

SKALABILITAS DAN KINERJA DALAM BIG DATA

Mata Kuliah Infrastruktur dan Teknologi Big Data

Oleh: Galih Hermawan, S.Kom., M.T.
Program Studi Teknik Informatika. FTIK.
Universitas Komputer Indonesia

Pengantar Skalabilitas

- ❖ Definisi Skalabilitas
 - Kemampuan sistem untuk menangani pertumbuhan beban kerja atau data.
- ❖ Pentingnya Skalabilitas dalam Big Data
 - Dengan pertumbuhan volume data yang cepat, skalabilitas menjadi kunci untuk memastikan kinerja yang optimal.

Skalabilitas Horizontal

- ❖ Paralelisasi dan Distribusi Sumber Daya
 - Proses mendistribusikan tugas ke beberapa sumber daya secara bersamaan.
 - Memastikan beban kerja terdistribusi merata.
- ❖ Keuntungan:
 - Kemampuan untuk menangani peningkatan beban secara linier.
- ❖ Contoh Implementasi:
 - Arsitektur Cluster dengan Apache Hadoop.

Skalabilitas Vertikal

- ❖ Meningkatkan Kapasitas pada Sumber Daya Tunggal
 - Peningkatan kapasitas pada satu sumber daya tunggal, seperti server.
- ❖ Keuntungan:
 - Maksimalkan kinerja pada satu node.
- ❖ Contoh Implementasi:
 - Peningkatan RAM pada server utama.

Penerapan Skalabilitas Horizontal

- ❖ Arsitektur Cluster untuk Distribusi Beban
 - Kumpulan server atau node yang bekerja bersama untuk menangani tugas-tugas secara bersamaan.
- ❖ Keuntungan:
 - distribusi beban kerja yang efisien,
 - menangani pertumbuhan volume data dengan mudah.

Teknologi Pendukung: Hadoop dan Apache Spark

❖ Hadoop

- Framework sumber terbuka untuk pemrosesan dan penyimpanan data terdistribusi.
- Peran dalam Skalabilitas Horizontal: memungkinkan pemrosesan data dalam skala besar secara terdistribusi.

❖ Apache Spark

- Sistem komputasi cepat dan umum untuk analisis data.
- Peran dalam Skalabilitas Horizontal: proses data secara paralel untuk meningkatkan kinerja.

Penerapan Skalabilitas Vertikal

- ❖ Skalabilitas Vertikal
 - Peningkatan kapasitas pada satu sumber daya tunggal, seperti server atau node.
- ❖ Keuntungan Skalabilitas Vertikal
 - Maksimalkan kinerja pada satu node.
 - Cocok untuk beban kerja yang membutuhkan kapasitas tinggi.
 - Pengelolaan yang lebih sederhana dibandingkan dengan skala horizontal.

Teknologi Pendukung: Memcached dan Redis

- ❖ Memcached
 - ❖ Sistem manajemen memori terdistribusi untuk mempercepat akses data.
 - ❖ Peran dalam Skalabilitas Vertikal: meningkatkan kinerja aplikasi dengan menyimpan data di dalam memori.
- ❖ Redis
 - ❖ Basis data berkinerja tinggi berbasis memori.
 - ❖ Peran dalam Skalabilitas Vertikal: menyediakan penyimpanan data yang cepat dan efisien.

Mengelola Pertumbuhan Volume Data

Tantangan Pertumbuhan Volume Data

- ❖ Pertumbuhan volume data yang pesat merupakan salah satu tantangan yang dihadapi oleh organisasi di era digital. Pertumbuhan data ini disebabkan oleh berbagai faktor, antara lain:
 - Peningkatan penggunaan perangkat digital
 - Peningkatan jumlah transaksi bisnis
 - Peningkatan penggunaan data sensor
 - Peningkatan penggunaan data media sosial
- ❖ Pertumbuhan volume data yang pesat dapat menimbulkan berbagai tantangan, antara lain:
 - Kenaikan biaya penyimpanan dan pemrosesan data
 - Kebutuhan infrastruktur data yang lebih besar
 - Kesulitan dalam mengelola dan menganalisis data

Teknik Kompressi Data

- ❖ Kompresi data adalah proses mengurangi ukuran data tanpa mengurangi informasi yang terkandung di dalamnya. Teknik kompresi data dapat digunakan untuk mengatasi tantangan pertumbuhan volume data.
- ❖ Ada beberapa jenis teknik kompresi data, antara lain:
 - Kompresi tanpa kehilangan (*lossless compression*), yang mempertahankan integritas data asli setelah proses kompresi. Artinya, setelah data dikompresi dan kemudian dikembalikan ke bentuk semula, tidak ada informasi yang hilang. Teknik kompresi tanpa kehilangan biasanya digunakan untuk data yang memerlukan pemulihan yang akurat, seperti file teks, basis data, dan file sistem. Beberapa teknik yang digunakan pada kompresi data tanpa kehilangan antara lain adalah Run Length Encoding (RLE), Lempel Ziv - Welch (LZW), Huffman Coding, dan Arithmetic Coding.
 - Kompresi dengan kehilangan (*lossy compression*), yang menghasilkan data yang berbeda dari data asli setelah proses kompresi. Artinya, setelah data dikompresi dan kemudian dikembalikan ke bentuk semula, ada informasi yang hilang. Teknik kompresi dengan kehilangan biasanya digunakan untuk data yang tidak memerlukan pemulihan yang sempurna, seperti gambar, audio, dan video. Beberapa teknik yang digunakan pada kompresi data dengan kehilangan antara lain adalah JPEG, MP3, MPEG, dan H.264.

Data Sharding untuk Distribusi Efisien

- ❖ *Data sharding* adalah teknik membagi data menjadi beberapa bagian yang lebih kecil. Teknik ini dapat digunakan untuk meningkatkan skalabilitas sistem data.
- ❖ *Data sharding* dapat dilakukan dengan berbagai cara, antara lain:
 - *Sharding* berdasarkan kunci partisi (*partition key sharding*), yang membagi data berdasarkan nilai dari atribut tertentu, seperti ID pelanggan, kode pos, atau tanggal. Metode ini memungkinkan akses cepat ke data yang berhubungan dengan kunci partisi, tetapi dapat menyebabkan ketidakseimbangan beban jika ada kunci partisi yang lebih populer daripada yang lain.
 - *Sharding* berdasarkan rentang (*range sharding*), yang membagi data berdasarkan rentang nilai dari atribut tertentu, seperti harga, umur, atau skor. Metode ini memungkinkan pencarian dan pengurutan data yang efisien, tetapi dapat menyebabkan hotspot jika ada rentang yang lebih sering diakses daripada yang lain.
 - *Sharding* berdasarkan *hash* (*hash sharding*), yang membagi data berdasarkan hasil fungsi hash dari atribut tertentu, seperti ID pelanggan, email, atau nama. Metode ini memungkinkan distribusi data yang merata dan acak, tetapi dapat menyulitkan pencarian dan pengurutan data.

Studi Kasus: Strategi Mengelola Pertumbuhan Volume Data

Berikut ini adalah contoh strategi mengelola pertumbuhan volume data:

- ❖ Menggunakan teknik kompresi data

Perusahaan dapat menggunakan teknik kompresi data untuk mengurangi ukuran data tanpa mengurangi informasi yang terkandung di dalamnya. Hal ini dapat membantu mengurangi biaya penyimpanan dan pemrosesan data.

- ❖ Melakukan data *sharding*

Perusahaan dapat melakukan data *sharding* untuk meningkatkan skalabilitas sistem data. Hal ini dapat membantu mengurangi kebutuhan infrastruktur data yang lebih besar.

- ❖ Menggunakan *cloud computing*

Perusahaan dapat menggunakan *cloud computing* untuk mengelola datanya. *Cloud computing* menawarkan berbagai keuntungan, termasuk skalabilitas yang mudah dan cepat, serta biaya yang lebih efisien.

Studi Kasus (lanjutan)

- ❖ Berikut ini adalah contoh strategi pengelolaan pertumbuhan volume data yang diterapkan oleh perusahaan e-commerce:
 - Perusahaan menggunakan teknik kompresi data untuk mengurangi ukuran data transaksi dan data pelanggan. Hal ini membantu mengurangi biaya penyimpanan dan pemrosesan data.
 - Perusahaan melakukan data *sharding* berdasarkan nilai kunci transaksi. Hal ini membantu meningkatkan skalabilitas sistem pemrosesan transaksi.
 - Perusahaan menggunakan *cloud computing* untuk menyimpan dan memproses datanya. Hal ini membantu perusahaan untuk meningkatkan skalabilitas dan efisiensi biaya.
- ❖ Strategi pengelolaan pertumbuhan volume data yang tepat akan tergantung pada kebutuhan dan persyaratan spesifik dari suatu organisasi.