# Information Retrieval

Dario Rigolin
Comperio srl
CTO

dario.rigolin@comperio.it

Bologna 22 Maggio 2009
Master in Tecnologie OpenSource

# Agenda

- Presentazioni di rito
- Piccola introduzione sull'IR
- Cosa offre il mercato “proprietario”
- Cosa offre l'OpenSource
- Esempi di applicazioni
- Considerazioni finali

# Presentazioni

- Dario Rigolin
  - Laurea in Scienze dell'informazione '97
    - Calcolo Parallelo e Distribuito
  - Esperienza internazionale nell'"Era .COM" :-(
  - Fondato Comperio nel 2004
  - Soluzioni per il mondo Biblioteche
  - Modello di business di tipo SaaS
  - Valutando la licenza OS più adatta dei prodotti.
- Presentatevi da soli...

# Comperio srl

- Soluzioni per il mondo delle Biblioteche
- Clavis e Discovery sono i nostri prodotti
- Web based: LAMP. (PHP5)
- Integriamo solo tecnologie OS: in vista di nostra distribuzione OS.
- Motore di IR: Solr e Zebra
- Oltre 500 biblioteche clienti...
- Mercato di nicchia e difficile...

# Information Retrieval (IR)

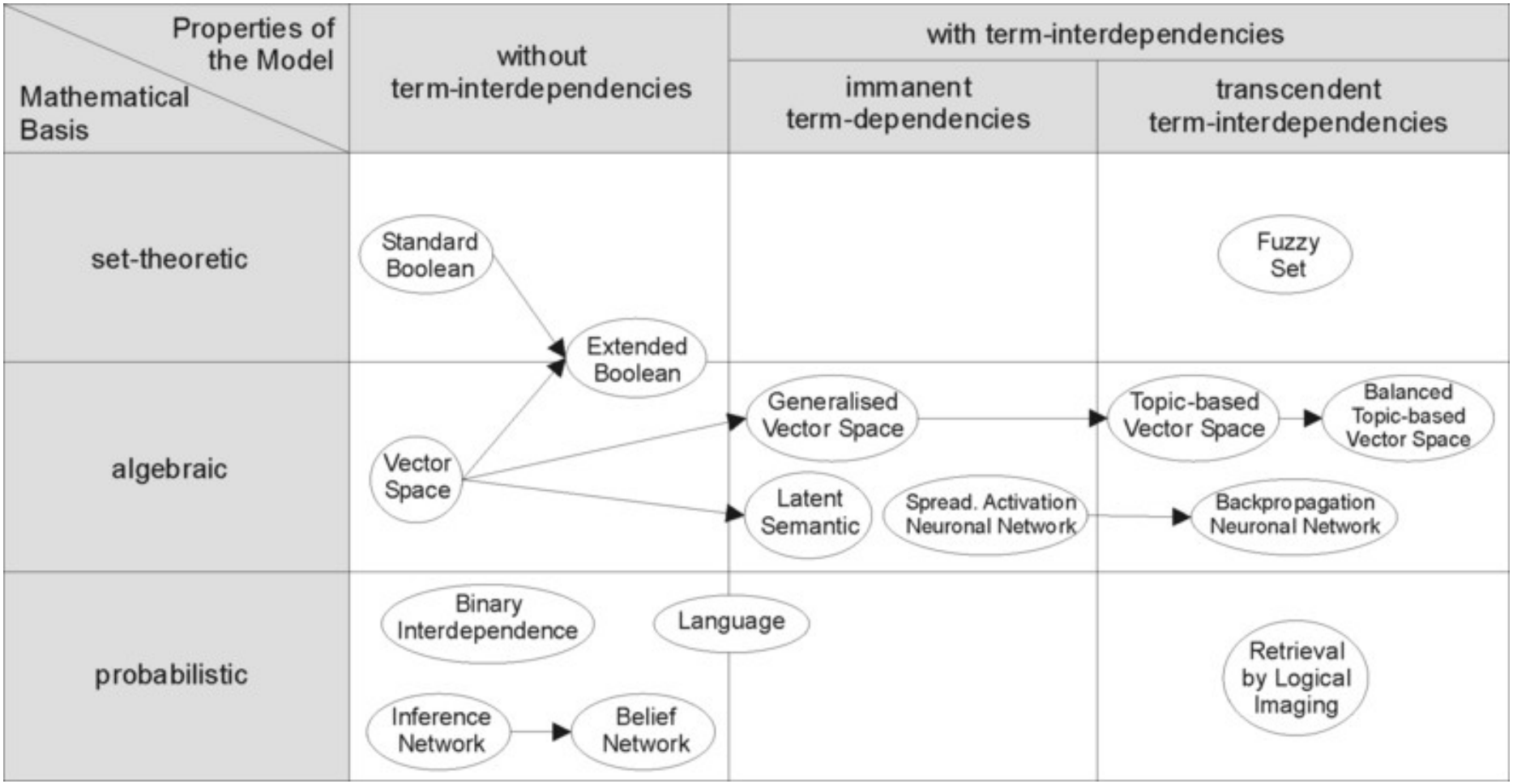
- E' un'area multidisciplinare della Computer Science: linguistica, algoritmica, semiotica...
  - Processing del linguaggio
  - Regole diverse a seconda della lingua
- Molti aspetti soggettivi: ranking...
- Ha applicazioni fantastiche nel mondo delle biblioteche... Che coincidenza...
- Wikipedia e la rete hanno moltissimo

# IR: Definizioni

- Rilevanza: come un set di documenti ritornati soddisfa la richiesta informativa dell'utente
- Pertinenza: documento che risponde alla query
- Precisione: numero di documenti pertinenti recuperati diviso i recuperati
- Recall: numero di doc rilevanti recuperati diviso il numero totale dei rilevanti

# Modelli (di query)

- Da wikipedia



# IR vs RDBMS

- Concetto di “collezioni” e non di “tabelle”.
- Non ci sono relazioni se non “link” tra documenti
- Molti RDBMS hanno IR al loro interno per consentire la ricerca sui campi di testo
- Molti sottovalutano i danni che Like ha fatto per l'IR
- La funzionalità si chiama Full text search
- Non è SQL!

# IR e i file

- IR in genere è collegato ad “indicizzare” documenti intesi come file
- PDF, Word, Excell, TXT etc...
- Quindi entra in gioco anche il “text extractor”
- Pdf2txt etc...
- Non è un problema banale...
- Le immagini e l'audio sono un'altra storia...

# IR in pratica

- Ho un elenco di “documenti”... strutturati o meno
- Li “indicizzo” nell' IR System...
- Li ricerco secondo un certo linguaggio di query
- Problemi di aggiornamento della collezione
  - Collezione statica
  - Collezione dinamica
  - Collezione MOLTO dinamica

# IR estrazione termini

- Prende il testo e ricava i “termini”... parole... numeri sigle e altro
- Una specie di normalizzazione...
- Gli accenti sono un piccolo casino...
- Stemming... Estrae radici delle parole per rendere i termini indipendenti tra plurale, singolare e genere.  Bello in teoria...
- Stop word

# IR algoritmi

- Il più diffuso è l'inverted file
  - Liste contenenti (termine, doc_id, posizione)
  - E' un concetto semplice...
  - Si implementa in genere con B-Tree e liste compresse di adiacenza...
- Vector space
  - Un bit vector con un bit per ogni termine. Ogni documento è un vettore di bit
  - Si implementa in tanti modi...

# IR e update

- Inverted file è pensato per indicizzare collezioni statiche che non cambiano spesso. Aggiornare IF è molto dispendioso. Ma si fa... se serve...
- E' importante capire come cambia la propria collezione nel tempo.
- Nel caso di documenti strutturati ho anche dei “campi”... Titolo, autore, argomento, data.

# IR e ricerca

- Modello di ricerca booleano è il più usato.
  - T1 AND t2 OR t3 etc...
- Se voglio documenti “simili tra loro”?
- Se cerco “gamba” vorrei trovare anche “arto inferiore”...
- Se cerco “gatto” vorrei trovare anche “cat or chat”...
- E' ancora un bel casino... Ma questo è il bello!

# IR e quantità

- Le informazioni crescono ed è comune indicizzare diversi GB di documenti.
- Noi abbiamo db catalografici di 10 milioni di schede... oltre 1000 campi... almeno 20 lingue...
- Poi la gente vuole dei “thriller” e non li trova!
- I bibliotecari fanno IR da centinaia di anni.
- Bibliotecari ed informatici si dovrebbero parlare di più...

# Protocolli

- Z39.50 – IR Distribuito usa RPN (Reverse Polish Notation)... Notazione Polacca Inversa!
  - Find @and @attr 1=4 nome @attr 1=32 eco
- CQL: Common Query Language.
  - dc.title any fish sortBy dc.date/sort.ascending
- SRU: Search Retrieval via URL
  - Usa CQL e XML
  - Successore di Z39.50

# Software IR Commerciali

- DtSearch (IR)
- Fast Search (IR)
- Oracle InterMedia (RDBMS)
- Miscrosoft SQL Server (Text search) RDBMS
- IBM DB2 (Multimedia) RDBMS
- Endeca (IR) - FANTASTICO!

# Software IR OS

- ht://Dig (GPL)
- Lucene (Apache)
- Solr (Apache)
- Zebra (GPL-Commercial)
- MySQL / PostgreSQL – Full text extension
- Sphinx Search (GPL-Commercial)
- Mnogosearch (GPL)
- E tanti altri piccoli e meno diffusi...

# ht://Dig

- Detto “lo Sbadilatore” di testi
- Basato su inverted file
- Pensato per indicizzare pagine Web
- Motore di ricerca per Intranet e domini web
- Nato “pre google”...
- Architettura: Spider, indexer e retriever
- API per vari linguaggi

# MySQL / PostgreSQL

- Full text search su campi testo di tabelle
- Linguaggio di interrogazione proprietario
- Stopword e poco altro
- Problemi di performace oltre 800K righe
- Storage, dimensioni dei file indice
- Pratico e veloce per evitare “LIKE”
- Integrato nel DB server

# mnoGoSearch

- Pensato per indicizzare WebSite.
- Spider e indexer
- Backend su RDBMS... Inverted File
- Un backend su SleepycatDB (ora Oracle)
- API per molti linguaggi diversi. PHP!
- Simile ad ht://Dig

# Sphinx

- Full text - Inverted File
- Molto performante multi field
- Integrabile con MySQL / Postgres
- GPL – Supporto Commerciale
- Stemming / C++
- PHP interface

# Lucene

- Java based inverted File API
- Molto sofisticato e ampiamente sviluppato
- Stemmer, indexer, unaccent stopword etc...
- Usato in tantissimi altri progetti.
- Esiste anche Clucene riscritto in C++
- Supporta campi
- API anche in PHP!!!

# Solr

- Usato da noi...
- Basato su Lucene aggiunge un'interfaccia XML
- Consente di avere un index server con definizione di uno schema del documento
- Supporta Facets, ranking e altro
- Tutti i plugin e moduli di Lucene
- Necessita di App server J2EE
- Usa molta memoria

# Zebra

- Usato da noi...Prodotto da IndexData(Dan)
- Pensato per le biblioteche ma evoluto
- Non tutte le cose di Lucene/Solr tipo stemmer
- Vari backend di indicizzazione per formati MARC e XML
- Interfaccia Z39.50 e SRU
- Performance di indicizzazione molto alte
- Shadow register e commit

# Come scegliere

- Che formato di documenti dobbiamo gestire
- Quanti sono e di che tipo. Quante lingue, quanti campi
- C'è un gestionale relazionale di mezzo
- Dove risiedono i documenti e come cambiano
- Ci serve supporto
- Performance di ricerca richieste
- Che linguaggio dobbiamo integrare / web?