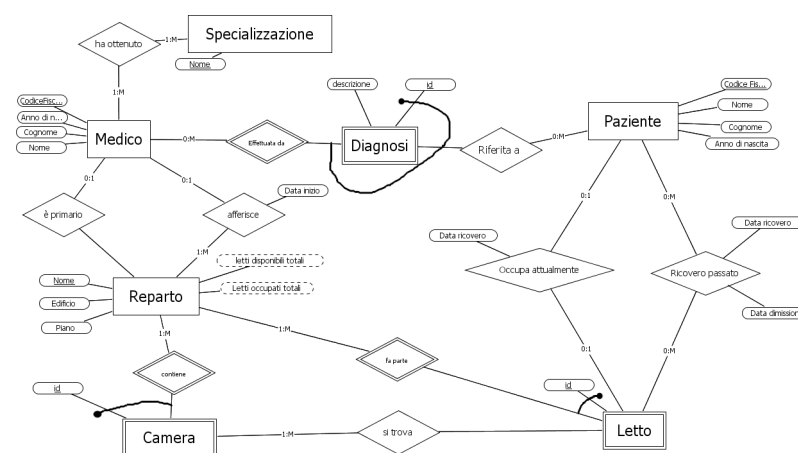
3.2 Eliminazione delle generalizzazioni

Non vi è presente nessuna generalizzazione, dunque si è passati direttamente al punto successivo.

3.3 Eliminazione degli attributi multi-valore

L'entità *Medico* ha l'attributo "specializzazione" che ha cardinalità 1:n. E' dunque necessario eliminare quell'attributo per sostituirlo con un'entità omonima e una relazione ("ha conseguito") che la lega con *Medico*. La nuova entità avrà gli attributi "Nome", che la identifica univocamente, e "Descrizione".

3.4 Modello ER ristrutturato



3.5 Modello relazionale

medico(cod_fiscale, nome, cognome, anno_di_nascita)
reparto(nome, edificio, piano, letti_disponibili, letti_occupati, primario, data_inizio)
camera(id, reparto)
letto(id, reparto)
paziente(cod_fiscale, nome, cognome, anno_di_nascita)
specializzazione(nome)
ha_ottenuto(medico, specializzazione)
afferisce(medico, reparto)
si_trova(id_letto, reparto, id_camera)
occupa_attualmente(paziente, id_letto, reparto, data_ricovero)
ricovero_passato(paziente, data_ricovero, data_dimissioni, letto, reparto)
diagnosi(id, medico, paziente, descrizione)

Chiavi esterne:

- *reparto.primario* → *medico.cod_fiscale*
- *camera.reparto* → *reparto.nome*
- *letto.reparto* → *reparto.nome*
- *ha_ottenuto.medico* → *medico.cod_fiscale*
- *ha_ottenuto.specializzazione* → *specializzazione.nome*
- *afferisce.medico* → *medico.cod_fiscale*
- *afferisce.reparto* → *reparto.nome*
- (*si_trova.id_letto*, *si_trova.id_reparto*) → (*letto.id*, *letto.reparto*)
- (*si_trova.id_camera*, *si_trova.reparto*) → (*camera.id*, *camera.reparto*)
- *occupa_attualmente.paziente* → *paziente.cod_fiscale*
- (*occupa_attualmente.id_letto*, *occupa_attualmente.reparto*) → (*letto.id*, *letto.reparto*)
- *ricovero_passato.paziente* → *paziente.cod_fiscale*
- (*ricovero_passato.id_letto*, *ricovero_passato.reparto*) → (*letto.id*, *letto.reparto*)
- *diagnosi.medico* → *medico.cod_fiscale*
- *diagnosi.paziente* → *paziente.cod_fiscale*

3.5.1 Normalizzazione

Il modello rispetta tutte e tre le forme di normalizzazione.

- **1° Forma:** non esistono attributi ripetuti o composti;
- **2° Forma:** tutti gli attributi dipendono dalla chiave primaria completa;
- **3° Forma:** gli attributi dipendono solo dalla chiave primaria;