Dipendenze Funzionali

Progettazione

- Abbiamo ipotizzato che gli attributi vengano raggruppati per formare uno schema di relazione **usando il buon senso** del progettista di basi di dati o **traducendo** uno schema di base di dati da un modello dei dati concettuale (E-R), **presumibilmente ben fatto**
- Ma abbiamo bisogno di **misurare formalmente** perché un raggruppamento di attributi in uno schema di relazione possa essere **migliore** di un altro
- **Obiettivo**: valutare la qualità della progettazione degli schemi relazionali

Approccio seguito

- **Top-down**:
  - abbiamo iniziato individuando un certo numero di **raggruppamenti di attributi** per formare relazioni che sussistono come tali nel mondo reale, ad esempio una fattura, un form o un report.
  - Queste relazioni sono poi state analizzate portando eventualmente a **decomposizioni successive**.

Obiettivi impliciti del progetto logico

- La **conservazione dell'informazione**, cioè il mantenimento di tutti i concetti espressi precedentemente mediante il modello concettuale, inclusi tipi di attributi, tipi di entità e tipi di associazioni.
- La **minimizzazione della ridondanza**, cioè l'evitare la memorizzazione ripetuta della stessa informazione, e quindi la necessità di effettuare molteplici aggiornamenti al fine di mantenere la consistenza tra le diverse copie della medesima informazione.
- Possiamo derivare da questi obiettivi **alcune linee guida** per il progetto

Linea Guida 1: semplice è bello

- Uno schema di relazione deve essere progettato in modo che **sia semplice spiegarne il significato**.
- Non si devono raggruppare attributi provenienti da più tipi di entità e tipi di relazione in un'unica relazione.
- Intuitivamente, se uno schema di relazione corrisponde a **un solo tipo di entità** o a **un solo tipo di *relationship***, risulta **semplice spiegarne il significato**.
- In caso contrario, nascerà **un'ambiguità semantica** e quindi lo schema non potrà essere spiegato con facilità.

Linea Guida 2: no alle anomalie

- Gli schemi vanno progettati in modo che **non possano presentarsi anomalie** di inserimento, cancellazione o modifica.
- La **mancanza** di anomalie va **certificata** usando una **descrizione formale della semantica** dei fatti descritti in uno schema relazionale
- Se possono presentarsi anomalie, vanno chiaramente **rilevate** e si deve assicurare che i programmi che aggiornano la base di dati operino **correttamente**.

Esempio 1

- **Fattura**(CodFatt, CodProd, TotDaPagare, CostoNettoProd, IVA)
- Semantica attributi:
  - CodFatt determina CodProd e TotDaPagare
  - CodProd determina CostoNettoProd e IVA
  - CostoNettoProd e IVA determinano TotDaPagare

Esempio 1

- Ovviamente TotDaPagare deve essere consistente con la regola che lo lega al CostoNettoProd e all'IVA
- Inoltre a CodProd deve essere attribuita la giusta percentuale di IVA
- Questo secondo legame è esterno al DB, se cambia per legge l'IVA di un certo prodotto, questo attributo deve essere modificato; però la sua modifica si porta dietro un'altra modifica dell'attributo TotDaPagare il cui significato è interno al DB ma è legato ad IVA.
- Per evitare anomalie di inserimento o modifica conviene che TotDaPagare non ci sia nella tabella Fattura

Esempio 2

- **Anagrafe**(CF, NomePersona, ViaRes, NomeCittaRes, NumAb)
- Semantica attributi:
  - CF determina NomePersona, ViaRes e NomeCittaRes
  - NomeCittaRes determina NumAb

Esempio 2

- NumAb è ripetuto per lo stesso NomeCittaRes per quanti sono i residenti
- Il valore deve essere mantenuto consistente (uguale) per ogni persona di una stessa città
- Come si può evitare il problema?
  - Trasformando Anagrafe in due schemi separati
    - Persona(CF, NomePersona, ViaRes, NomeCittaRes)
    - ListaComuni(NomeCitta, NumAb)
    - Con vincolo di integrità referenziale su NomeCittaRes verso NomeCitta e un vincolo aggiuntivo su NumAb...

Linea Guida 3: evitare frequenti valori nulli

- Si eviti di porre in una relazione attributi i cui valori possono essere frequentemente nulli.
- Se i valori nulli sono inevitabili, ci si assicuri che si presentino solo in casi eccezionali rispetto al numero di n-uple di una relazione.

Dipendenza Funzionale

- Una **dipendenza funzionale** (*functional dependency*, FD) esprime un **legame semantico** tra **due gruppi di attributi** di uno schema di relazione R
- Una FD è una proprietà di R, non di un particolare stato valido r di R
- Una FD non può essere dedotta a partire da uno stato valido r, ma deve essere definita esplicitamente da qualcuno che conosce la semantica degli attributi di R

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 normalizzata:
  - presenta **ridondanze**
  - si presta a **comportamenti poco desiderabili** durante gli aggiornamenti
- Le forme normali sono di solito definite sul **modello relazionale**, ma hanno senso in altri contesti, ad esempio il modello E-R

Normalizzazione

- Procedura che permette di **trasformare schemi** non normalizzati in schemi che soddisfano una forma normale
- La normalizzazione va utilizzata come **tecnica di verifica** dei risultati della progettazione di una base di dati
- **Non costituisce una metodologia di progettazione**

Relazione con anomalie

Impiegato	Stipendio	Progetto	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

Anomalie

- Lo stipendio di ciascun impiegato è ripetuto in tutte le n-uple relative
  - **ridondanza**
- Se lo stipendio di un impiegato varia, è necessario andarne a modificare il valore in diverse n-uple
  - **anomalia di aggiornamento**
- Se un impiegato interrompe la partecipazione a tutti i progetti, dobbiamo cancellarlo
  - **anomalia di cancellazione**
- Un nuovo impiegato senza progetto non può essere inserito
  - **anomalia di inserimento**

Causa dei problemi

- Abbiamo usato **un'unica relazione** per rappresentare **informazioni eterogenee**
  - gli impiegati con i relativi stipendi
  - i progetti con i relativi bilanci
  - le partecipazioni degli impiegati ai progetti con le relative funzioni
- Ora useremo il concetto di dipendenza funzionale per studiare meglio questi problemi

Definizione di dipendenza funzionale

- Dati:
  - una relazione r su $R(X)$,
  - due sottoinsiemi **non vuoti** Y e Z di X,
- esiste in r una **dipendenza funzionale** da Y a Z se, per ogni coppia di n-uple t_1 e t_2 di r con gli stessi valori su Y, risulta che t_1 e t_2 hanno gli stessi valori anche su Z
- Notazione: $Y \to Z$
  - Nota: Se $Y \to Z$, non è detto che esista $Z \to Y$

Dipendenze funzionali particolari

- Una dipendenza funzionale è **completa** quando $Y \to Z$ e, per ogni $W \subset Y$, non vale $W \to Z$
- Se Y è una **superchiave** di $R(X)$, allora Y determina ogni altro attributo della relazione, i.e., $Y \to X$
- Se Y è una **chiave**, allora $Y \to X$ è una dipendenza funzionale completa
- Una dipendenza funzionale è **banale** se è sempre soddisfatta
  - $Y \to Y$ è banale
  - $Y \to A$ è non banale se $A \notin Y$
  - $Y \to Z$ è non banale se nessun attributo di Z appartiene a Y

Esempio 1

- Caratterizziamo in termini di dipendenze le informazioni semantiche che abbiamo
  - Ogni impiegato ha un solo stipendio
    - Impiegato → Stipendio
  - Ogni progetto ha un solo bilancio
    - Progetto → Bilancio
  - Ogni impiegato ha una sola funzione per progetto
    - Impiegato, Progetto → Funzione

Esempio 1

Impiegato	Stipendio	Progetto	Bilancio	Funzione
Rossi	20	Marte	2	tecnico
Verdi	35	Giove	15	progettista
Verdi	35	Venere	15	progettista
Neri	55	Venere	15	direttore
Neri	55	Giove	15	consulente
Neri	55	Marte	2	consulente
Mori	48	Marte	2	direttore
Mori	48	Venere	15	progettista
Bianchi	48	Venere	15	progettista
Bianchi	48	Giove	15	direttore

- Impiegato $\rightarrow$ Stipendio
- Progetto $\rightarrow$ Bilancio
- Impiegato, Progetto $\rightarrow$ Funzione

Legami tra dipendenze funzionali e anomalie

<u>Impiegato</u>	Stipendio	<u>Progetto</u>	Bilancio	Funzione

- Impiegato → Stipendio
  - Ci sono ripetizioni
- Progetto → Bilancio
  - Ci sono ripetizioni
- Impiegato, Progetto → Funzione
  - Non ci sono ripetizioni
- Impiegato non è una chiave
- Progetto non è una chiave
- Impiegato, Progetto è una chiave

Legami tra dipendenze funzionali e anomalie

- Le dipendenze funzionali sono usate per **verificare l'eventuale presenza di anomalie** in un progetto
  - Vedremo che sono usate anche per "normalizzare" uno schema
- Data la loro importanza, quando necessario indicheremo con $R(X, F)$ uno schema di relazione $R(X)$ che **verifica un insieme di dipendenze funzionali** F

Implicazione

- Sia F un insieme di dipendenze funzionali definite su $R(Z)$ e sia $X \rightarrow Y$
  - Si dice che F **implica (logicamente)** $X \rightarrow Y$, in simboli $F \models X \rightarrow Y$, se, **per ogni possibile istanza** r di R che verifica tutte le dipendenze funzionali in F, risulta verificata **anche** la dipendenza funzionale $X \rightarrow Y$
  - Si dice anche che $X \rightarrow Y$ **è implicata (logicamente) da** F
- Esempio:
  - R(Impiegato, Categoria, Stipendio)
  - Le dipendenze funzionali
    - Impiegato → Categoria e
    - Categoria → Stipendio
  - implicano la dipendenza funzionale
    - Impiegato → Stipendio

Problema

- La definizione di implicazione **non è direttamente utilizzabile** nella pratica
  - Essa prevede una **quantificazione universale** sulle istanze della base di dati ("per ogni istanza ...")
  - **Non abbiamo un algoritmo** per calcolare tutte le dipendenze funzionali implicate da un insieme F
- **Armstrong** (1974) ha fornito delle **regole di inferenza** che permettono di **derivare costruttivamente** tutte le dipendenze funzionali che sono implicate da un dato insieme iniziale

Regole di inferenza di Armstrong

1. **Riflessività**:

   $$\text{Se } Y \subseteq X \text{ allora } X \to Y$$

2. **Additività** (o **arricchimento**):

   $$\text{Se } X \to Y \text{ allora } XZ \to YZ \text{ per qualunque } Z$$

3. **Transitività**:

   $$\text{Se } X \to Y \text{ e } Y \to Z \text{ allora } X \to Z$$

Derivazione

- Dati:
  - un **insieme** di **regole di inferenza** RI,
  - un **insieme** di **dipendenze funzionali** F e
  - una **dipendenza funzionale** f,
- una **derivazione di f da F secondo RI** è una **sequenza** finita $f_1, \ldots, f_m$ dove
  - $f_m = f$
  - ogni f_i è un elemento di F oppure è ottenuta dalle precedenti dipendenze $f_1, \ldots, f_{i-1}$ della derivazione usando una regola di inferenza RI
- Indichiamo con $F \vdash X \rightarrow Y$ il fatto che la **dipendenza funzionale** $X \rightarrow Y$ sia **derivabile** da F **usando** RI

Regole di derivazione comuni

- **Unione**:

$$\{X \to Y, X \to Z\} \vdash X \to YZ$$

- **Decomposizione**:

$$\{X \to YZ\} \vdash X \to Y$$

- **Indebolimento**:

$$\{X \to Y\} \vdash XZ \to Y$$

- **Identità**:

$$\{\} \vdash X \to X$$

Unione: dimostrazione

- **Unione**:

$$\{X \to Y, X \to Z\} \vdash X \to YZ$$

- *Dimostrazione*:

1. $X \to Y$ per ipotesi
2. $X \to XY$ per additività da 1
3. $X \to Z$ per ipotesi
4. $XY \to YZ$ per additività da 3
5. $X \to YZ$ per transitività da 2 e 4

Decomposizione: dimostrazione

- **Decomposizione**:

$$\{X \to YZ\} \vdash X \to Y$$

- *Dimostrazione*:

1. $X \to YZ$ per ipotesi
2. $YZ \to Y$ per riflessività
3. $X \to Y$ per transitività da 1 e 2

Indebolimento: dimostrazione

- **Indebolimento**:

$$\{X \to Y\} \vdash XZ \to Y$$

- *Dimostrazione*:

1. $XZ \to X$ per riflessività
2. $X \to Y$ per ipotesi
3. $XZ \to Y$ per transitività da 1 e 2

Chiusura di un insieme di attributi

- Dato uno schema $R(T, F)$ con $X \subseteq T$, **la chiusura di** X **rispetto a** F, indicata col simbolo X_F^+, è definita come

$$X_F^+ = \{A \in T \,|\, F \vdash X \to A\}$$

- Se non vi sono ambiguità scriveremo semplicemente X^+
- *Ritorneremo avanti su questa definizione, con qualche esempio*

Teorema della chiusura degli attributi

- **Teorema**:

$$F \vdash X \rightarrow Y \Leftrightarrow Y \subseteq X^{+}$$

Teorema della chiusura degli attributi

- **Teorema**:

$$F \vdash X \to Y \Leftrightarrow Y \subseteq X^+$$

- *Dimostrazione*

  Sia $Y = A_1 \cdots A_k$.

  ($\Rightarrow$) Per la **regola di decomposizione** abbiamo $F \vdash X \to A_i$; per definizione di X^+, $A_i \in X^+$ e quindi anche $Y \subseteq X^+$.

  ($\Leftarrow$) Per definizione di X^+, $F \vdash X \to A_i$. Per la **regola dell'unione**, $F \vdash X \to A_1 \cdots A_k$, cioè $F \vdash X \to Y$.

Correttezza e Completezza

- Dato un qualche insieme di regole di inferenza RI e un insieme di dipendenze funzionali F
  - RI è **corretto** se

$$F \vdash X \to Y \Rightarrow F \models X \to Y$$

  - *Applicando RI a un insieme F di dipendenze funzionali, si ottengono solo dipendenze logicamente implicate da F*
  - RI è **completo** se

$$F \models X \to Y \Rightarrow F \vdash X \to Y$$

  - *Applicando RI a un insieme F di dipendenze funzionali, si ottengono tutte le dipendenze logicamente implicate da F*

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione*:

  Supponiamo di avere un insieme di dipendenze funzionali F su T, e una dipendenza funzionale $X \to Y$

  Per prima cosa dimostriamo la correttezza:

  $$\text{Se } F \vdash X \to Y \text{ allora } F \models X \to Y$$

  Per seconda cosa dimostriamo la completezza

  $$\text{Se } F \models X \to Y \text{ allora } F \vdash X \to Y$$

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \vdash X \rightarrow Y$ *allora* $F \models X \rightarrow Y$

  Si procede per induzione sulla lunghezza della derivazione.

  Sia $f_1, \ldots, f_m$ la derivazione di $X \rightarrow Y$ da F e supponiamo che il teorema valga per tutte le derivazioni di lunghezza pari a $1, \ldots, m-1$.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \vdash X \to Y$ *allora* $F \models X \to Y$

  La dipendenza $f_m = X \to Y$ è un elemento di F oppure è stata derivata usando una regola di inferenza di Armstrong.

  Se è un elemento di F allora è implicata logicamente in maniera banale.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \vdash X \rightarrow Y$ *allora* $F \models X \rightarrow Y$

  Se f_m è stata inferita con la regola di riflessività allora $Y \subseteq X$ e l'implicazione logica è banale.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \vdash X \rightarrow Y$ *allora* $F \models X \rightarrow Y$

  Se f_m è stata inferita con la regola di additività da una $f_i = X' \rightarrow Y'$, allora per qualche Z si deve avere $X = X'Z$ e $Y = Y'Z$. Per ipotesi induttiva $F \models f_i$.

  Siano t_1 e t_2 due n-uple con $t_1[X'Z] = t_2[X'Z]$. Per definizione $t_1[X'] = t_2[X']$; per f_i, $t_1[Y'] = t_2[Y']$; per arricchimento $t_1[Y'Z] = t_2[Y'Z]$. Quindi $F \models X \rightarrow Y$.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \vdash X \to Y$ *allora* $F \models X \to Y$

  Se f_m è stata inferita con la regola di transitività da $f_i = X \to W$ e $f_j = W \to Y$ per un qualche W. Per ipotesi induttiva $F \models f_i$ e $F \models f_j$.

  Siano t_1 e t_2 due n-uple con $t_1[X] = t_2[X]$. Per f_i, $t_1[W] = t_2[W]$ e per f_j, $t_1[Y] = t_2[Y]$. Quindi $F \models X \to Y$.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \models X \rightarrow Y$ *allora* $F \vdash X \rightarrow Y$

  Consideriamo una relazione di due n-uple $r = \{t_1, t_2\}$ su T con $t_1[X^+] = t_2[X^+]$ e $t_1[A] \neq t_2[A]$ per ogni $A \in T - X^+$.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \models X \rightarrow Y$ *allora* $F \vdash X \rightarrow Y$

  Dimostriamo che la relazione r soddisfa F.

  Sia $V \rightarrow W \in F$.

  Se $V \nsubseteq X^+$ allora $t_1[V] \neq t_2[V]$ e r soddisfa la dipendenza.

  Se $V \subseteq X^+$ allora $F \vdash X \rightarrow V$ e per transitività $F \vdash X \rightarrow W$, da cui $W \subseteq X^+$ e quindi $t_1[W] = t_2[W]$ e r soddisfa la dipendenza

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**

- *Dimostrazione: se* $F \models X \rightarrow Y$ *allora* $F \vdash X \rightarrow Y$

  Siccome $F \models X \rightarrow Y$, la relazione r soddisfa $X \rightarrow Y$.

  Poiché $X \subseteq X^+$ e $t_1[X^+] = t_2[X^+]$, allora $t_1[Y] = t_2[Y]$.

  Quindi $Y \subseteq X^+$ e $F \vdash X \rightarrow Y$ per il teorema di chiusura degli attributi.

Teorema

- **Le regole di inferenza di Armstrong sono corrette e complete**
- Questo teorema ci permette di scambiare $\models$ con $\vdash$ ovunque. In particolare nella definizione di chiusura degli attributi, cioè

$$X_F^+ = \{A \in T \mid F \models X \to A\}$$

- Si può dimostrare che le regole di inferenza di Armstrong sono **minimali**, cioè **ignorando** anche una sola di esse, l'insieme di regole che rimangono **non è più completo.**
- Le regole di inferenza di Armstrong non sono l'unico insieme di regole corretto e completo!

Chiusura di un insieme di dipendenze funzionali

- Sia F un insieme di dipendenze funzionali definite su $R(Z)$
  - La **chiusura di** F è l'insieme F^+ di **tutte** le dipendenze funzionali implicate da F:

$$F^+ = \{X \rightarrow Y | F \Rightarrow X \rightarrow Y\}$$

- Dato un insieme di dipendenze funzionali F definite su $R(Z)$, un'istanza r di R che soddisfa F soddisfa anche le dipendenze funzionali di F^+

Calcolo di F^+

- Possiamo usare le regole di Armstrong per calcolare F^+

  Input: $R(T, F)$

  Output: F^+

  $F^+ \leftarrow F$

  while (F^+ non cambia) **do**

      for each $f \in F^+$ **do**

          applicare riflessività e additività a f e aggiungere a F^+ le dipendenze ottenute

      for each $f_1, f_2 \in F^+$ **do**

          se possibile, applicare transitività a f_1 e f_2 e aggiungere a F^+ la dipendenza ottenuta

  return F^+

Esempio 2

- Dati
  - $R(ABCGHI)$
  - $F = \{A \rightarrow B, A \rightarrow C, CG \rightarrow H, CG \rightarrow I, B \rightarrow H\}$
- Alcuni membri di F^+ sono:
  - $A \rightarrow H$
    - per transitività da $A \rightarrow B$ e $B \rightarrow H$
  - $AG \rightarrow I$
    - arricchendo $A \rightarrow C$ con G e per transitività con $CG \rightarrow I$
  - $CG \rightarrow HI$
    - arricchendo $CG \rightarrow I$ con CG, arricchendo $CG \rightarrow H$ con I e per transitività

F^+ e X^+

- Il calcolo di F^+ è **molto costoso**
  - **complessità esponenziale** nel numero di attributi dello schema nel caso peggiore
- Spesso però quello che ci interessa è **verificare** se F^+ contiene una certa dipendenza e **non generare** l'intera chiusura
- Per fare ciò basta calcolare X^+ per il teorema di chiusura degli attributi

$$F \vdash X \to Y \Leftrightarrow Y \subseteq X^+$$

$$(F \models X \to Y \Leftrightarrow Y \subseteq X^+)$$

Calcolo di X^+

Input: $R(T, F)$, $X \subseteq T$

Output: X^+

$X^+ \leftarrow X$

while (X^+ non cambia) **do**

  for each $W \rightarrow V \in F$ **do**

    if $W \subseteq X^+$ **and** $V \nsubseteq X^+$ **then**

      $X^+ \leftarrow X^+ \cup V$

return X^+

Esempio 3

- Dati
  - $R(ABCDE)$
  - $F = \{A \rightarrow B, BC \rightarrow D, B \rightarrow E, E \rightarrow C\}$
- Calcoliamo A^+:
  - $A^+ \leftarrow A$
  - $A^+ \leftarrow AB$ perché $A \rightarrow B$ e $A \subseteq A^+$
  - $A^+ \leftarrow ABE$ perché $B \rightarrow E$ e $B \subseteq A^+$
  - $A^+ \leftarrow ABEC$ perché $E \rightarrow C$ e $E \subseteq A^+$
  - $A^+ \leftarrow ABECD$ perché $BC \rightarrow D$ e $BC \subseteq A^+$
- Possiamo concludere che A è superchiave (e anche chiave)

Chiavi

- Dato uno schema $R(T, F)$
  - Un insieme di attributi $K \subseteq T$ si dice **superchiave** di R se la dipendenza funzionale $K \to T$ è implicata da F, ovvero se $K \to T \in F^+$
  - Un insieme di attributi $K \subseteq T$ si dice **chiave** di R se W è una superchiave di R e se non esiste alcum sottoinsieme proprio di K che sia superchiave di R
- Dato che in uno schema ci possono essere più chiavi, di solito ne viene scelta una, detta **chiave primaria**, come identificatore delle n-uple delle istanze dello schema

Trovare tutte le chiavi

- Il problema di **trovare tutte le chiavi** di una relazione $R(Z)$ richiede un algoritmo di **complessità esponenziale** nel caso pessimo
- Cosa si deve fare:
  - Gli attributi che stanno solo a sinistra stanno in tutte le chiavi, chiamiamo N questo insieme
  - Gli attributi che stanno solo a destra non stanno in nessuna chiave
  - Si aggiunge a N un attributo alla volta tra quelli che non stanno solo a destra, poi una coppia di attributi e così via, chiamiamo X_i questo sottoinsieme di attributi, ogni volta si controlla se la dipendenza $N \cup X_i \rightarrow Z$ esiste

Verificare una chiave

- L'algoritmo per il calcolo della chiusura di un insieme di attributi può essere usato per **verificare se un insieme di attributi è chiave o superchiave**
- $X \subseteq T$ è superchiave di $R(T, F)$
  - se e solo se $X \rightarrow T \in F^+$, ovvero
  - se e solo se $T \subseteq X^+$
- $X \subseteq T$ è chiave di $R(T, F)$
  - se e solo se $T \subseteq X^+$, e non esiste $Y \subset X$ tale che $T \subseteq Y^+$

Equivalenza

- **Due insiemi di dipendenze funzionali** F e G sugli attributi T di una relazione $R(T)$ sono **equivalenti**, in simboli $F \equiv G$, se e solo se $F^+ = G^+$
  - Se $F \equiv G$ allora F è una **copertura** di G e viceversa
- La relazione di equivalenza permette di stabilire se due schemi di relazione rappresentano **gli stessi fatti**
  - Basta che abbiano gli stessi attributi e dipendenze funzionali equivalenti
- Per **verificare l'equivalenza** è sufficiente che
  - tutte le dipendenze di F appartengano a G^+
  - tutte le dipendenze di G appartengano a F^+

Esempio 4

- Verificare se F e G sono equivalenti:
  - $F = \{A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H\}$
  - $G = \{A \rightarrow CD, E \rightarrow AH\}$
- Verificare che tutte le dipendenze di F appartengano a G^+
  - $A \rightarrow CD \Rightarrow A \rightarrow C$
  - $A \rightarrow CD \Rightarrow AC \rightarrow CD \Rightarrow AC \rightarrow D$
  - $E \rightarrow AH \Rightarrow E \rightarrow H$
  - $E \rightarrow AH \Rightarrow E \rightarrow A \Rightarrow E \rightarrow AE$
  - $A \rightarrow CD \Rightarrow A \rightarrow D \Rightarrow A \rightarrow AD \Rightarrow AE \rightarrow ADE$
  - $E \rightarrow ADE \Rightarrow E \rightarrow AD$

Esempio 4

- Verificare se F e G sono equivalenti:
  - $F = \{A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H\}$
  - $G = \{A \rightarrow CD, E \rightarrow AH\}$
- Verificare che tutte le dipendenze di G appartengano a F^+
  - $A \rightarrow C \Rightarrow A \rightarrow AC, AC \rightarrow D \Rightarrow A \rightarrow D \Rightarrow A \rightarrow CD$
  - $E \rightarrow AD \Rightarrow E \rightarrow A, E \rightarrow H \Rightarrow E \rightarrow AH$

Esempio 5

- Verificare se F e G sono equivalenti:
  - $F = \{A \to C, AC \to D, E \to AD, E \to H\}$
  - $G = \{A \to CD, E \to AH\}$
- Invece di verificare se $X \to Y \in F$ è anche in G^+ e viceversa, possiamo verificare se $Y \subseteq X_G^+$ e, viceversa, per ogni dipendenza funzionale
- Per esempio (verifichiamo F su X_G^+):
  - $A \to C$: $A_G^+ = ACD$, quindi $C \in A_G^+$
  - $AC \to D$: $AC_G^+ = ACD$, quindi $D \in AC_G^+$
  - $E \to AD$: $E_G^+ = EADCH$, quindi $AD \in E_G^+$
  - $E \to H$: $E_G^+ = EADCH$, quindi $H \in E_G^+$

Esempio 5

- Verificare se F e G sono equivalenti:
  - $F = \{A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H\}$
  - $G = \{A \rightarrow CD, E \rightarrow AH\}$
- Invece di verificare se $X \rightarrow Y \in F$ è anche in G^+ e viceversa, possiamo verificare se $Y \subseteq X_G^+$ e, viceversa, per ogni dipendenza funzionale
- Per esempio (verifichiamo G su X_F^+):
  - $A \rightarrow CD$: $A_F^+ = ACD$, quindi $CD \in A_F^+$
  - $E \rightarrow AH$: $E_F^+ = EADHC$, quindi $AH \in E_F^+$

Ridondanza

- Sia F un insieme di dipendenze funzionali
- Data $X \to Y \in F$, X contiene un **attributo estraneo** se e solo se $\big(F - \{X \to Y\}\big) \cup \big(X - \{A\} \to Y\big) \equiv F$, o, in altre parole, se e solo se $X - \{A\} \to Y \in F^+$.
- $X \to Y$ è una **dipendenza ridondante** se e solo se $\big(F - \{X \to Y\}\big) \equiv F$, o, in altre parole, se e solo se $X \to Y \in \big(F - \{X \to Y\}\big)^+$
- Le dipendenze che **non contengono attributi estranei** e la cui **parte destra è un unico attributo** sono dette **dipendenze elementari**

Esempio 6

- Sia $F = \{AB \to C, B \to A, C \to A\}$
  - L'unica dipendenza che può avere un attributo estraneo è $AB \to C$
    - $A^+ = \{A\}$ e $B^+ = \{B, A, C\}$, quindi A è un **attributo estraneo** in $AB \to C$
- Quindi $F \equiv G_1 = \{B \to C, B \to A, C \to A\}$
  - $\{B \to C, C \to A\}^+ = G$, quindi $B \to A$ è una **dipendenza ridondante**
- Quindi $F \equiv G_1 \equiv G_2 = \{B \to C, C \to A\}$
  - Se tentiamo di eliminare la dipendenza ridondante prima di eliminare l'attributo estraneo, non ci riusciamo, quindi **l'ordine** delle due attività **è importante**

Copertura minimale

- Sia F un insieme di dipendenze funzionali
- F è una **copertura minimale** se e solo se:
  - ogni parte destra di una dipendenza ha un unico attributo;
  - le dipendenze non contengono attributi estranei;
  - non esistono dipendenze ridondanti.
- In alcuni testi una copertura minimale è chiamata:
  - *insieme minimale*
  - *copertura canonica*
- Nell'esempio precedente, G_2 è una copertura minimale.

Calcolo della copertura minimale

Input: insieme di dipendenze funzionali F

Output: copertura minimale G di F

$G \leftarrow F$
for each $X \rightarrow Y \in G$ **do**
  $Z \leftarrow X$
  for each $A \in X$ **do**
    if $Y \in \left(Z - (A)\right)^{+}_{F}$ **then**
      $Z \leftarrow Z - \{A\}$
  $G \leftarrow \left(G - \{X \rightarrow Y\}\right) \cup \{Z \rightarrow Y\}$
for each $f \in G$ **do**
  if $f \in \left(G - \{f\}\right)^{+}$ **then**
    $G \leftarrow G - \{f\}$
return G

Calcoliamo gli attributi estranei delle dipendenze

Eliminiamo gli attributi estranei delle dipendenze

Eliminiamo le dipendenze ridondanti

Copertura minimale

- Il precedente algoritmo dimostra il seguente teorema.
- **Teorema**:
  - Per ogni insieme di dipendenze funzionali F esiste una copertura minimale
- Si noti che il teorema nulla dice sull'**unicità** della copertura minimale
- Infatti, per $F = \{AB \rightarrow C, A \rightarrow B, B \rightarrow A\}$,
  - $\{A \rightarrow C, A \rightarrow B, B \rightarrow A\}$ è una copertura minimale
  - $\{B \rightarrow C, A \rightarrow B, B \rightarrow A\}$ è una copertura minimale