BAGIAN 1

DEFINISI CLOUD COMPUTING

Revolusi Cloud Computing

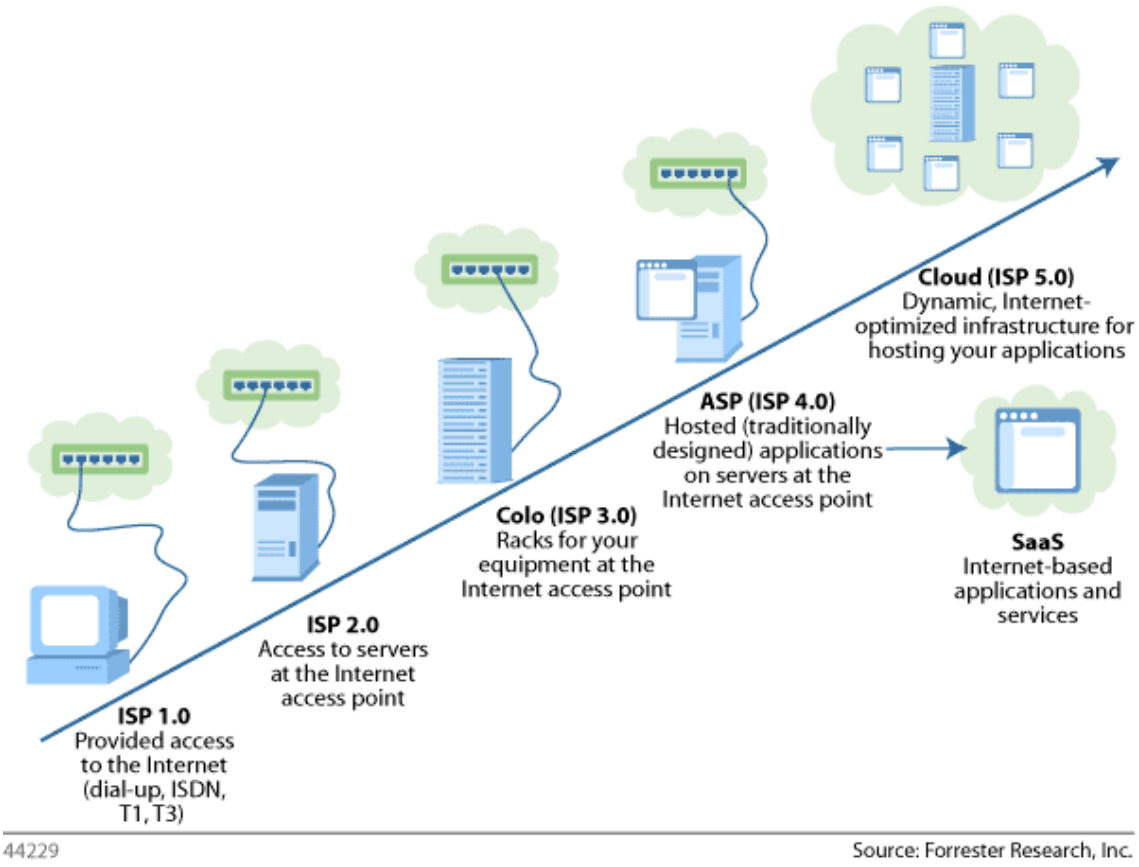
Cloud computing sebuah jargon baru diabad milenium yang lahir untuk merepresentasikan semua hal, sebuah konsep baru yang merubah mekanisme bagaimana mengembangkan sistem dilakukan. Sebuah metode virtualisasi yang memungkinkan sistem operasi, middleware, database server, email server sampai web itu sendiri adalah satu lapisan yang sama. Padahal kita tahu tidak ada satu server pun yang tidak dapat berjalan tanpa sistem operasi.

Cloud ini telah menjadi backbone dan infrastruktur pendukung baik di Google, Facebook, Yahoo maupun diberbagai dotcom dunia. Saat ini konsep cloud ini telah memasukin perusahaan-perusahaan, dan sedang mentransformasi penyimpanan dan operasi perusahaan. Mekanisme backup dan recovery yang dijadikan standar operasi pemeliharaan sistem, telah bertransformasi menjadi real time data warehousing, karena penambahan informasi tidak memungkinkan dibackup kembali. Karena sistem harus berjalan 24 jam setiap hari dan tidak boleh ada masalah. Google memiliki layanan gmail yaitu sebuah mail gratis untuk dunia, pernah mengalamin masalah 30 menit dalam sejarah Google berdiri, telah menjadi bulan-bulanan berita kehandalan diberbagai media masa. Quality of Services sangat penting dalam dunia Internet ini, tentu saja diperlukan untuk membuat para pengakses yang semakin banyak merasa nyaman.

Adapun berbagai tipe dari implementasi cloud computing adalah sebagai berikut:

- SaaS – Software as a Service, berbentuk aplikasi, contohnya adalah Salesforce, NetSuite
- PaaS – Platform as a Service, implementasi dari database, file system, web server, middleware, contohnya adalah Heroku, Engine Yard, Azure
- IaaS – Infrastructure as a Service, berbentuk virtualisasi dari infrastruktur seperti Amazon, Rackspace

Adapun implementasi ini semua adalah berjenjang, dimana tingkatan menciptakan piranti lunak sebagai layanan adalah yang terkompleks. Yang dapat dilihat pada diagram dibawah ini:



Gambar 1. Revolusi computasi
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Dari awal menawarkan konektivitas internet dasar untuk menawarkan perangkat lunak sebagai layanan ISP. ISP 1,0 adalah semua menyediakan internet akses ke pelanggan, ISP 2,0 adalah fase di mana ISP yang ditawarkan kemampuan hosting Langkah berikutnya adalah co-lokasi melalui yang ISP mulai leasing keluar ruang rak dan bandwidth. Dengan ini, perusahaan bisa host server mereka menjalankan, Line of Business (LOB) aplikasi yang dapat diakses melalui web dengan karycloud, mitra dagang dan pelanggan. ISP 3.0 adalah menawarkan aplikasi pada langganan mengakibatkan Application Service Provider (ASP) kemudian muncul Software terbaru sebagai Service atau SaaS, adalah model ASP matang dan langkah logis untuk ISP akan merangkul Cloud.

Pendekatan piranti lunak sebagai services atau SaaS, memungkinkan integasi dan interoperabilitas antara piranti lunak, yang mana dalam implementasinya akan terancukan dalam konsep grid atau clustering, dimana dalam utilisasi yang sebenarnya menjalankan grid diatas infrastruktur tervirtualisasi akan memakan banyak resource. Cloud infrastrcuture sangat efisien bilamana diimplementasikan dalam sistem kecil.



Gambar 2. Cloud Computing visual diagram
Sumber : http://infreemation.net

Evolusi Cloud

Cloud computing merupakan evolusi yang mengadopsi virtualization, service-oriented architecture and utility computing. Cloud computing memungkinkan konsumen teknologi untuk memikirkan komputasi secara efektif dengan biaya minimal dan dapat diandalkan. Hal lain yang juga tidak perlu lagi dikhawatirkan oleh pengguna adalah tentang bagaimana membangunnya, cara kerjanya, siapa yang mengoperasikan atau di mana harus meletakkan.

Virtualization adalah penciptaan versi virtual (bukan aktual) terhadap sumber daya teknologi informasi, seperti sistem operasi, server, perangkat penyimpanan (storage) atau sumber daya jaringan. Virtualisasi dapat dilihat sebagai bagian dari tren secara menyeluruh dari Enterprise TI yang mencakup autonomic computing. Autonomic computing merupakan sebuah skenario di mana lingkungan TI akan mampu melakukan pengelolan sendiri (self management) didasarkan pada aktivitas yang dirasakan dan utility computing. Kekuatan pemrosesan komputer dianggap sebagai sebuah utilitas yang memungkinkan klien membayar sesuai yang diperlukan.

da beberapa pemahaman tentang *Cloud Computing* yang dapat membantu kita untuk mengenal apa itu *Cloud Computing :*

a. Internet bisa dianggap cloud besar. Cloud berisi komputer yang semuanya saling tersambung. Dari situlah berasal istilah 'cloud'. Jadi semuanya disambungkan ke 'cloud', atau cloud itu." (Stevan Greve)

b. *Cloud Computing* adalah gabungan pemanfaatan teknologi komputer ('*komputasi*') dan pengembangan berbasis Internet ('cloud'). Cloud *(cloud)* adalah metefora dari internet, sebagaimana cloud yang sering digambarkan di diagram jaringan komputer, cloud *(cloud)* dalam Cloud Computing juga merupakan abstraksi dari infrastruktur kompleks yang disembunyikannya.

c. Dalam perspektif teknologi komunikasi sendiri, *cloud computing* atau komputasi cloud dapat diartikan sebagai suatu teknologi yang memanfaatkan internet sebagai *resource* untuk komputasi yang dapat di-*requset* oleh pengguna dan merupakan sebuah layanan dengan pusat server bersifat virtual atau berada dalam *cloud* (internet) itu sendiri (Krishnadi, 2010).



Gambar 3. Ilustrasi Cloud Computing

Sumber :(Pew Internet, 2011)

Cara Kerja Cloud Computing

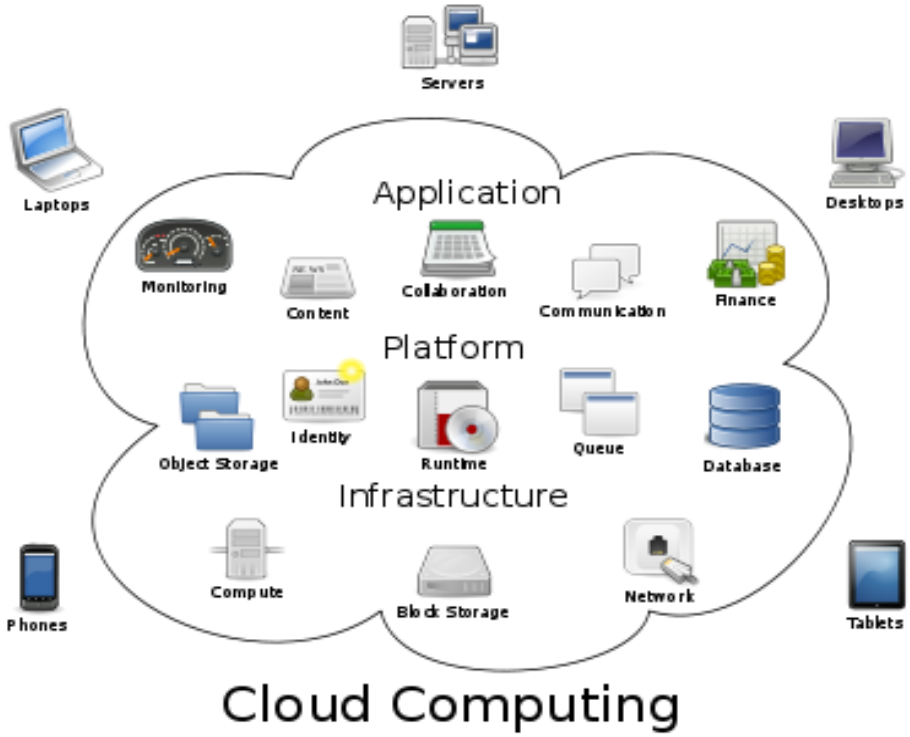
Seorang pengguna cloud membutuhkan perangkat klien seperti laptop atau komputer desktop, komputer pad, ponsel pintar, atau sumber daya komputasi lainnya dengan web browser (atau rute akses lain yang disetujui) untuk mengakses sistem cloud melalui World Wide Web. Biasanya pengguna akan login ke cloud pada penyedia layanan atau perusahaan swasta. Cloud computing bekerja secara client-server, menggunakan protokol web browser. cloud menyediakan server berbasis aplikasi dan semua layanan data kepada pengguna, dengan output ditampilkan pada perangkat klien. Jika pengguna ingin membuat dokumen menggunakan pengolah kata, misalnya, cloud menyediakan aplikasi yang cocok yang berjalan pada server yang menampilkan pekerjaan yang dilakukan oleh pengguna pada layar web browser klien.

Memori yang dialokasikan untuk web browser sistem klien digunakan untuk membuat data aplikasi muncul di layar sistem klien, tetapi semua perhitungan dan perubahan dicatat oleh server, dan hasil akhir termasuk file yang dibuat atau diubah secara permanen disimpan pada server cloud. Kinerja dari aplikasi cloud tergantung pada kecepatan akses jaringan, dan kehandalan serta kecepatan pemrosesan perangkat klien.

Sejak layanan cloud berbasis web, bekerja pada berbagai platform, termasuk Linux, Macintosh, dan komputer Windows. Ponsel Smart, bantalan dan perangkat tablet dengan Internet dan mengakses World Wide Web juga menyediakan layanan cloud untuk telecommuting dan pengguna ponsel. Sebuah penyedia layanan mungkin mempuyai kekuatan pemrosesan dari beberapa komputer remote dalam cloud untuk mencapai tugas-tugas rutin seperti back up sejumlah besar data ,pengolah kata , atau pekerjaan komputasi secara intensif. Tugas-tugas ini biasanya mungkin sulit, memakan waktu, atau mahal untuk pengguna individu atau perusahaan kecil untuk menyelesaikan, terutama dengan sumber daya komputasi yang terbatas dan dana.

Dengan komputasi cloud, klien hanya memerlukan komputer sederhana, seperti netbook , dirancang dengan komputasi cloud dalam pikiran, atau bahkan smartphone , dengan koneksi ke Internet, atau jaringan perusahaan, dalam rangka untuk membuat permintaan data dari cloud, maka istilah " perangkat lunak sebagai layanan "(SaaS). Perhitungan dan penyimpanan dibagi antara komputer

remote untuk menangani volume besar dari kedua, sehingga klien tidak perlu membeli perangkat keras mahal atau perangkat lunak untuk menangani tugas. Hasil dari tugas pengolahan dikembalikan ke klien melalui jaringan, tergantung pada kecepatan koneksi internet.



Gambar 1. Diagram konsepsual dari Komputasi awan
Sumber : http://id.wikipedia.org/wiki/Komputasi_awan

Resiko Cloud Computing

Dalam segala hal yang berhubungan dengan teknologi baru, selain menawarkan keunggulan-keunggulan dan segala kemudahannya, tentunya ada resiko yang harus siap ditanggung. Begitu pula dengan *cloud computing*. Di samping segala keunggulan dan kemudahannya, teknologi *cloud computing* tetap memiliki resiko. Beberapa resiko yang mungkin terjadi antara lain:

1. Service Level

   Cloud provider mungkin tidak akan konsisten dengan performance dari application atau transaksi. Hal ini mengharuskan anda untuk memahami service level yang anda dapatkan mengenai *transaction response time*, data *protection* dan kecepatan data *recovery*.

2. Privacy

   Karena orang lain / perusahaan lain juga melakukan hosting kemungkinan data anda akan keluar atau di baca oleh pemerintah U.S. dapat terjadi tampa sepengetahuan anda atau approve dari anda.

3. Compliance

   Anda juga harus memperhatikan regulasi dari bisnis yang anda miliki, dalam hal ini secara teoritis cloud service provider diharapkan dapat menyamakan level compliance untuk penyimpanan data di dalam cloud, namun karena service ini masih sangat muda anda diharapkan untuk berhati hati dalam hal penyimpanan data.

4. Data Ownership

   Apakah data anda masih menjadi milik anda begitu data tersebut tersimpan di dalam cloud? mungkin pertanyaan ini sedikit aneh, namun anda perlu mengetahui seperti hal nya yang terjadi pada Facebook yang mencoba untuk merubah *terms of use aggrement*-nya yang mempertanyakan hal ini.

5. Data Mobility

   Apakah anda dapat melakukan share data diantara cloud service? dan jika anda terminate cloud

Syarat Cloud Computing

- **On-Demand Self-Services**

  Sebuah layanan *cloud computing* harus dapat dimanfaatkan oleh pengguna melalui mekanisme swalayan dan langsung tersedia pada saat dibutuhkan. Campur tangan penyedia layanan adalah sangat minim. Jadi, apabila kita saat ini membutuhkan layanan aplikasi CRM (sesuai contoh di awal), maka kita harus dapat mendaftar secara swalayan dan layanan tersebut langsung tersedia saat itu juga.

- **Broad Network Access**

  Sebuah layanan *cloud computing* harus dapat diakses dari mana saja, kapan saja, dengan alat apa pun, asalkan kita terhubung ke jaringan layanan. Dalam contoh layanan aplikasi CRM di atas, selama kita terhubung ke jaringan Internet, saya harus dapat mengakses layanan tersebut, baik itu melalui laptop, desktop, warnet, handphone, tablet, dan perangkat lain.

- **Resource Pooling**

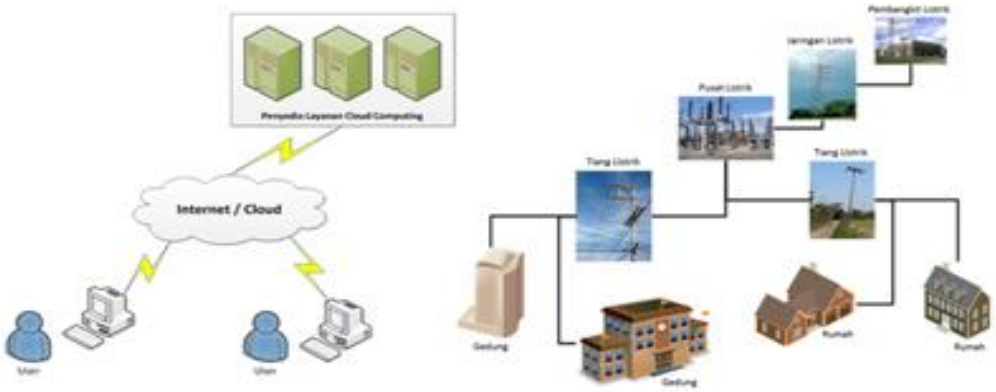
  Sebuah layanan *cloud computing* harus tersedia secara terpusat dan dapat membagi sumber daya secara efisien. Karena *cloud computing* digunakan bersama-sama oleh berbagai pelanggan, penyedia layanan harus dapat membagi beban secara efisien, sehingga sistem dapat dimanfaatkan secara maksimal.

- **Rapid Elasticity**

  Sebuah layanan *cloud computing* harus dapat menaikkan (atau menurunkan) kapasitas sesuai kebutuhan. Misalnya, apabila pegawai di kantor bertambah, maka kita harus dapat menambah *user* untuk aplikasi CRM tersebut dengan mudah. Begitu juga jika pegawai berkurang. Atau, apabila kita menempatkan sebuah *website* berita dalam jaringan *cloud computing* , maka apabila terjadi peningkatkan *traffic* karena ada berita penting, maka kapasitas harus dapat dinaikkan dengan cepat.

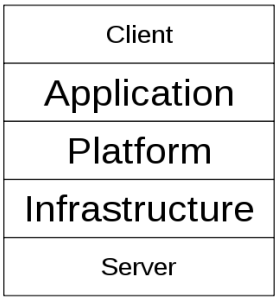
- **Measured Service**

  Sebuah layanan *cloud computing* harus disediakan secara terukur, karena nantinya akan digunakan dalam proses pembayaran. Harap diingat bahwa layanan *cloud computing* dibayar sesuai penggunaan, sehingga harus terukur dengan baik.



Gambar 2. Karakteristik Cloud Computing
Sumber : http://id.wikipedia.org/wiki/Komputasi_awan

Layer Cloud Computing



- **Clients (klien)**

  Sebuah Klien cloud (cloud client) terdiri dari perangkat keras komputer dan / atau perangkat lunak komputer yang mempercayakan pada Cloud Computing (Kom puterisasi cloud) untuk mengirimkan aplikasi, atau yang secara khusus didisain untuk pengiriman dari jasa cloud (cloud services) dan ini, pada kasus lain, sebenarnya sia-sia seperti antara lain: Mobile (Linux based – Palm Pre-WebOS Linux Kernel, Android-Linux Kernel, iPhone-Darwin Linux Kernel, Microsoft based – Windows Mobile) Thin client (CherryPal, Wyse, Zonbu, gOS-based systems) Thick client / Web browser (Internet Explorer, Mozilla Firefox, Google Chrome, WebKit).

- **Application (aplikasi)**

  Sebuah aplikasi cloud (Cloud Application) mempengaruhi Cloud Computing (Komputerisasi cloud) pada arsitektur perangkat lunak (software), sering menghilangkan kebutuhan untuk menginstal dan menjalankan aplikasi pada pengguna komputer (user) sendiri, dengan demikian mengurangi beban dari pemeliharaan perangkat lunak, operasi berkelanjutan, dan dukungan. Antara lain:

  - Peer-to-peer / volunteer computing (BOINC, Skype)
  - Web applications (Webmail, Facebook, Twitter, YouTube)
  - Security as a service (MessageLabs, Purewire, ScanSafe, Zscaler)
  - Software as a service (A2Zapps.com, Google Apps, Salesforce,Learn.com, Zoho, BigGyan.com)
  - Software plus services (Microsoft Online Services)
  - Storage [Distributed]
  - Content distribution (BitTorrent, Amazon CloudFront)
  - Synchronisation (Dropbox, Live Mesh, SpiderOak, ZumoDrive)

- **Platform**

  Suatu platform cloud (cloud platform) (PaaS) mengantarkan sebuah platform komputasi dan/atau solusi tumpukan (solution stack) sebagai satu jasa,

umumnya menggunakan infrastruktur cloud (cloud infrastructure) dan aplikasi pendukung cloud (supporting cloud applications). Hal ini memudahkan penyebaran aplikasi tanpa biaya dan kompleksitas dari pembelian dan mengatur perangkat keras dasar dan lapisan perangkat lunak. Antara lain:

- Services
- Identity (OAuth, OpenID)
- Payments (Amazon Flexible Payments Service, Google Checkout, PayPal)
- Search (Alexa, Google Custom Search, Yahoo! BOSS)
- Real-world (Amazon Mechanical Turk)
- Solution stacks
- Java (Google App Engine)
- PHP (Rackspace Cloud Sites)
- Python Django (Google App Engine)
- Ruby on Rails (Heroku)
- .NET (Azure Services Platform, Rackspace Cloud Sites)
- Proprietary (Force.com, WorkXpress, Wolf Frameworks)
- Storage [Structured]
- Databases (Amazon SimpleDB, BigTable)
- File storage (Centerra Blades,Amazon S3, Nirvanix, Rackspace Cloud Files) Queues (Amazon SQS)

- **Infrastructure**

  Infrastruktur cloud (Cloud infrastructure) (IaaS) adalah pengiriman dari infrastruktur komputer, yang secara khusus pada suatu platform lingkungan virtualisasi, sebagai satu jasa. Antara lain:

  - Compute (Amazon CloudWatch, RightScale)
  - Physical machines
  - Virtual machines (Amazon EC2, GoGrid, iland, Rackspace Cloud Servers)
  - OS-level virtualization
  - Network (Amazon VPC)
  - Storage [Raw] (Amazon EBS)

- **Servers**

  Lapisan server terdiri dari perangkat keras komputer dan / atau produk perangkat lunak komputer yang terperinci didisain untuk pengiriman dari jasa cloud (cloud services). Seperti: Fabric computing (Cisco UCS) Cloud Services (Layanan Cloud) = Pengguna dan produk bisnis, layanan dan solusi yang dikirim dan digunakan pada waktu yang real-time melalui jaringan InternetmCloud Computing (Komputerisasi Cloud) = sesuatu yang muncul setelah pengembangan IT, penyebaran dan model pengiriman, memungkinkan pengiriman produk secara real-time, pelayanan dan solusi-solusi melalui jaringan internet (dengan kata lain memungkinkan Cloud Service).

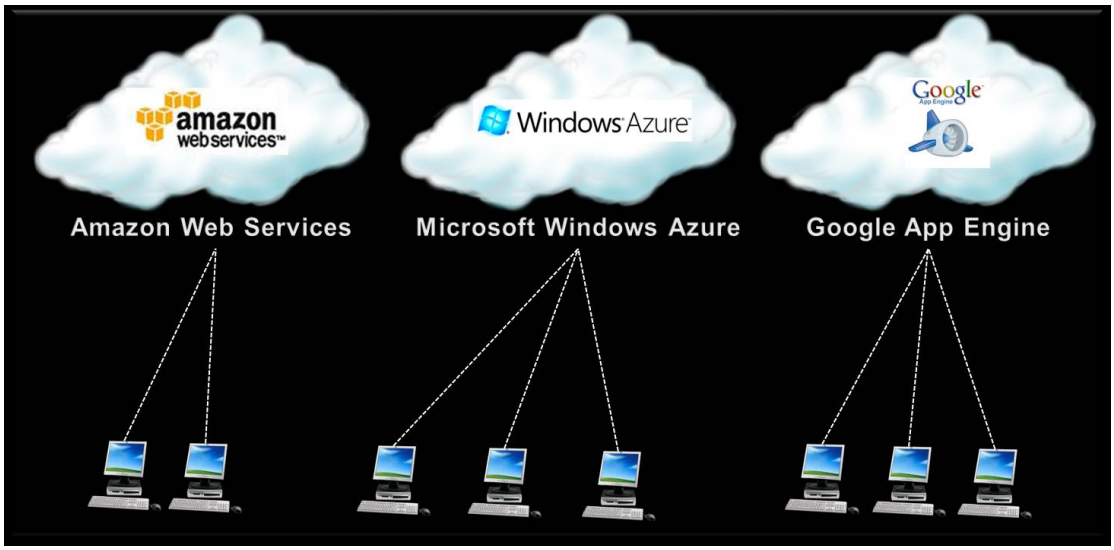
  Untuk beberapa tahun Kedepannya teknologi Cloud Computing (Komputerisasi cloud) akan berkembang secara pesat seiring dengan kemajuan teknologi yang didukung oleh perusahaan besar seperti IBM, Google, Microsoft, Yahoo, Amazon, dan lain-lain sehingga proses pengiriman dan penerimaan data menjadi lebih baik sehingga setiap orang dapat menikmati layanan internet dengan Cepat dan Murah.

Tipe Penerapan Layanan Cloud Computing

- **Public Cloud (Cloud Publik)**

  Cloud Public atau cloud eksternal mendeskripsikan komputasi cloud pada arti tendensi tradisional, yang mana sumber daya dengan ketentuan dinamis pada suatu fine-grained (perbaikan), basis pelayanan sendiri lewat Internet, melalui aplikasi web / jasa web, dari satu lokasi penyedia off-site oleh pihak ketiga yang berbagi sumber daya dan daftar kegunaan pada suatu fine-grained utility computing.

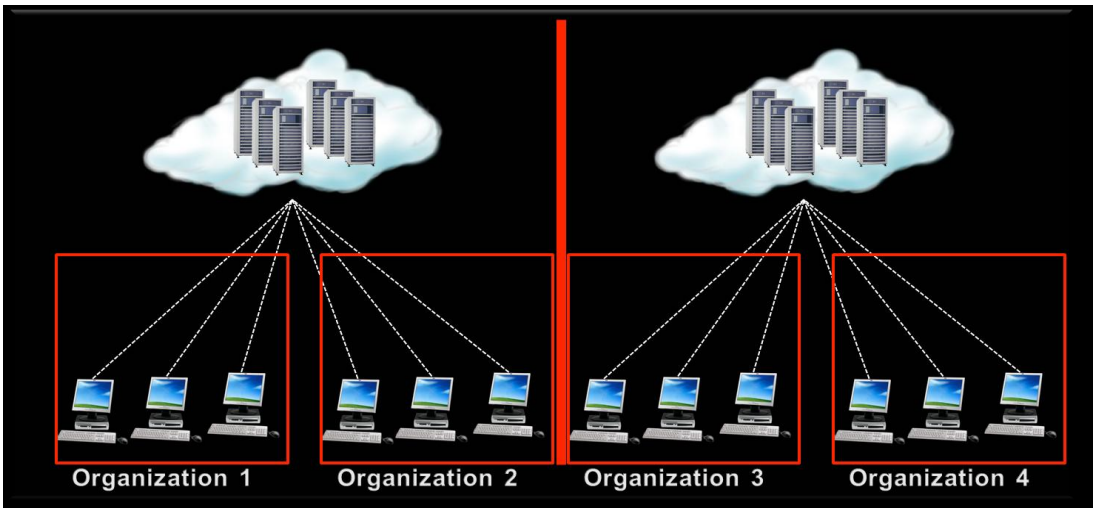
  Hybrid Cloud (Cloud Bastar) Satu lingkungan cloud bastar (Hybrid Cloud) terdiri dari beberapa penyedia layanan internal dan / atau penyedia layanan eksternal "akan di khususkan untuk banyak perusahaan". Satu cloud bastar dapat mendeskripsikan konfigurasi dengan mengombinasikan satu alat lokal (local device), seperti pada Plug computer dengan layanan Cloud (cloud). Hal ini juga dapat menjabarkan konfigurasi yang mengkombinasikan antara virtual dan physical, aset colocated misalnya, kebanyakan lingkungan yang divirtualisasikan yang memerlukan server fisik, Router, atau perangkat keras lain seperti satu alat jaringan yang berfungsi sebagai firewall atau spam filter.



Gambar 3. Public Cloud
Sumber : Janakiram MSV Cloud Computing Strategist 2010

- **Community Cloud**

  Dalam model ini, sebuah infrastruktur cloud digunakan bersama-sama oleh beberapa organisasi yang memiliki kesamaan kepentingan, misalnya dari sisi misinya, atau tingkat keamanan yang dibutuhkan, dan lainnya. Jadi, community cloud ini merupakan "pengembangan terbatas" dari private cloud. Dan sama juga dengan private cloud, infrastruktur cloud yang ada bisa di-manage oleh salah satu dari organisasi itu, ataupun juga oleh pihak ketiga.



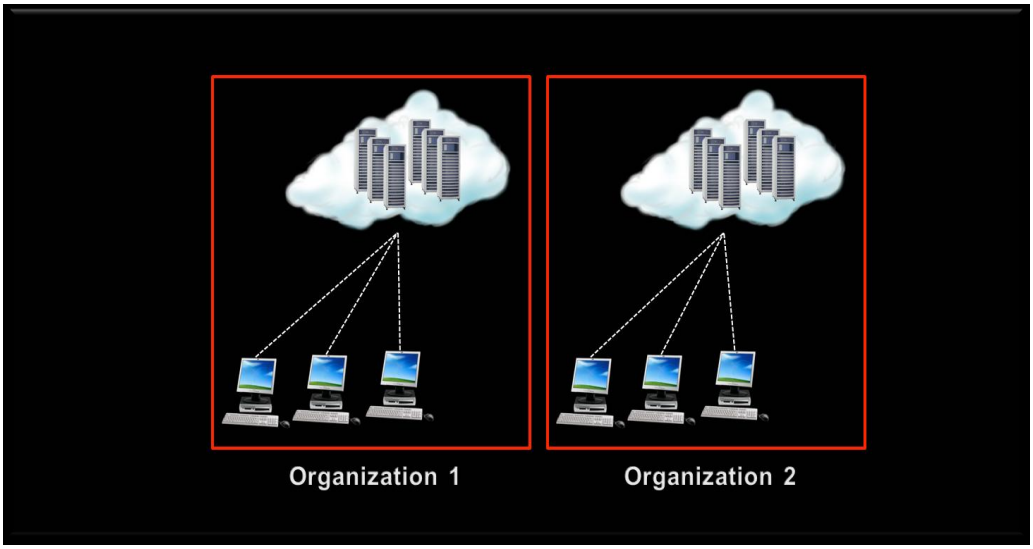
Gambar 4. Community Cloud
Sumber : Janakiram MSV Cloud Computing Strategist 2010

- **Private Cloud (Cloud Pribadi)**

  Private Cloud dan cloud internal adalah pembentukan kata baru yang yang baru-baru ini telah dipergunakan oleh beberapa penjual (vendor) untuk deskripsikan penawaran yang menandingi Cloud Computing (Komputerisasi cloud) pada jaringan pribadi. Produk ini (secara khas virtualisasi otomatisasi) menyatakan "mengantarkan beberapa keuntungan-keuntungan dari Cloud Computing (Komputerisasi cloud) tanpa jebakan (pitfalls)", kapitalisasi pada jaminan sekuritas data, perusahaan pemerintahan, dan dapat diandalkan. Mereka telah dikritik atas dasar pengguna itu "masih perlu untuk beli, bangun, dan mengatur mereka" dan seperti halnya tidak menguntungkan dari menurunkan biaya modal dan sedikit campur tangan manajemen langsung, sebenarnya "Kurangnya model ekonomi yang membuat Cloud Computing (Komputerisasi cloud) seperti satu konsep pembangkit minat".

Ketika seorang ahli analisa meramalkan pada tahun 2008 jaringan Private Cloud akan menjadi masa depan perusahaan IT, ada beberapa ketidak-pastian apakah mereka adalah satu hakikat walaupun diantara firma yang sama. Ahli analisa juga mengakui itu dalam lima tahun sebuah "persentase yang sangat besar" dari perusahaan kecil dan menengah akan mendapat semakin banyak dari sumber daya komputasi mereka dari penyedia eksternal Cloud Computing (Komputerisasi cloud) seperti mereka "tidak akan punya ekonomi dari skala untuk membuat ini berharga tinggal di dalam bisnis IT" atau mampu untuk mengusahakan cloud pribadi (Private Cloud). Ahli analisa telah laporkan pada pandangannya Platform pribadi itu cloud adalah satu batu loncatan ke cloud eksternal, terutama untuk jasa keuangan, dan kedepannya datacenters (pusat data) akan menyerupai cloud internal.

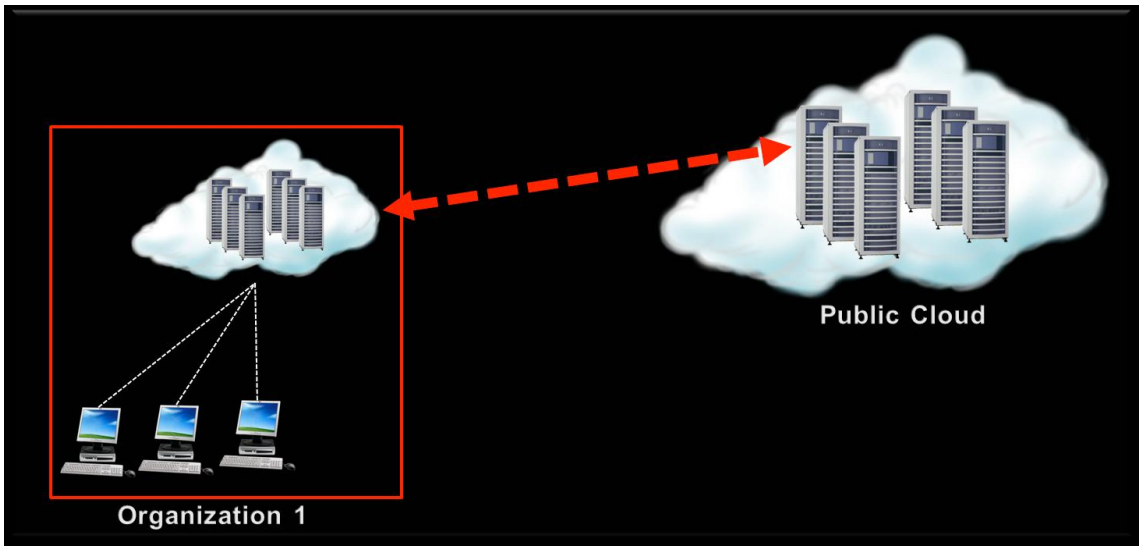
Masa telah dipergunakan secara logis agak dibandingkan rasa fisik, antara lain berhubungan dengan penawaran platform sebagai suatu jasa layanan meskipun demikian, penawaran program Microsoft's Azure Services tidak tersedia pada penyebaran on-premises (pendapat).



Gambar 5. Private Cloud
Sumber : Janakiram MSV Cloud Computing Strategist 2010

- **Hybrid Cloud**

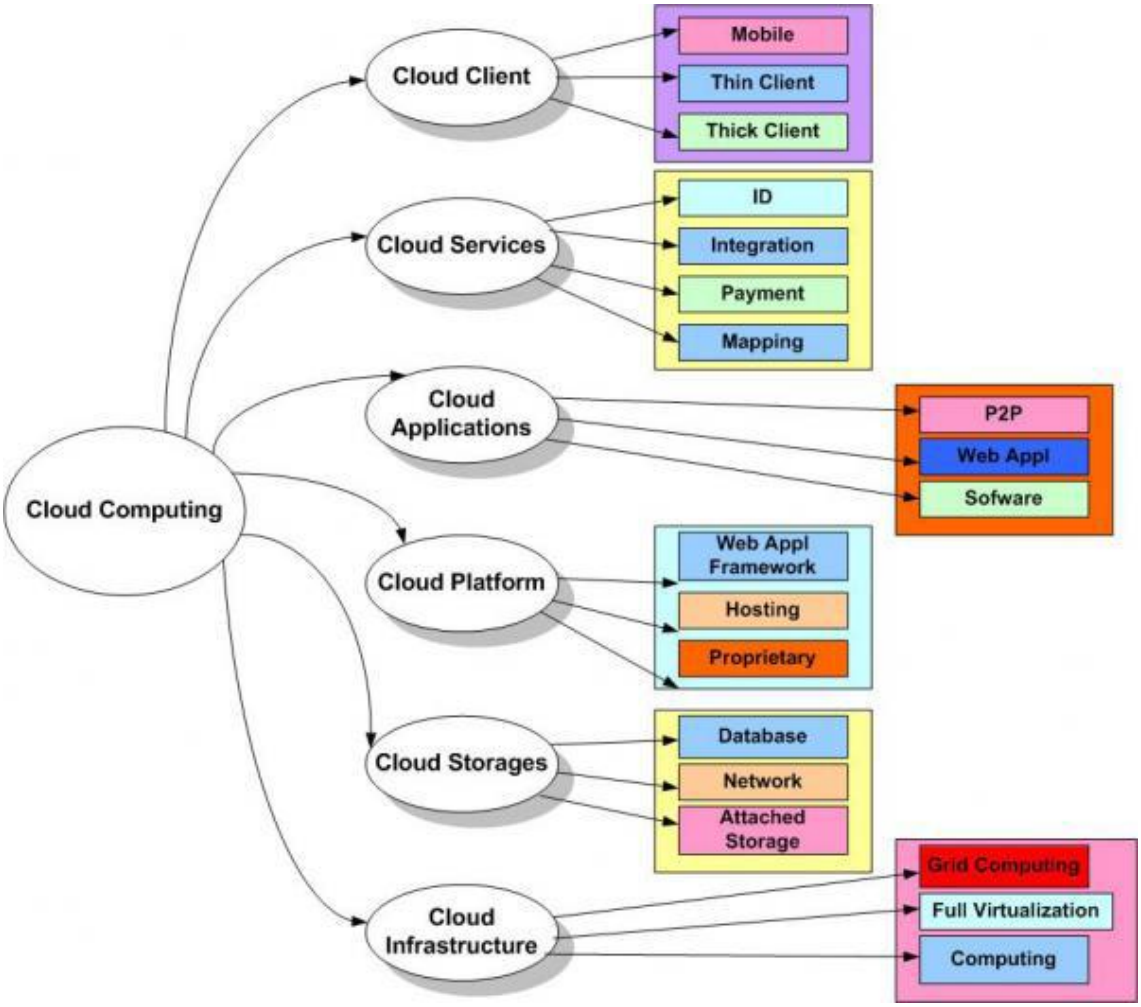
  Untuk jenis ini, infrastruktur cloud yang tersedia merupakan komposisi dari dua atau lebih infrastruktur cloud (private, community, atau public). Di mana meskipun secara entitas mereka tetap berdiri sendiri-sendiri, tapi dihubungkan oleh suatu teknologi/mekanisme yang memungkinkan portabilitas data dan aplikasi antar cloud itu Misalnya, mekanisme load balancing yang antar cloud, sehingga alokasi sumberdaya bisa dipertahankan pada level yang optimal.



Gambar 6. Hybrid Cloud Computing
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Mekanisme Cloud Computing

Mekanisme akses ke cloud computing "mungkin" dapat dijalankan secara beraneka ragam mulai dari akses standar LAN maupun intranet dengan sedikit aplikasi agen atau klien, sampai kepada akses extranet dan internet melalui browser yang terhubung ke sebuah portal aplikasi dari penyedia layanan cloud computing. Protokol aplikasi yang digunakan pun dapat beragam, tetapi hal ini tidaklah terlalu signifikan bila dilihat dari sisi pengguna dimana pengguna akhir cukup mengetahui bagaimana cara mengakses dan mempergunakan jasa layanan yang terdapat pada Cloud computing.



Gambar 7. Struktur Cloud Computing

Sumber : http://id.wikipedia.org/wiki/Komputasi_awan

Manfaat Cloud Komputing

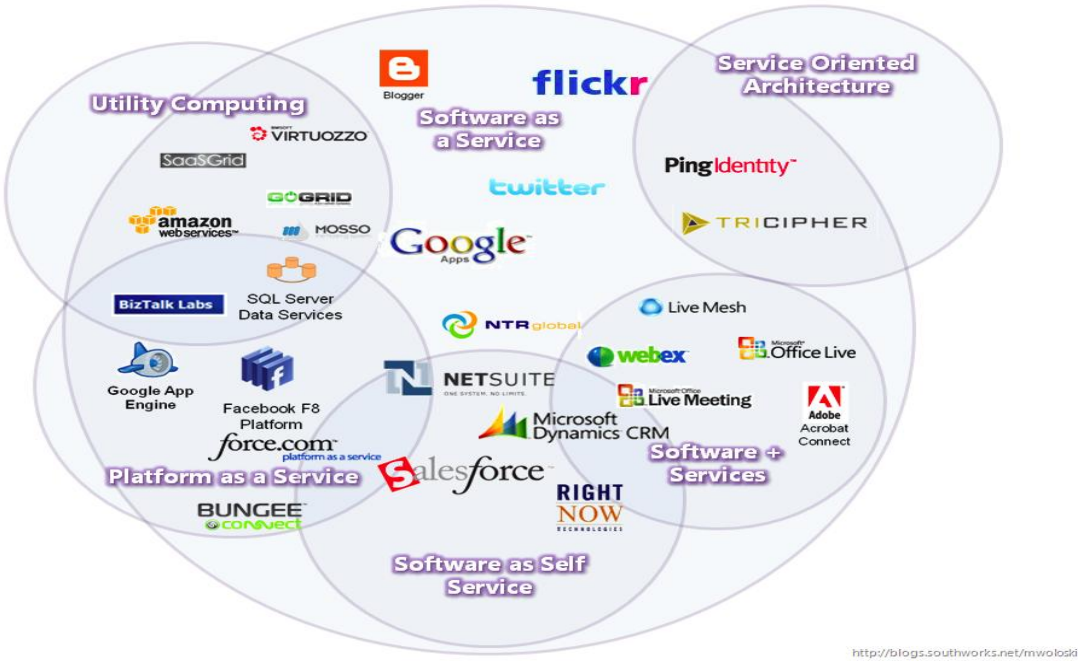
Dengan Cloud Computing kita tidak perlu lagi dikuatirkan dengan adanya kompleksitas Teknologi saat ini. Perusahaan dan organisasi yang dalam usahanya menggunakan Teknologi Informasi tidak perlu takut dengan hal-hal yang dapat mengancam keamanan sistem informasi mereka dan bahkan dalam hal peng-updatetan suatu Teknologi atau aplikasi yang dipakai , karena semuanya itu bisa diserahkan kepada penyedia layanan di Cloud Computing.

Cloud Computing jangan dijadikan sebagai —*Core Business*ll bagi sebuah perusahaan tapi sebaliknya jadikan-lah Cloud Computing ini sebagai —*Support Business*ll, prinsip ini yang benar karena Cloud Computing sebagai penunjang suatu perusahaan dalam mengelola sistem informasi yang ada di perusahaan tersebut dengan maksud dan tujuan untuk kelangsungan bisnis dari perusahaan tersebut, karena Cloud Computing memberikan solusi bagi perusahaan untuk meringankan operasional perusahaan tersebut dalam hal pengolahan data.

Ada beberapa keuntungan yang dapat dilihat dari perkembangan Cloud Computing ini, seperti :

1. Lebih efisien karena menggunakan anggaran yang rendah untuk sumber daya.
2. Membuat lebih eglity, dengan mudah dapat berorientasi pada profit dan perkembangan yang cepat.
3. Membuat operasional dan manajemen lebih mudah, dimungkinkan karena sistem pribadi atau perusahaan yang terkoneksi dalam satu cloud dapat dimonitor dan diatur dengan mudah.
4. Menjadikan koloborasi yang terpecaya dan lebih ramping.
5. Membantu dalam menekan biaya operasi biaya modal pada saat kita meningkatkan reliability dan kritikal sistem informasi yang kita bangun.

Seluruh nama besar seperti IBM, Microsoft, Google, dan Apple saat ini sedang terlibat dalam peperangan untuk menjadi penguasa terbesar terhadap cloud ini. Tentu saja masing-masing mengeluarkan jurusnya sendiri-sendiri.



Gambar 8. Berbagai Jenis Implementasi Layanan Internet

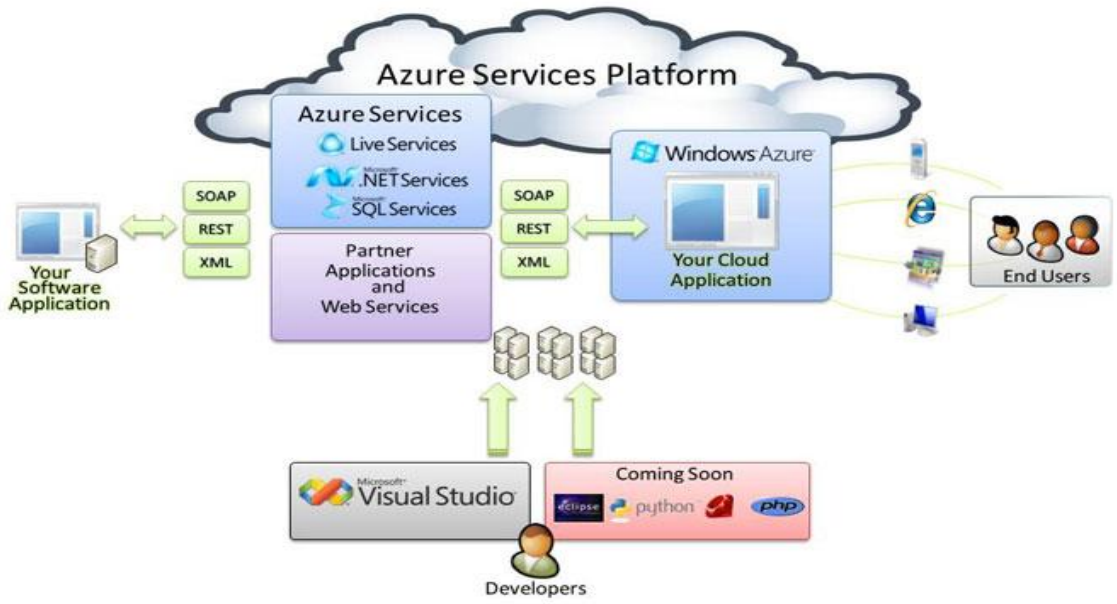
IBM di paruh akhir tahun 2009 kemarin telah meluncurkan LotusLive, layanan kolaborasi berbasis cloud.

Microsoft, yang sekarang di perkuat oleh Ray Ozzie sebagai Chief Software Architect pengganti Bill Gates, menggadang Windows Azure, sistem operasi berbasis cloud yang akan menjadi masa depan Windows OS.



Gambar 9. Windows Azure
Sumber: msdn.microsoft.com

Windows Azure merupakan sebuah sistem operasi untuk Cloud, namun sebelum melangkah lebih jauh, kita harus mengetahui apa yang dimaksud dengan Cloud itu sendiri. Cloud merupakan kumpulan dari server yang saling terhubung satu sama lain, dengan begitu performansi dari kumpulan server tersebut akan menigkat bila dibandingkan dengan single server biasa. Yang bisa dilakukan oleh para developer maupun IT Pro pada Cloud ini adalah seperti meng-install dan menjalankan service pada cloud environment, lalu kita juga bisa menyimpan dan mengambil data dari Cloud tersebut. Windows Azure, bukanlah sebuah sistem operasi yang akan menjadi pesaing Windows 7 , namun sistem operasi ini mencoba memberikan layanan bagi para developer mulai dari create, deploy dan pendistribusian webservice secara cepat , tentunya Sistem operasi ini berjalan dengan ada nya framewrok khusus "Azure Service Platform".



Gambar 10. Service Windows Azure
Sumber: msdn.microsoft.com

Apple mengambil sisi lain, telah menyediakan layanan Mobile Me yang memungkinkan pengguna produk Mac, untuk melakukan sinkronisasi data ke dalam cloud.



Gambar 11. Google Docs
Sumber : google.com 2011

Sementara Google, satu-satunya raksasa yang lahir di era internet, sudah sejak lama memberikan layanan Google Docs yang memungkinkan pengguna membuat dokumen atau bekerja dengan spreadsheet secara online tanpa perlu software terinstal di PC atau notebook. Jikalau dulu hanya berbentuk seperti Office online, sekarang Google Docs memberikan hard disk virtual untuk menyimpan data dalam berbagai macam format, video, audio, text, gambar, dan lain-lain. Hal ini dikarenakan Google ingin merajai dalam komputasi cloud (*cloud computing*) dan merespon tantangan Microsoft Office, Apple dan semua perusahaan-perusahaan lain yang mencoba bisnis ini.

Bahkan Google dalam waktu dekat akan meluncurkan sistem operasi cloud-nya, Chrome OS, yang akan menjadi ancaman serius bagi para penyedia sistem operasi lain.

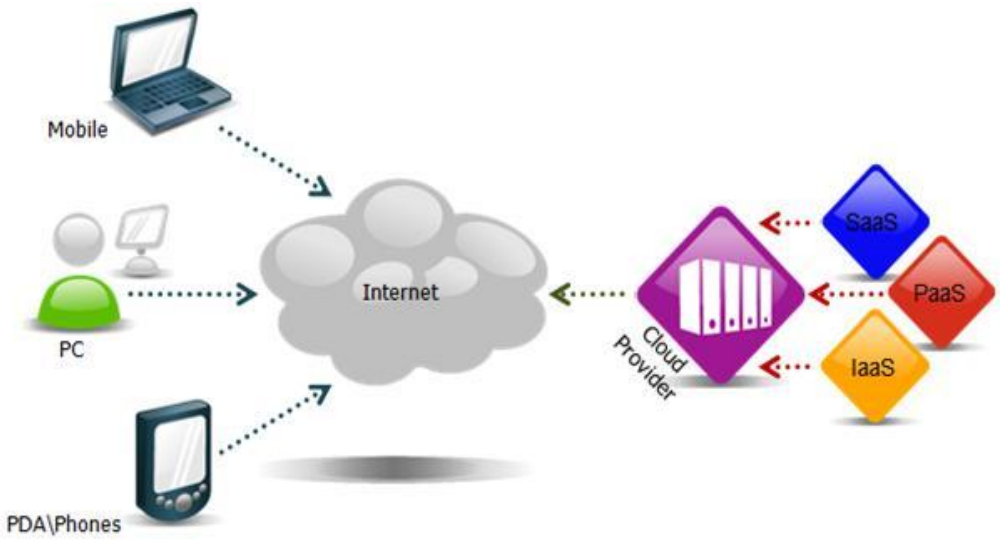
Namun bisa dibilang, keberhasilan Salesforce.com-lah yang membuka mata dunia bahwa cloud computing menjanjikan pundi-pundi emas yang menggiurkan.

BAGIAN 2

JENIS LAYANAN CLOUD

Layanan cloud memiliki tiga karakteristik khusus yang membedakannya dari hosting tradisional. Layanan ini dijual berdasarkan permintaan, yang biasanya per menit atau per jam dan bersifat elastis, user boleh memiliki berapapun layanan yang diinginkan sesuai waktu yang diberikan, dan layanan ini dikelolah penuh oleh provider (pelanggan hanya perlu komputer dan akses Internet). Inovasi-inovasi yang signifikan dalam hal virtualisasi dan distributed computing, termasuk juga peningkatan akses ke Internet berkecepatan tinggi dan perbaikan ekonomi, telah meningkatkan ketertarikan orang kepada cloud computing.

Sebuah cloud bisa berlabel privat atau publik. Public Cloud menjual layanan ke siapapun di internet. (Saat ini, Amazon Web Service merupakan provider public cloud terbesar.) Private Cloud adalah jaringan proprietary atau data center yang mensuplay layanan-layanan ter-host kepada orang-orang dalam jumlah terbatas. Jika sebuah service provider menggunakan sumber-sumber milik private cloud, maka hasilnya disebut virtual private cloud. Private atau publik, tujuan dari cloud computing adalah menyediakan akses yang mudah, skalabel kepada sumber-sumber komputasi dan layanan TI.



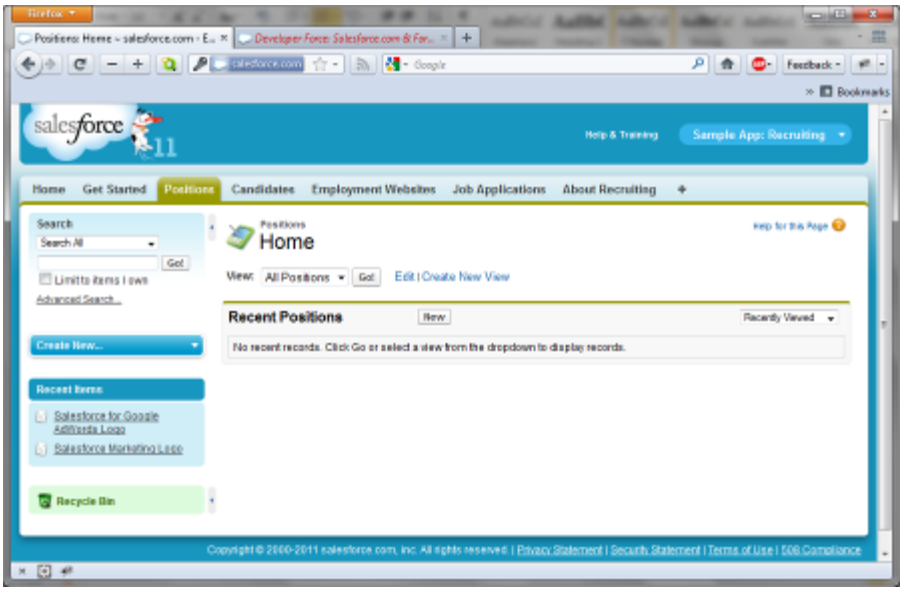
Gambar 12. Layanan Cloud
Sumber : http://id.wikipedia.org/wiki/Komputasi_awan

1. Software as a Service (SaaS)

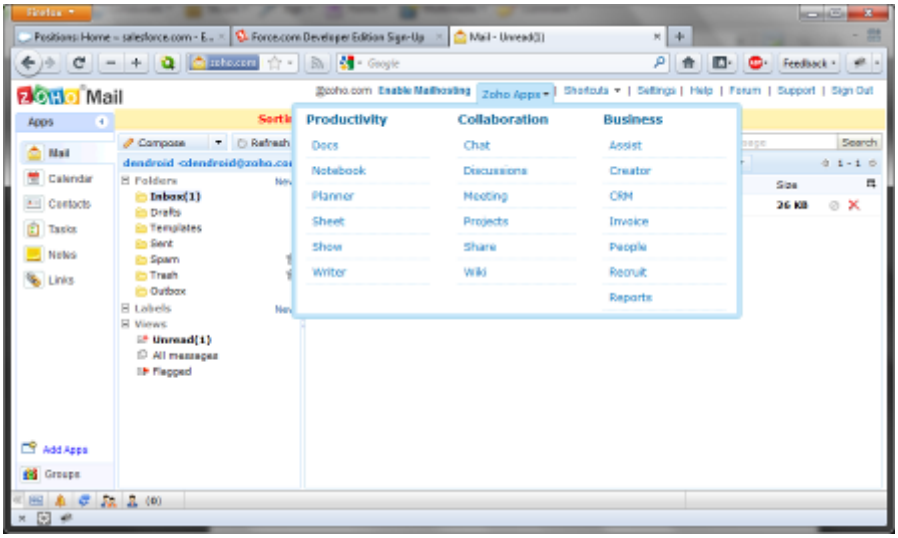
Model ini memberikan user sebuah aplikasi bisnis yang diakses melalui web. Umumnya user melakukan sewa aplikasi sehingga dapat mengakses fitur-fitur yang ada, user juga dapat membayar biaya tambahan untuk mengakses kapasitas / fitur yang lebih banyak. Dengan naiknya teknologi web seperti AJAX, memungkinkan web memiliki tingkat user experience yang mendekati desktop application.

Contoh model seperti ini sudah banyak :

- salesforce : Customer Relationship Management
- Yahoo : Email
- Google : Email, Google Doc
- Zoho : Collaboration Application



Gambar 13. SaaS SalesForce, dimana disini ditampilkan program recruitment
Sumber : http://salesforce.com



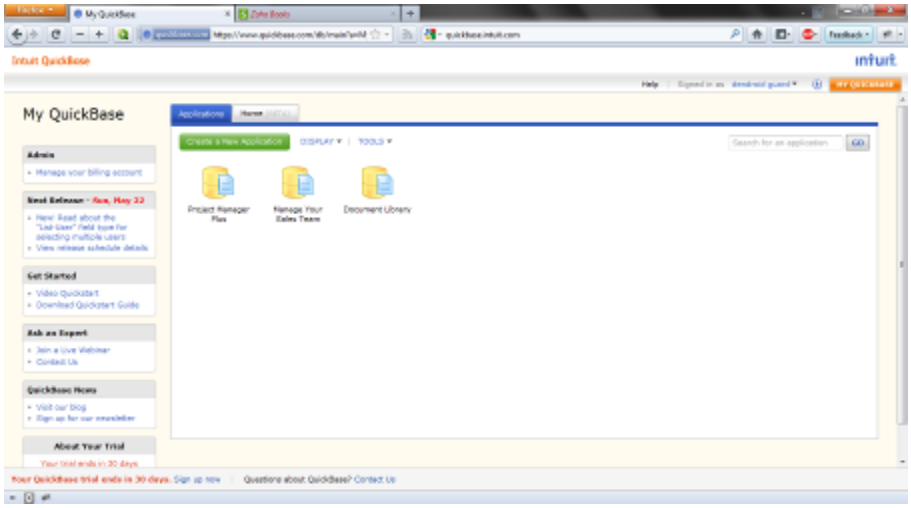
Gambar 14. SaaS ZOHO dimana ditampilkan daftar aplikasinya
Sumber : http://zoho.com

2. Platform as a Service (PaaS)

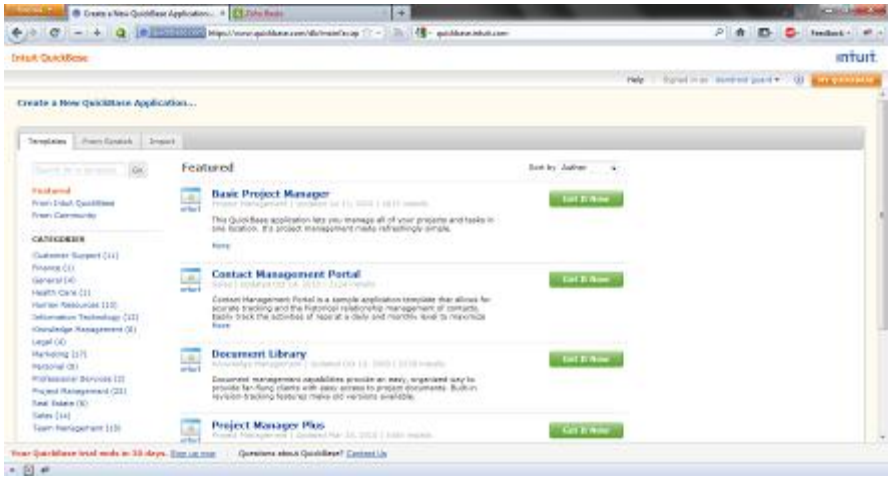
Disini, konsumen diberikan sebuah platform untuk pengembangan sampai implementasi sistem. Konsumen harus untuk membuat dan mengimplementasikan sistemnya sendiri. Umumnya tools untuk development disediakan dalam bentuk web application.

PaaS umumnya memiliki fitur sebagai berikut :

- Development tools berbasis browser internet
- Skalabilitas, access control, security, dan web service tersedia
- Integrasi yang mudah dengan aplikasi lain selama pada platform yang sama
- Tersedia connector untuk terhubung dengan sistem lain diluar komputasi cloud



Gambar 15. PaaS adalah QuickBase dimana kita dapat membuat applikasi



Gambar 16. vendor PaaS juga memberikan fasilitas SaaS

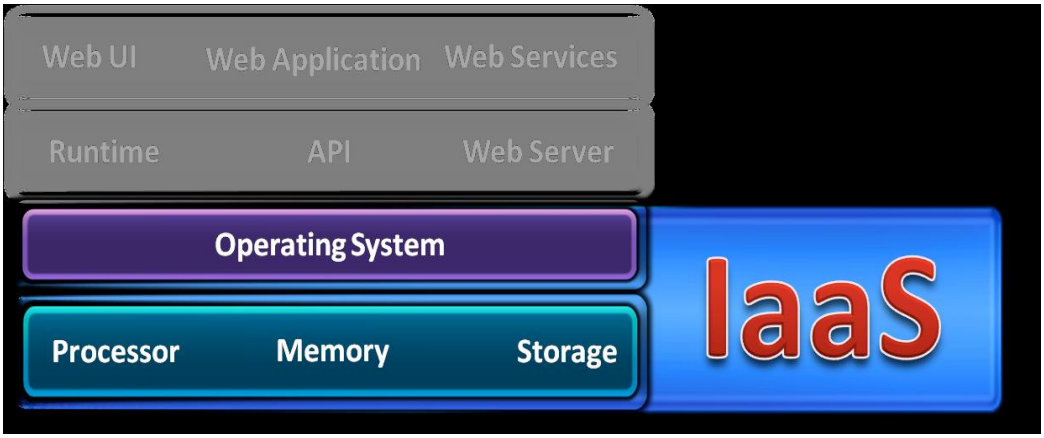
3. Infrastructure as a Service (IaaS)

Model ini hanya memberikan user aspek dasar dari computing seperti jaringan, storage, prosessor untuk computing. Infrastruktur komputasi cloud sangat bergantung pada *virtualisasi*. Untuk virtualisasi akan dijelaskan nanti pada bagian bawah.

IaaS umumnya memiliki fitur:

- memiliki pilihan virtual machine yang beragam, baik yang sama sekali kosong, memiliki OS preinstalled, bahka telah memiliki beberapa office productivity tools terinstall

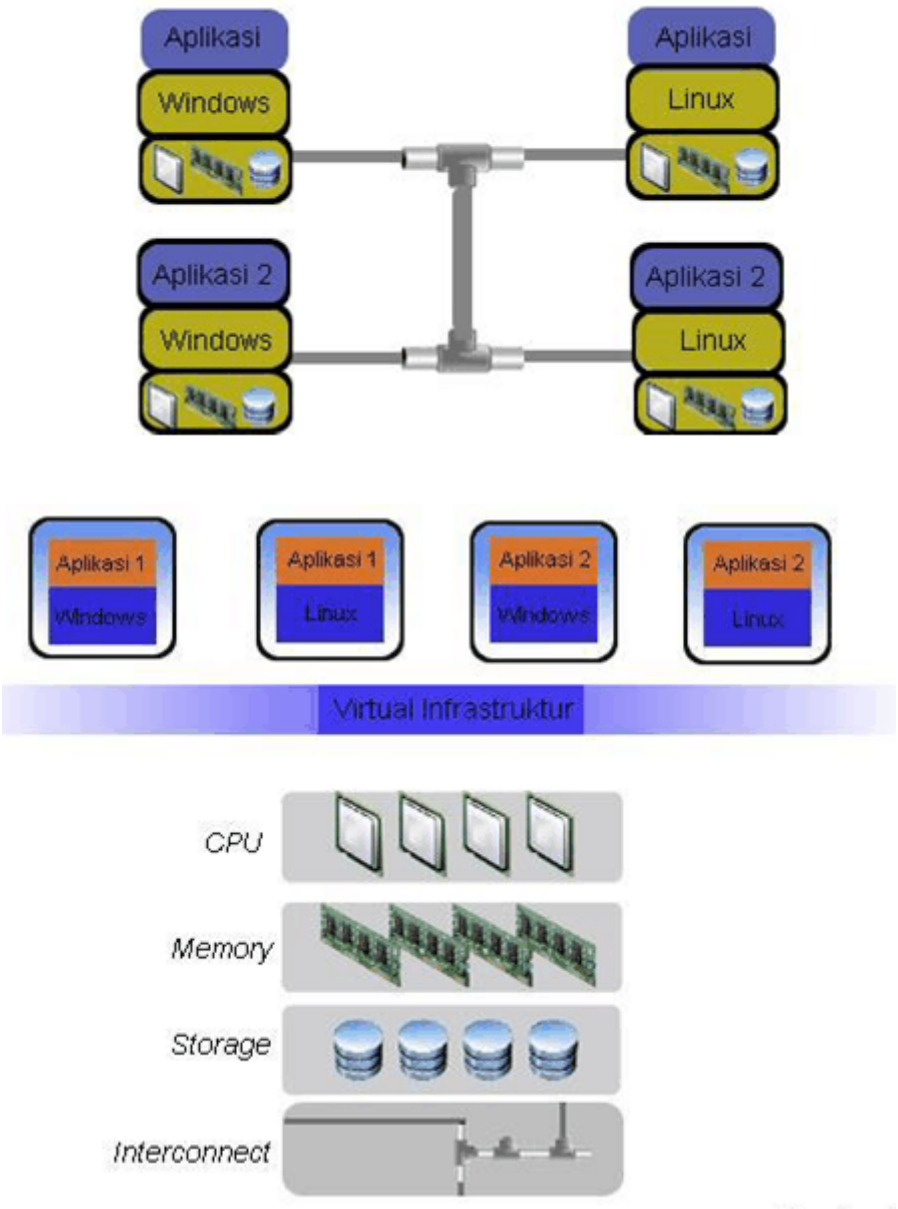
- kemampuan untuk meningkatkan atau menurunkan kemampuan computing baik secara manual atau otomatis (optimization)
- terdapat tools untuk memproses banyak data ataupun memproses aplikasi dengan perhitungan yang rumit
- dapat menyimpan data pada beberapa lokasi geografis fisik (memudahkan download)



Gambar 17. Infrastructure as a Service
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Utility Computing

Cloud computing tidak hanya melibatkan sisi aplikasi atau perangkat lunak saja, tetapi juga melibatkan perangkat keras atau *hardware* dan sumber daya penunjang. Seperti yang telah kita ketahui layanan *SaaS* lebih berfokus pada aplikasi atau perangkat lunak, sedangkan pada infrastruktur sebagai layanan *utility computing*. Layanan *utility computing* dikemas oleh provider dalam bentuk teknologi virtualisasi dan dikenal sebagai layanan *IaaS* (*Infrastructure as a Service*).



Gambar 18. Infrastructure Virtual

masing–masing aplikasi dan masing–masing sistem operasi (*windows* dan *linux*) menggunakan sumber daya komputer yang sama. Sistem operasi pada gambar tersebut bukanlah sesuatu yang special sebagai peranan utama dalam infrastruktur virtualisasi. Sistem operasi hanya sebagai perantara untuk dapat menjalankan *virtual* mesin. Peranan utama dalam infrastruktur virtualisasi adalah *hypervisor*. *Hypervisor* merupakan software yang menggantikan fungsi utama dari operating sistem ketika operating sistem selesai menjalankan *virtual* mesin. *Hypervisor* diasumsikan sebagai *virtual* machine manager, yang didesign untuk dapat menjalankan *virtual* mesin lainnya dan menjalankan sistem operasi dari awal seperti ketika komputer dinyalakan.

Dengan teknologi virtualisasi, pengguna atau penyewa IaaS dapat mengakses dan menggunakan seluruh sumber daya komputer dan seluruh sumber daya lainnya yang tersedia di dalam cloud sesuai kebutuhan dan keinginan pengguna.

Teknologi virtualisasi memungkinkan untuk diimplementasikan berbagai aplikasi dengan tujuan yang beragam dalam 1 *platform* atau aplikasi, seperti *storage computing, image manipulation, parallel processing, content distribution,* aplikasi web dan sebagainya.

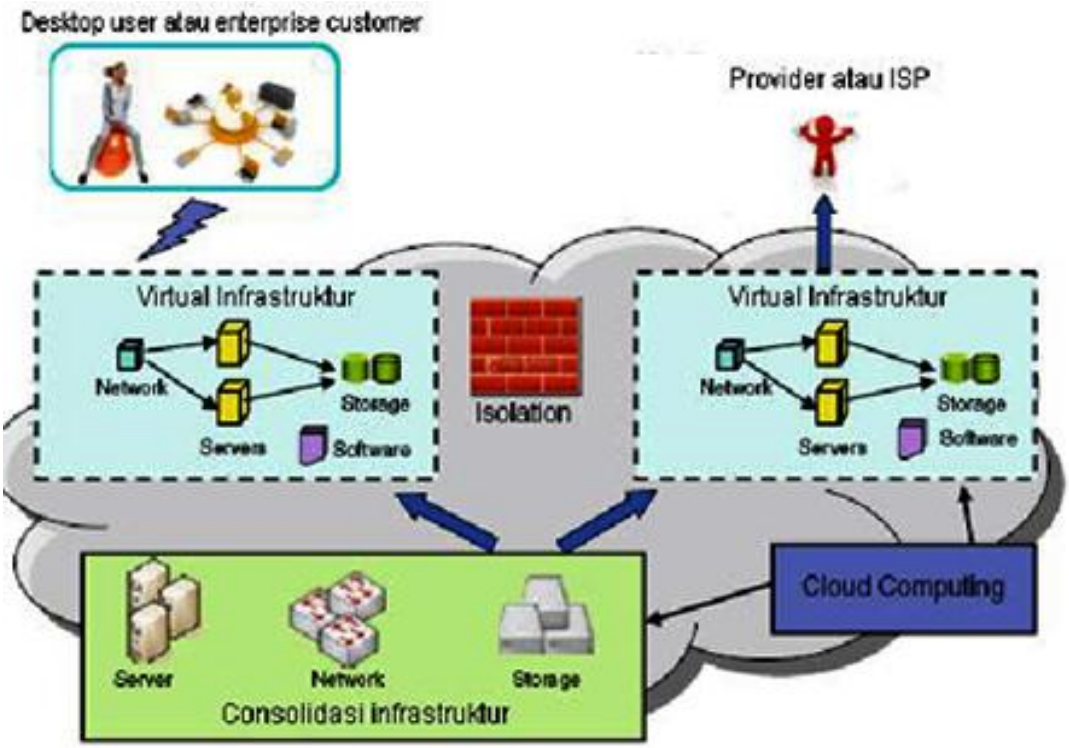
Dalam menawarkan layanan IaaS kepada pengguna atau penyewa, provider membagi IaaS dalam beberapa kategori layanan yaitu :

1. Layanan penyimpanan dan komputasi virtual : yaitu VMware rental, penyimpanan online (Online Storage).
2. Layanan kustomise : yaitu server template.
3. Layanan automasi dan control : yaitu automation.
4. Layanan penghubung : yaitu remote control, web 2.0.
5. Layanan monitoring : yaitu monitor secara fisik objek yang diinginkan (posisi koordinat bumi, peta, kamera).
6. Layanan optimasi objek : yaitu virtualisasi network, virtualisasi penyimpanan, virtualisasi server.
7. Layanan pengukuran objek : yaitu pengukuran fisik suatu objek.
8. Layanan integrated dan kombinasi objek : yaitu load balance.

9. Layanan security : yaitu enkripsi data penyimpanan, *VM isolation*, *VLAN* dan *SSL/SSH.*

Secara infrastruktur, penerapan teknologi virtualisasi pada *IaaS* di cloud computing memberikan beberapa kemudahan & keuntungan bagi penyewa.

Jantung dari teknologi *cloud computing* adalah *virtualisasi*, dimana virtualisasi dapat diterapkan pada 2 sisi yaitu pada sisi *provider* dan sisi pengguna seperti pada gambar



Gambar 19. Desktop Pengguna

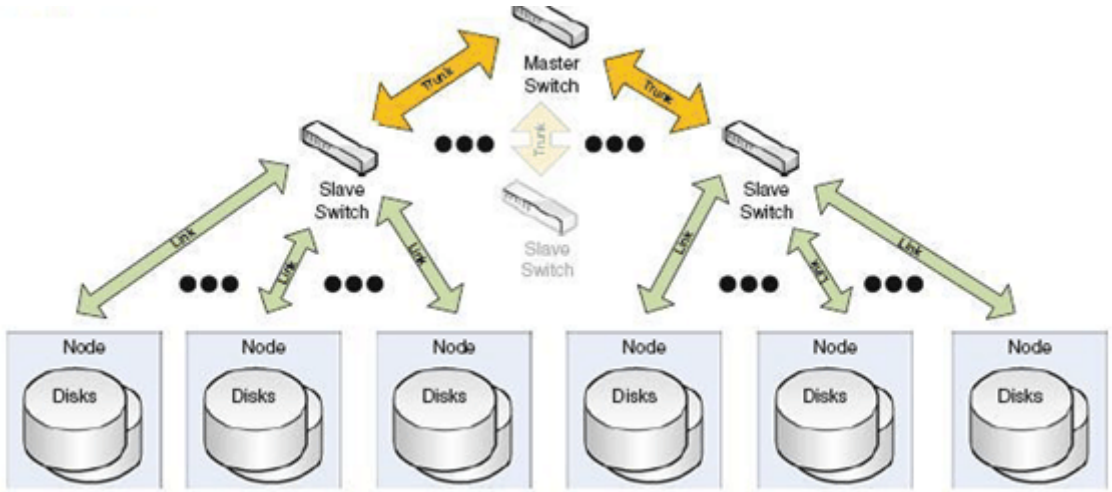
Beberapa software virtulisasi seperti VMware, citrix dan sebagainya mempunyai kemampuan untuk menciptakan fungsi lain yang disebut sebagai *virtual desktop interface* (*VDI*). *Virtual desktop interface* (*VDI*) menciptakan session untuk *client* atau *user* di dalam server, dan mengirimkan *virtual PC* tersebut kepada *client* atau *user* sehingga user dapat berinteraksi dengan server seakan client atau user tersebut berada di dalam server itu sendiri. Perbedaan yang cukup signifikan antara software remote dengan virtual PC :

- Software remote adalah software yang dapat digunakan untuk melakukan pengendalian jarak jauh ke satu komputer atau satu server dalam satu koneksi

hanya untuk satu user atau client. Jika satu komputer atau satu server diakses oleh lebih dari dua user maka komputer atau server yang diakses secara remote akan memutuskan salah satu koneksi dari dua koneksi yang terjadi.

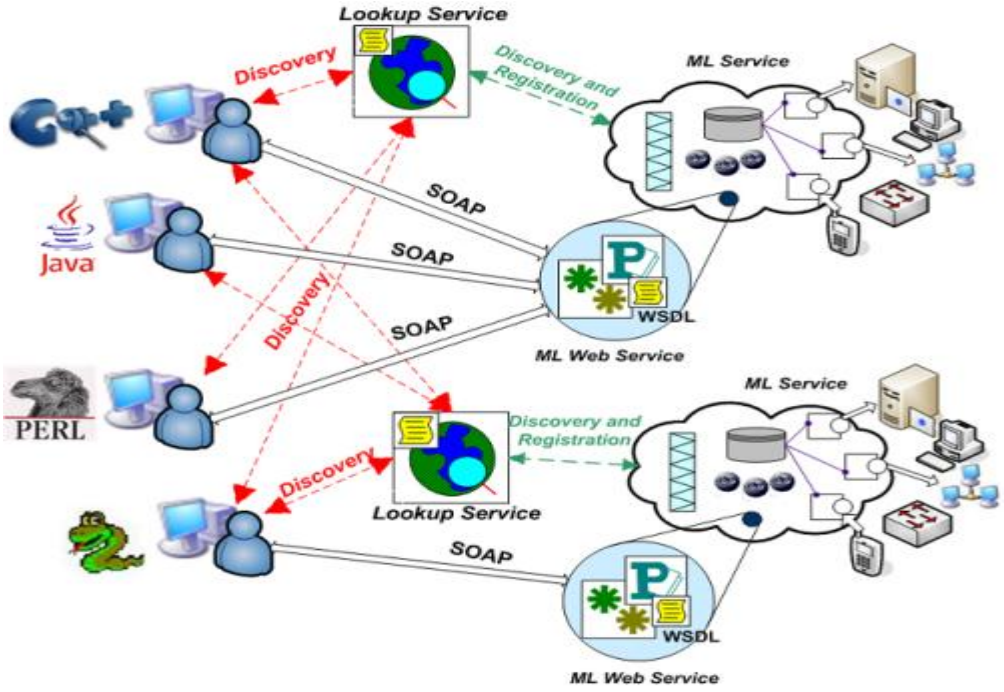
Software remote hanya software atau aplikasi penghubung ke komputer lain dan tidak dapat berfungsi untuk menciptakan komputer di dalam komputer itu sendiri. user terkoneksi dan menggunakan layanan IaaS ke server provider melalui virtual desktop interface (VDI) di internet. Sedangkan pada sisi provider, provider melakukan konfigurasi server melalui jalur yang sama (VDI) di internet. Untuk dapat menerapkan teknologi virtualisasi di *cloud* maka server yang sudah diimplementasikan teknologi virtualisasi diletakkan di dalam *cloud* (*private cloud* atau *public cloud*) sebagai *back end* infrastruktur. Dari prespektif ini, sumber daya teknologi virtualisasi atau *virtual resources* di dalam cloud diasumsikan sebagai sumber daya komputer yang bersifat independent atau mandiri termasuk lokasi dari sumber daya itu sendiri.

Infrastruktur juga memegang peranan utama untuk memastikan semua komponen bekerja dengan baik dalam kondisi multi tenant dan bertanggung jawab terhadap segala aktifitas yang terjadi. Seperti yang sudah dijelaskan sebelumnya bahwa teknologi virtualisasi merupakan jantung utama dari cloud computing, dimana teknologi virtualisasi hanyalah berupa aplikasi atau *software*. Teknologi virtualisasi tidak dapat berjalan sempurna tanpa didukung dengan infrastruktur yang baik dan solid. Teknologi virtualisasi memungkinkan untuk diterapkan *redundancy*, *replication* atau *cluster*, dan *workload balancing*.



Gambar 20. Ilustrasi infrastruktur yang baik dan solid

Kemampuan unik dari *web service* adalah membantu para programmer untuk membuat suatu aplikasi berbasis *web* dengan fungsi lain di atas *platform web* itu sendiri. Dalam beberapa kasus, *coding* – *coding* yang dihasilkan oleh programer yang menyewa layanan ini membagikan (share) dan dikumpulkan dalam penyimpanan data yang dikelola oleh *provider*.



Gambar 20. Ilustrasi Layanan web service
Sumber : http://id.wikipedia.org/wiki

Web Service merupakan fenomena yang sangat panas saat ini karena, banyak kelebihan yang ditawarkan oleh Web Service terutama interoperabilitas tinggi dan penggunaannya yang dapat diakses kapanpun dan dimanapun selama mesin kita terhubung oleh jaringan internet salah satunya.

Pengertian Web Service Sepenuhnya berdasarkan standard web dan xml. Web Service dapat membantu: Perantara pada integrasi platform sepanjang eksekusi mesin virtual. Integrasi antara Web dan OO middleware. Integrasi dari aliran kerja terisolasi dan sevice-service (Web Services Flow Language -WSFL). Pertukaran data pada

aplikasi yang berbeda-beda (X-Schema, XSLT ++) (Masa depan: standarisasi dari info konteks antara web servis dan klien – integrasi servis horizontal).

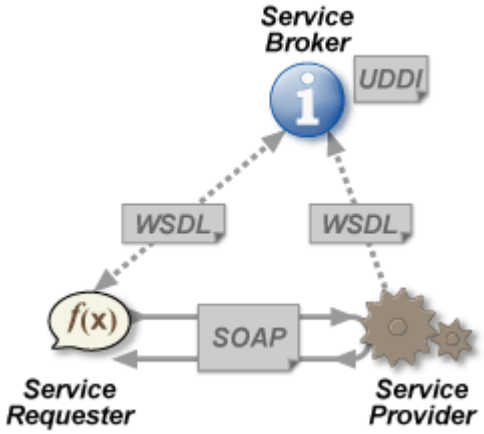
Pemain utama dan standard-standard :
Microsoft: .NET SUN: Open Net Environment (ONE) IBM: Web Service Conceptual Architecture (WSCA) W3C: Web Service Workshop Oracle: Web Service Broker Hewlett-Packard: Web Service Platform Kemampuan aplikasi, fungsi atau operasi yang di ekspos untuk program lain melalui standard yang terbuka, dan interoperable. "payloads" didefinisikan sebagai XML. "transports" melalui http atau Internet protocol terbuka lainnya. Data diakses dari berbagai bahasa pemrograman , platform hardware atau system operasi. Middleware dari Internet.

Keuntungan penggunaan Web Service

- Format penggunaan terbuka untuk semua platform.
- Mudah di mengerti dan mudah men-debug.
- Dukungan interface yang stabil.
- Menggunakan standard-standard "membuka service sekali" dan mempunyai pemakai banyak.
- Mudah untuk menengahi pesan-pesan proses dan menambahkan nilai. Routing and pengiriman.
- Security.
- management and monitoring.
- schema and service design.
- Akselerasi.
- mudah untuk mengembangkan dengan semantic transport tambahan.
- Terbuka, standard-standard berbasis teks.
- Pencapaian modular.
- Tidak mahal untuk diimplementasikan (relatif).
- Mengurangi biaya integrasi aplikasi enterprise.
- Implementasi yang incremental.

Mekanisme Kerja

Sebuah **layanan Web** adalah metode komunikasi antara dua perangkat elektronik melalui jaringan.



Gambar 21. Web Arsitektur
Sumber : http://id.wikipedia.org/wiki

The W3C mendefinisikan "layanan Web" sebagai "sebuah sistem perangkat lunak yang dirancang untuk mendukung interoperable mesin-ke-mesin interaksi melalui jaringan Bahasa. ini memiliki antarmuka yang dijelaskan dalam mesin-processable format (khusus Web Services Deskripsi WSDL). Sistem lain berinteraksi dengan layanan Web dalam cara yang ditentukan oleh deskripsi dengan menggunakan SOAP pesan, biasanya disampaikan menggunakan HTTP dengan serialisasi XML dalam hubungannya dengan standar Web-terkait lainnya. "

W3C juga menyatakan, "Kita dapat mengidentifikasi dua kelompok utama layanan Web, REST layanan Web-compliant, di mana tujuan utama pelayanan ini adalah untuk memanipulasi representasi sumber daya XML Web menggunakan seragam set "stateless" operasi; dan sewenang-wenang layanan Web, di mana layanan akan mengekspos serangkaian operasi sewenang-wenang. "

Big layanan Web

"Big layanan Web" menggunakan Extensible Markup Language (XML) pesan yang mengikuti SOAP standard dan telah populer dengan usaha tradisional. Dalam sistem seperti itu, sering kali ada yang bisa dibaca deskripsi mesin operasi yang ditawarkan oleh layanan ditulis dalam Web Services Description Language (WSDL). Yang terakhir

adalah tidak merupakan persyaratan dari sebuah *titik akhir* SOAP, tetapi merupakan prasyarat untuk otomatis sisi klien. SOAP kerangka (kerangka kerja seperti Apache axis2 , Apache CXF , dan Spring menjadi pengecualian). Beberapa organisasi industri, seperti WS-I , mandat baik SOAP dan WSDL dalam definisi mereka tentang layanan Web.

Web API



Gambar 22. Layanan web dalam service-oriented architecture
Sumber : http://id.wikipedia.org/wiki

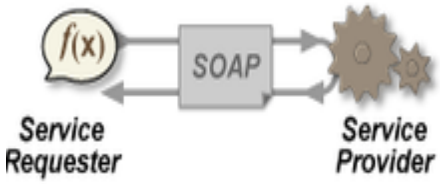
Web API adalah pembangunan di layanan Web (dalam gerakan yang disebut Web 2.0) dimana penekanan telah bergerak menjauh dari SOAP layanan berbasis terhadap Transfer (REST) komunikasi berbasis. REST tidak memerlukan layanan XML, SOAP, atau WSDL layanan-API definisi. Web API memungkinkan kombinasi dari berbagai layanan web ke dalam aplikasi baru yang dikenal sebagai mashup .

Ketika digunakan dalam konteks pengembangan Web , Web API ini biasanya kelompok yang telah ditetapkan dari Hypertext Transfer Protocol (HTTP) pesan permintaan bersama dengan definisi struktur pesan respon, biasanya dinyatakan dalam sebuah Extensible Markup Language (XML) atau JavaScript Object Notation (JSON) format.

Saat menjalankan layanan Web komposit, setiap layanan sub dapat dianggap otonom. User tidak memiliki kendali atas layanan ini. Juga Web services sendiri tidak dapat

diandalkan, penyedia layanan dapat menghapus, mengubah atau memperbarui jasa mereka tanpa memberikan pemberitahuan kepada pengguna. Toleransi kehandalan dan kesalahan tidak didukung; kesalahan mungkin terjadi selama eksekusi. penanganan Eksepsi dalam konteks layanan Web adalah masih merupakan masalah dalam penelitian terbuka. Tetap saja dapat ditangani dengan menanggapi dengan obyek kesalahan ke klien.

Prosedur Remote Panggilan



Gambar 23. Arsitektur elemen yang terlibat dalam XML-RPC.
Sumber : http://id.wikipedia.org/wiki

RPC Web layanan menyajikan fungsi didistribusikan (atau metode) antarmuka panggilan yang akrab bagi banyak pengembang. Biasanya, unit dasar layanan RPC Web adalah operasi WSDL.

Web pertama pelayanan alat difokuskan pada RPC, dan sebagai hasilnya gaya ini secara luas digunakan dan didukung. Namun, kadang-kadang dikritik karena tidak longgar ditambah, karena sering dilaksanakan oleh jasa pemetaan langsung ke-spesifik fungsi bahasa atau panggilan metode. Banyak vendor merasa pendekatan ini menjadi buntu, dan mendorong untuk RPC untuk menjadi batasan dalam WS-I Basic Profile .

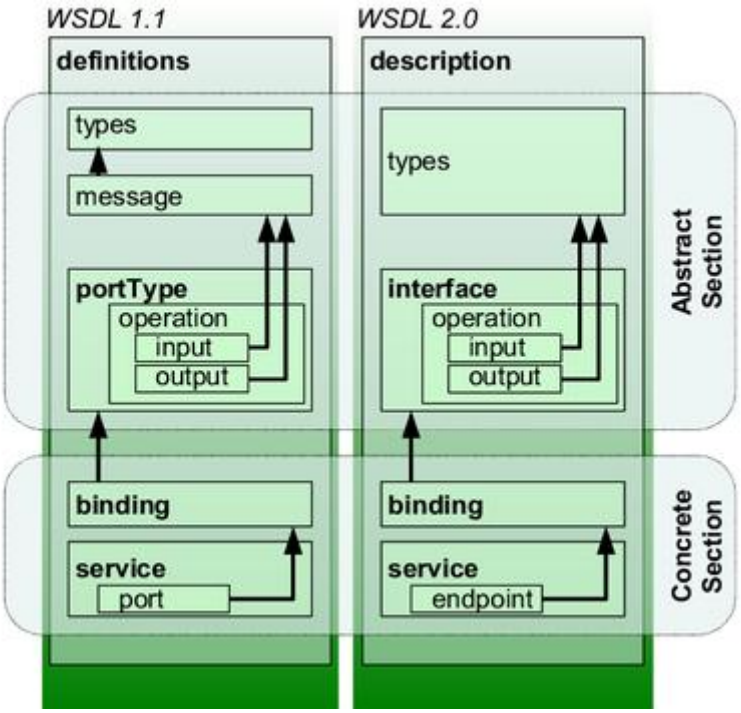
Pendekatan-pendekatan lain dengan hampir fungsi yang sama seperti RPC adalah Object Management Group 's (OMG) Common Object Request Broker Architecture (CORBA), Microsoft 's Distributed Component Object Model (DCOM) atau Sun Microsystems 's Java/Remote Method Invocation (RMI).

Arsitektur Berorientasi Layanan

Layanan Web juga dapat digunakan untuk mengimplementasikan arsitektur sesuai dengan konsep service-oriented architecture (SOA), di mana unit dasar komunikasi adalah pesan, bukan operasi. Hal ini sering disebut sebagai " pesan-berorientasi

"layanan. SOA Web layanan yang didukung oleh sebagian besar vendor software utama dan analis industri. Tidak seperti layanan Web RPC, kopling longgar lebih mungkin, karena fokusnya adalah pada "kontrak" yang WSDL menyediakan, bukan detail implementasi yang mendasarinya.

Middleware analis menggunakan bus pelayanan perusahaan yang menggabungkan pesan-berorientasi pengolahan dan layanan Web untuk membuat event-driven SOA . Salah satu contoh sumber terbuka ESB adalah Mule , satu lagi adalah Open ESB .



Gambar 24. Representasi konsep didefinisikan oleh WSDL 1.1 dan WSDL 2.0 dokumen. Sumber : http://id.wikipedia.org/wiki

Representasi Transfer (Rest)

REST berusaha untuk menggambarkan arsitektur yang menggunakan protokol HTTP atau serupa oleh menghambat antarmuka untuk satu set terkenal, operasi standar (seperti GET, POST, PUT, DELETE untuk HTTP). Di sini, fokusnya adalah pada berinteraksi dengan stateful sumber daya, daripada pesan atau operasi.

Sebuah arsitektur yang berbasis pada REST dapat menggunakan WSDL untuk menggambarkan pesan SOAP melalui HTTP, dapat diimplementasikan sebagai sebuah abstraksi murni di atas SOAP (misalnya, WS-Transfer), atau dapat dibuat tanpa menggunakan SOAP di semua.

WSDL versi 2.0 menawarkan dukungan untuk mengikat kepada semua metode permintaan HTTP (tidak hanya GET dan POST seperti pada versi 1.1) sehingga memungkinkan implementasi yang lebih baik dari layanan Web tenang. Namun, dukungan untuk spesifikasi ini masih miskin dalam pengembangan perangkat lunak kit yang sering menawarkan alat hanya untuk WSDL 1.1.

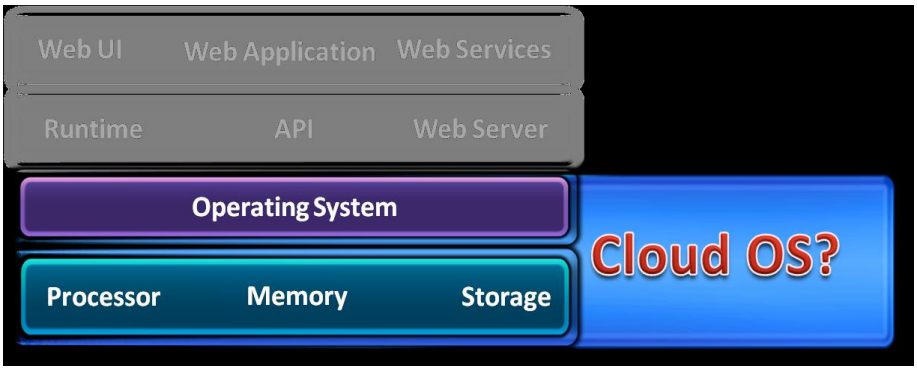
BAGIAN 3

IMPLEMENTASI CLOUD COMPUTING

Salah satu Operating System (OS) yang sudah menerapkan teknologi dari Cloud Computing ini adalah Chromium OS dari Google. Dimana kita menjalan OS ini secara online, tidak perlu lagi menginstall di lokal PC kita. Tidak membutuhkan hardisk atau media penyimpanan lagi. Semua program akan dijalankan secara online. Yang terpenting adalah koneksi internet yang cepat.

Pada Perangkat keras dan Sistem Operasi (OS) sebagai Web Service melalui Internet publik. Berdasarkan prinsip web service, kita dapat mengirim permintaan untuk layanan ini bersama dengan beberapa parameter karena OS diharapkan bertindak sebagai interface ke CPU dan perangkat, kita dapat memangil layanan yang menerima "pekerjaan" yang harus diproses oleh OS dan hardware.

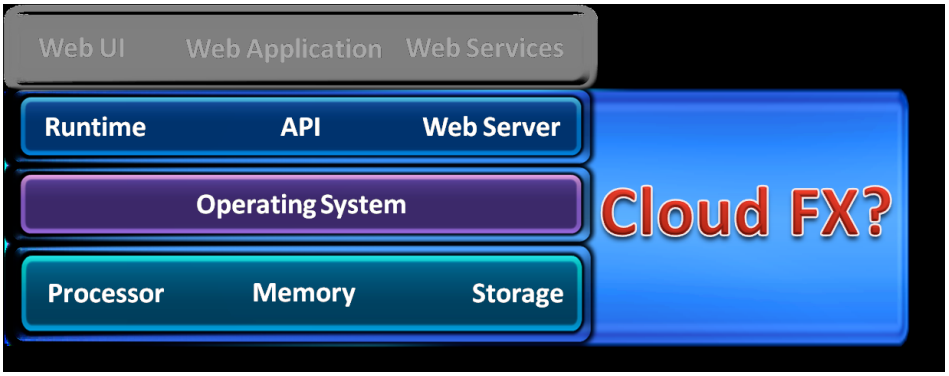
Secara teknis, Web Service baru saja berubah OS + H/W kombinasi menjadi "Layanan" Kita bisa memulai layanan mengkonsumsi ini dengan mengirimkan tugas CPU secara intensif ini adalah geerasi baru web service.



Gambar. 25. Mengekspos hardware dan OS sebagai Layanan
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Cloud FX

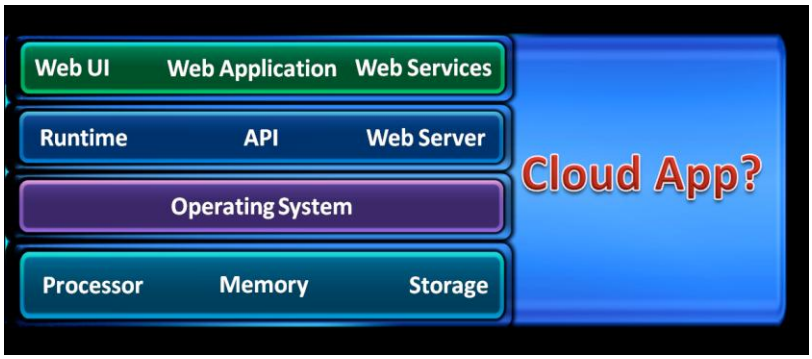
Pengembang selalu mengembangkan dan menyebarkan aplikasi pada platform pengembangan. Beberapa pengembangan aplikasi yang paling populer platform .NET dan Java. Dalam perkembangan terakhir, kita telah melihat bagaimana OS + H/W kombinasi yang di tawarkan sebagai layanan.



Gambar. 26. Exposing the Runtime + SDK as a Service
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Cloud Aplikasi

Saat ini, sebagian aplikasi desktop tradisional seperti pengolah kata dan paket spreadsheet yang tersedia melalui web. Ini generasi baru aplikasi hanya perlu browser. Hal ini fundamental mengubah cara perangkat lunak ini digunakan dan berlisensi. Anda tidak perlu klik dua kali setup.exe untuk menginstal office suite pada desktop anda. Berlangganan aplikasi dan fitur yang anda butuhkan dan hanya membayar yang ada gunakan. Hal hampir setara dengan mengekspos aplikasi sebagai apliaksi service. Dan disebut juga sebagai aplikasi cloud.

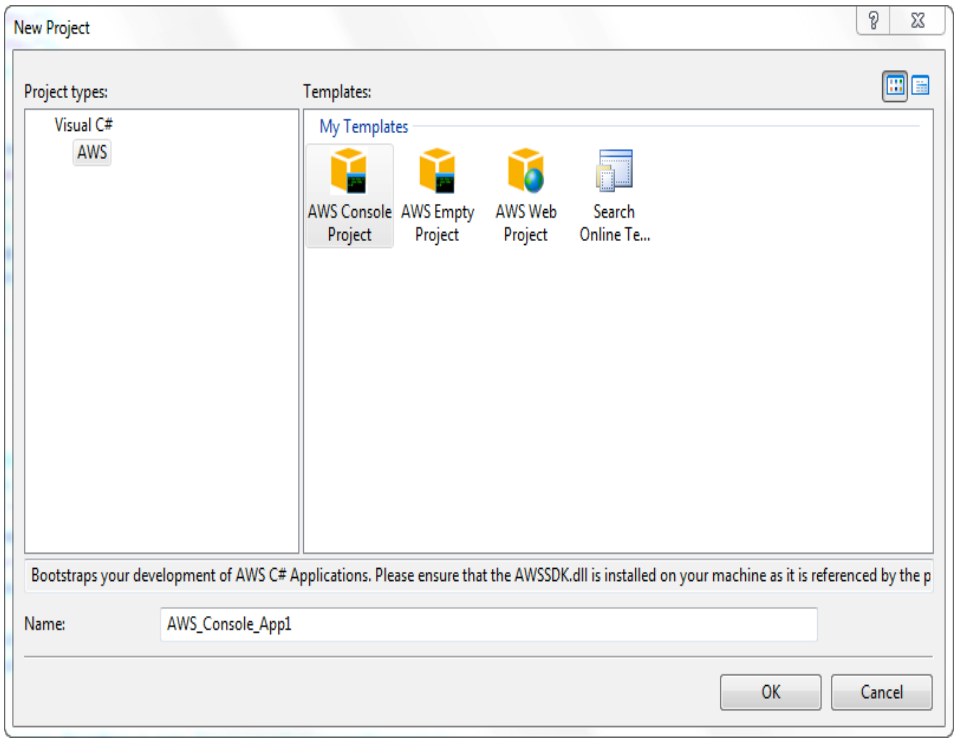


Gambar 27. Web App as a Service
Sumber : Janakiram MSV Cloud Computing Strategist 2010

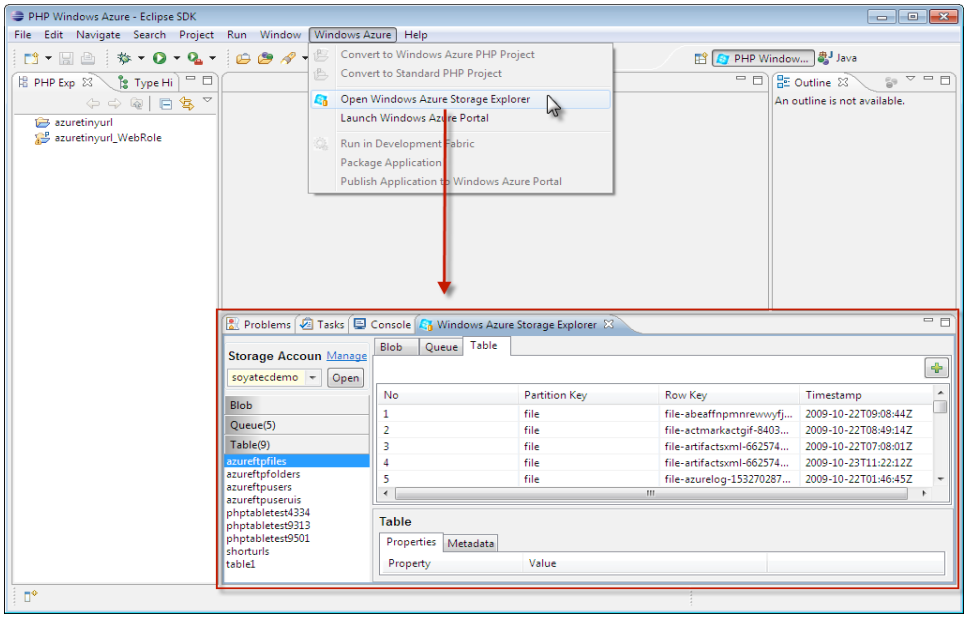
Programabilitas

Ini adalah parameter kritis dari Cloud, membuat pengembang sangat penting untuk konsep-konsep multi treading dimana mereka memerlukan thread baru untuk mencapai skalabilitas dan aplikasi.

Amazon web service memiliki API yang paling matang untuk mengendalikan sumber daya pemrograman berbasis cloud. Dengan memanfaatkan API, pengembang aplikasi untuk mengelola infrastruktur dan beberapa front-end berjalan di iPhonedan Windows Mobile seperti dalam gambar dibawah ini.



Gambar 28. AWS SDK for .NET
Sumber : amazon.com



Gambar 29. Azure Tools for Eclipse
Sumber: msdn.microsoft.com



Gambar 30. iPhone App to manage AWS
Sumber :amazon.com

Jadi, Cloud Computing memiliki 4 prinsip kunci, 1) Elastisitas, 2) Bayar-by- Gunakan, 3) Self Service, 4) Programmability.

Hosting vs Cloud Computing

Menyingkap perdebatan yang sedang berlangsung Computing Cloud vs hosting, mari kita lihat model atribut hosting yang mengekspos. Hosting tidak pernah dapat memenuhi janji elastisitas. Bahkan jika tidak, tidak akan cocok dengan ekonomi cloud. Hosting tidak menawarkan beberapa tingkat Self service tetapi tidak sampai batas menanipulasi server.

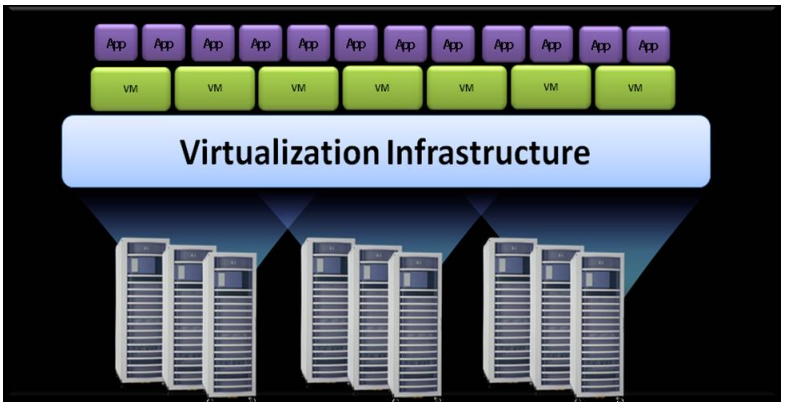
konfigurasi on the fly! Bayar-Dengan-Gunakan atribut ini ditiru oleh beberapa hosting perusahaan. Tapi, itu bukan norma dalam bisnis hosting. Programabilitas terlalu mahal untuk didukung oleh hasters karena mereka tidak bias berinventaris dalam SDK dan alat untuk mengelola infrastruktur. Jadi terbukti bahwa hosting tidak sama dengan Cloud Computing.

Kenyataannya adalah bahwa kemampuan ini dapat diterapkan ke pusat data dan disebut juga sebagai Private Cloud. Bisa kita lihat 4 mekanisme yang berbeda pada cloud dan hosting seperti pada gambar dibawah ini.

Attribute	Cloud Computing	Web Hosting
Elasticity		
Pay-By-Use		
Self Service		
Programmability		

Gambar 30. The 4 Implementations of the Cloud
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Virtualisasi adalah abstrak perangkat keras untuk menjalankan beberapa virtual system operasi pada system host. Kita dapat melihat virtualisasi dengan menginstal Microsoft Virtual PC, VMware Player atau Sun VirtualBox. Ini solusi virtualisasi desktop yang memungkinkan mengisntal dan menjalakan OS dalam OS host. Seperti dalam gambar beikut ini.



Gambar 31. Server Virtualization
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Ada banyak alasan untuk menjalankan Virtualisasi pada :

Mean Time to Restore

Hal ini jauh lebih fleksibel dan lebih cepat untuk mengembalikan web server gagal, app server atau database server yang menjalankan sebagai virtual. Karena yang fisik file pada hard disk untuk sistem operasi, hanya menyalin replika dari gambar server gagal jadi lebih cepat dari pada mengembalikan sebuah server fisik gagal.

Memaksimalkan utilisasi server

Hal ini sangat umum bahwa server tertentu di pusat data kurang dimanfaatkan beberapa server maxed out. Melalui virtualisasi, beban dapat merata di semua server.

Pengurangan dalam biaya pemeliharaan

Virtualisasi memiliki dampak langsung pada bottom line. Pertama, dengan mengkonsolidakan data center untuk berjalan pada server yang lebih sedikit tapi kuat. Daya yang dikonsumsi oleh pusat data dan biaya pemeliharaan peralatan pendigin turun drastis.

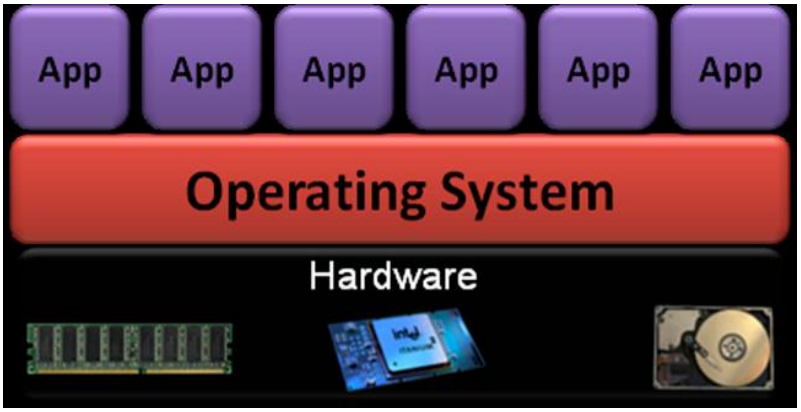
Ketika hardware mencapai akhir, server fisik perlu diganti. Back up dan memulihkan data dan instalasi perangkat lunak pada server produksi sangat kompleks dan mahal. Dengan virtualisasi membuat proses ini sangat sederhana dan biaya lebih efektif. Server fisik akan diganti dan VMs hanya cukup di restart tanpa perubahan apapun dalam konfigurasi. Hal ini memilki banyak dampak pada anggaran IT.

Efisien Manajemen

Semua software virtualisasi utama memiliki konsol terpusat untuk mengelola, memelihara, melacak dan memantau kesehatan server fisikdan VMs berjalan pada server. Karena kesederhanaan dan kemampuan dinamis, administrator IT akan menghabiskan sedikit waktu dalam mengelola infrastruktur.

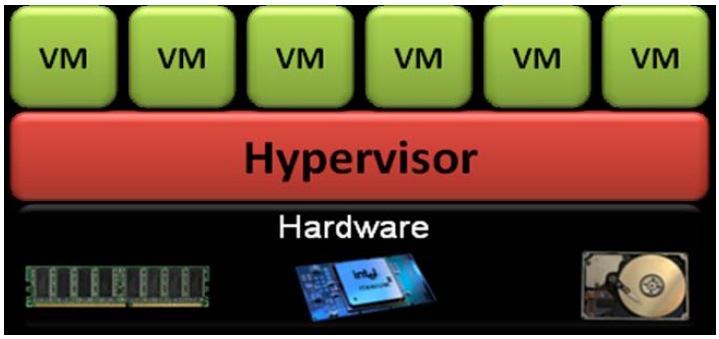
Virtualisasi pada Server

Mari kita memahami lebih lanjut tentang virtualisasi server. Biasanya OS dirancang untuk bertindak sebagai antarmuka antara aplikasi dan hardware. Hal ini tidak dirancang khusus untuk menjalankan OS.



Gambar 32. OS manages the applications
Sumber : Janakiram MSV Cloud Computing Strategist 2010

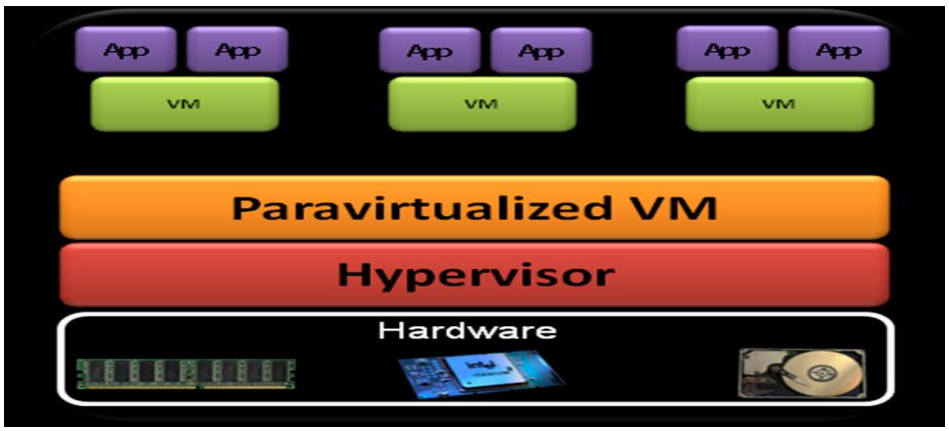
Bahkan, dalam virtualisasi server, OS host tidak terlalu signifikan. hal ini hanya terbatas pada boot up dan VMs. Mengingat fakta bahwa OS tidak ideal untuk menjalakan beberapa VMs dan memiliki peran kecil untuk bermain maka muncul generasi perangkat lunak yang disebut Hypervisor yang mengambil ahli OS. Hypervisor adalah virtual efisien Mesin Manager (VMM) dirancang dari bawah ke atas untuk menjalankan beberapa performant VMs.



Gambar 33. Hypervisor berpotensi dapat menggantikan OS dan bahkan dapat bootlangsung dari VM. Sumber : Janakiram MSV Cloud Computing Strategist 2010

Hypervisor hanya memilki daya tampung megabyte (vmWare ESXi hanya 32MB) dan memiliki OS. Hypervisors dibantu oleh virtualisasi hardware fitur dibangun oleh Intel dan AMD CPU. Kombinasi dari perngkat keras ini ternyata Hypervisor server menjadi mesin ramping dan untuk beberapa host VMs.

VM yang digunakan oleh Hypervisor untuk boot sebagai home disebut paravirtualized VM. Konsep ini membuat virtualisasi benar-benar kuat, menjadi sangat dinamis dan licah. Hypervisor dapat dikontrol oleh pusat konsol dan dapat diinstruksikan oleh VM host untuk boot dan VMs guest untuk dapat berjalan di atasnya.



Gambar 34. Bare Metal Virtualization
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Virtualisasi dan Cloud

Virtualisasi dan Cloud Computing adalah mekanisme teknologi yang cukup penting untuk dipelajari, terutama bagi para Administrator di perusahaan atau lembaga atau institusi yang ingin mengelola sistem komputerisasi secara lebih mudah dan mampu memberikan kinerja yang optimal. Sayangnya, tidak semua pihak memahami pentingnya virtualisasi dan relasinya dengan cloud computing. Meski sekarang banyak terdengar istilah "cloud computing", istilah tersebut lebih banyak didengar sebagai *marketing hype* atau jargon marketing yang belum terlalu jelas dipahami.

Apa Itu Virtualisasi & Cloud Computing

Virtualisasi bisa diartikan sebagai pembuatan suatu bentuk atau versi virtual dari sesuatu yang bersifat fisik, misalnya sistem operasi, perangkat storage/penyimpanan data atau sumber daya jaringan. Virtualisasi bisa diimplementasikan kedalam berbagai bentuk, antara lain (Harry Sufehmi, Pengenalan Virtualisasi, 20090607):

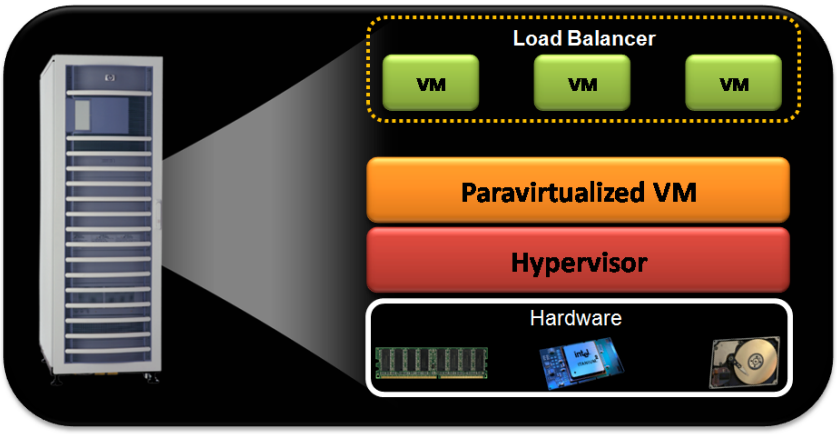
1. Network Virtualization : VLAN, Virtual IP (untclustering), Multilink
2. Memory Virtualization : pooling memory dari node-node di cluster
3. Grid Computing : banyak komputer = satu
4. Application Virtualization : Dosemu, Wine
5. Storage Virtualization : RAID, LVM
6. Platform Virtualization : virtual computer

Pembahasan kali ini akan menitikberatkan pada materi platform virtualization alias virtualisasi komputer dan sistem operasi. Cloud Computing adalah sistem komputerisasi berbasis jaringan/internet, dimana suatu sumber daya, software, informasi dan aplikasi disediakan untuk digunakan oleh komputer lain yang membutuhkan. Mengapa konsep ini bernama komputasi cloud atau cloud computing? Ini karena internet sendiri bisa dianggap sebagai sebuah cloud besar (biasanya dalam skema network, internet dilambangkan sebagai cloud) yang berisi sekumpulan besar komputer yang saling terhubung, jadi cloud computing bisa diartikan sebagai komputerisasi berbasis sekumpulan komputer yang saling terhubung. Cloud computing bisa dianggap sebagai perluasan dari virtualisasi. Perusahaan bisa menempatkan aplikasi atau sistem yang

digunakan di internet, tidak mengelolanya secara internal. Contoh cloud computing untuk versi public adalah layanan-layanan milik Google seperti Google Docs dan Google Spreadsheet. Adanya kedua layanan tersebut meniadakan kebutuhan suatu aplikasi office untuk pengolah kata dan aplikasi spreadsheet di internal perusahaan. Contoh cloud computing untuk keperluan non public adalah Amazon EC2 (Amazon Elastic Compute Cloud). Amazon menyediakan komputer induk, kita bisa mengirim dan menggunakan sistem virtual dan menggunakannya dalam jangka waktu dan biaya sewa tertentu.

Virtual Machine

Ini adalah lapisan yang akan langsung berinteraksi dengan Platform sebagai Service (PaaS), Anda mungkin tidak menyadari bahwa Anda berurusan dengan VM tetapi dalam kenyataannya sebagian besar implementasi Cloud akan menjadi kode atau aplikasi pada VM. Aplikasi yang berjalan pada sebuah VM yang di kelola oleh Hypervisor berjalan disemua server.



Gambar 35. Virtual Machine
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Keuntungan Virtualisasi & Cloud Computing

1) **Pengurangan Biaya Investasi Hardware**. Investasi hardware dapat ditekan lebih rendah karena virtualisasi hanya mendayagunakan kapasitas yang sudah ada. Tak perlu ada penambahan perangkat komputer, server dan pheriperal secara fisik. Kalaupun ada penambahan kapasitas harddisk dan memori, itu lebih ditujukan untuk mendukung stabilitas kerja komputer induk, yang jika dihitung secara finansial, masih jauh lebih hemat dibandingkan investasi hardware baru.

2) **Kemudahan Backup & Recovery**. Server-server yang dijalankan didalam sebuah mesin virtual dapat disimpan dalam 1 buah image yang berisi seluruh konfigurasi sistem. Jika satu saat server tersebut crash, kita tidak perlu melakukan instalasi dan konfigurasi ulang. Cukup mengambil salinan image yang sudah disimpan, merestore data hasil backup terakhir dan server berjalan seperti sedia kala. Hemat waktu, tenaga dan sumber daya.

3) **Kemudahan Deployment**. Server virtual dapat dikloning sebanyak mungkin dan dapat dijalankan pada mesin lain dengan mengubah sedikit konfigurasi. Mengurangi beban kerja para staff IT dan mempercepat proses implementasi suatu sistem

4) **Mengurangi Panas**. Berkurangnya jumlah perangkat otomatis mengurangi panasnya ruang server/data center. Ini akan berimbas pada pengurangan biaya pendinginan/AC dan pada akhirnya mengurangi biaya penggunaan listrik

5) **Mengurangi Biaya Space**. Semakin sedikit jumlah server berarti semakin sedikit pula ruang untuk menyimpan perangkat. Jika server ditempatkan pada suatu co-location server/data center, ini akan berimbas pada pengurangan biaya sewa

6) **Kemudahan Maintenance & Pengelolaan**. Jumlah server yang lebih sedikit otomatis akan mengurangi waktu dan biaya untuk mengelola. Jumlah server yang lebih sedikit juga berarti lebih sedikit jumlah server yang harus ditangani

7) **Standarisasi** Hardware. Virtualisasi melakukan emulasi dan enkapsulasi hardware sehingga proses pengenalan dan pemindahan suatu spesifikasi hardware tertentu tidak menjadi masalah. Sistem tidak perlu melakukan deteksi ulang hardware sebagaimana instalasi pada sistem/komputer fisik

8) **Kemudahan Replacement.** Proses penggantian dan upgrade spesifikasi server lebih mudah dilakukan. Jika server induk sudah overload dan spesifikasinya tidak mencukupi lagi, kita bisa dengan mudah melakukan upgrade spesifikasi atau memindahkan virtual machine ke server lain yang lebih powerful

Kerugian Penggunaan Virtualisasi

1. **Satu Pusat Masalah.** Virtualisasi bisa dianalogikan dengan menempatkan semua telur didalam 1 keranjang. Ini artinya jika server induk bermasalah, semua sistem virtual machine didalamnya tidak bisa digunakan. Hal ini bisa diantisipasi dengan menyediakan fasilitas backup secara otomatis dan periodik atau dengan menerapkan prinsip fail over/clustering

2. **Spesifikasi Hardware**. Virtualisasi membutuhkan spesifikasi server yang lebih tinggi untuk menjalankan server induk dan mesin virtual didalamnya

4. **Satu Pusat Serangan**. Penempatan semua server dalam satu komputer akan menjadikannya sebagai target serangan. Jika hacker mampu menerobos masuk kedalam sistem induk, ada kemungkinan ia mampu menyusup kedalam server-server virtual dengan cara menggunakan informasi yang ada pada server induk

Kebutuhan Sistem Untuk Virtualisasi

Pada dasarnya, kebutuhan spesifikasi server tergantung pada virtual server yang akan digunakan. Semakin tinggi spesifikasi yang akan dijalankan, semakin tinggi pula spesifikasi server yang akan digunakan sebagai server induk. Meski demikian, asumsi ini tidak 100% benar karena ada beberapa teknologi virtualisasi seperti OpenVZ yang mampu melakukan load balancing sehingga jika mesin virtual ada 5 yang masing-

masing membutuhkan memory 1 GB tidak berarti bahwa server harus memiliki spesifikasi diatas 5 X 1GB.



Gambar 36. Sebuah Cloud pusat data berjalan ratusan server
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Berikut adalah spesifikasi minimal server induk yang akan digunakan untuk menjadikan sebuah distro Linux sebagai virtual server :

1. Processor Pentium 4. Jika akan menggunakan arsitektur 64 bit, server harus memiliki kemampuan 64 bit juga
2. jika akan menggunakan model full virtualization pada Xen Hypervisor, prosessor memiliki model Intel VT (Virtualization Technology) atau AMD-V
3. Memory minimal 1 GB
4. Kapasitas Harddisk minimal 20 GB
5. Memiliki network card untuk keperluan networking

BAGIAN 4

PENERAPAN CLOUD COMPUTING

Google App Engine

Google App Engine adalah sebuah platform untuk menyebarkan dan menjalankan aplikasi web di Google infrastruktur. Muncul dengan server web dinamis dan dukungan penuh untuk umum teknologi web. Google App engine manawarkan menyimpanan data transaksional untuk data bertahan. Pengembang dapat mengitegrasikan aplikasi web dengan akun Google melalui API.

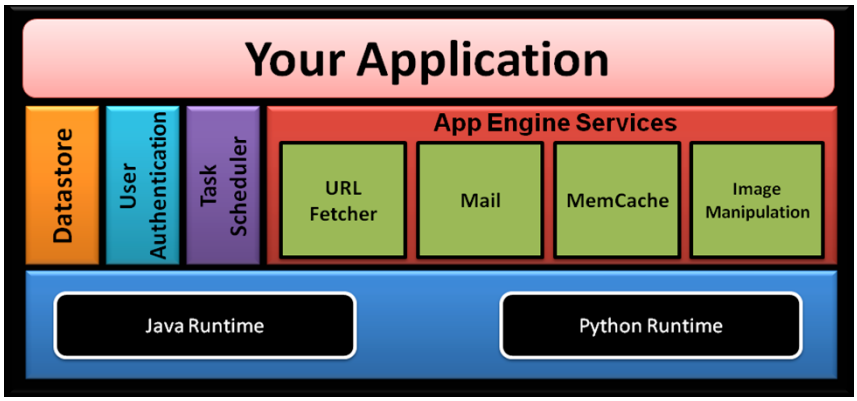
Keuntungan terbesar menjalankan aplikasi web pada GAME adalah skalabilitas yang menawarkan. Aplikasi web Anda akan menjadi seperti scalable karena beberapa yang populer layanan Google seperti pencarian.



Gambar 37. Web aplikasi yang berjalan bersama dengan properti Google
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Google App Engine saat ini mendukung Python dan Java. Pengembang Javadapat meyebarkan dan menjalakan JSP dan servlets sedangkan pengembang Python dapat menggunakan perpustakaan standar.

.



Gambar 38. Components of Google App Engine
Sumber : Janakiram MSV Cloud Computing Strategist 2010

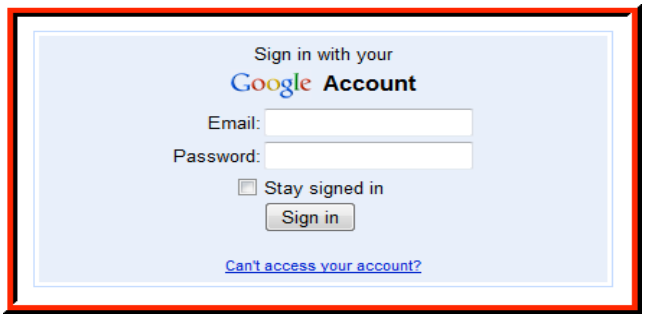
Lapisan logis berikutnya adalah seperangkat API dan service untuk mendukung aplikasi web pengembang. Lapisan ini memiliki Datastore persisten, Pengguna jasa Otentikasi, penjadwall tugas dan tugas antrian, URL, komponen Mail, memcache dan manipulasi gambar. Semua layanan ini terpapar melalui binding API asli. Misalnya pengembang Java dapat menggunakan JDO/JPA untuk berbicara dengan datastore tersebut.
Mari kita lihat lebih dekat beberapa layanan yang diberikan oleh GAME.

Java Runtime - Game didasarkan pada Java Servlet 6 VM dan 2,5 kontainer.datastore yang dapat diakses melalui JDO/JPA API. Ini mendukung JSR 107 untuk memcache API. Mail dapat diakses melalui API javax.mail.javax.net.URL connection menyediakan akses kelayanan URL Fetch. Selain bahasa java inti, dinamis lainnya bahasa berbasis java seperti JRuby dan Scala.

Rumtime python -Datang dengan satu set API dan alat untuk mengembangkan web aplikasi Python mendukung Python 2.5.2 dan Python 3 sedang dipertimbangkan untuk rilis dimasa depan. Python menyediakan API untukDatastore, akun googel, URLdan layanan email. App Engine juga menyediakan Python framework aplikasi web.

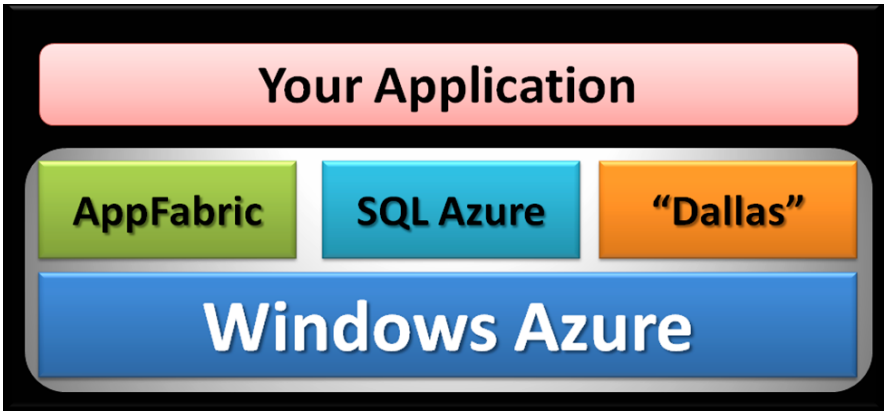
Datastore - App Engine dilengkapi dengan penyimpanan data yang sangat kuat yang skala dinamis. Ia juga memilki mesin query dan dukungan untuk transaksi. Entitas telah menetapkan properti yang dapat di query menggunakan SQL seperti tata bahasa GQL atau Google Query Language.

Pengguna Otentikasi - Salah satu keuntungan mengguanakan ini adalah integrasi dengan akun Google. Ini memberdayakan pengembang untuk memanfaatkan google yang aman otentikasi mesin untuk aplikasi. aplikasi dapat mengakses alamat emaial pengguna serta pengguna ID.



Gambar 39. Akun Google App Engine integrasi
Sumber : Janakiram MSV Cloud Computing Strategist 2010

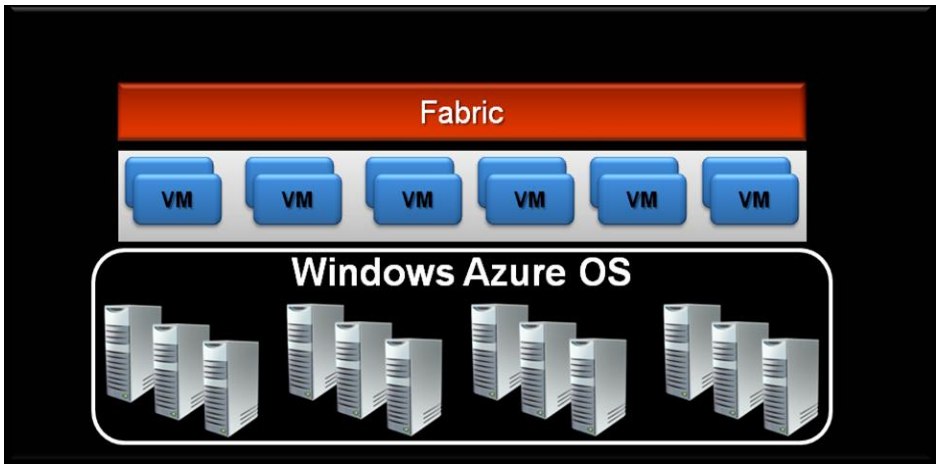
Pada tingkat tinggi, platform Windows Azure memiliki 4 layanan kunci di dalamnya yang pertama Windows Azure yang merupakan OS dari Microsoft Cloud. Layanan kedua adalah AppFabric yang memungkinkan integrasi layanan on-premisedengan. Layanan ketiga adalah Database Cloud disebut SQL Azure yang didasarkan pada Microsoft SQL Server. Penambahan terbaru untuk platform adalah layanan CODEC "Dallas" yang merupakan pasar untuk mempublikasikan, menemukan, mengkonsumsi dan menganalisa konten utama. Meskipun Windows Azure Platform dirancang untuk membangun pengembang aplikasi pada platform Microsoft, ini juga dapat dimanfaatkan oleh pengembang aplikasi pada lingkingan java dan PHP. Microsoft investasi dikanan set alat dan plug-in untuk Eclipse.



Gambar 40. Windows Azure platform
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Windows Azure

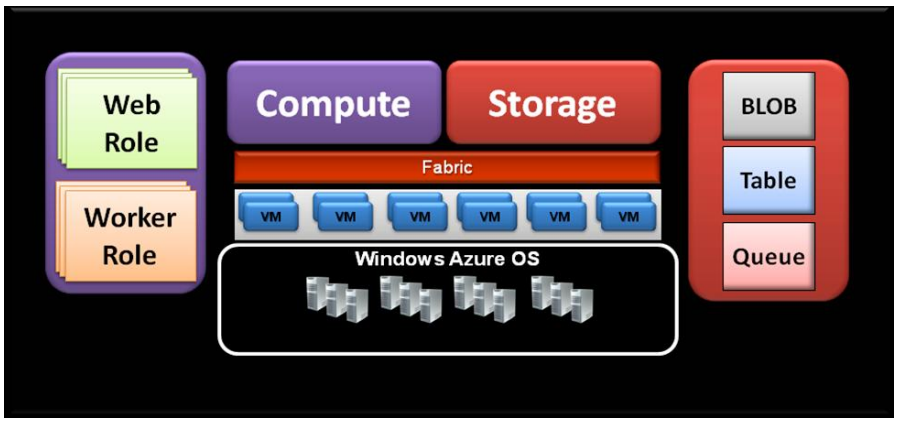
Windows Azure adalah jantung dan jiwa dari Platform Azure. Ini adalah OS yang berjalan pada setiap server dipusat data dibeberapa geografis lokasi. Sangat menarik untuk dicacat bahwa Windows Azure OS tidak tersedia sebagai ritel OS. Ini adalah versi home eksklusif dirancang untuk Cloud Microsoft infrastruktur. Karena OS ini berjalan di beberapa server fisik, ada lapisan di atas yang mengkoordinasikan pelaksanaan proses. Lapisan ini disebut Fabric. Di antara Fabric dan Windows biru OS, ada ratusan mesin virtual (VM) yang benar-benar menjalakan kode dan aplikasi.



Gambar 41. Windows Azure architecture
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Aplikasi diharapkan untuk menjalankan dalam salah satu dari dua peran yang disebut Peran web. Peran web dimaksudkan untuk host yang khas web ASP.NET aplikasi atau aplikasi web lainnya CGI. Peran web adalah host yang menjalankan panjang proses yang tidak memiliki UI. Peran web juga dapat host Layanan WCF yang mengekspos endpoint HTTP. Kode dalam Peran web akan berjalan independen.

Melalui Peran web, Anda bisa mengakses port baik aplikasi NET atau asli COM aplikasi untuk windows azure. Antrian adalah mekanisme untuk berkomunikasi antara peran web. Data disimpan dalam Azure penyimpanan dapat diakses melalui HTTP dan panggilan REST.

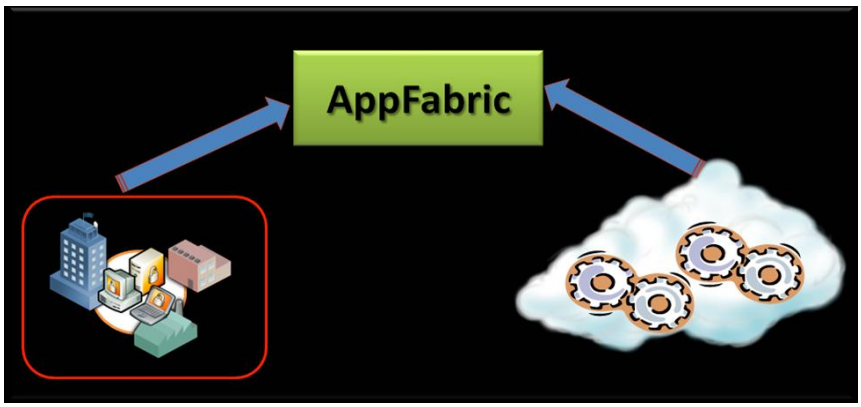


Gambar 42. Compute & Storage Service
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Jadi, baru saja kita bahas bahwa Windows Azure menawarkan Hitung dan layanan penyipanan. Hitung layanan dikonsumsi oleh deploying Aplikasi Web dalam Peran Web dan menjalankan proses yang panjang dalam Peran Pekerja. Penyimpanan dapat dikonsumsi melalaui table antrian.

AppFabric

Windows Azure Platform AppFabric sebelumnya disebut .NET Layanan. Layanan memungkinkan integrasi layanan yang berjalan dalam suatu organisasi di belakang firewall dengan layanan yang di sediakan oleh cloud. Ini mengamankan jembatan antara aplikasi legacy dan layanan cloud. AppFabric juga membawa identitas federasi untuk aplikasi berbasis cloud. dua komponen kunci AppFabric yaitu 1) Service Bus dan, 2) Access Control.



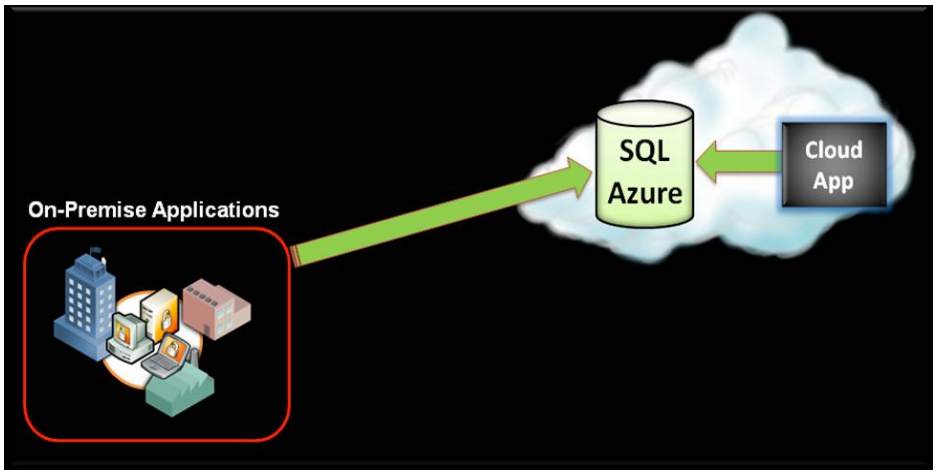
Gambar 43. AppFabric menghubungkan pada-premis ke Cloud
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Layanan Bus menyediakan konektivitas yang aman antara on-premise dan layanan Cloud. Hal ini dapat digunakan untuk mendaftar, menemukan dan menggunakan service. Layanan host dibelakang firewall dan .NET bias didaftarkan dengan layanan bus kemudian layanan ini dapat dipanggil oleh layanan Cloud. Layanan bus abstrak fisik dari layanan menyediakan URL yang dapat diipanggil oleh konsumen potensial. Access Control adalah mekanisme untuk mengamankan layanan Cloud dan aplikasi. Ini menyediakan cara deklaratif mendefinisikan aturan dan klaim melalui mana penelepon yang mendapatkan akses ke layanan Cloud. Akses Kontrol aturan dapat dengan mudah dan fleksibel dikonfigurasi untuk menutup berbagai kebutuhan keamanan dan berbeda identitas manajemen infrastruktur. Acces control memungkinkan perusahan untuk mengintegrasikan on-premise mekanisme keamanan seperti Active Direktory dengan

otentikasi berbasis Cloud. Pengembang dapat mengakses program control melalui layanan WCF.

SQL Azure

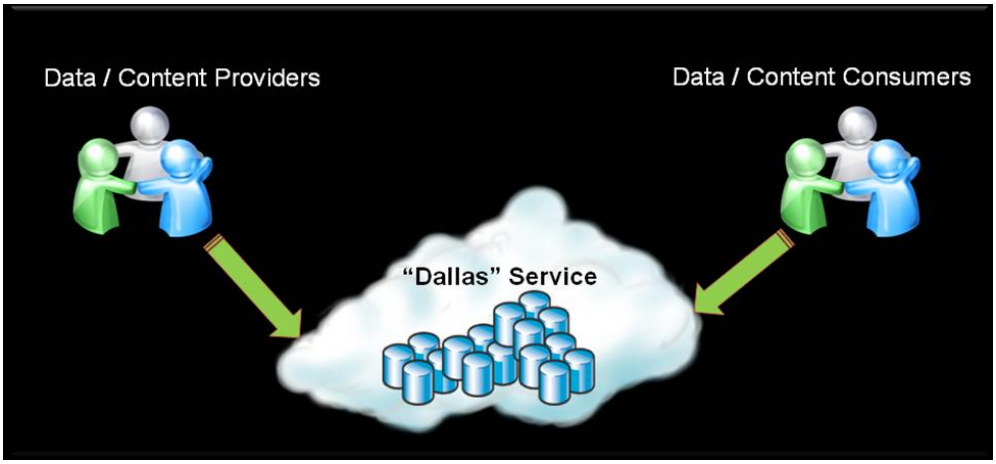
SQL Azure adalah Microsoft SQL Server pada Cloud. Tidak seperti penyimpanan Azure, yang dimaksud data tidak terstruktur, SQL Azure adalah mesin database relasioonal. Hal ini didasarkan pada mesin DB yang sama MS SQL Server dan dapat dilihat dengan T-SQL. Dengan MS SQL, on-premis aplikasi dengan cepat dapat mulai menggunakan layanan ini. Pengembang dapat berbicara dengan SQL Azure menggunakan ADO.NET atau ODBC API. Pengembang PHP dapat menggunakan melalui API PHPi. melalui Microsoft SQL Azure Data Sync, data dapat dengan mudah disinkronkan antara On-Premise SQL Server dan SQL Azure. Ini adalah fitur yang sangat kuat untuk membagun pusat-pusat data pada Cloud. tujuan praktis, SQL Azure dapat diperlakukan persis seperti server DB berjalan di pusat data tanpa overhead menjaga dan mengelola. Karena Microsoft bertanggung jawab untuk instalasi, pemeliharaan dan ketersediaan layanan bisnis DB, hanya bisa fokus pada memanipulasi dan mengakses data sebagai sebuah layanan. Dengan dimuka Bayar-as-you-go pendekatan, tidak ada investasi dan hanya akan membayar apa yang kita gunakan.



Gambar 44. SQL Azure
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Microsoft CODEC "Dallas"

Layanan ini merupakan pengaturan pertukaran oleh Microsoft untuk pihak yang dapat mempublikasikan data/konten untuk pihak yang dapat mengkonsumsi data dalam aplikasi mereka. Misalnya, Sektor Publik dapat mempublikasikan data sensus yang menarik dan bermanfaat dan beberapa perusahaan dalam bisnis kesehatan. perusahaan dapat mencari dan menemukan Dataset sensus dan membayar apa yang mereka mengkonsumsi. Data dapat diterbitkan dalam berbagai bentuk termasuk kumpulan file CSV Spreadsheet dan RSS feed. Dataset diterbitkan di Dallas dapat dikonsumsi secara langsung melalui alat seperti Microsoft Excel atau dapat diintegrasikan ke dalam kustom aplikasi dengan memanggil API berbasis REST. Data penerbit dan konsumen data melalui sebuah add-in untuk excel 2010 disebut PowerPivot pengguna akhir dapat langsung mengkonsumsi data dalam Microsoft Exel.



Gambar 45. Microsoft Codename "Dallas" service
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Kelebihan Microsoft Platform Windows Azure

Scalable Aplikasi Web

Karena Windows Azure Platform didasarkan pada platform akrab NET., ASP.NET pengembang dapat merancang dan mengembangkan aplikasi web pada mesin yang cukup murah. Ini akan memberdayakan para pengembang untuk segera menggunakan aplikasi web mereka, tanpa khawatir tentang biaya dan kompleksitas dari kebutuhan infrastruktur. Bahkan pengembang PHP dapat menikmati manfaat dari elastisitas dan bayar dengan menggunakan atribut dari platform.

Hitung Aplikasi Intensif

Windows Azure Platform dapat digunakan untuk menjalankan proses aplikasi intensif yang kadang-kadang membutuhkan sumber daya komputasi yang tinggi akhir. Dengan memanfaatkan Peran Pekerja, pengembang dapat memindahkan kode yang dapat berjalan di beberapa aplikasi. Data yang dihasilkan baik oleh peran web atau On-Primise aplikasi dapat diberikan ke peran web melalui penyimpanan Azure.

Sentralisasi Akses Data

Ketika data harus diakses untuk berbagai aplikasi yang berjalan di browser, desktop dan mobile, Azure Cloud berbasis penyimpanan dapat menjadi solusi bagus untuk bertahan dan mempertahankan Data yang dapat dengan mudah digunakan oleh aplikasi desktop, Silver light, flash dan ajax dan aplikasi berbasis web atauu mobile.

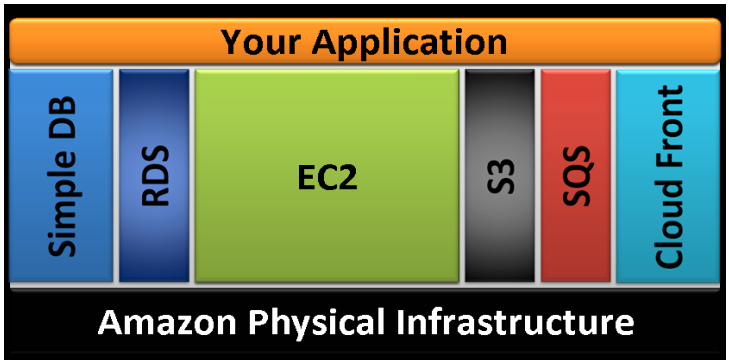
Aplikasi Hybrid (Cloud + On-Premise)

Mungkin ada persyaratan untuk memperluas bagian dari aplikasi ke Cloud atau membangun façade Cloud untuk aplikasi yang sudah ada. Dengan memanfaatkan AppFabric layanan seperti layanan Bus dan acces control, on-premis aplikasi dapat mulus dan aman diperluas ke cloud. AppFabric dapat mengaktifkan Hybrid cloud.

Berdasarkan data cloud Hub

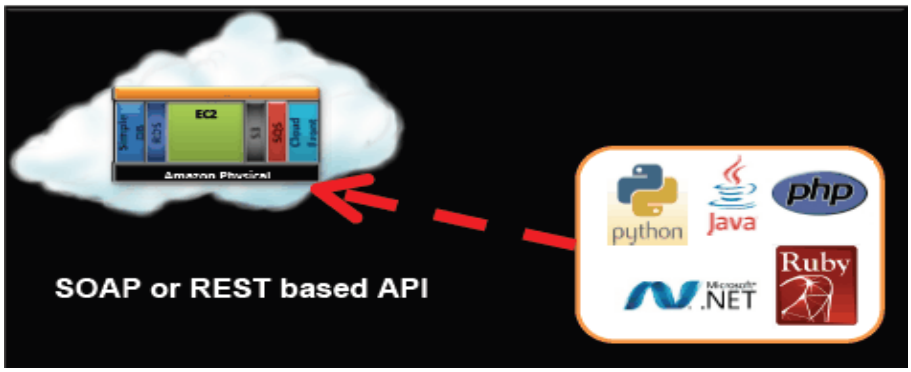
Melalui SQL Azure, perusahaan aman dapat membangun hub data yang akan terbuka untuk mitra dagang dan karycloud mobile. Sebagai contoh, Persediaan dari perusahan manufaktur dapat host di cloud yang selalu dalam sinkrin dengan database persediaan local. DB berbasis cloud akan dibuka untuk mitra B2B langsung query dan memesan.

Amazon Web Services adalah salah satu awal dan juga yang paling sukses impelemntasidari Cloud Publik. Banyak juga dikenal secara online memanfaatkan sifat AWS. Amazon awalnya mulai menawarkan pesan layanan cloud berbasis antrian yang disebut Amazon layanan antrian sederhana atau SQS. Selain itu Amazon menambahkan layanan seperti Mechanical Truk, layanan penyimpanan sederhana (S30, Elastic Compute Cloud (EC2), CDN A layanan yang disebut CloudFront, layanan database flesksibek dan di distribusikan disebut SimpleDB. Amazon baru-baru ini mengumumkan ketersediaan MySQL di Cloud melalui layanan yang disebut Relationl Layanan data (RDS).



Gambar 46. Amazon Web Services
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Mengingat bahwa Amazon menawarkan kemampuan inti untuk menjalankan aplikasi web lengkap atau line aplikasi bisnis, jelas itu adalah infrastruktur sebagai layanan (IaaS). AWS benar-benar platform dan kita dapat memilih OS, App server dan bahasa program pilihan anda. AWS SDK untuk sebagian besar bahsa popular termasuk Java, .NET, Python dan Ruby.

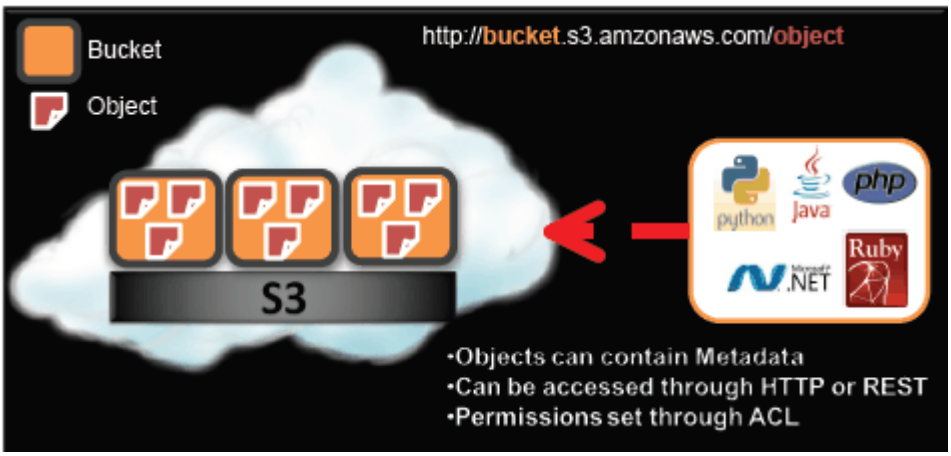


Gambar 47. AWS API Bindings
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Ada beberapa dari penawaran layanan Cloud utama dari Amazon :

S3

Penyimpanan sederhana Amazon Layanan atau S3 adalah cara yang bagus untuk menyimpan data pada Cloud yang dapat diakses ke internet. S3 dapat menyimpan data sebagai objek disertai dengan metadata. Objek-objek diatur kedalam kotak dan setiap kotak dan objek memiliki separangkat hak akses di definisikan Acces Control List (ACL). Semua yang tersimpan di S3 bisa menerima dokumen, file media, objek serial atau gambar Vitual Machine bahkan setiap objek dapat 5GB dalam ukuran sementara metadata bias sampai 2 KB. Semua dapat di akses menggunakan panggilan SOAP. Hal ini membuat sebuah penyimpanan yang ideal untuk pusat menyimpan dan mengambil data dibeberapa klien. S3 juga bias sebagai system virtual untuk memberikan kemampuan penyimpanan untuk aplikasi.

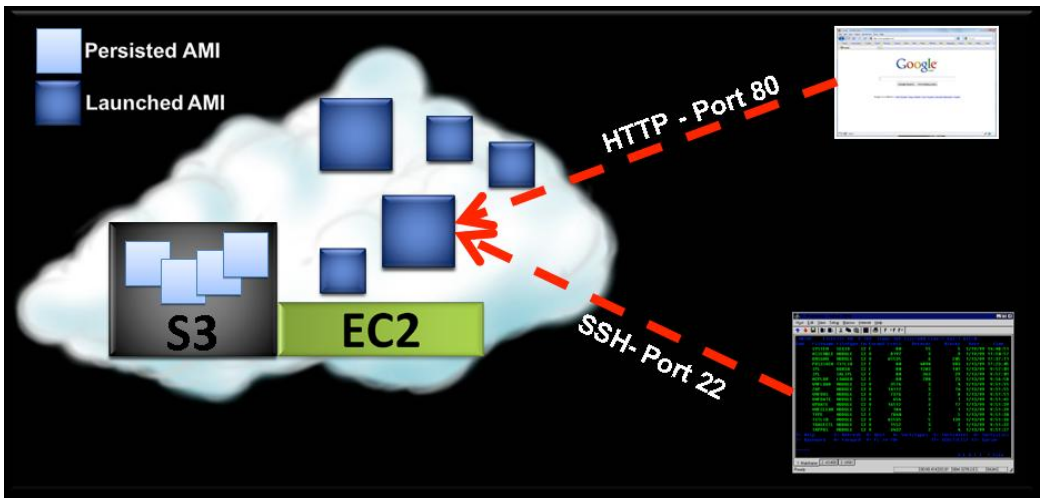


Gambar 48. Simple Storage Services
Sumber : Janakiram MSV Cloud Computing Strategist 2010

EC2

Secara sederhana, EC2 adalah menyewa sebuah server yang berjalan di lokasi yang jauh. server ini sebenarnya Virtual Machine yang berjalan diatas data yang kuat Amazon pusat. Server virtal sebagai mesin mesin foto amazon atau AMI. Contoh datang dalam ukuran yang berbeda yang dapat Anda pilih. Silahkan lihat http://aws.amazon.com/ec2/ # misalnya untuk rincian lebih lanjut pada contoh jenis. Ada banyak pra-dikonfigurasi AMIS yang dapat Anda pilih dari. untuk alur kerja pada EC2 adalah bahwa Anda memilih pra-dikonfigurasi AMI, peluncuran yang AMI, menyesuaikan dengan menambahkan perangkat lunak tambahan dengan

membuat sebuah aplikasi. AMI sebagai kustom Anda pada S3. Anda dapat memulai beberapa contoh AMI dan melampirkannya ke IP yang disebut IP elastic. hal itu disebut Elastic Compute Cloud.



Gambar 49. Elastic Compute Cloud
Sumber : Janakiram MSV Cloud Computing Strategist 2010

SQS

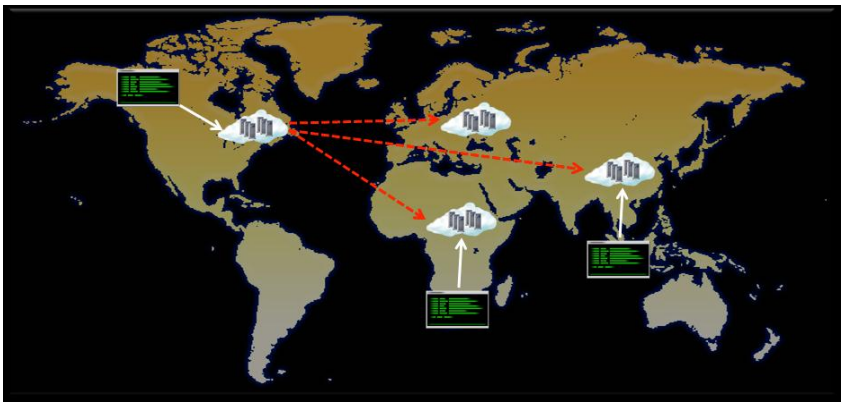
SQS adalah antrian pesan pada Cloud. Mendukung pengiriman program dari pesan melalui layanan aplikasi web sebagai cara untuk berkomunikasi melalui internet. Pesan Beriorientasi Middleware (MOM) adalah cara yang popular untuk memastikan bahwa dikirim sekali dan hanya sekali. SQS memberi Anda kemampuan ini on-demand dan melalui model bayar-per-digunakan. SQS diakses melalui REST dan SOAP berbasis API.



Gambar 50. Simple Message Queue
Sumber : Janakiram MSV Cloud Computing Strategist 2010

CloudFront

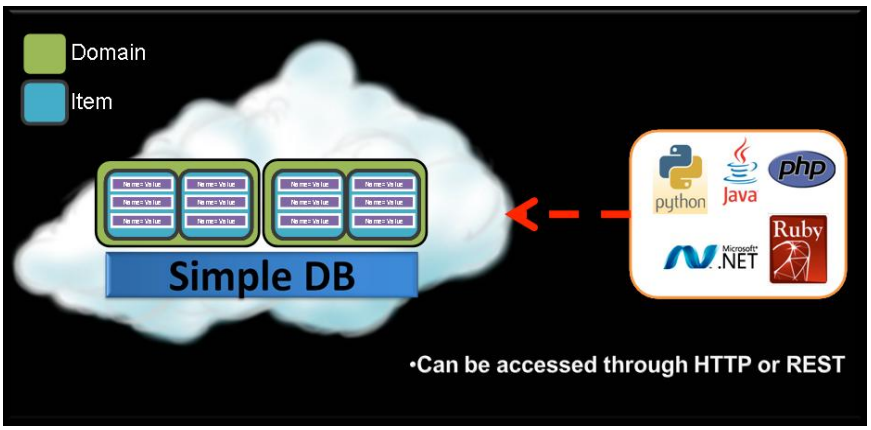
Ketika aplikasi web Anda menargetkan pengguna global, masuk akal untuk melayani isi statis melalui server yang lebih dekat ke pengguna. Salah satu solusi berdasarkan prinsip ini disebut Content Delivery Network (CDN). tapi ini infrastruktur server geografis tersebar untuk melayani konten statis dan bisa sangat mahal. CloudFront adalah CDN sebagai layanan Amazon memanfaatkan data center diseluruh dunia dengan melayani konten melalui loasi-lokasi tertentu.



Gambar 50. Cloud Front
Sumber : Janakiram MSV Cloud Computing Strategist 2010

SimpleDB

Jika S3 menawarkan penyimpanan untuk data biner di SimpleDB adalah cara yang fleksibel untuk Nama/Nilai pasang pada Cloud. Hal ini secara dramatis mengurangi overhead memelihara sebuah database relasional terus menerus. SimpleDB diakses melalui HTTP panggilan dan dapat dengan mudah dipakai oleh setiap klien yang dapat mengurai HTTP respon. Banyak aplikasi Web 2.0 yang dibangun menggunakan AJAX, Flash dan Silverlight dapat mudah di akses dari SimpeDB

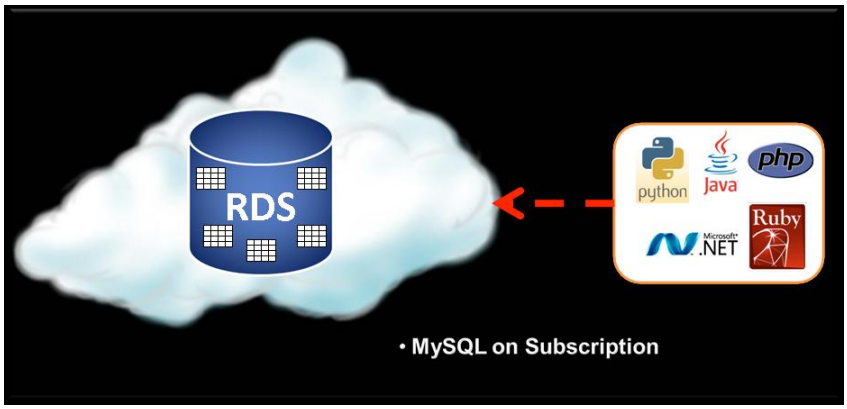


Gambar 51. Simple DB
Sumber : Janakiram MSV Cloud Computing Strategist 2010

RDS

Amazon RDS menawarkan database relasional pada Cloud. Hal ini didasarkan pada poluler Database MySql. Keuntungan dari RDS adalah bahwa Anda tidak mengisntal, mengkonfigurasi, mengelola dan menjaga server DB.

Operasi rutin seperti menambal server dan back up database diambil perawatan dan Anda hanya menggunakan layanan. RDS adalah harga di Pay-as-you-go model dan tidak ada investasi dimuka yang diperlukan. Hal ini dapat diakses melalui REST dan SOAP berdasarkan API.



Gambar 52. Relational Database Services
Sumber : Janakiram MSV Cloud Computing Strategist 2010

Kelebihan Menggunakan AWS

Scalable Aplikasi Web

Jika Anda adalah seorang wirausahcloud dan ingin go-live dengan aplikasi Anda tanpa investasi dimuka, Amazon adalah tempatnya. Dengan menjalankan aplikasi web di Amazon, Anda dinamis dapat skala aplikasi pada permintaan dan hanya membayar untuk apa yang anda gunakan.

Jalur Aplikasi Bisnis

Jika perusahaan Anda harus membuka aplikasi LOB internal untuk karycloud dan mitra dagang, dapat memperpanjang aplikasi ke Cloud dengan memanfaatkan konsep AWS disebut Cloud Virtual Private (VPC). Ini adalah layanan Hybrid Cloud dengan kemampuan sebagian aplikasi pindah ke Cloud sementara masih menjalankan bagian sensitif dan eksklusif dari aplikasi LOB diamankan belakang firewall.

Data Arsip

Data yang tidak terlalu sering diakses, tapi mungkin diperlukan karena data kebijakan retensi, dapat dengan mudah diarsipkan di Amazon S3. Selain itu dapat dicari frontend data dan diambil on-demand. Pindah data ke Cloud akan memastikan bahwa data tersedia dari mana dan kapan saja.

High-Performance Computing On Demand

Bagi banyak perusahaan, ada kebutuhan sesekali kinerja tinggi komputasi. Investasi dalam high-end server merupakan solusi yang optimal karena tidak dapat digunakan setelah tugas dilakukan. Dengan AWS, perusahaan hampir dapat mempekerjakan sebgai kekuatan komputasi sebanyak yang mereka butuhkandan membayar hanya yang digunakan. Hal ini akan menghilangkan proposisi invetasi mahal dalam infrastruktur.

Scalable Media Pengiriman

Sebuah saluran TV mungkin ingin memberikan dan menunjukkan direkam ke global penonton. Karena sebagian besar konten yang statis, mereka dapat memanfaatkan CDN,kemampuan Mendaftar dengan layanan seperti Akamai dan Limelight bisa mahal. Karena isi media sudah tersimpan pada S3, sangat mudah dan biaya efektif untuk memanfaatkan CloudFront Amazon untuk memberikan isi media melalui secara geografis menyebar lokasi tepi.

BAGIAN 5
NOSQL CLOUD DATABASE

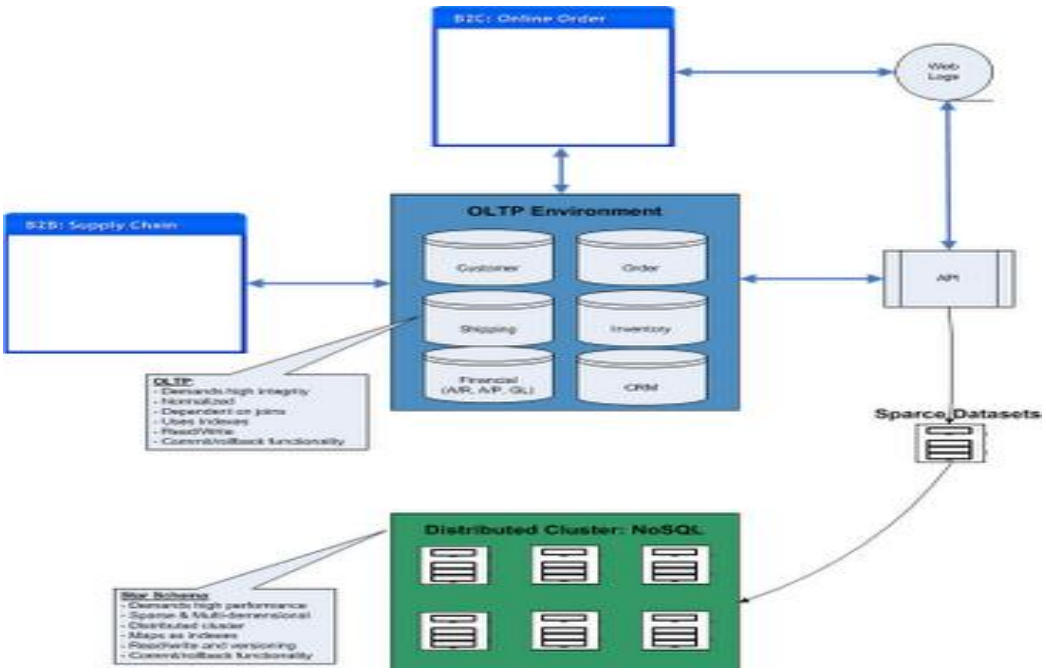
Mengenal NoSQL DataBase

NOSQL adalah istilah untuk menyatakan berbagai hal yang didalamnya termasuk database sederhana yang berisikan key dan value seperti Memcache, ataupun yang lebih canggih yaitu non-database relational seperti MongoDB, Cassandra, CouchDB, SQLAZure, Google Storage, OpenCloud Storage hadoop.apache.org, dll.
Wikipedia menyatakan NoSQL adalah sistem menejemen database yang berbeda dari sistem menejemen database relasional yang klasik dalam beberapa hal. NoSQL mungkin tidak membutuhkan skema table dan umumnya menghindari operasi join dan berkembang secara horisontal. Akademisi menyebut database seperti ini sebagai structured storage, istilah yang didalamnya mencakup sistem menejemen database relasional.

Database relasional sudah ada semenjak tahun 70-an sehingga teknologi mereka sudah sangat matang. Secara umum mereka mendukung operasi transaksi, yang mengijinkan kita merubah sebagian data, melakukan kontrol terhadap operasi database, support terhadap constraint seperti unique, primary key, foreign key dan check. Mereka juga memiliki bahasa SQL atau Simplified Query Language untuk mengakses data, merubah data seperti operasi insert, update dan delete.
Walaupun SQL dalam arti sesungguhnya adalah simple atau sederhana, dan developer selama bertahun tahun menggunakannya, tetapi mereka merasa kurang puas bahkan cenderung tidak menyukainya. Alasan lainnya, RDBMS atau Relational Database Management System tidak dapat berkembang horisontal secara baik. Seringnya kita mendapatkan database yang berkembang tetapi secara read-only melalui kemampuan replikasi database dan untuk mendapatkan database yang berkembang horisontal secara read-write itu sangat sulit. Oracle saja sampai perlu membangun ORACLE RAC atau Real Application Cluster, yang menemui banyak tantangan untuk melakukan sinkronisasi data di internal cache melalui inter-koneksi khusus. Faktanya, perubahan data yang terjadi itu memerlukan waktu untuk mengirimkannya ke berbagai sistem. Selama data tersebut belum terkirimkan, kita memakai data yang tidak valid atau stale data/delta data. Adanya database NoSQL seperti MongoDB yang mencoba untuk menyelesaikan permasalahan ini. Disini, Data tidak ditulis/dibaca dari database dengan menggunakan bahasa SQL, tetapi menggunakan metode object-oriented yang lebih

disukai oleh developers. Kelebihan lainnya adalah adanya dukungan adanya banyaknya tipe index yang berbeda beda untuk lookupsterhadap data tertentu. Mereka juga memiliki kemampuan clustering secara default.

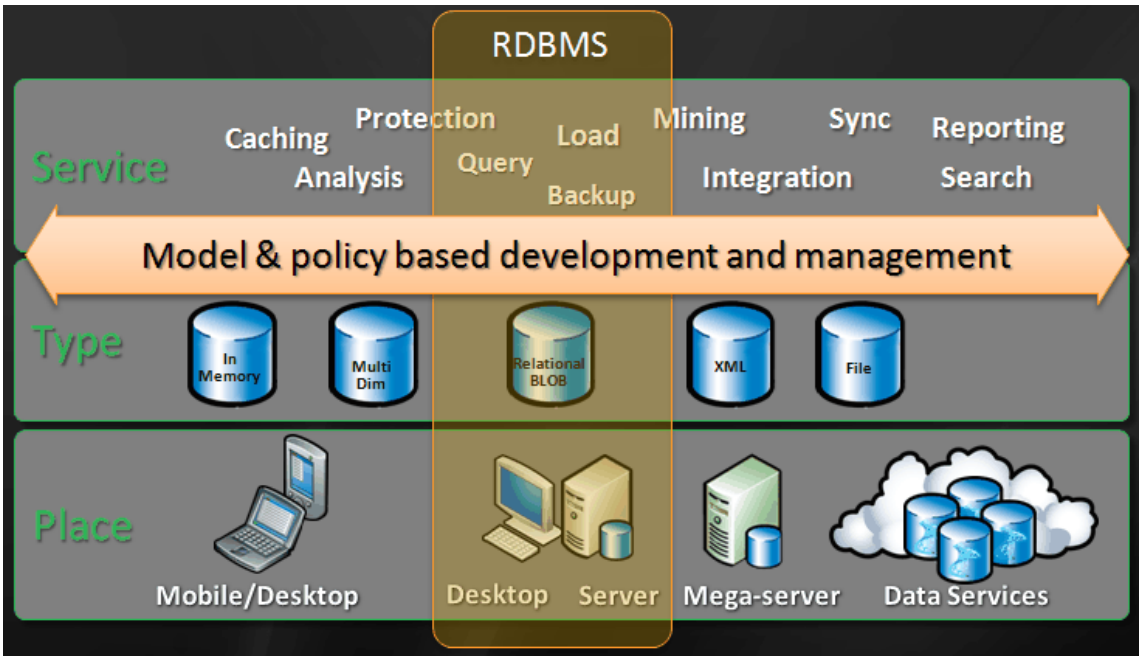
NoSQL mengambil yang terbaik dari RDBMS dan pendekatan Skema Star dan mengambil satu langkah lebih lanjut. NoSQL menggunakan struktur multi dimensi jarang data dan kelompok data yang relevan sama untuk mengurangi waktu i / o yang dibutuhkan untuk kembali hasil. NoSQL juga mendistribusikan pekerjaan di beberapa lokasi (sering digunakan pada sebuah grid) sehingga banyak benang yang bekerjasecara simultan dan independen. Sebaliknya indeks, NoSQL menggunakan konsep peta yang memegang beberapa nilai indeks memungkinkan untuk peta tunggal untuk menangani set dinamis pertanyaan berdasarkan banyak atribut. NoSQL juga memungkinkan untuk versioning catatan. Pada waktu-stamping perubahan, catatan baru ditambahkan ke database tanpa overhead yang update dan menghapus yang ada dalam sebuah RDBMS. Cara baru menangani database yang besar mungkin terlihat seperti ini di beberapa perusahaan. NoSQL menyediakan paket API sehingga para pengembang dapat menambahkandata ke struktur data dengan cepat tanpa harus melewati proses ETL batch. Sebuah proses ETL masih dapat digunakan jika diinginkan tetapi tidak lagi diperlukan karena komit / rollback kemampuan NoSQL



Gambar 53. Ilustrasi alur data NOSql
Sumber : http://id.wikipedia.org/wiki/Nosql

Memperkenalkan database SQL Azure

Microsoft SQL Azure adalah Database cloud berbasis platform database relasional dibangun di atas teknologi SQL Server. Dengan menggunakan Database SQL Azure, Anda dapat dengan mudah penyediaan dan menyebarkan solusi database relasional ke cloud, dan mengambil keuntungan dari pusat data terdistribusi yang menyediakan kelas enterprise ketersediaan, skalabilitas, dan keamanan dengan manfaat built-in perlindungan data dan self- penyembuhan.



Gambar. 54 Data Platform: Device to Cloud
Sumber: msdn.microsoft.com

Pengelolaan

Database SQL Azure menawarkan skala dan fungsi dari pusat data perusahaan tanpa overhead administrasi yang terkait dengan on-premis contoh dari SQL Server.

Kemampuan pengelolaan diri memungkinkan organisasi untuk penyediaan layanan data untuk aplikasi di seluruh perusahaan tanpa menambah beban dukungan departemen IT pusat atau mengganggu teknologi-cerdas karycloud dari tugas-tugas inti mereka dalam rangka untuk menjaga aplikasi database departemen.

Dengan database Azure SQL, Anda dapat penyediaan penyimpanan data Anda dalam hitungan menit. Hal ini mengurangi biaya awal layanan data dengan memungkinkan Anda untuk penyediaan hanya apa yang Anda butuhkan. Ketika kebutuhan Anda berubah, Anda dengan mudah dapat memperpanjang berbasis cloud Anda penyimpanan data untuk memenuhi kebutuhan tersebut.

Ketersediaan Tinggi

SQL Azure Database dibangun di atas Windows Server terbukti dan teknologi SQL Server, dan cukup fleksibel untuk mengatasi dengan variasi dalam penggunaan dan beban. Layanan ini ulangan beberapa salinan berlebihan data Anda ke server fisik untuk menjaga ketersediaan data dan kelangsungan usaha. Dalam kasus kegagalan hardware, Database SQL Azure menyediakan failover otomatis untuk mengoptimalkan ketersediaan untuk aplikasi Anda.

Skalabilitas

Keuntungan utama dari SQL Azure Database adalah kemudahan yang Anda dapat skala solusi Anda. Setelah partisi data Anda, skala layanan sebagai data Anda tumbuh. Sebuah membayar-as-you-tumbuh model harga memastikan bahwa Anda hanya membayar untuk penyimpanan yang Anda gunakan, sehingga Anda juga dapat menurunkan layanan ketika Anda tidak membutuhkannya.

Akrab Pengembangan Model

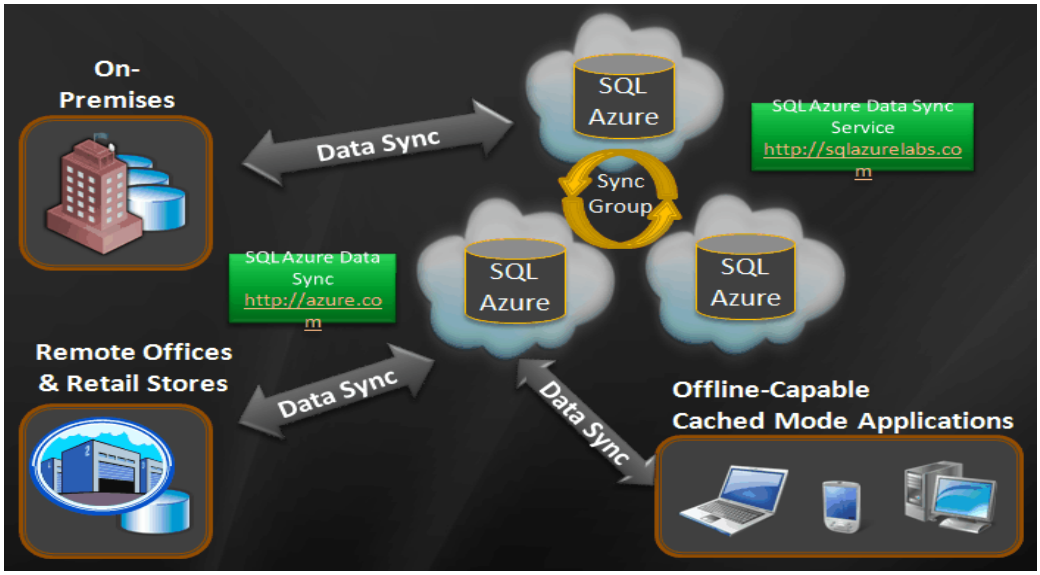
Ketika pengembang membuat on-premis aplikasi yang menggunakan SQL Server, mereka menggunakan perpustakaan klien yang menggunakan aliran data tabular (TDS) protokol untuk berkomunikasi antara klien dan server. SQL Azure Database menyediakan antarmuka TDS sama seperti SQL Server sehingga Anda dapat menggunakan alat yang sama dan perpustakaan untuk membangun aplikasi klien untuk data yang disimpan dalam database SQL Azure. Untuk lebih lanjut tentang TDS.

Model Data Relasional

Database SQL Azure akan tampak sangat akrab bagi pengembang dan administrator karena data disimpan dalam database SQL Azure persis seperti itu disimpan di SQL Server, dengan menggunakan Transact-SQL. Konseptual mirip dengan sebuah contoh on-premise dari SQL Server, SQL server Azure adalah kelompok logis database yang bertindak sebagai batas otorisasi.

Dalam setiap server Azure logis SQL, Anda dapat membuat beberapa database yang memiliki tabel, pandangan, prosedur tersimpan, indeks, dan lain objek database akrab. Model data membuat baik penggunaan desain database relasional yang ada dan Transact-SQL keterampilan pemrograman, dan menyederhanakan proses migrasi yang ada pada-premis aplikasi database ke Database SQL Azure.Untuk lebih lanjut tentang Transact-SQL dan hubungannya dengan database SQL Azure.

SQL Azure server dan database obyek virtual yang tidak sesuai dengan server fisik dan database. Dengan isolasi Anda dari pelaksanaan fisik, Database SQL Azure memungkinkan Anda untuk menghabiskan waktu pada desain database Anda.



Gambar 55. Menyimpan Data pada Azure
Sumber: msdn.microsoft.com

Microsoft SQL Database Azure adalah cloud berbasis layanan database relasional yang dibangun di atas teknologi SQL Server dan berjalan di pusat data Microsoft perangkat keras yang dimiliki, host, dan dipelihara oleh Microsoft. Dalam hal ini Database Azure SQL menjelaskan beberapa cara yang berbeda dari SQL Server.

Persamaan dan Perbedaan

Mirip dengan sebuah instance dari SQL Server di tempat anda, SQL Azure database menghadapkan aliran data tabular (TDS) antarmuka untuk Transact-SQL berbasis mengakses database. Hal ini memungkinkan aplikasi database menggunakan SQL database Azure dengan cara yang sama yang mereka gunakan SQL Server. Karena

SQL Azure Database adalah layanan administrasi dalam Database Azure SQL sedikit berbeda.

Karena Microsoft menangani semua administrasi fisik, ada beberapa perbedaan antara database SQL Azure dan turunan on-premis SQL Server dalam hal administrasi, pengadaan, Transact-SQL dukungan, model pemrograman, dan fitur.

Provisioning

Ketika mempersiapkan penyebaran lokal SQL Server, mungkin peran DBA atau departemen TI untuk mempersiapkan dan mengkonfigurasi hardware yang dibutuhkan dan perangkat lunak. Bila menggunakan Database SQL Azure, tugas ini dilakukan oleh proses Azure SQL provisioning.

Anda dapat mulai penyediaan database SQL Anda Azure setelah Anda membuat account platform Windows Azure. Akun ini memungkinkan Anda untuk mengakses semua layanan, seperti Windows Azure, Windows Azure AppFabric, dan Database SQL Azure, dan digunakan untuk mengatur dan mengelola langganan Anda.

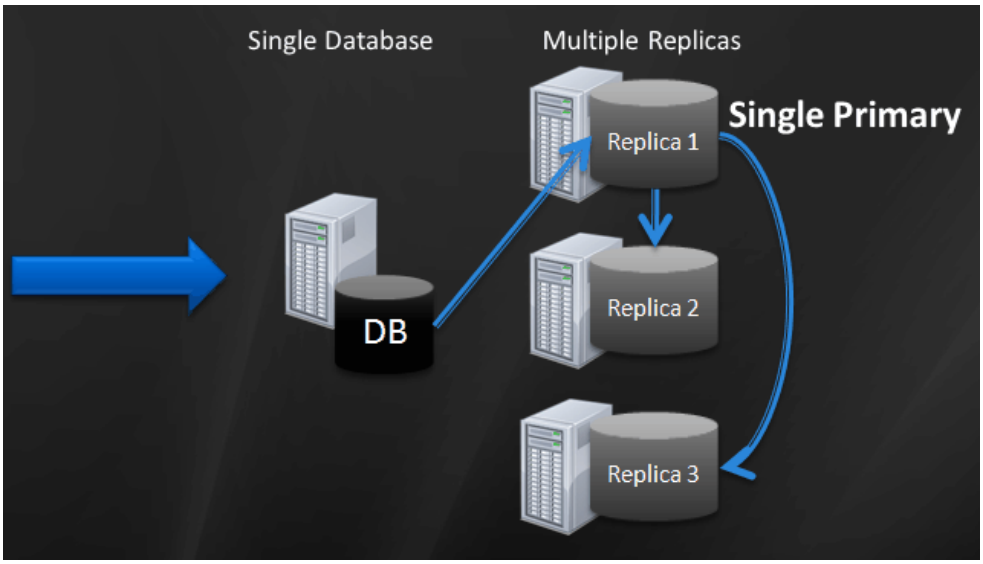
Setiap SQL Azure berlangganan dapat terikat pada satu atau lebih Azure SQL server di pusat data Microsoft. Anda SQL Azure server merupakan abstraksi yang mendefinisikan pengelompokan database. Untuk mengaktifkan load balancing dan ketersediaan tinggi, database yang terkait dengan server Azure SQL dapat berada pada komputer fisik yang terpisah di pusat data Microsoft. Untuk informasi lebih lanjut tentang pengadaan.

Transact-SQL Dukungan

Banyak SQL Server Transact-SQL memiliki parameter yang memungkinkan Anda untuk menentukan kelompok file atau path file fisik.Jenis parameter tidak didukung dalam Database SQL Azure karena mereka memiliki dependensi pada konfigurasi fisik. Dalam kasus tersebut, perintah ini dianggap sebagian didukung. Untuk informasi lebih lanjut tentang Transact-SQL.

Fitur dan Jenis

Database SQL Azure tidak mendukung semua fitur dan jenis data yang ditemukan di SQL Server. Jasa Analisis, Replikasi, dan Layanan Broker saat ini tidak disediakan sebagai pelayanan pada platform Windows Azure.

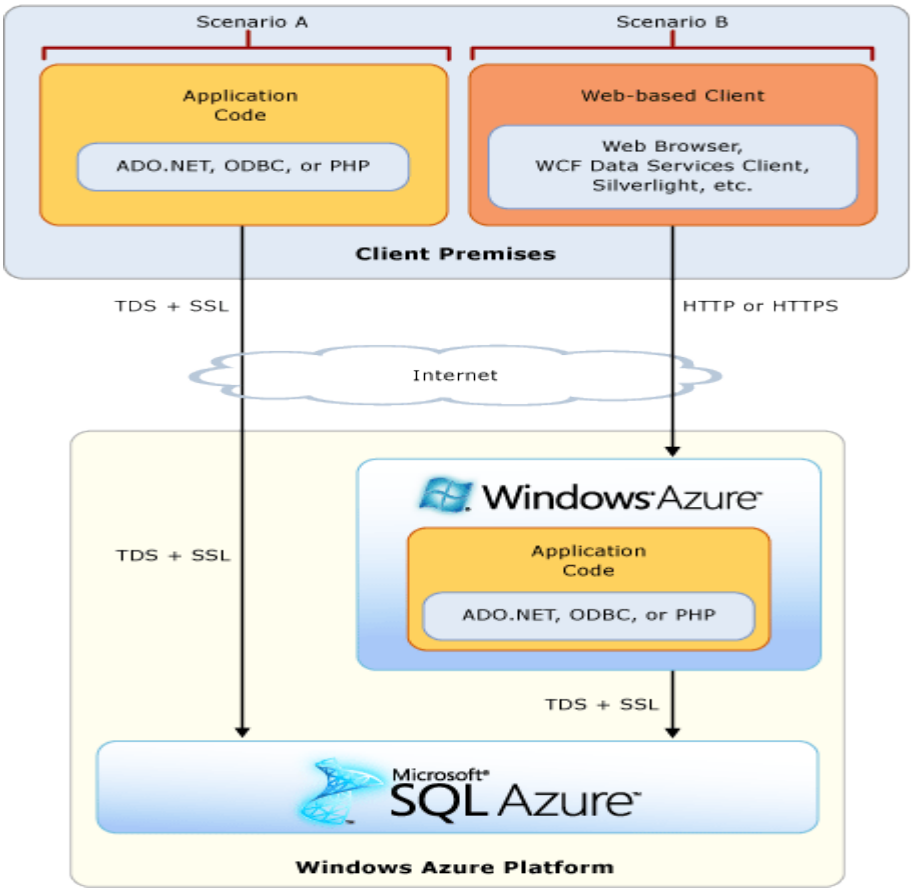


Gambar 56. Database Replicas
Sumber: msdn.microsoft.com

Karena Database SQL Azure melakukan administrasi fisik, pernyataan dan pilihan yang mencoba untuk secara langsung memanipulasi sumber daya fisik akan diblokir, seperti referensi file group, dan beberapa pernyataan DDL server fisik. Hal ini juga tidak mungkin untuk mengatur opsi server dan melacak SQL atau menggunakan SQL Server Profiler atau utilitas Tuning Advisor database. Database SQL Server Azure mendukung banyak tipe data SQL 2008, tetapi tidak mendukung tipe data yang telah usang dari SQL Server 2008.

Mengakses data di SQL Azure

Dalam aplikasi on-premise tradisional, kode aplikasi dan database yang terletak di pusat data yang sama fisik. SQL Azure dan platform Windows Azure menawarkan banyak alternatif untuk arsitektur itu. Diagram berikut menunjukkan dua alternatif umum yang tersedia untuk bagaimana aplikasi Anda dapat mengakses data dengan Database SQL Azure.



Gambar 57. Scenario akses data di SQL Azure
Sumber: msdn.microsoft.com

Hosting Data Anda di Cloud

Penampilan dan fungsionalitas dari pengalaman database SQL Server anda tidak berubah ketika Anda menggunakan SQL Azure bukan sebuah contoh premis di SQL Server. Ada beberapa perbedaan kecil yang mencakup Transact-SQL perubahan dan manajemen login.Untuk informasi lebih lanjutKetika data Anda hosting hanya di pusat data Microsoft, penting untuk mempertimbangkan latensi jaringan antara SQL Azure dan on-premise server menjalankan kode aplikasi Anda. Latensi jaringan ke SQL Azure adalah lebih tinggi dari itu akan jika database juga di tempat Anda dan Anda mungkin perlu untuk memperhitungkan ini dalam kode aplikasi Anda.

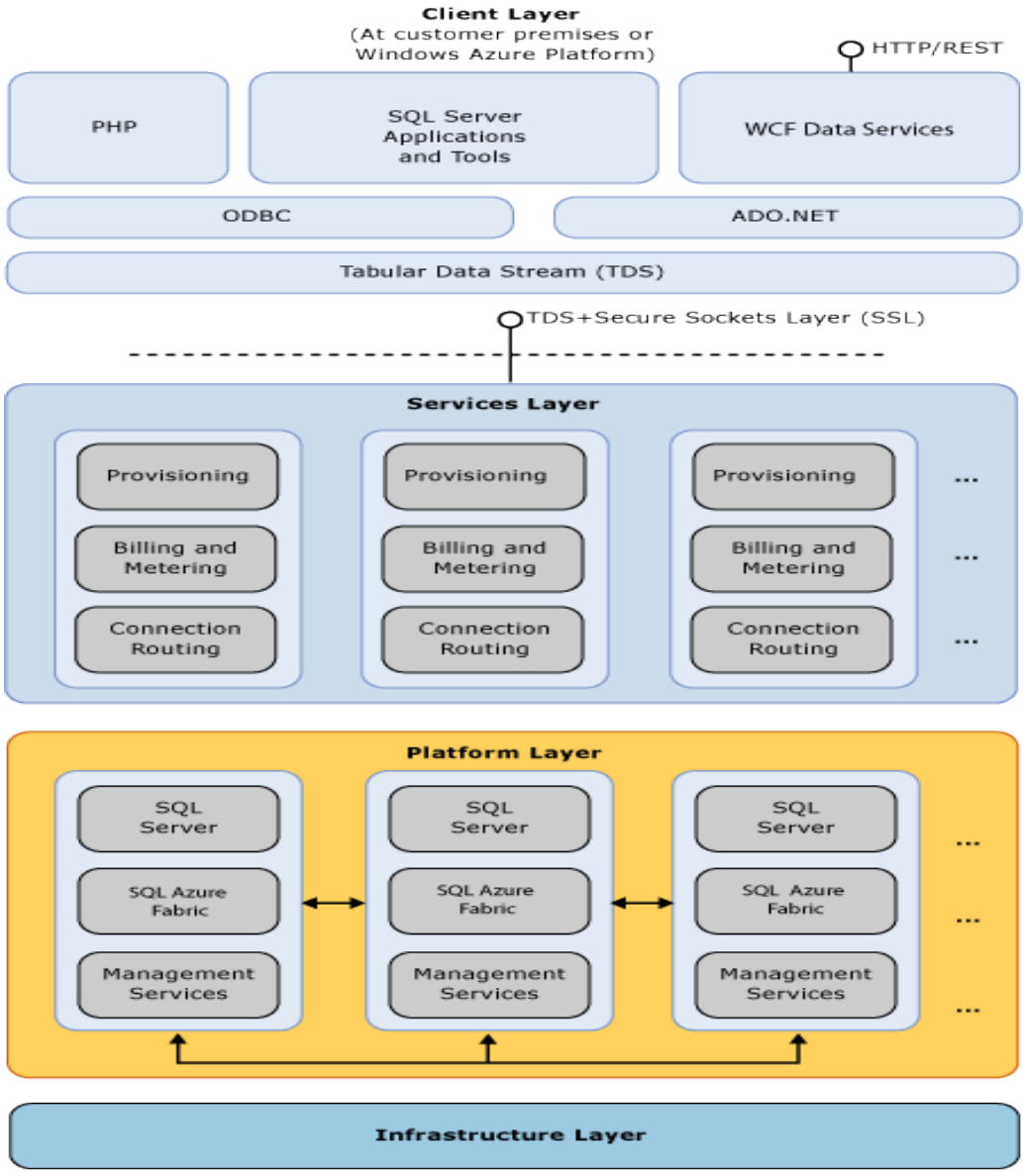
Hosting Aplikasi Anda di Cloud

Aplikasi Hosting di Windows Azure adalah opsional. Berdasarkan pada kebutuhan Anda, Anda dapat bermigrasi aplikasi Anda ke atau dari Windows Azure. Untuk informasi lebih lanjut tentang komputasi cloud. Hosting kode aplikasi Anda dalam

Windows Azure adalah bermanfaat bagi kinerja aplikasi Anda karena meminimalkan latency jaringan yang terkait dengan permintaan data aplikasi Anda ke SQL Azure.

SQL Azure Arsitektur

Microsoft SQL Azure Database host pada server yang menjalankan teknologi SQL Server di pusat data Microsoft. Dari perspektif arsitektur, ada empat lapisan yang berbeda abstraksi yang bekerja sama untuk menyediakan sebuah database relasional untuk aplikasi Anda untuk menggunakan: lapisan klien, lapisan layanan, lapisan platform, dan lapisan infrastruktur. Seperti yang terlihat dalam diagram berikut :



Gambar 58. SQL Azure Arsitektur
Sumber: msdn.microsoft.com

Layer Klien

Lapisan klien berada paling dekat dengan aplikasi Anda, dan digunakan oleh aplikasi Anda untuk berkomunikasi secara langsung dengan SQL Azure. Lapisan klien dapat berada pada-premis di datacenter Anda atau host di Windows Azure. Karena SQL Azure menyediakan aliran tabel data yang sama (TDS) antarmuka seperti SQL Server, Anda dapat menggunakan alat akrab dan perpustakaan untuk membangun aplikasi klien untuk data yang di cloud. Ada banyak pilihan yang tersedia untuk aplikasi Anda untuk berkomunikasi dengan SQL Azure.

Layer Layanan

Lapisan layanan berfungsi sebagai gateway antara lapisan klien dan lapisan platform, di mana data berada. Lapisan layanan menyediakan tiga fungsi: provisioning, penagihan dan metering, dan routing koneksi. Lapisan layanan ketentuan database yang Anda tentukan dengan accounts platform Windows Azure. Aspek penagihan dan metering dari lapisan layanan memungkinkan multi-penyewa dukungan dengan menyediakan monitoring dan penagihan untuk penggunaan database berdasarkan accounts individu Azure platform Windows. SQL Azure dibangun pada platform scalable yang melibatkan banyak server fisik; lapisan ini menangani semua koneksi routing antara aplikasi dan server fisik dimana data anda berada.

Layer platform

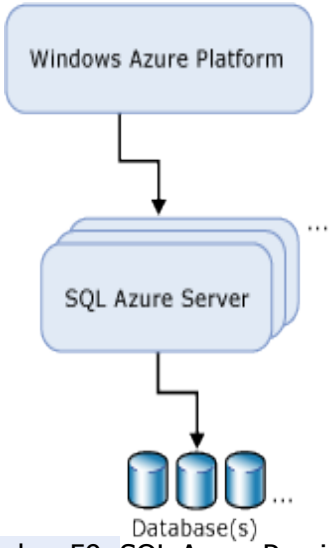
Lapisan Platform meliputi server fisik dan layanan yang mendukung lapisan layanan. Lapisan Platform terdiri dari banyak contoh dari SQL Server, masing-masing dikelola oleh kain Azure SQL. SQL Azure adalah sebuah sistem komputasi terdistribusi terdiri dari jaringan terintegrasi, server, dan penyimpanan. Hal ini memungkinkan failover otomatis, load balancing, dan replikasi otomatis antara server fisik. Jasa manajemen memantau kesehatan server individu dan memungkinkan instalasi otomatis upgrade layanan dan patch perangkat lunak.

Lapisan Infrastruktur

Lapisan infrastruktur merupakan administrasi TI perangkat keras fisik dan sistem operasi yang mendukung lapisan layanan.

SQL Azure Provisioning Model

Diagram berikut menunjukkan hubungan antara accounts platform Windows Azure, SQL Azure server, dan database.



Gambar 59. SQL Azure Provisioning Model
Sumber: msdn.microsoft.com

Seperti ditunjukkan dalam diagram, masing-masing accounts platform Windows Azure dapat dikaitkan dengan beberapa Azure SQL server. Setiap SQL Azure server dapat dikaitkan dengan satu atau lebih database.

Server

Sebuah SQL Azure server adalah sebuah kelompok logis dari database dan bertindak sebagai titik pusat untuk administrasi database multipel. Setiap SQL Server Azure termasuk login sama dengan yang di contoh dari SQL Server di tempat anda. Setiap SQL Azure server memiliki nama domain berkualifikasi lengkap yang unik, yang dihasilkan selama proses server provisioning Azure SQL.

Database

Setiap server SQL Azure dapat berisi database multipel. Dalam setiap database, Anda dapat membuat tabel, pandangan, indeks, prosedur tersimpan, dan lain objek database. Proses provisioning menciptakan read-only **Master** database secara otomatis. **Master** Database melacak yang login memiliki izin untuk membuat database atau login lainnya. Anda harus terhubung ke **Master** database yang setiap kali Anda MENCIPTAKAN , ALTER , atau DROP login atau database. **Master** database juga menyediakan metrik penggunaan SQL Azure yang dapat Anda lihat.

Login

Selama proses provisioning, SQL Azure menciptakan login untuk Anda yang pokok server tingkat server SQL Azure. Principal tingkat server mirip dengan **sa** login di SQL Server. Tambahan SQL Azure database dan login kemudian dapat dibuat di server, sesuai kebutuhan.Untuk informasi lebih lanjut tentang **sa** login di sebuah instance on-premise SQL Server.

SQL Azure menyediakan set yang sama keamanan yang tersedia di SQL Server, seperti:

- Login SQL Server: Digunakan untuk autentikasi akses ke SQL Azure di tingkat server.
- Pengguna database : Digunakan untuk memberikan akses ke SQL Azure pada tingkat database.
- Peran Database: Digunakan untuk pengguna kelompok dan memberikan akses ke SQL Azure pada tingkat database.

Database-tingkat administrasi bagi pengguna dan peran di SQL Azure adalah sama seperti untuk sebuah contoh premis-SQL Server.

SQL Azure Edisi

SQL Azure database yang tersedia dalam dua edisi: Edisi Web Database dan Database Edisi Bisnis. Kedua edisi menawarkan skalabilitas, ketersediaan tinggi otomatis, dan self-provisioning.

- Database Edisi Web paling cocok untuk aplikasi Web kecil dan aplikasi workgroup atau departemen. Edisi ini mendukung database dengan ukuran maksimal 1 atau 5 GB data.
- Database Business Edition sangat cocok untuk vendor software independen (ISV), aplikasi line-of-business (LOB), dan aplikasi perusahaan. Edisi ini mendukung database hingga 50 GB data, dengan penambahan sebesar 10 GB.

Kedua edisi membebankan biaya bandwidth yang berbasis tambahan ketika transfer data termasuk klien luar platform Windows Azure atau di luar wilayah dari database SQL Azure.

Bandwidth Tabungan pada Platform Windows Azure

Bandwidth yang digunakan antara SQL Azure dan Windows Azure atau Windows Azure AppFabric bebas dalam yang sama sub-wilayah atau pusat data. Ketika merancang sistem anda, mengaturnya sehingga operasi menimbulkan transfer data yang besar berada dalam platform Windows Azure dan sama sub-wilayah. Ketika deploying sebuah aplikasi Windows Azure, cari aplikasi dan database SQL Azure dalam sub-wilayah yang sama untuk menghindari biaya bandwidth.

Membatasi Ukuran Database dengan Setting maxsize

Bila Anda memilih edisi database, Anda juga pilih maxsize tersebut. Maxsize menyediakan kemampuan untuk membatasi ukuran database. Jika ukuran database mencapai maxsize yang Anda akan menerima kode galat 40544. Ketika ini terjadi, Anda tidak dapat memasukkan atau memperbarui data, atau membuat objek baru (seperti tabel, prosedur tersimpan, pandangan, dan fungsi). Namun, Anda masih dapat membaca dan menghapus data, tabel truncate, drop tabel dan indeks, dan membangun kembali indeks. Anda kemudian dapat memperbarui maxsize ke nilai lebih besar dari ukuran database saat ini Anda atau menghapus beberapa data untuk ruang penyimpanan gratis. Mungkin ada sebanyak penundaan lima belas menit sebelum Anda dapat memasukkan data baru. Untuk Edisi Web, nilai-nilai yang valid maxsize adalah: 1GB atau 5GB. Untuk Edisi Bisnis, nilai-nilai yang valid adalah: 10GB, 20GB, 30GB, 40GB, atau 50GB. Ukuran database meliputi data, indeks, dan objek. Ukuran tidak termasuk log, tabel sistem, katalog server, master database, atau replika tambahan. Jika Anda melebihi kapasitas penyimpanan, layanan menolak sisipan tambahan. Database Anda tidak akan secara otomatis dikonversi ke edisi yang mendukung menyisipkan tambahan.

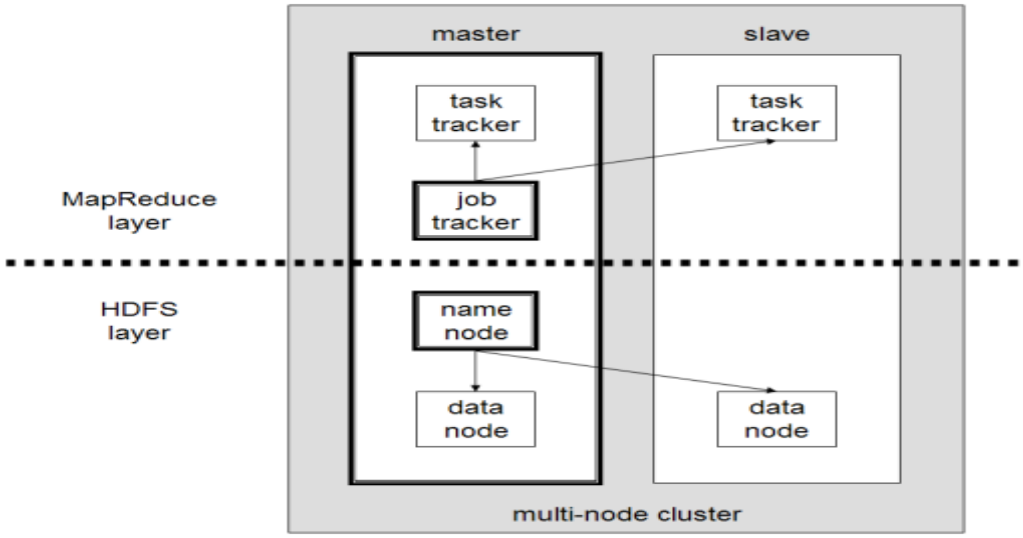
Apache Hadoop

Hadoop adalah framework software berbasis Java dan opensource yang berfungsi untuk mengolah data yang sangat besar secara terdistribusi dan berjalan di atas cluster yang terdiri dari beberapa komputer yang saling terhubung. Hadoop dapat mengolah data dalam jumlah yang sangat besar hingga petabyte dan dijalankan di atas ribuan komputer. Terinspirasi dari whitepaper Google yang membahas tentang **MapReduce** dan **Google File System** .

Hadoop adalah top-level Apache proyek yang dibangun dan digunakan oleh komunitas global kontributor, menggunakan Java bahasa pemrograman. Yahoo! telah menjadi penyumbang terbesar terhadap proyek, dan menggunakan Hadoop luas di seluruh usahanya. Dibuat oleh **Doug Cutting** dan nama hadoop ini berasal dari boneka gajah milik anaknya. Asal muasalnya hadoop ini adalah sub project dari **Nutch** yang digunakan untuk Search Engine. Hadoop bersifat open source dan berada dibawah bendera **Apache Software Foundation**.

Arsitektur

Hadoop terdiri dari *Common Hadoop* , yang menyediakan akses ke filesystem yang didukung oleh Hadoop. Common Hadoop berisi paket yang diperlukan JAR file dan skrip yang dibutuhkan untuk memulai Hadoop. Paket ini juga menyediakan kode sumber, dokumentasi, dan bagian kontribusi yang mencakup proyek dari Komunitas Hadoop. Untuk penjadwalan efektif kerja, setiap yang kompatibel filesystem Hadoop harus memberikan kesadaran lokasi: nama rak (lebih tepatnya, switch jaringan) di mana node pekerja ini. aplikasi Hadoop dapat menggunakan informasi ini untuk menjalankan pekerjaan pada node dimana data tersebut, gagal pada rak yang sama / switch, sehingga mengurangi lalu lintas backbone. Hadoop ini Distributed File System (HDFS) menggunakan ini, jika data replikasi untuk mencoba menyimpan salinan yang berbeda dari data pada rak yang berbeda.Tujuannya adalah untuk mengurangi dampak dari pemadaman listrik rak atau beralih kegagalan sehingga bahkan jika peristiwa ini terjadi, data masih dapat dibaca. Seperti pada gambar dibawah ini.



Gambar 60. Hadoop cluster multi-node
Sumber : http://id.wikipedia.org/wiki/Hadoop

Sebuah cluster Hadoop kecil akan mencakup node master tunggal dan beberapa pekerja. Node master terdiri dari jobtracker, tasktracker, name node, dan data node. Slave atau *node pekerja* terdiri dari data node dan tasktracker, meskipun adalah mungkin untuk memiliki hanya pekerja node-data, dan hanya pekerja node-menghitung; ini biasanya hanya digunakan dalam aplikasi non-standardad. Hadoop memerlukan JRE 1.6 atau lebih tinggi. Startup standar dan skrip shutdown membutuhkan ssh yang akan dibentuk antara node di cluster. Sementara Microsoft Windows dan OS / X yang didukung untuk pembangunan, per April 2011 tidak ada klaim publik yang ini sedang digunakan di server besar.

Dalam sebuah cluster yang lebih besar, filesystem HDFS dikelola melalui server namenode didedikasikan untuk host indeks filesystem, dan sebuah namenode sekunder yang dapat menghasilkan snapshot dari struktur memori namenode, sehingga mencegah korupsi sistem berkas dan mengurangi hilangnya data. Demikian pula, server jobtracker mandiri dapat mengelola penjadwalan job. Dalam cluster dimana Hadoop MapReduce mesin digunakan melcloud filesystem alternatif, namenode itu, namenode sekunder dan arsitektur datanode dari HDFS digantikan oleh setara filesystem-spesifik.

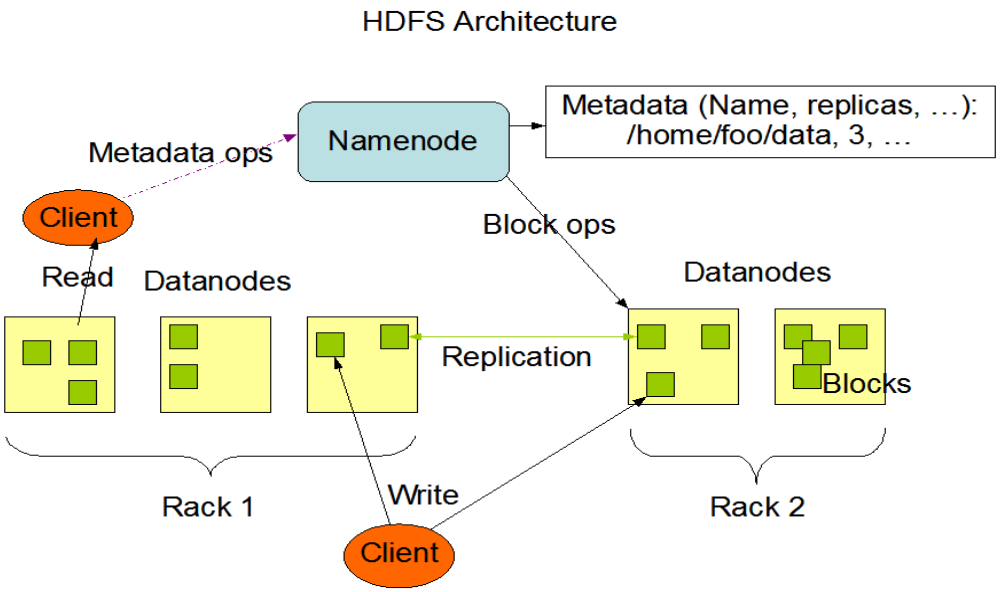
Inti dari hadoop adalah terdiri dari :

- HDFS (Hadoop Distributed File System) - Data yang terdistribusi
- MapReduce - Framework dari aplikasi yang terdistribusi

HDFS adalah filesystem yang berbasis Java, yang meyimpan file dalam jumlah besar dan disimpan secara terdistribusi didalam banyak komputer yang saling berhubungan (file dipecah dalam bentuk block sebesar 64MB - bisa diconfigure besarnya). Data direplikasi umumnya ke dalam 3 node, dua di rack yang sama dan satu di rack yang berbeda, hal ini bertujuan untuk menjaga realibility dari HDFS. Filesystem ini membutuhkan server induk yang dinamakan masternode, berfungsi untuk menyimpan metadata dari data yang ada didalam HDFS. Datanya sendiri disimpan didalam server-server yang dinamakan datanode yang dapat diakses dengan menggunakan protokol HTTP. Data nodes ini bisa saling berkomunikasi satu sama lain untuk menjaga konsistensi data dan memastikan proses replikasi data berjalan dengan baik.

Kelemahan dari HDFS ini adalah masternode masih bersifat Single Point of Failure sehingga apabila server masternode mati maka data akan hilang. Memang didalam HDFS ada yang dinamakan secondary namenode yang seringkali menimbulkan kesalahpahaman pengertian bahwa apabila primary namenode mati maka akan langsung digantikan oleh secondary namenode. Padahal secondary namenode ini hanya menyimpan informasi terbaru dari struktur direktori pada primary namenode. Salah satu jalan untuk mengantisipasi problem SPOF pada primary namenode adalah dengan membuat cloning dari server namenode ini di server yang berbeda, sehingga apabila server primary namenode mengalami masalah/mati dapat langsung digantikan oleh cloningnya.

Keuntungan dari menggunakan HDFS adalah data kesadaran antara jobtracker dan tasktracker. The jobtracker jadwal peta / mengurangi pekerjaan untuk tasktrackers dengan kesadaran dari data lokasi. Contoh ini akan jika data yang terdapat node A (x, y, z) dan data yang terdapat node B (a, b, c). jobtracker akan jadwal node B untuk melakukan peta / mengurangi tugas pada (a, b, c) dan node A akan dijadwalkan untuk melakukan peta / mengurangi tugas pada (x, y, z). Hal ini akan mengurangi jumlah lalu lintas yang berjalan di atas jaringan dan mencegah transfer data yang tidak perlu. Ketika Hadoop digunakan dengan filesystem lain keunggulan ini tidak tersedia. Ini dapat memiliki dampak yang signifikan terhadap kinerja waktu penyelesaian pekerjaan, yang telah ditunjukkan ketika menjalankan data pekerjaan intensif.



Gambar 61. HDFS Architecture
Sumber : http://id.wikipedia.org/wiki/Hadoop

MapReduce adalah framework software yang diperkenalkan oleh Google dan digunakan untuk mendukung distributed computing yang dijalankan di atas data yang sangat besar dan dijalankan secara simultan dibanyak komputer. Framework ini terinspirasi oleh konsep fungsi map dan reduce yang biasa digunakan di functional programming.

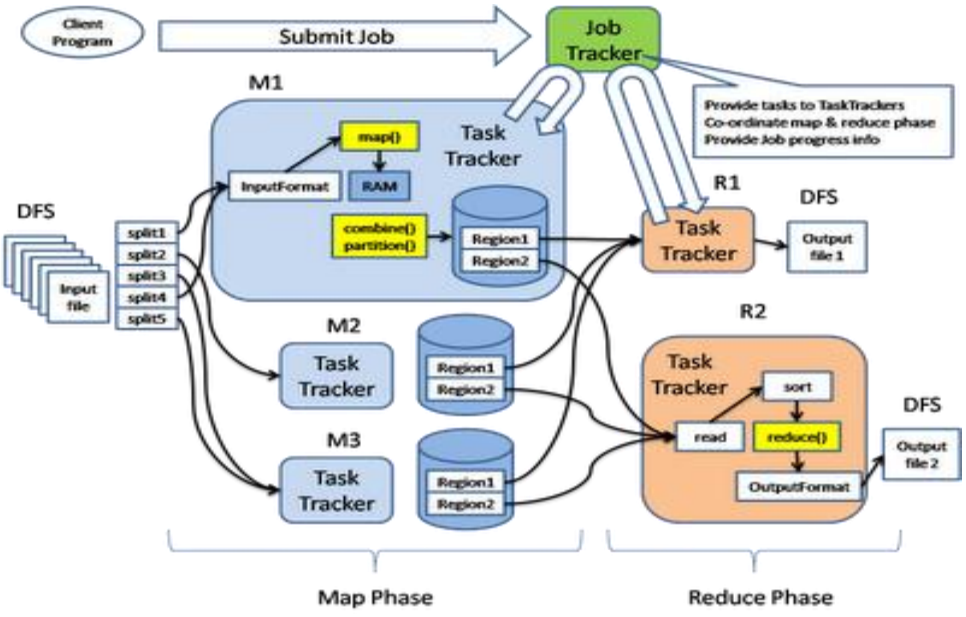
Proses "Map" : masternode menerima input, kemudian input tersebut dipecah menjadi beberapa sub problem yang kemudian didistribusikan ke worker nodes. Worker nodes ini akan memproses sub problem yang diterimanya untuk kemudian apabila problem tersebut sudah diselesaikan, maka akan dikembalikan ke masternode.

Proses "Reduce" : Masternode menerima jawaban dari semua sub problem dari banyak data nodes, menggabungkan jawaban-jawaban tersebut menjadi satu jawaban besar untuk mendapatkan penyelesaian dari permasalahan utama. Keuntungan dari MapReduce ini adalah proses map and reduce dijalankan secara terdistribusi. Dalam setiap proses mapping bersifat independen sehingga proses dapat dijalankan secara simultan dan paralel. Demikian pula dengan proses reducer dapat dilakukan secara paralel diwaktu yang sama, selama output dari operasi mapping mengirimkan key value yang sesuai dengan proses reducernya. Proses MapReduce ini dapat diaplikasikan di cluster server yang jumlahnya sangat banyak sehingga dapat

mengolah data dalam jumlah petabyte hanya dalam waktu beberapa jam. Didalam hadoop, mapreduce engine ini terdiri dari satu jobtracker dan satu/banyak tasktracker. JobTracker adalah server penerima job dari client, sekaligus akan mendistribusikan jobs tersebut ke tasktracker yang akan mengerjakan sub job sesuai yang diperintahkan jobtracker. Strategi ini akan mendekatkan pengolahan data dengan datanya sendiri, sehingga ini akan sangat signifikan mempercepat proses pengolahan data.

HDFS filesystem tidak hanya bisa digunakan untuk map/reduce. Ada beberapa project lain yang related dengan hadoop yang dapat dijalankan diatas HDFS seperti HBase, Pig, Hive, dll.

Hadoop ditujukan sebagian besar adalah untuk mengolah data dalam jumlah besar secara cepat dan efisien. Prosesnya sendiri adalah dengan menggunakan batch processing dan belum ditujukan untuk realtime processing, meskipun sekarang banyak aplikasi yang berjalan diatas hadoop yang sudah memungkinkan proses mendekati realtime.



Gambar 62. Architecture MapReduce
http://id.wikipedia.org/wiki/Hadoop

Peggunaan Hadoop

Pada tanggal 19 Februari 2008, Yahoo! Inc meluncurkan apa yang diklaim itu Hadoop aplikasi produksi terbesar dunia. Yahoo! Search WebMap Hadoop adalah aplikasi yang berjalan pada lebih dari 10.000 core Linux cluster dan menghasilkan data yang sekarang digunakan dalam setiap permintaan pencarian Web Yahoo!.

Ada beberapa Hadoop cluster di Yahoo!, masing-masing satu sebuah data center menempati (atau fraksinya). Tidak ada filesystem HDFS atau pekerjaan MapReduce terbagi di beberapa pusat data, melainkan data center masing-masing memiliki filesystem terpisah dan beban kerja. Server cluster menjalankan Linux, dan dikonfigurasi pada boot menggunakan Kickstart. Setiap mesin bootstraps gambar Linux, termasuk distribusi Hadoop. Pekerjaan bahwa cluster lakukan adalah dikenal untuk menyertakan perhitungan indeks untuk mesin pencari Yahoo.

Pada tanggal 10 Juni 2009, Yahoo tersedia source code untuk versi Hadoop berjalan dalam produksi. Sementara Yahoo! memberikan kontribusi kembali semua pekerjaan itu tidak pada Hadoop ke komunitas open source, perusahaan pengembang juga memperbaiki bug dan memberikan peningkatan stabilitas internal, dan rilis ini source code ditambal sehingga pengguna lain dapat manfaat dari usaha mereka.

Pengguna lain

Selain Yahoo!, organisasi lainnya yang menggunakan Hadoop untuk menjalankan perhitungan terdistribusi besar. Beberapa pengguna penting termasuk:

- 1 & 1
- A9.com
- Amazon.com
- AOL
- Apple
- Booz Allen Hamilton
- EHarmony
- eBay
- Facebook
- Fox Interactive Media
- Freebase
- Hewlett-Packard

- IBM
- ImageShack
- ISI
- Joost
- Last.fm
- LinkedIn
- Meebo
- Metaweb

- NetFlix
- New York Times
- Ning
- Powerset (sekarang bagian dari Microsoft)
- Rackspace
- Razorfish
- StumbleUpon
- Twitter

Hadoop pada layanan EC2/S3 Amazon

Hal ini mungkin untuk menjalankan Hadoop di Amazon Elastic Compute Cloud (EC2) dan Amazon Simple Storage Service (S3). Sebagai contoh The New York Times digunakan 100 contoh Amazon EC2 dan aplikasi Hadoop untuk memproses 4TB gambar mentah TIFF data (disimpan dalam S3) ke 11 juta selesai file PDF dalam waktu 24 jam dengan biaya perhitungan sekitar $ 240 (tidak termasuk bandwidth).

Ada dukungan untuk filesystem S3 di distribusi Hadoop, dan tim Hadoop menghasilkan gambar mesin EC2 setelah setiap rilis. Dari perspektif kinerja murni, Hadoop pada S3/EC2 tidak efisien, sebagai filesystem S3 sangat kecil dan penundaan kembali dari setiap operasi penulisan data sampai dijamin tidak akan hilang. Ini menghapus lokalitas keuntungan dari Hadoop, yang jadwal kerja data dekat untuk menghemat beban jaringan.

Pada April 2, 2009 Amazon mengumumkan rilis beta dari layanan baru yang disebut Amazon Elastic MapReduce yang mereka gambarkan sebagai "sebuah layanan web yang memungkinkan bisnis, peneliti, analis data, dan pengembang untuk dengan mudah dan efektif biaya proses sejumlah besar data. Ini memanfaatkan kerangka host

Hadoop berjalan pada skala infrastruktur web Amazon elastis Hitung Cloud (Amazon EC2) dan Amazon Wikipedia Storage Service (Amazon S3).

Hadoop di Google dan IBM

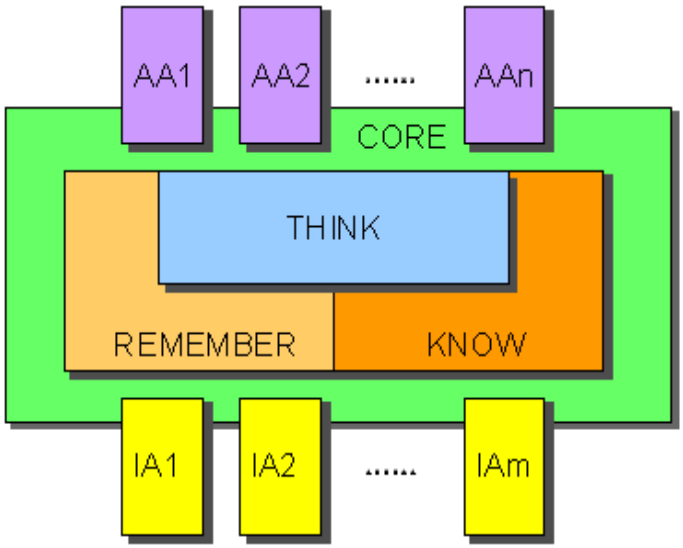
IBM dan Google mengumumkan inisiatif pada tahun 2007 untuk menggunakan Hadoop untuk mendukung program universitas dalam pemrograman komputer terdistribusi.
Pada tahun 2008 kolaborasi ini, Academic Cloud Computing Initiative (kecelakaan), bermitra dengan National Science Foundation untuk menyediakan dana hibah untuk peneliti akademis tertarik untuk menjelajahi-data aplikasi yang besar. Hal ini mengakibatkan penciptaan (eksplorasi PETUNJUK) program Cluster

Grid Engine Integrasi

Integrasi dengan Sun Grid Engine dirilis pada tahun 2008, dan menjalankan Hadoop pada Sun Grid (-demand Sun pada utilitas komputasi pelayanan) itu mungkin. Dalam implementasi awal integrasi, waktu scheduler CPU tidak memiliki pengetahuan tentang lokalitas data. Sayangnya, ini berarti bahwa pengolahan tidak selalu dilakukan pada rak yang sama dengan data, ini adalah fitur utama dari Runtime Hadoop. Sebuah integrasi ditingkatkan dengan data-lokalitas diumumkan pada Sun HPC Perangkat Lunak. Pada 2008-2009 Sun merilis *Hadoop Live CD* OpenSolaris proyek, yang memungkinkan menjalankan Hadoop cluster fungsional sepenuhnya menggunakan live CD.

Apa itu Cassandra?

Cassandra adalah sebuah platform penelitian yang dikembangkan oleh KnowGravity Inc di Zurich, Swiss. Ini adalah asisten-based dan memandu pengguna melalui tugas-tugas mereka dalam bisnis, sistem dan rekayasa perangkat lunak. Ini analisis data proyek dari banyak akrab UML ® berbasis CASE tools, menghasilkan pertanyaan-pertanyaan yang sesuai atau mengusulkan langkah-langkah berikutnya dalam rangka untuk bergerak proses ke depan. Cassandra hadir dalam dua varian. Cassandra dan Cassandra / CS. CS adalah singkatan dari Common Sense. satunya perbedaan antara keduanya adalah bahwa Cassandra / CS mencakup memakan memori yang lebih komponen. Gambar berikut memberikan gambaran singkat dari komponen teknologi utama Cassandra.



Gambar 63. komponen teknologi utama Cassandra
http://wiki.apache.org/cassandra

Cassandra dapat dibandingkan dengan arsitektur sistem operasi yang menyediakan layanan dasar untuk berbagai aplikasi (Agen Aplikasi Cassandra) dan menawarkan antarmuka untuk berbagai produk (Agen Antarmuka Cassandra). Cassandra adalah 100% diterapkan di WIN-PROLOG (Programming Logic Associates Ltd).

Apache Cassandra adalah sebuah database management system yang terdistribusi yang didesain untuk menangani data dalam jumlah yang sangat besar yang tersebar di berbagai commodity server. Cassandra merupakan kombinasi dari teknologi sistem terdistribusi dari Amazon Dynamo dan data model dari Google BigTable. Seperti Dynamo karena Cassandra juga eventually consistent. Seperti BigTable karena Cassandra menyediakan data model Berbasis ColumnFamily yang lebih baik dari sistem Key/Value biasa. Cassandra pertama kali dikembangkan oleh Facebook. Sebagai gambaran, Facebook merupakan platform social networking terbesar yang melayani ratusan juta pengguna dimana pada waktu puncak penggunaan (peak time), Facebook mengerahkan puluhan ribu mesin server yang dimilikinya yang tersebar di banyak data center di berbagai belahan dunia.

Cassandra sebenarnya pada awalnya didesain untuk mengatasi masalah yang terjadi pada Inbox Search Facebook. Inbox Search merupakan fitur yang memungkinkan pengguna untuk melakukan pencarian pada Facebook Inbox. Bagi Facebook ini artinya sistem harus mampu menangani performa penulisan data yang tinggi, jutaan penulisan data per hari, dan dari jutaan pengguna. Semenjak pengguna dilayani dari data center yang terdistribusi secara geografis, kemampuan untuk mereplika data antar data center merupakan kunci untuk menjaga latency (waktu) pencarian menjadi turun (cepat). Saat ini Inbox Search telah digunakan oleh lebih dari 250 juta pengguna dari sejak fitur ini diluncurkan, dimana pada saat itu pengguna Facebook kurang lebih 100 juta pengguna. Dan Cassandra telah menunjukkan performa yang sangat baik sampai saat ini walaupun dengan angka pengguna yang sedemikian besar. Cassandra saat ini telah digunakan sebagai backend storage service untuk banyak layanan di dalam Facebook.

Fitur-fitur yang dimiliki Cassandra antara lain:

- Decentralized : setiap node di dalam cluster adalah identik.
- Fault-tolerant : data secara otomatis akan direplikasi ke banyak node. Juga mendukung replikasi antar data center.
- Tunable Consistency
- Elasticity

Cassandra vs HBase : Pertempuran NoSQL

Didistribusikan, scalable database sangat diperlukan saat ini. Dari gudang besar data pada startup media sosial, "Data Besar" menjadi lebih penting setiap hari. Sementara Hadoop telah muncul sebagai *defacto* standar untuk penanganan masalah besar data, ada masih cukup database terdistribusi di luar sana dan masing-masing memiliki kekuatan unik mereka.

Dua database telah mengumpulkan perhatian paling besar adalah ***HBase dan Cassandra***. Perpecahan antara proyek-proyek ini sama-sama ambisius dapat dikategorikan ke dalam Fitur (hal yang hilang yang dapat ditambahkan setiap saat), dan Arsitektur (perbedaan mendasar yang tidak dapat dikodekan). HBase adalah klon dekat dari Bigtable Google, sedangkan Cassandra dimaksudkan untuk menjadi "Bigtable / Dynamo hybrid".

HBase adalah database yang lebih kuat untuk mayoritas penggunaan-kasus. Cassandra sebagian besar bergantung pada pasangan kunci-nilai untuk penyimpanan, dengan struktur tabel seperti ditambahkan untuk membuat struktur data yang lebih kuat mungkin. Dan fakta bahwa orang-orang yang jauh lebih banyak menggunakan HBase dari Cassandra saat ini, meskipun keduanya serupa.

Perbandingan Fitur beberapa dari kedua data stores:

Pengolahan

HBase adalah bagian dari ekosistem Hadoop, begitu banyak berguna kerangka kerja pemrosesan terdistribusi pendukungnya: Cascading, Hive, dll Hal ini membuat mudah untuk melakukan analisis data yang kompleks tanpa menggunakan tangan-coding. Efisien berjalan MapReduce pada Cassandra, di sisi lain, sulit karena semua kunci yang berada di salah satu "ruang" besar, sehingga kerangka MapReduce tidak tahu bagaimana untuk membagi dan membagi data aslinya. Perlu ada beberapa Hackery di tempat untuk menangani semua itu.

Bahkan, inilah beberapa kode dari sebuah patch Integrasi Cassandra / Hadoop:

+ / *

+ FIXME Ini pada dasarnya adalah kludge besar karena kita diperlukan akses ke

Cassandra + internal, dan akses yang diperlukan untuk internal Hadoop sehingga kita

+ Harus boot Cassandra ketika kita menjalankan Hadoop. Ini semua sangat

+ Sialan mengerikan.

+

+ PS tidak boot antarmuka penghematan.

+ * /

Bottom line? Cassandra mungkin berguna untuk penyimpanan, tapi tidak ada pengolahan data. HBase jauh handier untuk itu.

Instalasi & Kemudahan Penggunaan

Cassandra hanya menginstal Ruby, Itu cukup mengesankan. Anda masih harus melakukan sedikit konfigurasi manual, namun. HBase adalah tar (atau dikemas oleh Cloudera) yang Anda butuhkan untuk menginstal dan setup pada Anda sendiri. HBase memiliki dokumentasi yang menyeluruh, meskipun, membuat proses sedikit lebih mudah daripada itu bisa saja.

HBase dengan shell Ruby sangat bagus yang membuat mudah untuk membuat dan memodifikasi database, mengatur dan mengambil data, dan sebagainya. menggunakan terus-menerus untuk menguji kode kita. Cassandra tidak memiliki shell di semua - hanya API dasar. HBase juga memiliki UI berbasis web yang bagus yang dapat Anda gunakan untuk melihat status cluster, menentukan node menyimpan berbagai data, dan melakukan beberapa operasi dasar lainnya. Cassandra tidak memiliki ini web UI serta shell, sehingga sulit untuk beroperasi. Cassandra menang secara keseluruhan pada instalasi, tetapi tertinggal pada kegunaan.

Arsitektur

Perbedaan mendasar dari ide-ide dan arsitektur belakang drive Cassandra dan HBase banyak kontroversi yang lebih baik.

klaim Cassandra bahwa "menulis *tidak pernah* gagal ", sedangkan di HBase, jika server sedang down wilayah, menulis akan diblokir untuk data yang terkena sampai data didistribusikan. Hal ini jarang terjadi dalam praktek, tentu saja, tapi akan terjadi

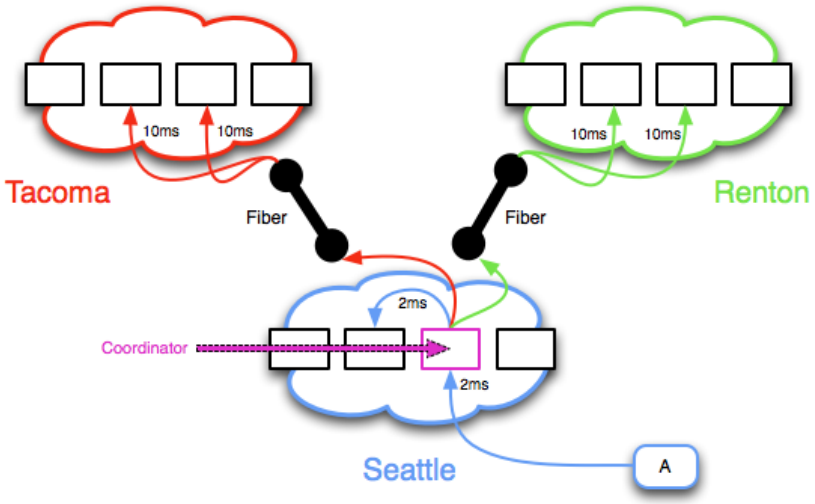
dalam cluster yang cukup besar. Selain itu, HBase memiliki satu titik-kegagalan (yang NameNode Hadoop), tapi itu akan kurang dari sebuah isu sebagai Hadoop berkembang. HBase tidak memiliki penguncian baris, Namun, yang Cassandra tidak.

Apps biasanya bergantung pada data yang akurat dan tidak berubah dari waktu akses, sehingga ide konsistensi akhirnya bisa menjadi masalah. bagaimanapun Cassandra, memiliki metode internal menyelesaikan up-to-dateness masalah dengan jam vektor - solusi kompleks namun bisa diterapkan di mana pada dasarnya menang timestamp terbaru. Para HBase / Bigtable menempatkan dorongan untuk menyelesaikan konflik konsistensi apapun pada aplikasi, karena semuanya disimpan berversi oleh timestamp.

Replikasi

Cassandra dioptimalkan untuk pusat data kecil (ratusan node) yang terhubung oleh serat sangat cepat. Itu bagian dari warisan Dynamo dari Amazon. HBase, yang berbasis pada penelitian awalnya diterbitkan oleh Google, senang untuk menangani replikasi untuk ribuan planet-node bertebaran di 'lambat', Internet terduga.

Perbedaan utama antara dua proyek adalah pendekatan mereka untuk replikasi dan beberapa pusat data. Cassandra menggunakan model sharing P2P, sedangkan HBase (versi mendatang) mempekerjakan lebih dari sebuah metode + data log cadangan, 'log pengiriman' alias. Masing-masing memiliki keanggunan tertentu. Seperti pada gambar dibawah ini

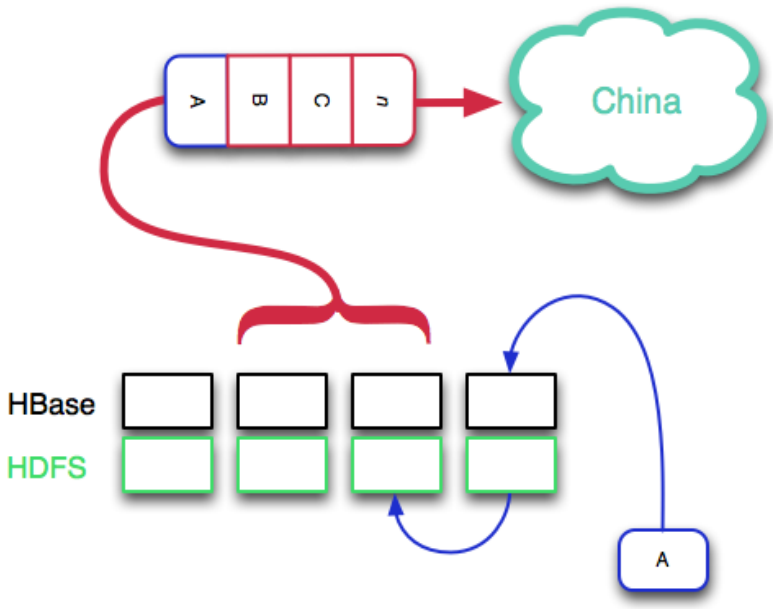


Gambar 64. Replikasi Cassandra
http://wiki.apache.org/cassandra

Ini diagram yang pertama adalah model dari skema replikasi Cassandra. Nilai ditulis untuk node "Koordinator" Nilai duplikat ditulis ke node lain dalam cluster yang sama Nilai ketiga dan keempat tertulis dari Koordinator untuk cluster lain di serat berkecepatan tinggi Nilai kelima dan keenam merupakan tertulis dari Koordinator untuk cluster ketiga di serat Setiap konflik diselesaikan di cluster dengan memeriksa cap dan menentukan yang nilai "terbaik".

Masalah utama dengan skema ini adalah bahwa tidak ada auditability dunia nyata. Node akhirnya konsisten - jika datacenter ("DC") gagal, tidak mungkin untuk mengatakan kapan jumlah yang diperlukan replika akan up-to-date. Hal ini dapat sangat menyakitkan dalam situasi hidup - ketika salah satu DC Anda turun, Anda sering ingin tahu * persis * kapan harus mengharapkan konsistensi data sehingga operasi pemulihan dapat melanjutkan lancar.

Sangat penting untuk dicatat bahwa Cassandra bergantung pada kecepatan tinggi serat antara pusat data. Jika menulis Anda yang mengambil 1 atau 2 ms, itu bagus. Tetapi ketika DC keluar dan Anda harus kembali ke satu sekunder di Cina bukan 20 mil jauhnya, latency yang luar biasa akan menyebabkan menulis timeout dan data yang sangat tidak konsisten. Mari kita melihat model replikasi HBase dibawah ini



Gambar 65. Replikasi Hbase
http://wiki.apache.org/cassandra

Apa yang terjadi di sini: Data ditulis ke HBase menulis-depan-log dalam RAM, maka kemudian memerah ke disk File pada disk secara otomatis direplikasi karena sifat Filesystem Hadoop itu Data memasuki "Log Replikasi", di mana ia disalurkan ke Data Center.

Dengan HBase/Hadoop's urutan peristiwa yang disengaja, konsistensi dalam datacenter tinggi. Biasanya ada hanya satu bagian dari data di sekitar periode waktu yang sama. Jika tidak ada, maka HBase yang memungkinkan kode Anda untuk mengetahui versi mana yang "benar" salah, bukan itu yang dipilih oleh cluster. Karena sifat Log Replikasi, orang selalu dapat mengetahui keadaan konsistensi data setiap saat - alat yang berharga untuk memiliki ketika pusat data lain turun. Selain itu, menggunakan struktur ini membuat mudah untuk pulih dari latency tinggi skenario yang dapat terjadi dengan antar-benua transfer data.

Mengetahui mana Untuk Pilih

Konteks bisnis dari Amazon dan Google menjelaskan penekanan pada fungsi yang berbeda antara Cassandra dan HBase. Cassandra mengharapkan Tautan Jaringan High Speed antara pusat data. Ini adalah artefak dari Dynamo Amazon: Amazon pusat data secara historis terletak sangat dekat satu sama lain (puluhan mil terpisah) dengan sangat cepat kabel serat optik antara mereka. Google, bagaimanapun, telah pusat data benua yang terhubung hanya dengan Internet standar, yang berarti mereka membutuhkan mekanisme replikasi lebih dapat diandalkan daripada konsistensi akhirnya P2P.

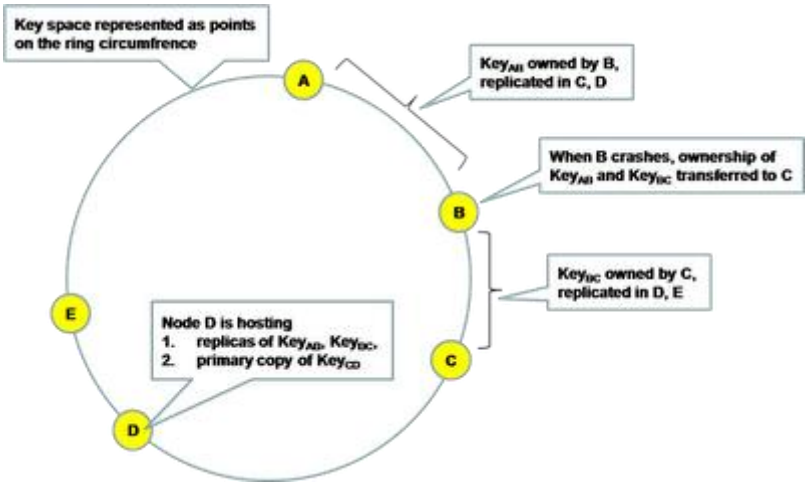
Jika Anda perlu menulis sangat tersedia dengan hanya konsistensi akhirnya, maka Cassandra adalah calon yang layak untuk saat ini. Namun, banyak aplikasi yang tidak senang dengan konsistensi akhirnya, dan itu masih kurang banyak fitur. Selanjutnya, bahkan jika menulis tidak gagal, masih ada downtime yang terkait dengan klaster bahkan perubahan skema kecil. HBase lebih difokuskan pada membaca, namun dapat menangani yang sangat tinggi membaca dan menulis throughput. Ini jauh lebih untuk Gudang Data, di samping untuk melayani jutaan request per detik. Integrasi HBase dengan MapReduce membuatnya berharga, dan serbaguna.

Ini adalah area di mana NoSQL bersinar. Dalam 18 bulan terakhir, NoSQL telah menjadi salah satu topik terpanas di industri perangkat lunak. Telah diperkenalkan sebagai solusi untuk masalah skala besar penyimpanan data pada kisaran terabyte atau petabyte. Puluhan produk NoSQL telah datang ke pasar, tetapi dua pemimpin **HBase dan Cassandra** tampaknya menonjol dari yang lain dalam hal adopsi mereka. Mengingat meningkatnya permintaan untuk menjelaskan ini 2 produk baru ini, Baik HBase dan Cassandra didasarkan pada model Google Bigtable, di sini memungkinkan memperkenalkan beberapa karakteristik kunci yang mendasari Bigtable pertama. **Pada dasarnya Terdistribusi** Bigtable dibangun dari bawah ke atas pada "sangat didistribusikan", arsitektur "berbagi apa-apa". Data seharusnya untuk menyimpan dalam jumlah besar tidak dapat diandalkan, kotak komoditas server dengan "partisi" dan "replikasi". Data partisi berarti data dipartisi oleh kunci dan disimpan dalam server yang berbeda. Replikasi berarti elemen data yang sama beberapa kali direplikasi pada server yang berbeda.

Cassandra Juga didasarkan pada model Bigtable, Cassandra DHT (didistribusikantabel has) model untuk partisi data, berdasarkan kertas yang dijelaskan dalam model Dynamo Amazon.

Konsisten Hashing melalui O (1) DHT
Setiap mesin (node) dikaitkan dengan id tertentu yang didistribusikan di keyspace (misalnya 128 bit). Semua elemen data juga berhubungan dengan kunci (dalam ruang kunci yang sama). Server memiliki semua data yang utama terletak antara id dan idserver sebelumnya yang.

Data juga direplikasi di beberapa server. Cassandra menawarkan skema replikasisimultiple termasuk menyimpan replika deserver (yang sukses id server yang memiliki data) atau strategi rak-sadar dengan menyimpan replika dilokasi fisik. Seperti pada gambar dibawah ini.



Gambar 66. Skema replikasisimultiple
http://wiki.apache.org/cassandra

Konsistensi Tidak seperti HBase, Cassandra memungkinkan Anda untuk memilih tingkat konsistensi yang cocok untuk aplikasi Anda, sehingga Anda bisa mendapatkan skalabilitas lebih jika bersedia untuk tradeoff beberapa konsistensi data. Misalnya, memungkinkan Anda untuk memilih berapa banyak ACK untuk menerima dari replika yang berbeda sebelum mempertimbangkan MENULIS untuk menjadi sukses. Demikian pula, Anda dapat memilih berapa banyak respon replika untuk diterima dalam kasus BACA sebelum kembali hasilnya ke klien.

Dengan memilih nomor yang sesuai untuk respon W dan R, Anda dapat memilih tingkat konsistensi yang Anda suka. Misalnya, untuk mencapai Konsistensi ketat, kita hanya perlu memilih W, R sehingga W + R> N. Ini termasuk kemungkinan (W = satu dan R = semua), (R = satu dan W = semua), (W = dan R = kuorum kuorum). Tentu saja, jika Anda tidak membutuhkan konsistensi yang ketat, Anda bahkan dapat memilih nilai yang lebih kecil untuk W dan R dan mendapatkan ketersediaan yang lebih besar. Terlepas dari apa tingkat konsistensi yang Anda pilih, data akan akhirnya konsisten oleh "mengisyaratkan handoff", "membaca perbaikan" dan "anti-entropi sync" mekanisme diuraikan di bawah ini.

Mengisyaratkan Handoff

Klien melakukan menulis dengan mengirimkan permintaan ke setiap node Cassandra yang akan bertindak sebagai proxy untuk klien. Node ini proxy akan berlokasi N node

sesuai yang memegang replika data dan meneruskan permintaan menulis kepada mereka semua. Dalam kasus setiap node gagal, ia akan memilih simpul acak sebagai node handoff dan menulis permintaan dengan sedikit menceritakannya untuk meneruskan permintaan menulis kembali ke node gagal setelah sembuh. Node handoff kemudian akan secara berkala memeriksa pemulihan node gagal dan meneruskan menulis untuk itu. Oleh karena itu, node asli akhirnya akan menerima semua permintaan menulis.

Resolusi Konflik

Karena menulis bisa mencapai replika yang berbeda, timestamp sesuai data digunakan untuk menyelesaikan konflik, dengan kata lain, menang timestamp terbaru dan mendorong cap waktu sebelumnya ke versi sebelumnya (mereka tidak hilang)

Baca Perbaikan

Ketika klien melakukan "membaca", node proxy akan masalah U membaca tapi hanya menunggu salinan R tanggapan dan kembali satu dengan versi terbaru. Dalam beberapa kasus node merespon dengan versi yang lebih tua, node proxy akan mengirimkan versi terbaru mereka asynchronous, maka simpul ini masih tertinggal akhirnya akan mengejar ketinggalan dengan versi terbaru.

Anti-Entropi sinkronisasi data

Untuk memastikan data tersebut masih sinkron bahkan tidak ada MEMBACA dan MENULIS terjadi untuk data, node replika berkala gosip satu sama lain untuk mencari tahu apakah ada yang tidak sinkron. Untuk setiap rentang kunci dari data, setiap anggota dalam kelompok replika menghitung pohon Merkel (pohon pengkodean hash di mana perbedaan dapat ditemukan dengan cepat) dan mengirimkannya ke tetangga lainnya. Dengan membandingkan pohon Merkel diterima dengan pohon sendiri, setiap anggota dapat dengan cepat menentukan bagian data yang tidak sinkron. Jika demikian, akan mengirim diff untuk kiri-belakang anggota.

Anti-entropi adalah "menangkap semua" cara untuk menjamin konsistensi akhirnya, tetapi juga cukup mahal dan karena itu tidak dilakukan sering. Dengan menggabungkan sinkronisasi data dengan perbaikan membaca dan mengisyaratkan handoff, kita dapat menjaga replika cukup up-to-date.

Google Storage

Google berencana untuk head to head dengan layanan cloud populer penyimpanan S3 Amazon dengan yang baru Penyimpanan Google untuk Pengembang. Seperti S3, layanan baru Google menawarkan pengembang sebuah cara, murah scalable untuk menyimpan data secara online. Meskipun tidak persis dongeng " GDrive, "Penyimpanan Google untuk Pengembang pasti meletakkan dasar bagi Google untuk membuat layanan penyimpanan user-friendly online. Penyimpanan untuk Pengembang Google menawarkan API, backup di pusat data dan bahkan memiliki dukungan untuk menyimpan file besar hingga ratusan gigabyte dalam ukuran.

Penyimpanan Google untuk Pengembang saat ini merupakan eksperimental Google Labs proyek. Untuk saat ini layanan ini tersedia dengan undangan saja dan terbatas pada pengembang AS. Anda dapat ke halaman sign up untuk meminta undangan yang akan memberikan Anda akses ke 100GB penyimpanan data dan 300GB per bulan data transfer bandwidth. Setelah aplikasi Anda hits batas-batas pay-as-you-go skema masuk harga tersebut kira-kira analog dengan layanan S3 Amazon. Versi Google akan berjalan Anda 17 sen per GB per bulan untuk penyimpanan sederhana, 10 sen per GB untuk meng-upload data dan 15 sampai 30 sen per GB untuk download. Ada juga biaya untuk jumlah permintaan - $ 0,01 per 1000 permintaan PUT, POST atau DAFTAR dan $ 0,01 per 10.000 permintaan menggunakan GET atau HEAD.

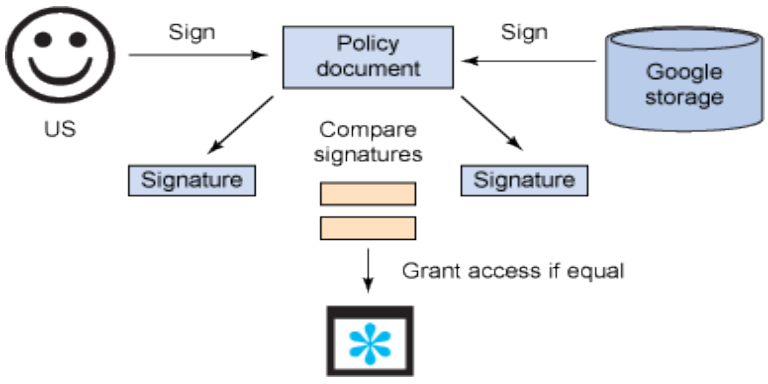
Sayangnya itu hanya cukup berbeda dari struktur harga Amazon (yang menurunkan harga GB per sebagai penggunaan Anda naik) sehingga sulit untuk mengatakan yang lebih murah. Sepintas Amazon S3 layanan terlihat sedikit lebih murah untuk penyimpanan, namun pada akhirnya biaya total - dan yang lebih murah - akan bervariasi tergantung pada sifat dari aplikasi web Anda dan bagaimana Anda menggunakan salah satu layanan penyimpanan. Google memang menawarkan beberapa alat tambahan bahkan Amazon tidak memiliki - API BigQuery dan API Prediksi. Menurut para penggunan Google Code , BigQuery dirancang untuk mengeksplorasi sejarah data Anda, dan lebih menarik API Prediksi memberikan Anda akses ke mesin belajar algoritma Google yang dirancang untuk "membuat aplikasi Anda lebih cerdas." API Prediksi dapat membantu membuat keputusan dalam waktu

sebenarnya "seperti merekomendasikan produk, menilai sentimen pengguna dari blog dan tweets, routing pesan atau menilai kegiatan yang mencurigakan," kata blog Google Code. Untuk saat ini tidak ada biaya untuk menggunakan API tambahan, meskipun mencatat bahwa dalam pengumuman tampaknya menunjukkan bahwa, ketika Penyimpanan Pengembang Google untuk bergerak keluar dari Labs, akan ada biaya tambahan. Karena Penyimpanan Google untuk Pengembang adalah beta Labs proyek, Anda tidak akan ingin beralih dari layanan Amazon saja, namun jika Anda ingin mengambil Penyimpanan Google untuk Pengembang untuk spin.

Penyimpanan Google untuk Pengembang adalah layanan baru bagi pengembang untuk menyimpan dan mengakses data di cloud Google. Menawarkan pengembang akses langsung ke penyimpanan scalable Google dan infrastruktur jaringan serta otentikasi kuat dan mekanisme berbagi data. Pengembang dapat menyimpan benda-benda dari berbagai ukuran dan mengelola akses ke data mereka pada seorang individu atau secara kelompok menggunakan Penyimpanan Google.
Google Penyimpanan dapat diakses melalui API REST atau menggunakan salah satu alat Penyimpanan Google yang tersedia, termasuk:

- Google Storage Manager - Sebuah antarmuka pengguna grafis dengan drag-and-drop fitur untuk mengelola ember Anda dan objek.
- GSUtil - Sebuah command line tool yang memungkinkan Anda melakukan tugas-tugas dasar dan lanjutan untuk Penyimpanan Google.

Anda dapat mengaktifkan Penyimpanan Google untuk Pengembang melalui API Google Konsol . Jika Anda belum pernah menggunakan Google API konsol, lihat Google API panduan bantuan Console .



Gambar 67. Bagaimana upload dikonfirmasi ke Google Penyimpanan
http://id.wikipedia.org/wiki/ google.com/storage

Google API Konsol

Google API Konsol adalah antarmuka grafis di mana Anda dapat mengaktifkan dan mengatur banyak Google API dan pengaturan yang terkait, seperti penagihan, aktivasi dan deaktivasi, otorisasi, dan API kustom fitur.

Google API Konsol terdiri dari proyek-proyek dan setiap diaktifkan Google API atau layanan milik sebuah proyek. Proyek dapat berisi satu atau lebih API diaktifkan. Bila Anda mengaktifkan Google API, memungkinkan penagihan, dan melakukan tugas-tugas administrasi lainnya, Anda melakukannya berdasarkan proyek. Sebagai contoh, Anda mungkin memiliki proyek yang disebut api_project yang berisi beberapa API diaktifkan. Ketika Anda mengaktifkan penagihan untuk api_project, Anda mengaktifkan untuk semua API dalam proyek itu. Demikian juga, jika Anda menambahkan anggota tim untuk api_project, mereka memiliki akses ke semua API dalam proyek itu, meskipun tingkat akses dapat didefinisikan oleh masing-masing secara terpisah API.
Untuk informasi lebih lanjut, kunjungi Google API Bantuan Konsol panduan.

Fitur & Kemanpuan

Penyimpanan google menyediakan beberapa fitur dan kemampuan yang membuat menyimpan, berbagi, dan mengelola data yang efisien dan dapat diandalkan. Beberapa fitur kunci dan kemampuan meliputi:

- Kapasitas dan skalabilitas tinggi
- Konsistensi data yang kuat
- OAuth 2,0 otentikasi
- Berbasis cookie download browser yang otentik
- Google API Konsol Proyek
- Account Google dukungan untuk berbagi
- REST API
- Bucket lokasi

Fitur-fitur dan kemampuan yang dibahas dalam bagian berikut.

Kapasitas dan Skalabilitas Tinggi

Google Penyimpanan mendukung objek yang dapat terabyte dalam ukuran. Penyimpanan Google juga mendukung sejumlah besar ember per account.

Konsistensi data yang kuat

Google Penyimpanan menyediakan kuat baca-setelah-menulis konsistensi untuk semua upload dan menghapus operasi. Dengan kata lain, setelah Anda meng-upload sebuah objek berhasil Anda segera dapat men-download, menghapus, atau mendapatkan metadata nya. Demikian juga, setiap usaha untuk mengakses objek segera setelah Anda berhasil menghapus itu menghasilkan 404 Tidak Ditemukan kode status. Operasi list pada akhirnya konsisten dari mana saja di Internet.

Dari sudut pandang ketersediaan, meng-upload operasi ke Google Penyimpanan adalah atom. Ketika Anda meng-upload sebuah objek, objek tidak tersedia sampai benar-benar upload. Objek upload tidak pernah tersedia dalam keadaan rusak atau sebagai obyek parsial. Objek baik tersedia atau mereka tidak.

Untuk informasi lebih lanjut, lihat Konsistensi dalam gambaran API.

OAuth 2.0 Otentikasi

Google Penyimpanan menggunakan otentikasi OAuth 2,0 dan otorisasi untuk berinteraksi dengan API. OAuth 2,0 otentikasi adalah otentikasi berbasis-token mana Anda dapat mengeluarkan token ke aplikasi untuk bertindak atas nama Anda. Anda dapat mengatur otentikasi OAuth 2.0 untuk aplikasi Anda dengan mengunjungi otentikasi dan otorisasi OAuth 2,0 membimbing dan membaca tentang otentikasi 2,0 OAuth khusus untuk Penyimpanan Google .

Browser dikonfirmasi Downloads

Google Penyimpanan memungkinkan Anda memberikan berbasis browser download otentik kepada pemegang akun Google individu. Anda dapat memberikan browser berbasis dikonfirmasi download dengan terlebih dahulu menerapkan ACL berbasis akun Google untuk obyek dan kemudian mengirimkan pengguna URL yang scoped ke objek. URL untuk download browser yang dikonfirmasi adalah:

https://sandbox.google.com/storage/ *ember* / *objek*

Untuk informasi lebih lanjut, lihat cookie Otentikasi berbasis di Otentikasi.

Google API Konsol Proyek

Google Penyimpanan tersedia sebagai layanan di Google API Konsol di mana Anda dapat menambahkan anggota proyek, menangani tagihan, mengelola otentikasi, dan bekerja dengan API lain. Anda dapat memiliki banyak proyek dan setiap proyek dapat memiliki contoh sendiri Penyimpanan Google.

Account Google Dukungan untuk Berbagi

Google Penyimpanan menggunakan ACL untuk mengontrol akses ke objek dan ember. Dengan mengkonfigurasi ACL Anda dapat berbagi objek dan ember dengan seluruh dunia, kelompok Google, domain Google-host, atau Google tertentu pemegang accounts. Untuk informasi lebih lanjut tentang model Penyimpanan akses kontrol Google, lihat Access Control .

SOAP API

Google Penyimpanan menyediakan antarmuka tenang pemrograman sederhana, sehingga Anda tidak harus bergantung pada SOAP toolkit rumit atau RPC pemrograman untuk membuat aplikasi yang menyimpan, berbagi, dan mengelola data pada Penyimpanan Google. Sebaliknya, Anda dapat menggunakan metode HTTP standar, seperti PUT, GET, POST, HEAD, dan DELETE.

Bucket Lokasi

Google Penyimpanan menyediakan kemampuan untuk menentukan di mana Anda secara geografis ember disimpan. Hal ini dimungkinkan untuk menentukan bahwa ember disimpan di Eropa atau di AS. Lihat Menentukan Lokasi Bucket untuk informasi lebih lanjut.

Kelebihan Menyimpan data Anda di Cloud Google

Penyimpanan Google untuk Pengembang adalah layanan tenang untuk menyimpan dan mengakses data Anda pada infrastruktur Google. Layanan ini menggabungkan kinerja dan skalabilitas cloud Google dengan keamanan yang canggih dan kemampuan berbagi. Meliputi:

Cepat, terukur, sangat tersedia menyimpan objek

- Semua data direplikasi ke beberapa pusat data
- Baca--Anda menulis konsistensi data

- Objek dapat terabyte dalam ukuran, dengan upload dan download resumable, dan berbagai-GET dukungan
- Domain-scoped ember namespace

Mudah, fleksibel, otentikasi dan berbagi

- OAuth 2.0 Otentikasi
- Dikonfirmasi download dari web browser
- Individu-, proyek, dan kelompok-tingkat akses kontrol

Selain itu, Penyimpanan Google untuk Pengembang menawarkan antarmuka berbasis web untuk mengelola penyimpanan Anda, perintah open source tool baris dan perpustakaan , dan fitur canggih seperti upload resumable dan fungsionalitas penyimpanan internasional .Dengan pay-as-you-go harga, mudah untuk memulai dan skala sebagai kebutuhan Anda tumbuh. Anda dapat mengaktifkan Penyimpanan Google pada Google API Console . Untuk mempelajari lebih lanjut, baca Pendahuluan .

Cloud Storage

Penyimpanan cloud adalah sebuah model jaringan penyimpanan data komputer di mana data disimpan pada server virtual, pada umumnya diselenggarakan oleh pihak ketiga, bukannya host di dedicated server. Hosting perusahaan mengoperasikan pusat data yang besar, dan orang yang membutuhkan data mereka untuk menjadi host membeli atau menyewa kapasitas penyimpanan dari mereka dan menggunakannya untuk kebutuhan penyimpanan mereka. Para pusat data operator, di latar belakang, virtualisasi sumber daya sesuai dengan persyaratan pelanggan dan mengekspos mereka sebagai server virtual, dimana pelanggan dapat sendiri mengelola. Dalam arti fisik, sumber daya mungkin span di beberapa server.

Jasa penyimpanan cloud dapat diakses melalui layanan web antarmuka pemrograman aplikasi (API), atau melalui web berbasis antarmuka pengguna.

Keuntungan Cloud Storage

- Perusahaan hanya perlu membayar untuk penyimpanan mereka benar-benar digunakan.
- Perusahaan tidak perlu menginstal perangkat penyimpanan fisik di datacenter mereka sendiri atau kantor, yang mengurangi biaya TI dan hosting.
- Tugas pemeliharaan penyimpanan, seperti backup, replikasi data, dan pembelian perangkat penyimpanan tambahan diturunkan untuk tanggung jawab penyedia layanan, memungkinkan organisasi untuk fokus pada bisnis inti mereka.

Keamanan Data

- Keamanan data yang disimpan dan data dalam transit dapat menjadi perhatian ketika menyimpan data sensitif di penyedia penyimpanan cloud.
- Kinerja dapat lebih rendah dari penyimpanan lokal
- Keandalan dan ketersediaan tergantung pada ketersediaan jaringan area luas dan pada tingkat tindakan pencegahan yang diambil oleh penyedia layanan.

Pengguna dengan spesifik catatan-persyaratan penyimpanan, seperti badan-badan publik yang harus mempertahankan catatan elektronik sesuai dengan undang-undang,

mungkin menghadapi komplikasi dengan menggunakan komputasi cloud dan penyimpanan.



Gambar. 68 Ilustrasi Cloud Storage
Sumber : http://id.wikipedia.org/wiki/cloudstorage

Gateway Cloud Storage

Sebuah gateway penyimpanan cloud dapat opsional digunakan di tempat pelanggan, yang mengekspos layanan penyimpanan cloud seolah-olah mereka mempuyai perangkat penyimpanan lokal. Gateway penyimpanan Cloud adalah jaringan peralatan atau server yang menerjemahkan API penyimpanan cloud standar seperti SOAP baik blok berbasis protokol penyimpanan data seperti iSCSI atau Fibre Channel, atau protokol file penyimpanan berbasis jaringan seperti NFS atau CIFS (SMB). cloud gateway penyimpanan adalah target yang sangat baik untuk melakukan backup. Kinerja LAN berkecepatan membantu backup untuk menyelesaikan dalam jendela dialokasikan waktu mereka, sementara cadangan kemudian direplikasi asynchronously untuk penyimpanan offsite, setelah data deduplication dan kompresi dilakukan.

Ada tiga kunci Gateway Cloud Storage

Sebagai alternatif data yang muncul perlindungan, penyimpanan cloud dengan cepat menjadi pilihan populer untuk backup dan pengarsipan. Meskipun berbasis disk

cadangan menyediakan pemulihan lebih cepat backup dan data, datang dengan biaya tinggi dalam hal harga dan infrastruktur dan manajemen penyimpanan. Selain itu, tape backup dan pengarsipan memakan waktu, rentan terhadap kerugian atau korupsi data dan kompleks untuk mengelola. Karena kekurangan ini, disk backup dan solusi pengarsipan biasanya tidak dianggap sebagai strategi jangka panjang layak perlindungan data, membuat penyimpanan cloud pilihan yang menarik untuk cadangan dan pengarsipan.

Cloud penyimpanan memberikan keuntungan seperti kapasitas yang fleksibel dan kelincahan bisnis yang meningkat. Dan satu manfaat utama mengemudi pertimbangan penyimpanan cloud adalah penghematan biaya dramatis dapat menghasilkan. Sedangkan disk dan tape sistem memerlukan biaya modal yang tinggi, peramalan menantang dan manajemen yang kompleks, penyedia penyimpanan cloud seperti **Amazon S3, AT & T** Penyimpanan Synaptic sebagai Layanan dan Nirvanix biaya SDN hanya pelanggan penyimpanan penggunaan, sementara perekonomian memanfaatkan skala mencapai titik harga menarik - sekitar 15 - 25 sen per gigabyte per bulan. Beralih ke penyimpanan cloud publik juga dapat menghasilkan penghematan biaya dalam manajemen sistem, daya, pendinginan dan biaya ruang karena pengurangan dalam infrastruktur pelanggan yang dimiliki dan dikelola.
Jelas, penyimpanan cloud adalah sangat biaya-efektif perlindungan data yang menengah. Namun, masih ada kekhawatiran yang valid tentang merangkul cloud. Sebelum pindah ke cloud, organisasi harus memperhitungkan resiko seperti data, ketersediaan kinerja dan keamanan.

Kinerja merupakan kekhawatiran signifikan mengenai adopsi cloud penyimpanan. Operasi perlindungan data dapat memperlambat respon untuk aplikasi kritis lain yang berjalan melalui jaringan wide area (WAN). Manajer TI dapat mengurangi ini dengan penjadwalan backup di off-peak, tapi ini tidak selalu memungkinkan - apakah karena replikasi data kontinu atau kebutuhan untuk keadaan darurat mengembalikan. Proses yang berbeda secara bersamaan bersaing untuk bandwidth WAN yang terbatas dapat memperlambat jaringan untuk semua fungsi TI, termasuk kemampuan untuk cepat replikasi data.

keamanan dianggap sangat penting terutama menyangkut penyimpanan cloud. Manajer TI perlu untuk mengidentifikasi pilihan yang optimal untuk masalah seperti enkripsi data, segregasi data dan akses pengguna istimewa. Selain itu, manajer TI harus menuntut cloud penyedia layanan membuat data tersedia pada permintaan, garis elemen yang dibutuhkan untuk layanan tingkat kesepakatan (SLA) jaminan, serta mempertimbangkan bagaimana untuk mengintegrasikan ke dalam cloud pemulihan bencana mereka (DR) strategi.

Untuk mengatasi tantangan ini, organisasi yang beralih ke cloud penyimpanan, ini merupakan generasi baru dari solusi yang dibangun untuk mengatasi tantangan yang unik dari lingkungan penyimpanan cloud. Cloud penyimpanan terbaik gateway mengoptimalkan perlindungan data oleh deduplicating, kompresi dan enkripsi backup data dan arsip set sebelum menyimpannya dalam cache lokal dan kemudian aman mempercepat mereka melalui jaringan dan masuk ke cloud. Hal ini tidak hanya membebaskan staf TI dari padat karya tape backup dan manajemen infrastruktur, tetapi dalam bencana, data dapat dikembalikan ke lokasi manapun di Internet.

Gateway penyimpanan cloud yang memungkinkan cloud publik untuk cepat menjadi penyimpanan yang layak tingkat dengan mengatasi masalah seperti kinerja dan keamanan yang terkait dengan penyimpanan di cloud. Tapi tidak semua solusi penyimpanan cloud gerbang yang sama. Sebelum bergerak ke depan, organisasi harus mempertimbangkan tiga pertimbangan utama ketika mengevaluasi solusi penyimpanan cloud gateway:

- **Aktifkan Optimasi Kinerja** : Jelaslah bahwa tanpa kinerja jaringan yang memadai secara keseluruhan, manfaat dari penyimpanan cloud dikompromikan. Bahkan dengan ROI yang menarik terkait dengan penyimpanan cloud, organisasi mungkin akan menunda adopsi jika ada kekhawatiran bahwa mungkin mengurangi produktivitas karena waktu respon dapat diterima saat mengakses file, aplikasi atau virtual desktop. Ketika mengevaluasi solusi penyimpanan cloud gateway, sangat penting bahwa solusi termasuk jaringan inline dan deduplication penyimpanan, yang keduanya diperlukan untuk meminimalkan biaya dan memaksimalkan kinerja untuk backup lebih cepat dan pemulihan. Dengan mengurangi jumlah penyimpanan

yang diperlukan untuk menyimpan cadangan beban kerja dalam cloud publik, Anda dapat meningkatkan kemampuan DR keseluruhan sementara juga secara signifikan mengurangi dampak pada jaringan dan biaya penyimpanan cloud.

- **Mengurangi Risiko dengan End-to-End Keamanan** : Membuat enkripsi - untuk kedua transmisi dan penyimpanan - persyaratan. Dengan memiliki tingkat enkripsi, Anda membantu memastikan bahwa setiap data yang pindah ke cloud penyimpanan tidak terganggu - menciptakan end-to-end security.

- **Hilangkan vendor Lock-in** : Periksa bahwa solusi dapat mendukung berbagai penyimpanan cloud dan solusi perangkat lunak cadangan sehingga Anda yakin dapat pindah ke penyedia penyimpanan cloud pilihan Anda tanpa mengubah perangkat lunak cadangan yang ada, jadwal pekerjaan atau proses. Memiliki dukungan untuk penyedia cloud penyimpanan di dalam daerah yang lebih memberikan fleksibilitas untuk memilih penyedia penyimpanan cloud yang paling memenuhi persyaratan khusus untuk DR, penyimpanan data lokasi, SLA dan biaya, sementara mengizinkan beralih ke penyedia baru sebagai IT membutuhkan perubahan.

Sebagai penyimpanan cloud publik terus memperoleh traksi dan matang, ia memiliki kemampuan untuk mengatasi kelemahan utama yang ditemukan dalam solusi proteksi data yang dibangun di sekitar tape dan pelanggan yang dimiliki disk. Perlindungan data dan proyek pengarsipan antara beban kerja yang paling populer saat ini untuk menggunakan cloud publik. Dengan penawaran gerbang cloud penyimpanan yang ada, yang signifikan dan segera penghematan biaya dapat dicapai. Ke depan, solusi penyimpanan gerbang cloud akan teknologi yang memungkinkan kunci yang akan memungkinkan organisasi untuk menjadi lebih lincah, mengurangi biaya dan meningkatkan kemampuan DR dengan memanfaatkan potensi penuh dan kekuatan penyimpanan cloud publik.

Beberapa layanan Cloud Storage gratis terbaik yang tersedia didunia maya.

Dropbox



Gambar. 69. Dropbox

Ini Merupakan Cloud storage yang menyediakan aplikasi bawaan managemen file dan folder, jadi untuk menggunakannya kita tidak harus mengaksesnya melalui web Interface. tersedia untuk windows dan linux. Untuk space dropbox menyediakan 2GB dan satu hal yang menarik dari dropbox adalah kita akan mendapatkan space tambahan jika berhasil mengajak orang lain untuk menggunakan dropbox berdasarkan rujukan kita (referral system). Beberapa fitur yang didapat dari dropbox shared folder memungkinkan beberapa orang untuk berkolaborasi pada file yang sama dan dapat secara otomatis syncs ketika file baru atau perubahan terdeteksi.

Box.net



Gambar 70. Box.net

Di Box.net ditawarkan tiga jenis Box yakni Basic, Individual, dan Business. Yang pertama Free, dan yang dua terakhir berbayar. Beberapa hal yang didapat saat kita terdaftar dengan fasilitas BASIC antara lain Space 1 GB dan Bandwith 10 GB dan juga selain bisa menyimpan file di box pribadi, file juga bisa di-share di ruang publik.Untuk

managemen file Box.net tidak menyediakan aplikasi bawaan seperti pada Dropbox jadi untuk menggunakannya kita harus berhadapan dengan web interface.

Memopal



Gambar 71. Memopal

Memopal merupakan layanan penyimpanan Online yang terkenal denga slogan More Friends = More Space artinya disini kita akan mendapatkan space tambahan setiap berhasil mengajak teman untuk menggunakan memopal yang merujuk pada nama (account) kita (referral system), memopal menawarkan space 3 GB dan menyediakan aplikasi bawaan managemen file dan folder (shell) , tersedia untuk Windows, Linux, Mac OS dan platform mobile.

Windows Skydrive

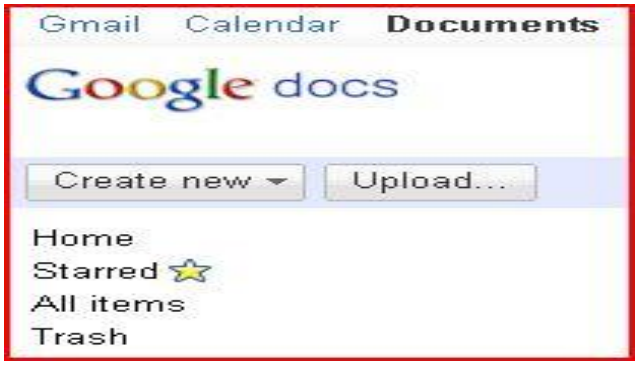


Gambar. 72 Windows Skydrive

SkyDrive merupakan layanan penyimpanan data Online yang diberikan oleh Microsoft dengan space 25 GB dan batasan ukuran 50 MB untuk setiap file, salah satu hal menarik yang diberikan oleh Skydrive adalah adanya fitur terintegrasi dengan layanan web office yang membuat kita dapat mengedit file microsoft office yang kita simpan

pada SkyDrive melalui web. walaupun tidak menyediakan aplikasi bawaan untuk manajeman file dan folder, kita tetap dapat menggunakan SkyDrive Exploreuntuk mengintegrasikan layanan ini dengan desktop.Sebuah layanan terpisah bernama SkyDrive synced storage juga akan kita dapatkan ketika mendaftar dengan Space 5 GB berbeda dengan SkyDrive yang memiliki batasan ukuran setiap file 50 MB, SkyDrive synced storage tidak memiliki ukuran batasan file dan untuk menggunakan SkyDrive synced storage pada windows terlebih dahulu harus menginstall Windows Live Mesh yang merupakan bagian dari Microsoft Live Essential.

Google Docs



Gambar. 73 Google Docs

Banyak yang belum mengetahui bahwa Google Docs menambahkan sebuah fasilitas baru pada layananya yang dapat digunakan untuk meng-upload semua jenis file, namun disini kita hanya akan mendapatkan space 1 GB, layanan ini sifatnya terpisah dengan G-mail dan Picasa dan tidak ada kaitannya dengan Gmail Drive.

Adopsi Teknologi

Layanan publik cloud penyimpanan cepat menjadi pilihan yang lebih menarik bagi perusahaan, menurut sebuah penelitian 2010. Pada tahun 2010, penyedia layanan termasuk Egnyte , Nirvanix, Nasuni , CTERA Jaringan dan Petaera mulai menggunakan berorientasi objek generasi baru teknologi penyimpanan, yang memungkinkan menyimpan volume yang sangat besar data dengan biaya yang jauh lebih rendah daripada sistem penyimpanan perusahaan konvensional. Perusahaan-perusahaan ini menawarkan pelanggan sebuah perusahaan gerbang Cloud penyimpanan yang

menerjemahkan protokol file tradisional berbasis cache penyimpanan berorientasi objek. Dalam konteks ini, Uni Eropa yang didanai Proyek VISI cloud sedang mengembangkan infrastruktur berbasis cloud, dibangun pada standar terbuka dan teknologi baru, untuk menyediakan kerangka kerja, fleksibel dan scalable diandalkan untuk pengiriman dioptimalkan data-intensif layanan penyimpanan.

Menurut IDC penyimpanan analis Brad Nisbet, solusi yang berfokus pada memberikan "biaya-efektif dan user-friendly" managed services cenderung mendorong adopsi penyimpanan cloud dalam bisnis kecil ruang pasar.

Referensi dan Bacaan Lebih Lanjut

1. Demystifying the Cloud An introduction to Cloud Janakiram MSV Cloud Computing Strategist www.janakiramm.net| mail@janakiramm.net
2. Book of Cloud Computing Herwin Angeriana, Andri, Yohanes Valentino Susilo, Frencis Matheos
3. http://alexbarnett.net/blog/archive/2008/04/08/so-what-is-this-platform-as-a-service-thing.aspx.
4. http://linux-bulukumba.blogspot.com/2011/01/cloud-computing.html#ixzz1Q5dsrNLT
5. http://itmove.wordpress.com/tag/cloud-computing/
6. http://www.webopedia.com/TERM/D/data_center_tiers.html
7. Gartner Says Cloud Computing Will Be As Influential As E-business
8. Cloud computing is a trap, warns GNU founder Richard Stallman, The Guardian, per Bobbie Johnson, 29 septembre 2008
9. Cloud Computing Defined 17 July 2010. Retrieved 26 July 2010.
10. Buyya, Rajkumar; Chee Shin Yeo, Srikumar Venugopal (PDF). Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. Department of Computer Science and Software Engineering, University of Melbourne, Australia. pp. 9. Retrieved 2008-07-31.
11. http://www.detikinet.com/read/2010/02/24/084138/1305595/328/lebih-jauh-mengenal-komputasi-cloud
12. http://teknoinfo.web.id/teknologi-cloud-computing/
13. http://blogoscoped.com/google-chrome
14. http://www.redbooks.ibm.com/redpapers/pdfs/redp4614.pdf
15. http://teknoinfo.web.id/teknologi-cloud-computing/
16. http://udaramaya.com/berita/3890/0/Pendiri_GNU_Jauhi_Cloud_Computing
17. http://google.com
18. http://opensource.telkomspeedy.com/wiki Computing
19. http://en.wikipedia.org/wiki/Cloud_computing
20. Amazon Server 'WikiLeaks Go Offline
21. http://avocadojuice.wordpress.com/2010/11/10/cloud-computing/
22. http://ervamadiyudhistira.wordpress.com/2010/12/11/cloud-computing/
23. http://www.gudanglinux.info/info/business/50-market/700-menghitung-keuntunganekonomi-
24. http://www.microsoft.com/windowsazure/sqlazure/.
25. Aplikasi dan organisasi yang menggunakan Hadoop
26. Hadoop Kredit Page
27. Yahoo! Dunia Luncurkan Hadoop Terbesar Produksi Aplikasi
28. pencipta Hadoop pergi ke Cloudera

29. Ashlee Vance (2009/03/17). "Hadoop, Program Perangkat Lunak Bebas, Menemukan Menggunakan Beyond Cari" . *Times New York* . Diperoleh 2010/01/20 .
30. Hadoop berisi platform komputasi terdistribusi yang sebelumnya merupakan bagian dari Nutch.. Ini mencakup Hadoop Distributed Filesystem (HDFS) dan penerapan MapReduce" Tentang Hadoop
31. http://hadoop.apache.org/common/docs/r0.20.2/hdfs_user_guide.html # Rack + Kesadaran
32. The File System Distributed Hadoop: Arsitektur dan Desain
33. Namenode Meningkatkan kinerja startup . "Skenario default untuk 20 juta file dengan ukuran tumpukan Jawa maks diatur ke 14GB: 40 menit:. Berbagai pilihan seperti Jawa Tuning muda sebagai ukuran sampah paralel, koleksi, Jawa tumpukan ukuran awal 14 menit"
34. Konfigurasi Sumber Daya Penyimpanan Lokal
35. (http://msdn.microsoft.com/en-us/library/ee758708.aspx)
36. Memahami dan gumpalan gumpalan Blok Halaman (http://msdn.microsoft.com/en-us/library/ee691964.aspx)
37. Layanan gumpalan Konsep (http://msdn.microsoft.com/en-us/library/dd179376.aspx)
38. Windows Azure Penyimpanan ISTIRAHAT Layanan Referensi API (http://msdn.microsoft.com/en-us/library/dd179355.aspx)
39. Windows Azure Drives kertas putih (http://go.microsoft.com/?linkid=9710117)
40. Layanan Antrian Konsep (http://msdn.microsoft.com/en-us/library/dd179353.aspx)
41. Antrian API Service (http://msdn.microsoft.com/en-us/library/dd179363.aspx)
42. Memahami Penyimpanan Tabel Data Model (http://msdn.microsoft.com/en-us/library/dd179338.aspx)
43. Tabel Layanan Konsep (http://msdn.microsoft.com/en-us/library/dd179463.aspx)
44. Memperkenalkan database SQL Azure (http://msdn.microsoft.com/en-us/library/ee336230.aspx)
45. Konsep Database SQL Azure (http://msdn.microsoft.com/en-us/library/ee336256.aspx)
46. Transact-SQL Dukungan (SQL Azure Database) (http://msdn.microsoft.com/en-us/library/ee336250.aspx)

47. SQL Server Tools dan Dukungan Utilitas (SQL Azure Database) (http://msdn.microsoft.com/en-us/library/ee621784.aspx)
48. Pedoman Umum dan Batasan (SQL Azure Database) (http://msdn.microsoft.com/en-us/library/ee336245.aspx)
49. SQL Azure kelangsungan hidup Panduan(http://social.technet.microsoft.com/wiki/contents/articles/sql-azure-survival-guide.aspx)
50. Windows Azure Harga (http://www.microsoft.com/windowsazure/pricing/)
51. Ikhtisar Pilihan untuk Data Migrasi dan Skema ke SQL
52. Tabel dan query Entitas (http://msdn.microsoft.com/en-us/library/dd894031.aspx)
53. Tipe Data (SQL Azure Database) (http://msdn.microsoft.com/en-us/library/ee336233.aspx)
54. Penanganan Transaksi di SQL
55. Windows Azure Penyimpanan Klien SDK (http://msdn.microsoft.com/en-us/library/microsoft.windowsazure.storageclient.aspx)
56. http://wiki.apache.org/cassandra/FrontPage
57. http://www.cs.cornell.edu/projects/ladis2009/papers/lakshman-ladis2009.pdf
58. http://cassandra.apache.org/
59. http://www.datastax.com/dev/blog/slides-and-videos-cassandra-summit-2010
60. http://arin.me/blog/wtf-is-a-supercolumn-cassandra-data-model
61. http://www.rackspace.com/cloud/blog/2010/05/12/cassandra-by-example/
62. http://jonathanhui.com/cassandra
63. Sumber: msdn.microsoft.com