

**MODUL PERKULIAHAN**

# Cloud Computing

## Topologi Jaringan & Perangkat Lunak Cloud Computing

### **Abstract**

Memahami tentang Topologi Jaringan dan Perangkat Lunak yang mendukung layanan Cloud Computing

### **Kompetensi**

Mampu memahami Topologi dan Software Cloud

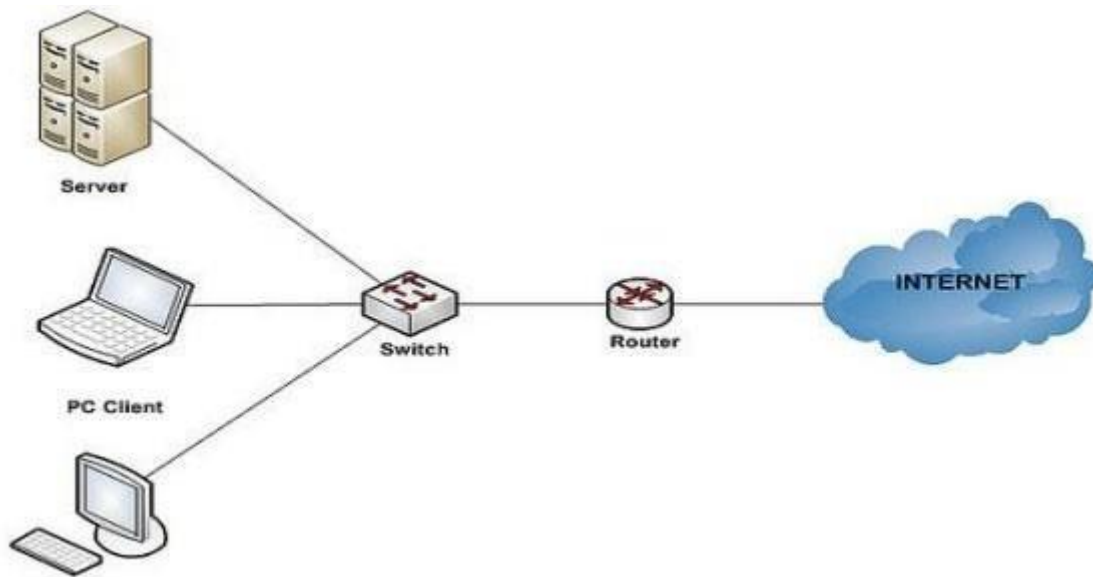
# Topologi Jaringan

Berbicara tentang sistem cloud computing, akan sangat membantu bila kita membaginya menjadi dua kelompok, yakni : front-end dan back-end. Keduanya terhubung melalui sebuah jaringan (Internet). Front-end terletak pada sisi pengguna atau client. Sementara back-end adalah bagian "awan" dalam sistem ini (dalam diagram jaringan internet kerap digambarkan sebagai awan).

Front-end mencakup komputer (atau jaringan komputer) client, dan aplikasi yang diperlukan untuk mengakses sistem cloud computing. Tidak semua sistem cloud computing memiliki interface yang sama. Untuk mengakses layanan Web 2.0 seperti email berbasis web hanya dibutuhkan web browser biasa, seperti Firefox, Internet Explorer, atau Opera. Namun, ada pula sistem cloud computing yang memiliki aplikasi sendiri (proprietary) yang harus diinstall di komputer client. Sementara itu, pada sisi back-end dari sistem cloud computing terdapat beragam komputer, server, dan sistem penyimpanan data, yang kesemuanya menciptakan "awan" bagi layanan komputasi.

Secara teori, sebuah sistem cloud computing mencakup semua program komputer yang dapat Anda bayangkan, dari data processing hingga video game. Biasanya, setiap aplikasi dijalankan dan memiliki server sendiri (dedicated server). Sebuah server pusat mengatur jalannya sistem, seperti memonitor lalu lintas, dan permintaan client untuk memastikan semuanya berjalan dengan baik.

Bila sebuah perusahaan cloud computing memiliki banyak client, maka kebutuhan akan ruang penyimpanan data (storage space) pun akan membengkak. Sistem cloud computing paling tidak membutuhkan ruang penyimpanan data dua kali lebih besar daripada kebutuhan riil untuk membuat salinan (copy) semua data client. Hal ini dimaksudkan untuk mencegah kehilangan data bila terjadi gangguan pada media penyimpanan utama.



Gambar 1. gambaran umum topologi Cloud Computing

Sumber : (herwin:2011)

## **Distribusi beban vertikal untuk Komputasi Awan melalui Pilihan Implementasi Multiple**

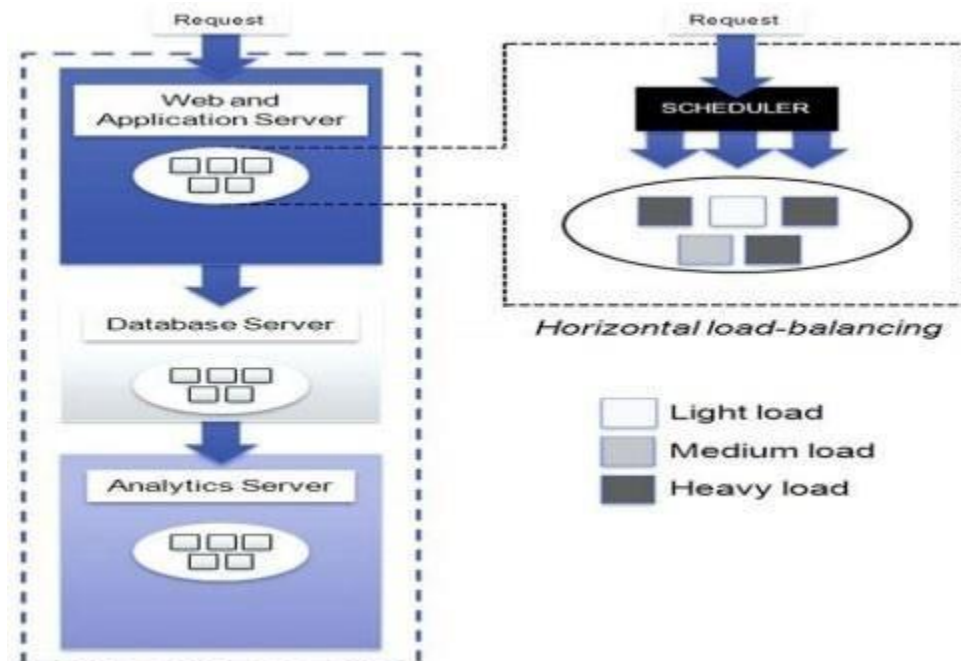
Cloud computing menyediakan perangkat lunak sebagai layanan cadangan untuk pengguna terakhir, tapi infrastruktur yang mendasari harus cukup terukur dan kuat dan harus fokus pada sistem Cloud perusahaan skala besar dan meneliti bagaimana perusahaan dapat menggunakan service - oriented architecture (SOA) untuk menyediakan antarmuka yang efisien untuk proses bisnis.

Untuk meningkatkan proses bisnis, masing-masing tingkatan SOA biasanya menyebarkan beberapa server untuk muatan distribusi dan toleransi kesalahan. Salah satu keterbatasan dari pendekatan ini adalah beban yang tidak dapat didistribusikan lebih lanjut saat semua server pada tingkatan/jajaran yang sama dimuat.

Cloud computing terlihat untuk perhitungan dan penyimpanan data menjauh dari end user dan ke server yang berlokasi di pusat data, dengan demikian mengurangi beban pengguna dari penyediaan aplikasi dan manajemen.

Dalam sistem awan enterprise , arsitektur berorientasi layanan (SOA) dapat digunakan untuk menyediakan antarmuka yang mendasari proses bisnis, yang ditawarkan melalui Awan (cloud). SOA dapat bertindak sebagai sebuah front-end terprogram ke berbagai komponen layanan yang dibedakan sebagai individu dan pendukung server. Permintaan yang masuk ke layanan yang di sediakan oleh gabungan SOA harus diteruskan ke komponen yang benar dan server masing - masing, dan seperti routing harus terukur untuk mendukung sejumlah besar permintaan.

Dalam rangka untuk meningkatkan proses bisnis, setiap tingkatan dalam sistem biasanya menyebarkan beberapa server untuk mendistribusikan beban dan toleransi kesalahan. seperti distribusi beban di beberapa server dalam tingkat yang sama dapat dilihat sebagai distribusi beban horisontal, tampak seperti gambar berikut :



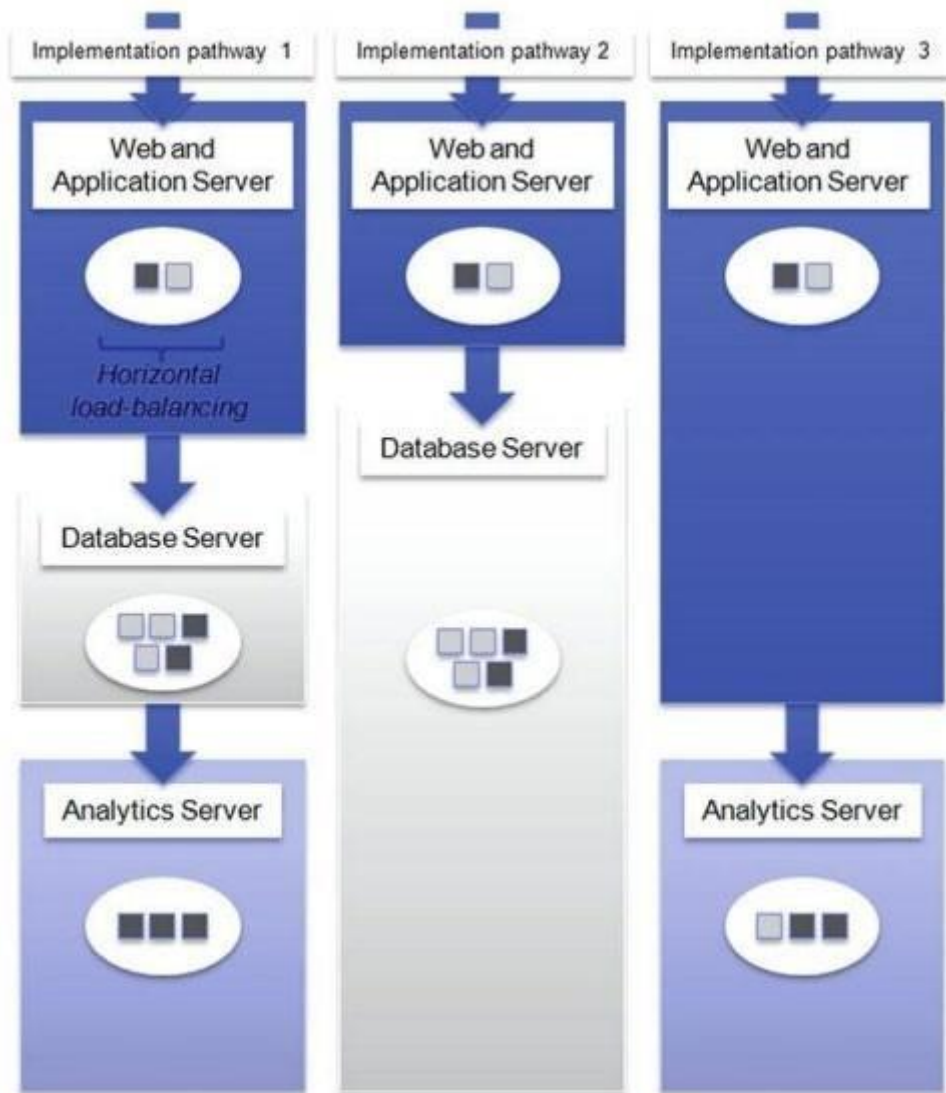
Gambar 2. Horizontal distribusi beban: beban didistribusikan di dalam server, dalam tingkat yang sama.

Sumber : (herwin:2011)

Salah satu batasan dari distribusi beban horisontal adalah bahwa beban tidak dapat didistribusikan lebih lanjut ketika semua server dalam tingkatan tertentu mengambil hasil dari kesalahan konfigurasi infrastruktur. Dimana terlalu banyak server yang dikerahkan pada satu tingkat sementara dilain pihak ada sedikit server yang dikerahkan di lain tingkatan.

Sebuah pengamatan penting adalah bahwa dalam sistem kompleks SOA multi-tier, proses bisnis tunggal sebenarnya bisa dilaksanakan oleh beberapa jalur yang berbeda melalui tingkat perhitungan dalam rangka memberikan ketahanan dan skalabilitas.

Sebuah layanan komposit dapat direpresentasikan sebagai tingkatan pemanggilan beberapa komponen dalam sebuah infrastruktur TI berbasis SOA. Dalam sistem seperti itu, kami membedakan distribusi beban horisontal, dimana beban dapat tersebar di beberapa server untuk satu komponen layanan, dari distribusi beban vertikal, dimana beban dapat tersebar di beberapa implementasi dari layanan yang diberikan. Gambar berikut menggambarkan istilah istilah di atas.



Gambar 3. Distribusi beban vertical.

Sumber : (herwin:2011)

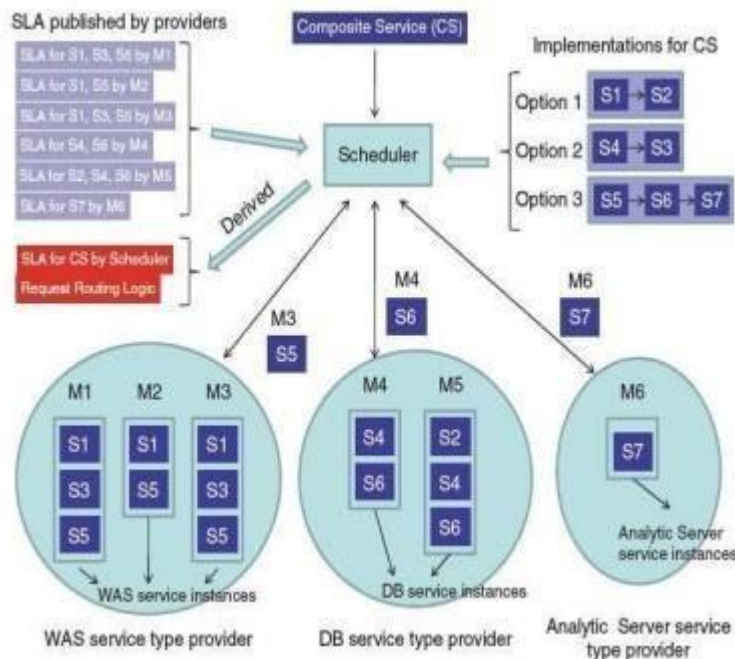
Berikut tugas analitik komposit online dapat direpresentasikan sebagai panggilan untuk Web dan Aplikasi Server (WAS) untuk melakukan pra-pemrosesan tertentu, diikuti dengan sebuah panggilan dari WAS ke server database (DB) untuk mengambil data yang dibutuhkan, setelah itu WAS meneruskan data yang ditetapkan ke server analitik khusus untuk tugas - tugas komputasi data mining yang mahal.

Tugas komposit memiliki beberapa implementasi di pusat data modern IT. Implementasi alternatif dapat memanggil prosedur yang tersimpan pada database untuk menjalankan data mining dan bukan memiliki server analitik khusus untuk melakukan tugas ini. Implementasi alternatif menyediakan distribusi beban vertikal dengan memungkinkan

penjadwalan pekerjaan untuk memilih implementasi WAS dan DB saat analitik server tidak tersedia.

Reusability adalah salah satu tujuan utama dari pendekatan SOA. Sehubungan dengan reusability yang tinggi dari komponen aplikasi, adalah mungkin untuk menentukan alur kerja yang kompleks dengan beberapa cara. Namun sulit untuk menilai, mana yang merupakan penerapan yang terbaik

Pada bagian ini diberikan gambaran sistem arsitektur dan contoh komputasi awan yang disederhanakan (seperti gambar berikut).



Gambar 4. Request routing for SOA-based enterprise computing with multiple implementation Options

Sumber : (herwin:2011)

di mana sebuah proses analitik berjalan pada Web dan Aplikasi Server (WAS), Database Server (DB), dan Server Analytic khusus. proses analitik dapat diimplementasikan oleh salah satu dari tiga pilihan (seperti yang ditunjukkan pada gambar di atas ):

1. Mengeksekusi beberapa pra-pengolahan di WAS (S1) dan kemudian memiliki DB untuk menyelesaikan perhitungan analitik (S2); atau
2. Mengambil data dari DB (S4) ke WAS dan kemudian menyelesaikan sebagian besar perhitungan analitik di WAS (S3); atau
3. Mengeksekusi beberapa pra-pengolahan di WAS (S5), dan kemudian memiliki DB setelah itu mengambil data yang diperlukan (S6), dan akhirnya menampilkan AS untuk melakukan perhitungan sisa analitik (S7).

Proses analitik memerlukan tiga jenis layanan yang berbeda, yaitu layanan jenis WAS, layanan jenis DB, dan layanan jenis AS. S1, S3, dan S5 adalah contoh dari jenis layanan WAS karena mereka adalah layanan yang diberikan /disediakan oleh WAS (Web and Application Server).

Demikian pula, S2, S4, dan S6 merupakan contoh dari jenis layanan DB ( Database Server), dan S7 adalah turunan dari jenis layanan AS (Analytic Server).

Selain itu, ada tiga jenis server : WS server ( M1, M2 , dan M3 ); DB server (M4 dan M5), dan AS server (M6). Meskipun server dapat mendukung hal lain dari jenis layanan yang diberikan, secara umum hal ini tidak selalu terjadi. Sebagai contoh : setiap server dapat mendukung semua contoh jenis layanan perusahaan, kecuali M2 dan M4 adalah server yang kurang kuat sehingga mereka tidak dapat mendukung layanan komputasi yang mahal, S3 dan S2.

Setiap server memiliki Service Level Agreement (SLA) untuk setiap contoh layanan yang mendukung, dan SLA ini diterbitkan dan tersedia untuk penjadwal. SLA termasuk informasi seperti beban profil versus waktu respon dan batas atas permintaan ukuran beban dimana server dapat memberikan jaminan waktu respon nya.

Scheduler bertanggung jawab untuk routing dan mengkoordinasikan pelaksanaan pelayanan komposit /gabungan dari satu atau lebih implementasi Sebuah SLA yang diperoleh hanya dapat digunakan sesuai logika routing. Scheduler dapat memperoleh SLA dan logika routing serta menangani permintaan routing. Atau, Scheduler dapat digunakan hanya untuk tujuan menurunkan SLA dan logika routing saat mengkonfigurasi isi router, seperti (Cisco System Inc), untuk kinerja tinggi dan hardware berbasis routing.

Scheduler juga dapat ditingkatkan untuk melakukan tugas pemantauan yang actual dari QoS (Quality of Service) yang dicapai oleh eksekusi alur kerja dan oleh penyedia layanan individu. Jika scheduler mengamati kegagalan penyedia layanan tertentu untuk QoS yang dipublikasikan, dapat menghitung kembali kelayakan dari QoS dan logika routing sesuai kebutuhan/permintaan yang dapat beradaptasi dengan lingkungan runtime.

## **OpenStack, perangkat lunak Cloud Computing Open Source.**

OpenStack merupakan open source cloud computing software untuk membangun infrastruktur cloud yang reliabel dimana baru saja dipublikasikan beberapa hari lalu yaitu pada tanggal 19 Juli 2010. Tujuan OpenStack adalah untuk memungkinkan setiap

organisasi atau perusahaan untuk membuat dan menyediakan layanan cloud computing dengan menggunakan perangkat lunak open source yang berjalan diatas perangkat keras yang standar.



Gambar 5. Software Open Stack

Sumber : <http://openstack.org>

Terdapat dua jenis OpenStack, yaitu OpenStack Compute dan OpenStack Storage.

**OpenStack Compute** adalah perangkat lunak untuk melakukan otomatisasi saat membuat ataupun mengelola virtual private server (VPS) dalam jumlah besar.

**OpenStack Storage** adalah perangkat lunak untuk membuat object storage yang bersifat scalable serta redundant dengan menggunakan cluster untuk menyimpan data data dalam ukuran terabytes atau bahkan petabytes.

Seluruh kode OpenStack berada dibawah lisensi Apache 2.0. Sehingga memungkinkan siapapun untuk menjalankan, membangun perangkat lunak lain diatas perangkat lunak OpenStack atau mengirimkan perubahan kode entah sebagai patch atau fitur baru.

OpenStack saat ini telah digunakan perusahaan besar hosting seperti Rackspace Hosting dan NASA. Mereka menggunakan teknologi OpenStack untuk mengelola puluhan ribu compute instance dan storage dalam ukuran petabytes.





Gambar 6. Web Rackspace  
Sumber : <http://rackspace.com>

### Amazon Elastic Compute Cloud (EC2).

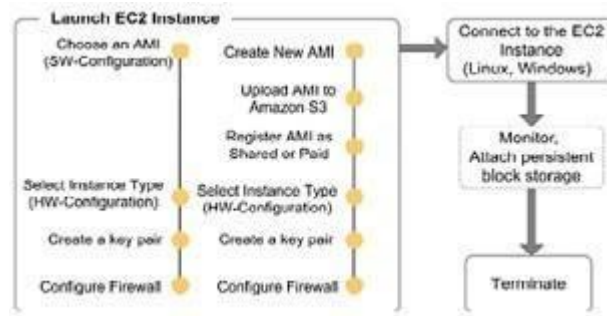
Amazon telah memberikan solusi universal dan komprehensif yang populer untuk Cloud Computing, yang disebut Amazon Elastic Compute Cloud (EC2) (2010). Solusi ini dirilis sebagai versi beta umum yang terbatas pada tanggal 25 Agustus 2006, tetapi tumbuh pesat di tahun-tahun berikutnya.



Gambar 7. Amazon EC2  
Sumber : <http://aws.amazon.com>

EC2 menyediakan banyak fitur yang berguna bagi pelanggan, termasuk sistem penagihan yang terencana dan biaya untuk komputasi yang murah pada tingkat yang sangat mantap (penggunaan memori, penggunaan CPU, transfer data, dll), penyebaran antara beberapa lokasi, elastis alamat IP, infrastruktur yang ada sambungan ke pelanggan

melalui Virtual Private Network (VPN), jasa pemantauan oleh Amazon CloudWatch, dan load balancing elastis. Amazon's EC2 provides virtual machine based computation environments. EC2 menggunakan hypervisor Xen (2010) untuk mengelola Amazon Mesin Gambar (AMI). AMI (Amazon EC2, 2010) adalah "gambar terenkripsi mesin yang berisi semua informasi yang diperlukan untuk perangkat lunak yang kita pakai". Dengan menggunakan interface layanan web sederhana, pengguna dapat memulai, menjalankan, memonitor dan menghentikan kasus mereka seperti ditunjukkan pada Gambar di bawah ini.



Gambar 8. Selain itu mereka dapat dengan cepat menambahkan satu fitur seperti yang disebutkan di atas untuk konfigurasi sesuai dengan apa yang pengguna inginkan.

Sumber : <http://aws.amazon.com>

## GoGrid.

GoGrid memiliki karakteristik umum dengan Amazon di area klasik komputasi awan, dalam hal ini mendukung beberapa sistem operasi melalui gambaran manajemen sendiri, dan mendukung dalam hal menyeimbangkan beban, penyimpanan awan, dan sebagainya . Selain itu, GoGrid menyediakan pelanggan dengan antarmuka web yang user-friendly service, mudah dimengerti demonstrasi video, dan sistem penagihan yang ketat tapi tidak mahal.



Gambar 9. Software Gogrid

Sumber : <http://gogrid.com>

Jadi baik EC2 dan GoGrid, kedua-nya menyediakan fitur dasar dan umum dari Cloud Computing. Perbedaan antara layanan yang mereka (EC2 dan GoGrid) berikan terutama berasal dari model bisnis mereka masing-masing.

Sebagai contoh, GoGrid menyediakan awan (Cloud) bebas dan penyimpanan yang spesifik, sedikit berbeda dari Amazon. GoGrid juga menyediakan Hybrid Hosting, yang merupakan fitur pembeda. Banyak aplikasi namun tidak dapat berjalan dengan baik di lingkungan server yang murni multi-tenant.

Performa Database lebih baik pada dedicated server, dimana EC2 dan GoGrid tidak perlu bersaing untuk input/output sumber daya, situasi ini mirip dengan aplikasi web server. GoGrid menyediakan aplikasi-aplikasi khusus dengan dedicated server yang memiliki jaminan keamanan yang tinggi .

### **Amazon Simple Storage Service (S3).**

Amazon Simple Storage Service (2010) (S3) adalah layanan web penyimpanan online yang ditawarkan oleh Amazon Web Services. S3 dapat diakses pengguna melalui layanan web, REST- style interface HTTP, atau dengan melibatkan antarmuka SOAP. Seperti halnya layanan komputasi awan lainnya, pengguna dapat meminta penyimpanan dalam jumlah kecil atau besar dengan cepat, serta menyediakan sistem penyimpanan sangat terukur.



Gambar 10. Amazon S3

Sumber : <http://aws.amazon.com>

Amazon S3 mengatur ruang penyimpanan ke dalam banyak kotak, dengan setiap kotak diberi namespace yang pada umumnya unik dengan maksud untuk membantu

menemukan alamat data, mengidentifikasi user account untuk pembayaran, dan mengumpulkan informasi penggunaan. Amazon S3 berurusan dengan semua jenis data sebagai objek. Sebuah objek dapat diakses melalui URL yang terdiri dari kunci dan versi ID dengan namespace sebagai awalan.

Pengguna Amazon S3 tersebar di banyak bidang, misalnya, SmugMug, Slideshare dan Twitter. Twitter menggunakan Amazon S3 untuk host images, Apache Hadoop menggunakan S3 untuk menyimpan data komputasi, dan utilitas sinkronisasi online seperti Dropbox dan Ubuntu One gunakan Amazon S3 sebagai tempat penyimpanan dan fasilitas transfer.



Gambar 11. Smug Mug

Sumber : <http://smugmug.com>



Gambar 12. Slide Share

Sumber : <http://slideshare.net>





Gambar 13. Twitter

Sumber : <http://twitter.com>



Gambar 14. Hadoop

Sumber : <http://hadoop.apache.org>



Gambar 15. Ubuntu One

Sumber : <http://one.ubuntu.com>

## Rackspace Cloud.

Rackspace Awan awalnya diluncurkan pada tanggal 4 Maret 2006 dengan nama "Mosso". Dalam tiga tahun berikutnya, ia (Raskspace Cloud) telah mengubah namanya dari "Mosso LLC" menjadi "Mosso: The Hosting Cloud ", dan akhirnya menjadi "Rackspace Cloud" pada tanggal 17 Juni 2009.



Gambar 16. Rackspace Cloud

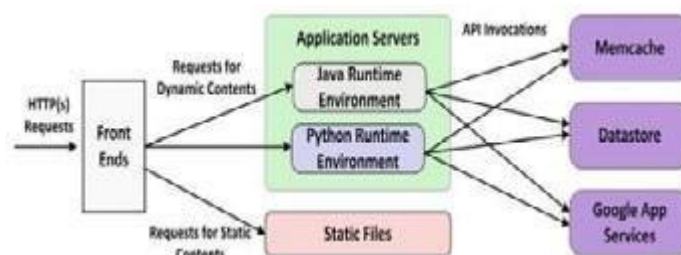
Sumber : <http://rackspace.com>

Perusahaan ini menyediakan layanan termasuk cloud server, cloud file, dan cloud site. Cloud file service adalah layanan penyimpanan awan (cloud) yang menyediakan penyimpanan online yang tak terbatas dan Jaringan Pengiriman Konten untuk media secara komputasi utilitas. Selain control panel online, perusahaan ini menyediakan layanan API (Application Programming Interface) yang dapat diakses melalui Application Programming Interface yang aman dengan kode klien open source.

Rackspace memecahkan masalah keamanan dengan mereplikasi tiga salinan penuh data di beberapa komputer pada beberapa zona, dengan setiap tindakan yang dilindungi oleh SSL (Secure Socket Layer).

## Google App Engine.

Google App Engine (GAE) tujuan utama adalah untuk mengefisienkan pengguna menjalankan aplikasi web. Seperti yang ditunjukkan pada gambar berikut.



Gambar 17. Arsitektur dari Google App Engine

Sumber : (herwin:2011)

Google App Engine mempertahankan Python dan lingkungan runtime Java pada server aplikasi, bersama dengan beberapa Application Programming Interface sederhana untuk mengakses layanan Google.



Gambar 18. Google App Engine

Sumber : <http://developer.google.com>

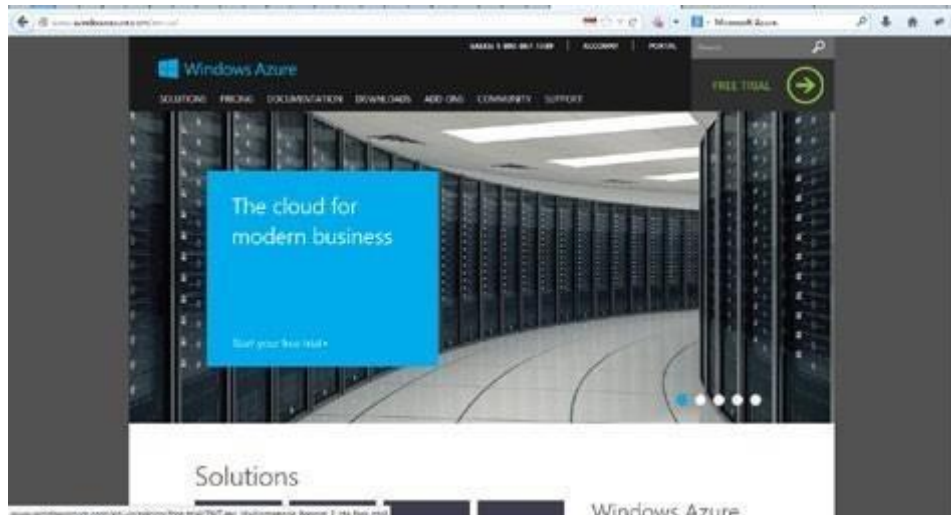
Selanjutnya menyebar permintaan HTTP dengan load balancing dan routing strategi yang didasarkan pada Contents (isi). Runtime sistem yang berjalan pada aplikasi server yang ideal dengan pengolahan logika aplikasi dan menyediakan konten web dinamis, sedangkan halaman statis dilayani bersama oleh infrastruktur Google.

Untuk memisahkan data terus-menerus dari server aplikasi, GAE (Google App Engine) menempatkan data ke dalam Datastore dari sistem file lokal. Aplikasi dapat mengintegrasikan layanan data dan Google App Layanan lainnya, seperti email, penyimpanan foto dan sebagainya melalui API (Application Programming Interface) yang disediakan oleh GAE (Google App Engine) .

Selain layanan, Google juga menyediakan beberapa tool untuk pengembang dalam hal ini membantu mereka (pengembang) membangun aplikasi web dengan mudah di GAE (Google App Engine). Namun, sejak mereka (pengembang) erat terhubung ke infrastruktur Google, ada beberapa pembatasan yang membatasi fungsionalitas dan portabilitas dari aplikasi.

### **Microsoft Azure.**

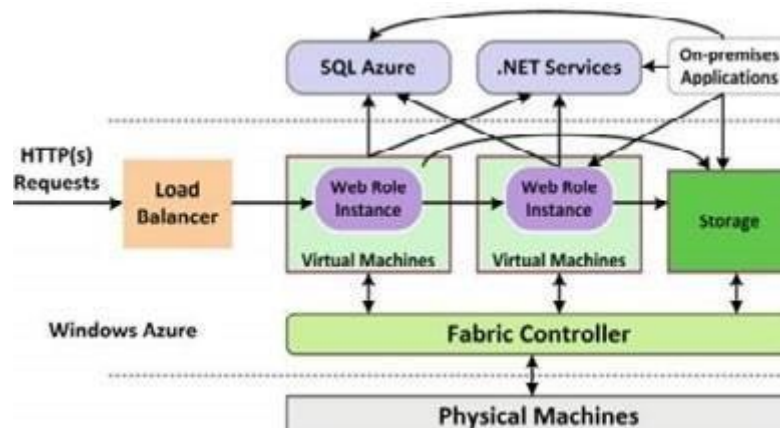
Strategi awan Microsoft adalah untuk membangun sebuah platform awan yang mana pengguna dapat memindahkan aplikasi mereka ke dalam cara yang sempurna, dan memastikan bahwa sumber daya yang dikelola dapat diakses untuk kedua layanan awan tersebut pada aplikasi lokal .



Gambar 19. Windows Azure

Sumber : <http://windowsazure.com>

Untuk mencapai ini, Microsoft memperkenalkan Windows Azure Platform (WAP), yang terdiri dari sistem operasi Awan (Cloud) yang bernama Windows Azure, dan satu set layanan pendukung, seperti ditunjukkan pada gambar berikut.



Gambar 20. Arsitektur dari platform Windows Azure

Sumber : (herwin:2011)

Windows Azure adalah bagian utama dari WAP (Wireless Application Protocol). WAP adalah sebuah protokol atau sebuah teknik messaging service yang memungkinkan sebuah telepon genggam digital atau terminal mobile yang mempunyai fasilitas WAP, melihat/membaca isi sebuah situs di internet dalam sebuah format teks khusus.

Ini mempekerjakan mesin virtual sebagai lingkungan runtime nya. Penawaran Aplikasi dalam awan Microsoft dibagi menjadi dua jenis: instansi peran Web, yang dapat melayani permintaan web melalui layanan informasi internet ; dan instansi peran pekerja, yang hanya dapat menerima pesan dari instansi peran Web lain atau aplikasi lokal.



Windows Azure mempekerjakan "controller kain" untuk mengelola semua mesin virtual dan server penyimpanan pada mesin fisik di pusat data Microsoft.

Windows Azure menggunakan sebuah pengendali kontrol untuk mengelola semua mesin virtual dan server penyimpanan pada mesin fisik di pusat data Microsoft. Serupa dengan Datastore di GAE (Google Application Engine), WAP (Wireless Application Protocol) juga menyediakan layanan database yang disebut SQL Azure, untuk menyimpan data di awan (cloud). Salah satu fitur dari SQL Azure adalah menyediakan alat untuk sinkronisasi data dilokasi lokal.

Layanan infrastruktur didukung oleh WAP melalui layanan. NET yang saat ini include dengan kontrol akses dan layanan ekspos. Keduanya tersedia untuk layanan Cloud dan layanan lokal.

Berikut ini ada 11 top open source cloud application yang diambil dari GigaOm untuk keperluan pelayanan, pendidikan, support, general item of interest, dan lainnya.



Gambar 21. GigaOm

Sumber : <http://gigaom.com>

1. **Eucalyptus.** Ostatic menggemparkan berita dimana UC Santa Barbara membuat sebuah open source cloud project tahun kemarin. Dikeluarkan sebagai open-source (dengan menggunakan lisensi FreeBSD- style) Eucalyptus dapat digunakan untuk infrastruktur cloud computing dalam cluster yang dapat menduplikasi fungsionalitas Amazon EC2, Eucalyptus secara langsung menggunakan command - line tool dari Amazon. Sebagai langkah awal Eucalyptus System terlebih dahulu membuat venture funding, untuk membiayai staff termasuk arsitek dari Eucalyptus project. Baru baru ini mereka mengeluarkan update software framework nya, yang juga dilengkapi dengan fitur cloud computing yang akan digunakan pada Linux Ubuntu versi terbaru.



Gambar 22. Ostatic

Sumber : <http://ostatic.com>

2. **Red Hat's Cloud.** Salah satu pemain open - source terlama Red Hat memang telah memfokuskan diri pada cloud computing. Pada akhir juli kemarin, Red Hat membuka sebuah Open Source Cloud Computing Forum, yang berisi banyak persentasi mengenai ide perpindahan dari open source untuk mengikuti teknologi cloud. Anda dapat mengikuti semua free webcast dari semua persentasi Redhat. Pembicaranya Rich Wolski (CTO dari Eucalyptus Systems), Brian Stevens (CTO dari Red Hat), dan juga Mike Olson (CEO dari Cloud era).

Steven akan membawa anda mengenai strategi Red Hat terhadap cloud computing.

Novell juga open source sedang mencoba untuk memfokuskan ke cloud computing.



Gambar 23. Radhat

Sumber : <http://radhat.com>



tested Hadoop distribution sendiri. Cloudera nampaknya saat ini menjajikan untuk tahap awal yang memberikan support komersil untuk Hadoop.



Gambar 26. Cloudera

Sumber : <http://cloudera.com>

5. **Puppet.** Adalah sebuah teknologi Virtual server yang dapat di implemetasikan pada cloud computing, dan juga dapat digunakan sebagai Reductive Lab open - source software (kurang faham maksudnya apa), software ini dibangun dengan menggunakan Cfengine system, dan hebatnya banyak system administrator yang memanfaatkan software ini . Anda dapat dengan mudah mengatur berapapun jumlah virtual machine dan dapat melakukan automated routine, tanpa harus melakukan complex scripting.



Gambar 27. Puppet

Sumber : <http://puppetlab.com>

6. **Enomaly.** Adalah Elastic Computing Platform (ECP) yang merupakan akar dari Enomalism open - source provisioning and management software, teknologi ini di desain untuk men gatur kompleksitas dari implementasi infrastruktur cloud. ECP adalah sebuah

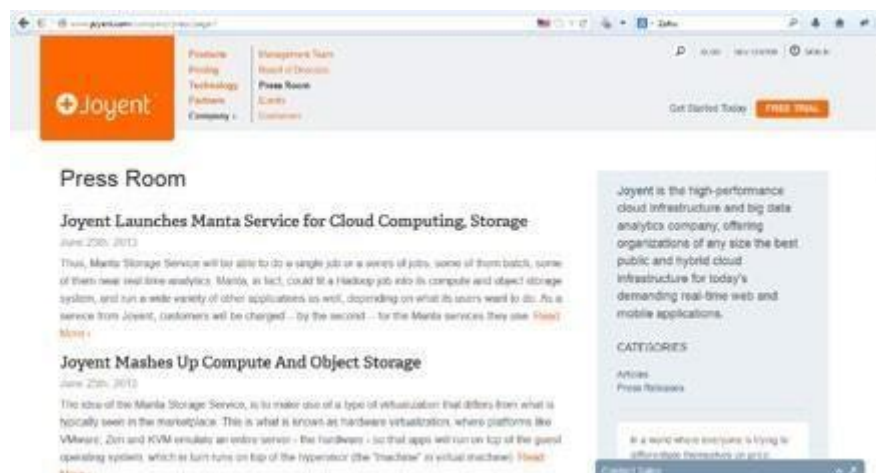
programmable virtual cloud computing infrastructure untuk ukuran kecil, sedang dan juga enterprise besar dan anda dapat membaca lebih detail disini.



Gambar 28. Enomaly

Sumber : <http://enomaly.com>

7. **Joyent.** Adalah sebuah software yang didirikan pada Januari awal tahun ini, yang memulai open - source cloud dengan memanfaatkan JavaScript dan Git. Infrastruktur Joyent cloud hosting dan cloud management software membuka banyak open - source tools untuk public dan private cloud. Perusahaan ini juga membantu mengoptimasi kecepatan implementasi dari open - source MySQL database untuk penggunaan cloud use.



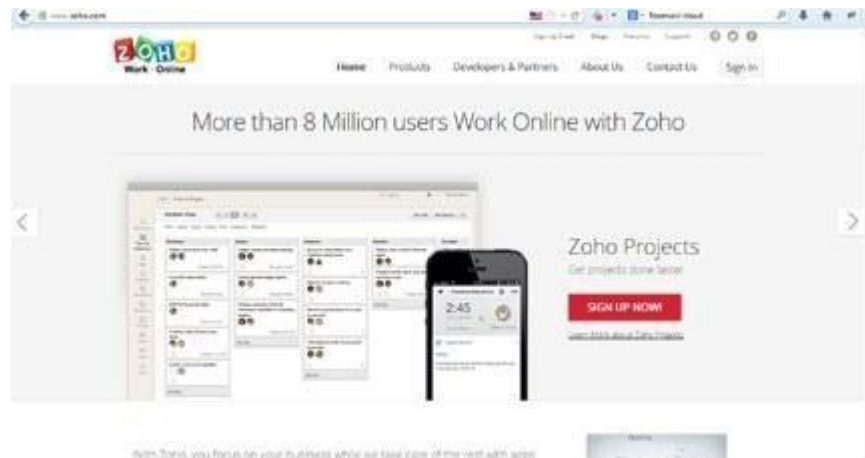
Gambar 29. Joyent

Sumber : <http://joyent.com>

8. **Zoho.** Banyak orang mengenal Zoho sebagai free, online application, yang menjadi pesaing dari Google Docs. Yang terpenting untuk diketahui adalah bawasanya Zoho core adalah betul betul open source; sebuah contoh bagaimana solusi SaaS dapat



bekerja secara harmonis dengan open source. Anda dapat menemukan bagaimana Zoho mengimplementasikan open - source tool melalui interview mereka.



Gambar 30. Zoho

Sumber : <http://zoho.com>

9. **Globus Nimbus.** Open - source toolkit ini mampu merubah bisnis anda dari infrastruktur cluster menjadi Infrastructure as a Service (IaaS) cloud. Amazon EC2 interface digunakan sepenuhnya namun ini bukan hanya sebuah interface yang dapat anda manfaatkan.



Gambar 31. Globus Nimbus

Sumber : <http://nimbusproject.org>

10. **Reservoir.** Adalah sebuah inisiatif dari European research untuk mengembangkan virtualized infrastructure and cloud computing. Akhirnya membawa mereka untuk mengembangkan teknologi open - source untuk cloud computing, dan membantu para pengguna bisnis untuk menghemat biaya IT.



Gambar 32. Reservoir

Sumber : <http://reservoir-tp7.eu>

11. **OpenNebula.** OpenNebula VM Manager adalah sebuah komponen dasar dari Reservoir. Ia adalah sebuah jawaban open-source untuk berbagai macam jenis virtual machine management yang banyak di gunakan secara proprietary, Interface nya pun dapat dengan mudah dipahami dengan cloud infrastructure tools and services. "OpenNebula adalah sebuah open-source virtual infrastructure engine yang akan memberikan anda implementasi dan replacement dari virtual machines pada physical resources," menurut project lead mereka.



Gambar 33. Open Nebula

Sumber : <http://opennebula.org>

Nampaknya banyak open source tools sudah mulai berkompetisi dalam dunia cloud computing. Hasil akhir dari ini tentu saja nantinya kita akan menemukan fleksibilitas dari organisasi untuk mengkostumasi pendekatan yang mereka inginkan. Open source cloud akan memberikan potensi akan harga yang sangat kompetitif untuk mendapatkan service cloud.

# Daftar Pustaka

1. Anggeriana Herwin, Cloud Computing, 2011
  2. Berkah I Santoso, Perkembangan Virtualisas, 2012
  3. Berkah I Santoso, Cloud Computing dan Strategi TI Modern, 2012
  4. Berkah I Santoso, Mobile Backend as a Services, 2012
  5. Demystifying the Cloud An introduction to Cloud Janakiram MSV Cloud Computing Strategist [www.janakiramm.net](http://www.janakiramm.net) | [mail@janakiramm.net](mailto:mail@janakiramm.net)
  6. Llorente, I. M. (July 2008). Towards a new model for the infrastructure grid. *Panel From Grids to Cloud Services in the International Advanced Research Workshop on High Performance Computing and Grids, Cetraro, Italy.*
  7. [http://id.wikipedia.org/wiki/Komputasi\\_awan](http://id.wikipedia.org/wiki/Komputasi_awan)
  8. <http://infreemation.net>
  9. <http://docs.google.com>
  10. <http://www.biznetnetworks.com/En/?menu=cloudhosting>
  11. <http://detik.com>
  12. <http://www.salesforce.com>
  13. <http://www.amazon.com>
  14. <http://www.okezone.com>
  15. <http://www.kompas.com>
  16. <http://www.insw.go.id/>
  17. <http://www.windowsazure.com/en-us/>
  18. <http://www.chip.co.id>
  19. <http://www.cloudindonesia.or.id>
  20. <http://eliyaningsih.wordpress.com/2020/09/11/praktek-aplikasi-membuat-layanan-cloud-storage-sendiri-dengan-owncloud/>
  21. <http://id.wikipedia.org/wiki/OwnCloud>
  22. <http://owncloud.org/>
-



23. [www.youtube.com](http://www.youtube.com)
24. <http://www.hightech-highway.com>
25. <http://basingna.wordpress.com>
26. <http://kompas.com>
27. <http://techno.okezone.com>