Penerbit
Gita Lentera

Pengantar BIG DATA

Dr. Erdisna, S. Kom., M. Kom. | M.Zuhri Halim, S. Kom., M. Kom.
Karno Diantoro, S. Kom, M. Kom. | Ahmad Soderi, S. Kom, MM.
Ardian Fachreza, S.T, M. Kom | Weni Kurnia Sari, S.ST., M. Biomed.
Desi anggreani, S. Kom.,M.T.
Dr. Ir. Indriyani, S. Kom., M. Kom. | Ir. Anwar T. Sitorus, M. Kom.

PENGANTAR BIG DATA

Penulis:
Dr. Erdisna, S. Kom., M. Kom
Zuhri Halim, S. Kom., M. Kom
Karno Diantoro, S. Kom, M. Kom
Ahmad Soderi, S. Kom, MM.
Ardian Fachreza, S.T, M. Kom
Weni Kurnia Sari, S.ST., M. Biomed
Desi anggreani, S. Kom.,M.T.
Dr. Ir. Indriyani, S. Kom., M. Kom.
Ir. Anwar T. Sitorus, M. Kom

Penerbit:
CV. Gita Lentera
One Step to Publish your Ideas

PENGANTAR BIG DATA

Penulis:
Dr. Erdisna, S. Kom., M. Kom
Zuhri Halim, S. Kom., M. Kom
Karno Diantoro, S. Kom, M. Kom
Ahmad Soderi, S. Kom, MM.
Ardian Fachreza, S.T, M. Kom
Weni Kurnia Sari, S.ST., M. Biomed
Desi anggreani, S. Kom.,M.T.
Dr. Ir. Indriyani, S. Kom., M. Kom.
Ir. Anwar T. Sitorus, M. Kom

Editor:
Januardi Nasir,S.Kom,M.Kom.

Proof Reader:
Dr. M. Oky Fardian Gafari, S.Sos.,M.Hum

ISBN 978-634-7237-08-8

Design Cover:
Maya Uswanti

Layout:
Adnan, S.H., M.H.

CV. Gita Lentera
Redaksi:
Perm. Permata hijau regency blok F/1 kelurahan Pisang kecamatan Pauh kota Padang, Sumatera Barat
https://gitalentera.com / git4lenter4@gmail.com
Anggota IKAPI

Cetakan Pertama: Mei 2024

Sinopsis

Buku Pengantar Big Data merupakan panduan komprehensif yang dirancang untuk membantu pembaca memahami konsep dasar, arsitektur, serta penerapan teknologi big data di berbagai bidang. Ditujukan bagi mahasiswa, peneliti, praktisi IT, maupun siapa saja yang ingin memahami dunia data besar, buku ini menyajikan materi secara sistematis dan mudah dipahami.

Pembaca akan diperkenalkan pada karakteristik big data yang dikenal dengan istilah 5V (Volume, Velocity, Variety, Veracity, dan Value), serta bagaimana data dalam jumlah besar dan kompleks tersebut dikelola, diproses, dan dianalisis untuk menghasilkan informasi yang bernilai. Buku ini juga membahas berbagai platform big data seperti Hadoop dan Spark, serta konsep-konsep penting seperti NoSQL, data warehouse, dan data lake.

Dengan dilengkapi studi kasus dan aplikasi nyata dari big data dalam bidang bisnis, kesehatan, pemerintahan, hingga media sosial, buku ini memberikan gambaran praktis bagaimana big data menjadi aset strategis di era digital saat ini.

Kata Pengantar

Segala puji dan Syukur bagi Tuhan Yang Maha Esa atas Rahmatdan karunia-Nya, Sehingga penulis dapat menyelesaikan buku yang berjudul "**Pengantar Big data"**. ini dapat diselesaikan dengan baik. Buku ini disusun sebagai kontribusi dalam memberikan pemahaman dasar mengenai konsep, teknologi, serta implementasi big data yang saat ini berkembang pesat dan semakin relevan dalam berbagai aspek kehidupan, khususnya di era transformasi digital.

Kemajuan pesat dalam bidang teknologi informasi telah menyebabkan akumulasi data dalam volume yang sangat besar, beragam, dan terus meningkat secara cepat. Fenomena ini memunculkan tantangan sekaligus peluang baru dalam hal penyimpanan, pengolahan, serta analisis data. Oleh karena itu, pemahaman terhadap konsep big data menjadi suatu kebutuhan yang esensial bagi berbagai kalangan, baik akademisi maupun praktisi.

Dengan pendekatan yang sistematis dan penyampaian yang komunikatif, buku ini diharapkan dapat menjadi referensi awal yang bermanfaat dalam mendalami bidang big data.

Padang, Mei 2025

Tim Penuli

Kata Sambutan

Assalamualaikum Warrohmatullahi Wabarakatuh
Salam sejahtera bagi kita semua.

Puji syukur kita panjatkan ke hadirat Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga buku yang berjudul "Pengantar Big Data" ini dapat disusun dan diterbitkan dengan baik. Kehadiran buku ini merupakan kontribusi yang sangat berharga dalam memperkaya literatur di bidang teknologi informasi, khususnya dalam topik yang semakin krusial di era digital, yaitu big data.

Perkembangan teknologi yang begitu pesat telah membawa dampak signifikan dalam berbagai sektor kehidupan. Data menjadi komoditas penting yang mendorong inovasi dan pengambilan keputusan berbasis informasi yang akurat. Oleh karena itu, pemahaman terhadap konsep, metodologi, dan teknologi big data sangat penting, tidak hanya bagi akademisi dan mahasiswa, tetapi juga bagi para praktisi di dunia industri dan pemerintahan.

Kami menyambut baik dan memberikan apresiasi atas terbitnya buku ini, yang telah disusun secara sistematis dan komunikatif untuk memperkenalkan big data kepada pembaca dari berbagai latar belakang. Dengan pendekatan yang aplikatif dan dilengkapi dengan contoh nyata, buku ini diharapkan mampu

menjadi referensi awal yang bermanfaat serta mendorong lahirnya kajian dan riset lanjutan di bidang big data.

Akhir kata, kami mengucapkan selamat kepada penulis atas selesainya penyusunan buku ini. Semoga buku ini dapat memberikan kontribusi nyata dalam pengembangan ilmu pengetahuan dan teknologi serta menjadi sumber inspirasi bagi para pembaca.

Wassalamu'alaikum warahmatullahi wabarakatuh.

Padang, Mei 2025

Tim Penulis

Daftar isi

Sinopsis iii

Kata Pengantar iv

Kata Sambutan v

BAB 1: Pengantar Big Data 1

1.1 Apa itu Big Data 1

1.2 Karakteristik Big Data 3

1.3 Aplikasi BIG DATA 7

1.4 Alat yang digunakan dalam BIG DATA 18

1.5 Tantangan dalam BIG DATA 24

BAB 2: Fondasi untuk Big Data 35

2.1 Apa itu Sistem Berkas? 36

2.2 Apa itu Sistem Berkas Terdistribusi? 36

2.3 Komputasi Skalabel Melalui Internet Apa itu Skalabilitas? 41

2.3.1 Era Komputasi Internet 42

2.3.2 Komputasi Kinerja Tinggi 44

2.4 Model populer untuk big data 47

2.5 Lima Alasan Anda Memerlukan Pendekatan Langkah demi Langkah untuk Orkestrasi Alur Kerja untuk Big Data 53

BAB 3: Model Data 65

3.1 Apa itu format data? 65

3.2 Apa itu Model Data 67

3.3 Manfaat Model dan Lingkungan Penyimpanan yang Tepat untuk Big Data 68

3.4 Apa itu data mart? 71

3.5 Berbagai jenis data mart 72

3.6 Apa Arti Streaming Big Data? 75

3.8 Aliran Data 75
3.9 Kasus Penggunaan untuk Data Real-Time dan Streaming 77
3.10 Danau Data 80
3.11 DataLake vs.Gudang Data 82
3.12 Elemen penting dari solusi Data Lake dan Analytics 83
3.13 Nilai dari Data Lake 84
3.14 Tantangan Data Lakes 85
3.15 Streaming data sensor 86
3.16 Penggunaan data sensor 89
BAB 4: Manajemen NOSQL 97
4.1 Apa itu Basis Data Relasional 97
4.2 Perbedaan antara RDBMS dan NOSQL 100
4.3 Jenis-jenis Database NOSQL 101
4.4 Model Data 109
4.5 Pengantar NOSQL 110
4.6 Model Relasional vs Model Data Agregat 111
BAB 05: Pengantar Hadoop 125
5.1 Manfaat Hadoop untuk Big Data 126
5.2 Komponen Tambahan Ekosistem Hadoop 127
5.3 Komponen Lainnya 129
5.4 Perangkat lunak sumber terbuka yang terkait dengan Hadoop 130
BAB 6: Administrasi Hadoop 144
6.1 HDFS 144
6.2 Arsitektur HDFS 145
BAB 7: Arsitektur Hadoop 153
7.1 Apa itu Hadoop Distribute File System (HDFS) 153
7.2 Komponen Hadoop 154
7.3 Rangkum Cara Kerja Hadoop Secara Internal 162

7.4 Gugus Hadoop....163
7.5 Apa itu Hadoop High Availability?....164
7.6 Arsitektur HDFS....164
BAB 8: Analisis Data dengan R....172
8.1 Metode Pembelajaran Mesin....172
8.2 Tantangan Pembelajaran Terbimbing....176
8.3 Algoritma Pembelajaran Mesin Penguatan....180
8.4 Karakteristik Pembelajaran Penguatan....183
8.5 Model Pembelajaran Penguatan....184
BAB 9: Manajemen Big Data menggunakan Splunk....195
9.1 Kategori Produk....195
9.2 Antarmuka SPLUNK....197
9.3 Penyerapan Data....199
9.4 Mengunggah Data....201
9.5 Aplikasi Pencarian & Pelaporan....207
9.6 Pencarian Splunk-Field....211

BAB 1: Pengantar Big Data

Dr. Erdisna, S.Kom., M.Kom

Tujuan

Setelah mempelajari unit ini, Anda akan dapat:

- memahami apa itu BIG DATA.
- Memahami Aplikasi BIG DATA
- pelajari alat yang digunakan dalam BIG DATA
- tantangan yang diketahui dalam BIG DATA

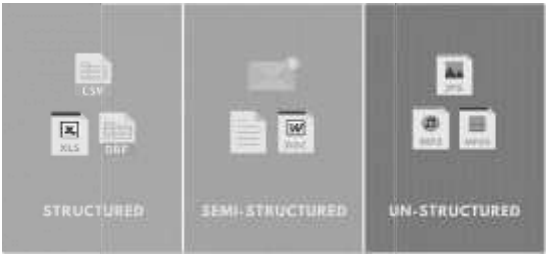
Perkenalan

Jumlah data yang dibuat oleh manusia meningkat dengan cepat setiap tahun sebagai akibat dari diperkenalkannya teknologi baru, gadget, dan saluran komunikasi seperti situs jejaring sosial. Big data adalah sekelompok kumpulan data besar yang tidak dapat ditangani dengan metode komputer biasa. Big data bukan lagi teknik atau alat tunggal; melainkan telah berkembang menjadi subjek komprehensif yang mencakup berbagai alat, teknik, dan kerangka kerja. Besaran, huruf, atau simbol yang digunakan komputer untuk melakukan operasi dan yang dapat disimpan dan dikomunikasikan sebagai sinyal listrik dan direkam pada media magnetik, optik, atau mekanis.

1.1 Apa itu Big Data

Big Data adalah kumpulan data besar yang terus bertambah secara drastis dari waktu ke waktu. Big Data adalah kumpulan data yang sangat besar dan rumit sehingga tidak ada teknologi manajemen data yang dapat menyimpan atau memprosesnya secara efektif(Varudharajulu & Ma,

2018). Big Data mirip dengan data biasa, kecuali ukurannya yang jauh lebih besar. Analisis Big Data adalah penggunaan teknik analisis tingkat lanjut untuk kumpulan data yang sangat besar dan heterogen, yang dapat berisi data terstruktur, semi-terstruktur, dan tidak terstruktur, serta data dari berbagai sumber dan ukuran mulai dari terabyte hingga zettabyte(Majumdar et al., 2017).



Gambar 1 *Terstruktur, Semiterstruktur dan Tidak Terstruktur*

Big data adalah istilah yang mendefinisikan sejumlah besar data terorganisasi dan tidak terstruktur yang ditemui perusahaan setiap hari(Santoso, 2020).

Catatan

- Hal ini dapat dipelajari untuk mendapatkan wawasan yang mengarah pada pilihan bisnis dan pergerakan strategis yang lebih baik.
- Ini adalah kumpulan data terorganisasi, semi-terstruktur, dan tidak terstruktur yang dapat ditambang untuk informasi dan digunakan dalam pembelajaran mesin, pemodelan prediktif, dan inisiatif analitik tingkat lanjut lainnya.

Contoh Big Data

Gambar 2 menunjukkan contoh big data. Setiap hari, lebih dari 500 terabyte data baru diserap ke dalam sistem Facebook. Informasi ini

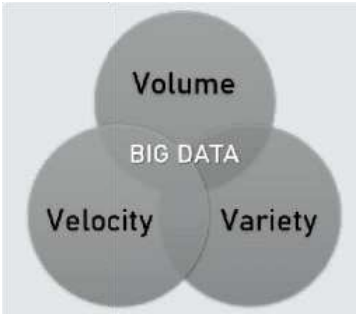
sebagian besar dikumpulkan melalui unggahan foto dan video, pertukaran pesan, dan posting komentar, di antara hal-hal lainnya(Muhammad Syarif Hartawan et al., 2022).

Dalam waktu terbang 30 menit, satu mesin Jet dapat menghasilkan 10+ gigabyte data. Dengan ribuan penerbangan setiap hari, jumlah data yang dihasilkan dapat mencapai beberapa Petabyte. Setiap hari, Bursa Efek New York menghasilkan sekitar satu terabyte data perdagangan baru(Wardani et al., 2025).

***Gambar 2**: Contoh Big Data*

1.2 Karakteristik Big Data

Big data dapat dijelaskan dengan karakteristik berikut seperti yang ditunjukkan pada Gambar 3.



***Gambar 3. 3** Karakteristik Big data*

- Volume

  Istilah 'Big Data' mengacu pada sejumlah besar informasi. Istilah "volume" mengacu pada sejumlah besar data. Besarnya data memainkan peran penting dalam menentukan nilainya. Ketika jumlah data sangat besar, maka disebut sebagai 'Big Data'. Artinya volume data menentukan apakah sekumpulan data dapat diklasifikasikan sebagai Big Data atau tidak. Data. Oleh karena itu, saat menangani Big Data, penting untuk mempertimbangkan 'Volume' tertentu(Siahaan, 2024).

 Contoh:

Pada tahun 2016, lalu lintas seluler di seluruh dunia diprediksi mencapai 6,2 Exabyte (6,2 miliar GB) per bulan. Lebih jauh lagi, pada tahun 2020, kita akan memiliki sekitar 40.000 Exabyte data.

- **K**ecepatan

  Istilah "kecepatan" mengacu pada pengumpulan data yang cepat. Data masuk dengan kecepatan tinggi dari mesin, jaringan, media sosial, ponsel, dan sumber lain dalam kecepatan Big Data. Terjadi aliran data yang besar dan konstan. Hal ini memengaruhi potensi data, atau seberapa cepat data dibuat dan diproses untuk memenuhi kebutuhan. Pengambilan sampel data dapat membantu dalam menangani masalah seperti 'kecepatan'. Misalnya, Google menerima lebih dari 3,5 miliar kueri setiap hari. Selain itu, jumlah pengguna Facebook tumbuh pada tingkat sekitar 22% setiap tahun.

- Variasi

  Data terstruktur hanyalah data yang telah diatur. Biasanya mengacu pada data yang telah ditentukan dalam hal panjang dan format.

Data semi terstruktur adalah jenis data yang semi-terorganisasi. Jenis data ini tidak mengikuti struktur data tradisional. Jenis data ini diwakili oleh berkas log.

Data tidak terstruktur hanyalah data yang belum tersusun. Biasanya mengacu pada data yang tidak sesuai dengan struktur baris dan kolom standar basis data relasional. Teks, gambar, video, dll. adalah contoh data tidak terstruktur yang tidak dapat disimpan dalam bentuk baris dan kolom.

1.2.1 Manfaat Pemrosesan Big Data

Kemampuan untuk memproses Big Data membawa banyak manfaat, seperti-

1. Bisnis dapat memanfaatkan intelijen luar saat mengambil keputusan.
2. Akses ke data sosial dari mesin pencari dan situs seperti Facebook, Twitter memungkinkan organisasi untuk menyempurnakan strategi bisnis mereka.
3. Peningkatan layanan pelanggan (Sistem umpan balik pelanggan tradisional digantikan oleh sistem baru yang dirancang dengan teknologi Big Data.
4. Peningkatan layanan pelanggan (Dalam sistem baru ini, Big Data dan teknologi pemrosesan bahasa alami digunakan untuk membaca dan mengevaluasi respons konsumen.
5. Identifikasi dini risiko terhadap produk/layanan, jika ada
6. Efisiensi operasional yang lebih baik

Teknologi Big Data dapat digunakan untuk membuat area persiapan atau zona pendaratan untuk data baru sebelum mengidentifikasi data apa yang harus dipindahkan ke gudang data(Veri Ferdiansyah & Muhammad Irwan Padli Nasution, 2023). Selain itu, integrasi teknologi Big Data dan

gudang data tersebut membantu organisasi untuk memindahkan data yang jarang diakses.

Mengapa Big Data Penting?

a) Penghematan Biaya. Data besar membantu dalam menyediakan intelijen bisnis yang dapat mengurangi biaya dan meningkatkan efisiensi operasi. Proses seperti jaminan kualitas dan pengujian dapat melibatkan banyak komplikasi terutama dalam industri seperti biofarmasi dan nanoteknologi
b) Pengurangan Waktu. Perusahaan dapat mengumpulkan data dari berbagai sumber menggunakan analitik dalam memori secara real-time. Alat seperti Hadoop memungkinkan bisnis mengevaluasi data dengan cepat, sehingga mereka dapat membuat keputusan cepat berdasarkan temuan mereka.
c) Memahami kondisi pasar. Bisnis dapat memperoleh manfaat dari analisis data besar dengan memperoleh pemahaman yang lebih baik tentang kondisi pasar.
d) Menganalisis perilaku pembelian klien, misalnya, memungkinkan bisnis menemukan barang yang paling populer dan mengembangkannya dengan tepat. Hal ini memungkinkan bisnis untuk tetap unggul dalam persaingan.
e) Mendengarkan Media SosialPerusahaan dapat melakukan analisis sentimen menggunakan alat Big Data. Hal ini memungkinkan mereka untuk mendapatkan umpan balik tentang perusahaan mereka, yaitu, siapa yang mengatakan apa tentang perusahaan tersebut. Perusahaan dapat menggunakan alat Big Data untuk meningkatkan kehadiran online mereka
f) Menggunakan Analisis Big Data untuk Meningkatkan Akuisisi dan Retensi Pelanggan. Pelanggan merupakan aset penting yang diandalkan oleh setiap perusahaan. Tanpa basis konsumen yang kuat, tidak ada perusahaan yang dapat berhasil. Namun, bahkan

dengan basis konsumen yang kuat, bisnis tidak dapat mengabaikan persaingan pasar. Akan sulit bagi bisnis untuk berhasil jika mereka tidak memahami apa yang diinginkan konsumen mereka. Akan sulit bagi bisnis untuk berhasil jika mereka tidak memahami apa yang diinginkan konsumen mereka. Hal ini akan mengakibatkan hilangnya pelanggan, yang akan berdampak negatif pada pertumbuhan bisnis. Bisnis dapat menggunakan analisis big data untuk mendeteksi tren dan pola yang terkait dengan pelanggan. Analisis perilaku pelanggan adalah kunci keberhasilan bisnis.

g) Menggunakan Analisis Big Data untuk Memecahkan Masalah Pengiklan dan Menawarkan Wawasan Pemasaran. Semua aktivitas perusahaan dibentuk oleh analisis big data. Hal ini memungkinkan bisnis untuk memenuhi harapan klien. Analisis big data membantu dalam modifikasi rangkaian produk perusahaan. Hal ini menjamin bahwa inisiatif pemasaran efektif.

h) Analisis Big Data sebagai Penggerak Inovasi dan Pengembangan Produk. Perusahaan dapat menggunakan big data untuk berinovasi dan memperbarui barang mereka.

1.3 Aplikasi BIG DATA

Semua data harus dicatat dan diproses, yang membutuhkan banyak keahlian, sumber daya, dan waktu(Eka Mayasari & Agussalim Agussalim, 2023). Data dapat digunakan secara kreatif dan bermakna untuk memberikan manfaat bisnis. Ada tiga jenis aplikasi bisnis, masing-masing dengan berbagai tingkat potensi revolusioner seperti yang ditunjukkan pada Gambar 4.

***Gambar 4** Aplikasi Big Data*

1. Aplikasi pemantauan dan pelacakan

Berikut ini adalah aplikasi Big Data yang pertama dan paling mendasar. Di hampir semua industri, aplikasi ini membantu meningkatkan efisiensi perusahaan. Berikut ini adalah beberapa contoh aplikasi khusus:

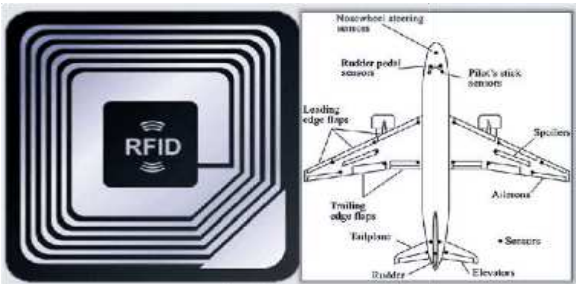
- Pemantauan kesehatan Masyarakat**.** Pemerintah AS mendorong semua pemangku kepentingan layanan kesehatan untuk membangun platform nasional untuk interoperabilitas dan standar berbagi data. Ini akan memungkinkan penggunaan sekunder data kesehatan, yang akan memajukan analisis BIG DATA dan pengobatan holistik presisi yang dipersonalisasi. Ini akan menjadi platform berbasis luas seperti Google flu trends.

Gambar 5 *Pemantauan kesehatan masyarakat*

- Pemantauan Sentimen Konsumen. Media sosial telah menjadi lebih kuat daripada iklan. Banyak perusahaan yang baik telah memindahkan sebagian besar anggaran iklan mereka dari media tradisional ke media sosial. Mereka telah menyiapkan platform mendengarkan Big Data, tempat aliran data media sosial (termasuk tweet, dan kiriman Facebook serta kiriman blog) difilter dan dianalisis untuk kata kunci atau sentimen tertentu, menurut demografi dan wilayah tertentu. Informasi yang dapat ditindaklanjuti dari analisis ini disampaikan kepada para profesional pemasaran untuk tindakan yang tepat, terutama saat produk tersebut baru di pasaran.

Gambar 6. Konsumen Pemantauan Sentimen

- Pelacakan Aset



Gambar 7 *Pelacakan Aset*

Departemen Pertahanan AS mendorong industri untuk merancang chip RFID kecil yang dapat mencegah pemalsuan komponen elektronik yang berakhir di avionik atau papan sirkuit untuk perangkat lain. Pesawat terbang merupakan salah satu pengguna sensor terberat yang melacak setiap aspek kinerja setiap bagian pesawat. Data dapat ditampilkan di dasbor serta disimpan untuk analisis terperinci nanti. Bekerja dengan perangkat komunikasi,

sensor ini dapat menghasilkan aliran data yang deras. Pencurian oleh pembeli dan karyawan merupakan sumber utama hilangnya pendapatan bagi pengecer. Semua barang berharga di toko dapat diberi tag RFID, dan gerbang toko dapat dilengkapi dengan pembaca RF. Ini dapat membantu mengamankan produk, dan mengurangi kebocoran (pencurian) dari toko.

- Pemantauan rantai pasokan. Semua kontainer di kapal mengomunikasikan status dan lokasinya menggunakan tag RFID. Dengan demikian, pengecer dan pemasoknya dapat memperoleh visibilitas waktu nyata terhadap inventaris di seluruh rantai pasokan global. Pengecer dapat mengetahui dengan tepat di mana barang-barang berada di gudang, dan dengan demikian dapat membawanya ke toko pada waktu yang tepat. Hal ini khususnya relevan untuk barang-barang musiman yang harus dijual tepat waktu, atau barang-barang tersebut akan dijual dengan harga diskon. Dengan tag RFID tingkat barang, pengecer juga memperoleh visibilitas penuh terhadap setiap barang dan dapat melayani pelanggan mereka dengan lebih baik.

***Gambar 8**. Pemantauan rantai pasokan*

- Perawatan mesin preventif. Semua mesin, termasuk mobil dan komputer, terkadang cenderung mengalami kegagalan. Hal ini disebabkan oleh satu atau beberapa komponennya yang tidak dapat berfungsi lagi. Sebagai tindakan pencegahan, peralatan berharga dapat dilengkapi dengan sensor. Aliran data yang terus-menerus dari sensor dapat dipantau dan dianalisis untuk memperkirakan status komponen utama, dan dengan demikian, memantau kesehatan mesin secara keseluruhan. Dengan demikian, perawatan preventif dapat mengurangi biaya waktu henti(Fern??ndez et al., 2015).

Gambar 9 *Pemeliharaan preventif*

2. Aplikasi Analisis dan Wawasan.

Ini adalah generasi berikutnya dari aplikasi big data. Mereka memiliki kemampuan untuk meningkatkan efektivitas perusahaan dan memiliki potensi transformasional. Big Data dapat diatur dan dianalisis untuk mengungkap tren dan wawasan yang dapat digunakan untuk :

- meningkatkan bisnis.
- Kepolisian Prediktif
- Memenangkan pemilihan politik
- Kesehatan Pribadi

- Kepolisian Prediktif

Gagasan tentang kepolisian prediktif diciptakan oleh Departemen Kepolisian Los Angeles. LAPD bekerja sama dengan akademisi UC Berkeley untuk memeriksa basis data besarnya yang berisi 13 juta kejahatan yang mencakup 80 tahun dan memperkirakan kemungkinan terjadinya jenis kejahatan tertentu pada waktu dan wilayah tertentu. Mereka mengidentifikasi titik-titik rawan kejahatan dari kategori tertentu, pada waktu dan wilayah tertentu. Mereka mengidentifikasi titik-titik rawan kejahatan tempat kejahatan telah terjadi dan kemungkinan akan terjadi di masa mendatang. Setelah wawasan dasar yang diperoleh dari metafora gempa bumi dan gempa susulannya, pola kejahatan disimulasikan secara statistik. Model tersebut menyatakan bahwa begitu kejahatan terjadi di suatu lokasi, hal itu merupakan gangguan TERTENTU dalam harmoni, dan dengan demikian, akan menyebabkan kemungkinan lebih besar terjadinya kejahatan serupa di sekitar lokasi tersebut segera. Model tersebut menunjukkan untuk setiap wilayah polisi, blok lingkungan tertentu dan slot waktu tertentu, di mana kejahatan kemungkinan besar terjadi. Dengan menyelaraskan jadwal patroli mobil polisi sesuai dengan prediksi model, LAPD dapat mengurangi kejahatan sebesar 12 persen hingga 26 persen untuk berbagai kategori kejahatan. Baru-baru ini, departemen Kepolisian SAN Francisco merilis kejahatannya sendiri selama lebih dari 2 tahun, sehingga analis data dapat memodelkan data tersebut dan mencegah kejahatan di masa mendatang.

Gambar 10. *Pemolisian prediktif*

- Memenangkan pemilihan politik

Presiden AS, Barack Obama adalah kandidat politik besar pertama yang menggunakan big data secara signifikan, dalam pemilihan umum 2008. Ia adalah presiden pertama yang menggunakan big data. Kampanyenya mengumpulkan data tentang jutaan orang, termasuk para pendukungnya. Mereka menemukan mekanisme untuk memperoleh sumbangan kampanye kecil dari jutaan pendukung. Mereka menggunakan Big Data membuat profil pribadi dari jutaan pendukung dan apa yang telah mereka lakukan dan dapat mereka lakukan untuk kampanye. Data digunakan untuk menentukan pemilih yang belum menentukan pilihan yang dapat diubah ke pihak mereka. Mereka memberikan nomor telepon pemilih yang belum menentukan pilihan ini kepada para relawan. Hasil panggilan direkam secara real time menggunakan aplikasi web interaktif. Obama sendiri menggunakan akun twitternya untuk mengomunikasikan pesannya secara langsung dengan jutaan pengikutnya. Setelah pemilihan, Obama mengubah daftar puluhan juta pendukungnya menjadi mesin advokasi yang akan memberikan dukungan akar rumput untuk inisiatif presiden. Sejak saat itu, hampir semua kampanye menggunakan big data.

Gambar 11 *Memenangkan pemilihan politik*

Senator Bernie Sanders menggunakan buku pedoman big data yang sama untuk membangun mesin politik nasional yang efektif yang sepenuhnya didukung oleh para donatur kecil. Analis pemilu, Nate Silver, menciptakan model prediktif yang canggih menggunakan masukan dari banyak jajak pendapat dan survei politik untuk meyakinkan para pakar agar berhasil memprediksi pemenang pemilu AS. Namun, Nate tidak berhasil memprediksi kebangkitan dan kemenangan akhir Donald Trump dan itu menunjukkan keterbatasan big data.

- Kesehatan pribadi

Pengetahuan dan teknologi medis berkembang pesat. Sistem Watson IBM adalah mesin analisis data besar yang menyerap dan mencerna semua informasi medis di dunia, lalu menerapkannya secara cerdas pada situasi individu. Watson dapat memberikan diagnosis medis yang terperinci dan akurat menggunakan gejala terkini, riwayat pasien, riwayat medis dan tren lingkungan, serta parameter lainnya. Produk serupa mungkin ditawarkan sebagai APP kepada dokter berlisensi, dan bahkan individu, untuk meningkatkan produktivitas dan akurasi dalam perawatan kesehatan.

3. Pengembangan Produk Baru

Ini adalah gagasan yang sama sekali baru yang sebelumnya tidak ada. Aplikasi ini memiliki kemampuan untuk mengubah seluruh sektor dan menyediakan aliran pendapatan baru bagi organisasi.

- Asuransi Mobil Fleksibel
- Promosi ritel berbasis lokasi
- Layanan rekomendasi

➢ Asuransi Mobil Fleksibel

Perusahaan asuransi mobil dapat menggunakan data GPS dari mobil untuk menghitung risiko kecelakaan berdasarkan pola perjalanan. Perusahaan mobil dapat menggunakan data sensor mobil untuk melacak kinerja mobil. Pengemudi yang lebih aman dapat diberi penghargaan dan pengemudi yang nakal dapat diberi sanksi.

Gambar 12 *Sistem pelacakan kendaraan GPS*

➢ Promosi ritel berbasis lokasi

Pengecer atau pengiklan pihak ketiga dapat menargetkan pelanggan dengan promosi dan kupon tertentu berdasarkan data lokasi yang diperoleh melalui sistem penentuan posisi global (GPS), waktu, keberadaan toko di sekitar, dan memetakannya ke data preferensi konsumen yang tersedia dari basis data media sosial. Iklan dan penawaran dapat disampaikan melalui aplikasi seluler, SMS, dan email. Berikut ini adalah contoh aplikasi seluler.

Gambar 13 *Promosi ritel berbasis lokasi*

- Layanan rekomendasi

E-dagang telah menjadi industri yang berkembang pesat dalam beberapa dekade terakhir. Berbagai macam produk dijual dan dibagikan melalui internet. Riwayat penelusuran dan pembelian pengguna web di situs e-dagang dimanfaatkan untuk mempelajari preferensi dan kebutuhan mereka, serta untuk mengiklankan penawaran produk dan harga yang relevan secara real-time. Amazon menggunakan sistem mesin rekomendasi yang dipersonalisasi untuk menyarankan produk tambahan baru kepada konsumen berdasarkan minat berbagai produk.

Gambar 14 *Layanan Rekomendasi*

Netflix juga menggunakan mesin rekomendasi untuk menyarankan pilihan hiburan kepada penggunanya. Big data berharga di semua industri.

Berikut adalah tiga jenis utama sumber data big data. Contohnya (komunikasi antarmanusia, komunikasi manusia-mesin, komunikasi mesin-mesin.) Setiap jenis memiliki banyak sumber data. Ada tiga jenis aplikasi. Ketiganya adalah jenis pemantauan, jenis analisis, dan pengembangan produk baru. Ketiganya berdampak pada efisiensi, efektivitas, dan bahkan disrupsi industri.

1.4 Alat yang digunakan dalam BIG DATA

Ada sejumlah alat yang digunakan dalam BIGDATA. Alat yang paling populer adalah: -

a.  Apache Hadoop. Kerangka kerja big data adalah pustaka perangkat lunak Apache Hadoop. Pustaka ini memungkinkan kumpulan data besar diproses di seluruh kluster komputer secara terdistribusi. Pustaka ini merupakan salah satu teknologi big data yang paling canggih, dengan kemampuan untuk berkembang dari satu server menjadi ribuan komputer.

- Fitur

- Saat memanfaatkan server proxy HTTP, autentikasi ditingkatkan.
- Spesifikasi upaya Sistem Berkas yang Kompatibel dengan Hadoop. Karakteristik yang diperluas untuk sistem berkas bergaya POSIX didukung. Memiliki teknologi dan alat big data yang menawarkan ekosistem kuat yang sangat sesuai untuk memenuhi kebutuhan analitis pengembang.
- Memberikan Fleksibilitas dalam Pemrosesan Data. Memungkinkan Pemrosesan Data yang lebih cepat

b. HPCC. HPCC adalah alat big data yang dikembangkan oleh LexisNexis Risk Solution. Alat ini menggunakan satu platform, satu arsitektur, dan satu bahasa pemrograman untuk pemrosesan data.

➢ Fitur

- Ini adalah salah satu alat data besar yang sangat efisien yang menyelesaikan tugas data besar dengan kode yang jauh lebih sedikit.
- Ini adalah salah satu alat pemrosesan data besar yang menawarkan redundansi dan ketersediaan tinggi.
- Dapat digunakan untuk pemrosesan data kompleks pada kluster Thor. IDE grafis menyederhanakan pengembangan, pengujian, dan debugging. Secara otomatis mengoptimalkan kode untuk pemrosesan paralel
- Memberikan peningkatan skalabilitas dan kinerja. Kode ECL dikompilasi menjadi C++ yang dioptimalkan, dan juga dapat diperluas menggunakan pustaka C++

c. 

Badai Apache. Storm adalah sistem komputasi big data sumber terbuka yang gratis. Ini adalah salah satu alat big data terbaik yang menawarkan sistem pemrosesan real-time terdistribusi dan toleran terhadap kesalahan. Dengan kemampuan komputasi real-time.

➢ Fitur

- Ini adalah salah satu alat terbaik dari daftar alat data besar yang diukur sebagai pemrosesan satu juta pesan 100 byte per detik per node
- Ia memiliki teknologi dan alat data besar yang menggunakan kalkulasi paralel yang berjalan pada sekumpulan mesin.
- Ini akan secara otomatis restart jika sebuah node mati. Pekerja akan di-restart pada node lain. Storm menjamin bahwa setiap unit data akan diproses setidaknya sekali atau tepat sekali.
- Setelah diterapkan, Storm tentu saja merupakan alat termudah untuk analisis Bigdata

d. 

: Qubole. Qubole Data adalah platform manajemen big data yang otonom. Ini adalah alat big data sumber terbuka yang dikelola sendiri, dioptimalkan sendiri, dan memungkinkan tim data untuk fokus pada hasil bisnis.

➢ Fitur

- Fitur:
- Platform Tunggal untuk setiap kasus penggunaan
- Ini adalah perangkat lunak data besar sumber terbuka yang memiliki Mesin, yang dioptimalkan untuk Cloud.
- Keamanan, Tata Kelola, dan Kepatuhan yang Komprehensif
- Menyediakan Peringatan, Wawasan, dan Rekomendasi yang dapat ditindaklanjuti untuk mengoptimalkan keandalan, kinerja, dan biaya.
- Secara otomatis memberlakukan kebijakan untuk menghindari melakukan tindakan manual yang berulang

e. 

Apache Cassandra. Basis data Apache Cassandra digunakan secara luas saat ini untuk menyediakan manajemen data dalam jumlah besar yang efektif.

➢ Fitur

- Dukungan untuk replikasi di beberapa pusat data dengan menyediakan latensi yang lebih rendah bagi pengguna
- Data secara otomatis direplikasi ke beberapa node untuk toleransi kesalahan
- Ini adalah salah satu alat big data terbaik yang paling cocok untuk aplikasi yang tidak mampu kehilangan data, bahkan ketika seluruh pusat data sedang tidak berfungsi.
- Cassandra menawarkan kontrak dukungan dan layanan tersedia dari pihak ketiga

f. 
Sayap Statis. Statwing adalah alat statistik yang mudah digunakan. Alat ini dibuat oleh dan untuk analis big data. Antarmukanya yang modern memilih uji statistik secara otomatis.

➢ Fitur

- Ini adalah perangkat lunak big data yang dapat menjelajahi data apa pun dalam hitungan detik. Statwing membantu membersihkan data, menjelajahi hubungan, dan membuat diagram dalam hitungan menit.
- Memungkinkan pembuatan histogram, diagram sebar, peta panas, dan diagram batang yang diekspor ke Excel atau PowerPoint. Ia juga menerjemahkan hasil ke dalam bahasa Inggris sederhana, sehingga analis yang tidak terbiasa dengan analisis statistik

g. 
SofaDB. CouchDB menyimpan data dalam dokumen JSON yang dapat diakses melalui web atau kueri menggunakan JavaScript. CouchDB menawarkan penskalaan terdistribusi dengan penyimpanan yang toleran terhadap kesalahan. CouchDB memungkinkan akses data dengan mendefinisikan Protokol Replikasi Couch.

➢ Fitur

- CouchDB adalah database node tunggal yang bekerja seperti database lainnya
- Ini adalah salah satu alat pemrosesan data besar yang memungkinkan menjalankan satu server basis data logis pada sejumlah server.
- Menggunakan protokol HTTP dan format data JSON yang ada di mana-mana. Replikasi database yang mudah di beberapa server. Antarmuka yang mudah untuk penyisipan, pembaruan, pengambilan, dan penghapusan dokumen
- Format dokumen berbasis JSON dapat diterjemahkan ke berbagai bahasa

h.  Pentaho. Pentaho menyediakan perangkat big data untuk mengekstrak, menyiapkan, dan memadukan data. Perangkat ini menawarkan visualisasi dan analitik yang mengubah cara menjalankan bisnis apa pun. Perangkat big data ini memungkinkan Anda mengubah big data menjadi wawasan besar.

- Fitur:

- Akses dan integrasi data untuk visualisasi data yang efektif. Ini adalah perangkat lunak big data yang memberdayakan pengguna untuk merancang big data di sumbernya dan mengalirkannya untuk analisis yang akurat. Beralih atau gabungkan pemrosesan data dengan eksekusi dalam klaster secara mulus untuk mendapatkan pemrosesan yang maksimal. Izinkan pengecekan data dengan akses mudah ke analitik, termasuk bagan, visualisasi, dan pelaporan
- Mendukung spektrum luas sumber data besar dengan menawarkan kemampuan unik

i.  Apache Flink. Apache Flink adalah salah satu alat analisis data sumber terbuka terbaik untuk pemrosesan aliran data besar. Ini adalah aplikasi pengaliran data yang terdistribusi, berkinerja tinggi, selalu tersedia, dan akurat.

- Fitur:

- Memberikan hasil yang akurat, bahkan untuk data yang tidak berurutan atau terlambat datang
- Ia bersifat stateful dan toleran terhadap kesalahan serta dapat memulihkan dari kegagalan.
- Ini adalah perangkat lunak analisis data besar yang dapat bekerja dalam skala besar, berjalan pada ribuan node
- Memiliki karakteristik throughput dan latensi yang baik

- Alat big data ini mendukung pemrosesan aliran dan windowing dengan semantik waktu kejadian. Alat ini mendukung windowing fleksibel berdasarkan waktu, jumlah, atau sesi ke jendela yang digerakkan oleh data
- Mendukung berbagai konektor ke sistem pihak ketiga untuk sumber data dan sink

j. cloudera awan. Cloudera adalah platform big data modern yang tercepat, termudah, dan sangat aman. Platform ini memungkinkan siapa saja untuk mendapatkan data apa pun di lingkungan apa pun dalam satu platform yang dapat diskalakan.

➢ Fitur:

- Perangkat lunak analitik data besar berkinerja tinggi
- Ini menawarkan ketentuan untuk multi-cloud
- Terapkan dan kelola Cloudera Enterprise di seluruh AWS, Microsoft Azure, dan Google Cloud Platform. Jalankan dan hentikan kluster, dan bayar hanya untuk apa yang dibutuhkan saat dibutuhkan
- Mengembangkan dan melatih model data
- Pelaporan, eksplorasi, dan layanan mandiri intelijen bisnis
- Memberikan wawasan waktu nyata untuk pemantauan dan deteksi
- Melakukan penilaian model yang akurat dan penyajian

k. Refine Buka Perbaiki. OpenRefine adalah perangkat lunak big data yang hebat. Perangkat lunak ini merupakan perangkat lunak analisis big data yang membantu mengolah data yang berantakan, membersihkannya, dan mengubahnya dari satu format ke format lain. Perangkat lunak ini juga memungkinkan perluasan data dengan layanan web dan data eksternal.

➢ Fitur:

- Alat OpenRefine membantu Anda menjelajahi kumpulan data besar dengan mudah. Alat ini dapat digunakan untuk menghubungkan dan memperluas kumpulan data Anda dengan berbagai layanan web. Mengimpor data dalam berbagai format.
- Jelajahi kumpulan data dalam hitungan detik
- Terapkan transformasi sel dasar dan lanjutan
- Memungkinkan untuk menangani sel yang berisi beberapa nilai Buat tautan instan antara kumpulan data. Gunakan ekstraksi entitas bernama pada bidang teks untuk mengidentifikasi topik secara otomatis. Lakukan operasi data tingkat lanjut dengan bantuan Refine

1.5 Tantangan dalam BIG DATA

Kurangnya pemahaman yang tepat tentang Big Data Perusahaan gagal dalam inisiatif Big Data mereka karena kurangnya pemahaman. Karyawan mungkin tidak tahu apa itu data, penyimpanannya, pemrosesannya, pentingnya, dan sumbernya. Profesional data mungkin tahu apa yang sedang terjadi, tetapi yang lain mungkin tidak memiliki gambaran yang jelas. Misalnya, jika karyawan tidak memahami pentingnya penyimpanan data, mereka mungkin tidak menyimpan cadangan data sensitif.

Data. Mereka mungkin tidak menggunakan basis data dengan benar untuk penyimpanan. Akibatnya, saat data penting ini dibutuhkan, data tersebut tidak dapat diambil dengan mudah.

Lokakarya dan seminar tentang Big Data harus diadakan di perusahaan untuk semua orang. Program pelatihan dasar harus diatur untuk semua karyawan yang menangani data secara rutin dan menjadi bagian dari proyek Big Data. Pemahaman dasar tentang konsep data harus ditanamkan pada semua tingkatan organisasi.

1.5.1 Masalah pertumbuhan data

Salah satu tantangan paling mendesak dari Big Data adalah menyimpan semua kumpulan data besar ini dengan benar. Jumlah data yang disimpan di pusat data dan basis data perusahaan meningkat dengan cepat. Karena kumpulan data ini tumbuh secara eksponensial seiring waktu, data tersebut menjadi sangat sulit untuk ditangani. Sebagian besar data tidak terstruktur dan berasal dari dokumen, video, audio, file teks, dan sumber lainnya. Ini berarti Anda tidak dapat menemukannya di basis data.

- Larutan

Untuk menangani kumpulan data yang besar ini, perusahaan memilih teknik modern, seperti kompresi, tiering, dan deduplikasi. Kompresi digunakan untuk mengurangi jumlah bit dalam data, sehingga mengurangi ukuran keseluruhannya. Deduplikasi adalah proses menghilangkan data duplikat dan yang tidak diinginkan dari kumpulan data. Tiering data memungkinkan perusahaan untuk menyimpan data dalam tingkatan penyimpanan yang berbeda. Ini memastikan bahwa data berada di ruang penyimpanan yang paling tepat. Tingkatan data dapat berupa cloud publik, cloud pribadi, dan penyimpanan flash, tergantung pada ukuran dan kepentingan data. Perusahaan juga memilih Alat Data Besar, seperti: Hadoop, NoSQL dan teknologi lainnya. Hal ini membawa kita pada masalah Big Data yang ketiga.

1.5.2 Kebingungan saat memilih alat Big Data

Perusahaan sering kali bingung saat memilih alat terbaik untuk analisis dan penyimpanan Big Data. Apakah HBase atau Cassandra teknologi terbaik untuk penyimpanan data? Apakah Hadoop MapReduce cukup baik atau Spark akan menjadi pilihan yang lebih baik untuk analisis dan penyimpanan data? Pertanyaan-pertanyaan ini mengganggu perusahaan dan terkadang mereka tidak dapat menemukan jawabannya.

Mereka akhirnya membuat keputusan yang buruk dan memilih teknologi yang tidak tepat. Akibatnya, uang, waktu, upaya, dan jam kerja terbuang sia-sia.

- Larutan

Cara terbaik untuk mengatasinya adalah dengan mencari bantuan profesional. Anda dapat menyewa tenaga profesional berpengalaman yang lebih memahami alat-alat ini. Cara lain adalah dengan berkonsultasi dengan konsultan Big Data. Di sini, konsultan akan memberikan rekomendasi alat terbaik, berdasarkan skenario perusahaan Anda. Berdasarkan saran mereka, Anda dapat menyusun strategi dan kemudian memilih alat terbaik untuk Anda.

1.5.3 Kurangnya profesional data

Untuk menjalankan teknologi modern dan perangkat Big Data ini, perusahaan memerlukan profesional data yang terampil. Profesional ini akan mencakup ilmuwan data, analis data, dan insinyur data yang berpengalaman dalam bekerja dengan perangkat dan memahami kumpulan data yang besar. Perusahaan menghadapi masalah kurangnya profesional Big Data. Hal ini karena perangkat penanganan data telah berkembang pesat, tetapi dalam kebanyakan kasus, para profesional belum berkembang. Langkah-langkah yang dapat ditindaklanjuti perlu diambil untuk menjembatani kesenjangan ini.

- Larutan

Perusahaan menginvestasikan lebih banyak uang dalam perekrutan profesional yang terampil. Mereka juga harus menawarkan program pelatihan kepada staf yang ada untuk mendapatkan hasil maksimal dari mereka. Langkah penting lainnya yang diambil oleh organisasi adalah pembelian solusi analisis data yang didukung oleh kecerdasan buatan/pembelajaran mesin. Alat-alat ini dapat dijalankan oleh para

profesional yang bukan ahli ilmu data tetapi memiliki pengetahuan dasar. Langkah ini membantu perusahaan menghemat banyak uang untuk perekrutan.

1.5.4 Mengamankan data

Mengamankan kumpulan data yang sangat besar ini merupakan salah satu tantangan yang berat dalam Big Data. Sering kali perusahaan begitu sibuk dalam memahami, menyimpan, dan menganalisis kumpulan data mereka sehingga mereka menunda keamanan data untuk tahap selanjutnya. Namun, ini bukanlah langkah yang cerdas karena repositori data yang tidak dilindungi dapat menjadi tempat berkembang biaknya para peretas jahat. Perusahaan dapat kehilangan hingga $3,7 juta karena pencurian data atau pelanggaran data.

- Larutan

Perusahaan merekrut lebih banyak profesional keamanan siber untukmelindungi data mereka. Langkah-langkah lain yang diambil untuk mengamankan data meliputi:

a. Enkripsi data
b. Pemisahan data
c. Identitas dan kontrol akses
d. Implementasi keamanan titik akhir
e. Pemantauan keamanan waktu nyat

1.5.5 Mengintegrasikan data dari berbagai sumber

Data dalam suatu organisasi berasal dari berbagai sumber, seperti halaman media sosial, aplikasi ERP, log pelanggan, laporan keuangan, email, presentasi, dan laporan yang dibuat oleh karyawan. Menggabungkan semua data ini untuk menyiapkan laporan merupakan tugas yang menantang. Ini adalah area yang sering diabaikan oleh perusahaan. Namun, integrasi data sangat penting untuk analisis, pelaporan, dan intelijen bisnis, jadi integrasi data harus sempurna.

- Larutan

Perusahaan harus memecahkan masalah integrasi data mereka dengan membeli alat yang tepat. Berikut ini adalah beberapa alat integrasi data terbaik:

- Integrasi Data Talend
- Integrator Data Centerprise
- ArcESB
- InfoSphere IBM
- Banyak sekali
- Pusat Daya Informatica
- SemanggiDX
- Bahasa Indonesia: Microsoft SQL
- Tampilan Qlik
- Integrator Layanan Data Oracle

Agar Big Data dapat digunakan secara optimal, perusahaan harus mulai melakukan berbagai hal secara berbeda. Ini berarti merekrut staf yang lebih baik, mengubah manajemen, meninjau kebijakan bisnis yang ada, dan teknologi yang digunakan. Untuk meningkatkan pengambilan keputusan, mereka dapat merekrut Chief Data Officer – sebuah langkah yang diambil oleh banyak perusahaan yang masuk dalam daftar Fortune 500.

Ringkasan

Big data mengacu pada data dalam jumlah besar yang sulit dikelola baik yang terorganisasi maupun tidak terstruktur – yang membanjiri perusahaan setiap hari. Big data dapat dievaluasi untuk mendapatkan wawasan yang membantu orang membuat penilaian yang lebih baik dan merasa lebih percaya diri dalam membuat keputusan bisnis yang penting.

Berikut ini adalah aplikasi Big Data yang paling mendasar dan mendasar. Aplikasi ini membantu meningkatkan efisiensi perusahaan di hampir setiap industri.

- Ini adalah aplikasi big data masa depan. Aplikasi ini berpotensi mengubah bisnis dan meningkatkan efektivitas perusahaan. Big data dapat diatur dan dianalisis untuk mengungkap pola dan wawasan yang dapat digunakan untuk meningkatkan kinerja perusahaan.
- Ini adalah konsep baru yang belum ada sebelumnya. Aplikasi ini berpotensi mengubah seluruh industri dan menghasilkan aliran pendapatan baru bagi bisnis.
- Apache Hadoop adalah seperangkat perangkat lunak sumber terbuka untuk memecahkan masalah yang melibatkan data dalam jumlah besar dan pemrosesan dengan memanfaatkan jaringan banyak komputer. Ia menggunakan konsep pemrograman MapReduce untuk membuat kerangka kerja perangkat lunak untuk penyimpanan dan pemrosesan data besar yang terdistribusi.
- Apache Cassandra adalah sistem manajemen basis data NoSQL penyimpanan kolom lebar terdistribusi yang dirancang untuk menangani data bervolume besar di banyak server komoditas sambil mempertahankan ketersediaan tinggi dan menghindari titik kegagalan tunggal.
- Cloudera, Inc. adalah perusahaan rintisan yang berkantor pusat di Santa Clara, California yang menawarkan cloud data perusahaan berbasis langganan. Platform Cloudera, yang berbasis pada teknologi sumber terbuka, memanfaatkan analitik dan pembelajaran mesin untuk mengekstrak wawasan dari data melalui koneksi yang aman.

- RapidMiner adalah platform perangkat lunak ilmu data yang dibangun oleh firma dengan nama yang sama yang menawarkan lingkungan terpadu untuk persiapan data, pembelajaran mesin, pembelajaran mendalam, penambangan teks, dan analitik prediktif.
- Kaggle, anak perusahaan Google LLC, adalah komunitas daring ilmuwan data dan pakar pembelajaran mesin.
- LexisNexis Risk Solutions menciptakan HPCC, yang sering dikenal sebagai DAS, sebuah platform sistem komputasi intensif data sumber terbuka. Platform HPCC didasarkan pada arsitektur perangkat lunak yang berjalan pada kluster komputasi komoditas dan menyediakan pemrosesan paralel data berkinerja tinggi untuk aplikasi big data.

Soal Latihan

Q1: Apa saja elemen fundamental BIG DATA?

A. Bahasa Indonesia: HDFS
B. BENANG
C. PetaKurangi
D. Semua ini

Q2: Apa yang membedakan Analisis BIG DATA dari jenis analisis lainnya?

A. Sumber Terbuka
B. Skalabilitas
C. Pemulihan Data
D. Semua ini

Q3: Apa saja V dari Big Data?

A. Volume
B. Kebenaran
C. Baik a dan b

D. Jelas

Q4: Harap identifikasi pernyataan yang benar.

A. Hadoop adalah platform yang sangat baik untuk mengekstrak dan menganalisis sejumlah kecil data.
B. Hadoop menggunakan HDFS untuk menyimpan data dan memungkinkan kompresi dan dekompresi data.
C. Untuk memecahkan masalah grafik dan pembelajaran mesin, kerangka kerja giraph kurang berguna dibandingkan kerangka kerja MapReduce.
D. Tidak ada yang disebutkan

Q5: Pada platform berikut mana Hadoop tersedia?

A. Logam polos
B. Lintas Platform
C. Mirip Unix
D. Tidak ada yang disebutkan

Q6: Daftar Hadoop mencakup basis data HBase, Sistem _____ Apache Mahout, dan operasi matriks.

A. Pengenalan pola
B. HPCC
C. Pembelajaran Mesin
D. SPSS

Q7: Elemen MapReduce bertugas memproses satu atau lebih potongan data dan menyediakan hasil keluaran.

A. Peta Tugas
B. Pemeta
C. Eksekusi tugas
D. Semua yang disebutkan

Q8: Meskipun kerangka Hadoop diimplementasikan dalam Java, aplikasi MapReduce tidak perlu ditulis dalam _____

A. Java
B. C

C. C#

D. Tidak ada yang disebutkan

Q9: Pasangan kunci/nilai masukan dipetakan ke kumpulan pasangan kunci/nilai perantara menggunakan _____.

A. Pemeta

B. Peredam

C. Keduanya

D. Tidak ada yang disebutkan

Q10: Jumlah peta biasanya ditentukan oleh total ukuran _____

A. masukan

B. keluaran

C. tugas

D. Tidak ada yang disebutkan

Q11: Pustaka perangkat lunak _____ merupakan kerangka kerja big data. Pustaka ini memungkinkan pemrosesan terdistribusi kumpulan data besar di seluruh kluster komputer.

A. Pemrograman Apple

B. Pemrograman R

C. Apache Hadoop

D. Semua di atas

Q12: Alat big data mana yang dikembangkan oleh LexisNexis Risk Solution?

A. Sistem SPCC

B. Sistem HPCC

C. Sistem TOCC

D. Tidak ada yang di atas

Q13: Alat big data mana yang menawarkan sistem pemrosesan terdistribusi real-time, toleran terhadap kesalahan, dengan kemampuan komputasi real-time.

A. Badai

B. HPCC
C. Bahasa Inggris: Qubole
D. Kasandra

Q14: Pernyataan mana yang benar tentang Apache Cassandra?

A. Ini adalah alat yang gratis dan bersumber terbuka.
B. Saat ini, teknologi ini banyak digunakan untuk menyediakan manajemen data dalam jumlah besar secara efektif.
C. Itu didistribusikan
D. Semua hal di atas

Q15: _____ menyimpan data dalam dokumen JSON yang dapat diakses melalui web atau query menggunakan JavaScript

A. SofaDB
B. Badai
C. Sarang lebah
D. Tidak ada yang di atas

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S.

(2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

BAB 2: Fondasi untuk Big Data

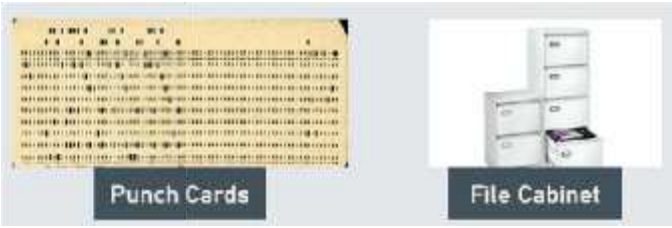
Zuhri Halim, S.Kom., M.Kom

Tujuan

- membedakan antara sistem berkas (FS) dan sistem berkas terdistribusi (DFS)
- memahami komputasi berskala melalui internet.
- memahami model pemrograman untuk Big Data.

Perkenalan

Mekanisme penyimpanan pertama yang digunakan oleh komputer untuk menyimpan data adalah kartu berlubang. Setiap kelompok kartu berlubang yang terkait (kartu berlubang yang terkait dengan program yang sama) biasanya disimpan dalam sebuah berkas; dan berkas-berkas disimpan dalam lemari arsip. Hal ini sangat mirip dengan apa yang kita lakukan saat ini untuk mengarsipkan dokumen-dokumen di kantor-kantor pemerintahan yang masih menggunakan dokumen sebagai dokumen harian. Dari sinilah istilah "Sistem Berkas" (FS) berasal. Sistem komputer berevolusi; tetapi konsepnya tetap sama.



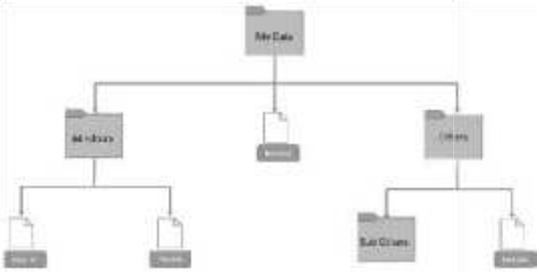
Gambar 2.1 *Mekanisme Penyimpanan*

2.1 Apa itu Sistem Berkas?

Alih-alih menyimpan informasi pada kartu berlubang; kita sekarang dapat menyimpan informasi/data dalam format digital pada perangkat penyimpanan digital seperti hard disk, flash drive…dll. Data terkait masih dikategorikan sebagai file; kelompok file terkait disimpan dalam folder.

Ekstensi file menunjukkan jenis informasi yang disimpan dalam file tersebut. Misalnya; ekstensi EXE mengacu pada file yang dapat dieksekusi, TXT mengacu pada file teks…dll. Sistem manajemen file digunakan oleh sistem operasi untuk mengakses file dan folder yang disimpan di komputer atau perangkat penyimpanan eksternal apa pun.

Gambar 2.2 *Penyimpanan Digital*



Gambar2. 3 *Contoh Sistem File*

2.2 Apa itu Sistem Berkas Terdistribusi?

Dalam Big Data, kita sering berurusan dengan beberapa kluster (komputer). Salah satu keuntungan utama Big Data adalah

kemampuannya melampaui kemampuan satu server super canggih dengan daya komputasi yang sangat tinggi. Seluruh ide Big Data adalah mendistribusikan data ke beberapa kluster dan memanfaatkan daya komputasi setiap kluster (node) untuk memproses informasi. Sistem berkas terdistribusi adalah sistem yang dapat menangani akses data ke beberapa kluster (node). Di bagian berikutnya, kita akan mempelajari lebih lanjut tentang cara kerjanya

DFS memiliki dua komponen

- **Transparansi Lokasi:** Transparansi Lokasi dicapai melalui komponen namespace.
- **Redundansi:** Redundansi dilakukan melalui komponen replikasi berkas.

Fitur Transparansi DFS

- ***Transparansi struktur*:**Klien tidak perlu mengetahui jumlah atau lokasi server file dan perangkat penyimpanan. Beberapa server file harus disediakan untuk kinerja, kemampuan beradaptasi, dan keandalan.
- ***Akses transparansi*:**Baik berkas lokal maupun berkas jarak jauh harus dapat diakses dengan cara yang sama. Sistem berkas harus secara otomatis ditempatkan pada berkas yang diakses dan mengirimkannya ke sisi klien.
- ***Transparansi penamaan*:**Tidak boleh ada petunjuk apa pun dalam nama berkas mengenai lokasi berkas. Setelah nama diberikan pada berkas, nama tersebut tidak boleh diubah selama pemindahan dari satu simpul ke simpul lainnya.
- ***Transparansi replikasi*:**Jika suatu berkas disalin pada beberapa node, salinan berkas dan lokasinya harus disembunyikan dari satu node ke node lainnya.
- **Mobilitas pengguna:**Secara otomatis akan membawa direktori asal pengguna ke node tempat pengguna masuk.

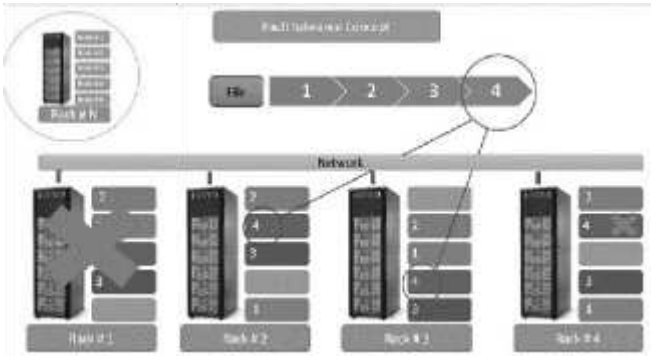
- **Pertunjukan:** Kinerja didasarkan pada jumlah waktu rata-rata yang dibutuhkan untuk memenuhi permintaan klien. Waktu ini mencakup waktu CPU + waktu yang dibutuhkan untuk mengakses penyimpanan sekunder + waktu akses jaringan. Sebaiknya kinerja Sistem Berkas Terdistribusi serupa dengan sistem berkas terpusat.
- **Kesederhanaan dan kemudahan penggunaan:** Antarmuka pengguna sistem berkas harus sederhana dan jumlah perintah dalam berkas harus kecil.
- **Ketersediaan tinggi:** Sistem Berkas Terdistribusi harus dapat terus berjalan jika terjadi kegagalan parsial seperti kegagalan tautan, kegagalan simpul, atau kerusakan drive penyimpanan. Sistem berkas terdistribusi yang sangat autentik dan adaptif harus memiliki server berkas yang berbeda dan independen untuk mengendalikan perangkat penyimpanan yang berbeda dan independen.

Bagaimana cara kerja sistem berkas terdistribusi (DFS)?

Sistem berkas terdistribusi bekerja sebagai berikut:

- **Distribusi:** Mendistribusikan blok-blok kumpulan data ke beberapa node. Setiap node memiliki daya komputasi sendiri; yang memberikan kemampuan DFS untuk memproses blok-blok data secara paralel.
- **Replikasi**: Sistem berkas terdistribusi juga akan mereplikasi blok data pada kluster yang berbeda dengan menyalin bagian informasi yang sama ke beberapa kluster di rak yang berbeda. Ini akan membantu mencapai hal berikut:
- ***Toleransi Kesalahan***: memulihkan blok data jika terjadi kegagalan kluster atau kegagalan rak. Replikasi data merupakan cara yang baik untuk mencapai toleransi kesalahan dan konkurensi tinggi; tetapi sangat sulit untuk mempertahankan perubahan yang sering terjadi.

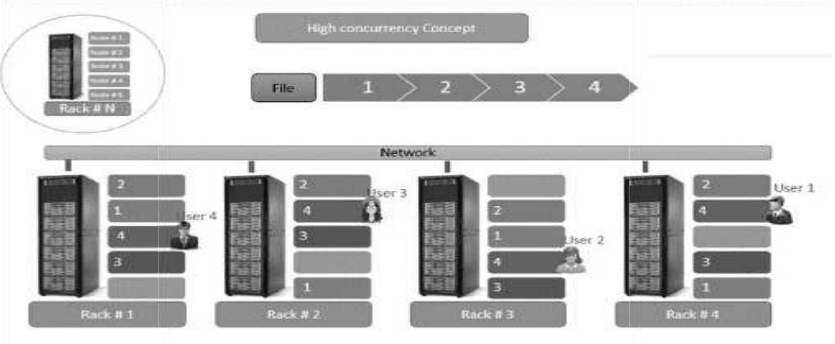
Asumsikan bahwa seseorang mengubah blok data pada satu kluster; perubahan ini perlu diperbarui pada semua replika data blok ini.



Gambar2.4 *Konsep Toleransi Kesalahan*

Replikasi data merupakan cara yang baik untuk mencapai toleransi kesalahan dan konkurensi yang tinggi; tetapi sangat sulit untuk mempertahankan perubahan yang sering terjadi. Asumsikan bahwa seseorang mengubah blok data pada satu kluster; perubahan ini perlu diperbarui pada semua replika data blok ini.

- ***Konkurensi Tinggi***: memanfaatkan bagian data yang sama untuk diproses oleh beberapa klien pada saat yang sama. Hal ini dilakukan dengan menggunakan daya komputasi setiap node untuk memproses blok data secara paralel.



Gambar 2.5 Konsep akurasi tinggi

Apa Keuntungan Sistem Berkas Terdistribusi (DFS)?

1. **Skalabilitas**: Anda dapat meningkatkan infrastruktur Anda dengan menambahkan lebih banyak rak atau kluster ke sistem Anda.
2. **Toleransi Kesalahan:** Replikasi data akan membantu mencapai toleransi kesalahan dalam kasus berikut:
   - Cluster sedang down
   - Rak sudah turun
   - Rak terputus dari jaringan.
   - Pekerjaan gagal atau dimulai ulang.
3. **Konkurensi Tinggi**: memanfaatkan daya komputasi setiap node untuk menangani beberapa permintaan klien (secara paralel) pada saat yang bersamaan. Gambar berikut mengilustrasikan konsep utama konkurensi tinggi dan cara mencapainya melalui replikasi data pada beberapa kluster.
4. DFS memungkinkan banyak pengguna untuk mengakses atau menyimpan data.
5. Memungkinkan data dibagikan dari jarak jauh.

6. Ini meningkatkan ketersediaan berkas, waktu akses, dan efisiensi jaringan.
7. Meningkatkan kapasitas untuk mengubah ukuran data dan juga meningkatkan kemampuan untuk bertukar data.
8. Sistem Berkas Terdistribusi memberikan transparansi data bahkan jika server atau disk gagal.

Apa kerugian dari Distributed File System (DFS)?

1. Dalam Sistem Berkas Terdistribusi, node dan koneksi perlu diamankan oleh karena itu kita dapat mengatakan bahwa keamanan dipertaruhkan.
2. Ada kemungkinan hilangnya pesan dan data dalam jaringan saat berpindah dari satu simpul ke simpul lainnya.
3. Koneksi basis data dalam kasus Sistem Berkas Terdistribusi rumit.
4. Penanganan basis data juga tidak mudah dalam Sistem Berkas Terdistribusi dibandingkan dengan sistem pengguna tunggal.
5. Ada kemungkinan kelebihan beban akan terjadi jika semua node mencoba mengirim data sekaligus.

2.3 Komputasi Skalabel Melalui Internet Apa itu Skalabilitas?

Dengan Cloud hosting, mudah untuk menambah dan mengurangi jumlah dan ukuran server berdasarkan kebutuhan. Hal ini dilakukan dengan menambah atau mengurangi sumber daya di cloud. Kemampuan untuk mengubah rencana karena fluktuasi dalam ukuran dan kebutuhan bisnis merupakan manfaat luar biasa dari komputasi cloud terutama saat mengalami pertumbuhan permintaan yang tiba-tiba. Komputasi Skalabel Melalui Internet:

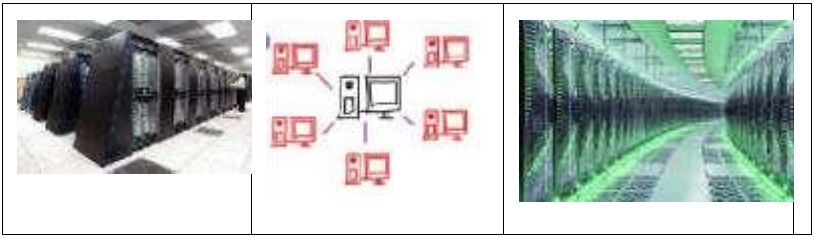
- *Era Komputasi Internet*
- *Komputasi Kinerja Tinggi*

- *Komputasi Berkapasitas Tinggi*
- *Tiga Paradigma Komputasi Baru*
- *Perbedaan Paradigma Komputasi*
- *Keluarga Sistem Terdistribusi*
- *Derajat Paralelisme*

2.3.1 Era Komputasi Internet

Miliaran orang menggunakan Internet setiap hari. Akibatnya, situs superkomputer dan pusat data besar harus menyediakan layanan komputasi berkinerja tinggi kepada sejumlah besar pengguna Internet secara bersamaan. Karena permintaan yang tinggi ini, Tolok Ukur Linpack untuk aplikasi komputasi berkinerja tinggi (HPC) tidak lagi optimal untuk mengukur kinerja sistem. Munculnya cloud komputasi justru menuntut sistem komputasi berthroughput tinggi (HTC) yang dibangun dengan teknologi komputasi paralel dan terdistribusi [5,6,19,25]. Kita harus meningkatkan pusat data menggunakan server cepat, sistem penyimpanan, dan jaringan bandwidth tinggi. Tujuannya adalah untuk memajukan komputasi berbasis jaringan dan layanan web dengan teknologi baru yang sedang berkembang.

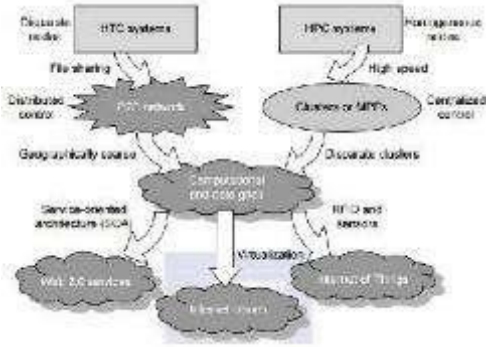




Gambar2.6 *Era Komputasi Internet*

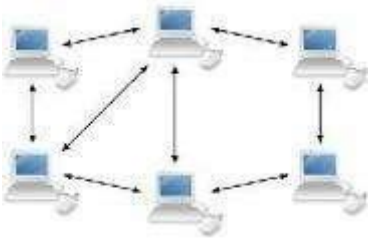
- Platform Evolusi

Kita harus meningkatkan pusat data menggunakan server cepat, sistem penyimpanan, dan jaringan bandwidth tinggi. Tujuannya adalah untuk memajukan komputasi berbasis jaringan dan layanan web dengan teknologi baru yang sedang berkembang. Di sisi HPC, superkomputer (prosesor paralel masif atau MPP) secara bertahap digantikan oleh kluster komputer kooperatif karena keinginan untuk berbagi sumber daya komputasi. Kluster sering kali merupakan kumpulan node komputasi homogen yang terhubung secara fisik dalam jarak dekat satu sama lain.



Gambar 2.7 *Evolusi Platform*

Di sisi HTC, jaringan peer-to-peer (P2P) dibentuk untuk berbagi berkas terdistribusi dan aplikasi pengiriman konten.



Gambar 2.8 *Jaringan Peer to Peer*

Sistem P2P dibangun di atas banyak mesin klien. Mesin peer didistribusikan secara global. Platform P2P, komputasi awan, dan layanan web lebih berfokus pada aplikasi HTC daripada aplikasi HPC. Teknologi pengelompokan dan P2P mengarah pada pengembangan jaringan komputasi atau jaringan data.

2.3.2 Komputasi Kinerja Tinggi

Selama bertahun-tahun, sistem HPC menekankan kinerja kecepatan mentah. Kecepatan sistem HPC telah meningkat dari Gflops pada awal 1990-an menjadi Pflops pada tahun 2010. Peningkatan ini terutama didorong oleh permintaan dari komunitas ilmiah, teknik, dan manufaktur. Misalnya, 500 sistem komputer paling canggih di dunia diukur dengan kecepatan floating-point dalam hasil benchmark Linpack. Namun, jumlah pengguna superkomputer terbatas hingga kurang dari 10% dari semua pengguna komputer. Saat ini, mayoritas pengguna komputer menggunakan komputer desktop atau server besar saat mereka melakukan pencarian Internet dan tugas komputasi yang digerakkan pasar.

Pengembangan sistem komputasi canggih yang berorientasi pasar tengah mengalami perubahan strategis dari paradigma HPC ke paradigma HTC. Paradigma HTC ini lebih memperhatikan komputasi fluks tinggi. Aplikasi utama komputasi fluks tinggi adalah dalam pencarian Internet dan layanan web oleh jutaan atau lebih pengguna secara bersamaan. Dengan demikian, sasaran kinerja bergeser untuk mengukur throughput tinggi atau jumlah tugas yang diselesaikan per unit waktu. Teknologi HTC tidak hanya perlu ditingkatkan dalam hal kecepatan pemrosesan batch, tetapi juga mengatasi masalah akut biaya, penghematan energi, keamanan, dan keandalan di banyak pusat komputasi data dan perusahaan. Tiga Paradigma Komputasi.

Kemajuan dalam virtualisasi memungkinkan pertumbuhan awan internet sebagai paradigma komputasi baru. Kematangan identifikasi frekuensi radio (RFID), Sistem Pemosisian Global (GPS), dan teknologi sensor telah memicu pengembangan Internet of Things (IoT).

Gambar 2.9 tiga Paragdima komputasi Baru

Perbedaan Paradigma Komputasi

Komunitas teknologi tinggi telah berdebat selama bertahun-tahun tentang definisi yang tepat dari komputasi terpusat, komputasi paralel, komputasi terdistribusi, dan komputasi awan. Komputasi terdistribusi adalah kebalikan dari komputasi terpusat. Bidang komputasi paralel tumpang tindih dengan komputasi terdistribusi hingga tingkat yang besar, dan komputasi awan tumpang tindih dengan komputasi terdistribusi, terpusat, dan paralel. Komputasi terpusat Ini adalah paradigma komputasi di mana semua sumber daya komputer dipusatkan dalam satu sistem fisik. Semua sumber daya (prosesor, memori, dan penyimpanan) sepenuhnya dibagi dan digabungkan erat dalam satu OS terintegrasi. Banyak pusat data dan superkomputer adalah sistem terpusat, tetapi mereka digunakan dalam aplikasi komputasi paralel, terdistribusi, dan awan. Komputasi paralel Dalam komputasi paralel, semua prosesor digabungkan erat dengan memori bersama terpusat atau digabungkan longgar dengan memori terdistribusi. Beberapa penulis menyebut disiplin ini sebagai pemrosesan paralel. Komunikasi antarprosesor dicapai melalui memori bersama atau melalui pengiriman pesan. Sistem komputer yang mampu melakukan komputasi paralel umumnya dikenal sebagai komputer paralel [28]. Program yang berjalan di komputer paralel disebut program paralel. Proses penulisan program paralel sering disebut sebagai pemrograman paralel

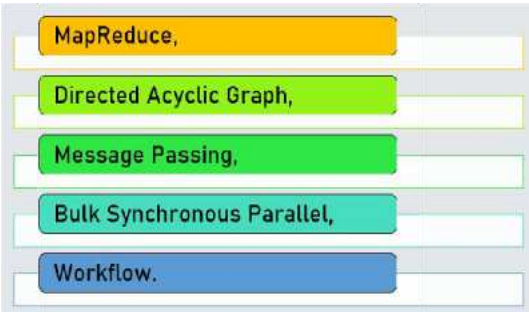
- Keluarga Sistem Terdistribusi

Sejak pertengahan 1990-an, teknologi untuk membangun jaringan P2P dan jaringan kluster telah dikonsolidasikan ke dalam banyak proyek nasional yang dirancang untuk membangun infrastruktur komputasi area luas, yang dikenal sebagai grid komputasional atau grid data. Baru-baru ini, kita telah menyaksikan lonjakan minat dalam mengeksplorasi sumber daya cloud Internet untuk aplikasi yang intensif

data. Cloud Internet adalah hasil dari pemindahan komputasi desktop ke komputasi berorientasi layanan dengan menggunakan kluster server dan basis data besar di pusat data. Grid dan cloud adalah sistem disparitas yang sangat menekankan pada pembagian sumber daya dalam perangkat keras, perangkat lunak, dan set data. Derajat Paralelisme

Lima puluh tahun yang lalu, ketika perangkat keras berukuran besar dan mahal, sebagian besar komputer dirancang dengan gaya bit-serial. Dalam skenario ini, paralelisme tingkat bit (BLP) mengubah pemrosesan bit-serial menjadi pemrosesan tingkat kata secara bertahap. Selama bertahun-tahun, pengguna beralih dari mikroprosesor 4-bit ke CPU 8-, 16-, 32-, dan 64-bit. Hal ini membawa kita ke gelombang perbaikan berikutnya, yang dikenal sebagai paralelisme tingkat instruksi (ILP), di mana prosesor mengeksekusi beberapa instruksi secara bersamaan, bukan hanya satu instruksi pada satu waktu. Selama 30 tahun terakhir, kami telah mempraktikkan ILP melalui pipelining, komputasi superskalar, arsitektur VLIW (very long instruction word), dan multithreading. ILP memerlukan prediksi cabang, penjadwalan dinamis, spekulasi, dan dukungan kompiler agar dapat bekerja secara efisien.

2.4 Model populer untuk big data



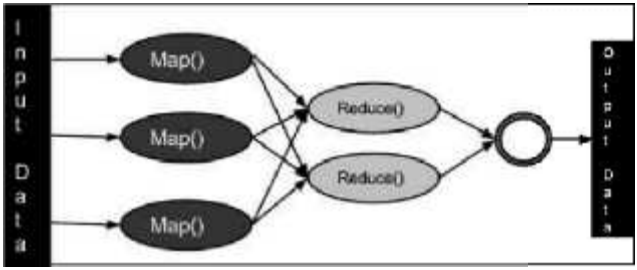
Gambar2.10 *Model populer untuk big data*

Apa itu MapReduce?

Map mengambil satu set data dan mengubahnya menjadi set data lain, di mana elemen-elemen individual dipecah menjadi tuple (pasangan kunci/nilai). Kedua, tugas reduce, yang mengambil output dari map sebagai input dan menggabungkan tuple data tersebut menjadi satu set tuple yang lebih kecil. Seperti yang tersirat dalam urutan nama MapReduce, tugas reduce selalu dilakukan setelah pekerjaan map. Kedua, tugas reduce, yang mengambil output dari map sebagai input dan menggabungkan tuple data tersebut menjadi satu set tuple yang lebih kecil. Seperti yang tersirat dalam urutan nama MapReduce, tugas reduce selalu dilakukan setelah pekerjaan map. Algoritma Secara umum, paradigma MapReduce didasarkan pada pengiriman komputer ke tempat data berada. Program MapReduce dijalankan dalam tiga tahap, yaitu tahap pemetaan, tahap pengacakan, dan tahap pengurangan.

- **Tahap peta**− Tugas mapper adalah memproses data input. Umumnya, data input berbentuk file atau direktori dan disimpan dalam sistem file Hadoop (HDFS). File input diteruskan ke fungsi mapper baris demi baris. Mapper memproses data dan membuat beberapa potongan data kecil.
- **Kurangi tahap**− Tahap ini merupakan gabungan dari tahap Shuffle dan tahap Reduce. Tugas Reducer adalah memproses data yang berasal dari mapper. Setelah diproses, ia menghasilkan serangkaian output baru, yang akan disimpan dalam HDFS.

Selama pekerjaan MapReduce, Hadoop mengirimkan tugas Map dan Reduce ke server yang sesuai di kluster.



Gambar2.11 MapReduce

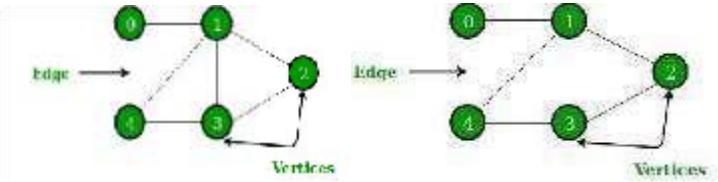
Kerangka kerja tersebut mengelola semua detail pengiriman data seperti penerbitan tugas, verifikasi penyelesaian tugas, dan penyalinan data di sekitar kluster di antara node. Sebagian besar komputasi dilakukan pada node dengan data pada disk lokal yang mengurangi lalu lintas jaringan. Setelah menyelesaikan tugas yang diberikan, kluster mengumpulkan dan mengurangi data untuk membentuk hasil yang sesuai, dan mengirimkannya kembali ke server Hadoop.

Keuntungan MapReduce

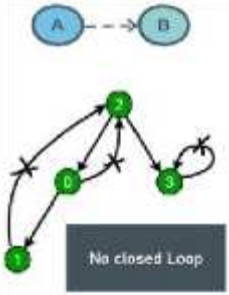
- Mudah untuk menskalakan pemrosesan data pada beberapa node komputasi.
- Di bawah model MapReduce, primitif pemrosesan data disebut mapper dan reducer.
- Menguraikan aplikasi pemrosesan data menjadi pemeta dan pereduksi terkadang bukan hal yang mudah.
- Namun, begitu kita menulis aplikasi dalam bentuk MapReduce, penskalaan aplikasi agar berjalan pada ratusan, ribuan, atau bahkan puluhan ribu mesin dalam satu klaster hanyalah sekadar perubahan konfigurasi.
- Skalabilitas sederhana inilah yang menarik banyak programmer untuk menggunakan model MapReduce.

Grafik Asiklik Berarah

Dalam ilmu komputer dan matematika, graf asiklik terarah (DAG) merujuk pada graf terarah yang tidak memiliki siklus terarah. Dalam teori graf, graf merujuk pada sekumpulan titik sudut yang dihubungkan oleh garis yang disebut tepi.

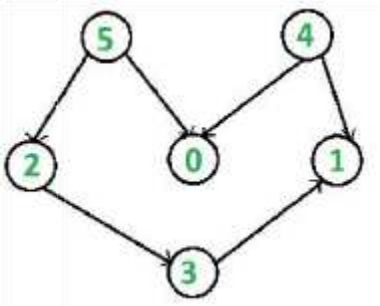


Dalam graf berarah atau digraf, setiap sisi dikaitkan dengan arah dari titik awal ke titik akhir. Jika kita melintasi sepanjang arah sisi dan kita menemukan bahwa tidak ada loop tertutup yang terbentuk di sepanjang lintasan apa pun, kita katakan bahwa tidak ada siklus berarah. Grafik yang terbentuk adalah grafik asiklik berarah. DAG selalu diurutkan secara topologi, yaitu, untuk setiap sisi dalam grafik, titik awal sisi muncul lebih awal dalam urutan daripada titik akhir sisi.



Pengurutan topologi untuk Directed Acyclic Graph (DAG) adalah pengurutan linear dari titik-titik sedemikian rupa sehingga untuk setiap sisi berarah uv, titik u muncul sebelum v dalam pengurutan tersebut. Pengurutan Topologi untuk sebuah grafik tidak mungkin dilakukan jika grafik tersebut bukan DAG. Misalnya, pengurutan topologi dari grafik

berikut adalah “5 4 2 3 1 0”. Mungkin ada lebih dari satu pengurutan topologi untuk sebuah grafik. Misalnya, pengurutan topologi lain dari grafik berikut adalah “4 5 2 3 1 0”. Titik pertama dalam pengurutan topologi selalu merupakan titik dengan derajat masuk sebagai 0 (titik tanpa sisi masuk).



Gambar2. 12 *Sortiran Topologi*

Bidang Aplikasi

Beberapa area aplikasi utama DAG adalah :

- Routing dalam jaringan komputer
- Penjadwalan pekerjaan
- Pengolahan data
- Silsilah
- Grafik kutipan

Pesan yang disampaikan

Komunikasi proses adalah mekanisme yang disediakan oleh sistem operasi yang memungkinkan proses untuk berkomunikasi satu sama lain. Komunikasi ini dapat melibatkan suatu proses yang memberi tahu proses lain bahwa suatu peristiwa telah terjadi atau mentransfer data dari satu proses ke proses lainnya. Salah satu model komunikasi proses adalah

model penyampaian pesan. Model penyampaian pesan memungkinkan beberapa proses untuk membaca dan menulis data ke antrean pesan tanpa terhubung satu sama lain. Pesan disimpan di antrean hingga penerima mengambilnya. Antrean pesan cukup berguna untuk komunikasi antarproses dan digunakan oleh sebagian besar sistem operasi.

Dalam diagram di atas, kedua proses P1 dan P2 dapat mengakses antrean pesan dan menyimpan serta mengambil data. Model penyampaian pesan jauh lebih mudah diimplementasikan daripada model memori bersama. Lebih mudah membangun perangkat keras paralel menggunakan model penyampaian pesan karena cukup toleran terhadap latensi komunikasi yang lebih tinggi.

Kerugian dari Message Passing

Model penyampaian pesan memiliki komunikasi yang lebih lambat daripada model memori bersama karena pengaturan koneksi memerlukan waktu.

Sinkron Massal Paralel

Meskipun tersebar luas, model MapReduce bukannya tanpa kekurangan. Ketika kita berbicara tentang model yang diterapkan dalam konteks Hadoop, misalnya, semua langkah klaster dan pemasangan massa akhir dengan hasil dilakukan melalui berkas pada sistem berkas Hadoop, HDFS, yang menghasilkan overhead dalam kinerja ketika harus melakukan pemrosesan yang sama secara berulang. Masalah lainnya adalah bahwa untuk algoritme grafik seperti DFS, BFS atau Pert, model MapReduce tidak memuaskan. Untuk skenario ini, ada BSP. Dalam algoritme BSP, kita memiliki konsep langkah super.

Langkah super terdiri dari unit pemrograman generik, yang melalui komponen komunikasi global, membuat ribuan pemrosesan paralel pada

sejumlah besar data dan mengirimkannya ke “pertemuan” yang disebut penghalang sinkronisasi.

Pada titik ini, data dikelompokkan dan diteruskan ke rantai superstep berikutnya.

Dalam model ini, lebih mudah untuk membangun beban kerja berulang, karena logika yang sama dapat dijalankan ulang dalam aliran langkah-langkah super. Keuntungan lain yang ditunjukkan oleh para pendukung model ini adalah kurva pembelajarannya lebih sederhana bagi pengembang yang berasal dari dunia prosedural.

Jadi, kita simpulkan bagian lain dari seri kita tentang Big Data. Hingga saat ini, berikut adalah model-model utama yang digunakan oleh platform Big Data. Sebagai teknologi yang sedang berkembang pesat, wajar saja jika di masa mendatang kita akan memiliki lebih banyak model yang muncul dan mendapatkan pangsa adopsi.

2.5 Lima Alasan Anda Memerlukan Pendekatan Langkah demi Langkah untuk Orkestrasi Alur Kerja untuk Big Data

Apakah organisasi Anda kesulitan untuk memenuhi tuntutan Big Data dan berada di bawah tekanan untuk membuktikan hasil yang cepat? Jika demikian, Anda tidak sendirian. Menurut analis, hingga 60% proyek Big Data gagal karena tidak dapat ditingkatkan pada tingkat perusahaan. Untungnya, mengambil pendekatan langkah demi langkah untuk orkestrasi alur kerja aplikasi dapat membantu Anda berhasil. Dimulai dengan menilai berbagai teknologi untuk mendukung beberapa proyek Big Data yang terkait dengan empat langkah ini:

- Mengonsumsi data
- Menyimpan data
- Memprosesnya
- Membuat data tersedia untuk analitik

Pendekatan ini juga memerlukan alat orkestrasi alur kerja aplikasi yang andal yang menyederhanakan kompleksitas alur kerja Big Data,

menghindari silo otomatisasi, menghubungkan proses, dan mengelola alur kerja dari satu titik. Hal ini memungkinkan Anda melakukan otomatisasi, integrasi, dan orkestrasi menyeluruh dari proses Big Data Anda, memastikan bahwa semuanya berjalan dengan sukses, memenuhi semua SLA, dan memberikan wawasan kepada pengguna bisnis tepat waktu. Menggabungkan alat otomatisasi dan orkestrasi yang berbeda yang tidak dapat diskalakan, dapat menyebabkan penundaan dan membahayakan seluruh proyek. Berikut adalah beberapa manfaat memulai proyek Big Data Anda dengan mempertimbangkan orkestrasi alur kerja aplikasi dan menggunakan alat yang mendukung langkah-langkah ini:

- Meningkatkan kualitas, kecepatan, dan waktu pemasaran

Banyak proyek Big Data yang tertunda atau gagal. Jika pengembang tidak memiliki alat untuk meningkatkan skala upaya mereka dengan benar, mereka mungkin menulis banyak skrip yang sulit dikelola atau mengandalkan alat dengan fungsionalitas terbatas untuk penjadwalan. Alat mereka mungkin tidak terintegrasi dengan baik dengan proses lain, seperti transfer file. Dengan solusi orkestrasi beban kerja, Anda dapat mengimplementasikan proyek Big Data dengan cepat untuk membantu mempertahankan basis pelanggan dan mempertahankan keunggulan kompetitif.

- Mengurangi kompleksitas di semua lingkungan – di tempat, hybrid, dan multi-cloud

Alat orkestrasi yang dapat mengotomatiskan, menjadwalkan, dan mengelola proses dengan sukses di seluruh komponen yang berbeda dalam proyek Big Data mengurangi kerumitan ini. Alat ini dapat mengelola langkah-langkah utama penyerapan data, penyimpanan data, pemrosesan data, dan akhirnya seluruh bagian analitik. Alat ini juga harus memberikan pandangan holistik dari berbagai komponen dan teknologi yang mereka

gunakan untuk mengorkestrasi alur kerja tersebut. Alur kerja Big Data biasanya terdiri dari berbagai langkah dengan berbagai teknologi dan banyak bagian yang bergerak. Anda perlu menyederhanakan alur kerja untuk mengirimkan proyek big data dengan sukses tepat waktu, terutama di cloud, yang merupakan platform pilihan untuk sebagian besar proyek Big Data. Namun, cloud menambah kerumitan, jadi solusi orkestrasi Anda harus bersifat agnostik platform, mendukung lingkungan lokal dan multi-cloud.

- Pastikan skalabilitas dan kurangi risiko

Seperti yang saya sebutkan sebelumnya, proyek Big Data harus dapat ditingkatkan skalanya, terutama saat Anda mulai beralih dari fase uji coba ke produksi. Proses untuk mengembangkan dan menerapkan pekerjaan Big Data perlu diotomatisasi dan dapat diulang. Setelah uji coba berjalan dengan sukses, bagian lain dari bisnis akan berupaya memanfaatkan proyek Big Data juga. Solusi orkestrasi beban kerja Anda harus memudahkan peningkatan skala dan mendukung permintaan bisnis yang terus meningkat.

- Mencapai Integrasi yang lebih baik

Solusi open-source otomatisasi Big Data umumnya memiliki kemampuan terbatas dan tidak memiliki fitur manajemen yang penting. Lebih dari itu, solusi tersebut cenderung terbatas pada lingkungan tertentu (misalnya Hadoop), tetapi perlu diingat bahwa Big Data bukanlah sebuah pulau. Big Data sering kali perlu diintegrasikan dengan bagian lain dari bisnis. Jadi, proyek Big Data Anda harus terhubung dengan aplikasi, platform, dan sumber data hulu dan hilir (misalnya sistem ERP, EDW, dll.) Solusi orkestrasi big data kami harus menyediakan kemampuan ini.

- Meningkatkan keandalan

penting untuk menjalankan alur kerja Big Data dengan sukses guna meminimalkan gangguan layanan. Menggunakan berbagai alat dan proses membuat sulit untuk mengidentifikasi masalah dan memahami akar penyebabnya, sehingga membahayakan SLA. Jika Anda dapat mengelola seluruh alur kerja Big Data dari A hingga Z, maka jika terjadi kesalahan dalam proses, Anda akan segera melihatnya dan mengetahui di mana kesalahan itu terjadi dan apa yang terjadi. Menggunakan solusi yang sama untuk mengatur seluruh proses dan mengelolanya dari satu bidang pandang, menyederhanakan pengelolaan layanan Anda dan memastikannya berjalan dengan sukses.

- Melihat ke depan

Mengambil pendekatan langkah demi langkah untuk orkestrasi alur kerja aplikasi menyederhanakan kompleksitas alur kerja Big Data Anda. Pendekatan ini menghindari silo otomatisasi dan membantu memastikan Anda memenuhi SLA dan memberikan wawasan kepada pengguna bisnis tepat waktu. Temukan bagaimana Control-M menyediakan semua kemampuan untuk memungkinkan organisasi Anda mengikuti pendekatan ini dan bagaimana pendekatan ini mudah diintegrasikan dengan teknologi Anda yang sudah ada untuk mendukung proyek Big Data.

Ringkasan

Sistem berkas adalah program yang mengendalikan bagaimana dan di mana data disimpan, diambil, dan dikelola pada cakram penyimpanan, biasanya hard disk drive (HDD). Ini adalah komponen cakram logis yang mengelola aktivitas internal cakram yang berkaitan dengan komputer namun tetap tidak terlihat oleh pengguna.

Sistem berkas terdistribusi (DFS) atau sistem berkas jaringan adalah jenis sistem berkas yang memungkinkan banyak host untuk berbagi berkas melalui jaringan komputer. Beberapa pengguna di beberapa mesin dapat berbagi data dan sumber daya penyimpanan sebagai hasilnya. Perbedaan antara teknik akses lokal dan jarak jauh seharusnya tidak dapat dibedakan. Pengguna yang memiliki akses ke layanan komunikasi serupa di beberapa lokasi disebut sebagai pengguna seluler. Misalnya, pengguna dapat menggunakan telepon pintar dan mengakses akun emailnya dari komputer mana pun untuk memeriksa atau menulis email. Perjalanan perangkat komunikasi dengan atau tanpa pengguna disebut sebagai portabilitas perangkat.

Big data mengacu pada data dalam jumlah besar yang sulit dikelola baik yang terorganisasi maupun tidak terstruktur – yang membanjiri perusahaan setiap hari. Big data dapat dievaluasi untuk mendapatkan wawasan yang membantu orang membuat penilaian yang lebih baik dan merasa lebih percaya diri dalam membuat keputusan bisnis yang penting. Berikut ini adalah aplikasi Big Data yang paling mendasar dan mendasar. Aplikasi ini membantu meningkatkan efisiensi perusahaan di hampir setiap industri. Ini adalah aplikasi big data masa depan. Aplikasi ini berpotensi mengubah bisnis dan meningkatkan efektivitas perusahaan. Big data dapat diatur dan dianalisis untuk mengungkap pola dan wawasan yang dapat digunakan untuk meningkatkan kinerja perusahaan.

Proses replikasi molekul DNA untai ganda menjadi dua molekul DNA identik dikenal sebagai replikasi DNA. Karena setiap kali sel membelah, dua sel anak baru harus memiliki informasi genetik yang sama, atau DNA, seperti sel induk, maka replikasi diperlukan.

Kemampuan suatu sistem untuk meningkatkan atau menurunkan kinerja dan biaya sebagai respons terhadap perubahan dalam aplikasi dan permintaan pemrosesan sistem dikenal sebagai skalabilitas. Ketika

mempertimbangkan perangkat keras dan perangkat lunak, bisnis yang berkembang pesat harus memberikan perhatian khusus pada skalabilitas.

Dalam kluster Hadoop, MapReduce adalah paradigma pemrograman yang memungkinkan skalabilitas luar biasa di ratusan atau ribuan komputer. MapReduce, sebagai komponen pemrosesan, merupakan inti dari Apache Hadoop. Pekerjaan reduksi selalu dilakukan setelah pekerjaan pemetaan, sebagaimana yang tersirat dalam istilah MapReduce.

Soal Latihan

Q1: Ekstensi EXE adalah singkatan dari _____

A. file yang dapat dieksekusi
B. file ekstensi
C. file yang diperluas
D. Tidak ada yang di atas

Q2: Pilih komponen sistem berkas terdistribusi

A. Transparansi dan redundansi istilah
B. Transparansi dan redundansi lokasi
C. Transparansi lokasi dan transparansi istilah
D. Tidak ada yang di atas

Q3: Replikasi data merupakan cara yang baik untuk mencapai _____ dan konkurensi tinggi; tetapi sangat sulit untuk mempertahankan perubahan yang sering.

A. toleransi kesalahan
B. toleransi deteksi
C. keduanya
D. tidak ada yang di atas

Q4: Pilih paradigma komputasi baru

A. Teknologi RFID
B. Teknologi sensor
C. Bahasa Indonesia: GPS
D. Semua hal di atas

Q5: Jenis file dilambangkan dengan _____

A. Nama berkas
B. Pengidentifikasi berkas
C. Ekstensi berkas
D. Tidak satu pun yang disebutkan.

Q6: Teknologi komputer apa yang digunakan untuk menjelaskan layanan dan aplikasi yang berjalan pada jaringan terdistribusi menggunakan sumber daya virtual?

A. Komputasi Terdistribusi
B. Komputasi Awan
C. Komputasi Lunak
D. Komputasi Paralel

Q7: Manakah dari pilihan berikut yang dapat dianggap sebagai Cloud?

A. Bahasa Indonesia: Hadoop
B. Jaringan dalam
C. Aplikasi Web
D. Semua yang disebutkan

Q8: Model populer untuk big data adalah

A. Grafik Asiklik Berarah
B. Pesan yang disampaikan
C. Alur kerja
D. Semua di atas

Q9: Nama tiga tahap di mana program MapReduce dieksekusi adalah:

A. Tahap peta, tahap acak, dan tahap pengurangan
B. tahap shuffle, tahap map dan tahap reduce C. tahap reduce, tahap shuffle, dan tahap map
D. Tidak satu pun yang benar.

Q10: Grafik asiklik terarah (DAG) merujuk pada grafik terarah yang tidak memiliki siklus _____.

A. Tak terbatas
B. Diarahkan
C. Arah
D. Tidak ada di atas

Q11: Pilih dua tugas penting dari algoritma mapreduce

A. Peta
B. Mengurangi
C. Keduanya
D. Tidak ada yang di atas

Q12: Area aplikasi utama dari Directed Acyclic Graph adalah

A. Silsilah
B. Grafik kutipan
C. Penjadwalan Pekerjaan
D. Semua di atas

Q13: Bentuk lengkap dari BSP adalah

A. Paket Sinkron Massal
B. Botak Sinkron Paralel
C. Sakelar Massal Paralel
D. Sinkron Massal Paralel

Q14: Peta mengambil sekumpulan data dan mengubahnya menjadi sekumpulan data lain, di mana elemen-elemen individual dipecah menjadi _____

A. Tabel.
B. Tuple (pasangan kunci/nilai).
C. Kurangi panggung.

D. Tidak satu pun yang di atas.

Q15: HDFS adalah singkatan dari

A. Sistem berkas Hadoop
B. Sistem datar Hadoop
C. Peralihan berkas Hadoop
D. Tidak ada yang di atas

Soal Essay

1. Bedakan antara sistem berkas dan sistem berkas terdistribusi.
2. Tuliskan ciri-ciri sistem berkas terdistribusi.
3. Tuliskan model-model BIG DATA yang populer.
4. Tuliskan tantangan BIG DATA.
5. Tulis catatan tentang berikut ini.

a. Era Komputasi Internet
b. Komputasi Throughput Tinggi

6. Apa keuntungan dan kerugian sistem berkas terdistribusi?....

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121.

https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

BAB 3: Model Data

Karno Diantoro, S. Kom, M.Kom

Tujuan

- Memahami apa itu data mart
- Memahami format data
- Memahami model data
- Bedakan antara gudang data dan data mart
- Memahami apa itu aliran data
- memahami data sensor streaming Pendahuluan

Perkenalan

Data mart adalah versi lebih kecil dari gudang data yang memenuhi persyaratan analisis data khusus. Data mart sering kali merupakan bagian dari gudang data yang lebih besar. Sasaran utama data mart adalah melakukan analisis yang sulit dilakukan di gudang data tradisional karena tingkat ketelitian data yang berbeda-beda atau kebutuhan untuk melakukan perhitungan yang canggih.

3.1 Apa itu format data?

Aplikasi perlu dapat menyetujui beberapa jenis sintaksis agar dapat bertukar data di antara mereka.

Contoh: Jika kita memberikan kuliah dalam bahasa Prancis untuk para mahasiswa yang hanya mengerti bahasa Inggris. Itu berarti protokol belum diikuti. Demikian pula, aplikasi juga perlu mengikuti beberapa sintaksis dan protokol agar dapat saling bertukar data. Untuk ini, aplikasi dapat menggunakan format data standar seperti JSON dan XML. Aplikasi tidak perlu menyetujui bagaimana data diformat dan bagaimana data disusun. Sama halnya dengan router dan switch yang memerlukan

protokol standar agar dapat berkomunikasi, aplikasi harus dapat menyetujui beberapa jenis sintaksis agar dapat saling bertukar data.

Untuk ini, aplikasi dapat menggunakan format data standar seperti JSON dan XML (antara lain). Aplikasi tidak hanya perlu menyetujui bagaimana data diformat, tetapi juga bagaimana data tersebut terstruktur. Model data menentukan bagaimana data yang disimpan dalam format data tertentu terstruktur.

Format Data

- Seorang programmer komputer biasanya menggunakan berbagai macam alat untuk menyimpan dan bekerja dengan data dalam program yang mereka buat. Mereka mungkin menggunakan variabel sederhana (nilai tunggal), array (beberapa nilai), hash (pasangan kunci-nilai), atau bahkan objek khusus yang dibuat dalam sintaksis bahasa yang mereka gunakan. Format portabel diperlukan.
- Program lain mungkin harus berkomunikasi dengan program ini dengan cara yang sama, dan program tersebut bahkan mungkin tidak ditulis dalam bahasa yang sama, seperti yang sering terjadi pada komunikasi klien-server tradisional seperti yang ditunjukkan pada Gambar 1.

Gambar3. 1 *Aplikasi Klien/Server Tradisional*

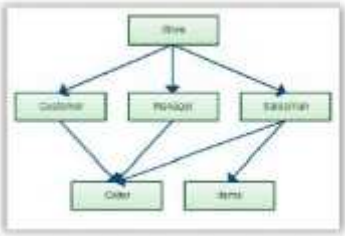
- Ini semua merupakan standar yang sempurna dalam batasan perangkat lunak yang sedang ditulis. Namun, terkadang format yang lebih abstrak dan portabel diperlukan. Misalnya, orang yang bukan

programmer mungkin perlu memindahkan data masuk dan keluar dari program ini.

- Misalnya, banyak antarmuka pengguna (UI) pihak ketiga digunakan untuk berinteraksi dengan penyedia cloud publik. Hal ini dimungkinkan (dengan cara yang disederhanakan) berkat format data standar. Inti ceritanya adalah kita memerlukan format standar untuk memungkinkan beragam perangkat lunak berkomunikasi satu sama lain, dan agar manusia dapat berinteraksi dengannya.

3.2 Apa itu Model Data

Di perpustakaan, kita perlu mengklasifikasikan semua buku dan menatanya di rak untuk memastikan kita dapat mengakses setiap buku dengan mudah. Demikian pula, jika kita memiliki data dalam jumlah besar, kita memerlukan sistem atau metode untuk menjaga semuanya tetap teratur. Proses penyortiran dan penyimpanan data disebut "pemodelan data." Model data adalah metode yang dapat kita gunakan untuk mengatur dan menyimpan data. Sama seperti Sistem Desimal Dewey yang mengatur buku-buku di perpustakaan, model data membantu kita mengatur data menurut layanan, akses, dan penggunaan. Torvalds, pendiri Linux, menyinggung pentingnya pemodelan data saat ia menulis artikel tentang "apa yang membuat seorang programmer hebat": "Programmer yang buruk peduli dengan kode, dan programmer yang baik peduli dengan struktur data dan hubungan antar data." Model dan lingkungan penyimpanan yang tepat menawarkan manfaat berikut untuk big data:



Mengatur dan Menyimpan Data

Mengatur buku-buku

- Model data secara eksplisit menentukan struktur data.
- Model data ditentukan dalam notasi pemodelan data (link is external) yang sering kali berbentuk grafis.
- Model data kadang-kadang dapat disebut sebagai struktur data (tautan eksternal), terutama dalam konteks bahasa pemrograman (tautan eksternal).

Bagaimana Model Data Dibangun?

Model data dibangun menggunakan komponen-komponen yang bertindak sebagai abstraksi dari berbagai hal di dunia nyata.

Model data yang paling sederhana terdiri dari entitas dan hubungan. Seiring berjalannya pengerjaan model data, detail dan kompleksitas tambahan ditambahkan, termasuk atribut, domain, kendala, kunci, kardinalitas, persyaratan, hubungan—dan yang terpenting, definisi dari semua hal dalam model data. Jika kita ingin memahami data yang kita miliki—dan cara menggunakannya—diperlukan model dasar.

3.3 Manfaat Model dan Lingkungan Penyimpanan yang Tepat untuk Big Data

- Kinerja: Model data yang baik dapat membantu kita dengan cepat meminta data yang diperlukan dan mengurangi throughput I/O.

- **Biaya:**Model data yang baik dapat secara signifikan mengurangi redundansi data yang tidak diperlukan, menggunakan kembali hasil komputasi, dan mengurangi biaya penyimpanan dan komputasi untuk sistem data besar.
- **Efisiensi:**Model data yang baik dapat meningkatkan pengalaman pengguna dan meningkatkan efisiensi pemanfaatan data.
- **Kualitas:**Model data yang baik membuat statistik data lebih konsisten dan mengurangi kemungkinan kesalahan komputasi.

Oleh karena itu, tidak diragukan lagi bahwa sistem data besar memerlukan metode pemodelan data berkualitas tinggi untuk mengatur dan menyimpan data, yang memungkinkan kita mencapai keseimbangan optimal antara kinerja, biaya, efisiensi, dan kualitas.

Tips untuk membuat model big data yang efektif

Pemodelan data adalah ilmu kompleks yang melibatkan pengorganisasian data perusahaan sehingga sesuai dengan kebutuhan proses bisnis. Hal ini memerlukan desain hubungan logis sehingga data dapat saling berhubungan dan mendukung bisnis. Desain logis kemudian diterjemahkan ke dalam model fisik yang terdiri dari perangkat penyimpanan, basis data, dan file yang menampung data. Secara historis, bisnis telah menggunakan teknologi basis data relasional seperti SQL untuk mengembangkan model data karena sangat cocok untuk secara fleksibel menghubungkan kunci set data dan tipe data bersama-sama untuk mendukung kebutuhan informasi proses bisnis. Sayangnya, big data, yang sekarang mencakup sebagian besar data yang dikelola, tidak berjalan pada basis data relasional. Ia berjalan pada basis data non-relasional seperti NoSQL. Hal ini mengarah pada keyakinan bahwa Anda tidak memerlukan model untuk big data. Masalahnya adalah, Anda memang memerlukan pemodelan data untuk big data.

Jangan mencoba menerapkan teknik pemodelan tradisional pada big data

Data rekaman tetap tradisional stabil dan dapat diprediksi pertumbuhannya. Hal ini membuatnya relatif mudah untuk dimodelkan. Sebaliknya, pertumbuhan eksponensial big data tidak dapat diprediksi, begitu pula berbagai bentuk dan sumbernya. Ketika situs mempertimbangkan pemodelan big data, upaya pemodelan harus berpusat pada pembangunan antarmuka data yang terbuka dan elastis, karena Anda tidak pernah tahu kapan sumber data atau bentuk data baru dapat muncul. Ini bukan prioritas dalam dunia data rekaman tetap tradisional. Rancang sistem, bukan skema

Dalam ranah data tradisional, skema basis data relasional dapat mencakup sebagian besar hubungan dan tautan antara data yang dibutuhkan bisnis untuk dukungan informasinya. Ini tidak berlaku untuk big data, yang mungkin tidak memiliki basis data, atau yang mungkin menggunakan basis data seperti NoSQL, yang tidak memerlukan skema basis data. Karena itu, model big data harus dibangun di atas sistem, bukan basis data. Komponen sistem yang harus dimiliki model big data adalah persyaratan informasi bisnis, tata kelola dan keamanan perusahaan, penyimpanan fisik yang digunakan untuk data, integrasi dan antarmuka terbuka untuk semua jenis data, dan kemampuan untuk menangani berbagai jenis data yang berbeda.

Cari alat pemodelan data besar

Ada alat pemodelan data komersial yang mendukung Hadoop, serta perangkat lunak pelaporan big data seperti Tableau. Saat mempertimbangkan alat dan metodologi big data, pengambil keputusan TI harus menyertakan kemampuan untuk membangun model data untuk big data sebagai salah satu persyaratan mereka.

Fokus pada data yang merupakan inti bisnis Anda

Segunung data besar mengalir ke perusahaan setiap hari, dan sebagian besar data ini tidak relevan. Tidak masuk akal untuk membuat model yang mencakup semua data. Pendekatan yang lebih baik adalah mengidentifikasi data besar yang penting bagi perusahaan Anda, dan memodelkan data tersebut.

Menyampaikan data berkualitas

Model data dan hubungan yang unggul dapat dihasilkan untuk big data jika organisasi berkonsentrasi pada pengembangan definisi yang baik untuk data dan metadata menyeluruh yang menjelaskan asal data, apa tujuannya, dll. Semakin banyak Anda mengetahui tentang setiap bagian data, semakin Anda dapat menempatkannya dengan tepat ke dalam model data yang mendukung bisnis Anda.

Mencari jalan masuk utama ke dalam data

Salah satu vektor yang paling umum digunakan dalam big data saat ini adalah lokasi geografis. Bergantung pada bisnis dan industri Anda, ada juga kunci umum lainnya dalam big data yang diinginkan pengguna. Semakin Anda dapat mengidentifikasi titik masuk umum ini ke dalam data Anda, semakin baik Anda akan dapat merancang model data yang mendukung jalur akses informasi utama untuk perusahaan Anda.

3.4 Apa itu data mart?

Ini adalah penyimpanan data yang dirancang untuk departemen tertentu dalam suatu organisasi, atau data mart adalah bagian dari Datawarehouse yang biasanya berorientasi pada tujuan tertentu. Data mart adalah bagian dari gudang data yang berorientasi pada lini bisnis tertentu. Data mart berisi repositori data ringkasan yang dikumpulkan untuk analisis pada bagian atau unit tertentu dalam suatu organisasi, misalnya, departemen penjualan. Gudang data adalah repositori data

terpusat yang besar yang berisi informasi dari banyak sumber dalam suatu organisasi. Data yang dikumpulkan digunakan untuk memandu keputusan bisnis melalui analisis, pelaporan, dan alat penambangan data. Alasan menggunakan data mart

- Akses mudah ke data yang sering diakses.
- Meningkatkan waktu respons pengguna akhir.
- Pembuatan data mart menjadi mudah.
- Biaya lebih rendah dalam membangun data mart.

3.5 Berbagai jenis data mart

3.5.1 Pasar data dependen

- Dalam hal ini, data mart dibangun dengan mengambil data dari gudang data pusat yang sudah ada. Data mart dependen memungkinkan sumber data organisasi dari satu Gudang Data. Ini adalah salah satu contoh data mart yang menawarkan manfaat sentralisasi. Jika Anda perlu mengembangkan satu atau lebih data mart fisik, maka Anda perlu mengonfigurasinya sebagai data mart dependen.
- Data mart dependen dapat dibangun dengan dua cara berbeda
- Baik di mana pengguna dapat mengakses data mart maupun gudang data, tergantung pada kebutuhan, atau di mana aksesnya hanya terbatas pada data mart.
- Pendekatan kedua tidaklah optimal karena menghasilkan sesuatu yang kadang-kadang disebut sebagai tempat pembuangan data. Di tempat pembuangan data, semua data berawal dari sumber yang sama, tetapi data tersebut dibuang, dan sebagian besar dibuang.

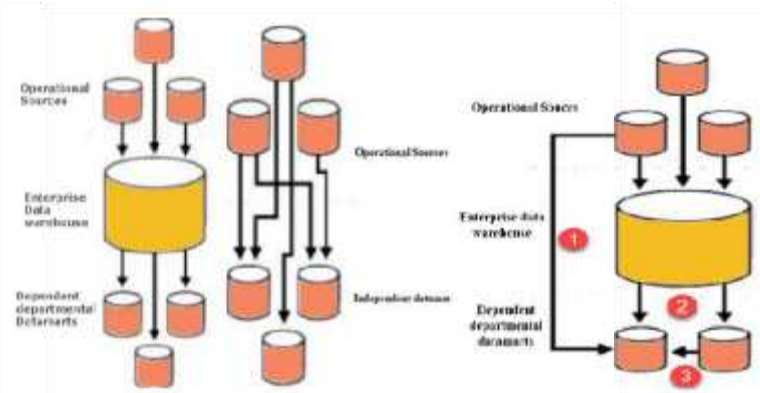
Pasar data independen

Dalam hal ini, data mart dibangun dengan mengambil data dari sumber operasional atau eksternal, atau keduanya.

Pasar Data Hibrida

Data mart hibrid menggabungkan input dari sumber selain dari gudang data. Ini adalah contoh data mart terbaik yang cocok untuk berbagai lingkungan basis data dan penyelesaian implementasi yang cepat untuk organisasi mana pun. Ini juga membutuhkan upaya pembersihan data yang paling sedikit. Data mart hibrid juga mendukung struktur penyimpanan yang besar, dan paling cocok untuk aplikasi yang berpusat pada data yang lebih kecil.

Tiga jenis datamart ditunjukkan pada Gambar 2.



Gambar3. 2 *Data mart independen, dependen dan hybrid*

Keuntungan Data Mart

- Lebih sederhana, lebih fokus & fleksibel. *Biaya rendah untuk perangkat keras dan perangkat lunak. Lebih cepat dan lebih murah untuk dibangun.*
- Menyimpan data lebih dekat yang meningkatkan kinerja. Kekurangan data mart
- Seringkali perusahaan membuat terlalu banyak data mart yang terpisah dan tidak berhubungan tanpa banyak manfaat. Hal ini dapat menjadi kendala besar untuk dipertahankan.

- Data Mart tidak dapat menyediakan analisis data untuk seluruh perusahaan karena kumpulan data mereka terbatas.
- Pembangunan tidak terorganisir
- Peningkatan ukuran datamart menimbulkan masalah seperti penurunan kinerja dan ketidakkonsistenan data.

Perbedaan antara Data Warehouse dan Data Mart

Data Warehouse	Data Mart
Data warehouse is a Centralised system.	While it is a Decentralised system
In data warehouse, lightly denormalization takes place.	While in Data mart, highly denormalization takes place
Data warehouse is top-down model.	While it is a bottom-up model.
To built a warehouse is difficult.	While to build a mart is easy
In data warehouse, Fact constellation schema is used.	While in this, Star schema and snowflake schema are used.

Data Warehouse	Data Mart
Data warehouse is a Centralised system.	While it is a Decentralised system
In data warehouse, lightly denormalization takes place.	While in Data mart, highly denormalization takes place
Data warehouse is top-down model.	While it is a bottom-up model.
To built a warehouse is difficult.	While to build a mart is easy
In data warehouse, Fact constellation schema is used.	While in this, Star schema and snowflake schema are used.

3.6 Apa Arti Streaming Big Data?

Streaming big data adalah proses di mana big data diproses dengan cepat untuk mengekstrak wawasan real-time darinya. Data yang diproses adalah data yang sedang bergerak. Streaming big data idealnya adalah pendekatan yang berfokus pada kecepatan di mana aliran data yang berkelanjutan diproses seperti yang ditunjukkan pada Gambar 3

Gambar 3.3 *Streaming data*

3.8 Aliran Data

Streaming data besar adalah proses di mana aliran besar data waktu nyata diproses dengan tujuan tunggal untuk mengekstrak wawasan dan tren yang berguna darinya. Aliran data tidak terstruktur yang berkelanjutan dikirim untuk analisis ke dalam memori sebelum menyimpannya ke disk. Ini terjadi di seluruh kluster server. Kecepatan paling penting dalam streaming data besar. Nilai data, jika tidak diproses dengan cepat, menurun seiring waktu. Dengan ledakan data dalam beberapa tahun terakhir, ada lebih banyak penekanan pada pengambilan keputusan berbasis data untuk semua perusahaan. Tetapi bagaimana jika kita dapat memproses data dan menindaklanjutinya secara waktu nyata? Bagaimana jika kita bisa proaktif alih-alih reaktif untuk meningkatkan kinerja? Streaming data dan analitik streaming sekarang memungkinkan itu bagi hampir setiap perusahaan di setiap industri untuk mendorong

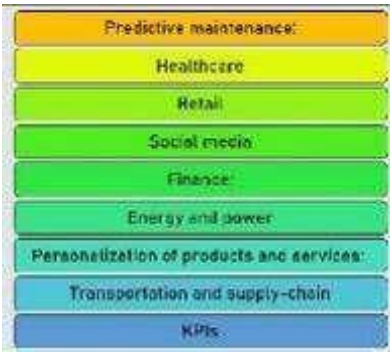
kecerdasan dan tindakan secara waktu nyata. Dan dengan dorongan yang dipercepat untuk otomatisasi dan digitalisasi di dunia pasca-virus corona, jelas bahwa perusahaan-perusahaan yang memanfaatkan data waktu nyata jauh lebih mungkin untuk mendapatkan keunggulan kompetitif atas pesaing mereka.

Kecepatan adalah hal terpenting dalam streaming data besar Nilai data, jika tidak diproses dengan cepat, akan menurun seiring waktu. Analisis data streaming waktu nyata merupakan analisis satu lintasan. Analis tidak dapat memilih untuk menganalisis ulang data setelah data tersebut dialirkan. Data dinamis yang dihasilkan secara terus-menerus dari berbagai sumber dianggap sebagai data streaming. Baik data tersebut berasal dari umpan media sosial atau sensor Gambar 4 dan kamera, setiap rekaman perlu diproses dengan cara yang menjaga hubungannya dengan data lain dan urutannya dari waktu ke waktu. File log, pembelian e-commerce, kejadian cuaca, penggunaan layanan utilitas, lokasi geografis orang dan benda, aktivitas server, dan banyak lagi adalah contoh di mana data streaming waktu nyata dibuat. Ketika perusahaan mampu menganalisis data streaming yang mereka terima, mereka bisa mendapatkan wawasan waktu nyata untuk memahami dengan tepat apa yang terjadi pada titik waktu tertentu. Hal ini memungkinkan pengambilan keputusan yang lebih baik serta menyediakan layanan yang lebih baik dan lebih personal bagi pelanggan. Hampir setiap perusahaan menggunakan atau dapat menggunakan data streaming.

Gamba3.4 *Media sosial dan sensor*

3.9 Kasus Penggunaan untuk Data Real-Time dan Streaming

Setiap organisasi yang digerakkan oleh data, yang saat ini hampir semuanya, dapat menggunakan data real-time dan streaming untuk meningkatkan hasil. Berikut ini adalah beberapa kasus penggunaan di berbagai industri:



Gambar3. 5 Kasus penggunaan untuk streaming data waktu nyata

Pemeliharaan Prediktif

Bila perusahaan dapat mengidentifikasi masalah pemeliharaan sebelum terjadi kerusakan atau kegagalan sistem, mereka akan menghemat waktu, uang, dan dampak bencana lainnya yang mungkin terjadi pada bisnis. Setiap perusahaan yang memiliki peralatan apa pun yang dilengkapi sensor atau kamera—sekali lagi, itulah sebagian besar peralatan saat ini—akan membuat data streaming. Mulai dari memantau kinerja truk dan pesawat terbang, hingga memprediksi masalah dengan peralatan manufaktur yang rumit, data dan analitik waktu nyata menjadi penting bagi perusahaan modern saat ini.

Gambar3. 6 Pemeliharaan prediktif

Pelayanan kesehatan

Sama seperti di lingkungan manufaktur, perangkat yang dapat dikenakan dan peralatan kesehatan seperti glukometer, timbangan yang terhubung, monitor detak jantung dan tekanan darah memiliki sensor yang memantau tanda-tanda vital dan fungsi tubuh penting pasien. Peralatan ini juga penting untuk pemantauan jarak jauh yang efektif. pemantauan pasien yang mendukung dokter yang tidak memiliki waktu untuk berada di mana-mana sepanjang waktu. Ini benar-benar masalah hidup dan mati. Wawasan langsung dapat meningkatkan hasil dan pengalaman pasien. Ritel Streaming data real-time dari sensor IoT dan video mendorong kebangkitan ritel modern. Toko ritel konvensional dapat melibatkan pelanggan saat itu juga berkat data streaming. Pemasaran berbasis lokasi, wawasan tren, dan peningkatan efisiensi operasional, seperti pergerakan produk atau kesegaran produk, semuanya dapat dilakukan dengan wawasan waktu nyata. Memahami apa yang diinginkan konsumen saat mereka menginginkannya "saat itu juga" tidak hanya penting dalam ritel. Setiap perusahaan yang mampu memahami dan segera menanggapi apa yang diinginkan pelanggannya dalam momen-momen mikro akan memiliki peluang lebih baik untuk sukses, baik itu untuk memberikan sesuatu yang ingin dipelajari, ditemukan, ditonton, atau dibeli oleh konsumen.

Media sosial

Dengan terus meningkatnya berita palsu dan kasus perundungan di media sosial, kebutuhan untuk memantau unggahan secara langsung untuk segera mengambil tindakan terhadap berita yang menyinggung dan "palsu" menjadi lebih penting dari sebelumnya. Di bawah tekanan yang meningkat, platform media sosial menciptakan berbagai alat untuk dapat memproses data dalam jumlah besar yang dibuat dengan cepat dan efisien agar dapat mengambil tindakan secepat mungkin, terutama untuk mencegah perundungan.

Keuangan

Di Rantai perdagangan, mudah untuk melihat betapa pentingnya memahami dan bertindak berdasarkan informasi secara real-time, tetapi streaming data juga membantu fungsi keuangan perusahaan mana pun dengan memproses informasi transaksional, mengidentifikasi tindakan penipuan, dan banyak lagi. Misalnya, MasterCard menggunakan data dan analitik untuk membantu organisasi keuangan mengidentifikasi pedagang yang melakukan penipuan dengan cepat dan mudah guna mengurangi risiko. Demikian pula, dengan memperoleh kemampuan untuk memproses data real-time, Rabobank mampu mendeteksi sinyal peringatan pada tahap yang sangat awal ketika klien mungkin gagal bayar.

Energi dan Tenaga

Di sektor energi, perusahaan berupaya mengoptimalkan bahan bakar fosil dan mengadopsi sistem tenaga yang lebih berkelanjutan. Aliran data yang berkelanjutan membantu pemeliharaan prediktif pada peralatan serta untuk lebih memahami permintaan konsumen dan meningkatkan bisnis dan operasi. Personalisasi produk dan layanan Perusahaan dapat merespons tuntutan konsumen untuk mendapatkan apa yang mereka inginkan (bahkan apa yang belum mereka ketahui) dengan lebih baik berkat data streaming. Dari publikasi berita daring yang menyajikan konten yang paling diminati pembaca tertentu hingga layanan streaming yang merekomendasikan hal-hal yang akan ditonton

berikutnya, personalisasi menambah nilai pada pengalaman pelanggan tetapi hanya mungkin dilakukan secara realtime karena data streaming.

Transportasi dan rantai pasokan

Data streaming memperkuat internet kereta api, memungkinkan kendaraan yang terhubung dan otonom menjadi lebih mungkin dan lebih aman, dan sangat penting dalam membuat operasi armada lebih efisien.

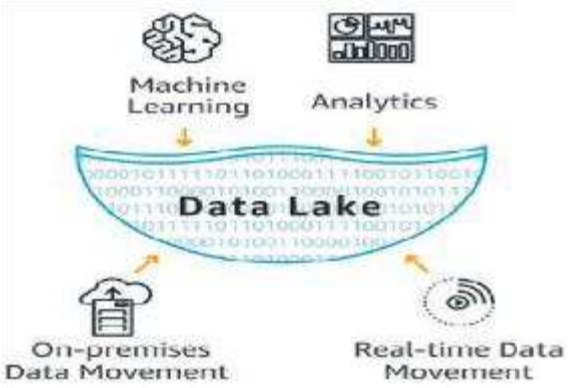
KPI

Pemimpin dapat membuat keputusan berdasarkan waktu nyata**KPI**seperti data kinerja keuangan, pelanggan, atau operasional. Sebelumnya, analisis ini bersifat reaktif dan melihat kembali kinerja masa lalu. Saat ini, data waktu nyata dapat dibandingkan dengan informasi historis untuk memberikan perspektif bisnis kepada para pemimpin yang menginformasikan keputusan waktu nyata. Seperti yang Anda lihat, data streaming semakin penting bagi sebagian besar perusahaan di sebagian besar industri. Perusahaan yang sukses mengintegrasikan analitik streaming untuk mengubah analitik data mereka dari pendekatan waktu nyata yang reaktif menjadi lebih proaktif. Yang terbaik akan berpikir untuk mengintegrasikan data waktu nyata mereka dengan model prediktif dan analisis skenario untuk mendapatkan pandangan ke depan yang strategis. Namun, untuk memanfaatkan data yang cepat dan streaming, organisasi saat ini memerlukan platform manajemen dan analitik data menyeluruh yang dapat mengumpulkan, memproses, mengelola, dan menganalisis data secara real-time untuk mendorong wawasan dan memungkinkan pembelajaran mesin untuk mengimplementasikan beberapa kasus penggunaan yang paling menarik. Yang terpenting, mereka harus dapat melakukan ini dengan keamanan, tata kelola, perlindungan data, dan kemampuan manajemen yang kuat yang dibutuhkan perusahaan.

3.10 Danau Data

Danau data adalah repositori terpusat yang memungkinkan Anda menyimpan semua data terstruktur dan tidak terstruktur dalam skala apa

pun. Anda dapat menyimpan data apa adanya, tanpa harus menyusun data terlebih dahulu, dan menjalankan berbagai jenis analitik—mulai dari dasbor dan visualisasi hingga pemrosesan big data, analitik real-time, dan pembelajaran mesin untuk memandu keputusan yang lebih baik seperti yang ditunjukkan pada Gambar 7.



Gambar 3.7 Data Lake

Mengapa Anda membutuhkan danau data?

Organisasi yang berhasil menghasilkan nilai bisnis dari data mereka akan mengungguli rekan-rekan mereka. Sebuah survei di Aberdeen menunjukkan bahwa organisasi yang menerapkan Data Lake mengungguli perusahaan sejenis sebesar 9% dalam pertumbuhan pendapatan organik. Para pemimpin ini mampu melakukan jenis analitik baru seperti pembelajaran mesin melalui sumber-sumber baru seperti berkas log, data dari aliran klik, media sosial, dan perangkat yang terhubung internet yang tersimpan di data lake. Hal ini membantu mereka mengidentifikasi dan bertindak atas peluang pertumbuhan bisnis lebih cepat dengan menarik dan mempertahankan pelanggan, meningkatkan produktivitas, memelihara perangkat secara proaktif, dan membuat keputusan yang tepat.

3.11 DataLake vs.Gudang Data

Bergantung pada persyaratan, organisasi pada umumnya akan memerlukan gudang data dan danau data karena keduanya melayani berbagai kebutuhan dan kasus penggunaan. Gudang data adalah basis data yang dioptimalkan untuk menganalisis data relasional yang berasal dari sistem transaksional dan aplikasi lini bisnis. Struktur data dan skema didefinisikan terlebih dahulu untuk mengoptimalkan kueri SQL yang cepat, yang hasilnya biasanya digunakan untuk pelaporan dan analisis operasional. Data dibersihkan, diperkaya, dan diubah sehingga dapat bertindak sebagai "satu-satunya sumber kebenaran" yang dapat dipercaya oleh pengguna.

Characteristics	Data Lakes	Datawarehouse
Data	Relational from transactional systems, operational databases, and line of business applications	Non-relational and relational from IoT devices, web sites, mobile apps, social media, and corporate applications
Schema	Designed prior to the DW implementation (schema-on-write)	Written at the time of analysis (schema-on-read)
Price/Performance	Fastest query results using higher cost storage	Query results getting faster using low-cost storage
Data Quality	Highly curated data that serves as the central version of the truth	Any data that may or may not be curated (ie. raw data)
Users	Business analysts	Data scientists, Data developers, and Business analysts (using curated data)
Analytics	Batch reporting, BI and visualizations	Machine Learning, Predictive analytics, data discovery and profiling

3.12 Elemen penting dari solusi Data Lake dan Analytics

Gambar 3.8 *Elemen penting dari sebuah data lake*

Pergerakan data

Data Lakes memungkinkan Anda mengimpor data dalam jumlah berapa pun yang dapat masuk secara real-time. Data dikumpulkan dari berbagai sumber, dan dipindahkan ke data lake dalam format aslinya. Proses ini memungkinkan Anda untuk menskalakan data dengan ukuran apa pun, sekaligus menghemat waktu dalam mendefinisikan struktur data, skema, dan transformasi.

Simpan dan katalogkan data dengan aman

Data Lakes memungkinkan Anda menyimpan data relasional seperti basis data operasional dan data dari berbagai aplikasi lini bisnis, serta data non-relasional seperti aplikasi seluler, perangkat IoT, dan media sosial. Data Lakes juga memberi Anda kemampuan untuk memahami data apa saja yang ada di dalam lake melalui perayapan, pengkatalogan, dan pengindeksan data. Terakhir, data harus diamankan untuk memastikan aset data Anda terlindungi.

Analitik

Data Lakes memungkinkan berbagai peran dalam organisasi Anda seperti ilmuwan data, pengembang data, dan analis bisnis untuk mengakses data dengan pilihan alat dan kerangka kerja analitik mereka. Ini termasuk kerangka kerja sumber terbuka seperti Apache Hadoop, Presto, dan Apache Spark, dan penawaran komersial dari vendor gudang

data dan intelijen bisnis. Data Lakes memungkinkan Anda menjalankan analitik tanpa perlu memindahkan data Anda ke sistem analitik terpisah.

Pembelajaran Mesin

Data Lakes akan memungkinkan organisasi menghasilkan berbagai jenis wawasan termasuk pelaporan data historis, dan melakukan pembelajaran mesin di mana model dibangun untuk memperkirakan kemungkinan hasil, dan menyarankan serangkaian tindakan yang ditentukan untuk mencapai hasil optimal.

3.13 Nilai dari Data Lake

Kemampuan untuk memanfaatkan lebih banyak data, dari lebih banyak sumber, dalam waktu yang lebih singkat, dan memberdayakan pengguna untuk berkolaborasi dan menganalisis data dengan berbagai cara menghasilkan pengambilan keputusan yang lebih baik dan lebih cepat. Contoh di mana Data Lakes memiliki nilai tambah meliputi seperti yang ditunjukkan pada Gambar 9:

Gambar 3.9 Nilai dari sebuah data lake

Peningkatan interaksi pelanggan

Data Lake dapat menggabungkan data pelanggan dari platform CRM dengan analisis media sosial, platform pemasaran yang mencakup riwayat pembelian, dan tiket insiden untuk memberdayakan bisnis dalam memahami kelompok pelanggan yang paling menguntungkan, penyebab churn pelanggan, serta promosi atau penghargaan yang akan meningkatkan loyalitas.

Meningkatkan pilihan inovasi R&D

Danau data dapat membantu tim R&D Anda menguji hipotesis, menyempurnakan asumsi, dan menilai hasil—seperti memilih bahan yang tepat dalam desain produk Anda yang menghasilkan kinerja lebih cepat, melakukan penelitian genomik yang menghasilkan pengobatan yang lebih efektif, atau memahami kemauan pelanggan untuk membayar atribut yang berbeda.

Meningkatkan efisiensi operasional

Internet of Things (IoT) memperkenalkan lebih banyak cara untuk mengumpulkan data pada proses seperti manufaktur, dengan data real-time yang berasal dari perangkat yang terhubung internet. Danau data memudahkan penyimpanan dan menjalankan analisis pada data IoT yang dihasilkan mesin untuk menemukan cara mengurangi biaya operasional dan meningkatkan kualitas.

3.14 Tantangan Data Lakes

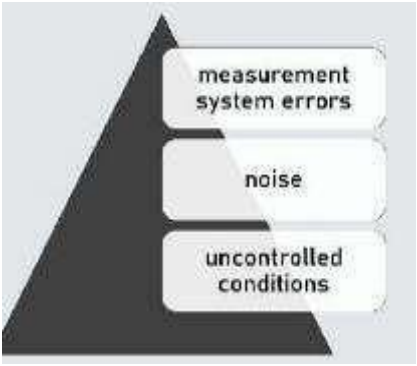
Tantangan utama dengan arsitektur data lake adalah bahwa data mentah disimpan tanpa pengawasan terhadap isinya. Agar data lake dapat digunakan, data lake perlu memiliki mekanisme yang ditetapkan untuk membuat katalog dan mengamankan data. Tanpa elemen-elemen ini, data tidak dapat ditemukan atau dipercaya sehingga menghasilkan "data swamp". Memenuhi kebutuhan audiens yang lebih luas mengharuskan data lake memiliki tata kelola, konsistensi semantik, dan kontrol akses.

Gambar3. 10 Tantangan danau data

3.15 Streaming data sensor

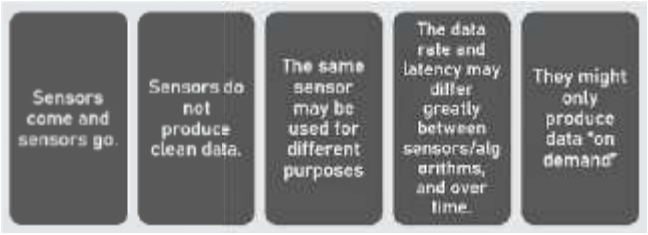
Meningkatnya ketersediaan perangkat keras sensor yang murah, kecil, dan berdaya rendah serta keberadaan jaringan kabel dan nirkabel di mana-mana telah menyebabkan prediksi bahwa 'lingkungan cerdas' akan muncul dalam waktu dekat. Seringkali sistem ini akan memantau kondisi di dunia nyata: cuaca, suhu, lalu lintas jalan, lokasi objek, harga di pasar saham. Dalam beberapa kasus, pembaruan rutin (di mana 'rutin' dapat berkisar dari milidetik hingga jam) tentang kondisi ini dihasilkan; dalam kasus lain, sistem memberikan pemberitahuan saat terjadi perubahan khusus. Sensor di lingkungan ini mengumpulkan informasi terperinci tentang situasi orang-orang, yang digunakan untuk meningkatkan aplikasi pemrosesan informasi yang ada di perangkat seluler dan 'lingkungan (ada atau ada di semua sisi)' mereka. Menjembatani kesenjangan antara data sensor dan informasi aplikasi menimbulkan persyaratan baru untuk manajemen data. Sisi pasokan: sensor

Sisi pasokan lingkungan cerdas terdiri dari banyak sensor yang menghasilkan data pada tingkat yang mungkin sangat tinggi. PocketLab One dirancang untuk menjadi solusi awal yang mendasar (dan akan sesuai dengan kebutuhan sebagian besar kelas sains). Ia dapat mengukur gerakan, percepatan, kecepatan sudut, medan magnet, tekanan, ketinggian, dan suhu. Penggunaan sensor untuk memasukkan data ke dalam sistem memiliki beberapa konsekuensi. Nilai-nilai tersebut tidak akan sama persis dengan dunia nyata karena kesalahan sistem pengukuran, kebisingan, dan kondisi yang tidak terkendali seperti yang ditunjukkan pada



Gambar 3.11 Sensor sisi suplai

Sifat jaringan sensor yang terdistribusi, nirkabel, dan bertenaga baterai akan memaksa manajemen data untuk memperhitungkan kegagalan sensor, latensi jaringan, dan kerugian. Di sisi lain, akan ada banyak data yang berlebihan (atau, dalam istilah statistik, sangat berkorelasi) untuk melawan fitur-fitur negatif ini. Beberapa catatan untuk menggambarkan situasi. Di sisi lain, akan ada banyak data yang berlebihan (atau, dalam istilah statistik, sangat berkorelasi) untuk melawan fitur-fitur negatif ini. Beberapa catatan untuk menggambarkan situasi.



***Gambar 3.12** Beberapa catatan untuk menggambarkan situasi*

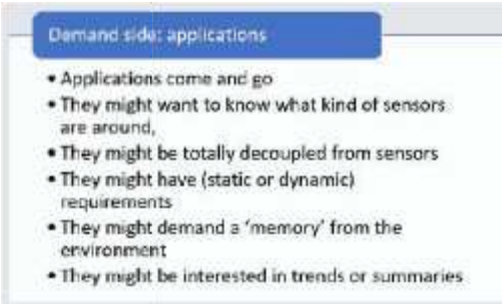
Sensor ada dan tidak. Sensor dapat rusak karena baterainya habis, dan dapat menyala lagi saat diganti. Sensor dapat dilepas, dipindahkan, dan disambungkan di tempat lain. Sensor dapat diganti seluruhnya

dengan model yang lebih baru. Sensor dapat memiliki koneksi nirkabel yang tidak berfungsi sepanjang waktu.

- Sensor tidak menghasilkan data yang bersih. Rata-rata harus diambil, filter noise harus diterapkan, pengaruh lingkungan (egechos) harus diperhitungkan.
- Sensor yang sama dapat digunakan untuk berbagai keperluan. Algoritme yang berbeda diterapkan pada data mentah, tergantung pada apa yang ingin Anda ketahui, misalnya menggunakan mikrofon untuk mengidentifikasi pembicara, memposisikan pembicara, atau memperkirakan tingkat kebisingan lingkungan.
- Laju data dan latensi dapat sangat berbeda antara sensor/algoritma, dan seiring waktu: Dalam beberapa kasus, hal itu dapat diparameterisasi (misalnya sensor atau algoritma dapat dikonfigurasi untuk menghasilkan output pada beberapa laju). Dalam beberapa kasus, istilah "laju data" bahkan mungkin tidak berlaku sama sekali (misalnya pembaca RFID yang menghasilkan pembacaan (atau serangkaian pembacaan) setiap kali tag terdeteksi).
- Mereka mungkin hanya menghasilkan data "sesuai permintaan" karena biaya yang terkait dengannya. Biaya ini bisa berupa daya, tetapi bisa juga berupa uang jika sensor tersebut milik pihak lain (pikirkan sensor cuaca atau lalu lintas).

Sisi permintaan: aplikasi

Aplikasi biasanya tidak tertarik pada detail tentang arsitektur penginderaan dan akan memerlukan 'model dunia' tingkat tinggi yang cukup sebagai dasar perilakunya. Saat aplikasi terhubung, aplikasi mungkin akan tertarik pada data langsung beresolusi tinggi; saat aplikasi terputus untuk sementara waktu, aplikasi mungkin lebih suka meminta ringkasan (misalnya properti seperti rata-rata, variasi, frekuensi). Beberapa karakteristik lainnya:



Gambar3. 13 Sisi permintaan: aplikasi

Aplikasi datang dan pergi. Aplikasi dapat dinyalakan dan dimatikan sesuai keinginan; aplikasi diduplikasi untuk setiap pengguna baru; aplikasi dimutakhirkan. Aplikasi terputus di satu tempat dan terhubung di tempat lain, dan mungkin tertarik dengan apa yang terjadi di saat yang bersamaan. Mereka mungkin ingin mengetahui jenis sensor apa yang ada di sekitar, dan menyesuaikan tuntutan informasi mereka dengan hal ini :

- Mereka mungkin benar-benar terpisah dari sensor, dan hanya ingin tahu, misalnya orang mana yang ada di meja tertentu.
- Mereka mungkin memiliki persyaratan (statis atau dinamis) tentang kecepatan pengiriman data kepada mereka. Kecepatan ini dapat sangat bervariasi dari satu aplikasi ke aplikasi lainnya.
- Mereka mungkin menuntut 'ingatan' dari lingkungan untuk menemukan rincian peristiwa tertentu di masa lalu.
- Mereka mungkin tertarik pada tren atau ringkasan, bukan pada hal spesifik.

3.16 Penggunaan data sensor

Data sensor ada di mana-mana dan dapat diterapkan pada berbagai sektor, termasuk perawatan kesehatan, prakiraan cuaca, analisis suara, dan streaming video. Mari kita lihat beberapa aplikasi data sensor.

Data sensor kesehatan:

Banyak fasilitas medis yang memantau tanda-tanda vital pasien secara real time. Denyut jantung, aktivitas elektrodermal, gelombang otak, suhu, dan indikator vital lainnya termasuk di antaranya. Jika dikumpulkan, data sensor real time sangat banyak dan dapat diklasifikasikan sebagai big data. Teknologi TIK dapat memanfaatkan informasi ini untuk mendiagnosis pasien.

Data cuaca

Banyak satelit yang menyediakan streaming data cuaca secara real-time untuk menangkap sinyal-sinyal penting terkait cuaca. Informasi ini digunakan untuk meramalkan cuaca.

Data sensor dari IOT

Perangkat internet khusus, seperti Raspberry Pi, Apple Watch, ponsel pintar, jam tangan kesehatan, dan sebagainya, dapat mengumpulkan banyak data sensor. Data ini dapat dikirimkan secara real time, sehingga server tertentu dapat mengaksesnya.

Ringkasan

- Data mart adalah struktur/pola akses yang digunakan untuk mendapatkan data yang berhadapan dengan klien dalam pengaturan gudang data. Data mart adalah bagian dari gudang data yang sering kali difokuskan pada satu lini bisnis atau tim.
- Data mart merupakan bagian dari gudang data yang difokuskan pada lini bisnis, departemen, atau area topik tertentu. Data mart menyediakan data khusus bagi sekelompok pengguna tertentu, yang memungkinkan mereka memperoleh wawasan penting dengan cepat tanpa harus memilah-milah seluruh gudang data.
- Data mart dependen memungkinkan data dari beberapa organisasi bersumber dari satu Gudang Data. Ini adalah contoh data mart yang memberikan manfaat sentralisasi. Anda harus menyiapkan satu atau

beberapa data mart fisik sebagai data mart dependen jika Anda perlu membuatnya.

- Tanpa menggunakan gudang data terpusat, sebuah pusat data independen terbentuk. Jenis Pusat Data ini paling cocok untuk kelompok yang lebih kecil di dalam sebuah perusahaan.
- Danau data adalah sistem atau repositori yang menyimpan data dalam bentuk asli/mentah, yang sering kali berupa gumpalan objek atau file.
- Data yang terus-menerus dibuat oleh beberapa sumber disebut sebagai data streaming. Tanpa memiliki akses ke semua data, data tersebut harus ditangani secara berurutan dengan memanfaatkan teknik pemrosesan aliran.

SOAL Latihan

Q1: Manakah dari berikut ini merupakan tujuan penambangan data?

A. Untuk menjelaskan suatu peristiwa atau kondisi yang diamati
B. Untuk menganalisis data untuk hubungan yang diharapkan
C. Untuk mengkonfirmasi bahwa data ada
D. Untuk membuat gudang data baru

Q2: Manakah dari berikut ini yang merupakan gudang data?

A. berfokus pada satu subjek
B. Disusun berdasarkan bidang subjek penting
C. Koleksi komputer
D. Tidak ada yang di atas

Q3: Pilih jenis data mart.

A. Datamart yang bergantung
B. Datamart independen

C. Datamart hibrida
D. Semua hal di atas

Q4: Pilih pernyataan yang tidak benar tentang gudang data

A. Gudang data bersifat fleksibel
B. Gudang data memiliki umur panjang
C. Gudang data adalah model top-down.
D. D. Data warehouse merupakan suatu sistem yang terdesentralisasiatau Data Besar

Q5: _____ dibangun dengan mengambil data dari gudang data pusat yang sudah ada.

A. Datamart yang bergantung
B. Datamart independen
C. Datamart hibrida
D. Semua hal di atas

Q6: _____ dibangun dengan mengambil data dari sumber operasional atau eksternal, atau keduanya.

A. Datamart yang bergantung
B. Datamart independen
C. Datamart hibrida
D. Semua hal di atas

Q7: Sebuah data mart _____ menggabungkan input dari sumber selain Data Warehouse

A. Datamart yang bergantung
B. Datamart independen
C. Datamart hibrida
D. Semua hal di atas

T8: Streaming data besar adalah suatu proses di mana data besar diproses dengan cepat untuk mengekstrak wawasan _____ darinya.

A. waktu nyata
B. data streaming
C. keduanya a dan b
D. Tidak ada yang di atas

Q9: Data dinamis yang dihasilkan secara terus-menerus dari berbagai sumber dianggap _____

A. waktu nyata
B. mengukus data
C. keduanya a dan b
D. Tidak ada yang di atas

Q10: _____ menggunakan data dan analitik untuk membantu organisasi keuangan mengidentifikasi pedagang yang melakukan penipuan dengan cepat dan mudah guna mengurangi risiko.

A. Kartu Debit
B. Kartu kredit
C. MasterCard
D. Tidak ada yang di atas

Q11: Manakah dari pernyataan streaming data berikut yang benar?

A. Data aliran pada hakikatnya tidak terstruktur.
B. Data aliran memiliki laju perubahan yang cepat.
C. Komponen aliran tidak dapat disimpan ke cakram.
D. Tidak ada yang di atas

Q12: Meningkatnya ketersediaan perangkat keras sensor yang murah, kecil, dan berdaya rendah telah menyebabkan prediksi bahwa _____ akan muncul dalam waktu dekat.

A. Lingkungan kecil
B. Sisi pasokan
C. keduanya a dan b
D. Tidak ada yang di atas

Q13: _____ dari lingkungan pintar terdiri dari banyak sensor yang menghasilkan data pada tingkat yang sangat tinggi secara real-time
A. data streaming
B. sisi pasokan
C. keduanya a dan b
D. Tidak ada yang di atas

Q14: _____ dirancang untuk menjadi solusi pemula yang mendasar
A. data streaming
B. sisi pasokan
C. Lab Saku
D. Tidak ada yang di atas

Q15: _____ dapat rusak karena baterainya habis, dan dapat menyala lagi ketika diganti
A. Sensor datang dan pergi
B. Sensor tidak menghasilkan data yang bersih
C. Keduanya
D. Tidak ada yang di atas

 **Daftar Pustaka**

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187.

https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

.

Tautan Web

1. Sumber daya Apache Hadoop: https://hadoop.apache.org/docs/r2.7.2/
2. Apache HDFS: https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html
3. Situs API Hadoop: http://hadoop.apache.org/docs/current/api/
4. Basis data NoSQL: http://nosql-database.org/
5. Apache Spark: http://spark.apache.org/docs/latest/
6. Tutorial tentang teknologi Big Data: https://www.tutorialspoint.com/

BAB 4: Manajemen NOSQL

Ahmad Soderi, S. Kom, MM.

Tujuan

- mengidentifikasi perbedaan utama antara NOSQL dan database relasional
- menghargai arsitektur dan jenis database NOSQL
- jelaskan jenis utama database NOSQL dan fitur-fiturnya
- belajar mendistribusikan model data
- pelajari partisi Hadoop

Perkenalan

Basis data NOSQL adalah cara cerdas untuk mengatur sejumlah besar data heterogen secara hemat biaya untuk akses dan pembaruan yang efisien. Basis data NOSQL yang ideal sepenuhnya selaras dengan sifat masalah yang sedang dipecahkan, dan sangat cepat dalam menyelesaikan tugas tersebut. Hal ini dicapai dengan melonggarkan banyak kendala di jantung komputer kosmik.

4.1 Apa itu Basis Data Relasional

Sistem manajemen data relasional (RDBM) merupakan teknologi basis data yang integritas dan redundansi dalam menyimpan data dalam basis data relasional. Dengan demikian, data disimpan dalam banyak format inovatif yang sangat sesuai dengan kebutuhan bisnis. Basis data NOSQL yang beragam pada akhirnya akan secara kolektif berkembang menjadi serangkaian pengetahuan holistik yang efisien dan elegan yang disimpan kuat dan digunakan secara universal oleh hampir semua perusahaan. Basis data relasional terstruktur dan dioptimalkan untuk memastikan keakuratan dan konsistensi data, sekaligus menghilangkan redundansi data. Basis data ini disimpan di komputer terbesar dan paling

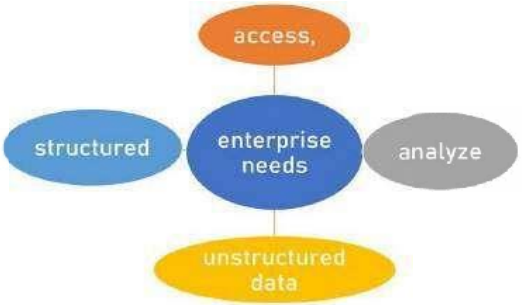
andal untuk memastikan bahwa data selalu tersedia pada tingkat yang terperinci dan dengan kecepatan tinggi.

Mengapa Basis Data NOSQL Muncul?

Namun, big data merupakan aliran data yang jauh lebih besar dan tidak dapat diprediksi. Basis data relasional tidak memadai untuk tugas ini, dan juga akan sangat mahal untuk volume data yang begitu besar. Mengelola biaya dan kecepatan pengelolaan aliran data yang begitu besar dan heterogen memerlukan pelonggaran terhadap banyak aturan dan persyaratan ketat dari basis data relasional. Bergantung pada batasan mana yang dilonggarkan, jenis struktur basis data yang berbeda akan muncul. Ini disebut basis data NOSQL, untuk membedakannya dari basis data relasional yang menggunakan Bahasa Kueri Terstruktur (SQL) sebagai sarana utama untuk memanipulasi data.

Basis data NOSQL adalah basis data generasi berikutnya yang tidak bersifat relasional dalam desainnya. Nama NOSQL dimaksudkan untuk membedakannya dari basis data 'pra-relasional' yang kuno. Saat ini, hampir setiap organisasi yang harus mengumpulkan umpan balik dan sentimen pelanggan untuk meningkatkan bisnis mereka, menggunakan basis data NOSQL. NOSQL berguna ketika suatu perusahaan perlu mengakses, menganalisis, dan memanfaatkan sejumlah besar data terstruktur atau tidak terstruktur yang disimpan dari jarak jauh di server virtual di seluruh dunia.

Database NOSQL berguna ketika



***Gambar 4.1**: Basis data NOSQL*

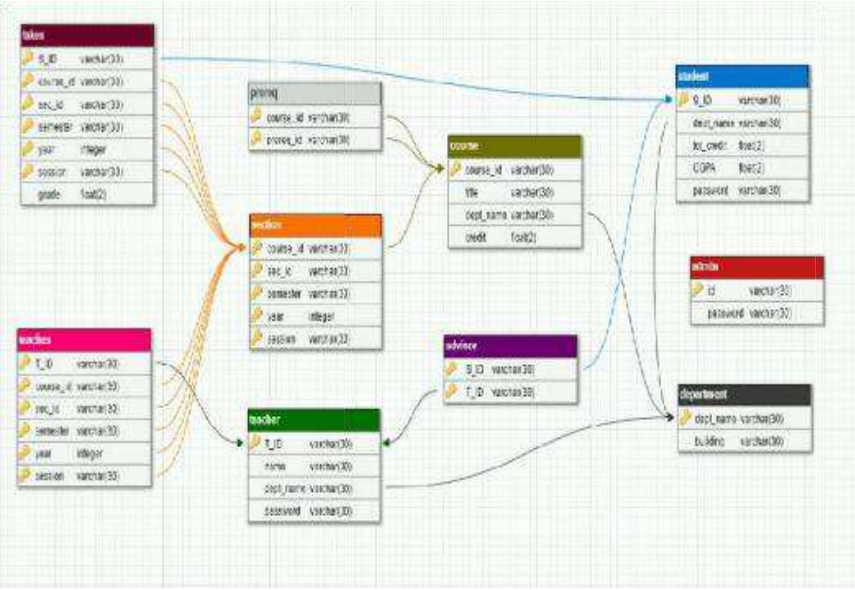
Kendala dari basis data relasional dilonggarkan dalam banyak hal. Misalnya, basis data relasional mengharuskan setiap elemen data dapat diakses secara acak dan nilainya dapat diperbarui di lokasi fisik yang sama. Namun, fisika penyimpanan yang sederhana mengatakan bahwa lebih mudah dan cepat untuk membaca atau menulis blok data berurutan pada disk. Oleh karena itu, file basis data NOSQL ditulis sekali dan hampir tidak pernah diperbarui di tempat. Jika versi baru dari suatu bagian data tersedia, itu akan ditambahkan ke file masing-masing. Sistem akan memiliki kecerdasan untuk menghubungkan data yang ditambahkan ke yang asli file. Keduanya berbeda satu sama lain dalam banyak hal. Pertama, basis data NOSQL tidak mendukung skema relasional atau bahasa SQL. Istilah NOSQL sebagian besar merupakan singkatan dari "Tidak hanya SQL".

Kedua, kemampuan pemrosesan transaksi mereka cepat tetapi lemah, dan mereka tidak mendukung properti ACID (Atomicity, Consistency, Isolation, Durability) yang terkait dengan pemrosesan transaksi menggunakan basis data relasional. Sebaliknya, mereka mendukung properti BASE (Basically Available, Soft State, dan Finally Consistent). Dengan demikian, basis data NOSQL kira-kira akurat pada

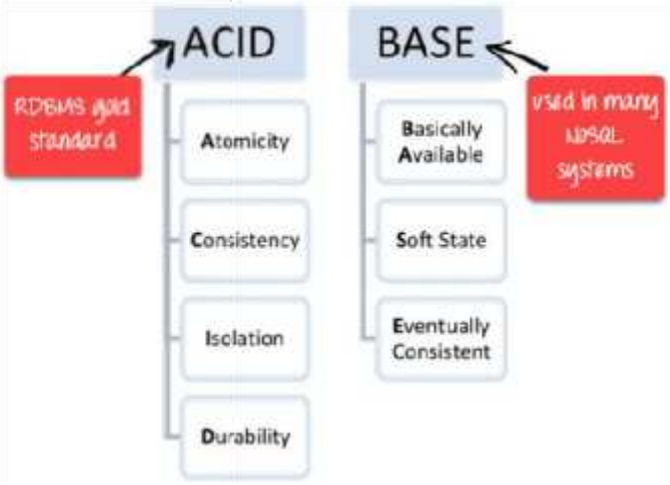
setiap titik waktu, dan pada akhirnya akan konsisten. Ketiga, basis data ini juga terdistribusi dan dapat diskalakan secara horizontal untuk mengelola basis data skala web menggunakan kluster penyimpanan Hadoop. Dengan demikian, mereka bekerja dengan baik dengan mekanisme penyimpanan write-once dan readmany dari kluster Hadoop. Tabel 6.1 mencantumkan fitur komparatif RDBMS dan NOSQL.

4.2Perbedaan antara RDBMS dan NOSQL

Ketiga, basis data ini juga terdistribusi dan dapat diskalakan secara horizontal untuk mengelola basis data skala web menggunakan kluster penyimpanan Hadoop.



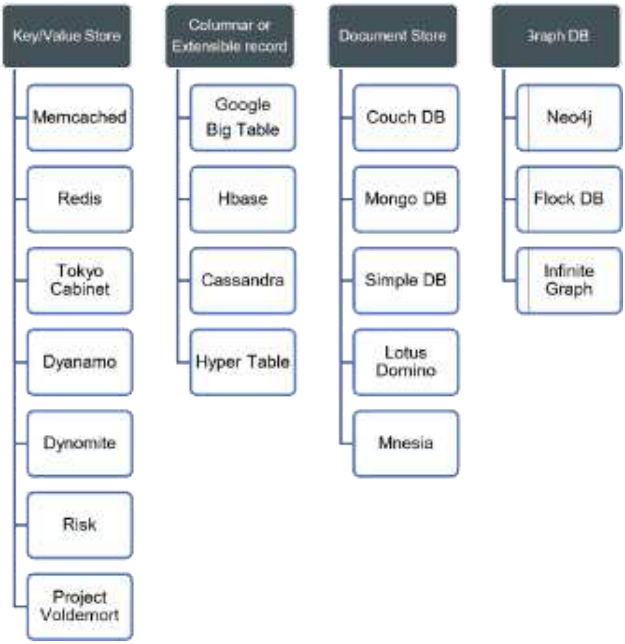
***Gambar4. 2**: Tidak mendukung skema relasional*



***Gambar4. 3**:RDBMS vs NOSQL*

4.3 Jenis-jenis Database NOSQL

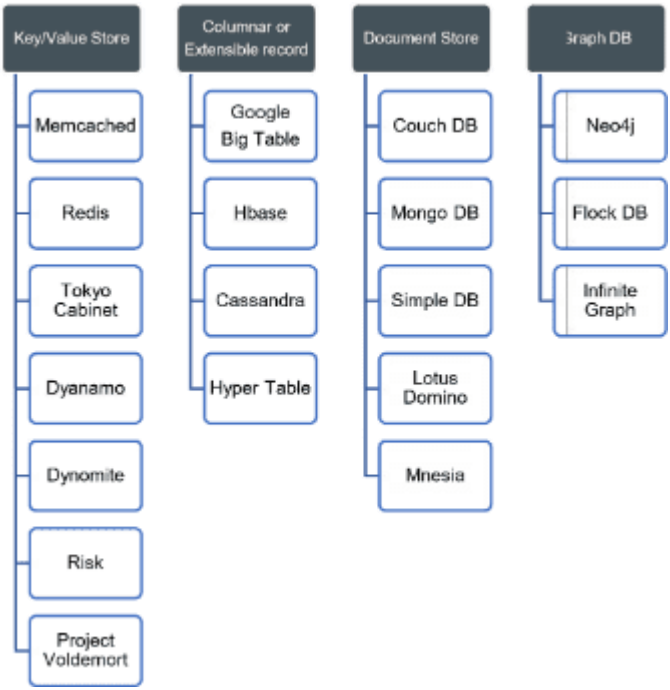
Keragaman big data berarti bahwa ukuran dan jenis file akan sangat bervariasi. Meskipun namanya, basis data NOSQL tidak serta merta melarang bahasa kueri terstruktur (seperti SQL). Sementara beberapa sistem NOSQL sepenuhnya non-relasional, yang lain hanya menghindari beberapa fungsi terpilih dari RDMS seperti skema tabel tetap dan operasi penggabungan. Untuk sistem NOSQL, alih-alih menggunakan tabel, data dapat diatur dalam format pasangan kunci/nilai, dan kemudian SQL dapat digunakan. Ada basis data NOSQL khusus yang sesuai dengan berbagai tujuan. Pilihan basis data NOSQL bergantung pada persyaratan sistem. Setidaknya ada 200 implementasi basis data NOSQL dari keempat jenis ini. Kunjungi NOSQL-database.org untuk informasi lebih lanjut. Seperti yang dapat Anda lihat di



Gambar4. 4. Berikut beberapa penawaran terkini dalam kategori ini.

Tabel 1 Fitur komparatif RDBMS dan NOSQL.

Feature	RDBMS	NOSQL
Applications	Mostly centralized Applications (e.g. ERP)	Mostly designed for the decentralized applications (e.g. Web, mobile, sensors)
Rigor	Support ACID properties for Transaction Processing	Support BASE properties for approximate Reporting
Applications	Mostly centralized Applications (e.g. ERP)	Mostly designed for the decentralized applications (e.g. Web, mobile, sensors)
Rigor	Support ACID properties for Transaction Processing	Support BASE properties for approximate Reporting
Availability	Moderate to high	Continuous availability to receive and serve data
Velocity	Moderate velocity of data	High velocity of data (devices, sensors, social media, etc.). Low latency of access
Data Volume	Moderate size; archived after for a certain period	Huge volume of data, stored mostly for a long time or forever; Linearly scalable DB.
Data Sources	Data arrives from one or few, mostly predictable sources	Data arrives from multiple locations and are of unpredictable nature
Data type	Data are mostly structured	Structured or unstructured data
Data Access	Primary concern is reading the data	Concern is both read and write
Technology	Standardized relational schemas; SQL language	Many designs with many implementations of data structures and access languages
Cost	Expensive; commercial	Low; open-source software

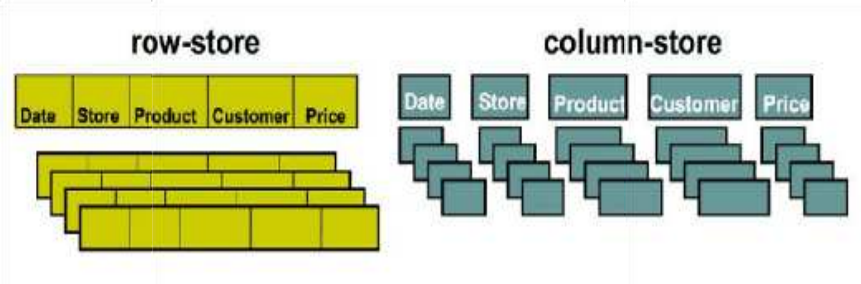


Gambar 4.5: *Penawaran dalam database NOSQL*

Basis Data Kolom:

Ini adalah struktur basis data yang hanya menyertakan kolom-kolom yang relevan dari kumpulan data, beserta informasi pengenal kunci. Ini berguna dalam mempercepat beberapa kueri yang sering dicari dari kumpulan data yang sangat besar. Misalkan ada gudang data yang sangat besar dari data akses log web, yang digulung berdasarkan jumlah akses web per jam. Ini perlu dikueri, atau diringkas secara berkala, yang hanya melibatkan beberapa bidang data dari basis data. Dengan demikian, kueri dapat dipercepat dengan mengatur basis data dalam format kolom. Ini berguna untuk sistem manajemen konten, platform blog, pemeliharaan penghitung, penggunaan yang kedaluwarsa, volume penulisan yang besar

seperti agregasi log. Basis data keluarga kolom untuk sistem dengan baik ketika pola kueri telah stabil.



Gambar 4.6: *Database kolom*

HBase dan Cassandra adalah dua dari penawaran basis data Columnar yang paling populer. HBase dikembangkan di Yahoo, dan hadir sebagai bagian dari ekosistem Hadoop. Cassandra awalnya dikembangkan di Facebook untuk melayani basis penggunanya yang tumbuh secara eksponensial, yang sekarang mendekati 2 miliar orang. Cassandra menjadi open source pada tahun 2008.

Basis Data Pasangan Kunci/Nilai

Mungkin ada kumpulan banyak elemen data seperti kumpulan pesan teks, yang juga dapat dimasukkan ke dalam satu blok penyimpanan fisik. Setiap pesan teks adalah objek unik. Data ini perlu sering ditanyakan. Kumpulan pesan tersebut juga dapat disimpan dalam format pasangan kunci-nilai, dengan menggabungkan pengenal pesan dan konten pesan. Basis data kunci-nilai berguna untuk menyimpan informasi sesi, profil pengguna, preferensi, dan data keranjang belanja. Basis data kunci-nilai tidak berfungsi dengan baik saat kita perlu menanyakan berdasarkan bidang non-kunci atau pada beberapa bidang kunci pada saat yang sama. Dynamo adalah NOSQL yang terstruktur dengan nilai kunci yang sangat tersedia sistem penyimpanan yang memiliki properti dari database dan

tabel hash terdistribusi. Amazon DynamoDB adalah layanan database NOSQL yang dikelola sepenuhnya yang memberikan kinerja cepat dan dapat diprediksi dengan skalabilitas yang mulus.

DynamoDB secara otomatis menyebarkan data dan lalu lintas untuk tabel Anda ke server yang cukup untuk menangani persyaratan throughput dan penyimpanan Anda, sambil mempertahankan kinerja yang konsisten dan cepat.

Gambar 4.7: Database Pasangan Kunci-Nilai

Basis Data Dokumen

Basis data ini menyimpan seluruh dokumen dengan ukuran apa pun, sebagai nilai tunggal untuk elemen kunci. Misalkan seseorang menyimpan file film video berukuran 10 GB sebagai objek tunggal. Indeks dapat menyimpan informasi pengenal tentang film, dan alamat blok awal. Sistem dapat menangani detail penyimpanan lainnya. Format penyimpanan ini disebut format penyimpanan dokumen. Basis data dokumen umumnya berguna untuk sistem manajemen konten, platform blog, analitik web, analitik waktu nyata, aplikasi e-niaga. Basis data dokumen tidak akan berguna untuk sistem yang memerlukan transaksi

kompleks yang mencakup beberapa operasi atau kueri terhadap berbagai struktur agregat.

***Gambar4.8**: Basis Data Dokumen*

MongoDB adalah basis data dokumen sumber terbuka yang menyediakan kinerja tinggi, ketersediaan tinggi, dan penskalaan otomatis. Catatan dalam MongoDB adalah dokumen, yang merupakan struktur data yang terdiri dari pasangan bidang dan nilai. Nilai bidang dapat mencakup dokumen lain, larik, dan larik dokumen.

Basis Data Grafik

Basis data grafik sangat cocok untuk ruang masalah tempat kita memiliki data yang terhubung, seperti jaringan sosial, data spasial, informasi perutean, dan mesin rekomendasi. Grafik berikut menunjukkan contoh grafik jaringan sosial. Dengan mempertimbangkan orang-orang (simpul) dan hubungan mereka (tepi), Anda dapat mengetahui siapa

"teman dari teman" orang tertentu—misalnya, teman dari teman Howard. Misalnya, data peta geografis yang digunakan di Google Maps disimpan dalam serangkaian hubungan atau tautan antar titik. Untuk penanganan hubungan data yang intensif, basis data grafik meningkatkan kinerja hingga beberapa kali lipat. Raksasa teknologi seperti Google, Facebook, dan LinkedIn menggunakan basis data grafik untuk memberikan layanan yang terukur, berwawasan, dan cepat.

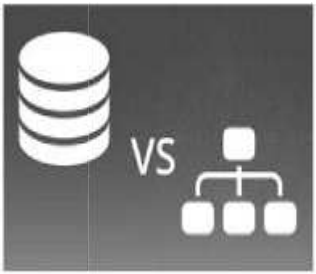
Neo4j adalah basis data transaksional yang sangat scalable dan paling populer yang sesuai dengan ACID dengan penyimpanan dan pemrosesan grafik asli. Ini adalah basis data grafik sumber terbuka, diimplementasikan dalam Java, dan dapat diakses dari perangkat lunak yang ditulis dalam bahasa lain. Basis data NOSQL pertama yang populer adalah HBase, yang merupakan bagian dari keluarga Hadoop. Basis data NOSQL paling populer yang digunakan saat ini adalah Apache Cassandra, yang dikembangkan dan dimiliki oleh Facebook hingga dirilis sebagai sumber terbuka pada tahun 2008. Sistem basis data NOSQL lainnya adalah SimpleDB, BigTable milik Google, MemcacheDB, Oracle NOSQL, Voldemort, dll.

***Gambar4.9**: Basis Data Grafik*

4.4 Model Data

Model data adalah model yang kita gunakan untuk memahami dan memanipulasi data kita. Bagi orang yang menggunakan basis data, model data menggambarkan cara kita berinteraksi dengan data dalam basis data. Hal ini berbeda dengan model penyimpanan, yang menggambarkan cara basis data menyimpan dan memanipulasi data secara internal. Dalam dunia yang ideal, kita seharusnya tidak mengetahui model penyimpanan, tetapi dalam praktiknya kita memerlukan setidaknya beberapa keterkaitannya - terutama untuk mencapai kinerja yang baik.

Database	Storage Model
How we interact with the data in the database	How the database stores and manipulates the data internally



Gambar 4.10: *Basis data berbeda dari model penyimpanan*

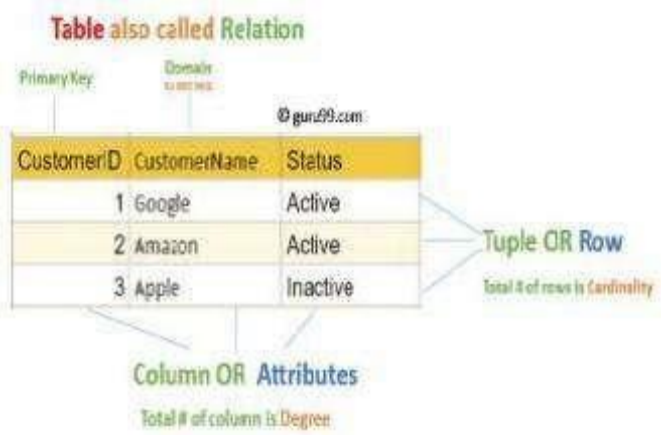
Model Data: Contoh

Dalam percakapan, istilah "model data" sering kali berarti model data tertentu dalam suatu aplikasi. Seorang pengembang mungkin menunjuk ke diagram hubungan entitas dari basis data mereka dan menyebutnya sebagai model data mereka yang berisi pelanggan, pesanan, produk, dan sejenisnya.

Apa Nama Model Data dalam Beberapa Dekade Terakhir?

Model data yang dominan dalam beberapa dekade terakhir adalah model data relasional, yang paling baik divisualisasikan sebagai

sekumpulan tabel, seperti halaman spreadsheet. Setiap tabel memiliki baris, dengan setiap baris mewakili beberapa entitas yang diminati. Kami menggambarkan entitas ini melalui kolom, yang masing-masing memiliki satu nilai. Kolom dapat merujuk ke baris lain dalam tabel yang sama atau berbeda, yang merupakan hubungan antara entitas tersebut.



***Gambar 4.11**: Model data relasional*

4.5 Pengantar NOSQL

Salah satu perubahan paling kentara pada NOSQL adalah menjauhnya dari model relasional. Setiap solusi NOSQL memiliki model berbeda yang digunakannya, yang kami kelompokkan ke dalam empat kategori yang banyak digunakan dalam ekosistem NOSQL:

- Kunci-nilai,
- Dokumen,
- Keluarga kolom
- Dan grafik

Tentu saja, tiga yang pertama memiliki karakteristik umum dari model datanya yang kita sebut orientasi agregat.

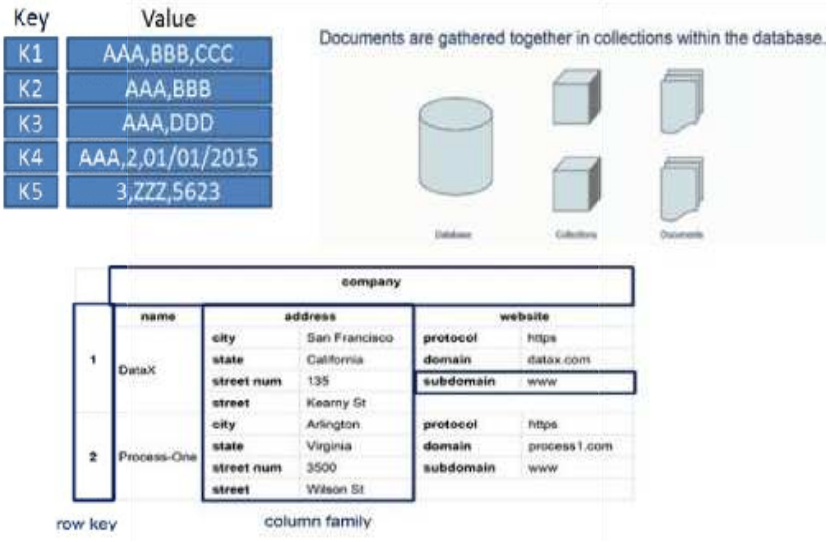
4.6 Model Relasional vs Model Data Agregat

Model Data Relasional

Model relasional mengambil informasi yang ingin kita simpan dan membaginya ke dalam tuple (baris). Tuple adalah struktur data terbatas. Ia menangkap sekumpulan nilai, jadi Anda tidak dapat menumpuk satu tuple di dalam tuple lain untuk mendapatkan rekaman bersarang, Anda juga tidak dapat meletakkan daftar nilai atau tuple di dalam tuple lain. Kesederhanaan ini mendukung model relasional-ia memungkinkan kita untuk menganggap semua operasi sebagai tuple yang beroperasi dan mengembalikan. Kesederhanaan ini mencirikan model relasional. Ia memungkinkan kita untuk menganggap manipulasi data sebagai operasi yang memiliki: – Sebagai tuple input, dan – Tuple pengembalian • Orientasi agregat mengambil pendekatan yang berbeda.

Model Data Agregat

Pemodelan basis data relasional sangat berbeda dengan jenis struktur data yang digunakan pengembang aplikasi. Agregat adalah kumpulan data yang berinteraksi dengan kita sebagai satu kesatuan. Satuan data atau agregat ini membentuk batasan untuk operasi ACID dengan basis data. Basis data Key-value, Document, dan Column-family semuanya dapat dilihat sebagai bentuk basis data berorientasi agregat.



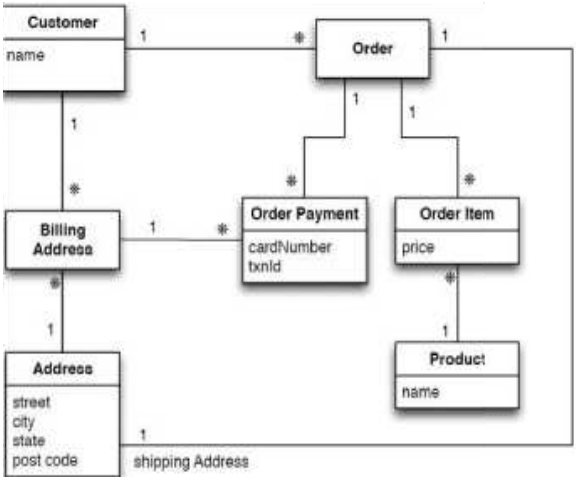
***Gambar4. 12**: Model data agregat*

Agregat memudahkan database untuk mengelola penyimpanan data melalui kluster, karena unit data sekarang dapat berada di mesin mana pun dan saat diambil dari database, semua data terkait akan ikut bersamanya. Database berorientasi agregat bekerja paling baik saat sebagian besar interaksi data dilakukan dengan agregat yang sama, misalnya saat ada kebutuhan untuk mendapatkan pesanan dan semua detailnya, lebih baik menyimpan pesanan sebagai objek agregat, tetapi menangani agregat ini untuk mendapatkan detail item pada semua pesanan tidaklah elegan.

Contoh Relasi dan Agregat

Kita harus membangun situs web e-commerce; kita akan menjual barang secara langsung ke pelanggan melalui web, dan kita harus

menyimpan informasi tentang pengguna, katalog produk, pesanan, alamat pengiriman, alamat penagihan, dan data pembayaran.



***Gambar 4.13**: Model Data Berorientasi pada Basis Data Relasional*

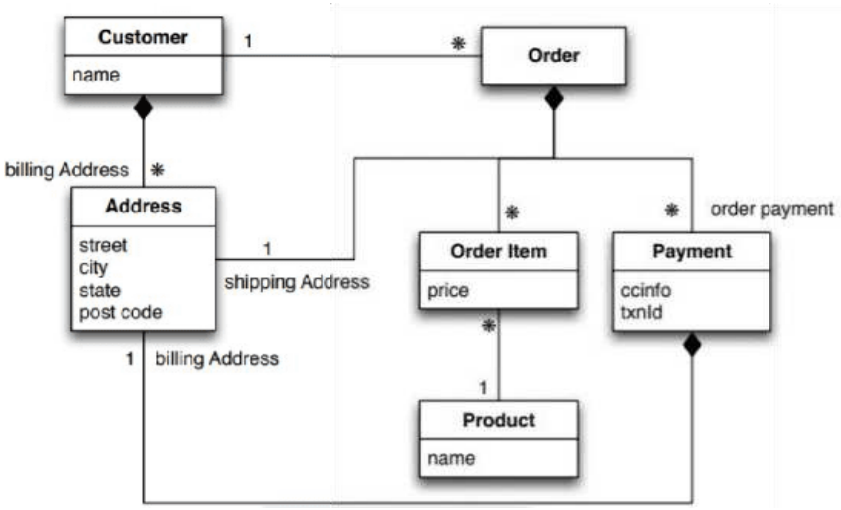
Kita dapat menggunakan skenario ini untuk memodelkan data menggunakan penyimpanan data relasi serta penyimpanan data NOSQL dan membicarakan kelebihan dan kekurangannya. Untuk model relasional, kita mulai dengan model data yang ditunjukkan pada gambar ini. Karena kita adalah prajurit relasional yang baik, semuanya dinormalisasi dengan benar, sehingga tidak ada data yang diulang dalam beberapa tabel. Kita juga memiliki integritas referensial. Sistem tatanan yang realistis tentu akan lebih rumit dari ini, tetapi inilah manfaat dari buku yang jernih. Mari kita lihat bagaimana model ini terlihat ketika kita berpikir dalam istilah yang lebih berorientasi agregat:

Dalam model ini, kami memiliki dua agregat utama: pelanggan dan pesanan. Kami telah menggunakan penanda komposisi berlian hitam di UML untuk menunjukkan bagaimana data sesuai dengan struktur agregasi. Pelanggan berisi daftar alamat penagihan; pesanan berisi daftar

item pesanan, alamat pengiriman, dan pembayaran. Pembayaran itu sendiri berisi alamat penagihan untuk pembayaran. Satu rekaman alamat logis muncul tiga kali dalam contoh data, tetapi alih-alih menggunakan ID, rekaman tersebut diperlakukan sebagai nilai dan disalin setiap kali. Ini sesuai dengan domain tempat kami tidak menginginkan alamat pengiriman, maupun alamat penagihan pembayaran, ia berubah. Dalam basis data relasional, kami akan memastikan bahwa baris alamat tidak diperbarui untuk kasus ini, dan membuat baris baru sebagai gantinya. Dengan agregat, kami dapat menyalin seluruh struktur alamat ke dalam agregat sesuai kebutuhan. Tautan antara pelanggan dan pesanan tidak berada dalam salah satu agregat—ini adalah hubungan antara agregat. Demikian pula, tautan dari item pesanan akan masuk ke dalam struktur agregat terpisah untuk produk, yang belum kami bahas. Kami telah menunjukkan nama produk sebagai bagian dari item pesanan di sini—jenis denormalisasi ini serupa dengan trade-off dengan basis data relasional, tetapi lebih umum dengan agregat karena kami ingin meminimalkan jumlah agregat yang kami akses selama interaksi data.

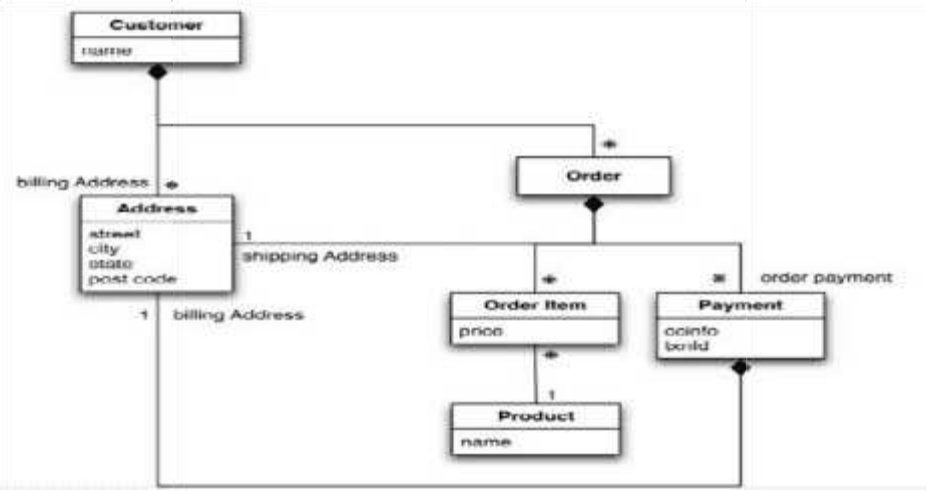
Contoh Model Agregat

Hubungan antara pelanggan dan pesanan merupakan hubungan antara agregat.



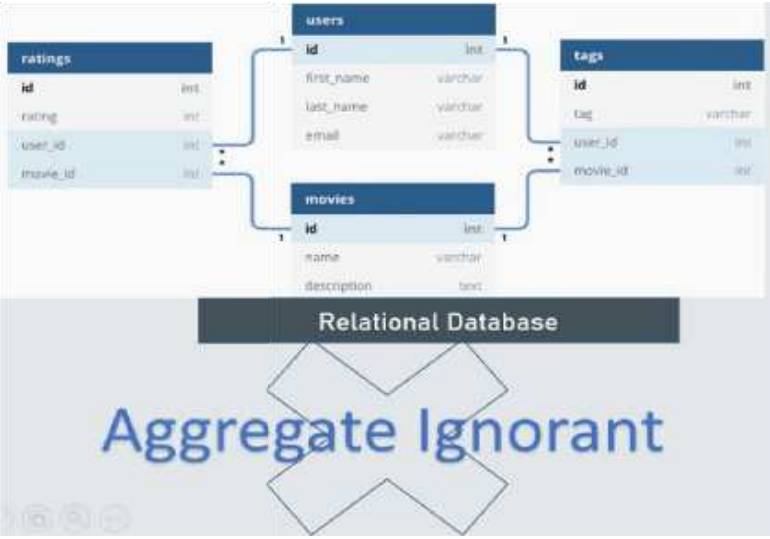
Gambar 4.14: *Contoh model data Agregat*

Hal penting yang perlu diperhatikan di sini bukanlah cara khusus kita menggambar batas agregat, melainkan fakta bahwa Anda harus berpikir tentang mengakses data tersebut - dan menjadikannya bagian dari pemikiran Anda saat mengembangkan model data aplikasi. Memang, kita dapat menggambar batas agregat secara berbeda, dengan memasukkan semua pesanan untuk pelanggan ke dalam agregat pelanggan.



Gambar 4.15: Sematkan semua objek untuk pelanggan dan pesanan pelanggan

Seperti kebanyakan hal dalam pemodelan, tidak ada jawaban universal untuk cara menggambar batas agregat Anda. Itu sepenuhnya tergantung pada bagaimana Anda cenderung memanipulasi data Anda. Jika Anda cenderung mengakses pelanggan bersama dengan semua pesanan pelanggan itu sekaligus, maka Anda akan lebih memilih agregat tunggal. Namun, jika Anda cenderung fokus pada mengakses satu pesanan pada satu waktu, maka Anda harus lebih memilih memiliki agregat terpisah untuk satu sama lain. Tentu saja, ini sangat spesifik konteks; beberapa aplikasi akan lebih memilih satu atau yang lain, bahkan dalam satu sistem, itulah sebabnya banyak orang lebih memilih ketidaktahuan agregat.



***Gambar 4.16**: Agregat Ignorant*

Konsekuensi Model Agregat

Pemetaan relasional menangkap berbagai elemen data dan hubungannya. Urutan terdiri dari Memesan barang, alamat pengiriman, dan metode pembayaran Semua dapat diekspresikan dalam model relasional dalam bentuk hubungan kunci asing. Sementara Pemetaan Relasi menangkap berbagai elemen data dan hubungannya dengan baik, ia melakukannya tanpa gagasan tentang entitas agregat. Dalam bahasa domain kita, kita dapat mengatakan bahwa pesanan terdiri dari item pesanan, alamat pengiriman, dan pembayaran. Ini dapat diekspresikan dalam model relasional dalam bentuk hubungan kunci asing-tetapi tidak ada yang membedakan hubungan yang mewakili agregasi dari yang tidak. Akibatnya, basis data tidak dapat menggunakan pengetahuan tentang struktur agregat untuk membantunya menyimpan dan mendistribusikan data.

Basis data tidak dapat menggunakan pengetahuan tentang struktur agregat untuk membantunya menyimpan dan mendistribusikan data. Berbagai teknik pemodelan data telah menyediakan cara untuk menandai struktur agregat atau komposit. Namun, masalahnya adalah bahwa pemodel jarang menyediakan semantik apa pun untuk apa yang membuat hubungan agregat berbeda dari yang lain; jika ada semantik, semantik itu bervariasi. Saat bekerja dengan basis data berorientasi agregat, kita memiliki semantik yang lebih jelas untuk dipertimbangkan dengan berfokus pada unit interaksi dengan penyimpanan data. Namun, ini bukan properti data logis: Ini semua tentang bagaimana data digunakan oleh aplikasi-perhatian yang sering kali berada di luar batas pemodelan data. Basis data relasional tidak memiliki konsep agregat dalam model datanya, jadi kami menyebutnya agregat-tidak tahu. Di dunia NOSQL, basis data grafik juga tidak tahu agregat. Di dunia NOSQL, basis data grafik juga tidak tahu agregat. Tidak tahu agregat bukanlah hal yang buruk. Sering kali sulit untuk menggambar batas agregat dengan baik, terutama jika data yang sama digunakan dalam banyak konteks yang berbeda. Pesanan merupakan agregat yang baik ketika pelanggan membuat dan meninjau pesanan, dan ketika pengecer memproses pesanan. Namun, jika pengecer ingin menganalisis penjualan produknya selama beberapa bulan terakhir, maka agregat pesanan menjadi masalah. Untuk mendapatkan riwayat penjualan produk, Anda harus menggali setiap agregat dalam basis data. Jadi, struktur agregat dapat membantu beberapa interaksi data tetapi menjadi kendala bagi yang lain. Model yang mengabaikan agregat memungkinkan Anda untuk dengan mudah melihat data dalam berbagai cara, jadi ini merupakan pilihan yang lebih baik ketika Anda tidak memiliki struktur utama untuk memanipulasi data Anda.

Agregat dan Operasi

Pesanan merupakan agregat yang baik ketika: Pelanggan membuat dan meninjau pesanan, dan – Ketika pengecer memproses pesanan Namun, ketika pengecer ingin menganalisis penjualan produknya selama beberapa bulan terakhir, maka agregat menjadi masalah. Kita perlu menganalisis setiap agregat untuk mengekstrak riwayat penjualan. Pesanan merupakan agregat yang baik ketika: Pelanggan membuat dan meninjau pesanan, dan – Ketika pengecer memproses pesanan Namun, ketika pengecer ingin menganalisis penjualan produknya selama beberapa bulan terakhir, maka agregat menjadi masalah. Kita perlu menganalisis setiap agregat untuk mengekstrak riwayat penjualan.Agregat dapat membantu dalam beberapa operasi dan tidak pada yang lain. Dalam kasus di mana tidak ada tampilan yang jelas, agregat, basis data yang tidak jelas adalah pilihan terbaik. Namun, ingat poin yang mendorong kita untuk mengagregat model (distribusi kluster). Menjalankan basis data pada kluster diperlukan saat menangani data dalam jumlah besar.

Apa Alasan yang Menentukan untuk Orientasi Agregat?

Alasan utama untuk orientasi agregat adalah bahwa hal itu sangat membantu dalam menjalankan pada kluster, yang seperti yang Anda ingat adalah argumen utama untuk munculnya NOSQL. Jika kita menjalankan pada kluster, kita perlu meminimalkan jumlah node yang perlu kita kueri saat kita mengumpulkan data. Dengan menyertakan agregat secara eksplisit, kita memberikan informasi penting kepada basis data tentang bit data mana yang akan dimanipulasi bersama, dan dengan demikian harus berada pada node yang sama.

Berjalan pada Cluster

Ini memberikan beberapa keuntungan pada daya komputasi dan distribusi data. Namun, ini memerlukan minimalisasi jumlah node yang akan di-query saat mengumpulkan data. Dengan menyertakan agregat

secara eksplisit, kami memberikan database informasi penting tentang informasi mana yang harus disimpan bersama.

Basis data NOSQL mampu menyimpan dan memproses data besar yang dicirikan oleh berbagai properti seperti volume, variasi, dan kecepatan. Basis data semacam itu digunakan dalam berbagai aplikasi pengguna yang membutuhkan data dalam jumlah besar yang sangat tersedia dan dapat diakses secara efisien. Namun, basis data tersebut tidak memaksakan atau memerlukan konsistensi data yang kuat, juga tidak mendukung transaksi. Misalnya, media sosial seperti Twitter dan Facebook menghasilkan terabyte data harian yang berada di luar kemampuan pemrosesan basis data relasional. Aplikasi semacam itu memerlukan kinerja tinggi tetapi mungkin tidak memerlukan konsistensi yang kuat. Vendor yang berbeda merancang dan mengimplementasikan basis data NOSQL secara berbeda. Memang, ada berbagai jenis basis data NOSQL seperti basis data dokumen, basis data nilai-kunci, penyimpanan kolom, dan basis data grafik. Namun tujuan umum mereka adalah menggunakan replikasi data untuk memastikan efisiensi, ketersediaan, dan skalabilitas data yang tinggi.

Jenis-jenis Database NOSQL

Mayoritas basis data NOSQL mendukung eventual continuity alih-alih strong continuity. Mereka tidak mendukung transaksi basis data yang memastikan konsistensi data yang kuat. Eventual continuity menjamin bahwa semua pembaruan akan mencapai semua replika setelah penundaan tertentu. Ini berfungsi untuk aplikasi tertentu seperti media sosial, catatan iklan, dll. Namun, beberapa aplikasi pengguna memerlukan konsistensi data yang kuat. Artinya, data rekening bank harus konsisten setiap kali ada pembaruan yang dilakukan pada data. Dalam aplikasi lain, seperti aplikasi permainan multipemain daring biasanya menyimpan data profil sejumlah besar pengguna yang memerlukan konsistensi yang kuat.

Oleh karena itu, basis data NOSQL akan berguna untuk mengelola data dalam aplikasi tersebut. Data rekening bank harus konsisten setiap kali ada pembaruan yang dilakukan pada data. Aplikasi permainan multipemain daring biasanya menyimpan data profil sejumlah besar pengguna yang memerlukan konsistensi yang kuat. Namun, kurangnya dukungan untuk transaksi, penggabungan tabel, dan integritas referensial dalam basis data NOSQL, berarti bahwa mereka tidak cocok untuk aplikasi seperti perbankan, permainan daring, dll. Aplikasi tersebut memerlukan semua replika data harus dibuat konsisten secara instan dan aplikasi harus memiliki versi data terbaru jika ada pembaruan.

Ringkasan

- Pengguna dapat mengelola hubungan data yang telah ditetapkan di berbagai basis data menggunakan basis data relasional standar. Microsoft SQL Server, Oracle Database, MySQL, dan IBM DB2 adalah contoh-contoh basis data relasional yang umum.
- Basis data non-tabular (kadang-kadang dikenal sebagai "bukan sekadar SQL") menyimpan data secara berbeda dari tabel relasional. Basis data NOSQL diklasifikasikan menurut model datanya. Dokumen, nilai-kunci, kolom-lebar, dan grafik adalah jenis yang paling umum. Basis data ini memiliki skema yang dapat disesuaikan dan dapat menangani volume data yang besar serta beban pengguna yang berat dengan mudah.
- Perangkat lunak yang memungkinkan pengguna memperbarui, meminta informasi, dan mengelola basis data relasional dikenal sebagai RDBMS, atau sistem manajemen basis data relasional. Bahasa pemrograman utama untuk mengakses basis data adalah Bahasa Kueri Terstruktur (SQL).

- KEASAMAN (Atomicity, Consistency, Isolation, Durability) Saat menganalisis basis data dan arsitektur aplikasi, Profesional Basis Data memeriksa ACID (akronim untuk Atomicity, Consistency, Isolation, dan Durability).

Soal Latihan.

1. Basis data NOSQL didefinisikan sebagai yang mana dari berikut ini?

A. Bahasa Indonesia: SQL Server
B. Bahasa pemrograman MongoDB
C. Kasandra
D. Tidak ada yang disebutkan

2. Basis data NOSQL digunakan terutama untuk menangani data _____ dalam jumlah besar.

A. Tidak terstruktur
B. Tersusun
C. Semi terstruktur
D. Semua yang disebutkan

3. NOSQL berguna ketika suatu perusahaan perlu mengakses, menganalisis, dan memanfaatkan sejumlah besar data terstruktur atau tidak terstruktur

A. Mengakses
B. Menganalisa
C. Memanfaatkan
D. Semua hal di atas

4.Apa nama model data dalam beberapa dekade terakhir?

A. Model data relasional
B. Pasar data
C. Model data pendukung
D. Tidak ada yang di atas

5. Meja juga disebut _____

A. Tupel
B. Kolom
C. Hubungan
D. Tidak ada yang di atas

6.Apa alasan yang mendasari orientasi agregat?

A. Sangat bagus jika berjalan di cluster
B. Sangat membantu dalam menjalankan regresi
C. Sangat membantu dalam menjalankan klasifikasi
D. Tidak ada yang di atas

7.NOSQL digunakan untuk

A. Data besar
B. Aplikasi web waktu nyata.
C. Keduanya
D. Tidak ada yang di atas

Pertanyaan Ulasan

1. Jelaskan jenis-jenis NOSQL.
2. Tuliskan fitur NOSQL.
3. Tuliskan tentang model data.
4. Perbedaan antara RDBMS vs NOSQL.
5. Apa tujuan utama penggunaan basis data NOSQL.
6. Apa kelebihan dan kekurangan NOSQL.

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data*

untuk prediksi permintaan produk dalam E-commerce. *1*.

BAB 05: Pengantar Hadoop

Ardian Fachreza, S.T, M. Kom

Tujuan

- Pelajari pengantar tentang Hadoop.
- Pelajari manfaat Hadoop untuk bigdata
- Pelajari Perangkat Lunak Open-Source Terkait Hadoop
- Pelajari apa itu big data
- Pelajari mengapa big data di cloud sangat masuk akal
- Pelajari Peluang besar, tantangan besar

5.1 Manfaat Hadoop untuk Big Data

- **Ketangguhan**— Data yang disimpan di node mana pun juga direplikasi di node lain dalam klaster. Hal ini memastikan toleransi kesalahan. Jika satu node mati, selalu ada cadangan data yang tersedia di klaster.
- **Skalabilitas**— Tidak seperti sistem tradisional yang memiliki keterbatasan dalam penyimpanan data, Hadoop dapat diskalakan karena beroperasi dalam lingkungan terdistribusi. Saat dibutuhkan, pengaturan dapat dengan mudah diperluas untuk menyertakan lebih banyak server yang dapat menyimpan hingga beberapa petabyte data.
- **Biaya rendah**— Karena Hadoop merupakan kerangka kerja sumber terbuka, tanpa lisensi yang harus diperoleh, biayanya jauh lebih rendah dibandingkan dengan sistem basis data relasional. Penggunaan perangkat keras komoditas yang murah juga menguntungkannya untuk menjaga solusinya tetap ekonomis.

- **Kecepatan**—Sistem berkas terdistribusi Hadoop, pemrosesan serentak, dan model MapReduce memungkinkan menjalankan kueri kompleks dalam hitungan detik.
- **Keanekaragaman data**— HDFS memiliki kemampuan untuk menyimpan berbagai format data seperti tidak terstruktur (misalnya video), semi terstruktur (misalnya file XML), dan terstruktur.

Saat menyimpan data, tidak diperlukan validasi terhadap skema yang telah ditetapkan sebelumnya. Sebaliknya, data dapat dibuang dalam format apa pun. Nantinya, saat diambil, data diurai dan disesuaikan ke dalam skema apa pun sesuai kebutuhan. Hal ini memberikan fleksibilitas untuk memperoleh wawasan berbeda menggunakan data yang sama.

5.2 Komponen Tambahan Ekosistem Hadoop

Ekosistem Hadoop adalah platform atau rangkaian yang menyediakan berbagai layanan untuk memecahkan masalah big data. Ekosistem ini mencakup proyek Apache dan berbagai alat serta solusi komersial.

Ada empat elemen utama Hadoop yaitu;

- HDFS,
- MapReduce,
- YARN, dan
- Hadoop Common.

Sebagian besar alat atau solusi digunakan untuk melengkapi atau mendukung elemen-elemen utama ini. Semua alat ini bekerja secara kolektif untuk menyediakan layanan seperti penyerapan, analisis, penyimpanan, dan pemeliharaan data, dsb. Berikut ini adalah beberapa komponen pelengkap yang banyak digunakan dalam ekosistem Hadoop.

- Sistem Berkas Terdistribusi Hadoop
- Negosiator Sumber Daya Lainnya
- Pemrograman Berbasis Pengolahan Data

- Pemrosesan data dalam memori
- Pemrosesan layanan data berbasis kueri
- Basis Data NOSQL
- Pembelajaran Mesin_perpustakaan algoritma
- Pencarian dan Pengindeksan
- Mengelola klaster
- Penjadwalan Pekerjaan

Apache

Ini adalah platform yang menangani semua tugas yang menghabiskan banyak waktu seperti pemrosesan batch, pemrosesan real-time interaktif atau iteratif, konversi grafik, dan visualisasi, dll. Platform ini menghabiskan sumber daya memori, sehingga lebih cepat dari sebelumnya dalam hal pengoptimalan. Spark paling cocok untuk data real-time sedangkan Hadoop paling cocok untuk data terstruktur atau pemrosesan batch, oleh karena itu keduanya digunakan di sebagian besar perusahaan secara bergantian. Spark paling cocok untuk data real-time sedangkan Hadoop paling cocok untuk data terstruktur atau pemrosesan batch, oleh karena itu keduanya digunakan di sebagian besar perusahaan secara bergantian.

Pig

Pig pada dasarnya dikembangkan oleh Yahoo yang bekerja pada bahasa Pig Latin, yaitu bahasa berbasis Query yang mirip dengan SQL. Pig adalah platform untuk menyusun aliran data, memproses, dan menganalisis kumpulan data yang besar. Pig melakukan pekerjaan mengeksekusi perintah dan di latar belakang, semua aktivitas MapReduce ditangani. Setelah pemrosesan, Pig menyimpan hasilnya dalam HDFS. Bahasa Pig Latin dirancang khusus untuk kerangka kerja ini yang berjalan pada Pig Runtime. Sama seperti Java yang berjalan pada JVM. Pig membantu mencapai kemudahan pemrograman dan pengoptimalan dan karenanya merupakan segmen utama dari Ekosistem Hadoop.

HIVE

Dengan bantuan metodologi dan antarmuka SQL, HIVE melakukan pembacaan dan penulisan set data yang besar. Akan tetapi, bahasa kuerinya disebut sebagai HQL (Hive Query Language). Bahasa ini sangat scalable karena memungkinkan pemrosesan real-time dan pemrosesan batch. Selain itu, semua tipe data SQL didukung oleh Hive sehingga memudahkan pemrosesan kueri. Mirip dengan kerangka kerja Pemrosesan Kueri, HIVE juga dilengkapi dengan dua komponen: Driver JDBC dan Baris Perintah HIVE. JDBC, bersama dengan driver ODBC bekerja untuk menetapkan izin penyimpanan data dan koneksi sedangkan Baris Perintah HIVE membantu dalam pemrosesan kueri. Basis H Ini adalah basis data NOSQL yang mendukung semua jenis data dan dengan demikian mampu menangani apa pun dari Basis Data Hadoop. Ia menyediakan kemampuan BigTable milik Google, sehingga mampu bekerja pada kumpulan Big Data secara efektif. Pada saat kita perlu mencari atau mengambil kejadian sesuatu yang kecil dalam basis data yang besar, permintaan tersebut harus diproses dalam rentang waktu yang singkat. Pada saat seperti itu, HBase sangat berguna karena memberi kita cara yang toleran untuk menyimpan data yang terbatas.Mahout**.** Mahout, memungkinkan Machine Learnability pada suatu sistem atau aplikasi. Machine Learning, seperti namanya, membantu sistem untuk mengembangkan dirinya sendiri berdasarkan beberapa pola, interaksi pengguna/lingkungan, atau berdasarkan algoritma. Ia menyediakan berbagai pustaka atau fungsi seperti penyaringan kolaboratif, pengelompokan, dan klasifikasi yang tidak lain adalah konsep Machine Learning. Ia memungkinkan pemanggilan algoritma sesuai kebutuhan kita dengan bantuan pustakanya sendiri.

5.3 Komponen Lainnya

Selain semua ini, ada beberapa komponen lain yang juga menjalankan tugas besar agar Hadoop mampu memproses kumpulan data besar. Komponen-komponen tersebut adalah sebagai berikut:

- Solr, Lucene-Ini adalah dua layanan yang menjalankan tugas pencarian dan pengindeksan dengan bantuan beberapa pustaka Java, khususnya Lucene yang berbasis Java yang juga memungkinkan mekanisme pemeriksaan ejaan. Namun, Lucene digerakkan oleh Solr.
- Zookeeper-Terdapat masalah besar dalam pengelolaan koordinasi dan sinkronisasi di antara sumber daya atau komponen Hadoop yang sering kali mengakibatkan ketidakkonsistenan. Zookeeper mengatasi semua masalah tersebut dengan melakukan sinkronisasi, komunikasi berbasis antar-komponen, pengelompokan, dan pemeliharaan.
- Oozie hanya menjalankan tugas penjadwal, dengan demikian menjadwalkan pekerjaan dan mengikatnya bersama-sama sebagai satu unit tunggal. Ada dua jenis pekerjaan, yaitu, alur kerja Oozie dan pekerjaan koordinator Oozie. Alur kerja Oozie adalah pekerjaan yang perlu dieksekusi secara berurutan sedangkan pekerjaan Koordinator Oozie adalah pekerjaan yang dipicu ketika beberapa data atau stimulus eksternal diberikan kepadanya.

5.4 Perangkat lunak sumber terbuka yang terkait dengan Hadoop

Proyek Open-Source Terkait Hadoop-adalah layanan berjasa kami dengan tujuan menyediakan proyek Hadoop yang sangat maju dan berkembang bagi para mahasiswa dan peneliti di seluruh dunia. Saat ini kami memusatkan perhatian pada konsep penelitian big data yang sedang tren termasuk model hemat energi & ramah lingkungan, penjadwalan sumber daya, isu keberlanjutan, toleransi kesalahan & keandalan, teknik pembelajaran mesin, analisis grafik, sistem rekomendasi skala besar, struktur indeks untuk analisis big data, analisis eksploratori, manajemen big data, komputasi ilmiah, dll. Saat ini, Hadoop adalah Alat Open-Source yang tersedia untuk umum. Ini adalah kerangka kerja yang menyediakan terlalu banyak layanan seperti Pig, Impala, Hive, HBase, dll. Jumlah alat

open-source yang berkembang di ekosistem Hadoop dan alat-alat ini terus meningkat. Mari kita lihat alat open-source yang terkait dengan Hadoop,

Gambar 5.1 *Tren konsep penelitian big data*

Alat Open-Source Terkait Hadoop Teratas

1. Lucene [Mesin pencari teks berbasis Java]
2. Eclipse [IDE populer yang ditulis dalam Java]
3. HBase [Basis Data Hadoop NOSQL]
4. Hive [Mesin kueri data]
5. Jaql [Bahasa kueri untuk JavaScript]
6. Pig [Platform analisis kumpulan data besar]
7. Zookeeper [Layanan konfigurasi terpusat]
8. Avro [Sistem serialisasi data]
9. UIMA [kerangka kerja analitik tak terstruktur]
10. Presto [Solusi kueri SQL terdistribusi]

1. ***Lusen-*** adalah pustaka pencarian berbasis Java sumber terbuka. Pustaka ini sangat populer dan merupakan pustaka pencarian yang cepat. Pustaka ini digunakan dalam aplikasi berbasis Java untuk menambahkan kemampuan pencarian dokumen ke semua jenis aplikasi dengan cara yang sangat sederhana dan efisien. Lucene adalah pustaka Pencarian berbasis Java yang sederhana namun canggih. Pustaka ini dapat digunakan dalam aplikasi apa pun untuk menambahkan kemampuan pencarian ke dalamnya. Lucene adalah proyek sumber terbuka. Pustaka ini dapat diskalakan. Pustaka berkinerja tinggi ini digunakan untuk mengindeks dan mencari hampir semua jenis teks. Pustaka Lucene menyediakan operasi inti yang dibutuhkan oleh semua aplikasi pencarian. Pengindeksan dan pencarian.

Bagaimana cara kerja Aplikasi Pencarian?

Gambar 5.2*: Bagaimana cara kerja aplikasi pencarian?*

Dapatkan Konten Mentah:

Tahap awal dalam setiap aplikasi pencarian adalah mengumpulkan konten target yang akan dilakukan pencarian.

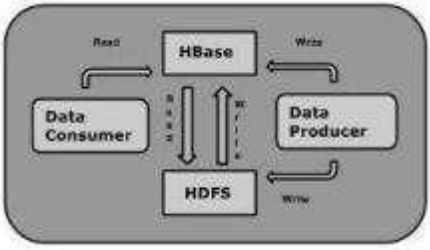
Bangun Dokumen:

Tahap selanjutnya adalah membuat dokumen dari awal, menggunakan materi yang dapat dipahami dan dianalisis dengan mudah oleh program pencarian. Analisis Dokumen:
Sebelum memulai proses pengindeksan, dokumen harus dievaluasi untuk menentukan apakah bagian-bagian teks tersebut layak untuk diindeks. Ini adalah tahap di mana dokumen diperiksa. Mengindeks Dokumen:
Setelah dokumen dibuat dan dianalisis, langkah selanjutnya adalah mengindeksnya sehingga dokumen ini dapat diambil berdasarkan kunci tertentu, bukan seluruh isi dokumen. Metode pengindeksan ini sebanding dengan indeks yang ditemukan di akhir buku, di mana istilah umum dicantumkan dengan nomor halaman sehingga dapat diikuti tanpa harus mencari seluruh buku. Antarmuka Pengguna untuk Pencarian:
Setelah basis data indeks dibuat, program dapat melakukan jenis pencarian apa pun. Untuk memudahkan pengguna melakukan pencarian, aplikasi harus menyediakan sarana atau antarmuka pengguna yang memungkinkan pengguna memasukkan teks dan memulai pencarian.

- Membangun Kueri- Saat pengguna meminta untuk mencari teks, aplikasi harus membuat objek kueri berdasarkan teks tersebut, yang dapat digunakan untuk mengkueri basis data indeks guna memperoleh informasi relevan.
- Hasil Render- Setelah hasil diperoleh, program harus memilih cara memberikan informasi kepada pengguna melalui antarmuka pengguna. Berapa banyak informasi yang harus ditampilkan.
- Eclipse adalah IDE Java yang merupakan salah satu dari 3 IDE terbesar dan terpopuler di dunia. Sebagian besar ditulis dalam Java, tetapi juga dapat digunakan untuk mengembangkan aplikasi dalam bahasa pemrograman lain selain Java menggunakan plug-in. Berikut ini adalah beberapa fitur Eclipse:
  - PDE (Plugin Development Environment) tersedia di Eclipse untuk programmer Java yang ingin membuat fungsi-fungsi tertentu dalam aplikasi mereka. Eclipse memamerkan alat-alat yang

hebat untuk berbagai proses dalam pengembangan aplikasi seperti pembuatan grafik, pemodelan, pelaporan, pengujian, dll. mungkin.Eclipse juga dapat digunakan untuk membuat berbagai dokumen matematika dengan LaTeX menggunakan plug-in TeXlipse serta paket untuk perangkat lunak Mathematica.Eclipse dapat digunakan pada platform seperti Linux, macOS, Solaris, dan Windows.

- HBase [Basis Data Hadoop NOSQL] HBase adalah model data yang mirip dengan tabel besar Google yang dirancang untuk menyediakan akses acak cepat ke sejumlah besar data terstruktur.



Gambar5. 3 *HBase [Basis Data Hadoop NoSQL]*

HBase adalah basis data berorientasi kolom terdistribusi yang dibangun di atas sistem berkas Hadoop. Ini adalah proyek sumber terbuka dan dapat diskalakan secara horizontal. HBase adalah model data yang mirip dengan tabel besar Google yang dirancang untuk menyediakan akses acak cepat ke sejumlah besar data terstruktur. Ini memanfaatkan toleransi kesalahan yang disediakan oleh Hadoop File System (HDFS). Ini adalah bagian dari ekosistem Hadoop yang menyediakan akses baca/tulis acak waktu nyata ke data dalam Hadoop File System. Seseorang dapat menyimpan data dalam HDFS baik secara langsung maupun melalui

HBase. Konsumen data membaca/mengakses data dalam HDFS secara acak menggunakan HBase. HBase berada di atas Hadoop File System dan menyediakan akses baca dan tulis.

Mekanisme Penyimpanan di HBase

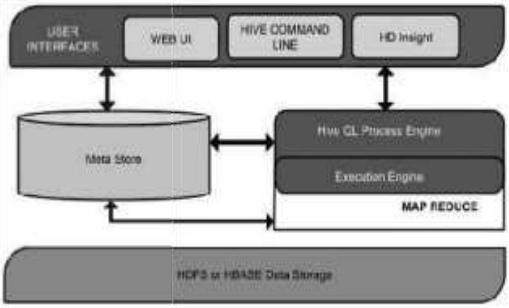
Rowid	Column Family			Column Family			Column Family			Column Family		
	col1	col2	col3	col1	col2	col3	col1	col2	col3	col1	col2	col3
1												
2												
3												

HBase adalah basis data berorientasi kolom dan tabel di dalamnya diurutkan berdasarkan baris. Skema tabel hanya mendefinisikan keluarga kolom, yang merupakan pasangan nilai kunci. Sebuah tabel memiliki beberapa keluarga kolom dan setiap keluarga kolom dapat memiliki sejumlah kolom. Nilai kolom berikutnya disimpan secara berurutan pada disk. Setiap nilai sel tabel memiliki stempel waktu. Singkatnya, dalam HBase:

- Hive [Mesin Kueri Data]- Hive adalah alat infrastruktur gudang data untuk memproses data terstruktur di Hadoop. Alat ini berada di atas Hadoop untuk meringkas Big Data, dan memudahkan pencarian dan analisis. Awalnya Hive dikembangkan oleh Facebook, kemudian Apache Software Foundation mengambilnya dan mengembangkannya lebih lanjut sebagai sumber terbuka dengan nama Apache Hive.Alat ini digunakan oleh berbagai perusahaan. Misalnya, Amazon menggunakannya di Amazon Elastic MapReduce. Hive tidak.

Basis data relasional. Desain untuk Pemrosesan Transaksi Daring (OLTP). Bahasa untuk kueri waktu nyata dan pembaruan tingkat baris. Menyimpan skema dalam basis data dan memproses data ke dalam HDFS. Dirancang untuk OLAP. Menyediakan bahasa tipe SQL untuk kueri yang disebut HiveQL atau HQL. Bahasa ini familier, cepat, dapat diskalakan, dan dapat diperluas.

Arsitektur Hive



Gambar 5. 4 *Arsitektur Hive*

Antarmuka Pengguna-Hive adalah program infrastruktur gudang data yang memungkinkan pengguna berinteraksi dengan HDFS. Hive menyediakan tiga antarmuka pengguna: UI Web Hive, baris perintah Hive, dan Hive HD Insight (di Windows Server)

- Pig [Platform Analisis Kumpulan Data Besar]- ApachePig merupakan abstraksi dari MapReduce. Ini adalah alat/platform yang digunakan untuk menganalisis kumpulan data yang lebih besar yang merepresentasikannya sebagai aliran data. Pig umumnya digunakan dengan Hadoop; kita dapat melakukan semua operasi manipulasi data di Hadoop menggunakan Apache

Pig. Untuk menulis program analisis data, Pig menyediakan bahasa tingkat tinggi yang dikenal sebagai Pig Latin.

***Gambar5.6** Pig[platform analisis kumpulan data besar]*

Bahasa ini menyediakan berbagai operator yang dapat digunakan oleh programmer untuk mengembangkan fungsi mereka sendiri guna membaca, menulis, dan memproses data. Untuk menganalisis data menggunakan Apache Pig, programmer perlu menulis skrip menggunakan bahasa Pig Latin. Semua skrip ini dikonversi secara internal ke tugas Map dan Reduce. Apache Pig memiliki komponen yang dikenal sebagai Pig Engine yang menerima skrip Pig Latin sebagai input dan mengonversi skrip tersebut ke dalam tugas MapReduce.

Ringkasan

- Apache Hadoop adalah seperangkat perangkat lunak sumber terbuka untuk memecahkan masalah yang melibatkan data dalam jumlah besar dan pemrosesan yang memanfaatkan jaringan banyak komputer. Ini adalah kerangka kerja perangkat lunak berbasis model pemrograman MapReduce untuk penyimpanan dan pemrosesan data besar yang terdistribusi.
- Big data mengacu pada data dalam jumlah besar yang sulit dikelola – baik yang terorganisasi maupun tidak terstruktur – yang membanjiri perusahaan setiap hari. Big data dapat dievaluasi untuk mendapatkan wawasan yang membantu orang membuat penilaian

yang lebih baik dan merasa lebih percaya diri dalam membuat keputusan bisnis yang penting.

- HDFS, atau Hadoop Distributed File System, adalah sistem berkas terdistribusi yang berjalan pada perangkat keras komoditas. Sistem ini memiliki banyak kesamaan dengan sistem berkas terdistribusi lainnya. Akan tetapi, terdapat perbedaan yang cukup besar antara sistem ini dan sistem berkas terdistribusi lainnya. HDFS dimaksudkan untuk berjalan pada perangkat keras berbiaya rendah dan sangat toleran terhadap kesalahan. HDFS adalah sistem berkas yang memungkinkan akses throughput tinggi ke data aplikasi dan sangat cocok untuk aplikasi dengan kumpulan data yang sangat besar. Untuk menyediakan akses streaming ke data sistem berkas, HDFS melonggarkan beberapa kriteria POSIX.
- Dalam kluster Hadoop, MapReduce adalah paradigma pemrograman yang memungkinkan skalabilitas luar biasa pada ratusan atau ribuan komputer. MapReduce, sebagai komponen pemrosesan, merupakan inti dari Apache Hadoop.
- Ekosistem Hadoop adalah platform atau rangkaian yang menawarkan berbagai layanan untuk mengatasi masalah big data. Ekosistem ini terdiri dari proyek Apache serta berbagai alat dan solusi komersial. HDFS, MapReduce, YARN, dan Hadoop Common adalah empat komponen inti Hadoop.
- Apache Pig adalah kerangka kerja tingkat tinggi untuk mengembangkan aplikasi berbasis Hadoop. Pig Latin adalah nama bahasa platform tersebut. Tugas Hadoop Pig dapat dijalankan di MapReduce, Apache Tez, atau Apache Spark.
- Eclipse adalah lingkungan pemrograman Java yang tangguh. Karena pemrograman Hadoop dan Mapreduce dilakukan di Java, kita harus

menggunakan Lingkungan Pengembangan Terpadu dengan banyak fitur (IDE)

- Jaql adalah salah satu bahasa yang digunakan untuk mengabstraksikan kerumitan arsitektur pemrograman MapReduce milik Hadoop. Bahasa ini merupakan bahasa fungsional dengan sintaksis yang diketik lemah dan evaluasi yang lambat.

Soal Latihan

Q1: Sistem komputer paralel dapat melakukan banyak hal.

A. Komputasi terdesentralisasi
B. Komputasi paralel
C. Komputasi terpusat
D. Semua ini

Q2: Proses pembuatan program paralel dikenal dengan _____

A. Komputasi paralel
B. Proses paralel
C. Pemrograman paralel
D. Pengembangan paralel

Q3: Pig paling peduli dengan ___ jumlah node.

A. Dua
B. Tiga
C. Empat
D. Lima

Q4: Pig pada dasarnya dikembangkan oleh _____

A. Twitter
B. Indonesia
C. Google
D. Bahasa Indonesia: Yahoo

Q5: HIVE melakukan _____ dan _____ pada set data besar.

A. Membaca dan menulis
B. Eksekusi dan penulisan
C. Keduanya
D. Tidak ada yang di atas

Q6:Hadoop adalah _____ yang tersedia di publik

A. Alat sumber terbuka
B. Alat komersial
C. Alat rumah
D. Alat vendor

Q7: Hadoop adalah kerangka kerja yang menyediakan terlalu banyak layanan seperti

A. Babi
B. Basis Data HB
C. Sarang lebah
D. Semua di atas

Q8: Alat Open-Source Terkait Hadoop Teratas adalah

A. Sarang lebah
B. Jaql
C. Babi
D. Semua di atas

Q9: Lucene adalah pustaka _____ berbasis Java yang sederhana namun kuat

A. Mencari
B. Pelaporan
C. Keduanya
D. Tidak ada yang di atas

Q10: _____ adalah IDE Java yang merupakan salah satu dari 3 IDE terbesar dan terpopuler di dunia

A. Cat

B. Buku catatan
C. Gerhana
D. Semua di atas

Q11: Konsep big data dan apa saja yang dicakupnya dapat dipahami lebih baik dengan empat V. Yaitu:

A. Volume
B. Kecepatan
C. Kebenaran
D. Semua di atas

Q12: _____ mengacu pada kepastian data dan bagaimana alat big data dan strategi analisis Anda dapat memisahkan data berkualitas buruk dari data yang benar-benar penting bagi bisnis Anda.

A. Volume
B. Kecepatan
C. Kebenaran
D. Semua di atas

Q13: Data dalam Big Data berukuran _____ byte.

A. Tanah
B. Mega
C. raksasa
D. Peta

Q14: Pilih jenis Big Data

A. Data Terstruktur
B. Data Tidak Terstruktur
C. Data Semi-terstruktur
D. Semua hal di atas

Q15: _____ merupakan komponen helikopter, pesawat terbang, dan jet, dll. Komponen ini menangkap suara awak pesawat, rekaman mikrofon dan earphone, serta informasi kinerja pesawat.

A. Data media sosial
B. Data kotak hitam
C. Data bursa saham
D. Data jaringan listrik

Jawaban untuk Penilaian Diri

Pertanyaan Ulasan

1. Perbedaan antara data mart dan data ware house.
2. Tuliskan Tips untuk Membuat Model Big Data yang Efektif.
3. Jelaskan berbagai jenis data mart.
4. Tuliskan keuntungan dan kerugian data mart.
5. Apa yang Anda pahami tentang streaming data? Jelaskan Kasus Penggunaan untuk Data Real-Time dan Streaming.

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*.

https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

BAB 6: Administrasi Hadoop

Weni Kurnia Sari, S.ST., M.Biomed

Tujuan

- Pelajari instalasi Hadoop langkah demi langkah
- Pelajari HDFS
- Pelajari tentang Arsitektur HDFS
- Pelajari Tujuan HDFS
- Pelajari perintah dasar di HDFS

6.1 HDFS

Sistem Berkas Hadoop dikembangkan menggunakan desain sistem berkas terdistribusi. Sistem ini dijalankan pada perangkat keras komoditas. Tidak seperti sistem terdistribusi lainnya, HDFS sangat toleran terhadap kesalahan dan dirancang menggunakan perangkat keras berbiaya rendah. HDFS menampung data dalam jumlah yang sangat besar dan menyediakan akses yang lebih mudah. Untuk menyimpan data sebesar itu, berkas-berkas tersebut disimpan di beberapa mesin. Berkas-berkas ini disimpan secara redundan untuk menyelamatkan sistem dari kemungkinan kehilangan data jika terjadi kegagalan. HDFS juga menyediakan aplikasi untuk pemrosesan paralel.

Fitur HDFS

- Cocok untuk penyimpanan dan pemrosesan terdistribusi.
- Hadoop menyediakan antarmuka perintah untuk berinteraksi dengan HDFS.
- Server bawaan namenode dan datanode membantu pengguna untuk memeriksa status kluster dengan mudah.
- Akses streaming ke data sistem berkas.
- HDFS menyediakan izin berkas dan autentikasi.

6.2 Arsitektur HDFS

HDFS memiliki arsitektur master/slave. Sebuah klaster HDFS terdiri dari satu NameNode, server master yang mengelola namespace sistem berkas dan mengatur akses ke berkas oleh klien. Selain itu, ada beberapa DataNode, biasanya satu per node dalam klaster, yang mengelola penyimpanan yang terpasang pada node tempat mereka berjalan. HDFS mengekspos namespace sistem berkas dan memungkinkan data pengguna disimpan dalam berkas. Secara internal, berkas dibagi menjadi satu atau beberapa blok dan blok-blok ini disimpan dalam satu set DataNode. NameNode menjalankan operasi namespace sistem berkas seperti membuka, menutup, dan mengganti nama berkas dan direktori. NameNode juga menentukan pemetaan blok ke DataNode. DataNode bertanggung jawab untuk melayani permintaan baca dan tulis dari klien sistem berkas. DataNode juga melakukan pembuatan, penghapusan, dan replikasi blok berdasarkan instruksi dari NameNode.

- HDFS mengikuti arsitektur master-slave
- N*ameNode*-NameNode merupakan bagian utama dari sistem berkas HDFS. Node ini menyimpan pohon direktori semua berkas dalam sistem berkas, dan melacak lokasi penyimpanan data berkas di seluruh klaster. Node ini tidak menyimpan sendiri data berkas-berkas ini.

  Aplikasi klien berkomunikasi dengan Name Node kapan pun mereka ingin menemukan berkas, atau saat mereka ingin menambahkan/menyalin/memindahkan/menghapus berkas. Name Node menanggapi permintaan yang berhasil dengan mengembalikan daftar server Data Node yang relevan tempat data berada. Name Node merupakan Titik Kegagalan Tunggal untuk Klaster HDFS. HDFS saat ini bukan sistem High Availability. Saat Name Node mati, sistem berkas akan offline. Ada Name Node Sekunder opsional yang dapat dihosting di mesin terpisah. Node

ini hanya membuat titik pemeriksaan namespace dengan menggabungkan berkas suntingan ke dalam berkas gambar fs dan tidak menyediakan redundansi yang sebenarnya. Hadoop 0.21+ memiliki Node Nama Cadangan yang merupakan bagian dari rencana untuk memiliki layanan nama HA, tetapi memerlukan kontribusi aktif dari orang-orang yang menginginkannya (yaitu Anda) untuk membuatnya Sangat Tersedia. Melacak di mana di seluruh klaster data file disimpan. Node ini tidak menyimpan sendiri data file-file ini. Aplikasi klien berkomunikasi dengan Node Nama. Node Nama menanggapi permintaan yang berhasil. Node Nama berfungsi sebagai Master di klaster Hadoop. Di bawah ini tercantum fungsi utama yang dilakukan oleh Node Nama:

1. Menyimpan metadata data aktual.
2. Mengelola namespace sistem berkas.

Mengatur permintaan akses klien untuk data file aktual. Menetapkan pekerjaan ke Slaves (DataNode). Menjalankan operasi namespace sistem file seperti membuka/menutup file, mengganti nama file dan direktori. Karena Name node menyimpan metadata dalam memori untuk pengambilan cepat, sejumlah besar memori diperlukan untuk operasinya. Ini harus dihosting pada perangkat keras yang andal.

- simpul data-**simpul data**berfungsi sebagai Slave di cluster Hadoop. Berikut ini adalah fungsi utama yang dilakukan oleh Data Node:
  1. Sebenarnya menyimpan data Bisnis.
  2. Ini adalah simpul pekerja sebenarnya di mana pemrosesan Baca/Tulis/Data ditangani. Atas instruksi dari Master, ia melakukan pembuatan/replikasi/penghapusan blok data. Karena semua data Bisnis disimpan di Data Node, sejumlah besar penyimpanan diperlukan untuk operasinya. Perangkat keras komoditas dapat digunakan untuk menghosting Data Node. Data Node bertanggung jawab untuk menyimpan data aktual di HDFS.

Data Node juga dikenal sebagai Slave. Name Node dan Data Node berkomunikasi secara konstan. DataNode bertanggung jawab untuk menyimpan data aktual di HDFS. DataNode juga dikenal sebagai Slave. NameNode dan DataNode berkomunikasi secara konstan. Ketika Data Node mati, itu tidak memengaruhi ketersediaan data atau klaster. Name Node akan mengatur replikasi untuk blok yang dikelola oleh Data Node yang tidak tersedia. DataNode biasanya dikonfigurasi dengan banyak ruang hard disk. Karena data aktual disimpan di DataNode. DataNode secara berkala mengirim HEARTBEATS ke NameNode. DataNode bertanggung jawab untuk menyimpan data aktual di HDFS. DataNode juga dikenal sebagai Slave. NameNode dan DataNode berkomunikasi secara konstan.

- Memblokir-Umumnya, data pengguna disimpan dalam berkas HDFS. Berkas dalam sistem berkas akan dibagi menjadi satu atau beberapa segmen dan/atau disimpan dalam simpul data individual. Segmen berkas ini disebut sebagai blok. Dengan kata lain, jumlah minimum data yang dapat dibaca atau ditulis oleh HDFS disebut Blok. Ukuran blok default adalah 64 MB, tetapi dapat ditingkatkan sesuai kebutuhan untuk mengubah konfigurasi HDFS.

  Tujuan HDFS

  - Deteksi dan pemulihan kesalahan− Karena HDFS mencakup sejumlah besar perangkat keras komoditas, kegagalan komponen sering terjadi. Oleh karena itu, HDFS harus memiliki mekanisme untuk deteksi dan pemulihan kesalahan yang cepat dan otomatis.
  - Kumpulan data yang sangat besar− HDFS harus memiliki ratusan node per klaster untuk mengelola aplikasi yang memiliki kumpulan data besar.

- Perangkat keras pada data− Tugas yang diminta dapat dilakukan secara efisien, saat komputasi dilakukan di dekat data. Terutama jika melibatkan kumpulan data besar, hal ini mengurangi lalu lintas jaringan dan meningkatkan throughput.

Ringkasan

- Apache Hadoop adalah kerangka kerja perangkat lunak sumber terbuka berbasis Java untuk mengelola pemrosesan dan penyimpanan data dalam aplikasi data besar. Hadoop bekerja dengan memecah set data besar dan pekerjaan analitis menjadi beban kerja yang lebih kecil yang dapat ditangani secara paralel di seluruh node dalam kluster komputasi. Hadoop dapat menangani data yang terorganisasi dan tidak terstruktur, dan dapat ditingkatkan dari satu server menjadi ribuan server dengan mudah.
- Java adalah bahasa pemrograman berorientasi objek dengan tingkat abstraksi yang tinggi dan dependensi implementasi sesedikit mungkin.

Soal Latihan

1._____ adalah prasyarat utama untuk Hadoop.

A. Jawa
B. Bahasa Pemrograman HTML
C. C#
D. Tidak ada yang di atas

2.Pada node _____ pelacak pekerjaan berjalan.

A. Datanode
B. NamaNode
C. Nama node sekunder

D. Node data sekunder

3. Perintah mana yang digunakan untuk memverifikasi keberadaan Java di sistem Anda?

A. $java +versi
B. $java @versi
C. $java -versi
D. $java =versi

4.Pilih paket Java SE.

A. JRE
B. JDK
C. Keduanya
D. Tidak ada yang di atas

5. Perintah untuk membuat pengguna Hadoop

A. penggunatambahkan nama pengguna
B. hadoopadd nama pengguna
C. nama pengguna addhadoop
D. Tidak ada yang di atas

6. Cluster Hadoop beroperasi dalam tiga mode yang didukung. Mode-mode tersebut adalah _____

A. Mode Lokal/Mandiri
B. Mode Terdistribusi Semu
C. Mode Terdistribusi Penuh
D. Semua di atas

7.Pilih perintah untuk memulai semua daemon Hadoop DFS.

A. semua-lari.sh
B. mulai-semua.sh
C. mulai-dfs.sh
D. jalankan-dfs.sh

8.Pilih perintah untuk menghentikan semua daemon Hadoop DFS.

A. hentikan-semua.sh

B. semua-berhenti.sh
C. tahan-semua.sh
D. batalkan-semua.sh

9.JAVA_HOME diatur dalam _____

A. hadoop-env.sh
B. hadoop-environment.sh
C. env-hadoop.sh
D. Tidak ada yang di atas

10. Pada hadoop-env.sh, properti berikut mana yang dikonfigurasi

A. Faktor replikasi
B. Nama direktori untuk menyimpan file hdfs C. Host dan port tempat tugas MapReduce dijalankan

D. Variabel Lingkungan Java.

11. Ketika komputer ditetapkan sebagai datanode, ruang disk yang tersedia untuknya berkurang.

A. Hanya dapat digunakan untuk penyimpanan HDFS
B. Dapat digunakan untuk penyimpanan HDFS dan non-HDF
C. Tidak dapat diakses oleh perintah non-hadoop
D. Tidak dapat menyimpan berkas teks.

12. HDFS adalah singkatan dari

A. Pencarian Berkas Hadoop
B. Peralihan Berkas Hadoop
C. Sistem Berkas Hadoop
D. Sistem Lapangan Hadoop

13. HDFS beroperasi dengan cara _____.

A. Mode pekerja ahli
B. Pekerja-tuan
C. Tuan dan budak
D. Mode tuan budak

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

BAB 7: Arsitektur Hadoop

Desi anggreani, S.Kom.,M.T.

Tujuan

- Pelajari apa itu hadoop
- Memahami komponen inti Hadoop
- Pelajari Cara Kerja Hdfs.
- Apa itu Hadoop Cluster
- Pelajari Arsitektur Cluster Hadoop
- Arsitektur HDFS dan fitur Hadoop

7.1Apa itu Hadoop Distribute File System (HDFS)

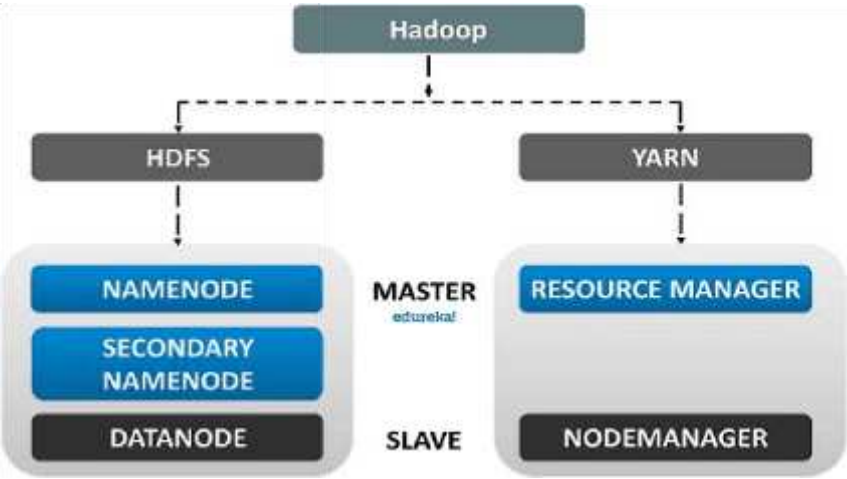
Sulit untuk mengelola data dalam jumlah besar dalam satu mesin. Oleh karena itu, data harus dipecah menjadi potongan-potongan yang lebih kecil dan disimpan di beberapa mesin. Sistem berkas yang mengelola penyimpanan di seluruh jaringan mesin disebut sistem berkas terdistribusi. Hadoop Distributed File System (HDFS) adalah komponen penyimpanan Hadoop. Semua data yang disimpan di Hadoop disimpan secara terdistribusi di seluruh kluster mesin. Namun, ada beberapa properti yang menentukan keberadaannya. Hadoop Distributed File System (HDFS) adalah komponen penyimpanan Hadoop. Semua data yang disimpan di Hadoop disimpan secara terdistribusi di seluruh kluster mesin. Namun, ada beberapa properti yang menentukan keberadaannya.

- **Volume Besar**– Karena merupakan sistem berkas terdistribusi, ia sangat mampu menyimpan petabyte data tanpa gangguan apa pun.
- **Akses Data**– Berdasarkan filosofi bahwa “pola pemrosesan data yang paling efektif adalah menulis sekali, pola membaca berkali-kali”.

- **Hemat biaya**– HDFS berjalan pada sekumpulan perangkat keras komoditas. Ini adalah mesin murah yang dapat dibeli dari vendor mana pun.

7.2 Komponen Hadoop

HDFS terdiri dari komponen-komponen berikut

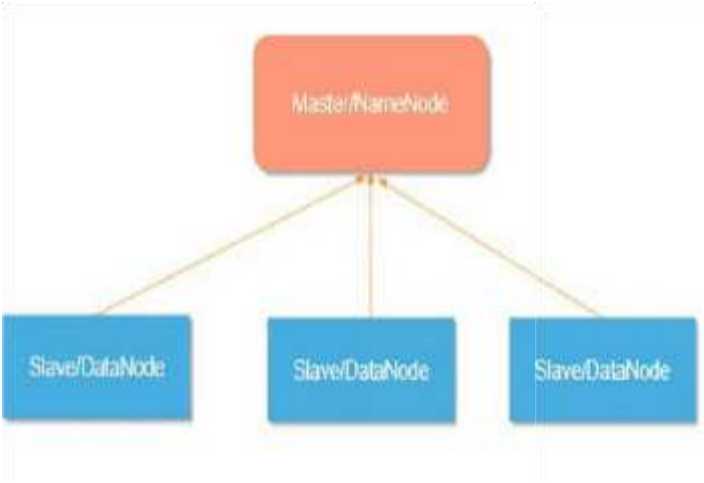


***Gambar 7.1**: Komponen HDFS*

HDFS beroperasi dalam arsitektur master-slave, artinya ada satu node master dan beberapa node slave dalam kluster. Node master adalah Namenode.

- **NameNode** adalah simpul utama yang berjalan pada simpul terpisah dalam kluster. Mengelola namespace sistem berkas yang merupakan hierarki atau pohon sistem berkas dari berkas dan direktori. Menyimpan informasi seperti pemilik berkas, izin berkas, dll. untuk semua berkas. Ia juga mengetahui lokasi semua blok berkas dan ukurannya.

Nama node dalam HDFS



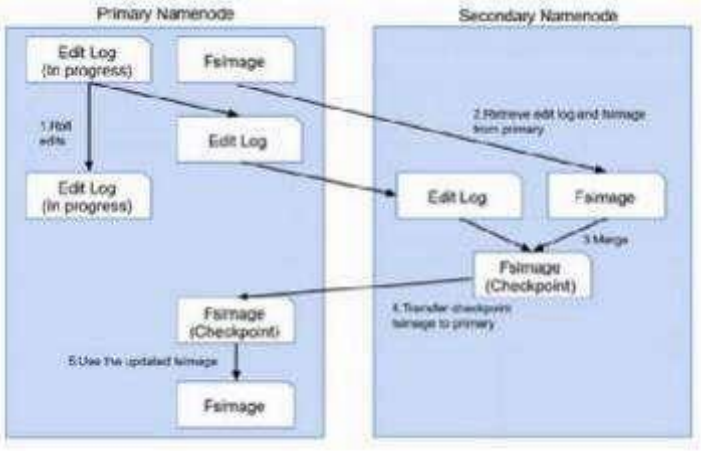
***Gambar 7.2**: Namenode di HDFS*

Semua informasi ini disimpan secara persisten pada disk lokal dalam bentuk dua file: Fsimage dan Edit Log.

Setiap kali klien ingin menulis informasi ke HDFS atau membaca informasi dari HDFS, klien akan terhubung dengan Namenode. Namenode mengembalikan lokasi blok ke klien dan operasi pun dilakukan. Ya, benar, Namenode tidak menyimpan blok. Untuk itu, kita memiliki node terpisah. Datanode adalah node pekerja. Node pekerja adalah perangkat keras komoditas murah yang dapat dengan mudah ditambahkan ke kluster. Node pekerja secara berkala mengirimkan detak jantung ke Namenode sehingga Namenode mengetahui kesehatannya. Dengan itu, DataNode juga mengirimkan daftar blok yang disimpan di dalamnya sehingga Namenode dapat mempertahankan pemetaan blok ke Datanode dalam memorinya. Namun, selain kedua jenis node ini di kluster, ada juga node lain yang disebut Namenode Sekunder. Datanode bertanggung jawab untuk menyimpan, mengambil, mereplikasi, menghapus, dll. blok saat

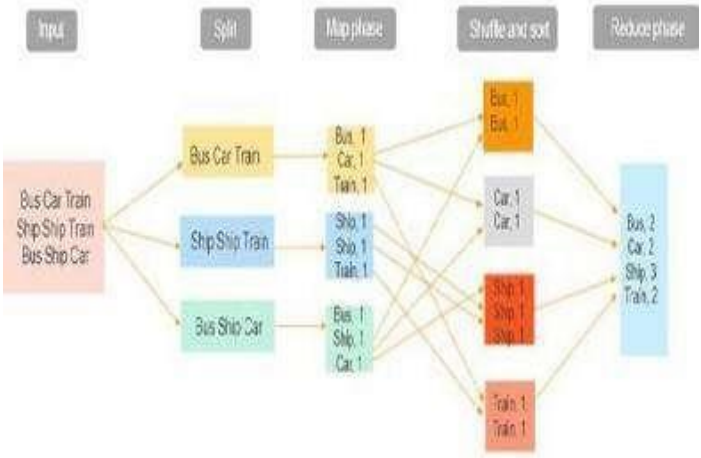
jantung ke Namenode sehingga Namenode mengetahui kesehatannya. Dengan itu, DataNode juga mengirimkan daftar blok yang tersimpan di dalamnya sehingga Namenode dapat memelihara pemetaan blok ke Datanode dalam memorinya. Namun selain kedua jenis node tersebut di dalam kluster, terdapat pula node lain yang disebut Namenode Sekunder.

Misalkan kita perlu me-restart Namenode, yang dapat terjadi jika terjadi diminta oleh Namenode. Node pekerja secara berkala mengirimkan detak kegagalan. Ini berarti kita harus menyalin Fsimage dari disk ke memori. Selain itu, kita juga harus menyalin salinan Edit Log terbaru ke Fsimage untuk melacak semua transaksi. Namun jika kita me-restart node setelah waktu yang lama, maka Edit log dapat bertambah besar. Ini berarti akan membutuhkan banyak waktu saatnya untuk menerapkan transaksi dari log Edit. Dan selama waktu ini, sistem berkas akan offline. Oleh karena itu, untuk mengatasi masalah ini, kami menggunakan Secondary Namenode.Secondary Namenode adalah node lain yang ada di cluster yang tugas utamanya adalah menggabungkan log Edit dengan Fsimage secara teratur dan menghasilkan pemeriksaan-titik metadata sistem berkas dalam memori primer. Ini juga disebut sebagai Checkpointing.



***Gambar7.3**: Namenode primer dan Namenode sekunder*

- *Namenode Secondary di HDFS***:** Secondary namenode berjalan pada node terpisah di cluster. Namun, terlepas dari namanya, Secondary Namenode tidak bertindak sebagai Namenode. Ia hanya ada untuk Checkpointing dan menyimpan salinan Fsimage terbaru.
- *MapReduce:* Ini adalah lapisan pemrosesan di Hadoop. Hadoop MapReduce memproses data yang disimpan di Hadoop HDFS secara paralel di berbagai node dalam kluster. Ia membagi tugas yang dikirimkan oleh pengguna ke dalam tugas independen dan memprosesnya sebagai subtugas di seluruh perangkat keras komoditas. Hadoop MapReduce adalah unit pemrosesan Hadoop. Dalam pendekatan MapReduce, pemrosesan dilakukan di node slave, dan hasil akhir dikirim ke node master. Data yang berisi kode digunakan untuk memproses seluruh data. Data berkode ini biasanya sangat kecil dibandingkan dengan data itu sendiri. Anda hanya perlu mengirim kode berukuran beberapa kilobyte untuk melakukan proses berat di komputer.

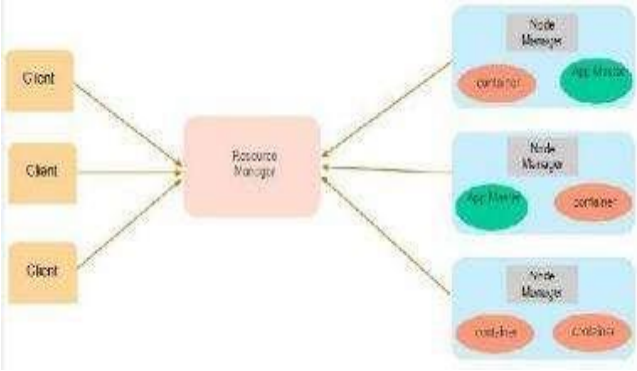


Gambar 7.4: *MapReduce*

Kumpulan data masukan pertama-tama dibagi menjadi beberapa bagian data. Dalam contoh ini, masukan memiliki tiga baris teks dengan tiga entitas terpisah - "kereta gerbong bus," "kereta kapal," "kereta bus kapal." Kumpulan data kemudian dibagi menjadi tiga bagian, berdasarkan entitas-entitas ini, dan diproses secara paralel. Dalam fase pemetaan, data diberi kunci dan nilai 1. Dalam kasus ini, kita memiliki satu bus, satu gerbong, satu kapal, dan satu kereta. Pasangan kunci-nilai ini kemudian diacak dan diurutkan bersama berdasarkan kuncinya. Pada fase pengurangan, agregasi terjadi, dan keluaran akhir diperoleh.

- Hadoop YARN adalah singkatan dari Yet Another Resource Negotiator. Ini adalah unit manajemen sumber daya Hadoop dan tersedia sebagai komponen Hadoop versi 2. Hadoop YARN berfungsi seperti OS untuk Hadoop. Ini adalah sistem file yang dibangun di atas HDFS. Sistem ini bertanggung jawab untuk mengelola sumber daya kluster guna memastikan Anda tidak membebani satu mesin. Sistem ini

melakukan penjadwalan pekerjaan guna memastikan bahwa pekerjaan dijadwalkan di tempat yang tepat.



Gambar 7.5: *Hadoop YARN*

Misalkan mesin klien ingin melakukan kueri atau mengambil beberapa kode untuk analisis data. Permintaan pekerjaan ini ditujukan ke manajer sumber daya (Hadoop Yarn), yang bertanggung jawab atas alokasi dan manajemen sumber daya. Di bagian node, setiap node memiliki manajer node-nya sendiri. Manajer node ini mengelola node dan memantau penggunaan sumber daya di node tersebut. Kontainer berisi kumpulan sumber daya fisik, yang dapat berupa RAM, CPU, atau hard drive. Setiap kali permintaan pekerjaan masuk, master aplikasi meminta kontainer dari manajer node. Setelah manajer node mendapatkan sumber daya, ia kembali ke Manajer Sumber Daya.

- *Daemon Hadoop* adalah proses yang berjalan di latar belakang. Keempat daemon ini berjalan agar Hadoop berfungsi.

Daemon Hadoop adalah:

a) *NameNode*– Berjalan pada node master untuk HDFS.
b) *Datanode*– Berjalan pada node slave untuk HDFS.

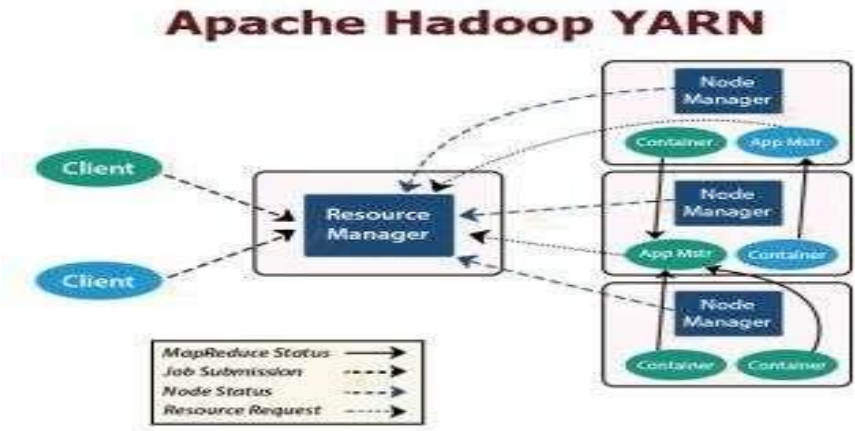
c) *Manajer Sumber Daya*– Berjalan pada node master YARN untuk MapReduce.

d) *Manajer Node*– Berjalan pada node slave YARN untuk MapReduce.

Setiap kali klien ingin melakukan pemrosesan apa pun pada datanya di kluster Hadoop, maka ia terlebih dahulu menyimpan data di Hadoop HDFS dan kemudian menulis program MapReduce untuk memproses Data. Hadoop MapReduce bekerja sebagai berikut:

1. Hadoop membagi pekerjaan menjadi dua jenis tugas, yaitu tugas pemetaan dan tugas pengurangan. YARN menjadwalkan tugas-tugas ini (yang akan kita lihat nanti dalam artikel ini). Tugas-tugas ini berjalan pada DataNode yang berbeda.
2. Input untuk pekerjaan MapReduce dibagi menjadi beberapa bagian berukuran tetap yang disebut pemisahan input.
3. Satu tugas peta yang menjalankan fungsi peta yang ditentukan pengguna untuk setiap rekaman dalam pemisahan input dibuat untuk setiap pemisahan input. Tugas peta ini berjalan pada DataNode tempat data input berada.
4. Keluaran dari tugas peta adalah keluaran antara dan ditulis ke disk lokal.
5. Output antara dari tugas pemetaan diacak dan diurutkan, lalu diteruskan ke reducer.
6. Untuk satu tugas reduksi, output antara mapper yang diurutkan diteruskan ke node tempat tugas reducer berjalan. Output ini kemudian digabungkan dan diteruskan ke fungsi reduce yang ditentukan pengguna.
7. Fungsi reduce meringkas output dari mapper dan menghasilkan output. Output dari reducer disimpan di HDFS.
8. Untuk beberapa fungsi reduce, pengguna menentukan jumlah reducer. Bila ada beberapa tugas reduce, tugas map membagi output-nya, membuat satu partisi untuk setiap tugas reduce.

Bagaimana HDFS Bekerja?



Gambar 7.6: *Apache Hadoop YARN*

YARN adalah lapisan manajemen sumber daya di Hadoop. Lapisan ini menjadwalkan tugas di kluster Hadoop dan menetapkan sumber daya ke aplikasi yang berjalan di kluster. Lapisan ini bertanggung jawab untuk menyediakan sumber daya komputasi yang dibutuhkan untuk menjalankan aplikasi. Ada dua daemon YARN yang berjalan di kluster Hadoop untuk melayani layanan inti YARN. Daemon tersebut adalah:

a. Manajer Sumber Daya: Ini adalah daemon utama YARN. Ia berjalan pada node utama per klaster untuk mengelola sumber daya di seluruh klaster. ResourceManager memiliki dua komponen utama yaitu Scheduler dan ApplicationManager. Penjadwal mengalokasikan sumber daya ke berbagai aplikasi yang berjalan di kluster. Manajer Aplikasi mengambil pekerjaan yang dikirimkan oleh klien, dan menegosiasikan wadah untuk menjalankan ApplicationMaster khusus aplikasi, dan memulai ulang Kontainer ApplicationMaster jika terjadi kegagalan.

b. Manajer Node: NodeManager adalah daemon slave dari YARN. Ia berjalan pada semua node slave dalam kluster. Ia bertanggung jawab untuk meluncurkan dan mengelola kontainer pada node. Kontainer menjalankan proses khusus aplikasi dengan serangkaian sumber daya terbatas seperti memori, CPU, dan sebagainya. Saat NodeManager dimulai, ia memberitahukan dirinya kepada ResourceManager. Ia secara berkala mengirimkan detak jantung ke ResourceManager. Ia menawarkan sumber daya ke kluster.
c. Master Aplikasi: ApplicationMaster per aplikasi menegosiasikan kontainer dari penjadwal dan melacak status kontainer serta memantau kemajuan kontainer. Klien mengirimkan aplikasi ke ResourceManager. ResourceManager menghubungi NodeManager yang meluncurkan dan memantau kontainer komputasi pada node dalam kluster. Kontainer mengeksekusi ApplicationMaster.

Tugas MapReduce dan ApplicationMaster berjalan dalam kontainer yang dijadwalkan oleh ResourceManager dan dikelola oleh NodeManager.

7.3 Rangkum Cara Kerja Hadoop Secara Internal

1. HDFS membagi data masukan klien menjadi blok berukuran 128 MB. Bergantung pada faktor replikasi, replika blok dibuat. Blok dan replikanya disimpan di DataNode yang berbeda.
2. Setelah semua blok disimpan pada HDFS DataNodes, pengguna dapat memproses data.
3. Untuk memproses data, klien mengirimkan program MapReduce ke Hadoop.
4. Manajer Sumber Daya, kemudian menjadwalkan program yang dikirimkan oleh pengguna pada node individual di kluster.
5. Setelah semua node selesai diproses, output ditulis kembali ke HDFS.

7.4 Gugus Hadoop

Cluster Hadoop tidak lain adalah sekelompok komputer yang terhubung melalui LAN. Kita menggunakannya untuk menyimpan dan memproses kumpulan data besar. Cluster Hadoop memiliki sejumlah perangkat keras komoditas yang terhubung bersama. Mereka berkomunikasi dengan mesin canggih yang bertindak sebagai master. Master dan slave ini menerapkan komputasi terdistribusi melalui penyimpanan data terdistribusi. Ia menjalankan perangkat lunak sumber terbuka untuk menyediakan fungsionalitas terdistribusi.

Fungsi Nama Node Mengelola namespace sistem berkas, Mengatur akses ke berkas oleh klien, Menyimpan metadata data aktual misalnya – jalur berkas, jumlah blok, ID blok, lokasi blok, dll.Menjalankan operasi namespace sistem berkas seperti membuka, menutup, mengganti nama berkas dan direktori. Node Nama menyimpan metadata dalam memori untuk pengambilan cepat. Oleh karena itu, kita harus mengonfigurasinya pada mesin kelas atas. Fungsi Manajer Sumber Daya. Ini menengahi sumber daya di antara node yang bersaing, Melacak node hidup dan mati

Menyimpan data bisnis, melakukan operasi pembacaan, penulisan, dan pemrosesan data, berdasarkan instruksi dari master, melakukan pembuatan, penghapusan, dan replikasi blok data.

Fungsi Manajer Node. menjalankan layanan pada node untuk memeriksa kesehatannya dan melaporkannya ke Resource Manager. Kita dapat dengan mudah menskalakan cluster Hadoop dengan menambahkan lebih banyak node ke dalamnya. Oleh karena itu, kita menyebutnya cluster berskala linier. Setiap node yang ditambahkan akan meningkatkan throughput cluster. Kami menginstal Hadoop dan mengonfigurasinya pada node klien Fungsi Node Klien Untuk memuat data pada klaster Hadoop, memberi tahu cara memproses data dengan mengirimkan pekerjaan MapReduce. Mengumpulkan keluaran dari lokasi yang ditentukan.

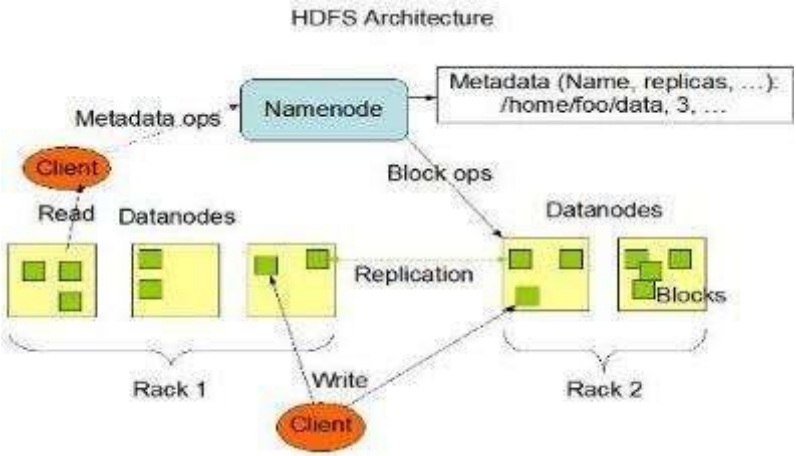
7.5 Apa itu Hadoop High Availability?

Dengan Hadoop 2.0, kami memiliki dukungan untuk beberapa Name Node dan dengan Hadoop 3.0 kami memiliki node siaga. Ini mengatasi masalah SPOF (Single Point Of Failure) dengan menggunakan Name Node tambahan (Passive Standby Name Node) untuk failover otomatis. Ini adalah ketersediaan tinggi di Hadoop.

7.6 Arsitektur HDFS

Arsitektur ini memberi Anda gambaran lengkap tentang Hadoop Distributed File System. Ada satu NameNode yang menyimpan metadata, dan ada beberapa DataNode yang melakukan pekerjaan penyimpanan aktual. Node disusun dalam rak, dan replika blok data disimpan di rak yang berbeda di kluster untuk memberikan toleransi kesalahan. Di bagian yang tersisa dari tutorial ini, kita akan melihat bagaimana operasi baca dan tulis dilakukan di HDFS? Untuk membaca atau menulis file di HDFS, klien perlu berinteraksi dengan NameNode. Aplikasi HDFS memerlukan model akses tulis-sekali-baca-banyak untuk file. File, setelah dibuat dan ditulis, tidak dapat diedit. NameNode menyimpan metadata, dan DataNode menyimpan data aktual. Klien berinteraksi dengan NameNode untuk

melakukan tugas apa pun, karena NameNode adalah pusat kluster. Ada beberapa DataNode di kluster yang menyimpan data HDFS di disk lokal. DataNode mengirimkan pesan detak jantung ke NameNode secara berkala untuk menunjukkan bahwa ia aktif. Selain itu, ia mereplikasi data ke DataNode lain sesuai dengan faktor replikasi.



Gambar 13: Arsitektur HDFS

Penyimpanan Terdistribusi. HDFS menyimpan data secara terdistribusi. HDFS membagi data menjadi potongan-potongan kecil dan menyimpannya di berbagai Node Data dalam klaster. Dengan cara ini, Hadoop Distributed File System menyediakan cara bagi MapReduce untuk memproses sebagian dari kumpulan data besar yang dipecah menjadi blok, secara paralel di beberapa node. MapReduce adalah jantung Hadoop, tetapi HDFS-lah yang menyediakan semua kemampuan ini.

Blok HDFS membagi file besar menjadi potongan-potongan kecil yang dikenal sebagai blok. Blok adalah unit data terkecil dalam sistem berkas. Kami (klien dan admin) tidak memiliki kontrol apa pun pada blok seperti lokasi blok. NameNode memutuskan semua hal tersebut. Ukuran blok default HDFS adalah 128 MB. Kami dapat menambah atau mengurangi ukuran blok sesuai kebutuhan kami. Ini tidak seperti sistem

berkas OS, di mana ukuran blok adalah 4 KB. Jika ukuran data kurang dari ukuran blok HDFS, maka ukuran blok akan sama dengan ukuran data. Misalnya, jika ukuran file adalah 129 MB, maka 2 blok akan dibuat untuknya. Satu blok akan berukuran default 128 MB, dan yang lainnya hanya akan berukuran 1 MB dan bukan 128 MB karena akan membuang-buang ruang (di sini ukuran blok sama dengan ukuran data). Hadoop cukup cerdas untuk tidak membuang-buang sisa 127 MB. Jadi, ia mengalokasikan 1 blok MB hanya untuk data 1 MB. Keuntungan utama menyimpan data dalam ukuran blok seperti itu adalah menghemat waktu pencarian disk dan keuntungan lainnya adalah dalam hal pemrosesan pemeta memproses 1 blok dalam satu waktu. Jadi, 1 pemeta memproses data besar dalam satu waktu.

Replikasi Hadoop HDFS membuat salinan duplikat dari setiap blok. Ini dikenal sebagai replikasi. Semua blok direplikasi dan disimpan pada DataNode yang berbeda di seluruh kluster. Ia mencoba menempatkan setidaknya 1 replika di rak yang berbeda.

Ketersediaan Tinggi Replikasi blok data dan penyimpanannya pada beberapa node di seluruh kluster menyediakan ketersediaan data yang tinggi. Seperti yang terlihat sebelumnya dalam tutorial Hadoop HDFS ini, faktor replikasi default adalah 3, dan kita dapat mengubahnya ke nilai yang diperlukan sesuai dengan kebutuhan dengan mengedit file konfigurasi (hdfs-site.xml).

Keandalan Data Seperti yang telah kita lihat dalam ketersediaan tinggi dalam tutorial HDFS ini, data direplikasi dalam HDFS; data juga disimpan dengan andal. Berkat replikasi, blok sangat tersedia bahkan jika beberapa node mogok atau beberapa perangkat keras gagal. Jika DataNode gagal, blok tersebut dapat diakses dari DataNode lain yang berisi replika blok tersebut. Selain itu, jika rak mati, blok tersebut masih tersedia di rak yang berbeda. Beginilah cara data disimpan dengan andal dalam HDFS dan menyediakan toleransi kesalahan dan ketersediaan tinggi.

Toleransi Kesalahan HDFS menyediakan lapisan penyimpanan yang toleran terhadap kesalahan untuk Hadoop dan komponen lain dalam ekosistem. HDFS bekerja dengan perangkat keras komoditas (sistem

dengan konfigurasi rata-rata) yang memiliki peluang tinggi untuk mengalami kerusakan sewaktu-waktu. Jadi, untuk membuat seluruh sistem sangat toleran terhadap kesalahan, HDFS mereplikasi dan menyimpan data di tempat yang berbeda.

Skalabilitas berarti memperluas atau mempersempit klaster. Kita dapat meningkatkan skala Hadoop HDFS dengan 2 cara.

1. **Skala Vertikal:**Kita dapat menambahkan lebih banyak disk pada node klaster. Untuk melakukannya, kita perlu mengedit berkas konfigurasi dan membuat entri terkait disk yang baru ditambahkan. Di sini kita perlu menyediakan waktu henti meskipun sangat singkat. Jadi orang-orang umumnya lebih menyukai cara penskalaan kedua, yaitu penskalaan horizontal.
2. **Skala Horizontal:**Pilihan skalabilitas lainnya adalah menambahkan lebih banyak node ke klaster dengan cepat tanpa waktu henti. Ini dikenal sebagai penskalaan horizontal. Kita dapat menambahkan node sebanyak yang kita inginkan ke klaster dengan cepat dan real-time tanpa waktu henti. Ini adalah fitur unik yang disediakan oleh Hadoop.

Akses Throughput Tinggi ke Data Aplikasi Hadoop Distributed File System menyediakan akses throughput tinggi ke data aplikasi. Throughput adalah jumlah pekerjaan yang dilakukan dalam satuan waktu. Throughput menggambarkan seberapa cepat data diakses dari sistem, dan biasanya digunakan untuk mengukur kinerja sistem. Dalam HDFS, ketika kita ingin melakukan tugas atau tindakan, maka pekerjaan dibagi dan dibagikan di antara berbagai sistem. Jadi semua sistem akan menjalankan tugas yang diberikan kepada mereka secara independen dan paralel. Jadi pekerjaan akan selesai dalam waktu yang sangat singkat. Jadi, dengan membaca data secara paralel, HDFS memberikan throughput yang baik.

Ringkasan

- Apache Hadoop adalah kerangka kerja perangkat lunak sumber terbuka berbasis Java untuk mengelola pemrosesan dan penyimpanan data dalam aplikasi data besar. Hadoop bekerja dengan memecah set data besar dan pekerjaan analitis menjadi beban kerja yang lebih kecil yang dapat ditangani secara paralel di seluruh node dalam kluster komputasi.
- Aplikasi Hadoop menggunakan Hadoop Distributed File Solution (HDFS) sebagai sistem penyimpanan data utamanya. HDFS adalah sistem file terdistribusi yang menggunakan arsitektur NameNode dan DataNode untuk memungkinkan akses data berkinerja tinggi di seluruh kluster Hadoop yang sangat skalabel.
- YARN adalah salah satu komponen utama Apache Hadoop, dan bertugas menetapkan sumber daya sistem ke banyak aplikasi yang beroperasi di klaster Hadoop, serta menjadwalkan pekerjaan untuk dijalankan di berbagai node klaster.
- MapReduce sangat cocok untuk komputasi iteratif dengan sejumlah besar data yang memerlukan pemrosesan paralel. Alih-alih sebuah metode, MapReduce menggambarkan aliran data. MapReduce dapat digunakan untuk memproses grafik secara paralel. Tahapan pemetaan, pengacakan, dan pengurangan dari algoritma grafik semuanya mengikuti pola yang sama.
- Sistem berkas HDFS dibangun di sekitar NameNode. Sistem ini mengelola pohon direktori semua berkas dalam sistem berkas dan mencatat tempat penyimpanan data berkas di seluruh kluster.... Sebagai respons terhadap kueri yang berhasil, NameNode mengembalikan daftar server DataNode yang relevan tempat data disimpan.

Soal Latihan

1. Sistem berkas yang mengelola penyimpanan di seluruh jaringan mesin disebut _____

A. Sistem berkas terdistribusi
B. Sistem medan terdistribusi
C. Peralihan file terdistribusi
D. Tidak ada yang di atas

2.Pilih lapisan Hadoop yang benar.

A. HDFS
B. MapReduce
C. Yarn
D. Semua hal di atas

3. HDFS beroperasi dalam arsitektur master-slave, artinya ada satu node master dan beberapa node slave dalam klaster. Node master adalah _____.

A. Datanode
B. NamaNode
C. Keduanya
D. Semua hal di atas

4.Manakah dari daemon Hadoop berikut ini?

A. Manajer Sumber Daya
B. Datanode
C. NamaNode
D. Semua hal di atas

Pertanyaan Ulasan

1. Jelaskan arsitektur Hadoop.
2. Jelaskan semua fitur Hadoop HDFS.
3. Tuliskan komponen-komponen HDFS.

 Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education.

International Journal of Innovative Research in Computer and Communication Engineering, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

BAB 8: Analisis Data dengan R

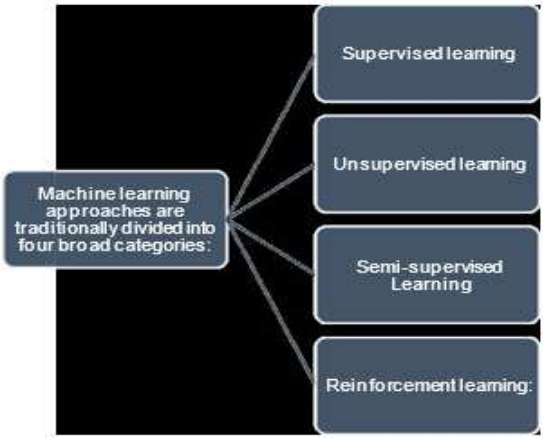
Dr. Ir. Indriyani, S.Kom., M.Kom

Tujuan

- pelajari konsep pembelajaran mesin.
- pelajari empat kategori pembelajaran mesin.

8.1 Metode Pembelajaran Mesin

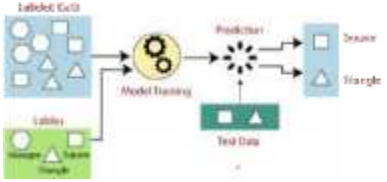
Pendekatan pembelajaran mesin secara tradisional dibagi menjadi empat kategori besar, tergantung pada sifat "sinyal" atau "umpan balik" yang tersedia untuk sistem pembelajaran:



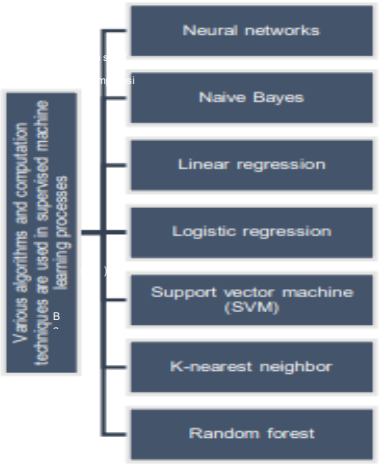
Gambar 8.1 *Metode Pembelajaran Mesin*

- **Pembelajaran yang diawasi:** Algoritme pembelajaran mesin yang diawasi dapat menggunakan contoh berlabel untuk menerapkan apa yang telah dipelajari di masa lalu ke data baru dan

memprediksi kejadian di masa mendatang. Algoritme pembelajaran membuat fungsi tersirat untuk menghasilkan prediksi tentang nilai keluaran berdasarkan pemeriksaan set data pelatihan yang diketahui. Setelah pelatihan yang cukup, sistem dapat menawarkan tujuan untuk setiap masukan baru. Algoritme pembelajaran juga dapat membandingkan keluarannya dengan keluaran yang tepat dan dimaksudkan serta mendeteksi kesalahan, yang memungkinkan model dimodifikasi sesuai kebutuhan.



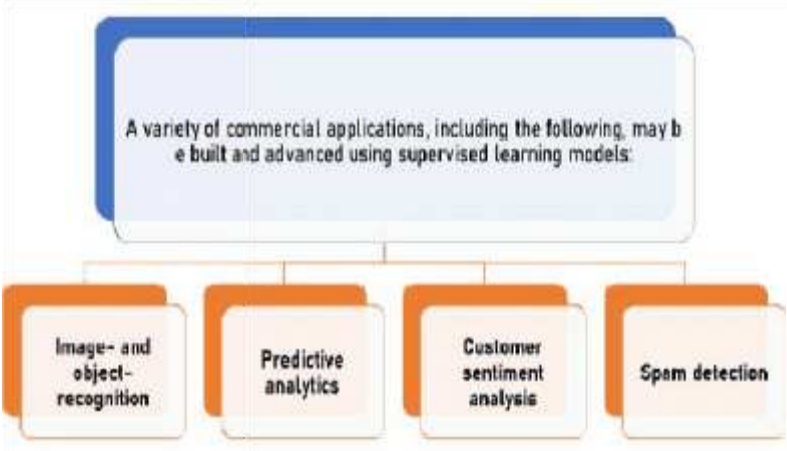
Gambar8.2 *Pembelajaran Terbimbing*



Gambar 8.3 Algoritma Pembelajaran Terbimbing

- ***Jaringan Saraf:***Jaringan saraf mencerminkan perilaku otak manusia, yang memungkinkan program komputer mengenali pola dan memecahkan masalah umum di bidang AI, pembelajaran mesin, dan pembelajaran mendalam.
- ***Bayes Naif:***Dalam statistik, pengklasifikasi Bayes naif adalah keluarga "pengklasifikasi probabilistik" sederhana yang didasarkan pada penerapan teorema Bayes dengan asumsi independensi yang kuat Analisis Data dengan R antara fitur-fitur. Mereka termasuk model jaringan Bayesian yang paling sederhana, tetapi jika digabungkan dengan estimasi kepadatan kernel, mereka dapat mencapai tingkat akurasi yang lebih tinggi
- ***Regresi Linier***: Analisis regresi linier digunakan untuk memprediksi nilai suatu variabel berdasarkan nilai variabel lain. Variabel yang ingin Anda prediksi disebut variabel dependen. Variabel yang Anda gunakan untuk memprediksi nilai variabel lain disebut variabel independen.
- ***Regresi Logistik:***Regresi logistik adalah metode analisis statistik yang digunakan untuk memprediksi nilai data berdasarkan pengamatan sebelumnya terhadap suatu set data. Model regresi logistik memprediksi variabel data dependen dengan menganalisis hubungan antara satu atau lebih variabel independen yang ada.
- ***Mesin Vektor Pendukung:***SVM (support vector machines) adalah teknik pembelajaran mesin terbimbing yang dapat digunakan untuk klasifikasi dan regresi. Namun, teknik ini paling sering digunakan dalam masalah kategorisasi. SVM awalnya dikembangkan pada tahun 1960-an, tetapi disempurnakan sekitar tahun 1990.

- ***KNN:***Metode pembelajaran mesin terawasi k-nearest neighbour (KNN) adalah teknik dasar dan mudah diimplementasikan yang dapat digunakan untuk mengatasi masalah klasifikasi dan regresi.
- ***Hutan Acak***: Hutan acak adalah pendekatan pembelajaran mesin untuk memecahkan masalah klasifikasi dan regresi. Pendekatan ini menggunakan pembelajaran ensemble, yaitu teknik untuk memecahkan masalah sulit dengan menggabungkan banyak pengklasifikasi. Metode hutan acak terdiri dari sejumlah besar pohon keputusan.



Gambar 8.4 *Contoh Pembelajaran Terbimbing*

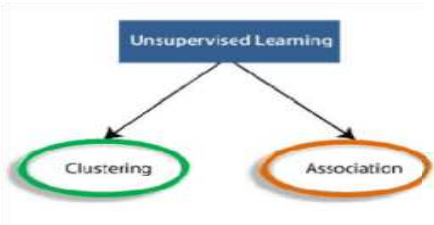
Pengenalan gambar dan objek: Ketika diterapkan pada berbagai teknik visi komputer dan analisis visual, algoritma pembelajaran terawasi dapat digunakan untuk menemukan, mengisolasi, dan mengkategorikan item dari film atau gambar, sehingga dapat digunakan. Analisis Prediktif: Pembuatan sistem analisis prediktif untuk memberikan wawasan mendalam ke dalam beberapa titik data bisnis merupakan kasus penggunaan umum untuk model pembelajaran terbimbing. Hal ini memungkinkan bisnis untuk memprediksi hasil tertentu tergantung pada

variabel keluaran tertentu, membantu eksekutif bisnis dalam membenarkan pilihan atau mengubah haluan untuk keuntungan organisasi. Analisis Sentimen Pelanggan: Organisasi dapat mengekstrak dan mengkategorikan sejumlah besar informasi dari sejumlah besar data menggunakan algoritma pembelajaran mesin yang diawasi dengan sangat sedikit interaksi manusia, termasuk konteks, emosi, dan tujuan. Ini mungkin cukup bermanfaat dalam hal memperoleh pengetahuan yang lebih baik tentang interaksi konsumen dan meningkatkan inisiatif keterlibatan merek. Deteksi Spam: Contoh lain dari model pembelajaran terbimbing adalah deteksi spam. Organisasi dapat melatih basis data untuk mengidentifikasi pola atau kelainan pada data baru menggunakan algoritma klasifikasi terbimbing, yang memungkinkan mereka mengatur korespondensi spam dan nonspam secara efisien.

8.2Tantangan Pembelajaran Terbimbing

Model pembelajaran yang diawasi mungkin memerlukan tingkat keahlian tertentu agar dapat disusun secara akurat. Pelatihan model pembelajaran terbimbing bisa memakan banyak waktu. Kumpulan data memiliki kemungkinan kesalahan manusia yang lebih tinggi, yang mengakibatkan algoritma belajar secara tidak benar. Tidak seperti model pembelajaran tanpa pengawasan, pembelajaran dengan pengawasan tidak dapat mengelompokkan atau mengklasifikasikan data secara sendirinya.

Pembelajaran Mesin Tanpa Pengawasan. Metode pembelajaran mesin tanpa pengawasan, di sisi lain, digunakan ketika data yang dilatih tidak dikategorikan atau diberi label. Pembelajaran tanpa pengawasan menyelidiki bagaimana komputer dapat menyimpulkan suatu fungsi dari data yang tidak diberi label untuk menggambarkan suatu struktur tersembunyi. Sistem tidak menentukan keluaran yang tepat, tetapi memeriksa data dan dapat menyimpulkan struktur tersembunyi dari data yang tidak berlabel menggunakan kumpulan data.



Gambar8.5 Pembelajaran Tanpa Pengawasan

- *Clustering:* Pengelompokan adalah cara mengatur item ke dalam kelompok sehingga item yang memiliki kesamaan terbanyak tetap berada dalam satu kelompok sementara item yang memiliki sedikit atau tidak memiliki kesamaan tetap berada di kelompok lain. Analisis kelompok mengidentifikasi kesamaan di antara item data dan mengklasifikasikannya menurut ada atau tidaknya kesamaan tersebut.
- *Aturan asosiasi:*Aturan asosiasi adalah pendekatan pembelajaran tanpa pengawasan yang digunakan untuk menemukan hubungan antara variabel dalam basis data besar. Aturan ini mengidentifikasi kelompok item yang muncul dalam kumpulan data secara bersamaan. Aturan asosiasi meningkatkan efektivitas strategi pemasaran. Orang yang membeli X (misalnya sepotong roti) cenderung membeli Y (mentega/selai). Analisis Keranjang Belanja Pasar adalah contoh aturan asosiasi yang baik.

Gambar8. 6 Algoritma pembelajaran tanpa pengawasan yang popular

Keuntungan Pembelajaran Tanpa Pengawasan

- Pembelajaran tanpa pengawasan digunakan untuk tugas yang lebih kompleks dibandingkan dengan pembelajaran dengan pengawasan karena, dalam pembelajaran tanpa pengawasan, kita tidak memiliki data masukan yang diberi label. Pembelajaran tanpa pengawasan lebih disukai karena lebih mudah untuk mendapatkan data yang tidak diberi label dibandingkan dengan data yang diberi label.

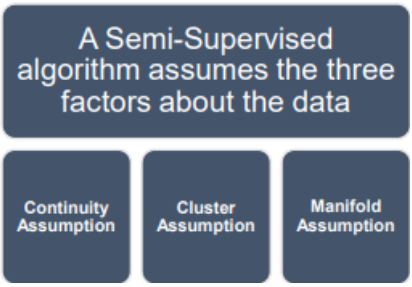
Kerugian Pembelajaran Tanpa Pengawasan

- Karena tidak memiliki output yang sebanding, pembelajaran tanpa pengawasan secara inheren lebih menantang daripada pembelajaran dengan pengawasan. Karena data masukan tidak diberi label dan algoritme tidak mengetahui output yang tepat sebelumnya, hasil dari metode pembelajaran tanpa pengawasan mungkin kurang akurat.

Algoritma pembelajaran mesin semi-supervised

- Karena menggunakan data berlabel dan tidak berlabel untuk pelatihan – umumnya sejumlah kecil data berlabel dan sejumlah besar data tidak berlabel – algoritme pembelajaran mesin semi-supervised berada di tengah-tengah antara pembelajaran tersupervised dan tidak tersupervised. Pendekatan ini dapat meningkatkan akurasi pembelajaran secara signifikan dalam sistem yang menggunakannya. Pembelajaran semi-supervised

sering digunakan ketika data berlabel yang dikumpulkan memerlukan penggunaan sumber daya yang kompeten dan tepat untuk melatih/mempelajarinya. Di sisi lain, memperoleh data tidak berlabel biasanya tidak memerlukan sumber daya tambahan.



Gambar 8.7 Tiga faktor diasumsikan dalam algoritma pembelajaran semi-supervised

Algoritma Semi-Supervised mengasumsikan tiga faktor tentang data seperti yang ditunjukkan pada Gambar 7. Asumsi Kontinuitas: Metode ini meyakini bahwa titik-titik yang lebih berdekatan memiliki probabilitas lebih tinggi untuk memiliki label keluaran yang sama.

- Asumsi Klaster**:** Data dapat dipecah menjadi beberapa klaster berbeda, dengan titik-titik dalam klaster yang sama memiliki peluang lebih tinggi untuk memiliki label keluaran yang sama.
- Asumsi Manifold: Data tersebut didistribusikan secara kasar pada manifold dengan ukuran yang jauh lebih kecil daripada ruang input. Asumsi ini memungkinkan jarak dan kepadatan yang ditetapkan pada manifold untuk digunakan.
- Aplikasi Praktis Pembelajaran Semi-Supervised. Analisis Ucapan: Karena mengkategorikan rekaman audio merupakan pekerjaan yang memakan waktu, pembelajaran semi-terawasi merupakan solusi yang jelas.

- Klasifikasi Konten Internet: Karena memberi label pada setiap halaman web sulit dan tidak mungkin, teknik pembelajaran semisupervised digunakan. Bahkan algoritme pencarian Google memberi peringkat relevansi halaman web untuk kueri tertentu menggunakan bentuk pembelajaran Semi-Supervised.
- Klasifikasi Urutan Protein: Karena untaian DNA umumnya agak panjang, munculnya pembelajaran semi-terawasi di sektor ini telah diprediksi.

8.3 Algoritma Pembelajaran Mesin Penguatan

Pembelajaran Mesin mencakup bidang pembelajaran penguatan. Ini semua tentang mengambil langkah yang tepat untuk memaksimalkan manfaat Anda dalam keadaan tertentu. Ini digunakan oleh berbagai perangkat lunak dan komputer untuk Pengenalan Big Data menentukan tindakan atau jalur terbaik yang layak dalam skenario tertentu. Pembelajaran penguatan berbeda dari pembelajaran terbimbing karena kunci solusi disertakan dalam data pelatihan, yang memungkinkan model dilatih dengan jawaban yang benar, tetapi dalam pembelajaran penguatan, tidak ada jawaban dan agen penguatan menentukan apa yang harus dilakukan untuk menyelesaikan pekerjaan. Ia berkewajiban untuk belajar dari pengalamannya tanpa adanya kumpulan data pelatihan.

Berikut adalah beberapa istilah penting yang digunakan dalam Penguatan.

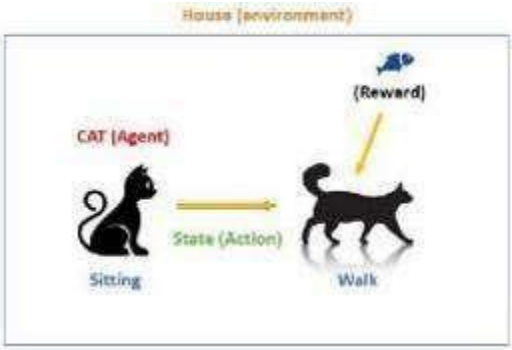
- Agen: Ia merupakan suatu entitas yang diasumsikan melakukan tindakan dalam suatu lingkungan untuk memperoleh imbalan tertentu.
- Lingkungan (e):Skenario yang harus dihadapi seorang agen.
- Hadiah (R):Pengembalian langsung yang diberikan kepada agen saat dia melakukan tindakan atau tugas tertentu.
- State mengacu pada situasi terkini yang dikembalikan oleh lingkungan. Kebijakan (π): Ini adalah strategi yang diterapkan oleh

agen untuk memutuskan tindakan selanjutnya berdasarkan keadaan terkini. Nilai (V): Ini adalah pengembalian jangka panjang yang diharapkan dengan diskon, dibandingkan dengan imbalan jangka pendek.

- Fungsi Nilai: Ini menentukan nilai dari suatu status yang merupakan jumlah total hadiah. Ini adalah agen yang diharapkan mulai dari status tersebut.
- Model lingkungan:Ini meniru perilaku lingkungan. Ini membantu Anda membuat kesimpulan yang harus dibuat dan juga menentukan bagaimana lingkungan akan berperilaku.
- Metode berbasis model: Ini adalah metode untuk memecahkan masalah pembelajaran penguatan yang menggunakan metode berbasis model. Nilai Q atau nilai tindakan (Q): Nilai Q cukup mirip dengan nilai. Satu-satunya perbedaan antara keduanya adalah bahwa ia mengambil parameter tambahan sebagai tindakan saat ini.

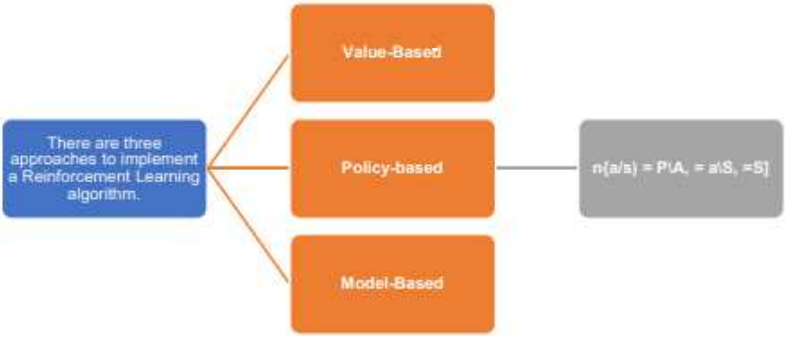
Bagaimana pembelajaran penguatan bekerja?

Dalam hal ini, kucing Anda adalah organisme hidup yang terpapar ke dunia luar. Dalam situasi ini, kucing Anda adalah rumah Anda. Kucing Anda mungkin sedang duduk, dan Anda menggunakan frasa tertentu dalam for cat to walk sebagai contoh kondisi. Agen kami merespons dengan melakukan transisi tindakan dari satu "kondisi" ke kondisi berikutnya.



Gambar8.8 Contoh Pembelajaran Penguatan

Agen kami merespons dengan melakukan transisi tindakan dari satu "keadaan" ke keadaan berikutnya. Misalnya, kucing Anda berubah dari duduk menjadi berjalan. Reaksi agen adalah tindakan, dan kebijakan adalah cara memilih tindakan dalam suatu situasi dengan harapan memperoleh hasil yang lebih baik. Mereka mungkin menerima hadiah atau hukuman sebagai hasil dari transfer tersebut.

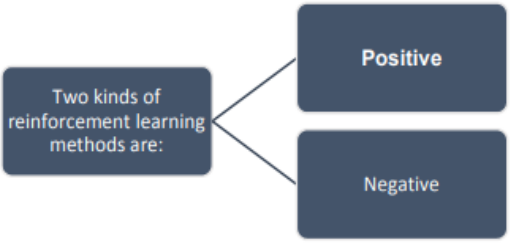


Gambar 8.9 Tiga Pendekatan untuk menerapkan algoritma pembelajaran penguatan

- *Berbasis Nilai*: Anda harus bertujuan untuk mengoptimalkan fungsi nilai V dalam berbasis nilai Teknik Pembelajaran Penguatan. Dalam teknik ini, agen mengantisipasi pengembalian jangka panjang dari kondisi kebijakan yang ada.
- *Berbasis kebijakan:* Dalam teknik RL berbasis kebijakan, Anda bertujuan untuk membuat kebijakan yang memungkinkan Anda menerima imbalan paling besar di masa mendatang dengan melakukan tindakan di setiap status. Ada dua jenis metode berbasis kebijakan:
- Deterministik: Kebijakan tersebut menghasilkan tindakan yang sama untuk negara mana pun.
- Stokastik: Setiap tindakan memiliki probabilitas, yang dapat dihitung menggunakan persamaan di bawah ini.
- Kebijakan Stokastisitas: Setiap tindakan memiliki probabilitas.
- *Berbasis model:* Anda harus membuat model virtual untuk setiap lingkungan dalam teknik Reinforcement Learning ini. Agen belajar cara bekerja dalam lingkungan tertentu.

8.4 Karakteristik Pembelajaran Penguatan

Tidak ada pengawas, hanya angka atau sinyal penghargaan. Pengambilan keputusan dilakukan secara berurutan. Dalam masalah Penguatan, waktu sangat penting. Umpan balik tidak pernah langsung; selalu tertunda. Data yang diterima agen ditentukan oleh aktivitasnya.

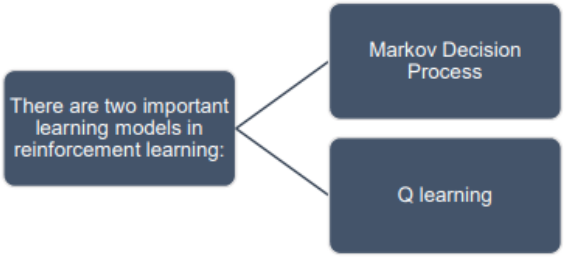


Gambar 8.10 Jenis Pembelajaran Penguatan

- **Positif:** Hal ini digambarkan sebagai suatu kejadian yang terjadi sebagai akibat dari tindakan tertentu. Hal ini meningkatkan kekuatan dan frekuensi perilaku dan memiliki pengaruh yang menguntungkan pada tindakan agen. Penguatan semacam ini membantu dalam memaksimalkan kinerja dan Pengenalan Big Data mempertahankan perubahan untuk jangka waktu yang lebih lama. Namun, terlalu banyak penguatan dapat menyebabkan pengoptimalan keadaan yang berlebihan, yang dapat berdampak pada hasil.
- **Negatif:**Penguatan Buruk didefinisikan sebagai penguatan perilaku yang terjadi sebagai akibat dari keadaan negatif yang seharusnya dihindari atau dihentikan. Penguatan ini membantu Anda menentukan tingkat kinerja minimal. Kerugian dari teknik ini adalah penguatan ini hanya memberikan cukup penguatan untuk memenuhi persyaratan perilaku minimal.

8.5 Model Pembelajaran Penguatan

`Ada dua model pembelajaran penting dalam pembelajaran penguatan:

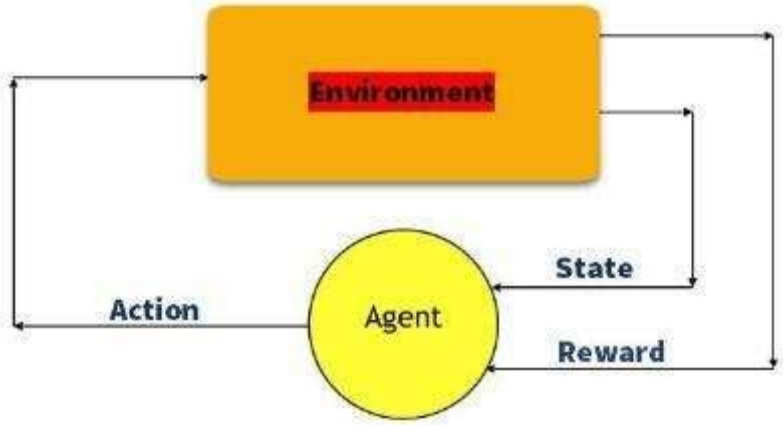


Gambar 8.11 Model pembelajaran penting dalam pembelajaran penguatan

Proses Keputusan Markov

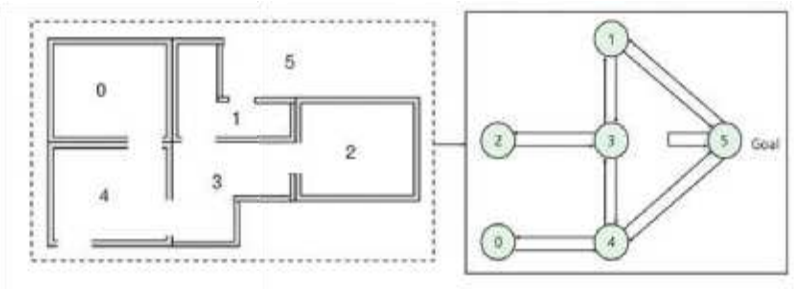
Parameter berikut digunakan untuk mendapatkan solusi

Serangkaian tindakan- A, Kumpulan keadaan –S, Imbalan – R, Kebijakan – n, Nilai – V. Pendekatan matematika untuk memetakan solusi dalam Pembelajaran penguatan direkonstruksi sebagai Proses Keputusan Markov atau (MDP) seperti yang ditunjukkan pada Gambar 12.



Gambar8. 12 Set Tindakan

Pembelajaran Q adalah pendekatan berbasis nilai yang memberikan informasi untuk membantu agen memutuskan tindakan mana yang harus dilakukan. Mari kita lihat contoh untuk lebih memahami metode ini: Di sebuah gedung, ada lima ruangan yang dihubungkan oleh pintu. Setiap ruangan diberi nomor dari 0 hingga 4. Bagian luar gedung mungkin berupa satu ruang luar yang besar (5) Dari ruangan 5, pintu 1 dan 4 mengarah ke dalam gedung.



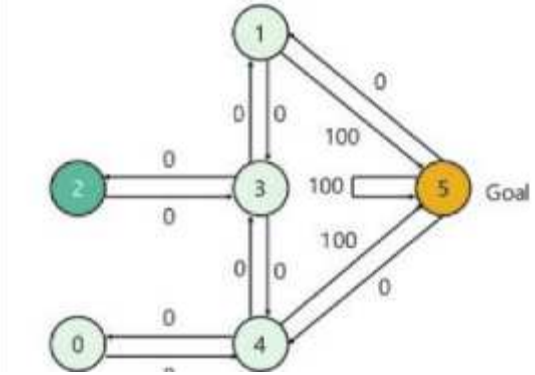
Gambar 8.13 Q Learning

Setelah itu, Anda harus menetapkan nilai hadiah untuk setiap pintu: 100 poin diberikan untuk pintu yang terhubung langsung ke tujuan. Tidak ada hadiah untuk pintu yang tidak terhubung langsung ke ruangan target. Karena pintu bersifat dua arah dan setiap ruangan memiliki dua anak panah, masing-masing anak panah pada gambar di atas mewakili nilai hadiah langsung.

Tabel 1 Penguatan vs Pembelajaran Terbimbing

Parameter	**Pembelajaran Penguatan**	**Pembelajaran yang Diawasi**
Gaya pengambilan keputusan	Pembelajaran penguatan memungkinkan Anda membuat penghakiman secara berurutan.	Masukan yang diberikan di awal digunakan untuk membuat pilihan dalam prosedur ini.
Bekerja pada	Berinteraksi dengan lingkungan sekitar adalah prioritas.	Bekerja dengan contoh atau data yang disediakan sebagai

		sampel.
Ketergantungan pada keputusan	Pilihan pembelajaran dalam teknik RL bersifat dependen. Akibatnya, semua keputusan dependen harus diberi label.	Pembelajaran yang diawasi atas penilaian yang tidak berhubungan satu sama lain, dengan label yang ditetapkan untuk setiap keputusan.
Paling cocok	Mendukung dan bekerja lebih baik di AI jika ada banyak kontak manusia.	Biasanya dikendalikan oleh sistem perangkat lunak atau aplikasi yang bersifat interaktif.
Contoh	Permainan catur	Pengenalan objek



Gambar 14 Q learning Menetapkan nilai hadiah untuk setiap pintu

Dalam gambar ini, Anda dapat melihat bahwa ruangan tersebut mewakili sebuah status. Pergerakan agen dari satu ruangan ke ruangan lain mewakili sebuah tindakan. Dalam gambar yang diberikan di bawah ini, sebuah status digambarkan sebagai sebuah simpul, sedangkan anak panah menunjukkan tindakan. Sebagai contoh, seorang agen bergerak dari ruangan nomor 2 ke 5 Keadaan awal = keadaan 2, Keadaan 2-> keadaan 3, Keadaan 3 -> keadaan (2,1,4), Keadaan 4-> keadaan (0,5,3)

Keadaan 1-> keadaan (5,3), Keadaan 0-> keadaan 4

Aplikasi Pembelajaran Penguatan

Gambar 15 Robotika untuk otomasi industri

Gambar 16 Perencanaan Strategi Bisnis

Ringkasan

Pembelajaran mesin (ML) adalah studi tentang algoritma komputer yang dapat meningkatkan kemampuannya seiring berjalannya waktu dengan memperoleh pengalaman dan menggunakan data. Algoritma pembelajaran mesin membuat model berdasarkan data pelatihan untuk membuat prediksi atau penilaian tanpa harus diprogram secara eksplisit untuk melakukannya. Proses penyediaan data input serta data output yang tepat ke model pembelajaran mesin dikenal sebagai pembelajaran terbimbing. Tujuan algoritma pembelajaran terbimbing adalah menemukan fungsi pemetaan yang akan menerjemahkan variabel input (x) ke variabel output (y) (y). Pembelajaran tanpa pengawasan, yang juga dikenal sebagai pembelajaran mesin tanpa pengawasan, menganalisis dan mengelompokkan informasi yang tidak berlabel menggunakan teknik pembelajaran mesin. Tanpa memerlukan interaksi manusia, algoritme ini mengungkap pola atau pengelompokan data yang tersembunyi.

Masalah pembelajaran dengan sejumlah kecil contoh berlabel dan sejumlah besar contoh tak berlabel dikenal sebagai pembelajaran semi-supervised. Pembelajaran penguatan (RL) adalah cabang pembelajaran mesin yang mempelajari bagaimana agen cerdas harus beroperasi dalam lingkungan tertentu untuk memaksimalkan konsep imbalan kumulatif. Pembelajaran penguatan, bersama dengan pembelajaran terbimbing dan tak terbimbing, adalah salah satu dari tiga paradigma pembelajaran mesin utama. Dalam statistik, pengklasifikasi Bayes naif merupakan bagian dari "pengklasifikasi probabilistik" yang didasarkan pada teorema Bayes dan asumsi independensi yang kuat antar fitur. Pengklasifikasi ini merupakan salah satu model jaringan Bayesian yang paling dasar, tetapi jika digabungkan dengan estimasi kepadatan kernel, pengklasifikasi ini dapat mencapai tingkat akurasi yang lebih tinggi. Dalam statistik, pengklasifikasi Bayes naif merupakan bagian dari "pengklasifikasi probabilistik" yang

didasarkan pada teorema Bayes dan asumsi independensi yang kuat antar fitur. Pengklasifikasi ini merupakan salah satu model jaringan Bayesian yang paling dasar, tetapi jika digabungkan dengan estimasi kepadatan kernel, pengklasifikasi ini dapat mencapai tingkat akurasi yang lebih tinggi. Analisis sentimen adalah identifikasi, ekstraksi, kuantifikasi, dan studi sistematis tentang keadaan emosional dan informasi subjektif menggunakan pemrosesan bahasa alami, analisis teks, linguistik komputasional, dan biometrik. Pengelompokan adalah proses pemisahan populasi atau kumpulan titik data ke dalam banyak kelompok sehingga titik data dalam kelompok yang sama lebih mirip daripada titik data dalam kelompok lain. Dengan kata lain, tujuannya adalah untuk memisahkan kelompok dengan karakteristik serupa dan menempatkannya ke dalam kelompok. Dalam psikologi, asosiasi merujuk pada hubungan mental yang terbentuk oleh pengalaman-pengalaman tertentu antara konsep, peristiwa, atau kondisi mental. Behaviorisme, asosiasionisme, psikoanalisis, psikologi sosial, dan strukturalisme adalah aliran-aliran pemikiran dalam psikologi yang menggunakan asosiasi.

Soal Latihan

1. Algoritma pembelajaran mesin yang diawasi dapat menggunakan _____- contoh untuk menerapkan apa yang telah mereka pelajari di masa lalu ke data baru dan memprediksi kejadian di masa mendatang.

A. Berlabel
B. Tidak berlabel
C. Diprediksi
D. Tak terduga

2.Manakah dari berikut ini yang merupakan contoh pembelajaran mesin?

A. Jaringan saraf
B. Pendekatan Bayes Naif
C. Regresi Linier
D. Semua hal di atas

3. Manakah dari pilihan berikut yang merupakan jaringan _____ yang mencerminkan perilaku otak manusia, yang memungkinkan komputer mengenali pola dan memecahkan masalah umum.

A. Jaringan saraf
B. Pendekatan Bayes Naif
C. Regresi Linier
D. Semua hal di atas

4. Pengklasifikasi Naive Bayes merupakan keluarga _____ sederhana.

A. pengklasifikasi nonprobabilistik
B. pusat probabilistik
C. pengklasifikasi probabilistik
D. Tidak ada yang di atas

5. Pembelajaran mesin adalah aplikasi _____

A. Kecerdasan Buatan
B. Rantai Blok
C. Baik a dan b

D. Tidak ada yang di atas

6. Teknik manakah yang digunakan dalam sistem rekomendasi?

A. Penyaringan berbasis konten

B. Penyaringan kolaboratif

C. Keduanya

D. Tidak ada yang di atas

7.Jika kita mempertimbangkan fitur untuk memahami selera pengguna, ini adalah contoh _____

A. Penyaringan berbasis konten

B. Penyaringan kolaboratif

C. Keduanya

D. Tidak ada yang di atas

8. Pada pilihan berikut manakah prediksi otomatis dilakukan untuk pengguna.

A. Penyaringan berbasis konten

B. Penyaringan kolaboratif

C. Keduanya

D. Tidak ada yang di atas

9. Penyaringan kolaboratif adalah _____

A. Pembelajaran yang diawasi

B. Pembelajaran tanpa pengawasan

C. Keduanya

D. Tidak ada yang di atas

10._____ menggunakan fitur item untuk merekomendasikan item lain yang serupa dengan apa yang disukai pengguna, berdasarkan tindakan sebelumnya atau umpan balik yang jelas.

A. Penyaringan berbasis konten

B. Penyaringan kolaboratif

C. Keduanya

D. Tidak ada yang di atas

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus*

Teknik. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

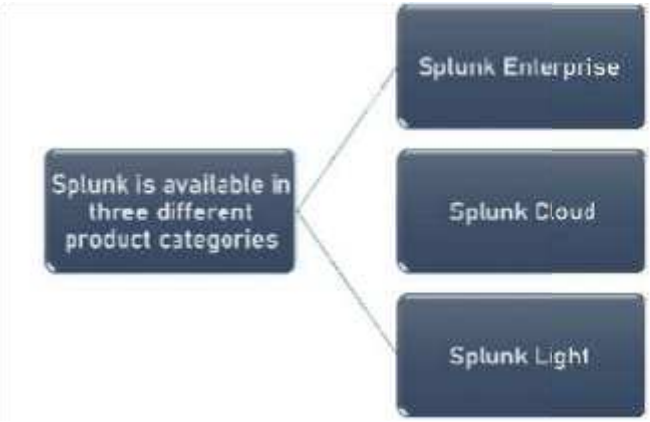
Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

BAB 9: Manajemen Big Data menggunakan Splunk

Ir. Anwar T. Sitorus, M.Kom

9.1 Kategori Produk

Splunk tersedia dalam tiga kategori produk berbeda sebagai berikut −



Gambar 9.1 Tiga kategori produk yang berbeda

- Splunk enterprise**.** Digunakan oleh organisasi dengan infrastruktur TI yang signifikan dan perusahaan yang sangat bergantung pada teknologi. Ini membantu dalam pengumpulan dan analisis data dari situs web, aplikasi, perangkat, dan sensor, di antara sumber lainnya.
- *Splunk Cloud* Ini adalah platform yang dihosting di cloud dengan fungsi yang sama dengan edisi korporat. Tersedia langsung dari Splunk atau melalui platform cloud AWS.
- *Splunk Light*− Memungkinkan pengguna untuk mencari, melaporkan, dan mendapatkan peringatan pada semua data log

secara real time dari satu lokasi. Dibandingkan dengan dua versi lainnya, versi ini menawarkan lebih sedikit kemampuan dan fitur.

- Fitur SPLUNK

Fitur SPLUNK ditunjukkan pada Gambar9. 2.

Gambar9.2 Fitur SPLUNK

- *Penyerapan Data*: Splunk menerima berbagai jenis data, termasuk JSON, XML, dan data mesin tak terstruktur seperti log web dan aplikasi. Pengguna dapat memodelkan data tak terstruktur menjadi struktur data sesuai keinginan.
- *Pengindeksan Data*: Splunk mengindeks data yang diimpor untuk pencarian dan kueri yang lebih cepat dalam berbagai situasi.
- *Pencarian Data:* Di Splunk, pencarian memerlukan pemanfaatan data terindeks untuk membuat metrik, memperkirakan tren masa depan, dan menemukan pola.
- *Menggunakan Peringatan:* Ketika kriteria tertentu diidentifikasi dalam data yang diperiksa, peringatan Splunk dapat digunakan untuk mengirim email atau umpan RSS.
- *Dasbor:* Dasbor Splunk dapat menampilkan hasil pencarian sebagai bagan, laporan, dan tabel pivot, antara lain.
- *Model Data:* Berdasarkan pengetahuan domain khusus, data yang diindeks dapat dimodelkan menjadi satu atau beberapa set data. Hal ini memudahkan pengguna akhir untuk menavigasi dan

mengevaluasi kasus bisnis tanpa harus memahami seluk-beluk bahasa pemrosesan pencarian Splunk.

9.2 Antarmuka SPLUNK

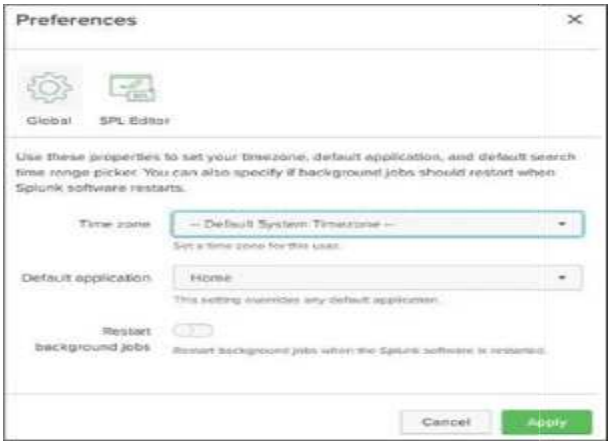
Antarmuka web Splunk mencakup semua alat yang Anda perlukan untuk mencari, melaporkan, dan menganalisis data yang telah Anda masukkan. Antarmuka web yang sama memungkinkan administrator untuk mengelola pengguna dan tanggung jawab mereka. Antarmuka ini juga mencakup koneksi untuk pengambilan data serta aplikasi bawaan Splunk. Cuplikan layar di bawah ini menampilkan layar pertama yang Anda lihat saat masuk ke Splunk dengan kredensial admin Anda.

Tautan Administrator

Menu drop down Administrator memungkinkan Anda untuk menyesuaikan dan mengubah informasi administrator. Dengan menggunakan antarmuka di bawah ini, kami dapat mengatur ulang ID email dan kata sandi admin.

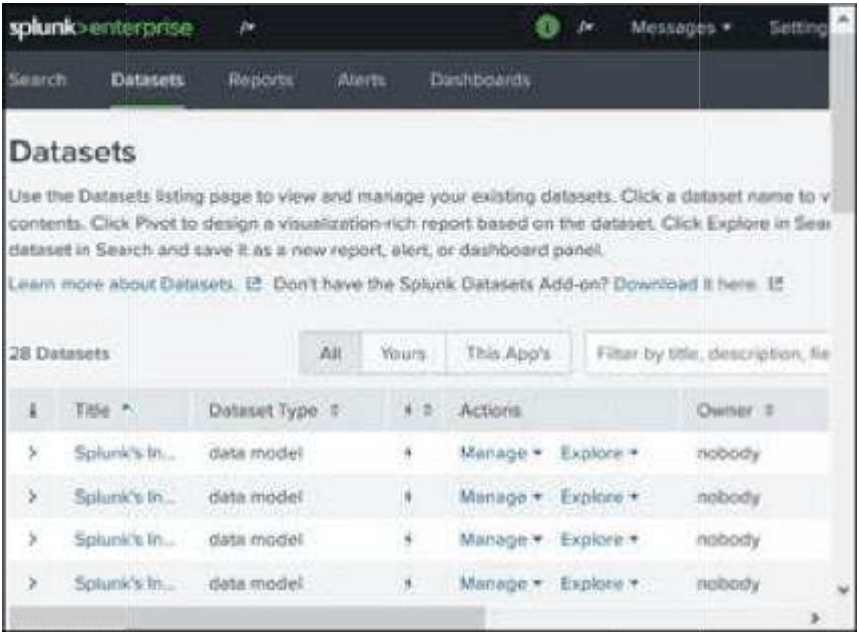
Gambar 9.3 Tautan Administrator

Kita juga dapat membuka opsi preferensi dari tautan administrator untuk memilih zona waktu dan aplikasi beranda tempat halaman arahan akan terbuka setelah Anda masuk. Halaman arahan sekarang muncul di beranda, seperti yang ditunjukkan di bawah ini pada Gambar 4.



Gambar 9.4 :Tampilan angka

Tautan Pengaturan Ini adalah tautan ke halaman yang mencantumkan semua fungsi utama Splunk. Dengan memilih tautan pencarian, Anda dapat menambahkan file pencarian dan definisi pencarian, misalnya Tautan Pencarian dan Pelaporan: Tautan ke pencarian dan pelaporan membawa kita ke fitur tempat kita dapat menemukan kumpulan data yang dapat diakses untuk mencari laporan dan peringatan yang telah dibuat untuk pencarian ini. Cuplikan layar di bawah ini dengan jelas menunjukkan hal ini. −



Fungsi Tambah Data Splunk, yang merupakan bagian dari antarmuka pencarian dan pelaporan, adalah tempat data diserap.

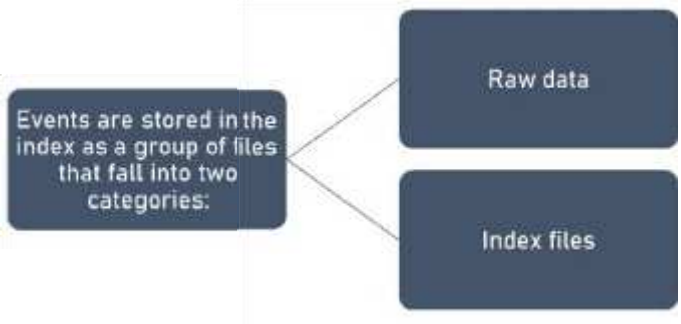
9.3 Penyerapan Data

Penyerapan data di Splunk dilakukan melalui fitur Tambah Data yang merupakan bagian dari aplikasi pencarian dan pelaporan. Setelah masuk, layar beranda antarmuka Splunk menampilkan ikon Tambah Data.Tentang mengunggah data Saat Anda menambahkan data ke penerapan Splunk Anda, data tersebut diproses dan diubah menjadi serangkaian peristiwa individual yang dapat Anda lihat, cari, dan analisis. Jenis data apa?

Di mana data disimpan?

Proses transformasi data disebut pengindeksan. Selama pengindeksan, data yang masuk diproses untuk memungkinkan pencarian

dan analisis yang cepat. Hasil yang diproses disimpan dalam indeks sebagai peristiwa. Indeks adalah tempat penyimpanan data dalam bentuk berkas datar. Indeks berada di komputer tempat Anda mengakses penyebaran Splunk.



Gambar9.6 *Peristiwa disimpan dalam indeks sebagai sekelompok file yang terbagi dalam dua kategori*

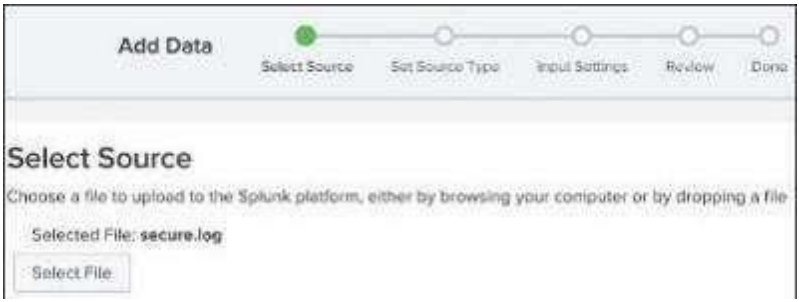
Gunakan Wizard ADD Data Fungsi Add Data Splunk, yang merupakan bagian dari antarmuka pencarian dan pelaporan, adalah tempat data diserap. Ikon Add Data muncul di layar beranda antarmuka Splunk setelah masuk, seperti yang diilustrasikan di bawah ini.

***Gambar 9.7** Wizard Tambah Data*

Saat kita mengklik opsi ini, kita akan diarahkan ke layar tempat kita dapat memilih sumber dan format data yang ingin kita kirim ke Splunk untuk dianalisis. Mengumpulkan Data. Data untuk analisis dapat diperoleh dari situs web resmi Splunk. Simpan berkas ini ke cakram lokal Anda dan ekstrak. Saat Anda mengakses folder tersebut, Anda akan melihat tiga berkas dalam berbagai format. Ketiga berkas tersebut adalah berkas log yang dibuat oleh beberapa aplikasi daring. Kita juga bisa mendapatkan kumpulan data lain dari Splunk, yang dapat ditemukan di situs web resmi Splunk.

9.4 Mengunggah Data

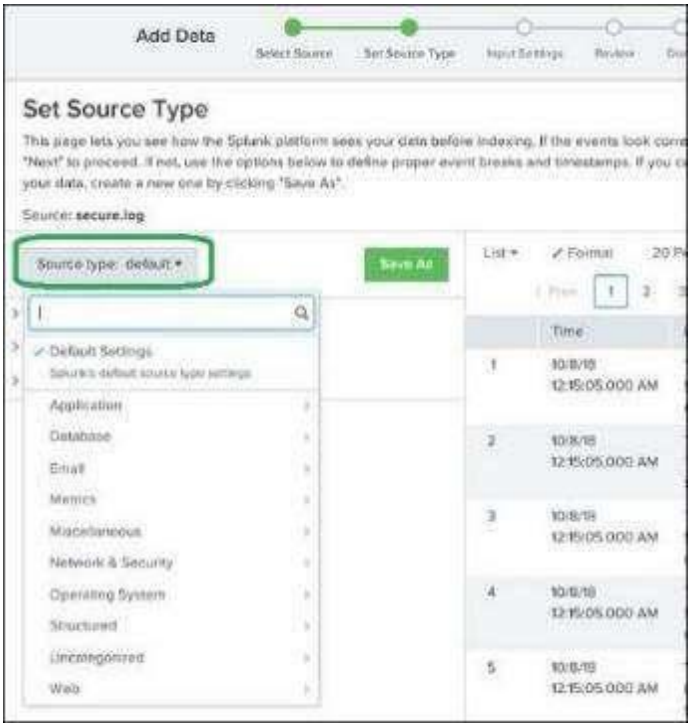
Kemudian, seperti yang dijelaskan pada paragraf sebelumnya, kita pilih file secure.log dari folder mailsv di komputer lokal kita. Dengan menggunakan tombol next berwarna hijau di pojok kanan atas, kita lanjut ke tahap berikutnya setelah memilih file.



***Gambar9.8** Mengunggah Data*

Memilih Jenis Sumber Splunk menyertakan fungsi bawaan yang mendeteksi tipe data yang sedang diserap. Fungsi ini juga memungkinkan pengguna untuk memilih tipe data selain yang disarankan oleh Splunk. Saat kita memilih tipe sumber dari menu tarik-turun, kita dapat melihat daftar jenis data yang dapat diserap dan dicari oleh Splunk.

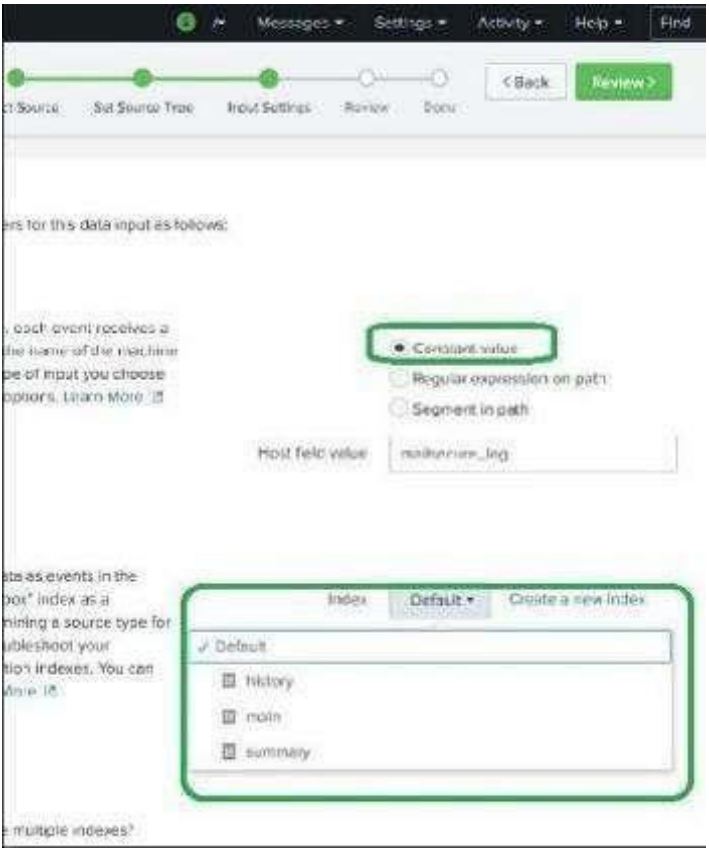
Dalam contoh berikut, kita akan menggunakan jenis sumber defaultGambar 9.



Gambar 9.9 *Mengatur Jenis Sumber*

Pengaturan Input Kami mengonfigurasi nama host tempat data diimpor dalam fase proses penyerapan data ini. Untuk nama host, ada beberapa kemungkinan yang dapat dipilih seperti yang ditunjukkan pada Gambar 10. Nilai konstan Ini adalah nama host lengkap dari server tempat data sumber disimpan. regex pada jalur Saat menggunakan ekspresi reguler untuk mendapatkan nama host. Kemudian, di area Ekspresi reguler, ketik regex untuk host yang ingin Anda ekstrak. segmen di jalur Masukkan nomor segmen di kotak Nomor segmen untuk mengekstrak nama host dari segmen di rute sumber data Anda. Misalnya, jika jalur sumber adalah

/var/log/ dan Anda ingin nilai host menjadi segmen ketiga (nama server host), masukkan "3." Langkah selanjutnya adalah memilih jenis indeks yang akan digunakan untuk mencari data masukan. Pendekatan indeks default dipilih. Indeks ringkasan digunakan untuk membuat ringkasan data dan membuat indeks berdasarkan data tersebut, sedangkan indeks riwayat digunakan untuk menyimpan riwayat pencarian. Pada gambar di bawah, indeks tersebut terwakili dengan jelas.

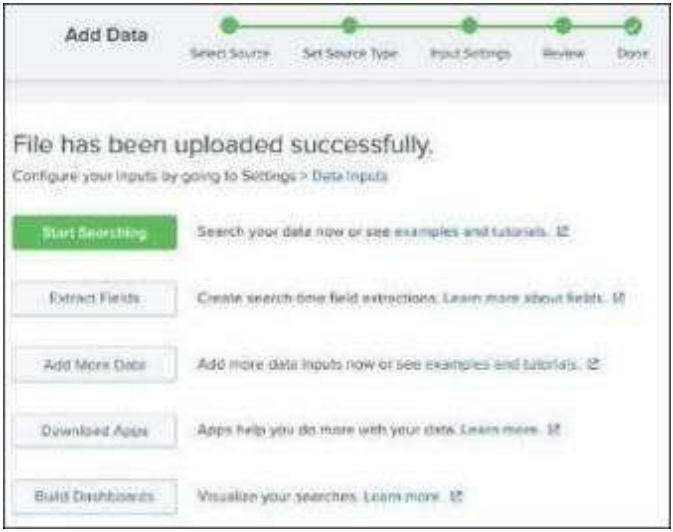


Gambar9 10 *Pengaturan Input*

Tinjau Pengaturan Setelah mengklik tombol berikutnya, kita akan melihat ringkasan pengaturan yang telah kita pilih. Kita tinjau dan pilih Berikutnya untuk menyelesaikan pengunggahan data seperti yang ditunjukkan pada Gambar 11.

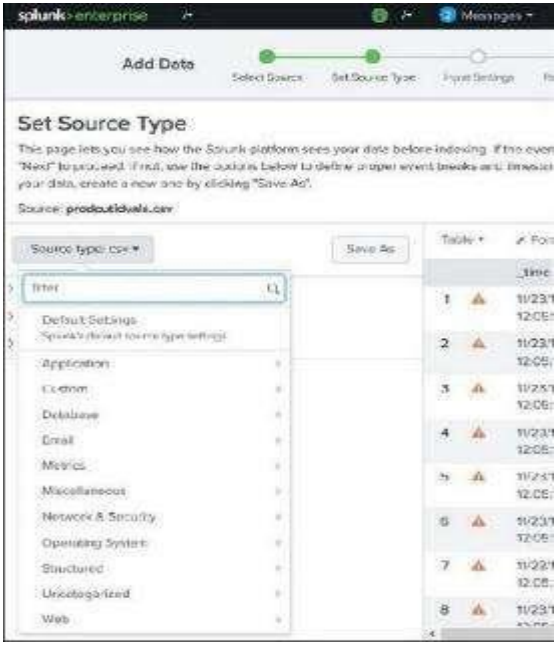
Gambar 9.11Tinjau Pengaturan

Ketika pemuatan selesai, layar di bawah akan terbuka, yang menunjukkan bahwa data berhasil diserap dan menjelaskan langkah selanjutnya yang dapat kita lakukan terhadap data tersebut.



Gambar 9.12 Data Berhasil Dimasukkan

Unit pemrosesan data bawaan Splunk mengevaluasi semua data yang masuk dan mengklasifikasikannya ke dalam beberapa jenis dan kategori data. Splunk, misalnya, dapat membedakan log dari server web Apache dan membuat bidang yang sesuai dari data yang dibaca. Kemampuan identifikasi jenis sumber Splunk melakukan ini dengan memanfaatkan jenis sumber bawaannya, terkadang dikenal sebagai jenis sumber "terlatih". Pengguna tidak perlu mengklasifikasikan data secara manual atau menetapkan tipe data apa pun ke bidang data yang masuk, sehingga memudahkan analisis.Jenis Sumber yang Didukung Mengunggah file menggunakan fungsi Add Data dan kemudian memilih Source Type dari menu akan menampilkan tipe sumber yang didukung di Splunk. Kami telah mengunggah file CSV dan kemudian mencentang semua pilihan yang memungkinkan pada Gambar 9.13 di bawah.



Gambar 13 Fitur Tambah Data

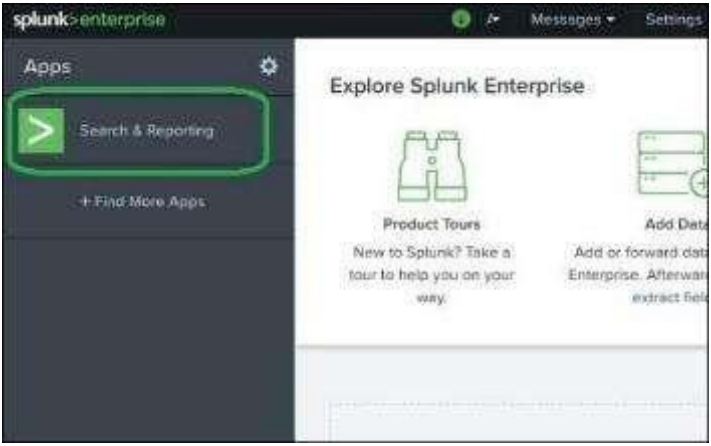
Jenis Sumber Sub-Kategori Bahkan di dalam kategori tersebut, kita dapat mengeklik untuk melihat semua subkategori yang didukung. Saat Anda memilih kategori basis data, Anda akan dapat melihat berbagai jenis basis data serta berkas yang didukung yang dapat dideteksi oleh Splunk.

Gambar 9.14 *Jenis Sumber Sub-Kategori*

Jenis Sumber yang Telah Dilatih Sebelumnya Beberapa jenis sumber pra-latihan terpenting disertakan dalam tabel di bawah ini. Splunk menyadari hal ini.

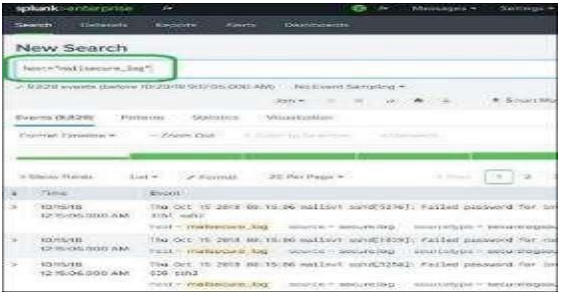
9.5 Aplikasi Pencarian & Pelaporan

Splunk menyediakan fitur pencarian canggih yang memungkinkan Anda mencari seluruh kumpulan data yang telah diserap. Fungsionalitas ini dapat digunakan melalui aplikasi Search & Reporting, yang dapat ditemukan di bilah sisi kiri setelah masuk ke situs daring.



Gambar 9.15 Pencarian dan Pelaporan

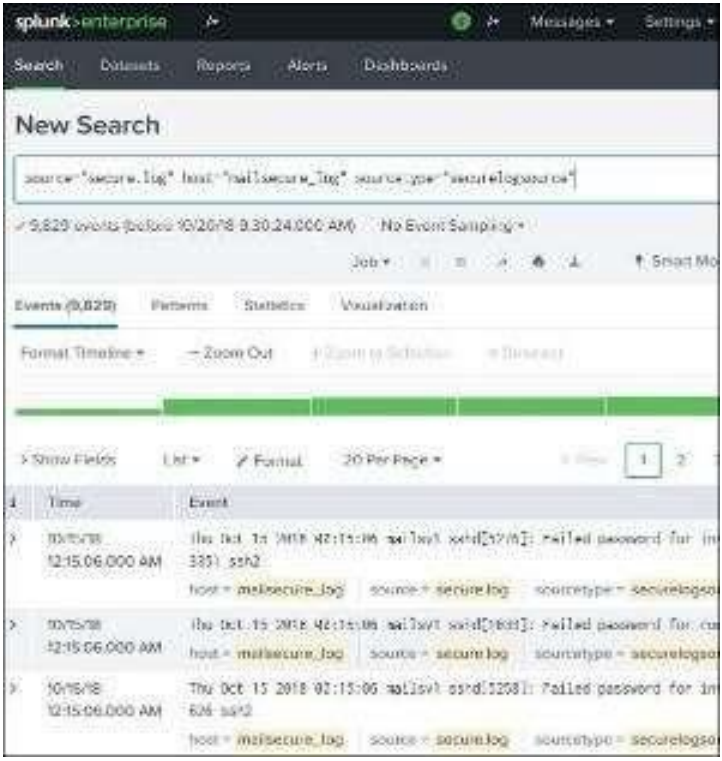
Saat kita memilih aplikasi Pencarian & Pelaporan, kita akan disambut dengan kotak pencarian tempat kita dapat memulai pencarian data logGambar 16. Kita memasukkan nama host dalam format yang ditunjukkan di bawah ini lalu mengklik ikon pencarian di sudut kanan atas. Ini akan menampilkan hasil yang menyorot kata pencarian.



Gambar 16 Kotak pencarian di aplikasi Pencarian dan Pelaporan

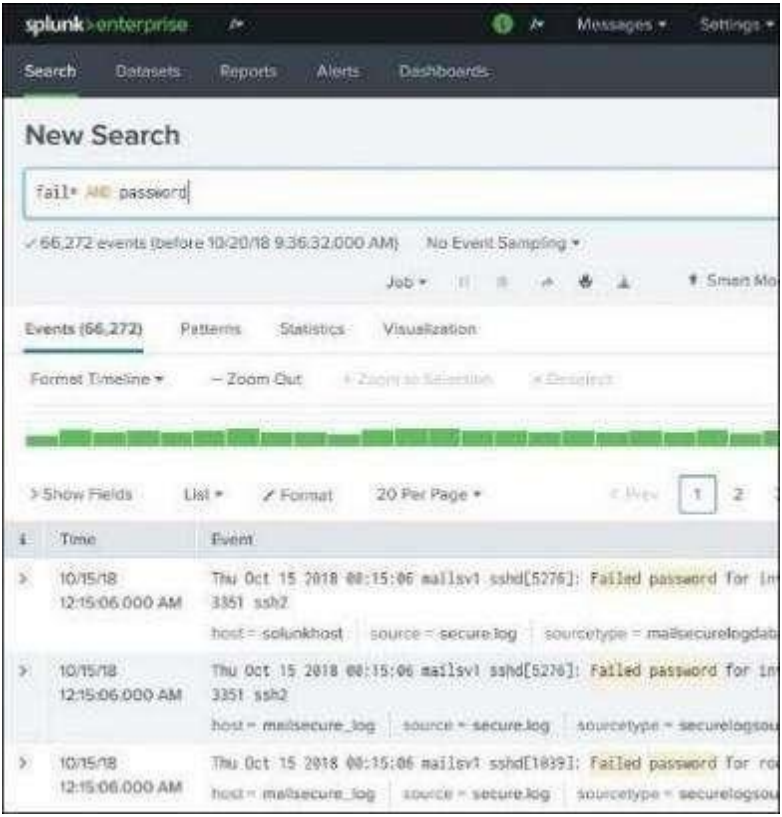
Menggabungkan Istilah Pencarian Dengan menuliskan frasa satu demi satu, sambil menyertakan string pencarian pengguna dalam tanda kutip

ganda, kita dapat menggabungkan istilah yang digunakan untuk pencarian seperti yang ditunjukkan pada Gambar 17.



Gambar 17 Menggabungkan Istilah Pencarian

Menggunakan Wild Card Kita dapat mencampur karakter pengganti dengan operator AND/OR dalam opsi pencarian kita. Pencarian berikut menghasilkan hasil yang berisi frasa fail, failed, failure, dll. dalam berkas log, serta istilah password pada baris yang sama.



Gambar 9.18 Menggunakan Wild Card

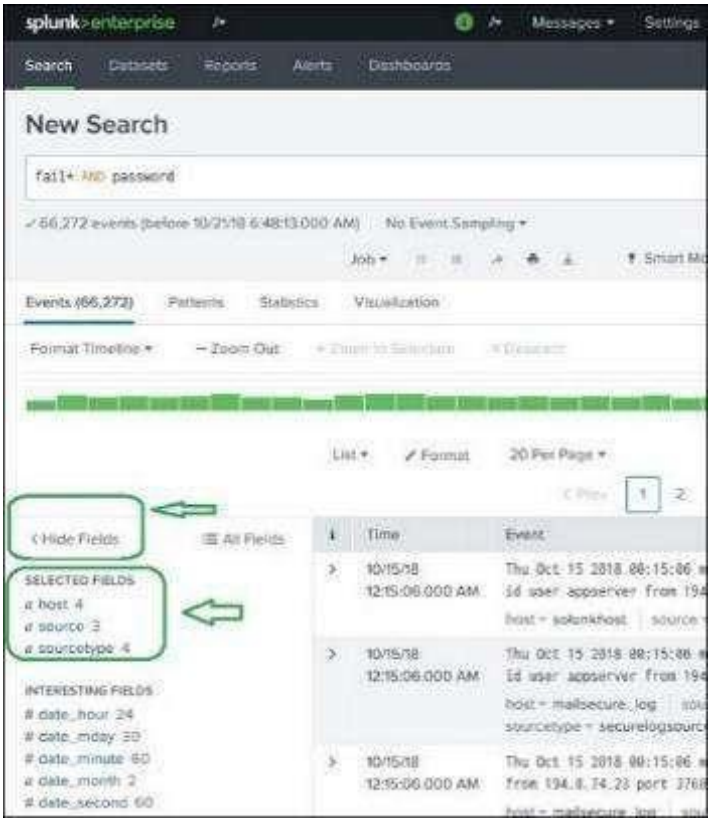
Memperbaiki Hasil Pencarian Dengan memilih string dan menambahkannya ke pencarian, kita dapat memfilter hasil lebih lanjut. Dalam contoh di bawah ini, kita memilih opsi Tambahkan ke Pencarian setelah mengarahkan kursor ke string 3351. Bila kita menambahkan 3351 ke frasa pencarian, kita akan mendapatkan hasil berikut, yang hanya menampilkan baris dari log yang menyertakan 3351. Perhatikan bagaimana garis waktu hasil pencarian berubah saat pencarian dipersempit seperti yang ditunjukkan pada Gambar 9.19.

Gambar 9.19 Penyempurnaan Hasil Pencarian

9.6 Pencarian Splunk-Field

Splunk mengevaluasi data mesin yang dikirimkan dan memisahkannya ke dalam sejumlah bidang, yang masing-masing mewakili satu kebenaran logis tentang rekaman data lengkap. Misalnya, satu

rekaman informasi dapat mencakup nama server, tanggal kejadian, jenis kejadian yang dicatat (seperti upaya login atau respons http), dan sebagainya. Splunk mencoba membagi bidang menjadi pasangan kunci dan nilai atau memisahkannya tergantung pada jenis data yang disertakan, seperti numerik dan teks, meskipun datanya tidak terstruktur.Kita dapat melihat kolom dari berkas secure.log dengan memilih opsi show fields, yang akan menampilkan halaman di bawah ini. Kolom yang dibuat oleh Splunk dari berkas log ini ditunjukkan pada Gambar 20.



Gambar 20 Pencarian Lapangan

Memilih Bidang Kita dapat memilih atau membatalkan pilihan bidang dari daftar semua bidang untuk menentukan bidang mana yang akan ditampilkan. Saat Anda mengeklik semua bidang, jendela akan muncul dengan daftar semua bidang. Beberapa bidang ini memiliki tanda centang di sebelahnya, yang menunjukkan bahwa bidang tersebut telah dipilih. Kita dapat menggunakan kotak centang untuk memilih bidang mana yang akan ditampilkan. Selain nama bidang, ia menunjukkan jumlah nilai berbeda yang dimilikinya, tipe data yang digunakannya, dan proporsi kejadian yang muncul di dalamnya.

Gambar 9.21 Memilih Bidang

Ringkasan

Splunk adalah alat untuk melacak dan mencari data dalam jumlah besar. Alat ini mengindeks dan mengorelasikan data dalam wadah yang dapat dicari dan memungkinkan pembuatan peringatan, laporan, dan visualisasi. Tujuan Splunk Enterprise adalah membantu Anda mengetahui apa yang terjadi di perusahaan Anda dan mengambil tindakan cepat. Splunk cloud adalah layanan platform data yang serbaguna, aman, dan hemat biaya yang memungkinkan Anda mencari, menganalisis, memvisualisasikan, dan bertindak berdasarkan data Anda. Splunk Light memecahkan masalah ini dengan memungkinkan Anda mengumpulkan dan menghubungkan data dari hampir semua sumber, format, atau lokasi. Data yang mengalir dari aplikasi klien dan paket, server aplikasi, server web, basis data, data jaringan, mesin virtual, sistem operasi, sensor, dan sumber lainnya hanyalah beberapa kemungkinan. Proses memperoleh dan mengimpor data untuk penggunaan atau penyimpanan langsung dalam basis data dikenal sebagai pengambilan data. Pengambilan data berarti "mengambil atau menyerap sesuatu." Data dapat diambil secara berkelompok atau disiarkan secara langsung. Pengindeksan adalah teknik untuk meningkatkan kecepatan basis data dengan mengurangi jumlah akses disk yang diperlukan saat kueri dijalankan. Ini adalah strategi struktur data untuk menemukan dan mengakses data dalam basis data dengan cepat. Beberapa kolom basis data digunakan untuk membuat indeks. Tampilan berbasis panel dikenal sebagai dasbor. Modul seperti kotak pencarian, kolom, diagram, tabel, dan daftar dapat disertakan dalam panel. Laporan sering kali ditautkan ke panel dasbor. Anda dapat menambahkan visualisasi pencarian atau laporan ke dasbor baru atau yang sudah ada setelah Anda membuatnya. Struktur data Anda ditentukan oleh model data Splunk, yang merupakan hierarki kumpulan data. Model data Anda

harus mewakili struktur dasar data serta laporan Pivot yang diminta oleh pengguna akhir Anda.

Soal latihan

1. Splunk adalah perangkat lunak yang digunakan untuk _____ data mesin.

A. pencarian dan perhatian
B. mencari dan menganalisis
C. berselancar dan menganalisis
D. tidak ada yang disebutkan

2. Menu drop down Administrator memungkinkan Anda untuk menyesuaikan dan memodifikasi informasi _____

A. Administrator
B. Pelaporan
C. Pelanggan
D. Pengguna

3. Tautan ke _____ membawa kita ke fitur tempat kita dapat menemukan kumpulan data yang dapat diakses untuk mencari laporan dan peringatan yang telah dihasilkan untuk pencarian ini.

A. pencarian dan catatan
B. pendek dan pelaporan
C. pencarian dan pelaporan
D. Tidak ada yang di atas

4. Pengambilan data di Splunk terjadi melalui fitur _____ yang merupakan bagian dari aplikasi pencarian dan pelaporan.

A. Tambahkan data
B. Unggah data
C. Menelan data
D. Tidak ada yang di atas

5.Apa yang disebut proses transformasi data?

A. Acara
B. Gudang
C. Pengindeksan
D. Tidak ada yang di atas

6. Opsi mana yang digunakan untuk meninjau pengaturan masukan.

A. Sumber
B. Tinjauan
C. Keduanya
D. Tidak ada yang di atas

7. Penggunaan TEZ adalah untuk _____

A. Membagi beban kerja menjadi bagian-bagian yang lebih kecil.
B. Mengunggah data
C. Menambahkan data
D. Tidak ada yang di atas

8. Masalah berikut manakah yang dihadapi setiap programmer dan perlu diingat oleh pengguna bisnis?

A. Tipe data
B. Ingatan

C. Penggunaan disk
D. Semua hal di atas

9.Pilih tahap-tahap di mana analitik bekerja.
A. Waktu desain/edit
B. Waktu eksekusi/berjalan
C. Keduanya
D. Tidak ada yang di atas

10. Pilih opsi yang dapat digunakan untuk menginstal splunk enterprise di windows
a. Antarmuka GUI
b. Antarmuka Baris Perintah
c. Keduanya
d. Tidak ada yang di atas

Daftar Pustaka

Eka Mayasari, & Agussalim Agussalim. (2023). Literature Review: Big Data dan Data Analys pada Perusahaan. *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(3), 171–187. https://doi.org/10.55606/juisik.v3i3.680

Fern??ndez, A., L??pez, V., Del Jesus, M. J., & Herrera, F. (2015). Revisiting Evolutionary Fuzzy Systems: Taxonomy, applications, new trends and challenges. *Knowledge-Based Systems*, *80*, 109–121. https://doi.org/10.1016/j.knosys.2015.01.013

Majumdar, J., Naraseeyappa, S., & Ankalaki, S. (2017). Analysis of

agriculture data using data mining techniques: application of big data. *Journal of Big Data*, *4*(1), 20. https://doi.org/10.1186/s40537-017-0077-4

Muhammad Syarif Hartawan, S. R., Hamid, A., Dari, W., & Putra, A. S. (2022). *Big Data (Informasi Dan Kasus)*.

Santoso, J. T. (2020). Analisis Big Data. In *Penerbit Yayasan Prima Agus Teknik*. https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/155

Siahaan, D. A. (2024). *MANAJEMEN PROYEK BIG DATA : TANTANGAN DAN STRATEGI DALAM MENGELOLA PROYEK ANALISIS DATA BESAR PADA ORGANISASI*. *03*(2), 53–60.

Varudharajulu, A. K., & Ma, Y. (2018). A Survey on Big Data Process Models for E-Business, E-Management, E-Learning, and E-Education. *International Journal of Innovative Research in Computer and Communication Engineering*, 220–222. https://doi.org/10.15680/IJIRCCE.2018

Veri Ferdiansyah, & Muhammad Irwan Padli Nasution. (2023). Penerapan Teknologi Big Data Dalam Pengembangan Database Pendidikan. *Jurnal Riset Manajemen*, *1*(3), 22–29. https://doi.org/10.54066/jurma.v1i3.591

Wardani, S., Lubis, S. S., & Dewantoro, R. W. (2025). *Analisis Big data untuk prediksi permintaan produk dalam E-commerce*. *1*.

Biodata Penulis

Dr. Erdisna, S. Kom, M. Kom, lahir di Padang Japang pada tanggal 09 Desember 1972. Menyelesaikan S1 dengan jurusan Manajemen Informatika Komputer pada tahun 1996 di UPI YPTK Padang. Pada tahun 2006 penulis lulus S2 di UPI YPTK Padang dengan jurusan Teknologi Informasi. Penulis menyelesaikan Program Doktor pada tahun 2020 di Universitas Negeri Padang, pada Fakultas Teknik dengan program studi Pendidikan Teknologi dan Kejuruan. Pertama kali diangkat sebagai Dosen tetap Yayasan di Kampus Universitas Putra Indonesia Yptk Padang Kopertis/LLDIKTI Wilayah X Padang pada tahun 2000 sampai tahun 2023 dan sekarang adalah dosen tetap Universitas Negeri Padang (UNP).

Zuhri Halim, S.Kom., M.Kom

Penulis lahir di Banyumas 13 Februari 1986, menyelesaikan Sekolah Menengah Atas 1 Muhammadiyah di Purwokerto tahun 2003, penulis melanjutkan pendidikan sarjana S1 Ilmu Komputer lulus pada tahun 2009 di STMIK Tasikmalaya dan melanjutkan pendidikan Magister Komputer S2 Ilmu Komputer, lulus pada tahun 2016 di STMIK ERESHA. Penulis saat ini sebagai dosen di Universitas Muhammadiyah Prof. Dr. Hamka

Karno Diantoro, M.Kom Penulis lahir di Jakarta 9 Oktober 1973. Menyelesaikan pendidikan Sekolah Dasar Negeri 02 (SDN 02) pada tahun 1986), pada tahun 1989 menyelesaikan Sekolah Menengah Pertama (SMPN) 138, pada tahun 1992 menyelesaikan Sekolah Menengah Atas (SMAN) 76, penulis melanjutkan pendidikannya sarjana S1 Ilmu Computer pada tahun 1997 di STI&K Jakarta Jurusan Teknik Komputer, lalu menyelesaikan Magister Komputer S2 Ilmu Komputer pada tahun 2016 di STMIK ERESHA. Penulis mengikuti seminar, kursus dan workshop yaitu : pada tahun 1992 "English Course" , pada tahun 1993 "Microsoft & Aldus Presentation", pada tahun 1995 "Latihan Dasar Kepemimpinan Manajemen Mahasiswa" pada tahun 1996 "Seminar & Workshop Internet" pada tahun 1996 "Data Communication", pada tahun 1999 "Japan Course", pada tahun 2001 "Developing the Internet Based Information System", pada tahun 2002 "ISO 9001", pada tahun 2004 "ASP.Net & MS SQL Server 2000" pada tahun 2004 "ISO 14001:1996", pada tahun 2004 "Supervisory Management I", pada tahun 2004 "Developing ERP Project" pada tahun 2005 "Supervisory Management II".

Ardian adalah seorang penulis kelahiran Semarang dengan latar belakang pendidikan yang kuat di bidang teknologi informasi. Ia meraih gelar Sarjana (S1) di bidang Telekomunikasi dari Sekolah Tinggi Teknologi Telkom Bandung pada tahun 2013, dan kemudian melanjutkan pendidikannya dengan meraih gelar Magister (S2) di bidang Sistem Informasi dari Universitas Dian Nuswantoro.Pengalaman profesional Ardian dimulai di PT Indonesia Comnet Plus PLN, di mana ia mengaplikasikan pengetahuan dan

keahliannya. Jiwa kewirausahaan kemudian mendorongnya untuk mendirikan usaha di kota kelahirannya, Semarang. Selain aktif sebagai pengusaha, Ardian juga berbagi ilmunya dengan menjadi seorang dosen di Universitas Wahid Hasyim Semarang. Dengan keahlian khusus di bidang **IT Security** dan **Networking**, tulisan-tulisan Ardian kemungkinan besar akan memberikan wawasan yang mendalam dan relevan terkait keamanan informasi, jaringan komputer, dan berbagai aspek teknologi lainnya. Pengalaman praktisnya sebagai pengusaha dan pendidik akan memperkaya perspektif yang ia tawarkan dalam setiap karyanya.

Desi Anggreani lahir di Nunukan, Kalimantan Utara, pada tanggal 12 November 1996. Penulis merupakan anak kedua dari empat bersaudara. Pendidikan dasar dan menengahnya ditempuh di Kabupaten Enrekang, Sulawesi Selatan, masing-masing di SD Negeri 109 Tuara Enrekang, SMP Negeri 3 Anggeraja Enrekang, dan SMA Negeri 1 Enrekang. Pendidikan tinggi jenjang Sarjana (S1) diselesaikan di Universitas Muslim Indonesia (UMI) Makassar pada Program Studi Teknik Informatika, Fakultas Ilmu Komputer. Selanjutnya, penulis melanjutkan studi Magister (S2) di Universitas Negeri Malang pada Program Studi Teknik Elektro, Fakultas Teknik. Bidang keilmuan yang ditekuni penulis mencakup teknologi informasi, kecerdasan buatan (artificial intelligence), dan ilmu data (data science). Penulis aktif dalam kegiatan penelitian dan pengembangan, khususnya dalam penerapan teknologi digital di berbagai sektor. Fokus utama penelitian penulis terletak pada pengembangan aplikasi berbasis kecerdasan buatan, sistem pendukung pengambilan keputusan serta peramalan (forecasting) berbasis AI.

Dr. Ir. Indriyani, S.Kom., M.Kom. adalah akademisi dan pebisnis yang berdedikasi di bidang Teknologi Informasi. Lahir di Surabaya pada 6 Maret 1990 dan tinggal di Bali, beliau menyelesaikan Pendidikan Informatika dan Komputer Terapan (PIKTI) di ITS 10 Nopember Surabaya pada 2009, gelar Sarjana Teknik Informatika di Universitas Wijaya Kusuma Surabaya (2014), S2 Teknik Informatika ITS 10 Nopember Surabaya (2017), dan gelar Doktor di Teknologi Informasi dari Universitas Udayana (2024). Sebagai dosen di Program Studi Sistem Komputer Institut Teknologi dan Bisnis STIKOM Bali sejak 2017 sampai sekarang, beliau membimbing mahasiswa di bidang teknologi dan inovasi digital. Saat ini, beliau terus berkontribusi pada dunia pendidikan dan kewirausahaan. Email: indriyani@stikom-bali.ac.id

Ir. Anwar T. Sitorus, M.Kom Penulis lahir di Medan 8 Januari 1967. Menyelesaikan pendidikan Sekolah Menengah Atas (SMAN) 11 di Medan, kemudian penulis melanjutkan pendidikannya sarjana S1 Teknik Informatika pada tahun 1991 di STT INTEN Bandung Jurusan Teknik Informatika, lalu menyelesaikan Magister Komputer S2 Ilmu Komputer pada tahun 2018 di STMIK Nusa Mandiri Jakarta. Penulis mengikuti seminar, kursus dan workshop yaitu : pada tahun 2022 “Sistem Manajemen Keamanan Sistem Informasi ISO/EIC 27001:2013”, pada tahun 2002 “Leadership Training for Managers by Dale Carnegie”, pada tahun 2001 “Performance Management Training by PQM Consultans”, pada tahun 2001 “Sql and Application Tuning of Oracle”, pada tahun 2000 “Internal Quality Auditors of ISO

9000", pada tahun 1996 "Sql, pl/sql plus, form, report and graphics of Oracle", pada tahun 1994 "Database Administrator (DBA) of Oracle", pada tahun 1994 "Database advance of Datafit (DP4)", pada tahun 1994 "Database (RDBMS) of Datafit (DP4)", pada tahun 1992 "RM-Cobol85 of AT&T Unix platform", pada tahun 1992 "System administration & Operating systems of AT&T Unix V Rel. 04", pada tahun 1991 "Fundamental & Operating systems of HP-Unix 8 Rel. 01", pada tahun 1991 "Database (RDBMS) & Project Management Systems of Artemis 7000".

Pengantar BIG DATA

Buku Pengantar Big Data disusun sebagai upaya untuk memberikan pemahaman dasar mengenai konsep, teknologi, serta implementasi big data yang terus berkembang pesat di tengah era transformasi digital. Buku ini mengupas berbagai aspek fundamental big data secara sistematis, menjadikannya sumber yang relevan dan bermanfaat baik bagi kalangan akademisi maupun praktisi yang ingin memahami dan memanfaatkan potensi data dalam skala besar.Dengan latar belakang kemajuan teknologi informasi yang menyebabkan lonjakan data dalam volume besar, beragam jenis, dan pertumbuhan yang sangat cepat, buku ini hadir untuk menjawab tantangan sekaligus membuka peluang baru dalam penyimpanan, pengolahan, dan analisis data. Melalui pembahasan yang komprehensif dan mudah dipahami, buku ini menjadi bekal penting bagi siapa pun yang ingin memahami peran strategis big data dalam kehidupan modern.

ISBN 978-634-7237-08-8
9 786347 237088

Penerbit
Gita Lentera

Office1:Perm. Permata hijau regency blok F/1 kelurahan Pisang kecamatan Pauh kota Padang, Sumatera Barat
Office2: Jl Weling no120 Gejayan, Yogyakarta
Cp. Admin: +62823-8699-7194
git4lenter4@gmail.com www.gitalentera.com

IKAPI
IKATAN PENERBIT INDONESIA
Anggota IKAPI
No. 042/SBA/2023