
Progetto di un database MySQL

Emanuele Zangarini

Diario delle Revisioni

Revisione 1.0	Febbraio 2005	EZ
	Prima stesura.	
Revisione 1.1	Febbraio 2005	EZ

Formattazione fronte-retro. Aggiunta nota array PHP. Immagine schema in SVG.

Sommario

1. Il database <code>theworks</code>	1
1.1. Struttura del database	1
1.2. Il DBMS MySQL	2
1.3. L'archivio bibliografico	2
1.4. Esportazione in XML/DTD del database: lo script PHP	3
1.5. Il foglio di stile XSL	3
A. MANIFEST: file del progetto	3

Presentazione di un progetto di database MySQL per il corso di Basi di Dati.

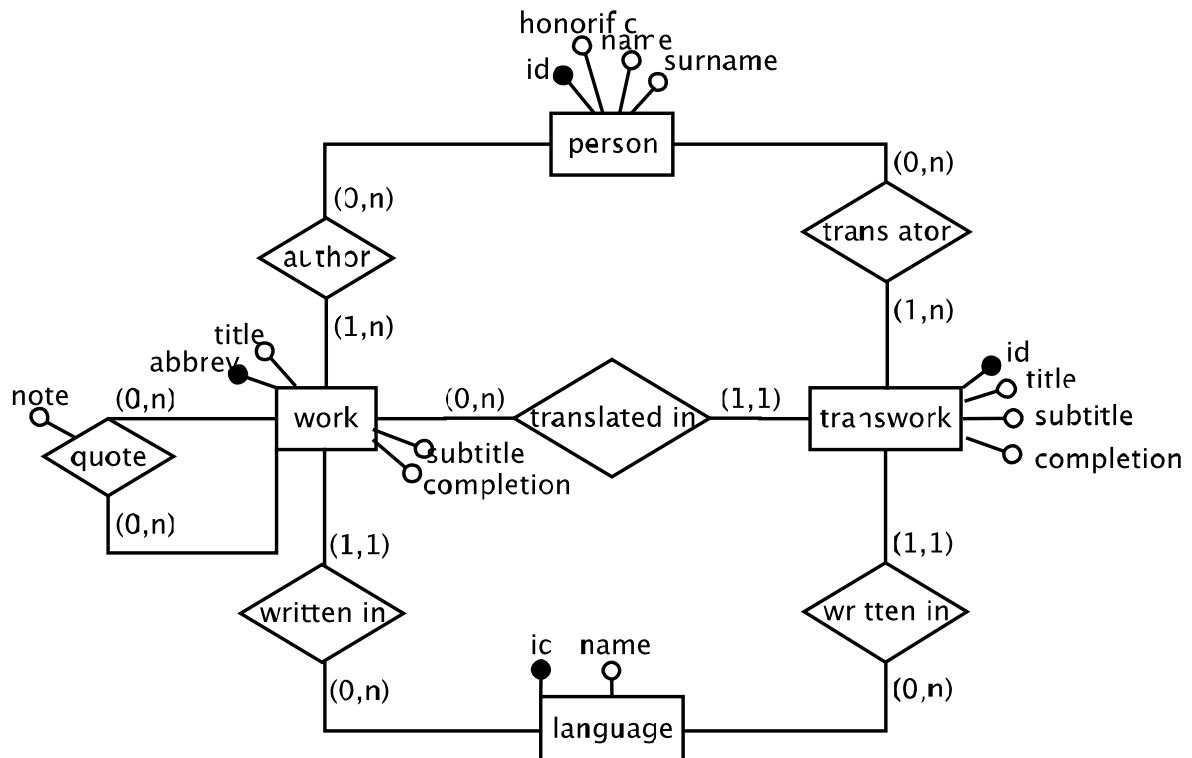


L'ultima versione di questo documento è reperibile a questo indirizzo [http://echowebpace.altervista.org/docs/essays/mysql_theworks.pdf].

1. Il database `theworks`

1.1. Struttura del database

Il database si struttura in quattro entità (`person`, `language`, `work`, `transwork`) e 7 relazioni di cui 3 (`author`, `translator`, `quote`) vengono convertite in altrettante tabelle, seguendo il principio per cui le relazioni M:N sono le uniche a necessitare obbligatoriamente la costituzione di entità comprendenti le chiavi primarie delle entità che mettono in relazione e gli attributi della relazione stessa.



Schema E/R del database 'theworks', disegnato con Dia [http://www.gnome.org/projects/dia/] 0.94 di Alexander Larsson.

1.2. II DBMS MySQL

Per questo progetto è stata utilizzata la versione 4.0.20 di MySQL su una piattaforma GNU/Linux.

Come storage engine la scelta è caduta su InnoDB, data l'incapacità della versione installata di garantire l'integrità referenziale per le tradizionali tabelle MyISAM (le foreign key erano difatti "parsed but ignored" come recita il manuale di MySQL al par. 13.8.2).

Sempre per le limitazioni della versione utilizzata, i campi chiave esterna vanno esplicitamente indicizzati tramite **INDEX (campo)**, in modo da velocizzare il recupero dei dati.



Nelle future versioni pianificate di MySQL verrà implementata l'integrità referenziale anche per MyISAM e l'indicizzazione dei campi chiave esterna avverrà in modo automatico.

1.3. L'archivio bibliografico

In vista della tesi di laurea, la necessità è quella di memorizzare agilmente le informazioni su scritti di diversa natura, ponendo particolare attenzione alla distinzione fra testi in lingua originale e testi tradotti.

Per ridurre la complessità del modello, si ignorano volutamente alcune entità come le case editrici ed i luoghi di edizione.

Altri limiti sono la possibilità per le sole opere in lingua originale di citare altre opere in lingua originale e non le sue traduzioni. La citazione di un lavoro tradotto in una data opera dovrà essere dunque sempre rappresentata dalla citazione dell'opera in lingua originale.

1.4. Esportazione in XML/DTD del database: lo script PHP

Dato il database, riempito di dati, si rendeva necessario da consegna esportare gli stessi nel formato XML. Ho scelto di utilizzare a questo scopo il linguaggio di scripting server-side PHP (ver. 4.3.7), data la sua particolare facilità d'uso nella connessione ai database.

Lo script riproduce le singole tuple di tutte le tabelle del database generate dall'esecuzione, ripetuta per ogni tabella, della query SQL **SELECT * FROM \$tablename[\$j];** dove l'argomento della query è un riferimento all'array PHP che contiene i nomi delle tabelle.

Parallelamente l'array `$attr_field` contiene i nomi dei campi che si vogliono utilizzare come attributi del tag contenitore della tabella e non come sottotag. Si assume che questi siano solo campi chiave. Vengono considerate chiavi primarie quelle comprese fra l'elemento 1 e il valore indicato nel primo elemento dell'array, estremi inclusi.

Se non vi è alcun campo chiave nella rispettiva tabella, `$attr_field` dovrà contenere all'indice corrispondente un array vuoto.



Gli array vengono generati automaticamente effettuando una query **DESCRIBE** sulla struttura di ogni singola tabella e controllando il valore del campo `Key`. **PRI** contrassegna una chiave primaria e **MUL** una chiave esterna.

Lo script genera pure un file descrittore DTD per ogni file XML. I tipi relativi agli attributi sono rispettivamente **ID** per le chiavi primarie e **IDREF** per le chiavi esterne. Tutte le chiavi sono contrassegnate come **#REQUIRED**.

La codifica di output per i file è Unicode (UTF-8).

I file vengono memorizzati in una directory con lo stesso nome del database all'interno della directory `mysql2xml`, creata all'interno della directory di esecuzione dello script.

1.5. Il foglio di stile XSL

L'interpretazione dei file XML precedentemente esportati è affidata a poche ed essenziali istruzioni XSL. Queste creano per ogni tag corrispondente ad un record una TABLE in HTML con il contenuto dei campi e degli attributi di cui sono composti ed i relativi nomi.

Gli attributi chiave vengono evidenziati graficamente.



Per discernere fra chiavi primarie ed esterne sarebbe necessario controllare la DTD del documento tramite codice XSL od aggiungere allo scopo un attributo addizionale ad ogni record esportato in XML.

A. MANIFEST: file del progetto

```
.
|-- ER
|   |-- theworks.dia
|   `-- theworks.svg
|-- MANIFEST
|-- SQL
|   |-- create.sql
|   |-- insert.sql
|   `-- select.sql
|-- XML
|   |-- mysql2xml
|   |   `-- theworks
|   |       |-- author.dtd
|   |       |-- author.xml
|   |       |-- language.dtd
|   |       |-- language.xml
|   |       |-- person.dtd
|   |       |-- person.xml
|   |       |-- quote.dtd
|   |       |-- quote.xml
|   |       |-- translator.dtd
|   |       |-- translator.xml
|   |       |-- transwork.dtd
|   |       |-- transwork.xml
|   |       |-- work.dtd
|   |       `-- work.xml
|   |-- mysql2xml.php
|   `-- theworks.xsl
`-- doc
    |-- MySQLtheworks.pdf
    `-- MySQLtheworks.xml
```

6 directories, 24 files