

**IMPLEMENTASI SPARQL UNTUK PEMBUATAN APLIKASI  
PENCARIAN ARTIKEL WIKIPEDIA**

**SKRIPSI**



Oleh

**Yohanes Septian**  
**22074299**

**Program Studi Teknik Informatika Fakultas Teknologi Informasi**

**Universitas Kristen Duta Wacana**

**2011**

**IMPLEMENTASI SPARQL UNTUK PEMBUATAN APLIKASI  
PENCARIAN ARTIKEL WIKIPEDIA**

**SKRIPSI**



Diajukan kepada Fakultas Teknologi Informasi Informatika  
Universitas Kristen Duta Wacana  
Sebagai salah satu syarat dalam memperoleh gelar  
Sarjana Komputer



Disusun Oleh :

**Yohanes Septian**

**22074299**

**Program Studi Teknik Informatika Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana  
2011**

## PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa tugas akhir dengan judul :

**Implementasi SPARQL Untuk Pembuatan Aplikasi  
Pencarian Artikel Wikipedia**

Yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan sarjana Program Studi Teknik Informatika, Fakultas Teknik Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika kemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar kesarjanaaan saya.

Yogyakarta, 10 oktober 2011



( Yohanes Septian  
22074299 )



## HALAMAN PERSETUJUAN

Judul : Implementasi SPARQL Untuk Pembuatan Aplikasi  
Pencarian Artikel Wikipedia  
Nama : Yohanes Septian  
NIM : 22074299  
Mata Kuliah : Tugas Akhir  
Kode : TI2126  
Semester : Gasal  
Tahun Akademik : 2011/2012

Telah diperiksa dan disetujui  
Di Yogyakarta,  
Pada Tanggal 22 September 2011

Dosen Pembimbing I



Budi Susanto, S.Kom, M.T.

Dosen Pembimbing II



Antonius Rachmat, S.Kom, M.Cs

# HALAMAN PENGESAHAN

SKRIPSI

IMPLEMENTASI SPARQL UNTUK PEMBUATAN APLIKASI  
PENCARIAN ARTIKEL WIKIPEDIA

Oleh : Yohanes Septian / 22074299

Dipertahankan di depan dewan Penguji Tugas Akhir/Skripsi  
Program Studi Teknik Informatika Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana – Yogyakarta  
Dan dinyatakan diterima untuk memenuhi salah satu  
Syarat memperoleh gelar  
Sarjana Komputer  
Pada tanggal  
10-10-2011

Yogyakarta, 3-11-2011  
Mengesahkan,

Dewan Penguji :

1. Budi Susanto, S.Kom., M.T.
2. Antonius Rachmat C, S.Kom., M.Cs.
3. Lucia Dwi Krisnawati, S.S., M.A.
4. Ir. Sri Suwarno, M.Eng.



Dekan

  
(Drs. Wimmie Handiwidjojo, MIT)

Ketua Program Studi

  
(Nugroho Agus Haryono S.Si.,MSi.)

## UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa yang telah melimpahkan rahmat dan anugerah, sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul Implementasi SPARQL Untuk Pembuatan Aplikasi Pencarian Artikel Wikipedia.

Penulisan laporan ini merupakan kelengkapan dan pemenuhan dari salah satu syarat dalam memperoleh gelar Sarjana Komputer. Selain itu bertujuan melatih mahasiswa untuk dapat menghasilkan suatu karya yang dapat dipertanggungjawabkan secara ilmiah, sehingga dapat bermanfaat bagi penggunanya.

Dalam menyelesaikan pembuatan program dan laporan Tugas Akhir ini, penulis telah banyak menerima bimbingan, saran, dan masukan dari berbagai pihak, baik secara langsung maupun secara tidak langsung. Untuk itu dengan segala kerendahan hati, pada kesempatan ini penulis menyampaikan ucapan terimakasih kepada :

1. Bpk. Budi Susanto, S.Kom, M.T. selaku pembimbing I yang telah memberikan bimbingannya dengan sabar dan baik kepada penulis, juga kepada
2. Bpk. Antonius Rachmat, S.Kom, M.Cs. selaku dosen pembimbing II atas bimbingannya, petunjuk, dan masukan yang diberikan selama pengerjaan tugas akhir ini sejak awal hingga akhir, juga kepada
3. Dosen-dosen Universitas Kristen Duta Wacana yang telah membantu memberikan pengarahan dan masukan kepada penulis.
4. Keluarga tercinta yang memberi dukungan dan semangat.
5. Fanny Pritami Widodo, Evarisma Wulandari, dan Danny Sebastian, Christian Irawan yang selalu memberikan dukungan dan semangat.

6. Sahabat dan teman-teman kost yang telah memberikan masukan dan semangat.
7. Pihak lain yang tidak dapat penulis sebutkan satu per satu, sehingga Tugas Akhir ini dapat terselesaikan dengan baik.

Penulis menyadari bahwa program dan laporan Tugas Akhir ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun dari pembaca sekalian. Sehingga suatu saat penulis dapat memberikan karya yang lebih baik lagi.

Akhir kata penulis ingin meminta maaf bila ada kesalahan baik dalam penyusunan laporan maupun yang pernah penulis lakukan sewaktu membuat program Tugas Akhir. Sekali lagi penulis memohon maaf yang sebesar-besarnya. Dan semoga dapat berguna bagi kita semua.

Yogyakarta,

Penulis

## INTISARI

### IMPLEMENTASI SPARQL UNTUK PEMBUATAN APLIKASI Pencarian ARTIKEL WIKIPEDIA

Semakin hari semakin data artikel pada Wikipedia semakin bertambah. Dengan jumlah data yang sangat banyak dan layanan yang diberikan semakin luas menyebabkan semakin meningkatnya tantangan dalam melakukan pencarian data pada jumlah artikel yang sangat banyak. Data tersimpan dalam bentuk struktur *semantic web* dapat menjadi salah satu solusi untuk melakukan penyimpanan data dengan jumlah yang besar, namun terdapat kendala pada saat melakukan pencarian data.

Untuk mempermudah proses pencarian data, penulis membangun sebuah sistem pencarian artikel yang dapat melakukan penyimpanan data secara lokal, sehingga pencarian data akan dilakukan dengan lebih cepat dan efisien. Penulis menggunakan *SPARQL* untuk melakukan proses pengambilan data dari DBpedia yang menyediakan data artikel pada Wikipedia dengan struktur *semantic web*. Setelah mendapatkan data secara lokal, proses pencarian artikel dan *pe-rankin*-an dilakukan dengan metode *Vector Space Model*. Pengujian dilakukan dengan memberikan beberapa uji masukan pada sistem dan dievaluasi hasil yang diberikan oleh sistem.

Sistem yang dibangun telah dapat menyimpan data yang didapat dari DBpedia secara lokal. Sistem dapat melakukan pencarian data artikel Wikipedia sesuai dengan relevansi dengan masukan terhadap sistem.



## DAFTAR ISI

HALAMAN JUDUL .....	
PERNYATAAN KEASLIAN SKRIPSI .....	i
HALAMAN PERSETUJUAN .....	ii
HALAMAN PENGESAHAN .....	iii
UCAPAN TERIMA KASIH .....	iv
INTISARI .....	vi
DAFTAR ISI .....	vii
DAFTAR TABEL .....	xi
DAFTAR GAMBAR .....	xii
<b>BAB 1 PENDAHULUAN .....</b>	
1.1 Latar Belakang Masalah .....	1
1.2 Perumusan Masalah .....	3
1.3 Batasan Masalah .....	3
1.4 Hipotesis .....	4
1.5 Tujuan Penelitian .....	4
1.6 Metode Penelitian .....	5
1.7 Sistematika Penulisan .....	5
<b>BAB 2 TINJAUAN PUSTAKA .....</b>	
2.1 Tinjauan Pustaka .....	7
2.2 Landasan Teori .....	8

2.2.1	DBpedia.....	8
2.2.2	RDF ( <i>Resource Description Framework</i> ).....	9
2.2.3	SPARQL.....	11
2.2.4	Information Retrieval.....	12
2.2.5	Indexing.....	14
2.2.6	VSM ( <i>Vector Space Model</i> ).....	14
<b>BAB 3 ANALISIS DAN PERANCANGAN SISTEM.....</b>		
3.1	Kebutuhan <i>Hardware</i> dan <i>Software</i> .....	16
3.1.1	Kebutuhan Sisi Server.....	16
3.1.2	Kebutuhan Sisi Client.....	17
3.2	Diagram <i>Use Case</i> .....	18
3.3	Spesifikasi Sistem.....	19
3.4	Arsitektur Sistem.....	20
3.5	Algoritma dan <i>Flowchart</i> .....	20
3.5.1	<i>Query Parsing</i> .....	21
3.5.2	<i>Indexing</i> .....	22
3.5.3	Pe-ranking-an dengan metode <i>Vector Space Model</i> .....	24
3.6	Kamus Data.....	25
3.6.1	Tabel <i>person</i> .....	25
3.6.2	Tabel <i>token</i> .....	26
3.6.3	Tabel <i>auto_property</i> .....	26
3.6.4	Tabel <i>property</i> .....	26
3.6.5	Tabel <i>person_has_property</i> .....	27

3.6.6	Tabel stopwords.....	28
3.6.7	Tabel keyword_frequency.....	28
3.6.8	Tabel term_frequency.....	28
3.7	Diagram Skema.....	29
3.8	Rancangan Antarmuka Sistem.....	30
3.8.1	<i>Site Map</i> .....	30
3.8.2	Rancangan Antarmuka.....	31
3.8.2.1	Rancangan Antarmuka Halaman Awal.....	31
3.8.2.2	Rancangan Antarmuka Halaman Hasil Pencarian.....	32
3.8.2.3	Rancangan Antarmuka Halaman Daftar Hasil Crawling.....	33
3.9	Rancangan Pengujian Sistem.....	33
<b>BAB 4 IMPLEMENTASI DAN ANALISIS SISTEM.....</b>		
4.1	Antarmuka Sistem.....	35
4.1.1	Halaman home.....	35
4.1.2	Halaman Crawler.....	36
4.1.3	Halaman Daftar Hasil Pencarian Artikel.....	37
4.2	Implementasi Sistem.....	38
4.2.1	Pengumpulan Artikel.....	38
4.2.2	Pra-pemrosesan Teks.....	39
4.2.3	Pencarian Artikel.....	43
4.3	Evaluasi Sistem.....	45
<b>BAB 5 KESIMPULAN DAN SARAN.....</b>		
5.1	Kesimpulan.....	55

5.2	Saran.....	55
	DAFTAR PUSTAKA.....	57
	LAMPIRAN	

© UKDW

## DAFTAR TABEL

Tabel 3.1	Tabel person .....	25
Tabel 3.2	Tabel token .....	26
Tabel 3.3	Tabel auto_property .....	26
Tabel 3.4	Tabel property .....	27
Tabel 3.5	Tabel person_has_property .....	27
Tabel 3.6	Tabel stopwords .....	28
Tabel 3.7	Tabel keyword_frequency .....	28
Tabel 3.8	Tabel term_frequency .....	29
Tabel 3.9	Tabel Contoh Masukan Sistem .....	34
Tabel 4.1	Tabel Evaluasi Sistem .....	45
Tabel 4.2	Tabel precision dan recall hasil pengujian .....	53



## DAFTAR GAMBAR

Gambar 2.1	Representasi RDF-graf .....	9
Gambar 2.2	Contoh RDF/XML .....	10
Gambar 2.3	Struktur query SPARQL .....	12
Gambar 3.1	Diagram <i>Use Case</i> .....	18
Gambar 3.2	Arsitektur Sistem .....	20
Gambar 3.3	<i>Flowchart</i> Query Parsing .....	22
Gambar 3.4	<i>Flowchart</i> Indexing .....	23
Gambar 3.5	<i>Flowchart</i> metode VSM .....	24
Gambar 3.6	Diagram Skema .....	29
Gambar 3.7	<i>Site Map</i> Sistem .....	30
Gambar 3.8	Rancangan Antarmuka Halaman Awal .....	31
Gambar 3.9	Rancangan Antarmuka Halaman Hasil Pencarian .....	32
Gambar 3.10	Rancangan Antarmuka Halaman Daftar Hasil Crawling .....	33
Gambar 4.1	Halaman Home .....	35
Gambar 4.2	Halaman Crawler .....	36
Gambar 4.3	Halaman Daftar Hasil Pencarian Artikel .....	37
Gambar 4.4	<i>Pseudocode</i> proses pembentukan query SPARQL .....	39
Gambar 4.5	<i>Pseudocode</i> proses pencatatan data person dan properti .....	40
Gambar 4.6	<i>Pseudocode</i> proses tokenisasi dan perhitungan $tf$ .....	41
Gambar 4.7	<i>Pseudocode</i> proses perhitungan $idf$ dan pembobotan .....	42
Gambar 4.8	<i>Pseudocode</i> proses pencarian artikel .....	43

Gambar 4.9	<i>Pseudocode</i> proses perankingan dengan metode <i>VSM</i> .....	44
Gambar 4.10	Hasil evaluasi sistem “american or indonesian journalist” .....	48
Gambar 4.11	Hasil evaluasi sistem “age:25 or tv reporter” bagian 1 .....	49
Gambar 4.12	Hasil evaluasi sistem “age:25 or tv reporter” bagian 2 .....	50
Gambar 4.13	Hasil evaluasi sistem “age:24 or age:25 and actor or singer” .....	51
Gambar 4.14	Hasil evaluasi sistem “gender:female and occupation:journalist” ....	52
Gambar 4.15	Hasil evaluasi sistem “american and editor” .....	53
Gambar 4.16	Grafik hasil pengujian sistem .....	54



## Bab 1

### PENDAHULUAN

#### 1.1. Latar Belakang Masalah

Wikipedia merupakan sebuah situs yang sangat sering dikunjungi dan mengalami revisi yang cukup konstan. Menurut alexa.com<sup>[1]</sup>. Wikipedia menduduki peringkat ke-8 di dunia berdasarkan *traffic* selama tiga bulan terakhir. Layanan yang diberikan oleh Wikipedia terdapat lebih dari 250 bahasa, dan sampai saat ini memiliki lebih dari 3,5 juta artikel dengan lebih dari 23 juta halaman. Dengan jumlah data dan layanan yang diberikan, tentunya Wikipedia mengalami suatu tantangan dalam kemampuan untuk melakukan pencarian terhadap data yang mereka miliki. Saat ini mesin pencari yang dimiliki oleh Wikipedia belum dapat melakukan pencarian dengan *metadata* khususnya yang terdapat pada *infobox*.

*Infobox* adalah sebuah tabel yang didesain khusus di ujung kanan atas pada suatu artikel yang terdapat pada Wikipedia yang menyediakan ringkasan dari artikel yang bersangkutan. Data yang disediakan di *infobox* antara lain adalah klasifikasi makhluk hidup, nama, tanggal lahir, ketertarikan terhadap suatu bidang dan berbagai macam informasi yang biasanya dimiliki oleh banyak subyek. *Infobox* juga bertujuan agar pembaca artikel dapat melakukan perbandingan antara beberapa subyek dengan cepat.

<sup>[1]</sup> alexa.com saat ini merupakan *web crawler* terbesar yang ada.

( <http://www.alexacom/company> )



Data yang terdapat di Wikipedia saat ini tersimpan dalam bentuk pengetahuan terstruktur dalam bentuk RDF (DBpedia). Data tersebut tersimpan dengan model *semantic web*. Dengan model *semantic web* tersebut, DBpedia dapat menerima *query* yang dapat menghasilkan data-data yang saling terkait satu dengan yang lain. Untuk dapat melakukan *query* pada data berbentuk RDF, dibutuhkan SPARQL yang merupakan *query language* untuk *semantic web*.

Pemilihan *query* yang kurang spesifik akan menghasilkan pencarian artikel yang tidak relevan dengan yang diinginkan. Hasil pencarian dengan SPARQL dalam peringkat awal belum tentu relevan. Berdasarkan hasil tersebut, dapat disimpulkan bahwa *information retrieval* dengan menggunakan *query* biasa tidaklah efektif. Permasalahan yang muncul adalah bagaimana sistem dapat menghasilkan pencarian artikel yang relevan untuk memenuhi kebutuhan pencarian?

Salah satu metode yang dapat digunakan untuk menghitung relevansi pada suatu sistem *information retrieval* adalah *Vector Space Model*. Penulis akan merancang sebuah sistem *information retrieval* dengan menggunakan *engine* SPARQL pada DBpedia, yang kemudian akan diurutkan tingkat relevansinya dengan menggunakan *Vector Space Model*. Diharapkan dengan menggunakan SPARQL dan *Vector Space Model*, Sistem dapat membantu melakukan pencarian artikel pada Wikipedia sampai dengan pencarian dengan menggunakan *metadata*.

## 1.2. Perumusan Masalah

Berdasarkan latar belakang masalah di atas, penulis akan merancang dan membangun sebuah sistem dengan memanfaatkan *engine* SPARQL yang dapat melakukan *information retrieval* (termasuk pada metada *infobox*) pada DBpedia sehingga dapat menghasilkan informasi artikel yang dibutuhkan. Masalah yang akan diteliti oleh penulis adalah seberapa besar *recall* dan *precision* yang dihasilkan oleh sistem *information retrieval* yang menggunakan SPARQL untuk pencarian artikel Wikipedia yang memiliki data terstruktur dan juga *metadata* yang disediakan oleh DBpedia.

## 1.3. Batasan Masalah

Untuk menghindari kesalahpahaman dan meluasnya pokok bahasan, penulis memberikan batasan masalah sebagai berikut :

- Sistem yang dibangun berupa aplikasi *web* yang akan diuji pada jaringan yang memiliki koneksi Internet.
- Data yang digunakan adalah data abstrak dan *infobox* Wikipedia.
- Sistem yang akan dibangun hanya untuk fasilitas pencarian artikel yang dapat melakukan pencarian termasuk pada *metadata* Wikipedia.
- Bentuk keluaran dari sistem adalah berupa artikel yang memiliki relevansi dengan masukan sistem.
- Data yang akan digunakan untuk penelitian ini akan dibatasi hanya pada kelas *person*. Data tersebut diambil dari abstrak dan *infobox* yang telah disediakan oleh DBpedia.
- Atribut data yang dianggap sebagai *metadata* adalah semua atribut yang ada pada *infobox* DBpedia yang merupakan kelas *person*.
- Proses pembobotan dilakukan dengan *Tf-Idf*.

- Masukan untuk sistem dapat mengandung operasi relasi AND dan/atau OR, namun tidak dapat melakukan proses masukan beberapa kata yang dianggap kesatuan.
- Masukan untuk sistem dapat berupa metadata saja, *keyword* abstrak saja, atau metadata dan *keyword* abstrak.

#### 1.4. Hipotesis

Pencarian dengan menggunakan SPARQL yang didukung dengan penggunaan atribut *metadata* akan menghasilkan pencarian yang lebih akurat dibandingkan pencarian data yang menggunakan *query* biasa, ditinjau dari hasil *precision* dan *recall*, yang akan dilakukan dengan metode *Vector Space Model*.

#### 1.5. Tujuan Penelitian

Tujuan dari penelitian ini adalah menerapkan penggunaan SPARQL untuk melakukan *information retrieval* artikel Wikipedia yang dapat melakukan pencarian terhadap *metadata* secara dinamis.

#### 1.6. Metodologi Penelitian

Metodologi yang akan digunakan oleh penulis dalam penelitian ini adalah sebagai berikut :

- Penulis mempelajari bagaimana struktur data yang terdapat pada DBpedia.
- Penulis mempelajari bagaimana penggunaan SPARQL yang merupakan *query language* yang digunakan untuk *semantic web*.
- Penulis merancang suatu fungsi yang dapat melakukan proses *parsing* terhadap *query* yang menjadi masukan kedalam sistem sehingga dapat menjadi bentuk SPARQL.
- Penulis merancang proses yang akan melakukan *indexing* terhadap hasil *query* SPARQL yang didapat dari DBpedia.
- Penulis mempelajari pembuatan modul *information retrieval* yang didalamnya termasuk proses *pe-ranking-an*, dimana akan menghasilkan tingkat relevansi setiap artikel yang relevan terhadap masukan sistem.
- Penulis akan melakukan pengujian terhadap sistem.

### 1.7. Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini akan dibagi menjadi beberapa bab, yaitu :

Bab 1 Pendahuluan, berfungsi untuk memberikan gambaran umum tentang penelitian yang dilakukan oleh penulis. Pada bagian pendahuluan ini berisi tentang latar belakang masalah, perumusan masalah, batasan masalah, hipotesis, tujuan penelitian, metodologi penelitian, dan sistematika penulisan.

Bab 2 Tinjauan Pustaka, yang berisi 2 bagian utama, yaitu tinjauan pustaka dan landasan teori. Bagian tinjauan pustaka akan menguraikan teori yang didapatkan dari berbagai sumber pustaka yang penulis gunakan dalam

penelitian ini. Untuk landasan teori, akan memuat penjelasan tentang konsep dan prinsip utama yang digunakan dalam pemecahan masalah.

Bab 3 Perancangan Sistem, bab ini berisi mengenai kebutuhan *hardware* dan *software* minimum yang digunakan penulis dan dibutuhkan pengguna, spesifikasi sistem yang akan dibuat, arsitektur sistem, diagram *use case*, algoritma dan *flowchart*, kamus data, diagram skema, rancangan antarmuka sistem dan rancangan pengujian terhadap sistem.

Bab 4 Implementasi dan Analisis Sistem, berisi pembahasan implementasi dan pengujian sistem yang ada pada bab 3, beserta hasil *print-screen* dan hasil analisis dari sistem yang dibuat.

Bab 5 Kesimpulan dan Saran, merupakan bagian kesimpulan dari hasil penelitian yang dilakukan, dan juga berisi saran untuk riset agar dapat memberikan hasil yang lebih baik lagi.



## BAB 5

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Berdasarkan sistem yang telah dibuat oleh penulis maka didapatkan kesimpulan sebagai berikut :

1. Pencarian menggunakan kata kunci metadata cenderung menghasilkan artikel relevan dengan nilai *precision* dan *recall* yang lebih tinggi.
2. Membuat struktur property pada metadata menjadikan pencarian lebih akurat.
3. Pembuatan *endpoint* lokal dibutuhkan agar *query* SPARQL dapat dilakukan dengan batasan *transaction timed-out* dan batasan jumlah hasil yang lebih bebas.

#### 5.2. Saran

Sistem yang dibuat oleh penulis dapat dikembangkan lebih lanjut agar dapat dihasilkan sistem yang lebih optimal dan memiliki *performance* lebih bagus. Saran yang diajukan penulis untuk mengembangkan sistem adalah :

1. Pembuatan *endpoint* lokal, sehingga konfigurasi server dapat dilakukan sendiri dan bisa mengatur batasan yang digunakan untuk membatasi *transaction time-out* dan jumlah hasil *query* SPARQL.
2. Proses *indexing* dapat ditambahkan dengan proses *stemming* agar dicapai hasil pencarian yang lebih akurat.
3. Penggunaan *store procedure* pada *query* SQL dan *query* SPARQL untuk mempercepat proses pencarian yang dilakukan.

## Daftar Pustaka

- [1] Auer S., Bizer C., Kobilarov G., Lehmann J., Cyganiak R., & Ives Z. (2008). *DBpedia: A Nucleus for a Web of Open Data*. Proceedings of the 6<sup>th</sup> International Semantic Web Conference (ISWC).
- [2] Castillo R. (2010). *Indexing RDF Data for SPARQL*. Humboldt University.
- [3] Feigenbaum L. (2009) . *SPARQL by Example a Tutorial* , Cambridge Semantics . Diakses pada 17 Januari 2011, dari <http://www.cambridgesemantics.com/2008/09/sparql-by-example/>
- [4] Herman I. (2008). *SPARQL is a Recommendation*. Diakses pada 18 Januari 2011, dari [http://www.w3.org/blog/SW/2008/10/15/sparql\\_is\\_a\\_recommentadion](http://www.w3.org/blog/SW/2008/10/15/sparql_is_a_recommentadion)
- [5] Hartig O., Bizer C., & Freytag J.C. (2010). *Executing SPARQL Queries Over the Web of Linked Data*. Berlin University.
- [6] Manning C.D., Raghavan P., & schutze H. (2009). *An Introduction to Information Retrieval*. England: Cambridge University Press.
- [7] Manola F., & Miller E. (2004). *RDF Primer*, Recommendation, World Wide Web Consortium (W3C). Diakses pada 10 Februari 2011, dari <http://www.w3.org/TR/rdf-primer/>
- [8] Matthews A. (2008). *Understanding SPARQL*. IBM Corporation
- [9] Stadler C., Martin M., Lehmann J., & Hellmann S.. (2010). *Update Strategies for DBpedia Live*. Germany.