

# Perbandingan Kinerja *Data Manipulation Language* MongoDB dan SQL Server

Andi Wahyu Rahardjo Emanuel dan Jimmy Sentosa

**Abstract** — *Database is the main element of data storing in this information technology era. With the increasing volumes of data being created in unstructured way, the new streams of database called NoSQL emerges challenging the traditional RDBMS system using SQL syntax. This paper compares the Data Manipulation Language performance of document-oriented NoSQL database which is MongoDB against the popular SQL Server database in CRUD operations (Create, Read, Updating, and Delete) of data in identical windows based application. The result shows that both databases has similar syntax, but SQL server is twice as fast during insert, delete and read while MongoDB is better during update. It is also found that designing database in MongoDB is simpler eventhough the database size in SQL server is more efficient.*

**Keywords** - *NoSQL, MongoDB, SQL Server, Data Manipulation Language, comparative study*

## 1. PENDAHULUAN

Data merupakan sesuatu yang sangat penting dalam penggunaan sehingga dibutuhkan suatu media penyimpanan khusus untuk menyimpan agar data tersebut dapat tersimpan dengan baik. Media penyimpanan tersebut dikenal dengan sebutan basis data atau *database*. Pada tahun 2012 penggunaan *database* masih di dominasi oleh sistem basis data yaitu RDBMS (*Relational Database Management System*) yang berbasis SQL yang pada awalnya sistem tersebut dirancang pada tahun 1970 oleh Edgar F. Codd di IBM [1][5]. Bentuk basisdata berbasis SQL ini mengandalkan penyimpanan data secara terstruktur yang semakin lama semakin kurang relevan dengan perkembangan yang ada.

Masalah yang dihadapi saat ini adalah bahwa RDBMS sebagai media penyimpanan/*database* tidak sepenuhnya dapat menjawab permasalahan penyimpanan untuk data yang semakin tidak terstruktur. Semakin banyak bentuk – bentuk data

baru yang sifatnya tidak terstruktur yang perlu disimpan kurang bisa dijawab dengan database relasional yang ada. Dokumen seperti berkas HTML, pengolah kata, multimedia dan lain – lain memerlukan tipe penyimpanan baru yang kemudian berkembang menjadi penyimpanan tidak relasional atau *NoSQL database* misalnya *Dynamo* oleh Amazon, *Big Table* oleh Google, *CouchDB* oleh Apache dan masih banyak lagi alternatif lainnya [1]. Tipe basisdata ini memiliki kelebihan dalam hal skalabilitas dan ketersediaan meskipun mengorbankan beberapa aspek yang biasanya dimiliki RDBMS yaitu aspek ACID (*Atomicity, Consistency, Isolation, Durability*) yang lebih rileks[2].

Pada paper ini akan dibandingkan kinerja dari sebuah basis data *NoSQL* yaitu *MongoDB* dibandingkan dengan salah satu basis data tipe RDBMS populer yaitu *SQL Server* dari *Microsoft Corporation*. *MongoDB* adalah salah satu *NoSQL* yang berorientasi dokumen yang dikembangkan secara kode terbuka (*Open Source*) oleh *10gen* [2][3][4][5]. Beberapa karakteristik penting dari *MongoDB* adalah penyimpanan data dalam *collections* dan masing – masing *collections* memiliki *documents* yang diserialisasi dengan menggunakan *BSON (Binary JSON)* [3]. Perbandingan antara *MongoDB* dan *SQL Server* pernah dilakukan dalam lingkungan sistem pengambilan keputusan dan layanan data secara interaktif yang mengatakan sisi kinerja *SQL Server* lebih unggul meskipun aspek lainnya *NoSQL* memiliki keunggulan yang berbeda [3]. Perbedaan antara berbagai *NoSQL* berbasis dokumen termasuk *MongoDB* dibandingkan jenis lain juga pernah diulas oleh Kaur dkk [4]. Aspek keamanan dari *MongoDB* yang masih banyak kekurangan juga pernah menjadi bahan penelitian [6]. *MongoDB* dan berbagai jenis *NoSQL* lainnya juga menjadi subyek studi komparasi [7]. Penelitian ini berbeda dibandingkan dengan penelitian sebelumnya dimana aspek yang dibandingkan adalah aspek koneksi, desain, dan kinerja dari *Data Manipulation Language* dari *MongoDB* apabila dibandingkan dengan *SQL*

Andi Wahyu Rahardjo Emanuel, Prodi S1 Teknik Informatika, Fakultas Teknologi Informasi UK. Maranatha, Jl. Prof. drg. Suria Sumantri no. 65, Bandung 40164 ([andi.wre@itmaranatha.org](mailto:andi.wre@itmaranatha.org)).

Jimmy Sentosa, Prodi S1 Teknik Informatika, Fakultas Teknologi Informasi UK. Maranatha, Jl. Prof. drg. Suria Sumantri no. 65, Bandung 40164 ([sentosa.jimmy@yahoo.com](mailto:sentosa.jimmy@yahoo.com)).

Server dengan menggunakan aplikasi yang identik.

## 2. METODE PENELITIAN

Aspek – aspek yang dijadikan acuan perbandingan dari kedua basis data ini adalah aspek *Data Manipulation Language*, yaitu operasi *CRUD* (*Create, Read, Update, Delete*). Perbandingan dari database – database tersebut dipisahkan menjadi 3 bagian, yaitu:

- Perbedaan *syntax DML* untuk *MongoDB* dan *SQL Server*.
- Perbedaan desain basisdata untuk *MongoDB* dan *SQL Server*.
- Perbedaan kinerja *CRUD* antara *SQL Server* dan *MongoDB*

Proses komparasi dilakukan dengan cara membuat dua buah aplikasi Sistem Informasi Presensi pada sebuah Rumah Sakit skala menengah di daerah Bandung. Aplikasi ini dikembangkan dengan *.NET Framework* dengan perbedaan utama pada kedua aplikasi ini adalah pada basisdata yang dipergunakan yaitu dengan menggunakan *MongoDB* dan *SQL Server*. Adapun perangkat keras yang dipergunakan adalah:

- *Processor: Intel Core 2 Duo*
- *RAM: 2 Giga Byte*
- *Harddisk: 40 Giga Byte*
- *Monitor: Beresolusi 1024 × 768*

Adapun data yang akan dipergunakan untuk perbandingan yaitu data pegawai yang mencakup 1 *field string autonumber* (NIK), 2 *field string* (Nama, Alamat), 1 *field integer* (Umur), 1 *field DateTime* (Tgl\_Awal), 1 *field decimal* (kekayaan), dan 1 *field byte[]* (photo) \* size: 2.06Kb. Pengujian akan dilakukan dengan melakukan *CRUD* (*Create, Read, Update, Delete*) secara berulang pada jumlah data 10, 20, 40, 80, 160, 320, 640, 1280.

## 3. HASIL DAN PEMBAHASAN

### 3. 1. Perbedaan Syntax Data Manipulation Language

*Data Manipulating Language* (DML) adalah seperangkat perintah yang dimiliki oleh *database* yang meliputi *Create* atau *Insert*, *Update*, *Delete*, serta *Read* atau *Select*. Perbedaan *syntax* antara *MongoDB* akan dianalisa dengan perbandingan dengan *syntax* pada RDBMS pembanding yaitu *SQL Server*.

Berikut ini adalah *syntax* perintah *DML* dalam *Microsoft SQL Server*:

#### 1. Insert

```
INSERT INTO [table] [kolom] VALUES [value]
```

#### 2. Update

```
UPDATE [table] SET column name = value where [condition]
```

#### 3. Delete

```
DELETE FROM [table] where [condition]
```

#### 4. Select

```
SELECT [column] from [table] where [condition]
```

Berikut ini adalah *syntax* perintah *DML* dalam *MongoDB*:

#### 1. Insert

```
server.GetDatabase("[db]").GetCollection("[table"]  
).Insert(object)
```

#### 2. Update

```
server.GetDatabase("[db]").GetCollection("[table"]  
).Update(Query, Update.Set("field"  
,object.ToBsonDocument))
```

#### 3. Delete

```
server.GetDatabase("[db]").GetCollection("[table"]  
).Delete(Query)
```

#### 4. Select

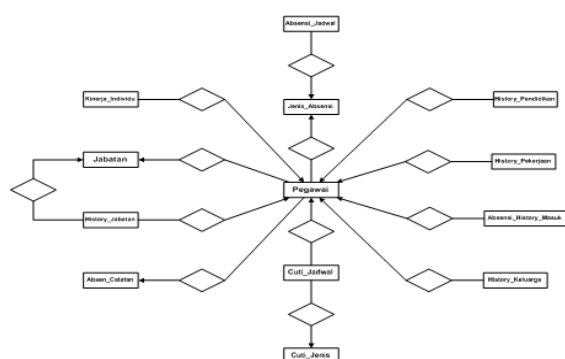
```
server.GetDatabase("[db]").GetCollection("[table"]  
).Find(Query)
```

Dapat dilihat dari *syntax* diatas bahwa proses *query* dengan menggunakan *MongoDB* memiliki kemiripan dengan *query* dengan menggunakan sistem *SQL Server*.

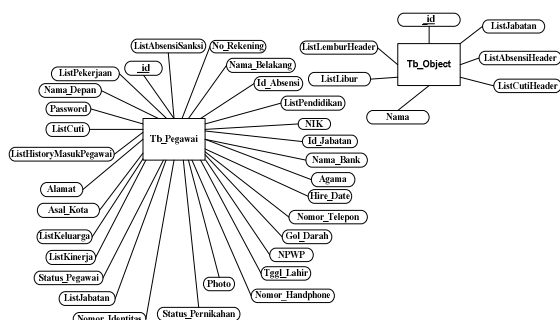
### 3. 2. Perbedaan Desain MongoDB dan SQL Server

Aplikasi yang dikembangkan adalah sebuah aplikasi Sistem Informasi Presensi untuk sebuah rumah sakit skala menengah di daerah Bandung. Kedua aplikasi tersebut dibuat identik dalam semua aspek kecuali dalam penggunaan basis data dimana satu aplikasi menggunakan *SQL Server* dan aplikasi yang lainnya menggunakan *MongoDB*.

Pada implementasi aplikasi dengan menggunakan basis data *SQL Server* perlu dirancang sebuah diagram relasi entitas (*Entity Relationship Diagram*) yang terdiri dari 13 tabel seperti yang ditunjukkan pada gambar 1. Sedangkan pada implementasi aplikasi dengan menggunakan *MongoDB* memerlukan 2 buah koleksi data seperti yang ditunjukkan pada gambar 2. Dari gambar 1 dan gambar 2 dapat dilihat bahwa penerapan *MongoDB* memiliki keunggulan dalam sisi kesederhanaan desain karena tidak memerlukan perancangan tabel.



### Gambar 1 ERD dalam Penerapan di SQL Server



**Gambar 2 Desain Document dalam Penerapan di MongoDB**

### 3.3. Pengujian Kinerja DML MongoDB dan SQL Server

Dengan menggunakan set data untuk aplikasi presensi yang meliputi beberapa data integer, desimal dan sebuah foto seperti yang dijelaskan di bagian 2, kinerja *Data Manipulation Language* dari kedua basis data dibandingkan, yaitu dalam aspek CRUD (*Create, Read, Update, dan Delete*) dengan hasil dari masing – masing aspeknya adalah sebagai berikut.

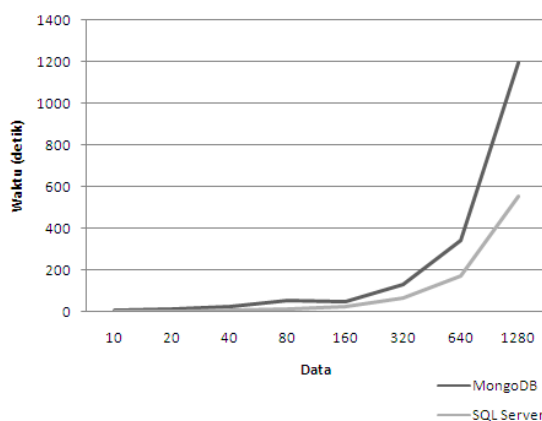
### 3. 3.1 Perbandingan Perintah Insert

Perbandingan perintah *insert* data pada *MongoDB* dan *SQL Server* ditunjukkan pada tabel 1 berikut (satuan dari hasil dalam detik).

**Tabel 1 Perbandingan perintah *insert* MongoDB vs SQL Server**

Data	MongoDB	SQL Server
10	7,10	1,48
20	14,43	2,81
40	27,34	5,73
80	54,65	12,17
160	50,13	27,59
320	130,92	68,65
640	341,03	171,24
1280	1193,05	554,30

Berdasarkan tabel 1 maka dibuat gambar 3 sebagai berikut (garis lebih rendah menunjukkan lebih baik):



**Gambar 1 Perbandingan perintah *insert* MongoDB vs SQL Server**

Gambar 1 diatas menunjukkan bahwa pada perintah *insert* dalam *SQL Server* lebih cepat sekitar dua kali lipat daripada perintah *insert* dalam *MongoDB*.

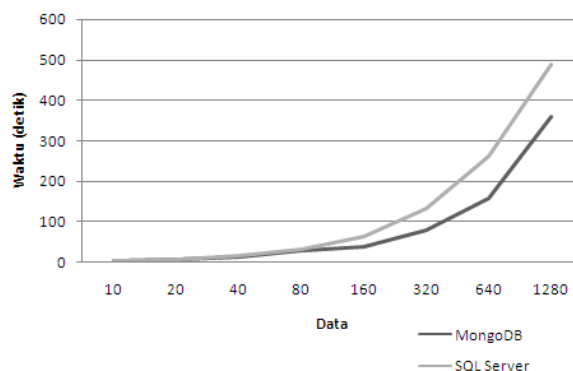
### 3. 3.2 Perbandingan Perintah Update

Perbandingan perintah *update* data pada *MongoDB* dan *SQL Server* ditunjukkan pada tabel 2 berikut (satuan dari hasil dalam detik).

**Tabel 2 Perbandingan perintah *update* MongoDB vs SQL Server**

Data	MongoDB	SQL Server
10	4,38	4,19
20	7,37	8,43
40	15,29	16,34
80	30,24	32,67
160	40,49	65,31
320	80,37	131,74
640	159,72	262,58
1280	359,66	488,34

Berdasarkan tabel 2 maka dibuat gambar 4 sebagai berikut ( garis lebih rendah menunjukkan lebih baik):



**Gambar 4 Perbandingan perintah *update* MongoDB vs SQL Server**

Gambar 4 diatas menunjukkan bahwa pada perintah *update* dalam *SQL Server* lebih lambat daripada perintah *update* dalam *MongoDB*.

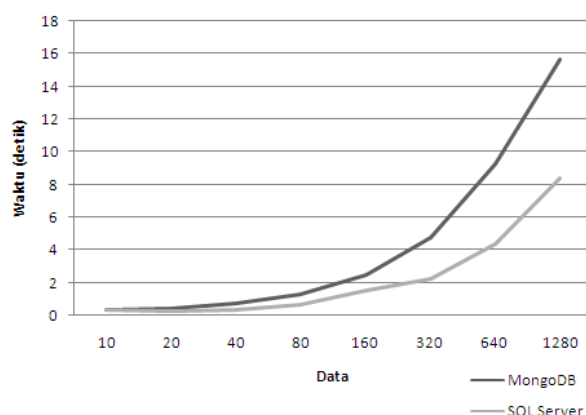
### 3.3.3 Perbandingan Perintah Delete

Perbandingan perintah *delete* data pada *MongoDB* dan *SQL Server* ditunjukkan pada tabel 3 berikut (satuan dari hasil dalam detik).

**Tabel 3 Perbandingan perintah *delete* *MongoDB* vs *SQL Server***

Data	MongoDB	SQL Server
10	0,35	0,31
20	0,43	0,23
40	0,73	0,34
80	1,33	0,60
160	2,50	1,51
320	4,80	2,23
640	9,23	4,37
1280	15,66	8,42

Berdasarkan tabel 3 maka dibuat gambar 5 sebagai berikut (garis lebih rendah menunjukkan lebih baik).



**Gambar 5 Perbandingan perintah *delete* *MongoDB* vs *SQL Server***

Gambar 5 diatas menunjukkan bahwa pada perintah *delete* dalam *SQL Server* lebih cepat sekitar dua kali lipat daripada perintah *delete* dalam *MongoDB*.

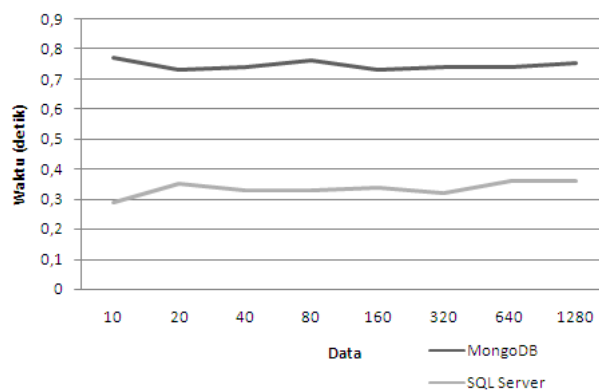
### 3.3.4 Perbandingan Perintah Read

Perbandingan perintah *read* data pada *MongoDB* dan *SQL Server* ditunjukkan pada tabel 4 berikut (satuan dari hasil dalam detik).

**Tabel 4 Perbandingan perintah *read* *MongoDB* vs *SQL Server***

Data	MongoDB	SQL Server
10	0,77	0,29
20	0,73	0,35
40	0,74	0,33
80	0,76	0,33
160	0,73	0,34
320	0,74	0,32
640	0,74	0,36
1280	0,75	0,36

Berdasarkan tabel 4 maka dibuat gambar 6 sebagai berikut (garis lebih rendah menunjukkan lebih baik).



**Gambar 6 Perbandingan perintah *read* *MongoDB* vs *SQL Server***

Gambar 6 diatas menunjukkan pada perintah *read* dalam *SQL Server* lebih cepat sekitar dua kali lipat daripada *read* dalam *MongoDB*.

### 3.3.4 Perbandingan Ukuran Basisdata

Aspek lainnya yang ditemukan selama perbandingan adalah ukuran dari basis data pada kedua database. Gambar 7 menunjukkan ukuran basis data untuk 1280 data pada *MongoDB* sedangkan gambar 8 menunjukkan ukuran basis data untuk jumlah data yang sama untuk *SQL Server*.

Db_BranchMark0	19-May-13 7:37 PM	0 File	65,536 KB
Db_BranchMark1	19-May-13 7:37 PM	1 File	131,072 KB
Db_BranchMark2	22-May-13 11:24 P...	2 File	262,144 KB
Db_BranchMarkns	19-May-13 7:37 PM	NS File	16,384 KB

**Gambar 7 Ukuran database *MongoDB***

Db_BranchMaking_log.LDF	24-May-13 7:58 AM	SQL Server Databa...	768 KB
Db_BranchMaking.mdf	24-May-13 7:58 AM	SQL Server Databa...	57,600 KB

**Gambar 8 Ukuran database *SQL Server***

Gambar 7 dan 8 dapat diketahui ukuran total ukuran basisdata *MongoDB* adalah 393Mb sedangkan pada jumlah data yang sama, ukuran basisdata pada *SQL Server* sebesar 58MB. Penggunaan ukuran basisdata di *SQL Server* lebih efisien sekitar tujuh kali lipat daripada ukuran basis data di *MongoDB*.

## 4. KESIMPULAN

Dari berbagai macam pengukuran dan observasi dalam membandingkan antara *MongoDB* dan *SQL Server*, dapat diambil kesimpulan bahwa:

- *Syntax* untuk melakukan perintah – perintah *CRUD* memiliki kemiripan antara *MongoDB* dan *SQL Server*.

- Desain basisdata untuk *MongoDB* lebih sederhana daripada menggunakan *SQL Server* karena tidak membutuhkan desain ERD.
- Kinerja operasi – operasi *CRUD* untuk *SQL Server* lebih cepat sekitar dua kali lipat dibandingkan dengan *MongoDB* untuk perintah *insert*, *delete*, dan *read*, sedangkan *MongoDB* lebih cepat daripada *SQL Server* hanya untuk perintah *update*.
- Ukuran basisdata untuk *SQL Server* lebih efisien sekitar tujuh kali lipat dibandingkan dengan *MongoDB*.

## 5. SARAN

Beberapa kajian lanjutan dapat dilakukan yang berkaitan dengan perbandingan kinerja dari basis data *MongoDB* dan *SQL Server* ini. Data yang dipergunakan bisa dibuat menjadi lebih beragam lagi untuk dapat menggambarkan situasi penyimpanan data yang lebih nyata. Perbandingan antara *MongoDB* dengan basisdata berbasis *NoSQL* lainnya juga dapat dilakukan untuk dapat menemukan sistem *NoSQL* yang lebih baik.

## UCAPAN TERIMA KASIH

Terima kasih kepada Universitas Kristen Maranatha, khususnya Program Studi S1 Teknik Informatika Fakultas Teknologi Informasi (<http://www.itmaranatha.org>) yang telah memberikan dukungan dalam penelitian ini.

## DAFTAR PUSTAKA

- [1] Leavit N., 2010, "Will NoSQL Databases Live Up to Their Promise?", Computer, IEEE Computer Society, 2010, pp 12 – 14.
- [2] Cattell R., 2010, "Scalable SQL and NoSQL Data Stores", SIGMOD Record, Vol. 39 No. 4, Desember 2010, pp 12 – 27
- [3] Floratou A., Teletia N., DeWitt D.J., Patel J.M., dan Zhang D., 2012, "Can the Elephant Handle the NoSQL Onslaught?", Prosiding dari VLDB Endowment, Vol. 5 No. 12, pp 1712 – 1723.
- [4] Kaur J., Kaur H., Kaur K., 2013, "A Review on Document Oriented and Column Oriented Databases", International Journal of Computer Trends and Technology, Vol. 4 Issue 3, 2013, pp 338 – 344.
- [5] Strauch C., 2013, "NoSQL Databases", Lecture Notes Selected Topics of Software Technology Ultra Large Scale Sites, University Hochschule der Medien, Stuttgart, xxxx, tersedia: <https://oak.cs.ucla.edu/cs144/handouts/nosql dbs.pdf>, diakses: 22 Agustus 2013
- [6] Okman L., Gal-Oz N., Gonen Y., Gudes E., dan Abramov J., 2011, "Security Issues in NoSQL Databases", Prosiding dari 2011 International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-11, 2011, pp 541 – 547.
- [7] Tauro C.J.M., Aravindh S., dan Shreeharsha A.B., 2012, "Comparative Study of the New Generation, Agile, Scalable, High Performance NOSQL Databases", International Journal of Computer Applications, Vol. 48 No. 20, June 2012, pp 1 – 4.