



Basi di Dati

Esercitazione in Lab n.4

Prof. Mauro Conti

Dipartimento di Matematica - Università degli studi di Padova

conti@math.unipd.it - <http://www.math.unipd.it/~conti>

Giuseppe Cascavilla, PhD student

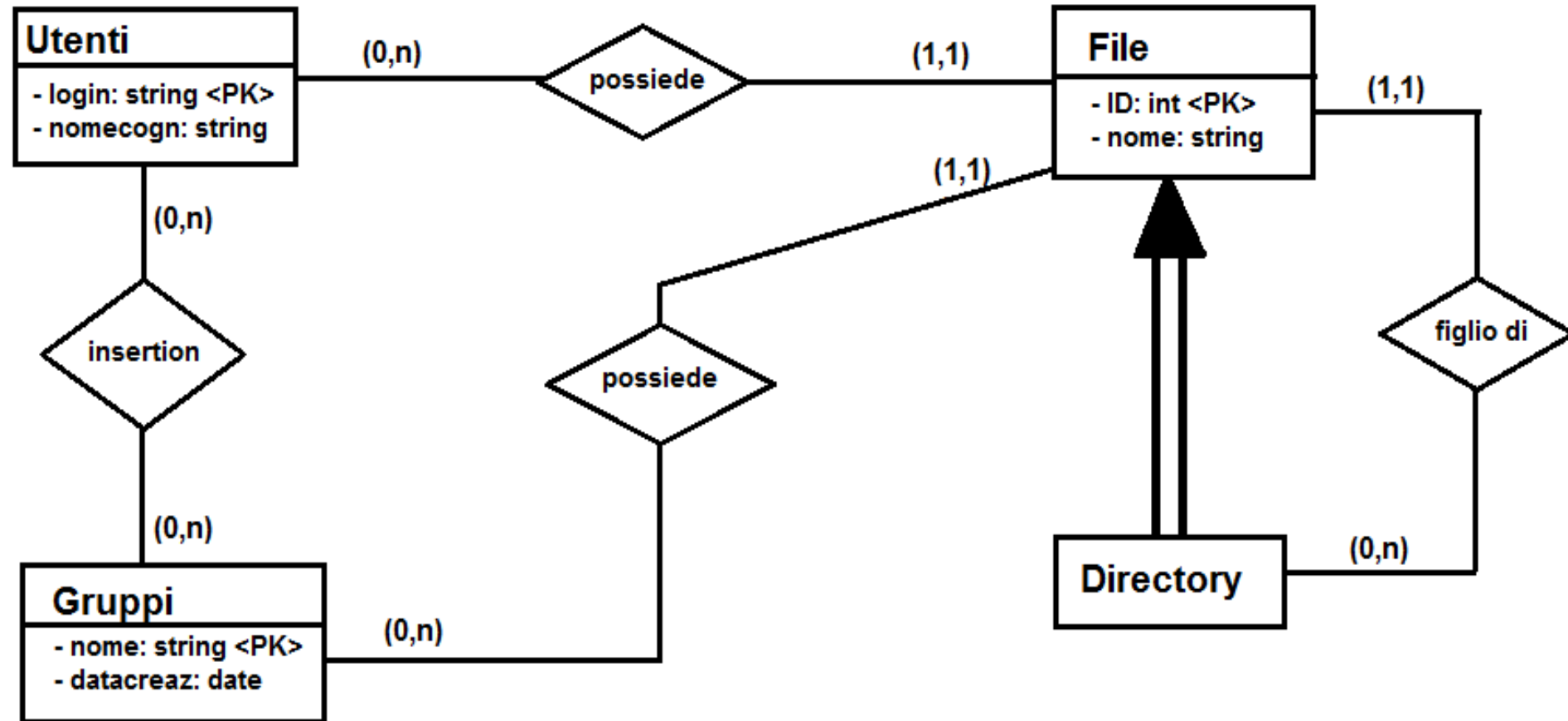
sites.google.com/site/gcascavilla



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- Da diagramma E-R alla creazione delle tabelle
- Popolamento delle tabelle
- Select - From - Where
- Le viste

Si vuole creare una base che memorizzi informazioni relative ad un file system.
Qui di seguito il diagramma E-R



I file sono organizzati in una struttura ad albero. La radice è l'unico file che ha come padre sé stesso. Le directory sono dei particolari file.

Si fanno le seguenti assunzioni:

- La cancellazione di un utente determina la cancellazione di tutti i file dei quali é proprietario.
- I gruppi non vengono mai cancellati.
- La cancellazione di una directory comporta la cancellazione dei file in essa contenuti.
- Il 'Nome' dei file e il 'NomeCogn' (nome e cognome) degli utenti occupano al massimo 20 caratteri. 'Nome' dei gruppi e 'Login' degli utenti occupano al massimo 8 caratteri.

Si realizzi lo schema in SQL, tenendo conto, nella definizione delle chiavi esterne, delle assunzioni elencate dopo lo schema concettuale.

Nota: Durante la creazione e la popolazione della base di dati vi può essere utile (ad es. nel caso di riferimenti ciclici) inibire il controllo sulle chiavi esterne rispettivamente con (**tramite terminale MySQL**):

per disattivare:

```
SET FOREIGN_KEY_CHECKS=0;
```

per attivare:

```
SET FOREIGN_KEY_CHECKS=1;
```

Si popoli la base di dati di modo che ricalchi la situazione descritta dalla seguente tabella.

ID	Nome	Utente	Gruppo
1	Radice/	root	admin
11	Var/	root	admin
111	Mail/	mail	mail
1111	rossi.mbx	rossi	mail
1112	verdi.mbx	verdi	mail
112	SubM/	root	admin
12	tmp/	root	admin
121	tmp0.txt	rossi	user
122	tmp1.txt	verdi	user

123	SubT/	root	admin
13	home/	root	admin
131	rossi/	rosssi	user
1311	slide.txt	rossi	user
1312	progetto.pdf	rossi	user
132	verdi/	verdi	user
1321	eserc1.sql	rossi	user

- Il primo campo indica l'Id del file. Il popolamento della BD può risultare semplificato se si nota che l'Id della directory padre si può ottenere da quello del figlio togliendo l'ultima cifra (tranne che per la radice).
- Il secondo campo rappresenta il nome. L'eventuale '/' finale non è parte del nome, ma invece, se presente indica che il file è una directory. La struttura rispecchia la struttura dell'albero delle directory. Ad esempio La directory Var contiene la Directory Mail che a sua volta contiene i file rossi.mbx e verdi.mbx
- Il terzo campo è l'utente che possiede il file.
- Il quarto campo è il gruppo che possiede il file.

I gruppi presenti, con la corrispondente data di creazione sono:

Gruppo	Data
user	2007-01-02
mail	2006-01-01
admin	2006-02-04
sys	2006-12-25
none	2007-01-01

Gli utenti presenti, con i relativi gruppi di appartenenza sono:

Login	Nome	Gruppi
root	NULL	user, mail, admin, sys, mail
verdi	Gino Verdi	user, mail
rossi	Anna Rossi	user, mail
mail	NULL	mail
nobody	NULL	

Ricordate che potete caricare le tabelle con il comando

INSERT INTO tabella VALUES

oppure

LOAD DATA LOCAL INFILE 'file.txt' INTO TABLE tabella

dove 'file.txt' è un file preventivamente creato che contiene le ennuple da inserire in 'tabella' con campi separati da TAB (TAB è il separatore di default, ma si possono specificare anche altri separatori).

Vi potrebbero essere utili i seguenti file (nei quali i campi sono separati da TAB ...)

Directory.txt GruppiUtenti.txt Utenti.txt File.txt Gruppi.txt

disponibili nella vostra home

(Nota: Per poter usare il comando 'LOAD DATA ...' occorre aver avviato il client con il flag '--local-infile=1')

Fornire le interrogazioni SQL per rispondere alle seguenti domande. Ogni query è accompagnata dal risultato che si otterrebbe rispetto alla popolazione indicata sopra.

1. Trovare il nome e Id della directory radice

Nome	Id
Radice	1

2. Elencare le directory vuote (nome e Id)

Nome	Id
SubM	112
SubT	123

3. Creare una vista FileDir che restituisce la tabella dei file che sono Directory. Selezionare tutti gli elementi della vista.

Id	Nome	Utente	Gruppo	Padre
1	Radice	root	admin	1
11	Var	root	admin	1
12	tmp	root	admin	1
13	home	root	admin	1
111	Mail	mail	mail	11
112	SubM	root	admin	11
123	SubT	root	admin	12
131	rossi	rossi	user	13
132	verdi	verdi	user	13

4. Utilizzare la vista FileDir per risolvere la query 2

Nome	Id
SubM	112
SubT	123

5. Creare la vista FileOrd che restituisce la tabella dei file ordinari (non directory).
Selezionare tutti gli elementi della vista.

Id	Nome	Utente	Gruppo	Padre
121	tmp0.txt	rossi	user	12
122	tmp1.txt	verdi	user	12
1111	rossi.mbx	rossi	mail	111
1112	verdi.mbx	verdi	mail	111
1311	slide.txt	rossi	user	131
1312	progetto.pdf	rossi	user	131
1321	eserc1.sql	rossi	user	132

6. Elencare gli utenti (login) ed il numero di file posseduti

Login	NumFile
mail	1
nobody	0
root	6
rossi	6
verdi	3

7. Elencare i gruppi i cui utenti posseggono, complessivamente, il massimo numero di file ordinari

Nome	COUNT(*)
mail	7
user	7

8. Elencare gli utenti che non appartengono a nessun gruppo o a tutti i gruppi

Login	NomeCognome
nobody	NULL

9. Aggiungere l'utente root al gruppo 'none' e riprovare la query precedente

Login	NomeCognome
nobody	NULL
root	NULL

10. Cancellare le directory vuote e mostrare le directory rimanenti

Id	Nome	Utente	Gruppo	Padre
1	Radice	root	admin	1
11	Var	root	admin	1
12	tmp	root	admin	1
13	home	root	admin	1
111	Mail	mail	mail	11
131	rossi	rossi	user	13
132	verdi	verdi	user	13

11. Cancellare l'utente 'rossi' e verificare che siano stati cancellati i suoi file mostrando la tabella 'File'

Id	Nome	Utente	Gruppo	Padre
1	Radice	root	admin	1
11	Var	root	admin	1
12	tmp	root	admin	1
13	home	root	admin	1
111	Mail	mail	mail	11
112	SubM	root	admin	11
122	tmp1.txt	verdi	user	12
123	SubT	root	admin	12
132	verdi	verdi	user	13
1112	verdi.mbx	verdi	mail	111