

DIPENDENTE(Id, SettId, Nome, Cognome, DataN)  
 PROGETTO(Id, SettId, Titolo, DataN)  
 TEAM(Progid, DipenId, Ruolo)  
 SETTORE(Id, CapId, Nome)

- 1) Elenco (nome, cognome e data di nascita) dei dipendenti che hanno lavorato su progetti più vecchi di 3 anni (iniziati prima dell'1/9/2020) nel ruolo di "Designer" oppure di "Programmer"
- 2) Titolo del progetto o dei progetti più vecchi (meno recenti) di cui ogni dipendente è stato Project Leader (nome e cognome del Project Leader e titolo del progetto).

1a)  $P = \sigma_{DATA < 1/9/2020} (PROGETTO)$   $T = \pi_{DIPENID} (\sigma_{RUOLO = 'DESIGNER' \vee RUOLO = 'PROGRAMMER'} (TEAM))$  Query finale:  $Q = DIPENDENTE \bowtie_{ID=DIPENID} T$

1b)  $MAXSTATPL = \pi_{DIPENID} (TEAM) - \pi_{DIPENID} (\sigma_{RUOLO = 'PROJECTLEADER'} (TEAM))$   $ANPL = TEAM - (TEAM \bowtie_{ID=STATPL} MAXSTATPL)$

Query finale:  $Q = \pi_{NOME, COGNOME, TITOLO} ((ANPL \bowtie_{PROGID=110} PROGETTO) \bowtie_{DIPENID=110} DIPENDENTE)$

2) Dato il seguente schema R = ABCDEH sul quale è definito il seguente insieme di dipendenze funzionali:

F = {BCD → AH, BD → CH, A → D, C → E, CD → H, H → E, AH → B}

2a) trovare le chiavi dello schema motivando la risposta\*

2b) dire se lo schema è in 3NF motivando la risposta\*

2c) trovare una decomposizione dello schema\*, in modo tale che ogni sottoschema sia in 3NF, e che la decomposizione preservi F e abbia un join senza perdita.

(\* scrivere le definizioni formali degli elementi teorici a cui si fa riferimento e lo pseudo codice degli algoritmi utilizzati)

Prima di cominciare con i punti, mi è comodo trovare una copertura minimale di F. Inizio

minimizzando i determinati:  $F = \{BCD \rightarrow A, BCD \rightarrow H, BD \rightarrow C, BD \rightarrow H, A \rightarrow D, C \rightarrow E, CD \rightarrow H, H \rightarrow E, AH \rightarrow B\}$ , controllo ora i determinanti:

•  $BCD \rightarrow A$ ,  $BC^+ = BCE$   $BD^+ \Rightarrow A$ , posso sostituire. •  $BCD \rightarrow H$ , so che  $BD \rightarrow H \in F$ , quindi posso rimuoverla.

Nota che  $B^+ = B$  e  $D^+ = D$ , quindi  $BD \rightarrow C$  e  $BD \rightarrow H$  rimangono. •  $CD \rightarrow H$ , ho che  $C^+ = CE$ , quindi rimane.

•  $AH \rightarrow B$ , ho che  $A^+ = AD$  e  $H^+ = HE$ , quindi rimane. Ho  $F = \{BD \rightarrow A, BD \rightarrow C, BD \rightarrow H, A \rightarrow D, C \rightarrow E, CD \rightarrow H, H \rightarrow E, AH \rightarrow B\}$ .

Controllo le ridondanze. •  $BD \rightarrow A$ , ho  $BD^+_{F \setminus BD \rightarrow A} = BCDEH$ , rimane. •  $BD \rightarrow C$ , ho  $BD^+_{F \setminus BD \rightarrow C} = ABDEH$ , rimane.

•  $BD \rightarrow H$ , ho  $BD^+_{F \setminus BD \rightarrow H} = R$ , è ridondante. OSS: evito di controllare le dip. che hanno come determinato un attributo

che appare in una sola dip. •  $C \rightarrow E$ , ho  $C^+_{F \setminus C \rightarrow E} = C$  •  $H \rightarrow E$ , ho  $H^+_{F \setminus H \rightarrow E} = H$ , rimane. La copertura minimale è:

$F = \{BD \rightarrow A, BD \rightarrow C, A \rightarrow D, C \rightarrow E, CD \rightarrow H, H \rightarrow E, AH \rightarrow B\}$ .

2a) Controllo le X l.c.  $X_F^+ = RA \nexists X^+ \times I \mid X_F^+ = R$ . Nota che E non compare come determinante  $\Rightarrow$  non è nella chiave.

Nota che nessun attributo singolo è chiave, le coppie di 2 da ABCDH sono 10, le controllo manualmente:

$AB^+ = R$   $AH^+ = R$   $BH^+ = BHE$   $DH^+ = DEH$   
 $AC^+ = R$   $BC^+ = BCE$   $CD^+ = CDEH$   
 $AD^+ = AD$   $BD^+ = R$   $CH^+ = CEH$

Le 4 chiavi son {AB, AC, AH, BD}

2b) Non è in 3NF, dato che C non è superchiave ed E non è primo, ma  $C \rightarrow E \in F$ .

2c) Applico l'algoritmo, la dec. è  $\rho = \{BDA, BDC, AD, CE, CDH, HE, AHB\}$ .  $\leftarrow$  ha un lossless Join.

3) E' dato un file di 145.950 record. Ogni record occupa 255 byte. Un blocco contiene 2048 byte. Un puntatore a blocco occupa 5 byte. Si utilizza una organizzazione hash con record distribuiti uniformemente tra 250 bucket.

3a) Calcolare l'occupazione in blocchi della bucket directory e dei bucket

3b) Calcolare il costo medio di una ricerca considerando chiavi di hash univoche

3c) Quanti bucket occorrerebbero per avere un costo medio di ricerca minore o uguale a 10 accessi?

$pointer \times block = L \frac{2048}{5} \approx 409 \Rightarrow$  basta 1 blocco per la bucketDir.

$record \times block = L \frac{2048}{255} \approx 8$   $record \times bucket = \lceil \frac{145950}{250} \rceil = 584$   $blocchi \times bucket = \lceil \frac{584}{8} \rceil = 73$   $Avg Acc = \lceil \frac{73}{2} \rceil = 37$