Basi di Dati
La teoria della normalizzazione

Contenuti - Roadmap

Lab (progettazione)	Corso di Teoria	Lab (SQL)
• Metodologie e modello Entità Associazioni • Progettazione concettuale e logica	• Modello Relazionale • Algebra relazionale • Ottimizzazione logica • Calcolo relazionale • **La normalizzazione** • Metodi di accesso e indici • Gestione della concorrenza • Gestione del ripristino	• Linguaggio SQL

Outline

- **Introduzione**
- Anomalie di inserimento, cancellazione e aggiornamento
- Dipendenze funzionali
- Teoria di Armstrong
- Chiusura di un insieme di dipendenze funzionali
- Chiusura di un insieme di attributi
- Dipendenze funzionali e superchiavi

Forme normali

- Una forma normale è una proprietà di una base di dati relazionale che ne garantisce la "qualità", cioè l'assenza di determinati difetti
- Quando una relazione *non* è in forma normale:
  - presenta ridondanze,
  - causa anomalie quando si aggiornano, cancellano e inseriscono dati

Forme normali

- Le forme normali sono di solito definite sul modello relazionale
- Discuteremo anche il rapporto con il modello E-R
- La metodologia di progettazione discussa di solito permette di ottenere schemi già in forma normale

Normalizzazione

- Procedura che permette di trasformare schemi non normali in schemi che soddisfano una forma normale
- La normalizzazione può essere usata come tecnica di verifica dei risultati della progettazione di una base di dati
- Non costituisce una metodologia di progettazione

Outline

- Introduzione
- **Anomalie di inserimento, cancellazione e aggiornamento**
- Dipendenze funzionali
- Teoria di Armstrong
- Chiusura di un insieme di dipendenze funzionali
- Chiusura di un insieme di attributi
- Dipendenze funzionali e superchiavi

Esempio

Immaginiamo una relazione ESAMI

ESAMI(<u>MATR</u>, NomeS, IndirizzoS, CAPS, CodiceFiscaleS, DataNascitaS, <u>Corso</u>, Voto, Lode, DataEsame, CodProf, NomeProf, Qualifica, TipoUfficio)

Anche se poco usuale, dal punto di vista del modello relazionale è una relazione valida

Useremo delle abbreviazioni

ESAMI(<u>MATR</u>, NS, IS, CAP, CF, DN, <u>Co</u>, Vo, Lo, DE, CP, NP, Q, TU)

Criticità delle relazioni

Istanziamo la relazione con dei dati ragionevoli

<u>MATR</u>	NS	IS	CAP	CF	DN	<u>Co</u>	Vo	Lo	DE	CP	NP	Q	TU
341	Piero	TO	101	PX	1990	BD	27	F	1/4/15	P1	Anselma	Ric	ExLab
343	Giorgio	NO	102	GY	1991	Prog1	30	T	2/5/15	B1	Cardone	ProfA	Piano1
341	Piero	TO	101	PX	1990	Prog1	25	F	3/5/15	B1	Cardone	ProfA	Piano1
343	Giorgio	NO	102	GY	1991	BD	18	F	6/4/15	P1	Anselma	Ric	ExLab
…	…	…	…	…	…	…	…		…	…	…	…	…

Supponiamo che ESAMI sia l'unica relazione che descrive il sistema informativo; essa contiene infatti tutti i dati che ci serve gestire: Anagrafica studenti, Anagrafica docenti, Esami superati

Cosa succede se:

i. *Inseriamo* un nuovo studente o un nuovo professore?

ii. *Cancelliamo* degli esami?

iii. *Aggiorniamo* l'indirizzo di uno studente o l'ufficio di un professore?

Anomalie di inserimento (i)

1. Un <u>nuovo studente</u> intende immatricolarsi
2. Lo studente, ovviamente, non ha ancora superato nessun esame
3. Ho bisogno di inserire una nuova tupla

<u>**MATR**</u>	**NS**	**IS**	**CAP**	**CF**	**DN**	<u>**Co**</u>	**Vo**	**Lo**	**DE**	**CP**	**NP**	**Q**	**TU**
341	Piero	TO	101	PX	1990	BD	27	F	1/4/15	P1	Anselma	Ric	ExLab
343	Giorgio	NO	102	GY	1991	Prog1	30	T	2/5/15	B1	Cardone	ProfA	Piano1
341	Piero	TO	101	PX	1990	Prog1	25	F	3/5/15	B1	Cardone	ProfA	Piano1
343	Giorgio	NO	102	GY	1991	BD	18	F	6/4/15	P1	Anselma	Ric	ExLab
444	**Laura**	**TO**	**101**	**LZ**	**1992**	**NULL**	**NULL**	**N**	**NULL**	**N**	**NULL**	**NULL**	**NULL**

nuovo studente

- Una parte della tupla non sarà valorizzata
- Ma Corso è parte della chiave primaria e non può avere valori nulli

Anomalie di inserimento (i)

1. Viene assunto un <u>nuovo docente</u>
2. Il docente non ha ancora firmato alcun esame
3. Voglio inserirlo nel DB

<u>MATR</u>	NS	IS	CAP	CF	DN	<u>Co</u>	Vo	Lo	DE	CP	NP	Q	TU
341	Piero	TO	101	PX	1990	BD	27	F	1/4/15	P1	Anselma	Ric	ExLab
343	Giorgio	NO	102	GY	1991	Prog	21	T	2/5/15	B1	Cardone	ProfA	Piano1
341	Piero	TO	101	PX	1990	Prog	25	F	3/5/15	B1	Cardone	ProfA	Piano1
343	Giorgio	NO	102	GY	1991	BD	18	F	6/4/15	P1	Anselma	Ric	ExLab
NULL	**NULL**	**NULL**	**NULL**	**NULL**	**NULL**	**NULL**	**NULL**	**N**	**NULL**	**R1**	**Basile**	**Ric**	**Piano2**

- Come per lo studente, non posso inserirlo perché non posso valorizzare la chiave primaria

Anomalie di inserimento

Criticità di *esprimibilità*: l'anomalia consiste nell'incapacità di inserire un'informazione **concettualmente significativa** proveniente dal sistema informativo

MATR	NS	IS	CAP	CF	DN	Co	Vo	Lo	DE	CP	NP	Q	TU
341	Piero	TO	101	PX	1990	BD	27	F	1/4/15	P1	Anselma	Ric	ExLab
343	Giorgio	NO	102	GY	1991	Prog1	30	T	2/5/15	B1	Cardone	ProfA	Piano1
341	Piero	TO	101	PX	1990	Prog1	25	F	3/5/15	B1	Cardone	ProfA	Piano1
343	Giorgio	NO	102	GY	1991	BD	18	F	6/4/15	P1	Anselma	Ric	ExLab
444	**Laura**	**TO**	**101**	**LZ**	**1992**	**NULL**	**NULL**	**N**	**NULL**	**N**	**NULL**	**NULL**	**NULL**

Anomalia di cancellazione (ii)

- Voglio cancellare gli esami degli studenti laureati
- Se un professore non ha esami di studenti che devono ancora laurearsi, viene cancellato anche ogni riferimento al professore anche se è ancora in servizio

MATR	NS	IS	CAP	CF	DN	Co	Vo	Lo	DE	CP	NP	Q	TU
~~341~~	~~Piero~~	~~TO~~	~~101~~	~~PX~~	~~1990~~	~~BD~~	~~27~~	~~F~~	~~1/4/15~~	~~P1~~	~~Anselma~~	~~Ric~~	~~ExLab~~
~~343~~	~~Giorgio~~	~~NO~~	~~102~~	~~GY~~	~~1991~~	~~Prog1~~	~~30~~	~~T~~	~~2/5/15~~	~~B1~~	~~Cardone~~	~~ProfA~~	~~Piano1~~
~~341~~	~~Piero~~	~~TO~~	~~101~~	~~PX~~	~~1990~~	~~Prog1~~	~~25~~	~~F~~	~~3/5/15~~	~~B1~~	~~Cardone~~	~~ProfA~~	~~Piano1~~
~~343~~	~~Giorgio~~	~~NO~~	~~102~~	~~GY~~	~~1991~~	~~BD~~	~~18~~	~~F~~	~~6/4/15~~	~~P1~~	~~Anselma~~	~~Ric~~	~~ExLab~~
…	…	…	…	…	…	…	…		…	…	…	…	…

- Criticità di *esprimibilità*

Anomalia di aggiornamento (iii)

Anomalia di aggiornamento può portare a inconsistenza.

Immaginiamo di aggiornare l'indirizzo e il CAP dello studente 341, che ha registrato già molti esami.

Dal punto di vista concettuale, se voglio una base di dati consistente, la modifica deve interessare tutte le tuple in cui appare lo studente 341.

MATR	NS	IS	CAP	CF	DN	Co	Vo	Lo	DE	CP	NP	Q	TU
341	**Piero**	**TO**	**101**	**PX**	**1990**	**BD**	**27**	**F**	**1/4/15**	**P1**	**Anselma**	**Ric**	**ExLab**
343	Giorgio	NO	102	GY	1991	Prog1	30	T	2/5/15	B1	Cardone	ProfA	Piano1
341	**Piero**	**TO**	**101**	**PX**	**1990**	**Prog1**	**25**	**F**	**3/5/15**	**B1**	**Cardone**	**ProfA**	**Piano1**
343	Giorgio	NO	102	GY	1991	BD	18	F	6/4/15	P1	Anselma	Ric	ExLab
…	…	…	…	…	…	…	…		…	…	…	…	…

Anomalia di aggiornamento (iii)

Immaginiamo ora che il docente *P1* cambi ufficio

Se vogliamo una base di dati consistente, la modifica va apportata a tutte le tuple corrispondenti agli esami firmati da *P1*

Se anche una sola tupla non viene modificata (update incompleto) lo stato della base di dati diventa concettualmente inconsistente

Inoltre la modifica interessa un numero enorme di tuple (un docente può avere firmato decine di migliaia di esami)

Criticità di *efficienza*

Azioni necessarie

- A cosa sono dovute queste anomalie? È possibile eliminarle?
- Il problema nasce dal fatto che la relazione ESAMI rappresenta in realtà tre concetti diversi: gli studenti, i professori e gli esami
- Formalizzeremo questa idea con le *dipendenze funzionali*
- Usando le dipendenze funzionali definiremo dei processi di **normalizzazione** che permettono di minimizzare le anomalie trasformando la relazione

Outline

- Introduzione
- Anomalie di inserimento, cancellazione e aggiornamento
- **Dipendenze funzionali**
- Teoria di Armstrong
- Chiusura di un insieme di dipendenze funzionali
- Chiusura di un insieme di attributi
- Dipendenze funzionali e superchiavi

Dipendenza funzionale

Nel sistema informativo "segreteria studenti" alcuni ***attributi* caratterizzano dei *concetti***; ad esempio:

- Studenti caratterizzati da *MATR, NomeS, IndirizzoS…*
- Docenti caratterizzati da *CodProf, NomeProf, Qualifica…*

In particolare con la *Matricola* dello studente caratterizzo tutti gli attributi dello *studente.*

C'è cioè una **correlazione** tra la matricola dello studente MATR e la sua descrizione in termini di attributi NomeS, IndirizzoS, CAPS, CodiceFiscaleS, DataNascitaS.

Dipendenza funzionale

- In altre parole, immaginiamo due tuple t_1 e t_2 di ESAMI relative a due esami.
- **Se t_1 e t_2 coincidono su *MATR***, ci aspettiamo che **riguardino lo stesso studente** e, quindi, che t_1 e t_2 siano **uguali anche su tutti gli altri attributi** che riguardano lo studente: *NomeS, IndirizzoS, CAPS, CodiceFiscaleS, DataNascitaS*.
- Quindi il seguente ***vincolo*** deve essere soddisfatto: presa una coppia di tuple della relazione ESAMI, se queste tuple coincidono sul valore della matricola, anche tutti gli altri attributi relativi allo studente devono coincidere.

Dipendenza funzionale (definizione)

Dati una relazione *r(A)* e due sottoinsiemi *X* e *Y* di attributi di *A* ($X,Y \subseteq A$),
il **vincolo di dipendenza funzionale**

$$X \rightarrow Y$$

(X determina Y)

è soddisfatto se e solo se

$$\forall t_1,t_2 \in r(t_1[X]=t_2[X] \Rightarrow t_1[Y]=t_2[Y])$$

Attenzione: anche se graficamente usiamo una freccia, la dipendenza funzionale ($\rightarrow$*) è un concetto diverso da quello di implicazione logica (*$\Rightarrow$*)*

Esempi di vincoli

Conoscendo il dominio (cioè la realtà che vogliamo modellare), possiamo **ricavare le dipendenze funzionali** relative a ESAMI.

ESAMI(<u>MATR</u>, NomeS, IndirizzoS, CAPS, CodiceFiscaleS, DataNascitaS, <u>Corso</u>, Voto, Lode, DataEsame, CodProf, NomeProf, Qualifica, TipoUfficio)

ESAMI(<u>MATR</u>, NS, IS, CAP, CF, DN, <u>Co</u>, Vo, Lo, DE, CP, NP, Q, TU)

Dipendenze funzionali (almeno 6):

- MATR → NS, IS, CAP, CF, DN
- …

Esempi di vincoli

ESAMI(<u>MATR</u>, NS, IS, CAP, CF, DN, <u>Co</u>, Vo, Lo, DE, CP, NP, Q, TU)

Dipendenze funzionali:

- MATR → NS, IS, CAP, CF, DN
- CF → MATR
- IS → CAP (nell'ipotesi di indirizzo completo)
- MATR, Co → Vo, Lo, DE, CP, NP
- CP → NP, Q (prof)
- Q → TU (qualifica → titolo ufficiale)

Immaginiamo un vincolo per cui ogni docente è titolare di un solo corso

- CP → Co (dipendenza funzionale che useremo in seguito)

Ha senso la dipendenza funzionale Vo → Lo? E Lo → Vo?

Progettazione e dipendenze funzionali

- Le dipendenze funzionali vengono raccolte attraverso un'analisi attenta della realtà

- Per ogni relazione *r* abbiamo un **insieme *F*** di dipendenze funzionali

Insieme di dipendenze funzionali

Il nostro insieme F conterrà:

{ MATR → NS, IS, CAP, CF, DN;

CF → MATR;

IS → CAP;

MATR, Co → Vo, Lo, DE, CP, NP;

CP → NP, Q;

Q → TU;

CP → Co }

Formulazioni alternative

Ci possono essere diverse formulazioni alternative e equivalenti delle dipendenze funzionali

Un progettista vede:

- MATR → NS, IS, CAP, CF, DN
- CF → MATR
- IS → CAP

Un altro progettista vede:

- MATR → NS, IS
- MATR → CF, DN
- CF → MATR, NS, DN
- IS → CAP

Formulazioni alternative

I due progettisti hanno letto la realtà in modo **diverso**?

Sì, i due insiemi sono sintatticamente diversi

I due progettisti hanno letto la realtà in modo **equivalente**?

Lo studio del problema dell'equivalenza tra insiemi di dipendenze funzionali ci accompagnerà per questo blocco di slide e ci permetterà di sviluppare strumenti utili per la normalizzazione.

Equivalenza delle dipendenze funzionali

Primo insieme F':

f'_1: MATR → NS, IS, CAP, CF, DN

Secondo insieme F'':

f''_1: MATR → NS, IS, CAP

f''_2: MATR → CF, DN

$F'=\{f'_1\}$ e $F''=\{f''_1, f''_2\}$ sono equivalenti?

Primo approccio al problema dell'equivalenza:

Applichiamo la definizione di dipendenza funzionale e dimostriamo che vale $f''_1 \wedge f''_2$ se e solo se vale f'_1

Verifica dell'equivalenza $(f''_1 \wedge f''_2 \Rightarrow f'_1)$

Usiamo la definizione di dipendenza funzionale su $f''_1 \wedge f''_2$:

$\boldsymbol{f''_1}$: $\forall t_1,t_2 \in r(t_1[MATR]=t_2[MATR] \Rightarrow t_1[NS,IS,CAP]=t_2[NS,IS,CAP]) \wedge$

$\boldsymbol{f''_2}$: $\forall t_1,t_2 \in r(t_1[MATR]=t_2[MATR] \Rightarrow t_1[CF,DN]=t_2[CF,DN])$

cioè

$$\forall t_1,t_2 \in r(t_1[MATR]=t_2[MATR] \Rightarrow t_1[NS,IS,CAP]=t_2[NS,IS,CAP] \wedge$$
$$t_1[MATR]=t_2[MATR] \Rightarrow t_1[CF,DN]=t_2[CF,DN])$$

quindi

$$\forall t_1,t_2 \in r(\ t_1[MATR]=t_2[MATR] \Rightarrow$$
$$(t_1[NS,IS,CAP]=t_2[NS,IS,CAP] \wedge t_1[CF,DN]=t_2[CF,DN])\ )$$

che è equivalente a

$\boldsymbol{f'_1}$: $\forall t_1,t_2 \in r(t_1[MATR]=t_2[MATR] \Rightarrow$

$$t_1[NS,IS,CAP,CF,DN]=t_2[NS,IS,CAP,CF,DN])$$

che è la definizione di f'_1: MATR → NS, IS, CAP, CF, DN

Verifica dell'equivalenza $(f'_1 \Rightarrow f''_1 \wedge f''_2)$

Ma vale anche il viceversa

$$\boldsymbol{f'_1}: \forall t_1,t_2 \in r\ (t_1[MATR]=t_2[MATR] \Rightarrow t_1[NS,IS,CAP,CF,DN]=t_2[NS,IS,CAP, CF,DN])$$

quindi, a maggior ragione,

$$\boldsymbol{f''_1}: \forall t_1,t_2 \in r\ (t_1[MATR]=t_2[MATR] \Rightarrow t_1[NS,IS,CAP]=t_2[NS,IS,CAP])$$

che è la definizione di f''_1: MATR → NS, IS, CAP

Verifica dell’equivalenza $(f'_1 \Rightarrow f''_1 \wedge f''_2)$

In modo analogo da f’$_1$ posso ricavare f’’$_2$:

Da

$\boldsymbol{f'_1}$: $\forall t_1, t_2 \in r\ (t_1[MATR]=t_2[MATR] \Rightarrow$
$t_1[NS,IS,CAP,CF,DN]=t_2[NS,IS,CAP,CF,DN])$

ricavo

$\boldsymbol{f''_2}$: $\forall t_1, t_2 \in r\ (t_1[MATR]=t_2[MATR] \Rightarrow t_1[CF,DN]=t_2[CF,DN])$

che è la definizione di f''_2: MATR → CF, DN

Quindi il rispetto di f'_1 implica il rispetto di f''_1 e f''_2

Significato di equivalenza

Due basi di dati, una progettata con i vincoli F’ e un’altra progettata con i vincoli F’’, se F’ e F’’ sono equivalenti, evolvono esattamente nello stesso modo.

Outline

- Introduzione
- Anomalie di inserimento, cancellazione e aggiornamento
- Dipendenze funzionali
- **Teoria di Armstrong**
- Chiusura di un insieme di dipendenze funzionali
- Chiusura di un insieme di attributi
- Dipendenze funzionali e superchiavi

Teoria di Armstrong

- Utilizzare la sola definizione di dipendenza funzionale per verificare le equivalenze può risultare molto pesante
- Conviene quindi avere un modo più efficiente di procedere
- Armstrong ha definito una teoria con la quale caratterizza la dipendenza funzionale, ovvero l'oggetto "$\rightarrow$". In questo modo è possibile evitare di ricorrere alla definizione di dipendenza funzionale

«Assiomi» della Teoria di Armstrong

Dati X e Y insiemi di attributi:

- Assioma di **riflessività**

$$se\ Y \subseteq X,\ allora\ X \rightarrow Y$$

- Assioma di **unione**

$$se\ X \rightarrow Y\ e\ X \rightarrow Z,\ allora\ X \rightarrow YZ$$
$$dove\ YZ = Y \cup Z$$

- Assioma di **transitività**

$$se\ X \rightarrow Y\ e\ Y \rightarrow Z,\ allora\ X \rightarrow Z$$

- (In letteratura vengono chiamati «assiomi» ma sono più propriamente proprietà / regole inferenziali, dimostrabili partendo dalla definizione di dipendenza funzionale)

Assioma di riflessività

se $Y \subseteq X$, allora $X \rightarrow Y$

Se considero un insieme di attributi (ad es. X={CF,Co,Q}) e ne prendo un sottoinsieme (ad es. Y={Co,Q}), sicuramente c'è una dipendenza funzionale (ad es. {CF,Co,Q} → {Co,Q})

Esempio:

Primo insieme F':

- f'_1: MATR → NS, IS, CAP, CF, DN

Secondo insieme F'':

- f''_1: MATR → NS, IS, CAP, CF, DN
- f''_2: CF,Co,Q → Co,Q

Si può ricavare f''_1 e f''_2 partendo da f'_1 applicando l'assioma di riflessività e quindi stabilire che F' e F'' sono equivalenti.

Assioma di unione

se $X \rightarrow Y$ e $X \rightarrow Z$, allora $X \rightarrow YZ$

Esempio:

Primo insieme F':

f'_1: MATR → NS, IS, CAP, CF, DN

Secondo insieme F'':

f''_1: MATR → NS, IS, CAP

f''_2: MATR → CF, DN

Si può ricavare f'_1 partendo da f''_1 e f''_2 applicando l'assioma di unione e stabilire che F' e F'' sono equivalenti.

Assioma di transitività

se $X \rightarrow Y$ e $Y \rightarrow Z$, allora $X \rightarrow Z$

Esempio:

Insieme di vincoli F'

- f'_1: MATR → NS, IS, CF, DN
- f'_2: IS → CAP

Insieme di vincoli F''

- f''_1: MATR → NS, IS, CAP, CF, DN
- f''_2: IS → CAP

Applicando l'assioma di transitività si ricava f''_1 da f'_1 e f'_2, quindi F' e F'' sono equivalenti.

Completezza e correttezza della Teoria di Armstrong

La teoria di Armstrong è una teoria **corretta** e **completa**.

Correttezza della Teoria di Armstrong

Per ***correttezza*** si intende: dato un insieme di dipendenze funzionali *F*, se è possibile dedurre $X \rightarrow Y$ tramite gli assiomi di Armstrong, allora è possibile ricavare $X \rightarrow Y$ tramite la definizione di dipendenza funzionale.

- La dimostrazione si fa dimostrando la correttezza di ogni assioma rispetto alla definizione di dipendenza funzionale e poi per induzione mostrando che i tre assiomi generano dipendenze che soddisfano i vincoli.

Completezza della Teoria di Armstrong

Per ***completezza*** si intende: dato un insieme di dipendenze funzionali *F*, se è possibile ricavare $X \rightarrow Y$ tramite la definizione di dipendenza funzionale, allora è possibile dedurre $X \rightarrow Y$ tramite gli assiomi di Armstrong.

Completezza e correttezza della Teoria di Armstrong

La teoria di Armstrong è una teoria **corretta e completa**

quindi

ragionare con gli assiomi di Armstrong è perfettamente equivalente a ragionare con le definizioni di dipendenza funzionale

Regole aggiuntive

Dagli «assiomi» presentati è possibile ricavare delle regole aggiuntive che possono risultare utili:

- Regola dell’espansione
- Regola di decomposizione
- Regola pseudo-transitività
- Regola del prodotto

Queste regole non fanno parte degli assiomi perché è possibile dimostrarle usando gli assiomi (cioè senza usare la definizioni di dipendenza funzionale).

Regola dell'espansione

Dati una dipendenza funzionale $X \rightarrow Y$ e un insieme di attributi W, allora $WX \rightarrow WY$

Esempio: se considero la dipendenza funzionale $Q \rightarrow TU$ e l'attributo *CF*, posso ottenere $CF,Q \rightarrow CF,TU$

Esercizio: applicare la definizione di dipendenza funzionale per verificare la validità dell'esempio.

Dimostrazione

Ipotesi: data X → Y, dato W

Tesi: WX → WY

Dimostrazione (con la teoria di Armstrong)

1. Partiamo da *WX*
2. Scelgo un sottoinsieme: *W*
3. Per l'assioma di **riflessività**: *WX* → *W*
4. Per l'assioma di **riflessività**: *WX* → *X*
5. Per **ipotesi** sappiamo che *X* → *Y*
6. Allora per **transitività** da 4 e 5: *WX* → *Y*
7. Ma se *WX* → *W* e *WX* → *Y* (da 3 e 6), per l'assioma **dell'unione**: *WX* → *WY*

[CVD]

Regola di decomposizione

Se $X \rightarrow YZ$, allora $X \rightarrow Y$ e $X \rightarrow Z$

Esempio

Se MATR → CF, DN, allora MATR → CF e MATR → DN

N.B.: <u>Non</u> è vero che, se XY→Z, allora X→Z o Y→Z

Dimostrazione

Ipotesi: dato $X \rightarrow YZ$

Tesi: $X \rightarrow Y$ e $X \rightarrow Z$

Dimostrazione

1. Partiamo dall'ipotesi $X \rightarrow YZ$
2. Per l'assioma di **riflessività**: $YZ \rightarrow Y$
3. Per l'assioma di **transitività** su 1 e 2: $X \rightarrow Y$
4. Per l'assioma di **riflessività**: $YZ \rightarrow Z$
5. Per l'assioma di **transitività** su 1 e 4: $X \rightarrow Z$
6. In 3 e 5 abbiamo la tesi

[CVD]

Regola di pseudo-transitività

Se X → Y e WY → Z, allora WX → Z

Esempio

Se CF → MATR e Co, MATR → CP, allora Co, CF → CP

Regola di pseudo-transitività

Date $X \rightarrow Y$, $WY \rightarrow Z$, allora $WX \rightarrow Z$

Dimostrazione

1. Partiamo da $X \rightarrow Y$
2. Per il teorema di **espansione** $WX \rightarrow WY$
3. Per l'assioma di **transitività**, $WX \rightarrow WY$ (da 2) e $WY \rightarrow Z$ (dall'ipotesi), allora $WX \rightarrow Z$

[CVD]

Regola del prodotto

Date le dipendenze funzionali $X \rightarrow Y$ e $W \rightarrow Z$, allora vale $XW \rightarrow YZ$.

Esempio:

Se CF → MATR e CP → NP, allora CF, CP → MATR, NP.

Regola del prodotto

Date le dipendenze funzionali $X \rightarrow Y$ e $W \rightarrow Z$, allora vale $XW \rightarrow YZ$

Dimostrazione

1. Partiamo dall'assioma di **riflessività** $XW \rightarrow X$
2. Ma è data $X \rightarrow Y$, quindi per l'assioma di **transitività** $XW \rightarrow Y$
3. Per l'assioma di **riflessività** vale anche $XW \rightarrow W$
4. Ma è data $W \rightarrow Z$, quindi per l'assioma di **transitività** $XW \rightarrow Z$
5. Per l'assioma **dell'unione** su 2 e 4 abbiamo quindi $XW \rightarrow YZ$

[CVD]

Esercizio

Porre come assioma la proprietà dell'espansione e dimostrare l'unione.

fatto

Outline

- Introduzione
- Anomalie di inserimento, cancellazione e aggiornamento
- Dipendenze funzionali
- Teoria di Armstrong
- **Chiusura di un insieme di dipendenze funzionali**
- Chiusura di un insieme di attributi
- Dipendenze funzionali e superchiavi

Equivalenza

- Immaginiamo che un progettista P1 individui le dipendenze F e un altro progettista P2 individui le dipendenze G, dove G è sintatticamente diverso da F:
- P1 e P2 stanno dicendo la stessa cosa? Ovvero, F e G sono equivalenti?
- Secondo approccio al problema dell'equivalenza: due insiemi di dipendenze funzionali F e G sono equivalenti se e solo se l'insieme di dipendenze funzionali derivabili da F è uguale all'insieme di dipendenze funzionali derivabili da G.
- Formalizziamo questa intuizione con la definizione di **chiusura**.

Chiusura di un insieme F (definizione)

La **chiusura** di un insieme di dipendenze funzionali *F* è l'insieme F^+ di tutte le dipendenze funzionali derivabili da *F*.

(L'insieme F^+ è finito, infatti è finito il numero di modi di combinare un numero finito di attributi in forma di vincolo di dipendenza funzionale.)

Definizione di equivalenza

$F \equiv G$ (F è equivalente a G) se e solo se $F^+ = G^+$

Abbiamo ricondotto l'equivalenza a un'uguaglianza insiemistica verificabile sintatticamente:
se le dipendenze derivate da *F* sono le stesse dipendenze derivate da *G*, le due basi di dati evolvono allo stesso modo e quindi sono equivalenti.

Equivalenza

- Risulta però costoso costruire F^+ e G^+ per verificare l'equivalenza perché calcolare F^+ richiede tempo almeno esponenziale in $|F|$.
- Infatti, partendo da F=$\{A \rightarrow B_1, A \rightarrow B_2, ..., A \rightarrow B_n\}$ solo ($|F|=n$), applicando le regole della decomposizione e dell'unione, si ha che $F^+ \supseteq \{A \rightarrow Z \mid Z \subseteq B_1 B_2 ... B_n\}$; pertanto $|F^+| \geq 2^n$-1.

Equivalenza delle definizioni

Abbiamo visto finora **due** definizioni di equivalenza:

1) Primo approccio:
   F è equivalente a G se e solo se è possibile derivare G da F e viceversa (ad es. usando la definizione di dipendenza funzionale o le regole di Armstrong).
2) Secondo approccio:
   F è equivalente a G se e solo se F e G hanno la stessa chiusura.

Più formalmente…

Due definizioni di equivalenza

1) $F \equiv G$ (F è equivalente a G) se e solo se
$F \vdash G$ *(*G è deducibile da F*) e* $G \vdash F$ *(*F è deducibile da G*)*

- $F \vdash G$*:* presa una qualsiasi dipendenza g di G, g è deducibile da F.
- $G \vdash F$*:* presa una qualsiasi dipendenza f di F, f è deducibile da G.

2) $F \equiv G$ (F è equivalente a G) se e solo se $F^+ = G^+$.

Equivalenza delle definizioni

Le due definizioni

$$F \equiv G \text{ se e solo se } F \vdash G \text{ e } G \vdash F$$

e

$$F \equiv G \text{ se e solo se } F^+ = G^+$$

sono equivalenti?

Dobbiamo dimostrare che

$$F \vdash G \text{ e } G \vdash F \quad \text{se e solo se} \quad F^+ = G^+$$

Definizioni equivalenti

Vediamo per brevità solo un verso della dimostrazione (se $F \vdash G$ e $G \vdash F$, allora $G^+ = F^+$) e ricordiamo che l'uguaglianza insiemistica $F^+ = G^+$ significa che $F^+ \subseteq G^+$ e $G^+ \subseteq F^+$

Dimostriamo che, se $F \vdash G$ e $G \vdash F$, allora $G^+ \subseteq F^+$

1. Consideriamo una dipendenza $g \in G^+$
2. Per definizione di chiusura, g è derivabile da G: $G \vdash g$
3. Per le proprietà di deduzione (transitive), se $F \vdash G$ e $G \vdash g$, allora $F \vdash g$, cioè, per definizione di chiusura, $g \in F^+$
4. Abbiamo quindi dimostrato che, preso un elemento qualsiasi $g \in G^+$, vale anche $g \in F^+$, quindi $G^+ \subseteq F^+$

Definizioni equivalenti

Dimostriamo che, se $F \vdash G$ *e* $G \vdash F$, $F^+ \subseteq G^+$

1. Consideriamo una dipendenza $f \in F^+$
2. Per definizione di chiusura, f è deducibile da *F:* $F \vdash f$
3. Per le proprietà di deduzione (transitive), se $G \vdash F$ *e* $F \vdash f$, allora $G \vdash f$, cioè, per definizione di chiusura, $f \in G^+$
4. Abbiamo quindi dimostrato che, preso un elemento qualsiasi $f \in F^+$, vale anche $f \in G^+$, quindi $F^+ \subseteq G^+$

Valgono quindi sia $F^+ \subseteq G^+$ che $G^+ \subseteq F^+$, ovvero $F^+ = G^+$

[CVD]

Outline

- Introduzione
- Anomalie di inserimento, cancellazione e aggiornamento
- Dipendenze funzionali
- Teoria di Armstrong
- Chiusura di un insieme di dipendenze funzionali
- **Chiusura di un insieme di attributi**
- Dipendenze funzionali e superchiavi

Chiusura di un insieme di attributi

Per determinare l’equivalenza tra insiemi di dipendenze funzionali abbiamo visto finora due approcci entrambi troppo costosi da applicare.

Vedremo un approccio più efficiente, ma prima introduciamo la **chiusura di un insieme di attributi**.

Chiusura di un insieme di attributi

Dato un insieme di attributi R su cui è definito l'insieme di dipendenze funzionali F, dato un sottoinsieme $X \subseteq R$, la chiusura X_F^+ di X (o semplicemente X^+ se non ci sono ambiguità) è definita come

$$X_F^+ = \{A \mid X \rightarrow A \in F^+\}$$

Esempio

- Considero $X=\{MATR\}$ e ne calcolo la chiusura X_F^+ seguendo la definizione.
- Guardo le dipendenze F

{ MATR → NS, IS, CAP, CF, DN;

CF → MATR;

IS → CAP;

MATR, Co → Vo, DE, CP, NP;

CP → NP, Q;

Q → TU;

CP → Co }

Esempio

- $\{MATR\}_F^+$ dato F={ MATR → NS, IS, CAP, CF, DN; CF → MATR; IS → CAP; MATR, Co → Vo, DE, CP, NP; CP → NP, Q; Q → TU; CP → Co }
- Nella chiusura di *MATR* c'è sicuramente *MATR* stesso (per l'assioma di riflessività MATR→MATR $\in F^+$ e quindi $MATR \in X^+$).
- Da *MATR* → *NS, IS, CAP, CF, DN* deduco inoltre che $NS, IS, CAP, CF, DN \in X^+$.
- Le altre dipendenze in F non aggiungono ulteriori attributi.
- La chiusura di *X={MATR}* è quindi

$$\{MATR\}^+ = \{MATR, NS, IS, CAP, CF, DN\}$$

~~Algoritmo per il calcolo di X_F^+~~

~~Ad alto livello:~~

1. ~~Parti da X~~
2. ~~Finché in F ci sono d.f. applicabili~~

   ~~Ricava i nuovi attributi e ripeti~~

Algoritmo per il calcolo di X_F^+

Più nel dettaglio:

1. $C := X$
   $F' := F$
2. per ogni d.f. $Y \rightarrow Z$ in F' tale che $Y \subseteq C$
   $C := C \cup Z$
   $F' := F' - \{Y \rightarrow Z\}$

   return C

corretta es

Esempio

Calcolo $\{CP\}^+$

$C := CP, NP, Q, TU, CO$

$F' =$

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
~~CP → Co~~

C := X

F' := F

per ogni d.f. Y → Z in F' tale che
Y ⊆ C

C := C ∪ Z

F' := F' − {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
CP → NP, Q
Q → TU
CP → Co

C := X
F' := F
per ogni d.f. Y → Z in F' tale che $Y \subseteq C$
    C := C ∪ Z
    F' := F' – {Y → Z}
return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
CP → NP, Q
Q → TU
CP → Co

C := X

F' := F

per ogni d.f. Y → Z in F' tale che Y $\subseteq$ C

    C := C ∪ Z

    F' := F' – {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
CP → NP, Q
Q → TU
CP → Co

C := X
F' := F
per ogni d.f. Y → Z in F' tale che Y ⊆ C
    C := C ∪ Z
    F' := F' – {Y → Z}
return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
Q → TU
CP → Co

C := X

F' := F

per ogni d.f. Y → Z in F' tale che $Y \subseteq C$

C := C ∪ Z

F' := F' – {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F’ := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
Q → TU
CP → Co

C := X

F’ := F

per ogni d.f. Y → Z in F’ tale che Y ⊆ C

  C := C ∪ Z

  F’ := F’ – {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
Q → TU
CP → Co

C := X

F' := F

per ogni d.f. Y → Z in F' tale che Y ⊆ C

  C := C ∪ Z

  F' := F' – {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU
- C := {CP,NP,Q} ∪ {TU} = {CP,NP,Q,TU}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
CP → Co

C := X

F' := F

per ogni d.f. Y → Z in F' tale che Y ⊆ C

  C := C ∪ Z

  F' := F' – {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU
- C := {CP,NP,Q} ∪ {TU} = {CP,NP,Q,TU}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
CP → Co

C := X
F' := F
per ogni d.f. Y → Z in F' tale che Y ⊆ C
  C := C ∪ Z
  F' := F' – {Y → Z}
return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU
- C := {CP,NP,Q} ∪ {TU} = {CP,NP,Q,TU}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU}
- Esiste? Sì: CP → Co

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
CP → Co

C := X

F' := F

per ogni d.f. Y → Z in F' tale che Y ⊆ C

C := C ∪ Z

F' := F' – {Y → Z}

return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU
- C := {CP,NP,Q} ∪ {TU} = {CP,NP,Q,TU}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU}
- Esiste? Sì: CP → Co
- C := {CP,NP,Q,TU} ∪ {Co} = {CP,NP,Q,TU,Co}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
~~CP → Co~~

C := X
F' := F
per ogni d.f. Y → Z in F' tale che Y ⊆ C
    C := C ∪ Z
    F' := F' – {Y → Z}
return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU
- C := {CP,NP,Q} ∪ {TU} = {CP,NP,Q,TU}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU}
- Esiste? Sì: CP → Co
- C := {CP,NP,Q,TU} ∪ {Co} = {CP,NP,Q,TU,Co}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU,Co}

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
~~CP → Co~~

C := X
F' := F
per ogni d.f. Y → Z in F' tale che Y ⊆ C
  C := C ∪ Z
  F' := F' – {Y → Z}
return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipendenza funzionale in F' con antecedente in {CP}
- Esiste? Sì: CP → NP, Q
- C := {CP} ∪ {NP,Q} = {CP,NP,Q}
- Cerco una d.f. con antecedente in {CP,NP,Q}
- Esiste? Sì: Q → TU
- C := {CP,NP,Q} ∪ {TU} = {CP,NP,Q,TU}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU}
- Esiste? Sì: CP → Co
- C := {CP,NP,Q,TU} ∪ {Co} = {CP,NP,Q,TU,Co}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU,Co}
- Esiste? **No!**

F'
MATR → NS, IS, CAP, CF, DN
CF → MATR
IS → CAP
MATR, Co → Vo, DE, CP, NP
~~CP → NP, Q~~
~~Q → TU~~
~~CP → Co~~

C := X
F' := F
per ogni d.f. Y → Z in F' tale che Y ⊆ C
    C := C ∪ Z
    F' := F' – {Y → Z}
return C

Esempio

Calcolo $\{CP\}^+$

- Parto con X = {CP}
- C := {CP} e F' := F
- Cerco una dipe
  in F' con antece
- Esiste? Sì: CP →
- C := {CP} ∪ {NP,
- Cerco una d.f. c
- Esiste? Sì: Q →
- C := {CP,NP,Q} ∪
- Cerco una d.f. con antecedente in {CP,NP,Q,TU}
- Esiste? Sì: CP → Co
- C := {CP,NP,Q,TU} ∪ {Co} = {CP,NP,Q,TU,Co}
- Cerco una d.f. con antecedente in {CP,NP,Q,TU,Co}
- Esiste? **No!**

$$\{CP\}^+ = \{CP,NP,Q,TU,Co\}$$

F'
MATR → NS, IS, CAP, CF, DN CF → MATR IS → CAP → Vo, DE, CP, NP

per ogni d.f. Y → Z in F' tale che
Y ⊆ C
C := C ∪ Z
F' := F' – {Y → Z}
return C

Correttezza e completezza dell'algoritmo

Qual è il rapporto tra l'output C dell'algoritmo e la definizione $X_F^+ = \{A \mid X \rightarrow A \in F^+\}$?

L'algoritmo è *corretto*, cioè gli attributi C che ricava sono effettivamente in X_F^+ (cioè $C \subseteq X_F^+$).

- Intuitivamente gli attributi che aggiungiamo a C sono ricavabili in F^+ tramite gli assiomi di riflessività, transitività e unione.

L'algoritmo è *completo*, cioè tutti gli attributi in X_F^+ sono in C (cioè $X_F^+ \subseteq C$).

- Intuitivamente, data una qualsiasi d.f. $X \rightarrow A_i$ di F^+, prima o poi A_i viene aggiunto a C perché sono sufficienti gli assiomi di riflessività, transitività e unione per ricavare tutto F^+.

Quindi C= X_F^+. Tralasciamo la dimostrazione.

Complessità dell'algoritmo

La complessità dell'algoritmo è **polinomiale** rispetto al numero di dipendenze funzionali e al numero di attributi.

Infatti il ciclo contiene unione e differenza tra insiemi di attributi, operazioni che sono polinomiali nel numero di attributi, e viene eseguito una volta per ogni d.f.

Proprietà della chiusura di un insieme di attributi

- La ***chiusura di un insieme di dipendenze funzionali*** e la ***chiusura di un insieme di attributi*** sono strettamente legati.
- Infatti possiamo ricondurre il problema di decidere se una d.f. $X \rightarrow Y$ appartiene a F^+ alla verifica se Y è in X_F^+.

$$X \rightarrow Y \in F^+ \quad \textbf{se e solo se} \quad Y \subseteq X_F^+$$

Dimostrazione

- Definizione di X_F^+: $X_F^+ = \{A \mid X \rightarrow A \in F^+\}$
- Se $X \rightarrow Y \in F^+$, allora $Y \subseteq X_F^+$
  - Supponiamo che $Y=A_1 \dots A_n$
  - Per la regola della decomposizione, se $X \rightarrow A_1 \dots A_n \in F^+$, allora $X \rightarrow A_1 \in F^+, \dots, X \rightarrow A_n \in F^+$.
    Per definizione di X^+, $A_1 \in X^+, \dots, A_n \in X^+$ e quindi $Y \subseteq X^+$
- Se $Y \subseteq X_F^+$, allora $X \rightarrow Y \in F^+$
  - Se $Y \subseteq X^+$, per definizione di X^+, $X \rightarrow A_1 \in F^+, \dots, X \rightarrow A_n \in F^+$
  - Per la regola dell'unione $X \rightarrow Y \in F^+$.

[CVD]

Esempio

Dato F

{ ~~MATR → NS, IS, CAP, CF, DN~~;   matr, NS, cap, CF, DN, IS

~~CF → MATR~~;

~~IS → CAP~~;

MATR, Co → Vo, DE, CP, NP;

CP → NP, Q;

Q → TU;

CP → Co }

Verifichiamo se la dipendenza *MATR* → *CAP* appartiene a F^+.

Esempio

Per verificare se la d.f. *MATR* $\rightarrow$ *CAP* è in F^+, cioè se è possibile derivare *MATR* $\rightarrow$ *CAP* in F, ricaviamo $\{MATR\}_F^+$ e controlliamo se CAP è in $\{MATR\}_F^+$.

Precedentemente abbiamo già ottenuto
$\{MATR\}_F^+ = \{MATR, NS, IS, CAP, CF, DN\}$,
quindi possiamo concludere che *MATR* $\rightarrow$ *CAP* è una dipendenza valida nel sistema delle dipendenze F.

Esempio

Verifichiamo ora se MATR → Vo è derivabile in F.

Applichiamo l'algoritmo di chiusura su X={MATR} e, come prima, otteniamo $\{MATR\}_F^+=\{MATR, NS, IS, CAP, CF, DN\}$.

Vo non è incluso in X^+ quindi *MATR* → *Vo* non è deducibile da *F*.

Verifica di equivalenza

Possiamo ora sfruttare la chiusura degli attributi per verificare l'equivalenza tra due insiemi di dipendenze funzionali F e G.

<u>Terzo approccio al problema dell'equivalenza</u>:

- Controlliamo se **ogni d.f. *X*→*Y* di F** è deducibile in **G**… …verificando se **Y è nella chiusura di X <u>usando le d.f. di G</u>**, cioè se $Y \subseteq X_G^+$
- e viceversa:
  Controlliamo se **ogni d.f. *X*→*Y* di G** è deducibile in **F** verificando se **Y è nella chiusura di X**
  <u>usando le d.f. di F</u>, cioè se $Y \subseteq X_F^+$

Verifica di equivalenza

Cioè

$F \equiv G$ se e solo se

$$\boldsymbol{\forall (X \rightarrow Y) \in F (X \rightarrow Y \in G^{+}) \quad \text{e} \quad \forall (X \rightarrow Y) \in G (X \rightarrow Y \in F^{+})}$$

Dato che l'appartenenza di una d.f. a F^+ o a G^+ è risolvibile in modo efficiente tramite la chiusura degli attributi,

$F \equiv G$ se e solo se

$$\boldsymbol{\forall (X \rightarrow Y) \in F (Y \subseteq X_G^{+}) \quad \text{e} \quad \forall (X \rightarrow Y) \in G (Y \subseteq X_F^{+})}$$

[Matr := matr, NS, IS, DN, CF, Cap } in G

CF: CF, matr, NS, IS, DN, CAP

IS: IS Cap

Cp:= Cp, NP, Q, TU

Co

Q → Q, TU

Matr, Co Matr, Co

Vo, Lo, de,

Esempio

F (Progettista 1)	G (Progettista 2)
MATR → NS, IS, CAP, CF, DN	MATR → NS,IS
CF → MATR	MATR → DN,CF
IS → CAP	CF → MATR
MATR, Co → Vo,Lo, DE, CP, NP	IS → CAP
CP → NP, Q	MATR,Co → Vo,Lo,DE
Q → TU	MATR,Co,NS → CP
CP → Co	CP → NP,Q
	Q → TU
	CP → Co

Controlliamo se ogni vincolo di F può essere derivato in G (il viceversa è lasciato come esercizio)

Esempio

F (Progettista 1)	G (Progettista 2)
MATR → NS, IS, CAP, CF, DN CF → MATR IS → CAP MATR, Co → Vo,Lo, DE, CP, NP CP → NP, Q Q → TU CP → Co	MATR → NS,IS MATR → DN,CF CF → MATR IS → CAP MATR,Co → Vo,Lo,DE MATR,Co,NS → CP CP → NP,Q Q → TU CP → Co

Diversi vincoli sono uguali in F e in G; questi banalmente apparterranno alla chiusura di entrambi

Rimangono da verificare le seguenti d.f. in F:

- MATR → NS, IS, CAP, CF, DN
- MATR, Co → Vo, Lo, DE, CP, NP

Esempio

F (Progettista 1)	G (Progettista 2)
MATR → NS, IS, CAP, CF, DN CF → MATR IS → CAP MATR, Co → Vo,Lo, DE, CP, NP CP → NP, Q Q → TU CP → Co	MATR → NS,IS MATR → DN,CF CF → MATR IS → CAP MATR,Co → Vo,Lo,DE MATR,Co,NS → CP CP → NP,Q Q → TU CP → Co

Controlliamo se MATR → NS, IS, CAP, CF, DN di F appartiene a G^+.

Cioè controlliamo se, dati X=Matr e Y={NS, IS, CAP, CF, DN}, $X \rightarrow Y \in G^+$:

$\{X\}_G^+$={Matr,NS,IS,DN,CF,CAP} e $Y \subseteq \{X\}_G^+$: Sì

Esempio

F (Progettista 1)	G (Progettista 2)
MATR $\rightarrow$ NS, IS, CAP, CF, DN CF $\rightarrow$ MATR IS $\rightarrow$ CAP MATR, Co $\rightarrow$ Vo,Lo, DE, CP, NP CP $\rightarrow$ NP, Q Q $\rightarrow$ TU CP $\rightarrow$ Co	MATR $\rightarrow$ NS,IS MATR $\rightarrow$ DN,CF CF $\rightarrow$ MATR IS $\rightarrow$ CAP MATR,Co $\rightarrow$ Vo,Lo,DE MATR,Co,NS $\rightarrow$ CP CP $\rightarrow$ NP,Q Q $\rightarrow$ TU CP $\rightarrow$ Co

Controlliamo se MATR, Co $\rightarrow$ Vo, Lo, DE, CP, NP di F appartiene a G^+.

Cioè controlliamo se, dati X={Matr,Co} e Y={Vo,Lo,DE,CP,NP}, $X \rightarrow Y \in G^+$:

$\{X\}_G^+$=A (tutti gli attributi) e $Y \subseteq \{X\}_G^+$: Sì

Esempio

F (Progettista 1)	G (Progettista 2)
MATR → NS, IS, CAP, CF, DN CF → MATR IS → CAP MATR, Co → Vo,Lo, DE, CP, NP CP → NP, Q Q → TU CP → Co	MATR → NS,IS MATR → DN,CF CF → MATR IS → CAP MATR,Co → Vo,Lo,DE MATR,Co,NS → CP CP → NP,Q Q → TU CP → Co

Si può verificare come esercizio l'altro verso e concludere che F ≡ G

Outline

- Introduzione
- Anomalie di inserimento, cancellazione e aggiornamento
- Dipendenze funzionali
- Teoria di Armstrong
- Chiusura di un insieme di dipendenze funzionali
- Chiusura di un insieme di attributi
- **Dipendenze funzionali e superchiavi**

Applicazione: ricerca chiavi

Ricordiamo la definizione di superchiave.

Data la relazione r(A), l'insieme di attributi $K \subseteq A$ è una superchiave se e solo se:

$$\forall t_i, t_j \in r(t_i[K] = t_j[K] \Rightarrow t_i[A] = t_j[A])$$

La definizione è simile alla definizione di vincolo di dipendenza funzionale.

In particolare è identica alla definizione di $K \rightarrow \boldsymbol{A}$ in cui la chiave determina tutti gli attributi di r.

Nuova definizione di superchiave

Dato uno schema di relazione *R(A)* con un insieme di dipendenze funzionali F, un insieme di attributi $K \subseteq A$ è **superchiave** se e solo se $A = K_F^+$
(cioè se e solo se in F^+ si trova il vincolo di dipendenza funzionale $K \rightarrow A$).

Esempio (superchiave)

Dati

ESAMI(MATR, NS, IS, CAP, CF, DN, Co, Vo, Lo, DE, CP, NP, Q, TU) e

F = {

MATR → NS, IS, CAP, CF, DN;

CF → MATR;

IS → CAP;

MATR, Co → Vo, Lo, DE, CP, NP;

CP → NP, Q;

Q → TU;

CP → Co }

MATR è superchiave della relazione ESAMI?

- Abbiamo già visto che la chiusura di MATR è {MATR,NS,IS,CAP,CF,DN}, che non comprende tutti gli attributi di ESAMI. La congettura è quindi sbagliata.

{MATR,Co,NS} è superchiave?

Esempio (superchiave)

Applicazione dell'algoritmo con X={MATR,Co,NS}

- Sappiamo già che il sottoinsieme {MATR} ci dà {MATR,NS,IS,CAP,CF,DN}, quindi

  C={MATR,NS,IS,CAP,CF,DN,Co}

- Inoltre {MATR,Co} ci dà anche {Vo,Lo,DE,CP}, quindi

  C={MATR,NS,IS,CAP,CF,DN,Co,Vo,Lo,DE,CP}

- {CP} ci dà {NP,Q} quindi

  C={MATR,NS,IS,CAP,CF,DN,Co,Vo,Lo,DE,CP,NP,Q}

- {Q} ci dà {TU}, quindi

  C={MATR,NS,IS,CAP,CF,DN,Co,Vo,Lo,DE,CP,NP,Q,TU}

Esempio (superchiave)

Abbiamo quindi che

$X=\{MATR,Co,NS\}$ e

$X^{+}=\{MATR,NS,IS,CAP,CF,DN,Co,Vo,Lo,DE,CP,NP,Q,TU\}=A$

cioè: $X \rightarrow A$

Quindi $X=\{MATR,Co,NS\}$ è **superchiave**

Esempio (chiave candidata)

{MATR,Co,NS} è chiave?

Una chiave (candidata) è una superchiave minimale,
bisogna quindi ancora verificare se {MATR,Co,NS} è minimale:

- Provo la chiusura di {Co,NS} e la proprietà di superchiave **non è** verificata (**esercizio**).
- Provo la chiusura di {MATR,Co} e verifico che **è** superchiave (**esercizio**), quindi {MATR,Co,NS} non è minimale.

{MATR,Co} è chiave?

- So già che MATR non è superchiave, quindi {MATR,Co} è minimale ed è una chiave candidata.

Esempio (chiave candidata)

Le chiavi candidate possono essere più di una.

Nel nostro esempio possiamo scoprire che sono chiavi candidate:

- {MATR,Co}
- {CF,Co}
- {MATR,CP}
- {CF,CP}

Esempio (chiave candidata)

Verifica che $K=\{MATR,CP\}$ sia una chiave:

- La chiusura di MATR ci dà {**MATR**,NS,IS,CAP,CF,DN}
- La chiusura di CP ci dà {CP,NP,Q,**Co**}

Quindi, in particolare, ho ottenuto {MATR,Co}, e ricado nell’esempio precedente.

Quindi anche in questo caso $K^{+}=A$.

Ho già calcolato che né MATR né CP sono superchiavi, quindi $K=\{MATR,CP\}$ è una chiave.

Perciò ho quattro diverse chiavi candidate a diventare chiave primaria di ESAMI.

Proprietà

Se due progettisti identificano, su una medesima base di dati, due insiemi di dipendenze funzionali F e G equivalenti, i due progettisti arriveranno alle **stesse identiche** chiavi candidate.

(Infatti F e G avranno la stessa chiusura)

Esempio

Immaginiamo ora una relazione conto corrente CC così composta:

CC(Titolare, NumeroContoCorrente, NumeroAgenzia, CittaAgenzia, DirettoreAgenzia, SaldoContoCorrente, DataUltimaMovimentazione)

Abbreviando:

CC(T,NCC,NA,CA,DA,SCC,DM)

Esempio

Cerchiamo ora delle dipendenze funzionali significative su

CC(Titolare, NumeroContoCorrente, NumeroAgenzia, CittaAgenzia, DirettoreAgenzia, SaldoContoCorrente, DataUltimaMovimentazione)

CC(T,NCC,NA,CA,DA,SCC,DM)

F:

{ NCC → NA,CA

NCC → SCC,DM

NA,CA → DA (il direttore è l'unico direttore dell'agenzia)

DA → CA (un direttore non può dirigere agenzie in città diverse) }

Esempio

Vogliamo ricavare le chiavi candidate di

CC(T,NCC,NA,CA,DA,SCC, DM)

dato F={NCC → NA,CA; NCC → SCC,DM; NA,CA → DA; DA → CA}.

Notiamo che l'attributo T *non appare nel conseguente di nessuna dipendenza funzionale*. Infatti, uno stesso conto corrente può essere intestato a più persone (quindi non ho il vincolo NCC→T).

Supponiamo di usare il procedimento descritto sopra per la ricerca delle chiavi candidate: è impossibile che *T* compaia nella chiusura di qualsiasi insieme di attributi X a meno che non faccia già parte di X.

Di conseguenza, *T* **deve fare parte** di ogni chiave candidata!

Esempio

Vogliamo ricavare le chiavi candidate di

CC(T,NCC,NA,CA,DA,SCC, DM)

dato F={NCC → NA,CA; NCC → SCC,DM; NA,CA → DA; DA → CA}.

Inoltre: *T non appare nell'antecedente di nessuna d.f.*, quindi la chiusura di T comprende solo T stesso e da solo non può essere una superchiave.

Posso quindi considerare, ad es., *K={T,NCC}*.

Nota bene

1. Gli attributi coinvolti dalle dipendenze funzionali **non necessariamente** coprono tutti gli attributi della relazione.
2. Quindi gli attributi della relazione non coinvolti dalle dipendenze funzionali **devono sempre** fare parte delle chiavi candidate congetturate (infatti non possono essere derivati tramite le dipendenze funzionali).
3. Una chiave K su una relazione R(A) implica una d.f. K→A anche quando non è dichiarata esplicitamente.