

UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea in INGEGNERIA INFORMATICA

Realizzazione di una basi di dati per la gestione di bibliografie scientifiche digitali

Laureando:

Francesco Zausa

Relatore:

Prof. Maristella Agosti

Anno Accademico 2010/2011

Indice

1	Introduzione	1
2	Motivazione e background	3
2.1	Bib \TeX	5
2.2	L \TeX	8
2.3	Realtà di interesse	10
3	Raccolta ed analisi dei requisiti	13
3.1	Figure coinvolte	14
3.2	Glossario dei termini	15
4	Progettazione Concettuale	19
4.1	Schema ER	19
4.2	Tabella Entità	26
4.3	Tabella Associazioni	29
5	Progettazione Logica	33
5.1	Ristrutturazione dello schema ER	33
5.1.1	Eliminazione delle generalizzazioni	33
5.1.2	Eliminazione degli attributi multivalore	34
5.1.3	Scelta degli identificatori	34
5.2	Schema ER Ristrutturato	35
5.3	Schema Logico	36
5.3.1	Qualità dello schema logico	47
6	Progettazione Fisica	49
6.1	Scelte Progettuali	49
6.1.1	pgAdmin 1.12.3	50
6.1.2	PostgreSQL 8.4.8	53
6.2	Codice SQL e Licenze d'uso	54
6.2.1	Licenza GNU General Public License	55
6.2.2	Licenza GNU Lesser General Public License	56

6.2.3	Scelta della licenza	56
7	Conclusioni	57
A	Contratti Licenze d'uso	59
A.1	GNU GPL	59
A.2	GNU LGPL	70
B	Codice SQL	73
	Bibliografia	87
	Sitografia	90

Elenco delle figure

1.1	Bibliographia gallica di Louis Jacob	1
1.2	Bibliografia moderna	2
2.1	Bibliografia Politica di Gabriel Naudé (1633)	3
2.2	Versione JabRef.	7
2.3	Schermata d'ambiente JabRef.	8
2.4	Esempio della compliazione del codice 2.2 a pagina 8 e 2.1 a pagina 6	9
2.5	Loghi di alcuni strumenti di ricerca per informazioni bibliografiche	10
3.1	Esempio di International Standard Book Number (ISBN).	14
4.1	Schema ER Dott. Matteo Marittan	21
4.2	Schema ER Dott. Germano Rocco	22
4.3	Schema ER	23
4.4	Generalizzaione tra le entità Author ed Editor	24
4.5	Relazioni che coinvolgono l'attributo cross_ref	25
5.1	Generalizzazione totale tra l'entità padre Thesis e le entità figlie Masterthesis e Phdthesis	34
5.2	Possibile schema logico dell'attributo multivalore local file name	34
5.3	Schema ER Ristrutturato	35
5.4	Schema logico Journal, Article, By_Article, LFN_Article, Author.	37
5.5	Schema logico di Incollections, LFN_Incollections, Published_Incollections, Publisher, By_Incollections, Edited_Incollections.	38
5.6	Schema logico di Book, LFN_Book, Published_Book, By_Book.	39
5.7	Schema logico di Inproceedings, LFN_Inproceedings, Cross_Ref_Proceedings, Edi- ted_Inproceedings, By_Inproceedings, Published_Inproceedings.	40
5.8	Schema logico di Proceedings, LFN_Proceedings, Edited_Proceedings, Published_Pro- ceedings.	41
5.9	Schema logico Misc, LFN_Misc, By_Misc.	42
5.10	Schema logico di Manual, LFN_Manual, By_Manual, Published_Manual.	43
5.11	Schema di Techreport, LFN_Techreport, By_Techreport.	44
5.12	Schema logico di Unpublished, LFN_Unpublished, By_Unpublished.	45

5.13	Schema logico di Thesis, By_Thesis, Supervisor.	46
6.1	Versione pgAdmin	50
6.2	Schermata di ambiente pgAdmin	51
6.3	Schermata tabella Article	51
6.4	Schermata strumento per le interrogazioni Structured Query Language (SQL) del database	52
6.5	Logo Sviluppatore PostgreSQL Global Development Group	53

Elenco delle tabelle

3.1	Glossario dei termini.	15
4.1	Tabella Entità.	26
4.2	Tabella Associazioni	29
6.1	Limiti PostgreSQL	55

Elenco dei listati codice

2.1	Esempio di un file Bib \TeX : bibliografia.bib	6
2.2	Esempio di codice L \TeX	8
B.1	Creazione database e domini.	73
B.2	Creazione tabella Author e Journal.	73
B.3	Creazione tabella Article, By_Article, LFN_Article e Publisher.	74
B.4	Creazione tabella Book, Incollections, LFN_Incollections e Published_Incollection. . .	75
B.5	Creazione tabella By_Incollections, Edited_Incollections, LFN_Book e Published_Book. .	76
B.6	Creazione tabella By_Book, Inproceedings, Proceedings e LFN_Inproceedings.	77
B.7	Creazione tabella Cross_Ref_Proceedings, Edited_Inproceedings, By_Inproceedings, Published_Inproceedings e LFN_Proceedings.	78
B.8	Creazione tabella Edited_Proceedings, Published_Proceedings, Misc, LFN_Misc e By_Misc.	79
B.9	Creazione tabella Manual, LFN_Manual, By_Manual e Published_Manual.	80
B.10	Creazione tabella Techreport, LFN_Techreport, By_Techreport e Unpublished.	81
B.11	Creazione tabella LFN_Unpublished, By_Unpublished, Thesis e LFN_Thesis.	82
B.12	Creazione tabella By_Thesis e By_Supervisor.	83

Capitolo 1

Introduzione

La bibliografia ha da sempre rappresentato un ruolo importante nella stesura di una pubblicazione scientifica, in quanto è in grado di dare solide basi agli argomenti trattati dagli autori e utili indicazioni per ulteriori approfondimenti. Con l'invenzione della stampa ci fu un incremento della produzione di testi, antichi e nuovi, determinando un'esigenza di informazione alla quale erano interessati varie categorie di lettori, a cominciare dagli studiosi.

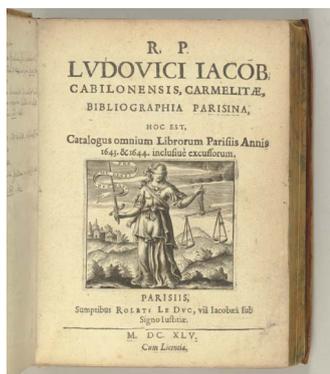


Figura 1.1: Bibliographia gallica di Louis Jacob

La rivoluzione digitale rappresenta una straordinaria opportunità di informazione, condivisione della conoscenza e crescita culturale. Con l'avvio delle Biblioteche Digitali -- *Digital Library (DL)* -- viene incrementata la possibilità di accedere ad un insieme di risorse informative.

Fin dalla storia si ha quindi la necessità di gestire l'informazione bibliografica, richiedendo strumenti di informazione adeguati alle varie esigenze del pubblico. Con la naturale evoluzione della bibliografia storica tramite l'utilizzo delle moderne tecnologie informatiche anche la bibliografia scientifica diventa digitale.

Obiettivo di questa tesi è progettare ed implementare un database per la gestione di bibliografie digitali per il gruppo di ricerca Information Management System (IMS)¹ dell'Università degli Studi di Padova². IMS è un gruppo di ricercatori che scrivono articoli, papers, documentazione scientifica, report, ... Se ne deduce quindi che la gestione delle bibliografie e dei riferimenti bibliografici assume un

¹<http://ims.dei.unipd.it/> ultima visita settembre 2011

²<http://www.unipd.it/> ultima visita settembre 2011

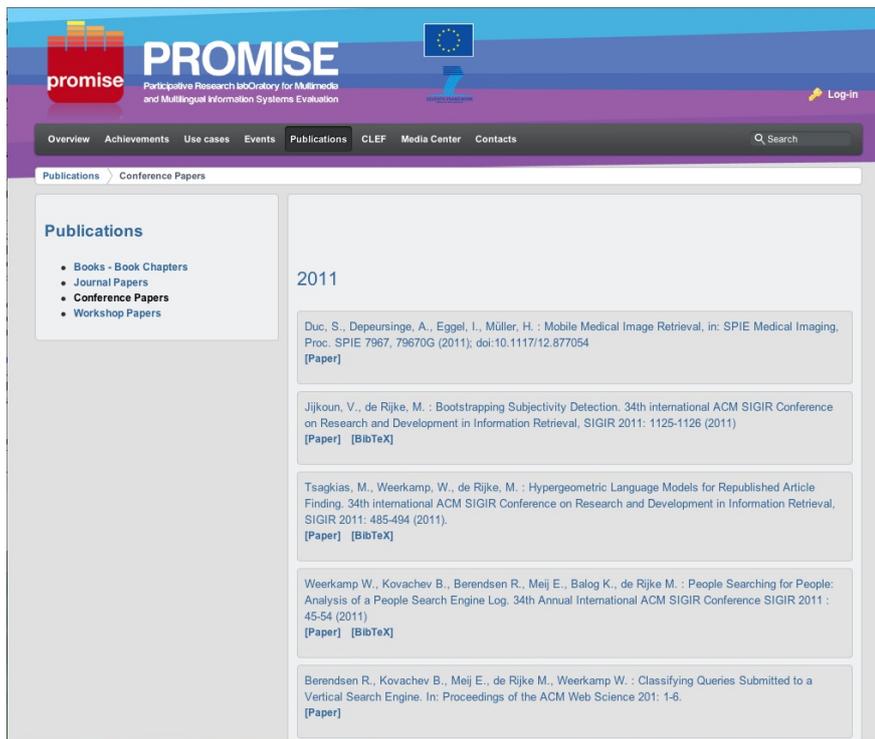


Figura 1.2: Bibliografia moderna

ruolo importante per il gruppo.

Gli strumenti utilizzati dal gruppo IMS sono soprattutto Bib $\text{T}_{\text{E}}\text{X}^3$ [13] e $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}^4$ [24], ma anche word processor, applicativi per la creazione di presentazione, pubblicazione di pagine web. Dalla difficoltà della condivisione e del riutilizzo dei riferimenti bibliografici all'interno del gruppo è nata l'esigenza di una base di dati in grado di gestire i dati.

Lo sviluppo di tale progetto si è articolato in varie fasi.

Innanzitutto è stata esaminata la realtà di interesse e le esigenze del gruppo IMS. Si sono analizzati gli strumenti ad oggi utilizzati per gestire i riferimenti bibliografici, in particolar modo Bib $\text{T}_{\text{E}}\text{X}$ e $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$.

Una volta raccolti i requisiti per la realizzazione del database, è stata stesa una bozza del progetto ed una descrizione completa dei dati coinvolti.

La fase successiva è stata la *progettazione concettuale*, nella quale vengono rappresentate le specifiche informali della realtà di interesse, indipendentemente dai sistemi di gestione di basi dati. Il prodotto di tale fase è lo schema Entity–Relationship (ER).

Quanto prodotto nella progettazione concettuale, è stato elaborato al fine di ottenere uno schema logico. Tale fase costituisce la vera base per l'effettiva implementazione del database. Lo schema logico è stato a sua volta tradotto nel linguaggio Structured Query Language (SQL).

In fine, attraverso l'utilizzo del DataBase Management System (DBMS) PostgreSQL⁵ [29], è stato realizzato il database.

³<http://www.bibtex.org/> ultima visita settembre 2011

⁴<http://www.latex-project.org/> ultima visita settembre 2011

⁵<http://www.postgresql.org> ultima visita settembre 2011

Motivazione e background

La gestione di bibliografie digitali è un argomento di estrema attualità e importanza, essendo le Biblioteche Digitali -- *Digital Library (DL)* -- strumenti indispensabili nella vita quotidiana di molte persone in tutto il mondo, consentendo la condivisione di contenuti ed esperienza. Il concetto elementare e più diffuso di *bibliografia* è quello di “elenco” di testi, preparati a scopo di informazione. Con l’avvento delle tipografia, l’incremento della produzione libraria determinò un’esigenza di informazione a diversi livelli. La circolazione del libro comportava la diffusione incrementata di testi, vecchi e nuovi, cui erano interessati diverse categorie di lettori. Il termine bibliografia venne utilizzato per la prima volta nel 1633 da Gabriel Naudé col valore di descrizione di libri nella sua *Bibliografia politica*, ma si attestò stabilmente solo nel corso dell’Ottocento [35].

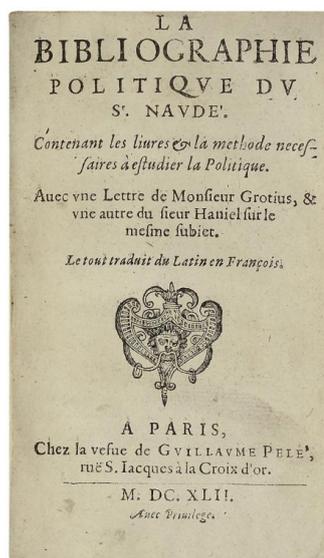


Figura 2.1: Bibliografia Politica di Gabriel Naudé (1633)

Al concetto di bibliografia si collega il concetto di *riferimento bibliografico*. “Il riferimento bibliografico è costituito da tutte le informazioni che, ordinate secondo certe regole, permettono l’identificazione e il reperimento di un’opera scritta o di parte di essa. Il riferimento bibliografico specifica anche con la massima precisione una particolare edizione scritta, in modo che tutti possano ritrovarla,

controllarne i contenuti, valutare la correttezza e l'attendibilità del lavoro svolto da chi ha usato quell'opera" [17].

L'organizzazione dell'informazione bibliografica continua tutt'oggi a seguire uno schema strutturale basato su tre fasi:

1. raccolta dei dati;
2. memorizzazione;
3. rapido recupero delle informazioni;

La necessità di reperire le informazioni per orientarsi a scegliere tra una vastità di libri, prodotti con crescita esponenziale, chiede strumenti di informazione adeguati alle diverse esigenze del pubblico. La continuità della tradizione bibliografica è individuabile nell'attuale organizzazione dei sistemi informativi.

Nei vari colloqui sostenuti con lo staff del gruppo di ricerca *Information Management System (IMS)*, si è potuto notare l'importanza della pubblicazione scientifica. Essa rappresenta la principale forma di comunicazione ufficiale della comunità scientifica, tramite la quale i singoli ricercatori o i gruppi di ricerca rendono pubblici i metodi ed i risultati dei propri lavori scientifici [32].

I riferimenti bibliografici vengono scritti per le seguenti motivazioni: [2]

1. Per riconoscere l'apporto tratto dalle fonti al lavoro che si sta sviluppando .
2. Per dimostrare il "corpus" di conoscenze su cui si basa il lavoro svolto.
3. Per dare un adeguato credito alle pubblicazioni e idee di altri, dimostrando di non averle plagiate.
4. Per rendere agevole per chi legge l'identificazione delle fonti.

Normalmente nella stesura di un testo il processo con cui si fa riferimento alle pubblicazioni consiste in due parti: citare le fonti nel testo principale del documento e scrivere la *bibliografia* corrispondente, in forma estesa, alla fine del documento.

Quindi tutto ciò che può costituire un riferimento utile per il proprio lavoro deve essere letto, catalogato ed infine, se utilizzato, deve essere citato come fonte di riferimento bibliografico nelle proprie pubblicazioni.

Si deduce quindi che una delle attività principali del lavoro di un gruppo di ricerca consista anche nel recuperare e gestire tutti i riferimenti bibliografici utili per il proprio lavoro: libri, articoli, manuali tecnici. . . A volte nasce anche l'esigenza di poter condividere tali risorse bibliografiche con altre persone. Ad esempio i membri di un gruppo di ricerca possono aver l'esigenza di comunicare ai propri colleghi quali riferimenti utilizzare per lo sviluppo di un determinato progetto comune a cui essi stanno lavorando. In particolar modo, le comunità scientifiche possono trarre un vantaggio disponendo un database comune. Esistono già diversi sistemi in grado di gestire questi riferimenti. Tra questi all'interno delle comunità scientifiche ha trovato impiego l'uso di Bib_TE_X.

2.1 BibT_EX

BibT_EX¹ è uno strumento utilizzato per la formattazione di liste di riferimenti bibliografici. BibT_EX semplifica la citazione di riferimenti bibliografici in maniera consistente, separando l'informazione bibliografica dalla modalità di presentazione. BibT_EX utilizza un formato di file di tipo testuale, senza informazioni sullo stile di presentazione, contenente un elenco di voci bibliografiche che spazia dai libri, agli articoli di riviste, a tesi, etc. Gli elementi della bibliografia vengono divisi per tipo. I seguenti tipi sono compresi virtualmente in ogni stile di BibT_EX:

1. **article**: un articolo da un giornale o da una rivista.
 - Campi richiesti: author, title, journal, year
 - Campi opzionali: volume, number, pages, month, note, key
2. **book**: un libro con un esplicito editore.
 - Campi richiesti: author/editor, title, publisher, year
 - Campi opzionali: volume, series, address, edition, month, note, key, pages
3. **booklet**: un lavoro che è stampato e rilegato, ma senza un editore o un'istituzione che lo sponsorizzi.
 - Campi richiesti: title
 - Campi opzionali: author, howpublished, address, month, year, note, key
4. **Inbook**: la parte di un libro, che può essere un capitolo, (o una sezione o quant'altro) o un breve intervallo di pagine.
 - Campi richiesti: author/editor, title, chapter/pages, publisher, year
 - Campi opzionali: volume, series, address, edition, month, note, key
5. **Incollection**: la parte di un libro avente il proprio titolo.
 - Campi richiesti: author, title, booktitle, year
 - Campi opzionali: editor, pages, organization, publisher, address, month, note, key
6. **Inproceedings**: un articolo negli atti di una conferenza.
 - Campi richiesti: author, title, booktitle, year
 - Campi opzionali: editor, pages, organization, publisher, address, month, note, key
7. **Conference**: lo stesso che in inproceedings, incluso per compatibilità con Scribe.
 - Campi richiesti: author, title, booktitle, year
 - Campi opzionali: editor, pages, organization, publisher, address, month, note, key
8. **Manual**: documentazione tecnica.
 - Campi richiesti: title

¹<http://www.bibtex.org/> ultima visita settembre 2011

- Campi opzionali: author, organization, address, edition, month, year, note, key
9. **Masterthesis**: una tesi di laurea.
- Campi richiesti: author, title, school, year
 - Campi opzionali: address, month, note, key
10. **Phdthesis**: una tesi di dottorato.
- Campi richiesti: author, title, school, year
 - Campi opzionali: address, month, note, key
11. **Misc**: da usare quando l'elemento della bibliografia non corrisponde a nessun altro tipo.
- Campi richiesti: none
 - Campi opzionali: author, title, howpublished, month, year, note, key
12. **Proceedings**: gli atti di una conferenza.
- Campi richiesti: title, year
 - Campi opzionali: editor, publisher, organization, address, month, note, key
13. **Techreport**: un resoconto pubblicato da una scuola o da un'altra istituzione, solitamente all'interno di una serie.
- Campi richiesti: author, title, institution, year
 - Campi opzionali: type, number, address, month, note, key
14. **Unpublished**: un documento con un autore ed un titolo, ma non formalmente pubblicato.
- Campi richiesti: author, title, note
 - Campi opzionali: month, year, key

Di solito questo database bibliografico testuale è contenuto in un file con il suffisso *.bib*. Il file con estensione *.bib* contiene al proprio interno un certo numero di record come i seguenti:

```
@manual{postgresql,
  Author = {The PostgreSQL Global Development Group},
  Date-Added = {2011-08-12 15:15:38 +0000},
  Date-Modified = {2011-08-19 13:22:29 +0000},
  Title = {PostgreSQL 8.4.6 Documentation},
  Year = {2009}}

@book{pantier2011,
  Author = {Lorenzo Pantieri, Tommaso Gordini},
  Date-Added = {2011-08-10 17:57:33 +0000},
  Date-Modified = {2011-08-18 17:18:00 +0000},
  Month = {Aprile},
  Publisher = {Gruppo utilizzatori TeX e LaTeX},
  Title = {L'arte di scrivere con Latex},
```

```
Year = {2011}}

@mastersthesis{RGB,
  Author = {Matteo Maritan},
  Date-Added = {2011-07-28 16:55:38 +0200},
  Date-Modified = {2011-07-28 16:57:45 +0200},
  Read = {0},
  School = {Universit\`a degli Studi di Padova},
  Title = {RGB (Research Group Bibliography) un'applicazione WEB in PHP per la
          gestione della bibliografia di un gruppo di ricerca},
  Year = {2006/2007}}
```

Codice 2.1: Esempio di un file BibT_EX: bibliografia.bib

Si può notare che all'interno del codice 2.1 a fronte, ogni record corrisponde ad una pubblicazione, il cui tipo viene specificato subito dopo il carattere @. Si tratta in questo caso di un manuale (record di tipo manual), un libro (record di tipo book) ed una tesi di laurea (record di tipo masterthesis). Subito dopo il tipo di record si indica una *chiave*, che servirà per identificare la pubblicazione. Infine, si riempie una serie di campi che definiscono la pubblicazione. Ogni campo assume la forma:

< nome del campo > = < contenuto del campo >

Sebbene sia possibile gestire i riferimenti in formato BibT_EX con un semplice editor di testo, è spesso più comodo e veloce usare degli strumenti appositi. In rete è possibile reperire molteplici software open source: ad esempio uno dei software conosciuti è JabRef², sviluppato in Java e compatibile con sistemi operativi Mac OS X, Microsoft Windows e Linux. JabRef ha una intuitiva interfaccia grafica che consente di editare i file BibT_EX, importare i dati scientifici da database online, e gestire e ricercare file in formato BibT_EX. Si illustra una schermata dell'applicazione JabRef, figura 2.3 nella pagina successiva.

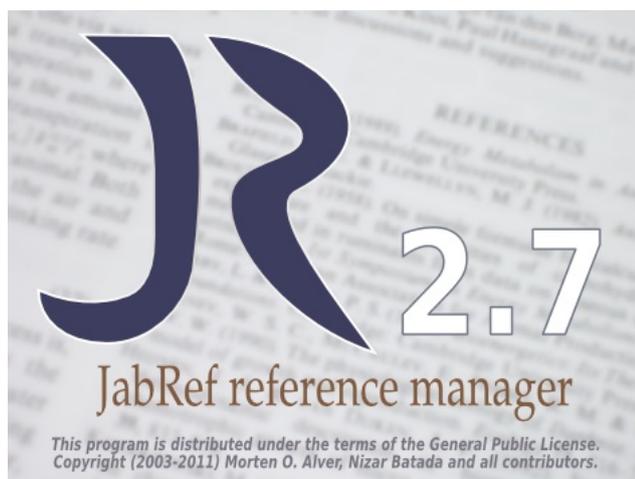


Figura 2.2: Versione JabRef.

²<http://jabref.sourceforge.net/> ultima visita settembre 2011

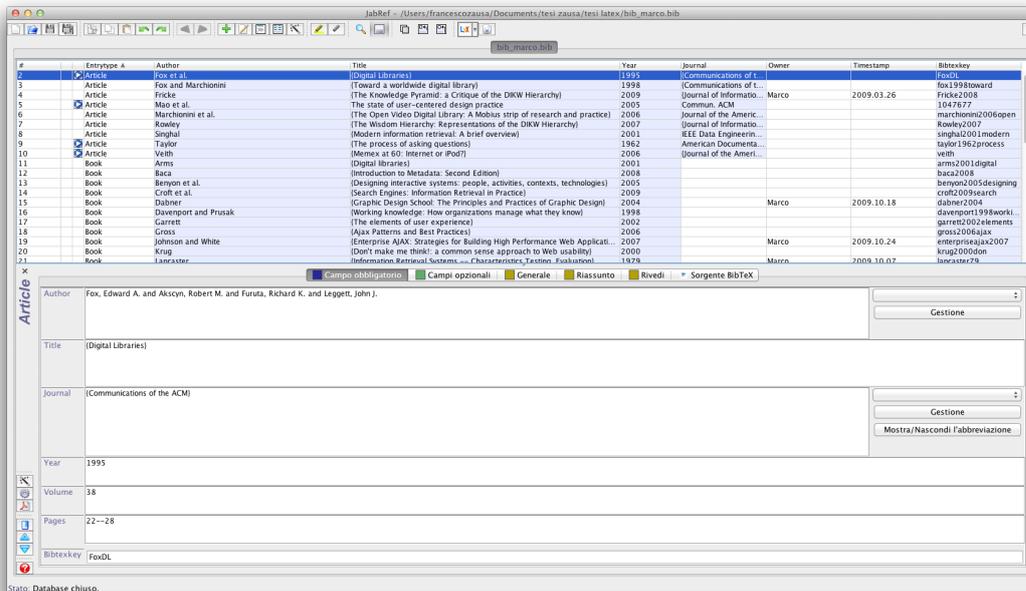


Figura 2.3: Schermata d'ambiente JabRef.

2.2 L^AT_EX

L^AT_EX³ è un linguaggio usato per la preparazione di testi. L^AT_EX ha trovato un'ampia diffusione nel mondo accademico, grazie all'ottima gestione dell'impaginazione delle formule matematiche ed alla gestione dei riferimenti bibliografici resa possibile dal progetto gemello BibT_EX.

In L^AT_EX la bibliografia viene creata automaticamente attraverso l'uso del pacchetto biblatex. Attraverso l'uso del file con estensione .bib è possibile generare automaticamente la bibliografia del proprio documento in funzione delle citazioni in esso contenute. Ogni volta sarà necessario citare la fonte bibliografica all'interno del file di L^AT_EX si utilizza il comando apposito `\cite` (o analoghi), il cui campo è costituito dalla chiave che identifica la pubblicazione. Il risultato della compilazione del file L^AT_EX rappresentato nel listato 2.2 è rappresentato dalla figura 2.4 nella pagina successiva. Per una trattazione più approfondita si rimanda al [33] ed al capitolo 7.2 del libro [6].

```
\documentclass[a4paper]{book}
```

```
\begin{document}
```

```
Esempio della citazione della pubblicazione~\cite{pantier2011}
```

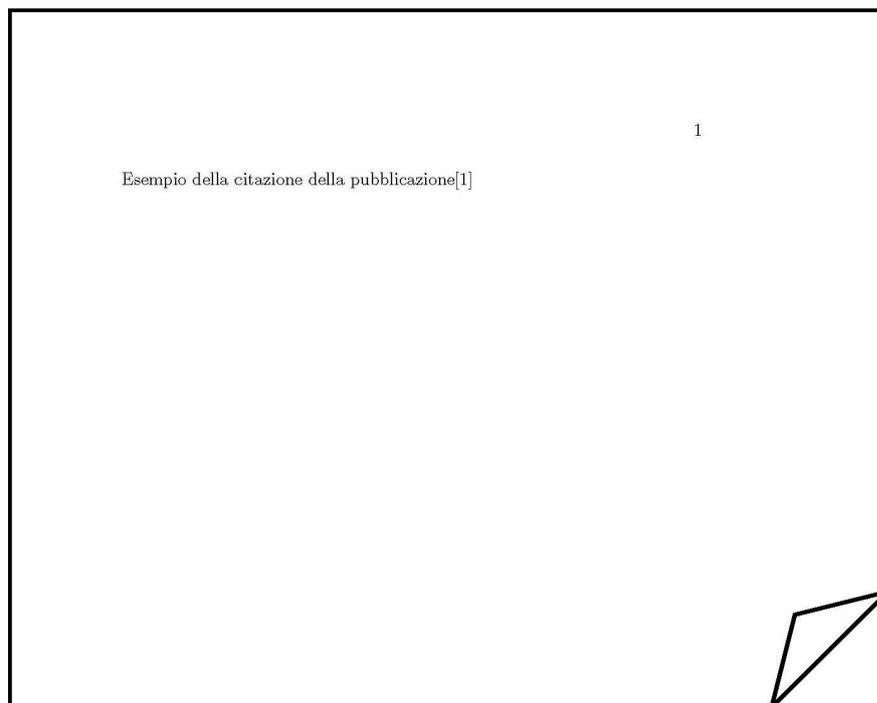
```
\bibliographystyle{plain}
```

```
\bibliography{bibliografia} % viene importato il file con estensione .bib
                             % all'interno del file .tex di LaTeX
```

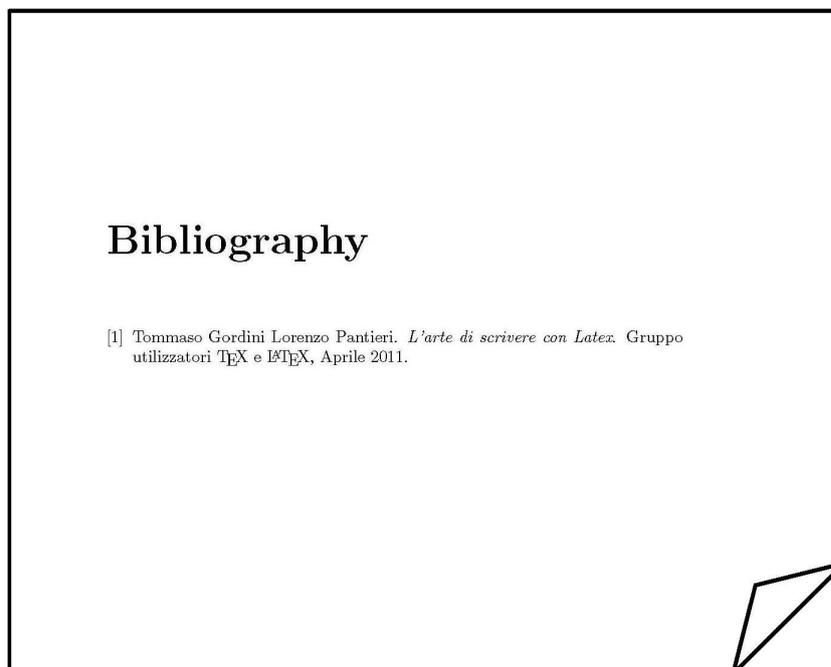
```
\end{document}
```

Codice 2.2: Esempio di codice L^AT_EX.

³<http://www.latex-project.org/> ultima visita settembre 2011



(a)



(b)

Figura 2.4: Esempio della compilazione del codice 2.2 a fronte e 2.1 a pagina 6

2.3 Realtà di interesse

Attualmente ogni membro del gruppo gestisce singolarmente le informazioni delle pubblicazioni utilizzate per il proprio lavoro aggiornando localmente la propria lista di pubblicazioni di riferimento per la gestione delle pubblicazioni suddette. Il gruppo usa le specifiche BibTeX⁴ [13] per la gestione dei riferimenti bibliografici. Un requisito fondamentale per il lavoro della tesi è rispettare le specifiche del formato BibTeX. Le informazioni delle pubblicazioni consultate vengono reperite attraverso vari modi:

- ricerca sul web attraverso sistemi di ricerca che gestiscono le informazioni delle pubblicazioni scientifiche di interesse:
 - *Digital Bibliography & Library Project (DBLP)*⁵
 - *Google Scholar*⁶
 - *Mendeley*⁷
 - *Refworks*⁸
 - *BibSonomy*⁹
- scambio dei file locali che gestiscono i riferimenti
- consultazione diretta della pubblicazione



(a) DBLP



(b) Google Scholar



(c) Mendeley



(d) Refworks



(e) BibSonomy

Figura 2.5: Loghi di alcuni strumenti di ricerca per informazioni bibliografiche

DBLP è un sistema di ricerca fruibile tramite Web delle bibliografie di Computer Science della comunità scientifica internazionale, il cui sito è ospitato presso l'Università di Trier, in Germania. Esso salva tutti i riferimenti bibliografici in un unico file XML e si presenta con interfaccia semplice e minimale. A settembre 2011 risultano inseriti più di 1.7 milioni di riferimenti a pubblicazioni.

Google Scholar è un motore di ricerca accessibile liberamente via Web che consente di individuare testi della letteratura accademica come articoli sottoposti a revisione paritaria, tesi di laurea e dottorato, libri, preprint, sommari, recensioni e rapporti tecnici di tutti i settori della ricerca scientifica, sviluppato da Google Incorporation. Google Scholar consente di reperire articoli da una vasta

⁴<http://www.bibtex.org/> ultima visita settembre 2011

⁵<http://www.informatik.uni-trier.de/~ley/db/> ultima visita settembre 2011

⁶<http://scholar.google.it/> ultima visita settembre 2011

⁷<http://www.mendeley.com/> ultima visita settembre 2011

⁸<http://www.refworks.com/> ultima visita settembre 2011

⁹<http://http://www.bibsonomy.org/> ultima visita settembre 2011

gamma di case editrici che si rivolgono al mondo dello studio e della ricerca, da associazioni scientifiche e professionali, depositi di preprint e università, oltre che nella galassia di articoli scientifici e culturali distribuiti sul Web [20].

Mendeley è un software proprietario gratuito sviluppato dalla Mendeley Limited. Sul sito è scaricabile un programma desktop oppure è possibile utilizzare un interfaccia Web per la gestione e condivisione di documenti oltre che per la ricerca e la collaborazione online [27].

Refworks è un programma per raccogliere riferimenti bibliografici e creare bibliografie, sviluppato da ProQuest LLC.

BibSonomy è un social bookmark e nello stesso tempo un sistema di condivisione di pubblicazioni. Un utente dopo essersi registrato ha la possibilità di condividere i propri segnalibri Web con chiunque esso voglia, inoltre può condividere link a pubblicazioni registrate nel Web. Il servizio è stato sviluppato dal team Knowledge and Data Engineering Group dell'Università di Kassel, in Germania.

Ognuno di questi sistemi consente di ottenere ed esportare le informazioni dei riferimenti bibliografici in formato Bib \TeX .

Ogni membro dello staff IMS grazie a questi strumenti riesce ad ottenere velocemente il frammento di codice in formato Bib \TeX da inserire nel proprio file locale. I file locali vengono gestiti tramite applicazioni open source come ad esempio JabRef illustrato precedentemente in sezione 2.1. Una gestione tale non risulta essere centralizzata in quanto ognuno aggiorna localmente i propri riferimenti bibliografici, non potendo così scambiare ed aggiornare i propri file all'interno del gruppo poiché ogni persona dello staff potrebbe utilizzare identificatori diversi per indicare una stessa pubblicazione avendo così notevole ridondanza, a meno di revisionare tutto l'elenco delle bibliografie cambiando le chiavi che le identificano con una consistente perdita di tempo. Il gruppo ha l'esigenza quindi non solamente di poter gestire i riferimenti in maniera centralizzata per un uso interno ma anche per poter lavorare con gruppi di dipartimenti di Università di tutto il mondo. Inoltre in un futuro il gruppo potrebbe avere l'esigenza di poter accedere al database tramite applicativo per esportare i riferimenti bibliografici in file con estensioni diverse dal formato Bib \TeX . Da queste esigenze, il gruppo IMS ha richiesto l'implementazione di una base di dati per la gestione delle pubblicazioni.

Una base di dati è una collezione di dati correlati, utilizzati per rappresentare le informazioni di interesse per un sistema informativo. Il sistema informativo gestisce e organizza le informazioni necessarie per perseguire gli scopi che si sono prefissati. Le informazioni vengono rappresentate per mezzo di dati, che hanno bisogno di essere interpretati per fornire informazioni. Per dati si intendono fatti noti che possono essere memorizzati e che hanno un significato implicito. Una base di dati ha le seguenti proprietà:

- rappresenta un certo aspetto del mondo reale;
- è una collezione di dati logicamente coerenti con un significato intrinseco;
- è progettata, costruita e popolata per uno scopo specifico.

Un *sistema di gestione di basi di dati* (DBMS) è un insieme di programmi che permettono agli utenti di creare e mantenere una base di dati. Il DBMS è quindi un sistema software con scopi generali che facilita il processo di definire, costruire, manipolare e condividere basi di dati per varie applicazioni. Un DBMS deve essere in grado di gestire collezioni di dati grandi, condivise e persistenti, assicurando loro

affidabilità e privacy. Inoltre un DBMS deve essere efficiente ed efficace.

Le basi di dati possono essere *grandi* in relazione alla quantità di dati gestiti, misurabile in byte, quindi un sistema deve poter gestire questa mole di dati.

Le basi di dati sono *condivise*, nel senso che utenti e applicazioni devono poter accedere a dati comuni. In questo modo si riduce la ridondanza dei dati, poiché si evitano ripetizioni e si evita la possibilità di inconsistenze: se esistono varie copie degli stessi dati è possibile che in un qualche momento risultino disallineati, cioè non uguali.

Le basi di dati sono *persistenti* nel senso che i dati presenti hanno un tempo di vita non limitato a quelle delle singole esecuzioni delle applicazioni che li utilizzano.

I DBMS sono inoltre *affidabili*, hanno la capacità di conservare intatto il contenuto della base di dati, o almeno di permetterne la ricostruzione in caso di malfunzionamenti hardware o software.

I DBMS consentono la *privacy* dei dati. Essi permettono di specificare diritti di accesso a determinati utenti, consentendo loro di svolgere determinate azioni sui dati.

Per *efficienza* dei DBMS si intende la capacità di svolgere determinate operazioni con un rapporto spazio tempo accettabile per gli utenti. Ovviamente il sistema informatico su cui verrà installato il DBMS dovrà essere dimensionato adeguatamente.

Infine per *efficacia* si intende la capacità della base di dati di rendere produttive le attività dei suoi utenti. Questa definizione è generica e non fa riferimento ad un aspetto specifico, visto che un DBMS fornisce vari servizi e funzionalità ai suoi utenti. L'attività di progettazione della base di dati e delle applicazioni che la utilizzano mira a garantire una buona efficacia complessiva del sistema.

Da queste proprietà si possono trarre alcuni vantaggi nell'utilizzo delle basi di dati e dei DBMS:

- i DBMS permettono di considerare i dati come una risorsa comune di una organizzazione;
- con l'uso di un DBMS è possibile un controllo centralizzato dei dati;
- la condivisione permette di ridurre ridondanze e inconsistenze;
- i DBMS consentono l'indipendenza dei dati, ossia permettono a utenti e programmi che utilizzano una base di dati di interagire indipendentemente dai dettagli costruttivi di quest'ultima;
- la base di dati fornisce una rappresentazione della realtà di interesse, utilizzabile nelle applicazioni attuali e, con possibili estensioni, in applicazioni future.

Il punto di partenza per la progettazione è stato il lavoro svolto dal Dott. Matteo Maritan [7] e dal Dott. Germano Rocco [3]. Il lavoro svolto dal Dott. Matteo Maritan si focalizzava sullo sviluppo di un database per un generico gruppo di ricerca. Il lavoro del Dott. Germano Rocco sviluppa un database per inserire i dati delle pubblicazioni presenti in DBLP rispecchiando in parte le specifiche BibTeX. Il gruppo di basi di dati ha richiesto espressamente di poter inserire i dati secondo le specifiche BibTeX.

Dopo l'analisi del progetto iniziale, sono stati rivisti i requisiti e si è riprogettata la base di dati sulle specifiche BibTeX in quanto più comunemente usato all'interno della comunità scientifica.

Raccolta ed analisi dei requisiti

In questa fase sono state raccolte le richieste del gruppo Information Management Systems Research Group (IMS). Si sono consultate le specifiche BibTeX per rispettarle il più possibile per l'applicazione di interesse.

L'applicazione deve fornire a chiunque sia interessato la possibilità di inserire e consultare i riferimenti alle pubblicazioni. Per *pubblicazione* si intende un documento (elettronico o cartaceo). Di ogni pubblicazione, identificata univocamente da un id, è di interesse mantenere l'autore, l'anno e il mese di pubblicazione, il collegamento alla pubblicazione, il materiale annesso alla pubblicazione (ad esempio file pdf, slide, etc.) ed eventuali note.

Gli elementi di una *bibliografia* si suddividono in vari tipi. Nel formato BibTeX vengono divisi in: article, book, booklet, conference, inbook, incollection, inproceedings, proceedings, misc, manual, techreport, unpublished, thesis che a loro volta si suddivide in masterthesis e phdthesis, come presentato in 2.1.

Per article, incollections, book, bookle, inproceedings, proceedings, misc, manual, techreport, thesis è di interesse mantenere delle informazioni aggiuntive.

Per *article* si intende un articolo di un giornale o di una rivista. Di un article è di interesse mantenere il collegamento verso un testo -- ee, l'autore.

Per *incollections* si intende la parte di un libro avente il proprio titolo. È di interesse mantenere: numero di pagine -- pages, indirizzo dell'editore (se disponibile) -- address, il nome della casa editrice (se disponibile) -- publisher, lo sponsor della conferenza (se disponibile) -- organization, il riferimento al libro in cui è stato pubblicato, l'autore, gli editor.

Per *book* si intende un libro con un esplicito editore. È di interesse mantenere: numero di pagine -- pages, indirizzo dell'editore (se disponibile) -- address, il nome della casa editrice (se disponibile) -- publisher, l'edizione del libro (se disponibile) -- edition, la collana di libri all'interno della quale è stato pubblicato (se disponibile) -- series, ISBN del libro, il volume (se disponibile) -- volume.

Per *inbook* si intende la parte di un libro, che può essere un capitolo, (o una sezione o quant'altro) o un breve intervallo di pagine. È di interesse mantenere le stesse informazioni di book.

Per *booklet* si intende un libro senza un esplicito editore. È di interesse mantenere le stesse informazioni di book.

Per *inproceedings/conference* si intende un articolo negli atti di una conferenza. È di interesse mantenere: indirizzo dell'editore (se disponibile) -- address, il nome della casa editrice (se disponibile) --

publisher, lo sponsor della conferenza (se disponibile) -- *organization*, il riferimento alla conferenza in cui l'articolo è stato discusso, l'autore, gli editor.

Per *proceedings* si intendono gli atti di una conferenza. È di interesse mantenere: l'indirizzo dell'editore (se disponibile) -- *address*, il nome della casa editrice (se disponibile) -- *publisher*, lo sponsor della conferenza (se disponibile) -- *organization*, gli editori e ISBN.

Misc si usa quando gli altri elementi non identificano ciò che si sta pubblicando. È di interesse mantenere l'autore e il modo con cui è stata pubblicata (se diverso dal modo standard).

Per *manual* si intende documentazione tecnica. È di interesse mantenere: l'indirizzo dell'editore (se disponibile) -- *address*, il nome della casa editrice (se disponibile) -- *publisher*, lo sponsor della conferenza (se disponibile) -- *organization*, l'autore.

Per *techreport* si intende un resoconto pubblicato da una scuola o da un'altra istituzione, solitamente all'interno di una serie. È di interesse mantenere: il tipo di resoconto tecnico -- *type*, per esempio "Appunto di ricerca", indirizzo dell'editore (se disponibile) -- *address*, il numero del giornale, rivista o resoconto tecnico, se applicabile, l'istituzione che è stata coinvolta nella pubblicazione -- *institution*.

Per *unpublished* si intende un documento con un autore ed un titolo, ma non formalmente pubblicato.

Per *thesis* si intende una tesi. È di interesse mantenere l'indirizzo dell'editore (se disponibile) -- *address*, l'istituto presso il quale è stata scritta la tesi e il supervisore. A sua volta una tesi può essere una *phdthesis* (una tesi di dottorato) od una *masterthesis* (una tesi di laurea).

Si può osservare, che oltre alle specifiche BibTeX, alle pubblicazioni book e proceedings il committente ha richiesto di poter mantenere l'informazione relativa all'International Standard Book Number (ISBN).

ISBN è un numero che identifica a livello internazionale in modo univoco e duraturo un titolo o una edizione di un titolo di un determinato editore. Oltre a identificare il libro, si attribuisce a tutti quei prodotti creati per essere utilizzati come libro.

L'ISBN a partire dal 1 gennaio 2007 è formato da un codice di 13 cifre, suddivise in 5 parti dai trattini di divisione:



Figura 3.1: Esempio di ISBN.

Per una trattazione più approfondita si rimanda all'indirizzo Web [23].

3.1 Figure coinvolte

Assicurare la privacy dei dati implica che non tutti gli utenti abbiano accesso alle informazioni contenute nel database. Questo consente che solamente gli utenti con particolari diritti di accesso possano

modificare o cancellare i dati presenti. Il gruppo IMS ha richiesto di implementare un unico utente amministratore del database in quanto tutti i membri del gruppo devono avere pieno accesso al database.

3.2 Glossario dei termini

La tabella 3.1 rappresenta i concetti più importanti presenti nelle specifiche, dando un nome che li identifica univocamente, una breve descrizione e mettendoli in relazione con altri concetti a loro collegati.

Tabella 3.1: Glossario dei termini.

TERMINE	DESCRIZIONE	SINONIMI	TERMINI COLLEGATI
Publication	scritto redatto in modo oggettivo su un argomento scientifico	pubblicazione scientifica	Article Journal Incollections Book Booklet Inbook Inproceedings Proceedings Mi- sc Techreport Unpublished The- sis Masterthesis Phdthesis
Article	articolo tratto da un giornale o rivista scientifica	articolo	Giornale, autore
Journal	giornale contenente articoli	giornale	articolo
Incollections	parte di un libro, può avere la supervisione di un editore ed essendo un libro avere una casa editrice		libro, editore, casa editrice
Book/Booklet/Inbook	testo con trattazione scientifica	libro	autore, incollections, casa editrice
Inproceedings	articolo inserito negli atti di una conferenza		proceedings, autore, casa editrice, editore

Tabella 3.1: continua nella prossima pagina

Tabella 3.1: continua dalla pagina precedente

TERMINE	DESCRIZIONE	SINONIMI	TERMINI COLLEGATI
Proceedings	atti di una conferenza		inproceedings, autore, casa editrice, editore
Misc	viene utilizzato quando il documento non corrisponde a nessuna degli altri tipi di pubblicazioni		autore
Manual	testo cui tratta documentazione tecnica	manuale	autore, casa editrice
Techreport	documentazione pubblicata da una scuola o da un'altra istituzione		autore,
Unpublished	documento avente autore e titolo ma non formalmente pubblicato		autore
Thesis	tesi avente un autore ed un supervisore		autore, supervisore
Masterthesis	tesi di laurea		tesi, autore, supervisore
Phdthesis	tesi di dottorato		tesi, autore, supervisore
Author	persona che crea la pubblicazione scientifica	autore	
Editor	persona che si occupa e determina il contenuto finale		
Supervisor	persona incaricata di visionare una tesi		autore

Tabella 3.1: continua nella prossima pagina

Tabella 3.1: continua dalla pagina precedente

TERMINE	DESCRIZIONE	SINONIMI	TERMINI COLLEGATI
Publisher	società che prepara e pubblica la pubblicazione scientifica	casa editrice	

Tabella 3.1: si conclude dalla pagina precedente

Progettazione Concettuale

Lo scopo della progettazione concettuale è quello di

rappresentare le specifiche informali della realtà di interesse in termini di una descrizione formale e completa, ma indipendente dai criteri di rappresentazione utilizzati nei sistemi di gestione di basi di dati [1].

Il modello concettuale utilizzato è il modello Entità-Relazione (Entity-Relationship (ER)). In tale modello si prevede che i concetti ricavati dall'analisi dei requisiti vengano rappresentati da opportuni costrutti. I costrutti principali di tale modello sono:

Entità : rappresentano classi di oggetti che hanno proprietà comuni ed esistenza autonoma ai fini dell'applicazione di interesse. Una occorrenza di un'entità è un oggetto della classe che l'entità rappresenta.

Associazioni (Relazioni) : rappresentano legami logici, significativi per l'applicazione di interesse, tra due o più entità. Una occorrenza di relazione è un'entità costituita da occorrenze di entità, una per ciascuna delle entità coinvolte.

Attributi : descrivono le proprietà elementari di entità o relazioni che sono di interesse ai fini dell'applicazione. Un attributo associa a ciascuna occorrenza di entità (o di relazione) un valore appartenente a un insieme, detto dominio, che contiene i valori ammissibili per l'attributo.

Per una trattazione più approfondita del modello Entità-Relazione si rimanda al capitolo 6.2 in [1].

4.1 Schema ER

Analizzando lo schema ER di figura 4.1 a pagina 21 del Dott. Matteo Maritan si può notare che non vengono distinti i vari tipi di pubblicazioni ma si utilizza un'unica entità *Publication*. Il lavoro svolto dal Dott. Germano Rocco, schema ER di figura 4.2 a pagina 22, si avvicina molto all'idea di ciò che si intende realizzare. Tuttavia in tale schema non sono presenti le entità e relazioni che gestiscono il tipo *manual, misc, techreport*. Il gruppo Information Management Systems Research Group (IMS) ha chiesto espressamente la presenza di tale entità in quanto potrebbero essere utilizzate all'interno del gruppo stesso. Inoltre lo schema di figura 4.2 considera le entità *book, incollections, inproceedings, proceedings*

indipendenti senza nessun legame logico tra loro. Ad esempio per inproceeding si intende un articolo negli atti di un proceeding. Questo legame logico è stato rappresentato attraverso una relazione.

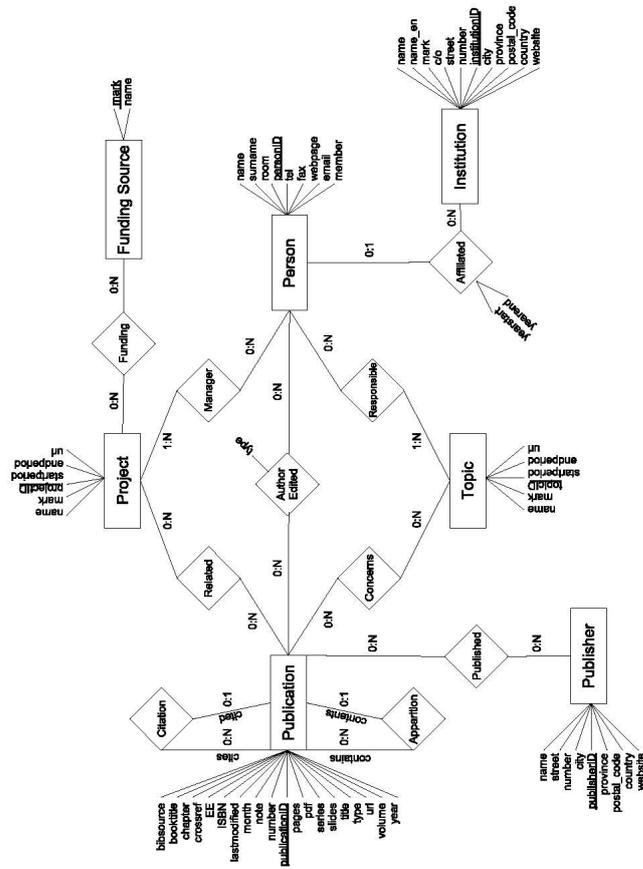


Figura 4.1: Schema ER Dott. Matteo Marittan

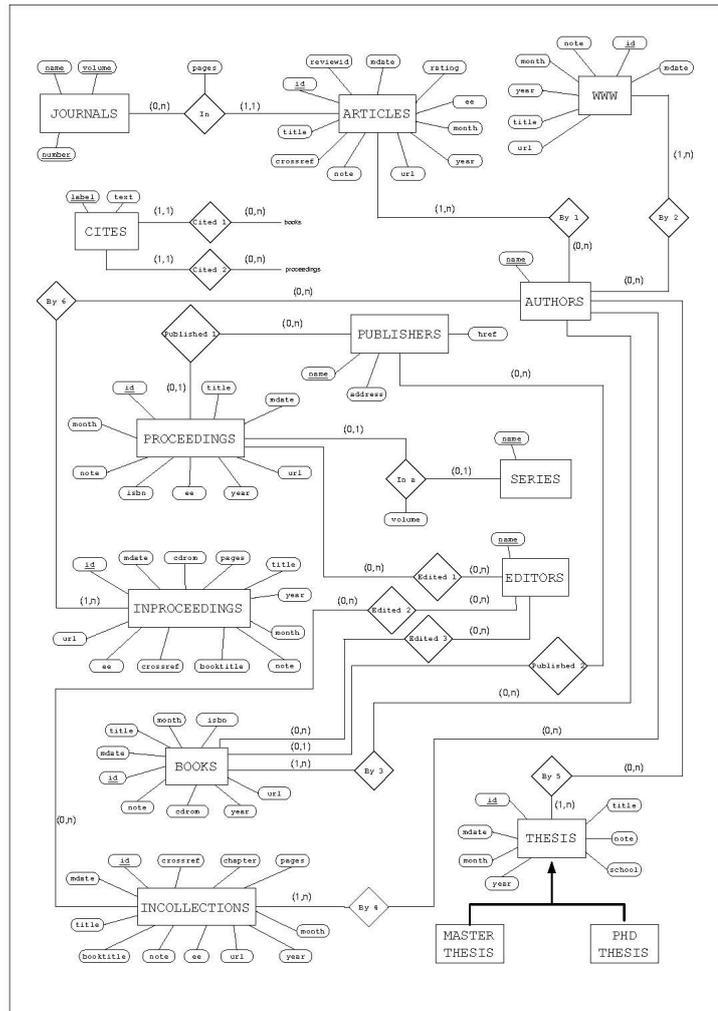


Figura 4.2: Schema ER Dott. Germano Rocco

Si è deciso quindi di riprogettare la base di dati. Lo schema ER che ne risulta è rappresentato in figura 4.3 a pagina 23:

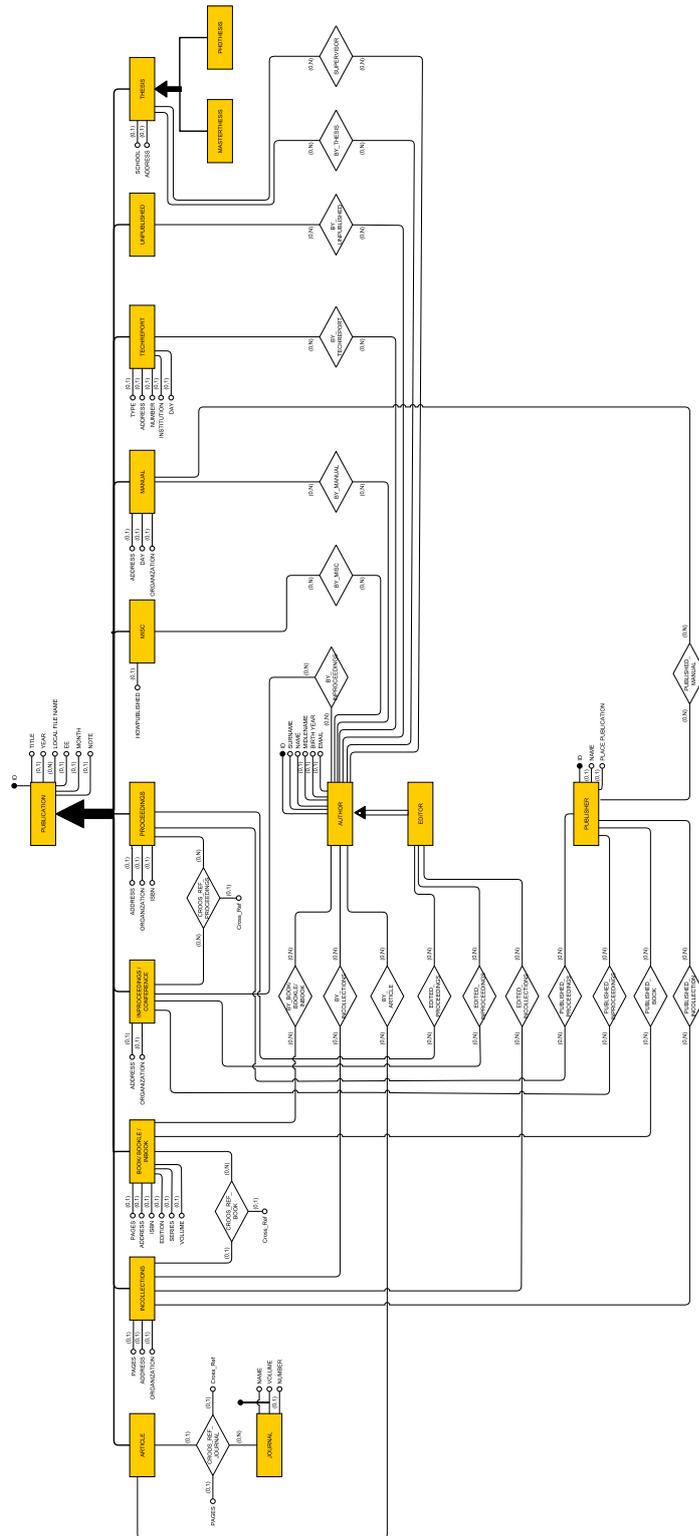


Figura 4.3: Schema ER

Il gruppo IMS ha sottolineato il fatto di mantenere lo schema ER il più rilassato possibile, nel senso di avere uno schema con meno vincoli, rispettando le quattro proprietà di uno schema concettuale: *correttezza, completezza, leggibilità, minimalità* [1]. Infatti le cardinalità sono state scelte tutte opzionali (ad eccezione degli identificatori primari) in modo tale da consentire l'inserimento dei dati senza conoscere alcuni attributi fondamentali. Inoltre le cardinalità delle relazioni tra le entità che partecipano con l'entità *author* dovrebbero avere cardinalità (1,N) perché ogni pubblicazione ha uno o più autori. Il committente ha scelto di non rispettare tale vincolo in maniera da consentire l'inserimento della pubblicazione anche se non si è a conoscenza dell'autore.

Si può notare dallo schema ER di figura 4.3 nella pagina precedente che la generalizzazione tra l'entità padre *Publication* e le entità figlie è totale ed esclusiva. Totale perché ogni occorrenza della classe padre è un'occorrenza di almeno una delle entità figlie; esclusiva perché ogni occorrenza della classe padre è al più una occorrenza di una delle entità figlie. Infatti *article*, *incollections*, *book*, ecc. sono tutte pubblicazioni, ed ogni pubblicazione può essere o *article*, o *incollections*, o *book*, ecc. Ogni figlia ha una relazione con autore con cardinalità (0,N) da entrambe le parti perché una pubblicazione può avere un autore o più, e lo stesso autore può partecipare ad una o a più pubblicazioni. L'attributo *local file name* è un attributo multivalore. Esso rappresenta, se esiste, l'indirizzo Uniform Resource Locator (URL) in locale del materiale annesso alla pubblicazione. Tale attributo è stato scelto multivalore perché si possono avere più indirizzi URL per una stessa pubblicazione.

La generalizzazione rappresentata qui sotto in figura 4.4 è parziale ed esclusiva: un editore è anche un autore, ma un autore può non partecipare come editore.

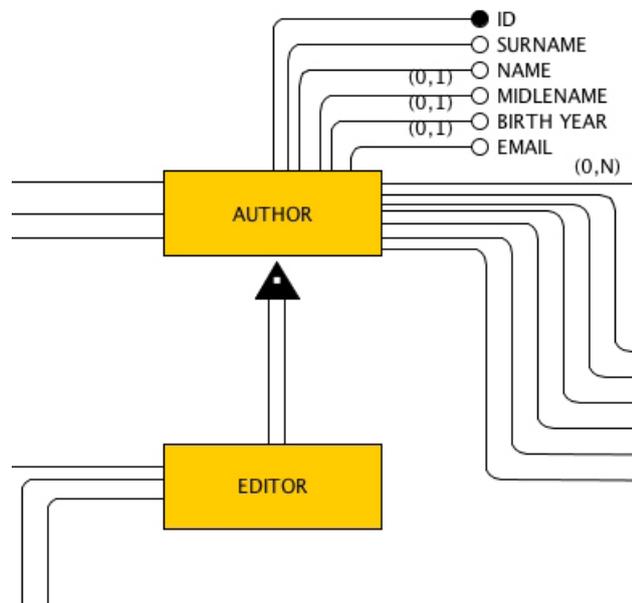
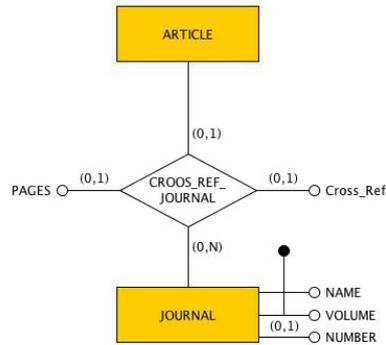
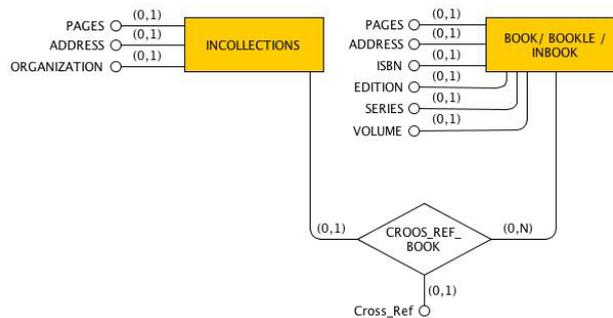


Figura 4.4: Generalizzazione tra le entità Author ed Editor

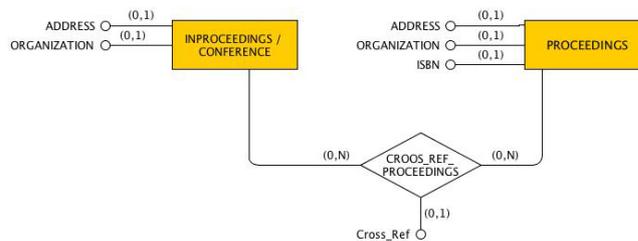
In figura 4.5 le relazioni *cross_ref_journal*, *cross_ref_book* e *cross_ref_proceedings* hanno un attributo *cross_ref*. Ad esempio se in un articolo è pubblicato in un determinato giornale, questo viene rappresentato dalla relazione tra *Article*, *Journal*. L'attributo *cross_ref* rappresenta il riferimento in cui l'oggetto *Article* è contenuto nell'oggetto *Journal*.



(a) Relazione Author-Journal



(b) Relazione Incollections-Book



(c) Relazione Inproceedings-Proceedings

Figura 4.5: Relazioni che coinvolgono l'attributo *cross_ref*

4.2 Tabella Entità

La tabella 4.1 elenca le entità coinvolte nello schema ER di figura 4.3 a pagina 23. Per ogni entità viene data una breve descrizione, l'elenco degli attributi e gli identificatori. A ciascun attributo è associato un dominio, ovvero un insieme che contiene i valori ammissibili per quell'attributo. Per gli attributi composti da stringhe di carattere è stato scelto un dominio di tipo VARCHAR, che consiste in una stringa di lunghezza variabile la cui lunghezza massima è indicata tra parentesi. Per gli attributi che indicano date, come ad esempio year, è stato utilizzato il dominio date.

Tabella 4.1: Tabella Entità.

ENTITÀ	DESCRIZIONE	ATTRIBUTI E DOMINI		IDENTIFICATORE
Publication	Pubblicazione scientifica generica	Id	VARCHAR (16)	Id
		Title	VARCHAR (256)	
		Year	DATE	
		Local file name	VARCHAR (256)	
		EE	VARCHAR (256)	
		Month	DATE	
		Note	VARCHAR (1000)	
Article	Articolo tratto da un giornale o da una rivista			
Journal	Giornale contenente vari articoli	Name	VARCHAR (256)	Name, Volume
		Volume	VARCHAR (8)	
		Number	VARCHAR (8)	
Incollections	Parte di libro avente proprio titolo	Pages	VARCHAR (16)	
		Address	VARCHAR (100)	
		Organization	VARCHAR (100)	

Tabella 4.1: continua nella prossima pagina

Tabella 4.1: continua dalla pagina precedente

ENTITÀ	DESCRIZIONE	ATTRIBUTI E DOMINI		IDENTIFICATORE
Book/Booklet/Inbook	Libro	Pages	VARCHAR (16)	
		Address	VARCHAR (100)	
		ISBN	VARCHAR (17)	
		Edition	VARCHAR (20)	
		Series	VARCHAR (20)	
		Volume	VARCHAR (20)	
Inproceedings	Un articolo negli atti di una conferenza	Address	VARCHAR (100)	
		Organization	VARCHAR (100)	
Proceedings	Gli atti di una conferenza	Address	VARCHAR (100)	
		Organization	VARCHAR (100)	
		ISBN	VARCHAR (17)	
MISC	Si usa quando la pubblicazione o il documento non corrisponde a nessuna delle altre entità	howpublished	VARCHAR (100)	
Manual	Documentazione Tecnica	Address	VARCHAR (100)	
		Organization	VARCHAR (100)	
		Day	DATE	
Techreport	Resoconto pubblicato da una scuola o da altra istituzione	Type	VARCHAR (100)	
		Address	VARCHAR (100)	
		Number	VARCHAR (8)	
		Institution	VARCHAR (100)	
		Day	DATE	

Tabella 4.1: continua nella prossima pagina

Tabella 4.1: continua dalla pagina precedente

ENTITÀ	DESCRIZIONE	ATTRIBUTI E DOMINI		IDENTIFICATORE
Unpublished	Documento con autore e titolo, ma non formalmente pubblicato			
Thesis	tesi generica	School	VARCHAR (100)	
		Address	VARCHAR (100)	
Masterthesis	Tesi di laurea			
Phdthesis	Tesi di dottorato			
Author	Autore di una pubblicazione	Id	VARCHAR (16)	Id
		Surname	VARCHAR (35)	
		Name	VARCHAR (35)	
		Middlename	VARCHAR (35)	
		Birth_day	DATE	
		Email	VARCHAR (50)	
Editor	Persona che si occupa e determina il contenuto finale dei proceedings, inproceedings, incollections [15]			
Publisher	Società che prepara e pubblica book, proceedings, inproceedings, incollections [16]	Id	VARCHAR (16)	Id
		Place_publication	VARCHAR (50)	
		Name	VARCHAR (100)	

Tabella 4.1: si conclude dalla pagina precedente

4.3 Tabella Associazioni

La tabella 4.2 elenca le relazioni presenti nello schema ER di figura 4.3 a pagina 23. Per ogni relazione vi è una breve descrizione. Anche in questo caso sono presenti gli attributi con i relativi domini. Viene inoltre elencato per ogni relazione le entità coinvolte con le relative cardinalità tra parentesi.

Tabella 4.2: Tabella Associazioni

ASSOCIAZIONE	DESCRIZIONE	ATTRIBUTI E DOMINI		ENTITÀ COINVOLTE
Cross_Ref_Journal	Associa un articolo ad un giornale	Pages	VARCHAR (16)	ARTICLE (0,1)
		Cross_Ref	VARCHAR (156)	JOURNAL (0,N)
Cross_Ref_Book	Associa la parte di un libro al libro	Cross_Ref	VARCHAR (156)	INCOLLECTIONS (0,1)
				BOOK/BOOKLET/ INBOOK (0,N)
Cross_Ref_Proceedings	Associa un articolo negli atti di una conferenza ad una conferenza	Cross_Ref	VARCHAR (156)	INPROCEEDINGS (0,N)
				PROCEEDINGS (0,N)
By_Book	Associa un libro ad un autore			BOOK (0,N)
				AUTHOR (0,N)
By_Incollection	Associa la parte di un libro ad un autore			INCOLLECTIONS (0,N)
				AUTHOR (0,N)
By_Article	Associa un articolo ad un autore			ARTICLE (0,N)
				AUTHOR (0,N)
By_Inproceedings	Associa un articolo negli atti di una conferenza ad un autore			INPROCEEDINGS (0,N)
				AUTHOR (0,N)
By_Misc	Associa un misc ad un autore			MISC (0,N)

Tabella 4.2: continua nella prossima pagina

Tabella 4.2: continua dalla pagina precedente

ASSOCIAZIONE	DESCRIZIONE	ATTRIBUTI E DOMINI	ENTITÀ COINVOLTE
			AUTHOR (0,N)
By_Techreport	Associa un resoconto pubblicato da una scuola o da una istituzione ad un autore		TECHREPORT (0,N)
			AUTHOR (0,N)
By_Unpublished	Associa un documento non pubblicato ad un autore		UNPUBLISHED (0,N)
			AUTHOR (0,N)
By_Thesis	Associa una tesi ad un autore		THESIS (0,N)
			AUTHOR (0,N)
Supervisor	Associa una tesi ad un relatore		THESIS (0,N)
			AUTHOR (0,N)
By_Manual	Associa un documento tecnico ad un autore		MANUAL (0,N)
			AUTHOR (0,N)
Edited_Proceedings	Associa una conferenza ad un editore		PROCEEDINGS (0,N)
			EDITOR (0,N)
Edited_Inproceedings	Associa un articolo negli atti di una conferenza ad un editore		INPROCEEDINGS (0,N)
			EDITOR (0,N)
Edited_Incollections	Associa la parte di un libro ad un editore		INCOLLECTIONS (0,N)
			EDITOR (0,N)

Tabella 4.2: continua nella prossima pagina

Tabella 4.2: continua dalla pagina precedente

ASSOCIAZIONE	DESCRIZIONE	ATTRIBUTI E DOMINI	ENTITÀ COINVOLTE
Published_Proceedings	Associa una conferenza ad una casa editrice		PROCEEDINGS (0,N)
			PUBLISHER (0,N)
Published_Inproceedings	Associa un articolo negli atti di una conferenza ad una casa editrice		INPROCEEDINGS (0,N)
			PUBLISHER (0,N)
Published_Book	Associa un libro ad una casa editrice		BOOK (0,N)
			PUBLISHER (0,N)
Published_Incolections	Associa la parte di un libro ad una casa editrice		INCOLLECTIONS (0,N)
			PUBLISHER (0,N)

Tabella 4.2: si conclude dalla pagina precedente

Progettazione Logica

In questa fase si riorganizzerà lo schema ER ottenuto nel capitolo 4 operando una *ristrutturazione*. Per operare correttamente si dovrebbe tener conto delle prestazioni dell'applicazione che si sta realizzando analizzando il carico applicativo del sistema, ovvero il calcolo del numero di occorrenze che l'applicazione deve gestire, ed il numero degli accessi alle varie entità ed associazioni. Tuttavia il committente ha sottolineato il fatto di non soffermarsi su tale procedimento e di sviluppare direttamente lo schema ER ristrutturato risolvendo ovviamente le generalizzazioni, scegliendo gli identificatori per le varie entità e accorpendo o scomponendo concetti.

Prodotto finale di questa fase di progettazione è lo *schema logico*, ottenuto traducendo lo schema ER ristrutturato.

5.1 Ristrutturazione dello schema ER

5.1.1 Eliminazione delle generalizzazioni

Nello schema ER di figura 4.3 a pagina 23 sono presenti due generalizzazioni totali ed una parziale, eliminate come segue.

- **ELIMINAZIONE DELLE GENERALIZZAZIONI TOTALI**

La prima generalizzazione è stata eliminata accorpendo l'entità padre *Publication* alle figlie *Article*, *Incollections*, *Book*, *Inproceedings*, *Proccedings*, *Misc*, *Manual*, *Techreport*, *Unpublished*, *Thesis*. Tale scelta è dettata dal fatto che si effettuano prevalentemente operazioni sulle occorrenze di una entità figlia (vedi figura 5.3 a pagina 35).

La seconda generalizzazione di figura 5.1 nella pagina successiva è stata eliminata accorpendo le figlie al padre. Al padre è stato aggiunto l'attributo *category* che serve a distinguere il tipo di una occorrenza di *Thesis*, cioè se tale occorrenza appartiene a *Masterthesis* o *Phdthesis* (vedi figura 5.3 a pagina 35).

- **ELIMINAZIONE DELLA GENERALIZZAZIONE PARZIALE**

La generalizzazione di figura 4.4 a pagina 24 è stata eliminata accorpendo la figlia al padre. Al padre è stato aggiunto l'attributo *editor*.

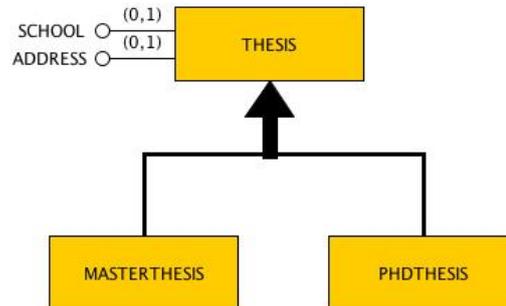


Figura 5.1: Generalizzazione totale tra l'entità padre Thesis e le entità figlie Masterthesis e Phdthesis

5.1.2 Eliminazione degli attributi multivalore

Il solo attributo multivalore presente nello schema ER di figura 5.3 a fronte è l'attributo *local file name* appartenente all'entità *Publication*. L'attributo è stato sostituito realizzando tante entità e associazioni quante erano le figlie della generalizzazione *Publication*. L'altra possibile scelta era realizzare un'unica entità "Local File Name", ma realizzando lo schema logico si ottiene un'unica tabella con molte chiavi referenzianti (figura 5.2). Nell'inserimento dei dati si avrebbero molti valori nulli. Per evitare ciò è stata scelta la prima ipotesi.

LOCAL_FILE_NAME										
PATH	ARTICLE	INCOLLECTIONS	BOOK	INPROCEEDINGS	PROCEEDINGS	MISC	MANUAL	TECHREPORT	UNPUBLISHED	THESIS

Figura 5.2: Possibile schema logico dell'attributo multivalore local file name

5.1.3 Scelta degli identificatori

In questa fase si selezionano gli identificatori qualora vi siano più chiavi possibili per le varie entità. Nello schema ER ristrutturato si sono mantenuti gli identificatori scelti nello schema ER.

5.2 Schema ER Ristrutturato

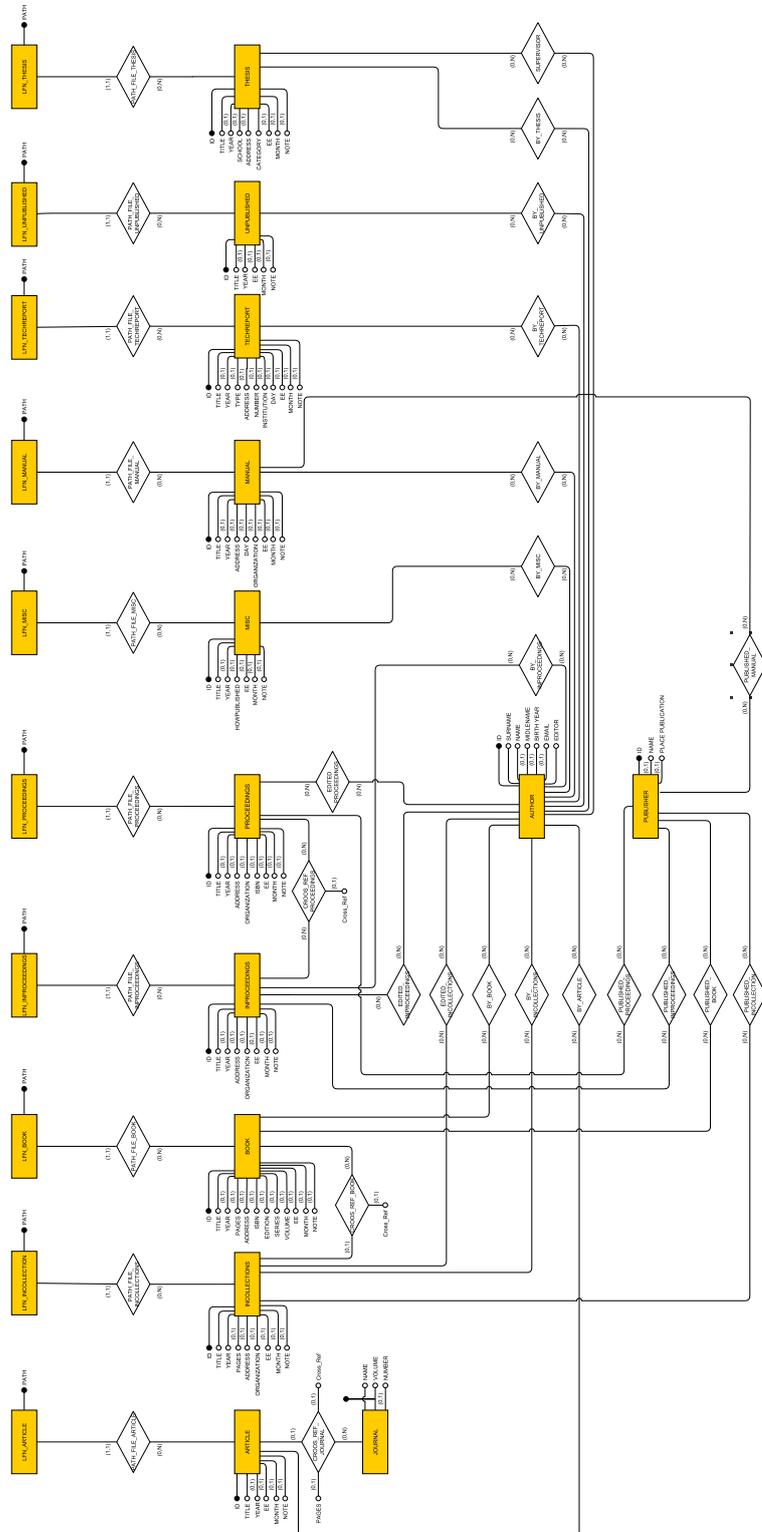


Figura 5.3: Schema ER Ristrutturato

5.3 Schema Logico

La seconda fase della progettazione logica corrisponde alla traduzione dello schema ER ristrutturato (figura 4.3 a pagina 23) in uno schema logico equivalente. Tale schema rappresenta i dati secondo il modello relazionale, in cui un dato è visto come una relazione avente un nome ed un elenco di attributi. La relazione è paragonabile ad una tabella in cui gli attributi sono le intestazioni delle colonne. Le righe inserite nella tabella vengono dette tuple e rappresentano le occorrenze vere e proprie dei dati. Nella traduzione da schema ER ristrutturato a schema logico ogni entità è stata tradotta come una relazione, rappresentata da una tabella. La prima riga contiene il nome della relazione. La seconda riga contiene l'elenco degli attributi che la compongono. Le chiavi primarie, i cui valori sono usati per identificare tuple nella relazione e che quindi rispetta il vincolo di univocità secondo il quale nessuna coppia di tuple distinte possa avere valori uguali, sono sottolineate. Nello schema logico sono state tradotte anche alcune associazioni, mentre altre sono state accorpate nelle entità.

I vincoli di integrità referenziale sono rappresentati diagrammaticamente attraverso una linea di colore blu che collega ciascuna chiave esterna verso la relazione che essa riferisce. Il vincolo di integrità referenziale stabilisce che una tupla in una relazione che fa riferimento ad un'altra relazione deve far riferimento a una tupla esistente in quella relazione. Il vincolo si basa sul concetto di *chiave esterna*. Un insieme di attributi *foreign key (FK)* nello schema di relazione R_1 è una chiave esterna di R_1 che riferisce la relazione R_2 se:

1. gli attributi della chiave esterna hanno gli stessi domini degli attributi della chiave primaria a cui si riferiscono
2. il valore degli attributi della chiave esterna sono presenti negli attributi della chiave primaria oppure il valore è nullo.

Se questi due condizioni sono verificate si dice che esiste un vincolo di integrità referenziale.

Per consentire una maggiore leggibilità, lo schema logico proposto è stato suddiviso in più pagine dove:

- Le chiavi primarie sono sottolineate.
- Le chiavi esterne sono collegate dalla linea di colore blu collegate con una freccia alla relazione a cui fanno riferimento.
- Le relazioni già analizzate sono rappresentate in grigio.

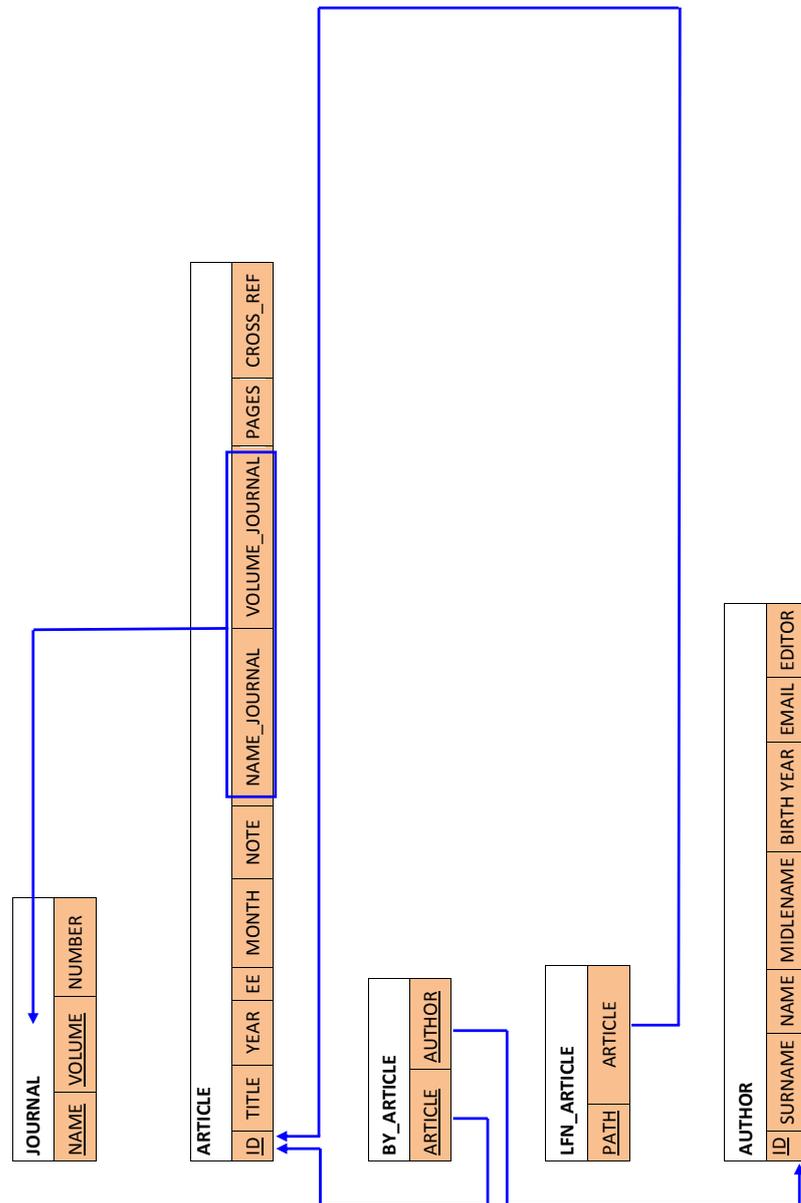


Figura 5.4: Schema logico Journal, Article, By_Article, LFN_Article, Author.

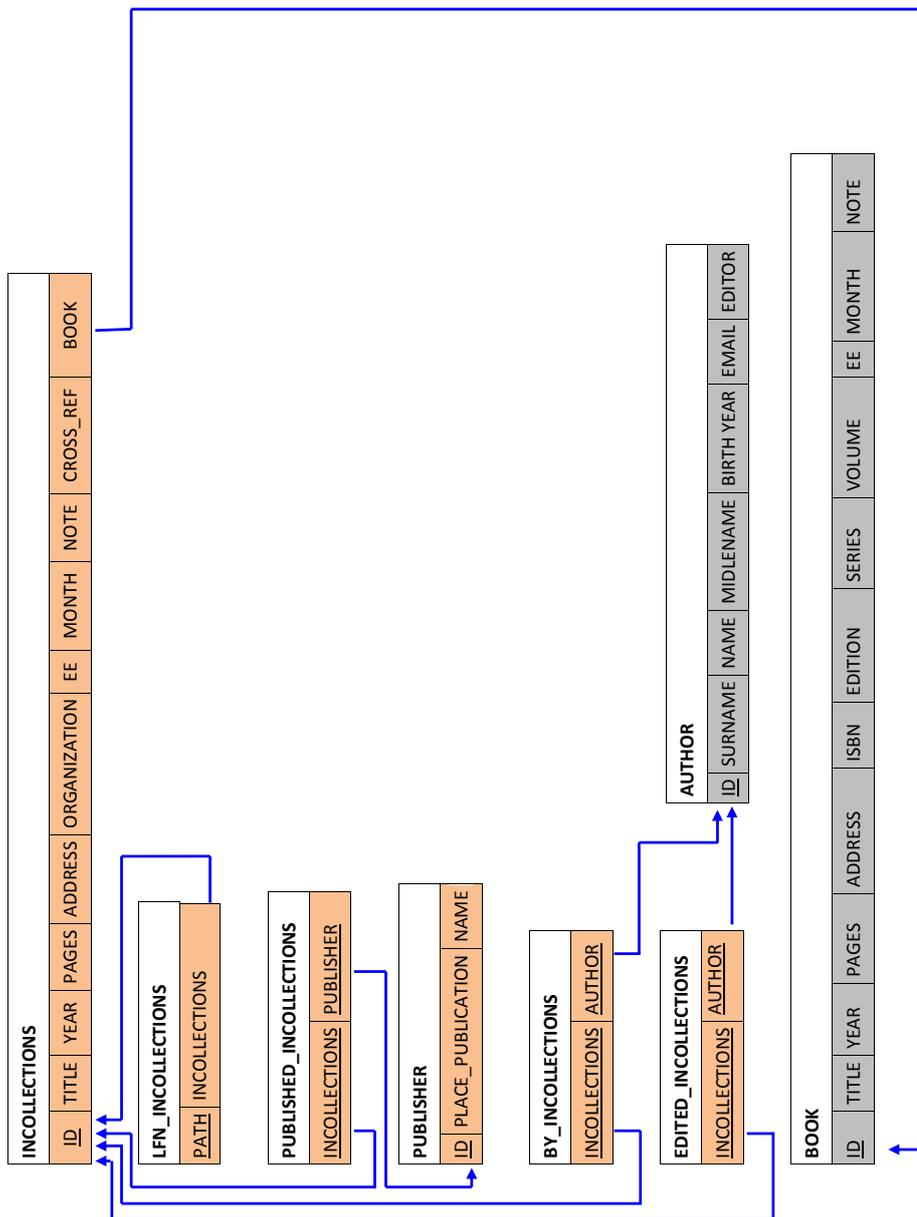


Figura 5.5: Schema logico di Incollections, LFN_Incollections, Published_Incollections, Publisher, By_Incollections, Edited_Incollections.

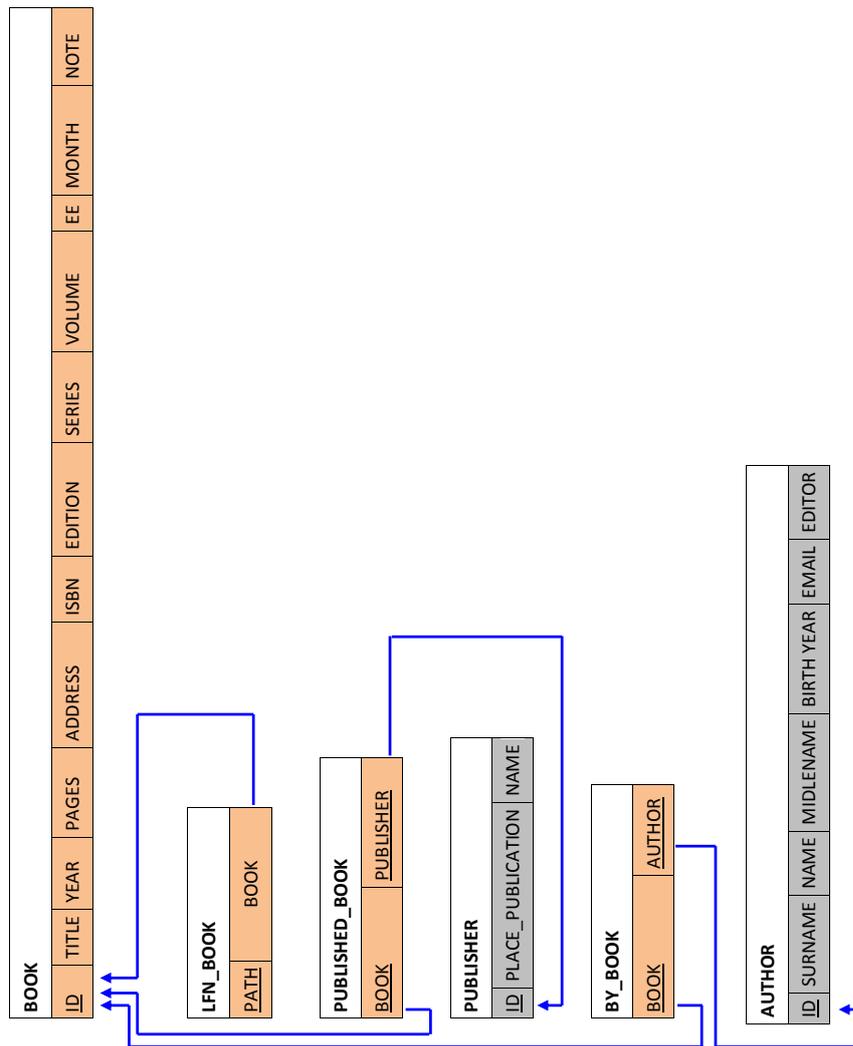


Figura 5.6: Schema logico di Book, LFN_Book, Published_Book, By_Book.

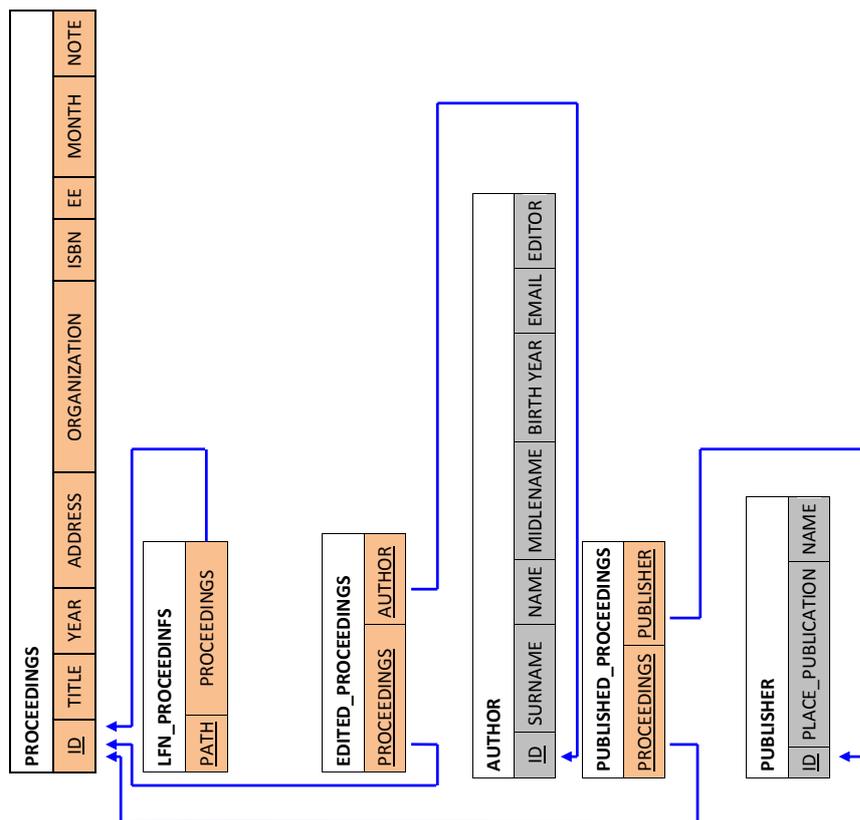


Figura 5.8: Schema logico di Proceedings, LFN_Proceedings, Edited_Proceedings, Published_Proceedings.

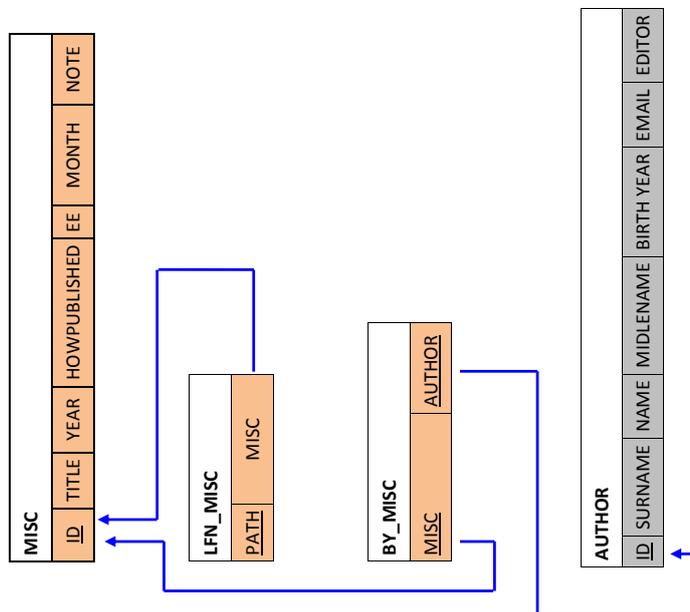


Figura 5.9: Schema logico Misc, LFN_Misc, By_Misc.

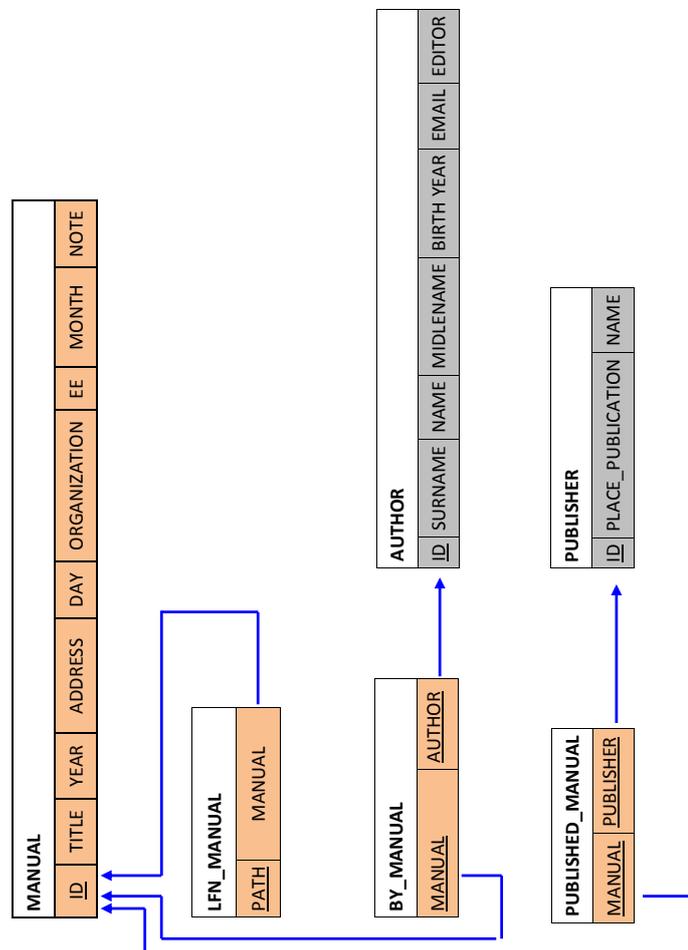


Figura 5.10: Schema logico di Manual, LFN_Manual, By_Manual, Published_Manual.

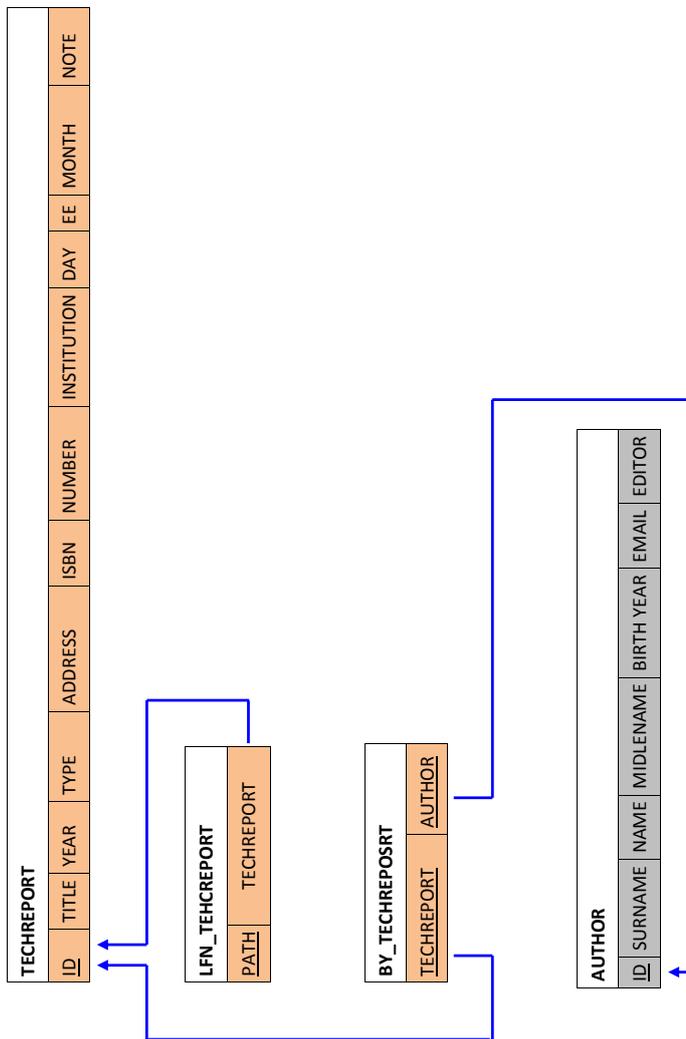


Figura 5.11: Schema di Techreport, LFN_Techreport, By_Techreport.

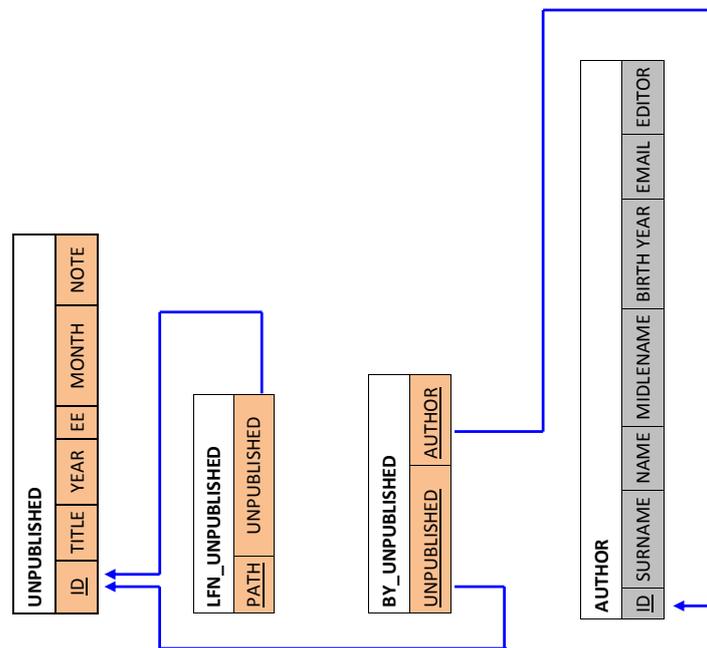


Figura 5.12: Schema logico di Unpublished, LFN_Unpublished, By_Unpublished.

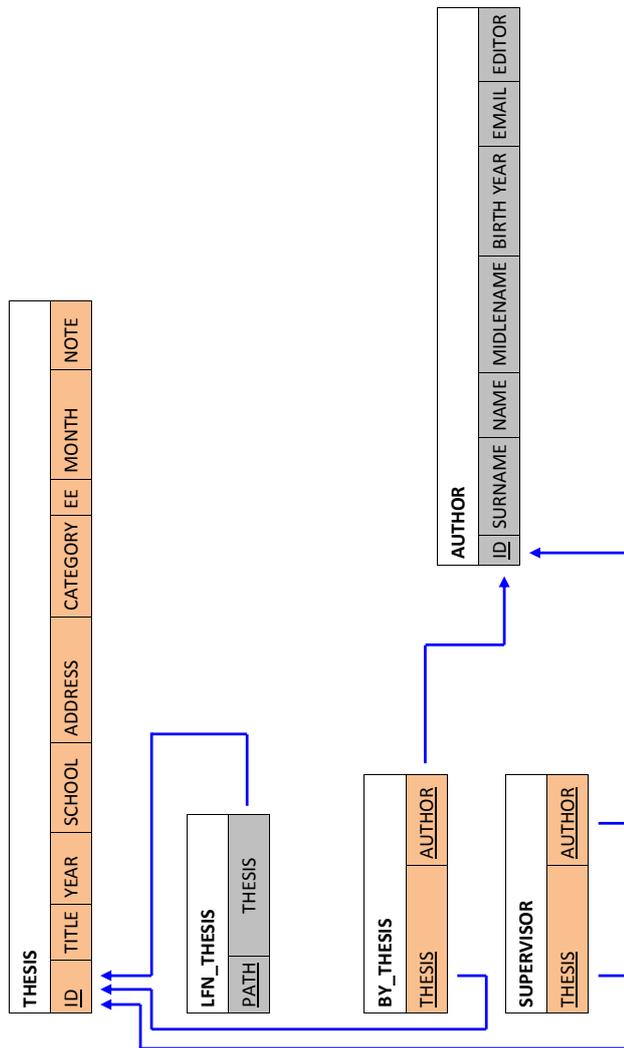


Figura 5.13: Schema logico di Thesis, By_Thesis, Supervisor.

5.3.1 Qualità dello schema logico

La qualità dello schema logico viene valutata secondo gli strumenti messi a disposizione dalla *teoria della normalizzazione* [1] [9].

La normalizzazione è un procedimento che consente di trasformare schemi non normalizzati in nuovi schemi i quali soddisfano una forma normale. Quando uno schema non soddisfa una forma normale esso presenta ridondanza ed anomalie nelle operazioni di aggiornamento.

Il processo di normalizzazione sottopone uno schema di relazione a una serie di test per “certificare” se soddisfa una data *forma normale*. Va detto che le metodologie di progetto viste nei capitoli precedenti permettono di solito di ottenere schemi che soddisfano una forma normale.

Vi sono diverse forme normali, tutte soddisfatte dallo schema proposto.

La *prima forma normale* (1NF) prevede che non vi siano attributi multivalore; l’unico attributo presente era l’attributo `local file name` che è stato eliminato sostituendolo con una entità-associazione.

La *seconda e terza forma* (2NF e 3NF) normale si occupano di verificare la bontà delle chiavi scelte, analizzando il rapporto che intercorre tra queste e gli attributi che da esse dipendono. Lo schema proposto soddisfa ambedue queste forme normali. Vi è un’altra forma normale, detta di *Boyce e Codd*. Per soddisfare tale forma normale sono state fatte delle verifiche di qualità che garantiscono l’assenza di anomalie di aggiornamento nella base di dati in quanto concetti indipendenti sono rappresentati da relazioni distinte. Per una trattazione approfondita delle forme normali si rinvia a [1] e [9].

Progettazione Fisica

Ultima fase della progettazione è la cosiddetta progettazione fisica, ovvero la traduzione dello schema logico in linguaggio Structured Query Language (SQL), il linguaggio standard per i sistemi di gestione di basi di dati commerciali. L'intero listato è riportato in appendice B a pagina 73.

Ogni relazione è stata tradotta come una tabella, sono stati indicati i domini degli attributi, le chiavi primarie e le chiavi esterne. Il linguaggio prevede che ci sia la possibilità di decidere come il database debba comportarsi in caso di modifica o cancellazione di una relazione che possiede chiavi esterne. In caso di cancellazione di una occorrenza che viola il vincolo di integrità referenziale si può avere vari modi con le occorrenze contenenti i valori referenzianti: si può settare i valori referenzianti ad un valore di default (in SQL corrisponde a `ON DELETE SET DEFAULT`), cancellare le occorrenze (`ON DELETE CASCADE`) o impedire la cancellazione della tupla contenente il valore referenziato (`ON DELETE NO ACTION`). Stessa cosa per le modifiche: si può propagare la modifica (`ON UPDATE CASCADE`), impedirla (`ON UPDATE NO ACTION`) o settare i valori che fanno riferimento ad un valore referenziato ad un valore di default `ON UPDATE SET DEFAULT`. In questo caso la scelta generale è stata quella di impedire le cancellazioni, in modo che non si possa eliminare una tupla se prima non si è eliminato tutto ciò che le si riferisce, mentre per le modifiche si è deciso di propagarle in modo da consentire aggiornamenti o una correzione di eventuali errori. Ad esempio in questo schema vi sono alcune relazioni come `journal`, `book`, cui fanno riferimento altre relazioni. Consentire una propagazione delle cancellazioni significherebbe la perdita di molti dati in caso di cancellazione accidentale di un'occorrenza. Tuttavia la scelta intrapresa presenta molte eccezioni. Vi sono propagazioni di cancellazioni la cui perdita di dati non sia significativa. Ad esempio se vi è una cancellazione di un libro viene cancellato anche le tuple presenti nella tabella `LFN_BOOK` che si riferiscono a quel libro, in quanto non vi è interesse mantenere l'URL locale.

6.1 Scelte Progettuali

Il gruppo IMS come specifiche per l'implementazione del DBMS ha richiesto l'utilizzo di PostgreSQL¹ [29]. La release usata di PostgreSQL per l'implementazione del DBMS è la release 8.4.8. Il

¹<http://www.postgresql.org> ultima visita settembre 2011

DBMS è stato realizzato attraverso l'utilizzo dell'interfaccia grafica *pgAdmin*² [28] versione 1.12.3.

6.1.1 pgAdmin 1.12.3

pgAdmin è un'applicazione C++, una interfaccia grafica che consente di amministrare in modo semplificato database di PostgreSQL. L'applicazione è indirizzata sia agli amministratori del database, sia agli utenti. Gestisce i permessi prelevandoli dal database PostgreSQL. pgAdmin permette di creare un database da zero, creare le tabelle ed eseguire operazioni di ottimizzazione sulle stesse. Tale interfaccia ha il vantaggio di realizzare un database importando direttamente codice SQL.

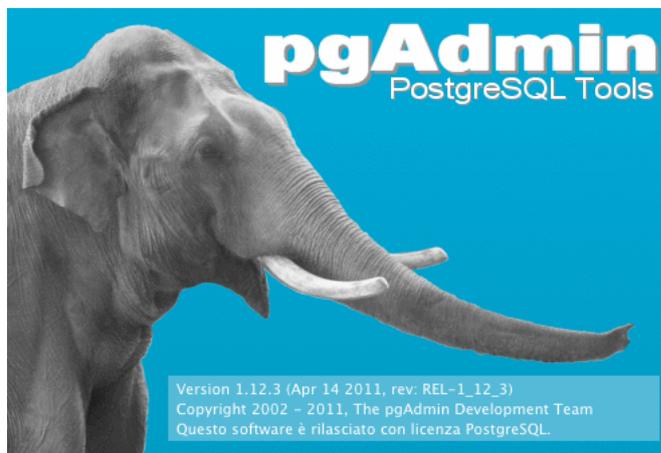


Figura 6.1: Versione pgAdmin

Osservando la figura 6.2 a fronte, la finestra principale di pgAdmin visualizza la struttura dei database gestiti dal server Postgres. Permette di creare, cancellare o modificare oggetti, se i privilegi dell'utente attualmente connesso lo permettono. La parte sinistra della finestra mostra l'albero dei server Postgres raggiungibili e gli oggetti che i server contengono. Il vantaggio di pgAdmin è che consente di visualizzare e inserire i dati attraverso una visualizzazione stile foglio di calcolo, figura 6.3 nella pagina successiva. L'inserimento può essere effettuato mediante la riga contrassegnata dal simbolo asterisco. In questa modalità è anche possibile modificare una tupla o eliminarla. PgAdmin inoltre tramite lo strumento per le interrogazioni -- *Query Tool*, permette di eseguire comandi SQL. Il risultato dell'istruzione viene visualizzato in una finestra (figura 6.4 a pagina 52).

²<http://www.pgadmin.org/> ultima visita settembre 2011

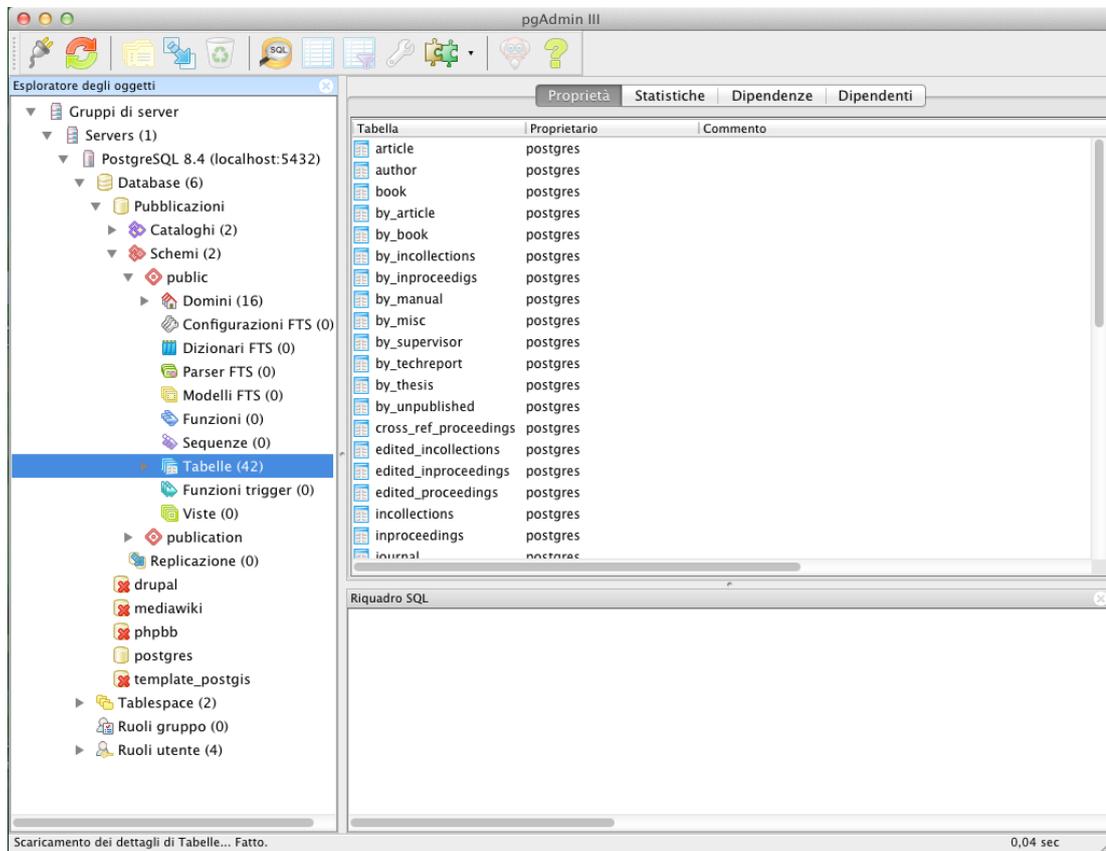


Figura 6.2: Schermata di ambiente pgAdmin

Modifica dati - PostgreSQL 8.4 (localhost:5432) - Pubblicazioni - article

Nessun limite

	id [PK] codice	titolo	year anno	ee riferim	month mese	note appunti	name_journal character var	volume_journ character var	pages character var	cross_ref character var
1	AgostiCG91	Issues of Data Modelling in Information Retrieval	1991				Electronic	4	4	
2	DBLP:journ	Specialised Hardware for Database Management and Ir	1984	db/jou			SIGIR Forum	18	13-40	
*										

Spazio per appunti

2 righe.

Figura 6.3: Schermata tabella Article

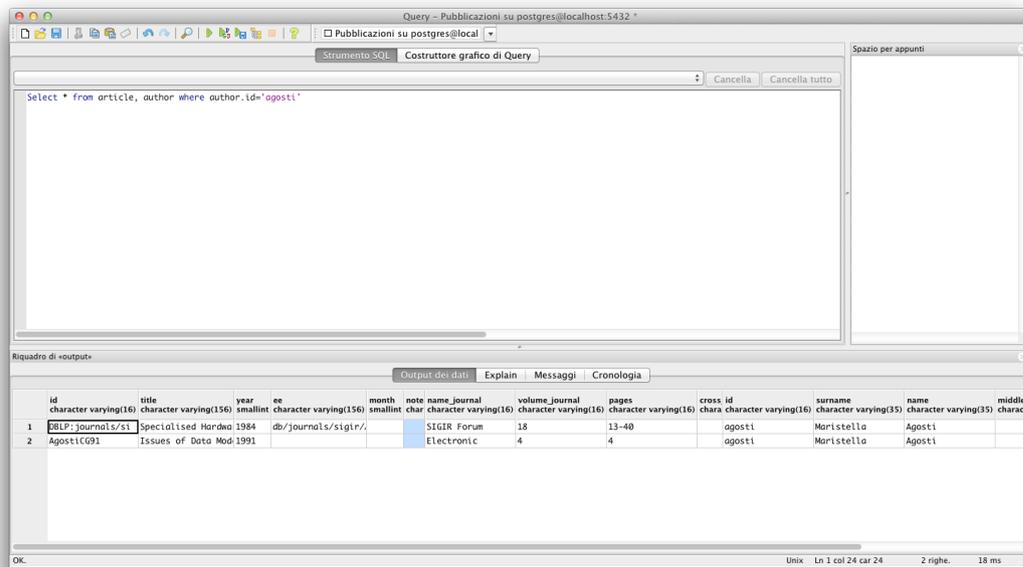


Figura 6.4: Schermata strumento per le interrogazioni SQL del database

6.1.2 PostgreSQL 8.4.8



Figura 6.5: Logo Sviluppatore
PostgreSQL
Global
Development
Group

PostgreSQL è un sistema di gestione di basi di dati relazionale ad oggetti (object relational database management system (ORDBMS)) basato su POSTGRES, versione 4.2³, sviluppato dal Computer Science Department dell'Università Berkeley della California⁴. PostgreSQL viene distribuito con licenza libera (stile licenza Berkeley Software Distribution (BSD)⁵). Questa licenza non impone vincoli e non garantisce nessuna garanzia all'utente finale a differenza della licenza General Public License (GPL).

Storia

Fonte tratta da <http://it.wikipedia.org/wiki/PostgreSQL> ultima visita settembre 2011

Inizialmente il DBMS si chiamava INGRES ed era un progetto della Università di Berkeley. Nel 1982 il capo progetto, Michael Stonebraker, lasciò la Berkeley per commercializzare il prodotto, ma in seguito tornò nel contesto accademico. Nel 1985 lasciò nuovamente l'università per dare vita a un progetto post-Ingres (Postgres) che superasse gli evidenti limiti dei prodotti concorrenti dell'epoca. Le basi dei sorgenti di Ingres e di Postgres erano, e sono rimaste nel tempo, ben distinte. Il nuovo progetto puntava a fornire un supporto completo ai tipi di dati, in particolare la possibilità di definire nuovi tipi di dati, UDF (User Defined Function), User Defined Types. Vi era anche la possibilità di descrivere la relazione tra le entità (tabelle), che fino ad allora veniva lasciata completamente all'utente. Perciò non solo Postgres preservava l'integrità dei dati, ma era in grado di leggere informazioni da tabelle relazionate in modo naturale, seguendo le regole definite dall'utente. Dal 1986 gli sviluppatori diffusero un gran numero di articoli che descrivevano il nuovo sistema e nel 1988 venne rilasciato un primo prototipo funzionante. La versione 1 venne rilasciata nel giugno del 1989 per un numero di utenti contenuto. Seguì una versione 2 nel giugno del 1990, in cui il sistema delle regole venne completamente riscritto. Nella versione 3, del 1991, questo sistema venne riscritto ancora, ma venne aggiunto anche il supporto a gestori multipli di immagazzinamento dei dati e un motore di query migliorato. Nel 1993 vi era già un numero di utenti notevole che inondava il team di sviluppo con richieste di supporto e di nuove caratteristiche. Dopo aver rilasciato la versione 4, che fu principalmente un ripulimento del codice, il progetto terminò. Sebbene il progetto Postgres fosse ufficialmente abbandonato, la licenza BSD dava modo agli sviluppatori Open Source di ottenere una copia del software per poi migliorarlo a loro discrezione. Nel 1994 due studenti del Berkeley, Andrew Yu e Jolly Chen aggiunsero a Postgres un interprete SQL per rimpiazzare il vecchio QUEL che risaliva ai tempi di Ingres. Il nuovo software venne quindi rilasciato sul web col nome di Postgres95. Nel 1996 cambiò nome di nuovo: per

³<http://s2k-ftp.CS.Berkeley.EDU:8000/postgres/postgres.html> ultima visita settembre 2011

⁴<http://www.cs.berkeley.edu/> ultima visita settembre 2011

⁵<http://opensource.org/licenses/bsd-license.php> ultima visita settembre 2011

evidenziare il supporto al linguaggio SQL, venne chiamato PostgreSQL. Il primo rilascio di PostgreSQL è stata la versione 6. Da allora ad occuparsi del progetto è una comunità di sviluppatori volontari provenienti da tutto il mondo che si coordina attraverso Internet. Alla versione 6 ne sono seguite altre, ognuna delle quali ha portato nuovi miglioramenti. Nel gennaio 2005 è stata rilasciata la 8, la prima nativa per Windows. Sebbene la licenza permetta la commercializzazione del software, il codice di Postgres non è stato sviluppato commercialmente con la stessa rapidità di Ingres. Ad un certo punto, però, Paula Hawthorn (membro originale del team di Ingres) e Michael Stonebraker formarono una società chiamata Illustra Information Technologies per commercializzarlo.

Descrizione

PostgreSQL è un DBMS open source. Ha più di 15 anni di sviluppo attivo ed ha un'architettura testata che gli ha permesso di acquisire una reputazione positiva per affidabilità, integrità dei dati e precisione. Funziona su tutti i sistemi operativi maggiormente noti incluso Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), e Windows. Rispetta completamente il modello Atomicity, Consistency, Isolation, Durability (ACID) [10], quindi le transazioni rispettano queste quattro proprietà. Una transazione è una sequenza di istruzioni collegate che devono essere trattate come un'unità indivisibile rispettando le quattro proprietà:

- Atomicità (*Atomicity*): la transizione è nella sua esecuzione, produce un effetto visibile oppure fallisce senza alcun effetto sulla base di dati; non sono ammesse esecuzioni parziali.
- Consistenza (*Consistency*): la transizione porta la base di dati da uno stato coerente ad un altro stato coerente, non devono verificarsi contraddizioni tra i dati archiviati nella base di dati (devono essere rispettati i vincoli di integrità).
- Isolamento (*Isolation*): ogni transazione deve essere eseguita in modo isolato dalle altre transazioni, il fallimento di una transizione non deve interferire con le altre transazioni in esecuzione.
- Durabilità o persistenza (*Durability*): l'effetto di una transizione è permanente, ossia i cambiamenti apportati da quest'ultima non dovranno essere più persi.

PostgreSQL ha pieno supporto per foreign keys, joins, views, triggers e procedure di memorizzazioni (in linguaggi multipli). Include più tipi di dati SQL:2008, incluso INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, e TIMESTAMP [31].

In tabella 6.1 nella pagina successiva vengono rappresentati alcuni dati significativi di PostgreSQL.

6.2 Codice SQL e Licenze d'uso

Il codice SQL tramite il quale è stato implementato il database viene esposto in appendice B a pagina 73. Per comodità di lettura è stato suddiviso in più pagine. Nel listato B.1 viene rappresentato la creazione del database e dei domini. Le istruzioni nei listati B.2, B.3, B.4, B.5, B.6, B.7, B.8, B.9, B.10, B.11 e B.12 permettono la creazione delle varie relazioni.

Il codice SQL per la creazione del database viene distribuito con licenza d'uso open source. La particolarità di tale licenza è che consente di usare, modificare, integrare, riprodurre, duplicare e distribuire, senza imporre obblighi a ricompensare economicamente gli autori del codice. Lo scopo primario

Limiti	Valore
Grandezza max del DB	Illimitato
Grandezza max di una tabella	32 TB
Grandezza max di righe	1.6 TB
Grandezza max di un campo	1 GB
Max righe in una tabella	Illimitato
Max colonne in una tabella	250 - 1600 (dipende dal tipo di dati nelle colonne)
Max indici per tabella	Illimitato

Tabella 6.1: Limiti PostgreSQL

delle licenze open source non è la gratuità del software, ma la sua sopravvivenza ovvero la certezza che vi sia la possibilità per chiunque e in qualunque momento, anche futuro, di apportare miglioramenti o comunque modifiche al programma, e di installarlo senza alcuna limitazione [26]: il codice SQL del database realizzato lo si vuole distribuire con tale politica.

La scelta tra le licenze d'uso da includere nel codice ricade tra General Public License (GPL) e Lesser General Public License (LGPL). Di seguito verranno descritte in breve queste due licenze open source. Per un trattazione più approfondita si rimanda al [19]

6.2.1 Licenza GNU General Public License

La General Public License (GNU) è una licenza per software libero. È comunemente indicata con l'acronimo GNU GPL o semplicemente GPL. Il progetto GNU è partito nel 1984 da Richard Stallman e si basa su una gestione particolare dei diritti d'autore sul software. Contrapponendosi alle licenze per software proprietario, la GNU GPL assicura all'utente libertà di utilizzo, copia, modifica e distribuzione. La GPL ha incontrato un gran successo fra gli autori di software sin dalla sua creazione, e ad oggi è la più diffusa licenza per il software libero. Come ogni licenza software, la GPL è un documento legale associato al programma rilasciato sotto tale licenza. Come ogni licenza di software libero, essa concede ai licenziatari il permesso di modificare il programma, di copiarlo e di ridistribuirlo con o senza modifiche, gratuitamente o a pagamento. La GPL può essere identificata come "persistente" e "propagativa". È persistente perché se l'utente distribuisce copie del software, deve farlo secondo i termini della GPL stessa. In pratica, deve distribuire il testo della GPL assieme al software e corredarlo del codice sorgente o di istruzioni per poterlo ottenere ad un costo nominale (questo concetto è stato battezzato da Richard Stallman come "copyleft"). È propagativa nel senso che l'unione di un codice coperto da GPL con un altro può essere distribuito solamente sotto licenza GPL [18]. L'utente è libero di prendere qualsiasi listato rilasciato sotto GPL e modificarlo a proprio piacimento. Fino a quando il lavoro derivato da questo rimane all'interno di un ambito circoscritto, l'utente non è chiamato a rendere disponibile il codice sorgente dell'applicazione derivata. Tale diritto si estende anche agli ambienti professionali. I vincoli di GPL divengono più stringenti in caso si desiderasse rendere disponibile al pubblico un programma tutelato da tale licenza oppure un suo derivato. In tale circostanza, il soggetto è chiamato ad utilizzare la stessa licenza GPL, e non è concesso imporre vincoli più restrittivi.

6.2.2 Licenza GNU Lesser General Public License

La Lesser General Public License (LGPL) è una licenza di software libero creata dalla Free Software Foundation, studiata come compromesso tra la GNU General Public License. La differenza tra le due consiste nel fatto che la licenza LGPL permette il riutilizzo del software all'interno di programmi con licenza diversa dalla GPL, ossia all'interno di programmi proprietari.

6.2.3 Scelta della licenza

La scelta tra le due licenze è influenzata solamente dagli scopi del singolo sviluppatore. Il codice del database è distribuito liberamente secondo licenza GNU LGPL. Questa scelta è stata dettata dal fatto che non si pensa sia un ostacolo la diffusione, l'utilizzo e/o la modifica del codice all'interno di soli software liberi o proprietari.

I contratti di licenza GPL e LGPL si possono leggere in appendice A.

Conclusioni

In questo lavoro, sono state discusse la progettazione e implementazione di una base di dati per la gestione di bibliografie scientifiche digitali, ponendo l'attenzione sulle specifiche richieste del gruppo Information Management System (IMS). La documentazione contenuta in questa tesi fornisce una panoramica completa del lavoro svolto, consentendo al committente di capire le scelte progettuali intraprese.

La collaborazione in questi mesi con il gruppo IMS, ha dato l'opportunità di confrontarsi con un committente reale e conoscerne l'ambiente che lo circonda. Al conseguimento di questo lavoro si è arrivati anche grazie alla interazione e integrazione con il team. La realizzazione concreta della collaborazione all'interno del gruppo è stata permessa dal confronto e dal passaggio da un punto di vista individuale ad un punto di vista comune e condiviso per realizzare al meglio l'obiettivo previsto. La comunicazione poi è stata il processo chiave per permettere lo scambio di informazioni finalizzato al raggiungimento del risultato. Questa comunicazione aperta e chiara è stata frutto dell'ottima atmosfera che si è potuto percepire all'interno di IMS.

Il progetto è nato con lo scopo di essere una solida base che possa essere in futuro modellata e ampliata. Per quanto riguarda gli aspetti tecnici, la base di dati è stata pensata per rimanere stabile nel tempo e per poter sopportare interrogazioni e un buon carico di dati.

Un successivo passo per il lavoro futuro potrebbe essere quello di installare e testare il database nell'architettura del Dipartimento di Ingegneria dell'Informazione (DEI)¹.

Un'ulteriore estensione potrebbe essere quella di implementare un'applicazione che si interfacci alla base di dati per consentire l'inserimento, la consultazione e l'editing dei dati.

Potrebbe essere poi costruito uno strato software in grado di manipolare e trasformare i risultati nei formati BibTeX, html, xml, foglio di calcolo, pdf, ecc...

Questa tesi è stata un'esperienza preziosa, sia per la crescita delle competenze individuali, sia per aver acquisito i meccanismi per poter lavorare in modo efficace all'interno di un gruppo di lavoro.

¹<http://www.dei.unipd.it/> ultima visita settembre 2011

Contratti Licenze d'uso

A.1 GNU GPL

GNU GENERAL PUBLIC LICENSE Version 3, 29 June 2007
Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users’ Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- (a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- (b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- (c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- (d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- (a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- (b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- (c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- (d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- (e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- (a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- (b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

- (c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- (d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- (e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- (f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of

this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others’ Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
```

```
Copyright (C) <textyear> <name of author>
```

```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
```

```
This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.
```

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

A.2 GNU LGPL

GNU LESSER GENERAL PUBLIC LICENSE Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the “GNU GPL” refers to version 3 of the GNU General Public License.

“The Library” refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An “Application” is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A “Combined Work” is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the “Linked Version”.

The “Minimal Corresponding Source” for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The “Corresponding Application Code” for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
- b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
- c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.
- d) Do one of the following:
 - 0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.
 - 1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
- e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.
- b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy’s public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Appendice **B**

Codice SQL

```
CREATE SCHEMA PUBLICATION;  
  
CREATE DOMAIN CODE AS VARCHAR (16);  
4 CREATE DOMAIN TITLES AS VARCHAR (256);  
CREATE DOMAIN YEARS AS SMALLINT;  
6 CREATE DOMAIN REFERENT_E AS VARCHAR (256);  
CREATE DOMAIN MONTHS AS SMALLINT CHECK (VALUE IN ('1','2','3','4','5','6','7','8','9',  
8 '10','11','12'));  
CREATE DOMAIN NOTES AS VARCHAR (1000);  
10 CREATE DOMAIN PATHNAME AS VARCHAR (256);  
CREATE DOMAIN TYPE_THESIS AS VARCHAR (6) CHECK (VALUE IN ('MASTER', 'PHD'));
```

Codice B.1: Creazione database e domini.

```
12 CREATE TABLE AUTHOR  
( ID CODE PRIMARY KEY,  
14 SURNAME VARCHAR (35) NOT NULL,  
NAME VARCHAR (35) NOT NULL,  
16 MIDDLENAME VARCHAR (35),  
BIRTHYEAR YEARS,  
18 EMAIL VARCHAR (50)  
);  
  
CREATE TABLE JOURNAL  
22 ( NAME VARCHAR (256),  
VOLUME VARCHAR (8),  
24 NUMBER VARCHAR (8),  
PRIMARY KEY (NAME, VOLUME)  
26 );
```

Codice B.2: Creazione tabella Author e Journal.

```
CREATE TABLE ARTICLE
28 ( ID      CODE PRIMARY KEY,
    TITLE   TITLES NOT NULL,
30  YEAR    YEARS,
    EE      REFERENT_E,
32  MONTH   MONTHS,
    NOTE    NOTES,
34  NAME_JOURNAL VARCHAR (256),
    VOLUME_JOURNAL VARCHAR (16),
36  PAGES   VARCHAR (16),
    CROSS_REF VARCHAR (156),
38  FOREIGN KEY(NAME_JOURNAL, VOLUME_JOURNAL) REFERENCES JOURNAL (NAME, VOLUME)
        ON DELETE NO ACTION ON UPDATE CASCADE
40 );

42 CREATE TABLE BY_ARTICLE
( ARTICLE  CODE NOT NULL,
44  AUTHOR   CODE NOT NULL,
    PRIMARY KEY (ARTICLE,AUTHOR),
46  FOREIGN KEY (ARTICLE) REFERENCES ARTICLE (ID)
        ON DELETE CASCADE ON UPDATE CASCADE,
48  FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
        ON DELETE CASCADE ON UPDATE CASCADE
50 );

52 CREATE TABLE LFN_ARTICLE
( PATH  PATHNAME PRIMARY KEY,
54  ARTICLE  CODE NOT NULL,
    FOREIGN KEY (ARTICLE) REFERENCES ARTICLE (ID)
56  ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE PUBLISHER
60 ( ID      CODE PRIMARY KEY,
    PLACE_PUBLICATION VARCHAR (50),
62  NAME    VARCHAR(100)
);
```

Codice B.3: Creazione tabella Article, By_Article, LFN_Article e Publisher.

```

64 CREATE TABLE BOOK
  ( ID      CODE PRIMARY KEY,
66  TITLE   TITLES NOT NULL,
    YEAR   YEARS,
68  PAGES   VARCHAR (16),
    ADDRESS VARCHAR (100),
70  ISBN    VARCHAR (17),
    EDITION VARCHAR (20),
72  SERIES  VARCHAR (20),
    VOLUME  VARCHAR (20),
74  EE      REFERENT_E,
    MONTH   MONTHS,
76  NOTE    NOTES
  );

CREATE TABLE INCOLLECTIONS
80 ( ID      CODE PRIMARY KEY,
    TITLE   TITLES NOT NULL,
82  YEAR   YEARS,
    PAGES   VARCHAR (16),
84  ADDRESS VARCHAR (100),
    ORGANIZATION VARCHAR (100),
86  EE      REFERENT_E,
    MONTH   MONTHS,
88  NOTE    NOTES,
    CROSS_REF VARCHAR (156),
90  BOOK     CODE,
    FOREIGN KEY (BOOK) REFERENCES BOOK (ID)
92          ON DELETE NO ACTION ON UPDATE CASCADE
  );

CREATE TABLE LFN_INCOLLECTIONS
96 ( PATH     PATHNAME PRIMARY KEY,
    INCOLLECTIONS CODE NOT NULL,
98  FOREIGN KEY (INCOLLECTIONS) REFERENCES INCOLLECTIONS (ID)
    ON DELETE CASCADE ON UPDATE CASCADE
100 );

102 CREATE TABLE PUBLISHED_INCOLLECTIONS
  ( INCOLLECTIONS CODE NOT NULL,
104  PUBLISHER CODE NOT NULL,
    FOREIGN KEY (INCOLLECTIONS) REFERENCES INCOLLECTIONS (ID)
106          ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (PUBLISHER) REFERENCES PUBLISHER (ID)
108          ON DELETE CASCADE ON UPDATE CASCADE
  );

```

Codice B.4: Creazione tabella Book, Incollections, LFN_Incollections e Published_Incollection.

```
110 CREATE TABLE BY_INCOLLECTIONS
( INCOLLECTIONS CODE NOT NULL,
112 AUTHOR CODE NOT NULL,
PRIMARY KEY (INCOLLECTIONS,AUTHOR),
114 FOREIGN KEY (INCOLLECTIONS) REFERENCES INCOLLECTIONS (ID)
ON DELETE CASCADE ON UPDATE CASCADE,
116 FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
ON DELETE CASCADE ON UPDATE CASCADE
118 );

120 CREATE TABLE EDITED_INCOLLECTIONS
( INCOLLECTIONS CODE NOT NULL,
122 AUTHOR CODE NOT NULL,
FOREIGN KEY (INCOLLECTIONS) REFERENCES INCOLLECTIONS (ID)
124 ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
126 ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE LFN_BOOK
132 ( PATH PATHNAME PRIMARY KEY,
BOOK CODE NOT NULL,
134 FOREIGN KEY (BOOK) REFERENCES BOOK (ID)
ON DELETE CASCADE ON UPDATE CASCADE
136 );

138 CREATE TABLE PUBLISHED_BOOK
( BOOK CODE NOT NULL,
140 PUBLISHER CODE NOT NULL,
FOREIGN KEY (BOOK) REFERENCES BOOK (ID)
142 ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (PUBLISHER) REFERENCES PUBLISHER (ID)
144 ON DELETE CASCADE ON UPDATE CASCADE
);
```

Codice B.5: Creazione tabella By_Incollections, Edited_Incollections, LFN_Book e Published_Book.

```
146 CREATE TABLE BY_BOOK
( BOOK      CODE NOT NULL,
148  AUTHOR   CODE NOT NULL,
  PRIMARY KEY (BOOK,AUTHOR),
150  FOREIGN KEY (BOOK) REFERENCES BOOK (ID)
      ON DELETE CASCADE ON UPDATE CASCADE,
152  FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
      ON DELETE CASCADE ON UPDATE CASCADE
154 );

156 CREATE TABLE INPROCEEDINGS
( ID      CODE PRIMARY KEY,
158  TITLE  TITLES NOT NULL,
  YEAR   YEARS,
160  ADDRESS VARCHAR (100),
  ORGANIZATION VARCHAR (100),
162  EE     REFERENT_E,
  MONTH  MONTHS,
164  NOTE  NOTES
);

CREATE TABLE PROCEEDINGS
168 ( ID      CODE PRIMARY KEY,
  TITLE  TITLES NOT NULL,
170  YEAR   YEARS,
  ADDRESS VARCHAR (100),
172  ISBN   VARCHAR (17),
  EE     REFERENT_E,
174  MONTH  MONTHS,
  NOTE  NOTES
176 );

CREATE TABLE LFN_INPROCEEDINGS
180 ( PATH     PATHNAME PRIMARY KEY,
  INPROCEEDINGS CODE NOT NULL,
182  FOREIGN KEY (INPROCEEDINGS) REFERENCES INPROCEEDINGS (ID)
      ON DELETE CASCADE ON UPDATE CASCADE
184 );
```

Codice B.6: Creazione tabella By_Book, Inproceedings, Proceedings e LFN_Inproceedings.

```

CREATE TABLE CROSS_REF_PROCEEDINGS
186 ( INPROCEEDINGS CODE NOT NULL,
    PROCEEDINGS CODE NOT NULL,
188 CROSS_REF VARCHAR (156),
    FOREIGN KEY (INPROCEEDINGS) REFERENCES INPROCEEDINGS (ID)
190         ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (PROCEEDINGS) REFERENCES PROCEEDINGS (ID)
192         ON DELETE CASCADE ON UPDATE CASCADE
    );

CREATE TABLE EDITED_INPROCEEDINGS
196 ( INPROCEEDINGS CODE NOT NULL,
    AUTHOR CODE NOT NULL,
198     FOREIGN KEY (INPROCEEDINGS) REFERENCES INPROCEEDINGS (ID)
        ON DELETE CASCADE ON UPDATE CASCADE,
200     FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
        ON DELETE CASCADE ON UPDATE CASCADE
202 );

204 CREATE TABLE BY_INPROCEEDINGS
    ( INPROCEEDINGS CODE NOT NULL,
206     AUTHOR CODE NOT NULL,
        PRIMARY KEY (INPROCEEDINGS,AUTHOR),
208     FOREIGN KEY (INPROCEEDINGS) REFERENCES INPROCEEDINGS (ID)
        ON DELETE CASCADE ON UPDATE CASCADE,
210     FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
        ON DELETE CASCADE ON UPDATE CASCADE
212 );

214 CREATE TABLE PUBLISHED_INPROCEEDINGS
    ( INPROCEEDINGS CODE NOT NULL,
216     PUBLISHER CODE NOT NULL,
        FOREIGN KEY (INPROCEEDINGS) REFERENCES INPROCEEDINGS (ID)
218         ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY (PUBLISHER) REFERENCES PUBLISHER (ID)
220         ON DELETE CASCADE ON UPDATE CASCADE
    );

224 CREATE TABLE LFN_PROCEEDINGS
    ( PATH PATHNAME PRIMARY KEY,
226     PROCEEDINGS CODE NOT NULL,
        FOREIGN KEY (PROCEEDINGS) REFERENCES PROCEEDINGS (ID)
228         ON DELETE CASCADE ON UPDATE CASCADE
    );

```

Codice B.7: Creazione tabella Cross_Ref_Proceedings, Edited_Inproceedings, By_Inrproceedings, Published_Inproceedings e LFN_Proceedings.

```

230 CREATE TABLE EDITED_PROCEEDINGS
    ( PROCEEDINGS CODE NOT NULL,
232     AUTHOR CODE NOT NULL,
        FOREIGN KEY (PROCEEDINGS) REFERENCES PROCEEDINGS (ID)
234         ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
236         ON DELETE CASCADE ON UPDATE CASCADE
    );

CREATE TABLE PUBLISHED_PROCEEDINGS
240 ( PROCEEDINGS CODE NOT NULL,
    PUBLISHER CODE NOT NULL,
242     FOREIGN KEY (PROCEEDINGS) REFERENCES PROCEEDINGS (ID)
        ON DELETE CASCADE ON UPDATE CASCADE,
244     FOREIGN KEY (PUBLISHER) REFERENCES PUBLISHER (ID)
        ON DELETE CASCADE ON UPDATE CASCADE
246 );

248 CREATE TABLE MISC
    ( ID CODE PRIMARY KEY,
250     TITLE TITLES NOT NULL,
        YEAR YEARS,
252     HOWPUBLISHED VARCHAR (100),
        EE REFERENT_E,
254     MONTH MONTHS,
        NOTE NOTES
256 );

258 CREATE TABLE LFN_MISC
    ( PATH PATHNAME PRIMARY KEY,
260     MISC CODE NOT NULL,
        FOREIGN KEY (MISC) REFERENCES MISC (ID)
262         ON DELETE CASCADE ON UPDATE CASCADE
    );

CREATE TABLE BY_MISC
266 ( MISC CODE NOT NULL,
    AUTHOR CODE NOT NULL,
268     PRIMARY KEY (MISC,AUTHOR),
        FOREIGN KEY (MISC) REFERENCES MISC (ID)
270         ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
272         ON DELETE CASCADE ON UPDATE CASCADE
    );

```

Codice B.8: Creazione tabella Edited_Proceedings, Published_Proceedings, Misc, LFN_Misc e By_Misc.

```
274 CREATE TABLE MANUAL
( ID      CODE PRIMARY KEY,
276 TITLE  TITLES NOT NULL,
  YEAR    YEARS,
278 ADDRESS VARCHAR (100),
  EE      REFERENT_E,
280 MONTH  MONTHS,
  NOTE    NOTES
282 );

284 CREATE TABLE LFN_MANUAL
( PATH     PATHNAME PRIMARY KEY,
286 MANUAL  CODE NOT NULL,
  FOREIGN KEY (MANUAL) REFERENCES MANUAL (ID)
288         ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE BY_MANUAL
292 ( MANUAL  CODE NOT NULL,
  AUTHOR   CODE NOT NULL,
294 PRIMARY KEY (MANUAL, AUTHOR),
  FOREIGN KEY (MANUAL) REFERENCES MANUAL (ID)
296         ON DELETE CASCADE ON UPDATE CASCADE,
  FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
298         ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE PUBLISHED_MANUAL
302 ( MANUAL  CODE NOT NULL,
  PUBLISHER CODE NOT NULL,
304 FOREIGN KEY (MANUAL) REFERENCES MANUAL (ID)
         ON DELETE CASCADE ON UPDATE CASCADE,
306 FOREIGN KEY (PUBLISHER) REFERENCES PUBLISHER (ID)
         ON DELETE CASCADE ON UPDATE CASCADE
308 );
```

Codice B.9: Creazione tabella Manual, LFN_Manual, By_Manual e Published_Manual.

```

CREATE TABLE TECHREPORT
310 ( ID      CODE PRIMARY KEY,
    TITLE   TITLES NOT NULL,
312  YEAR    YEARS,
    ADDRESS  VARCHAR (100),
314  ISBN    VARCHAR (17),
    NUMBER   VARCHAR (8),
316  INSTITUTION VARCHAR (100),
    DAY      INT CHECK (DAY>=1 OR DAY <=31),
318  EE      REFERENT_E,
    MONTH    MONTHS,
320  NOTE    NOTES
    );

324 CREATE TABLE LFN_TECHREPORT
    ( PATH     PATHNAME PRIMARY KEY,
326  TECHREPORT CODE NOT NULL,
    FOREIGN KEY (TECHREPORT) REFERENCES TECHREPORT (ID)
328      ON DELETE CASCADE ON UPDATE CASCADE
    );

CREATE TABLE BY_TECHREPORT
332 ( TECHREPORT CODE NOT NULL,
    AUTHOR     CODE NOT NULL,
334  PRIMARY KEY (TECHREPORT,AUTHOR),
    FOREIGN KEY (TECHREPORT) REFERENCES TECHREPORT (ID)
336      ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
338      ON DELETE CASCADE ON UPDATE CASCADE
    );

CREATE TABLE UNPUBLISHED
342 ( ID      CODE PRIMARY KEY,
    TITLE   TITLES NOT NULL,
344  YEAR    YEARS,
    EE      REFERENT_E,
346  MONTH    MONTHS,
    NOTE    NOTES
348 );

```

Codice B.10: Creazione tabella Techreport, LFN_Techreport, By_Techreport e Unpublished.

```
CREATE TABLE LFN_UNPUBLISHED
350 ( PATH      PATHNAME PRIMARY KEY,
      UNPUBLISHED CODE NOT NULL,
352 FOREIGN KEY (UNPUBLISHED) REFERENCES UNPUBLISHED (ID)
      ON DELETE CASCADE ON UPDATE CASCADE
354 );

CREATE TABLE BY_UNPUBLISHED
356 ( UNPUBLISHED CODE NOT NULL,
358 AUTHOR      CODE NOT NULL,
      PRIMARY KEY (UNPUBLISHED,AUTHOR),
360 FOREIGN KEY (UNPUBLISHED) REFERENCES UNPUBLISHED (ID)
      ON DELETE CASCADE ON UPDATE CASCADE,
362 FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
      ON DELETE CASCADE ON UPDATE CASCADE
364 );

CREATE TABLE THESIS
366 ( ID      CODE PRIMARY KEY,
368 TITLE    TITLES NOT NULL,
      YEAR    YEARS,
370 SCHOOL   VARCHAR (100),
      ADDRESS VARCHAR (100),
372 CATEGORY TYPE_THESIS NOT NULL,
      EE     REFERENT_E,
374 MONTH    MONTHS,
      NOTE    NOTES
376 );

CREATE TABLE LFN_THESIS
378 ( PATH      PATHNAME PRIMARY KEY,
380 THESIS     CODE NOT NULL,
      FOREIGN KEY (THESIS) REFERENCES THESIS (ID)
382           ON DELETE CASCADE ON UPDATE CASCADE
      );
```

Codice B.11: Creazione tabella LFN_Unpublished, By_Unpublished, Thesis e LFN_Thesis.

```
384 CREATE TABLE BY_THESIS
( THESIS    CODE NOT NULL,
386  AUTHOR   CODE NOT NULL,
  PRIMARY KEY (THESIS,AUTHOR),
388  FOREIGN KEY (THESIS) REFERENCES THESIS (ID)
      ON DELETE CASCADE ON UPDATE CASCADE,
390  FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
      ON DELETE CASCADE ON UPDATE CASCADE
392 );

394 CREATE TABLE BY_SUPERVISOR
( THESIS    CODE NOT NULL,
396  AUTHOR   CODE NOT NULL,
  PRIMARY KEY (THESIS,AUTHOR),
398  FOREIGN KEY (THESIS) REFERENCES THESIS (ID)
      ON DELETE CASCADE ON UPDATE CASCADE,
400  FOREIGN KEY (AUTHOR) REFERENCES AUTHOR (ID)
      ON DELETE CASCADE ON UPDATE CASCADE
402 );
```

Codice B.12: Creazione tabella By_Thesis e By_Supervisor.

Ringraziamenti

Se questo percorso è giunto a termine è grazie a molte persone che mi sono state vicine.

Un Grazie a Silvia per aver creduto in me ed essermi accanto in questo lungo percorso chiamato vita. Grazie alla famiglia, mamma, papà, Valeria, Ferruccio e Pietro perché mi sostengono e mi amano per ciò che sono. Grazie ai miei amici per le risate, la compagnia, il sostegno e perché ci sono sempre.

Ringrazio tutto lo staff del gruppo IMS e in special modo l'Ing. Marco Dussin, per le utili discussioni sui temi affrontati in questa tesi e per la fedele amicizia dimostratemi in questi anni di università.

Francesco

Bibliografia

- [1] Paolo Atzeni, Stefano Ceri, Stefano Paraboschi, and Riccardo Torlone. *Basi di Dati*. McGraw-Hill, Milano, 2002.
- [2] Fisher and Hanstoc D. e T. *Come scrivere i riferimenti bibliografici*. Ed. Blackwell's, 1998.
- [3] Rocco Germano. DBLP Ontology. Master's thesis, Sapienza Università di Roma, 2006/2007.
- [4] The PostgreSQL Global Development Group. *PostgreSQL 8.4.6 Documentation*, 2009.
- [5] ICCU-ROMA. Digitalia, rivista del digitale nei beni culturali. (Numero 0), 2005.
- [6] Pantieri Lorenzo and Gordini Tommaso. *L'arte di scrivere con Latex*. Gruppo utilizzatori T_EX e L^AT_EX, Aprile 2011.
- [7] Maritan Matteo. RGB (Research Group Bibliography) un'applicazione web in php per la gestione della bibliografia di un gruppo di ricerca. Master's thesis, Università degli Studi di Padova, 2006/2007.
- [8] Patashnik Oren. Bibtexing, February 1988.
- [9] Elmastri Ramez A. and Navathe Shamkant B. *Sistemi di basi di dati. Fondamenti*. Pearson, Milano, Settembre 2004.
- [10] Elmastri Ramez A. and Navathe Shamkant B. *Sistemi di basi di dati. Complementi*. Pearson, Milano, Febbraio 2005.
- [11] Mazza Riccardo, Gaspari Roberto, and Tam Davide. *Un sistema per la gestione di riferimenti bibliografici con schede di catalogazione in XML*. ACM Italian Chapter - Special Interest Group on Applied Computing, 2001.

Sitografia

- [12] Bibliografia. <http://it.wikipedia.org/wiki/Bibliografia>.
- [13] Bibtex. <http://www.bibtex.org/>.
- [14] The dblp computer science bibliography. <http://www.informatik.uni-trier.de/~ley/db/>.
- [15] Definition editor. <http://oxforddictionaries.com/definition/editor>.
- [16] Definiton publisher. <http://oxforddictionaries.com/definition/publisher>.
- [17] Definizione riferimento bibliografico. http://win.galileivr.it/doc/radici_alii/Citazione_dei_riferimenti_bibliografici.html.
- [18] General public license. http://it.wikipedia.org/wiki/GNU_General_Public_License.
- [19] Gnu operating system. <http://www.gnu.org/licenses/>.
- [20] Google scholar. http://it.wikipedia.org/wiki/Google_Scholar.
- [21] Gpl. <http://www.megalab.it/4108/2/gpl-ed-altre-licenze-libere-cosa-e-possibile-fare-e-cosa-no>.
- [22] Information management systems (ims). <http://ims.dei.unipd.it>.
- [23] International standard book number (isbn). <http://www.isbn.it>.
- [24] Latex. <http://www.latex-project.org/>.
- [25] Lesser general public license. http://it.wikipedia.org/wiki/GNU_Lesser_General_Public_License#Differenze_dalla_GPL.
- [26] Licenza open source. http://it.wikipedia.org/wiki/Licenza_open_source.
- [27] Mendeley. <http://it.wikipedia.org/wiki/Mendeley>.
- [28] pgadmin. <http://www.pgadmin.org/>.
- [29] Postgresql. <http://www.postgresql.org>.

-
- [30] Postgresql. <http://it.wikipedia.org/wiki/PostgreSQL>.
- [31] Postgresql about. <http://www.postgresql.org/about/>.
- [32] Pubblicazione scientifica. http://it.wikipedia.org/wiki/Pubblicazione_scientifica.
- [33] the biblatex package. <http://ctan.mirror.garr.it/mirrors/CTAN/macros/latex/contrib/biblatex/doc/biblatex.pdf>.
- [34] yed graph editor. http://www.yworks.com/en/products_yed_about.html.
- [35] Rosa Marina Borraccini. Materiale di studio. http://docenti.unimc.it/docenti/rosa-marisa-borraccini/2010/storia-del-libro-e-delleditoria-2010/storia-del-libro-e-delleditoria-a.a.-2010-2011/Storia-libro-Balsamo.pdf/at_download/file.