

## DESAIN INTEGRASI ZIMBRA EMAIL AND COLLABORATION CLOUD SERVER UNTUK PENINGKATAN EFEKTIFITAS DAN EFISIENSI KOMUNIKASI DAN KOLABORASI

Rico Agung Firmansyah

Magister Teknik Informatika STMIK AMIKOM Yogyakarta  
Jl. Ring Road Utara Condong Catur Depok Sleman Yogyakarta  
email : [ricoagung@gmail.com](mailto:ricoagung@gmail.com)

### Abstrak

Kesuksesan suatu organisasi dalam mengelola sumber dayanya pada umumnya bertumpu pada baik-tidaknya pengelolaan/menejemen seluruh sumberdaya organisasi yang dimilikidimana hal itu bergantung pada baik tidaknya koordinasi dan kolaborasi antara pengelolanya. Mis-komunikasi serta rumitnya jalur biroksasi internal organisasi dapat menghambat produktifitas, bahkan menjadi masalah tersendiri.

Zimbra Email and Collaboration Cloud Server merupakan suatu aplikasi jaringan yang menawarkan banyak layanan yang dapat mendukung kemudahan komunikasi dan kolaborasi sebagai alat meningkatkan produktifitas organisasi. Aplikasi ini dapat digunakan sebagai Email Server, VOIP telecommunication Server, Office Document Collaboration, EHRM Automation Server, Report Server, Calendar and Organizer Server, serta beberapa fitur lain yang dapat ditambahkan (embedded) kedalam satu service.

Multi-Services yang mampu diaplikasikan dalam sebuah Cloud Platform tentunya dapat menekan biaya pengadaan dan pengelolaan sumber daya IT dalam organisasi, berbeda jika kita memilih banyak layanan ke banyak vendor di internet yang membuat biaya sangat tinggi. Efisiensi, efektifitas dan optimalisasi sumberdaya organisasi dapat diwujudkan dengan adanya sistem ini

**Kata kunci** : cloud application server; sharing, communication and collaboration server; open document; multi-services application server; peningkatan produktifitas organisasi

### 1. Pendahuluan

Sistem informasi saat ini menjadi pendukung utama dalam operasional bisnis dan organisasi lainnya untuk meningkatkan efektifitas dan mengefisienkan proses bisnis organisasi tersebut [1]. Komunikasi sekarang ini menjadi hal yang sangat mutlak dimiliki dan dimanfaatkan setiap individu dan/atau organisasi sebagai salah satu dari beberapa alat pencapaian tujuan tertentu. Sukses tidaknya individu/organisasi tak luput dari

sukses tidaknya metode komunikasi yang digunakannya, seberapa berkualitas, efektif dan efisiennya komunikasi yang digunakan [2].

Banyak model komunikasi yang dapat diterapkan, mulai dari komunikasi berbasis audio, visual, audio-visual, dan kolaborasi data secara umum, dimana masing-masing metode memiliki karakteristik, kelebihan dan kekurangannya. Komunikasi audio yang sudah lazim dan cukup efektif digunakan manusia ternyata memiliki keterbatasan. Komunikasi visual-audio mampu menutupi keterbatasan komunikasi audio dalam hal visualisasi. Komunikasi realtime pada fitur audio-visual mampu memberi kesan, manfaat dan penggunaan secara khusus dibanding audio saja. Untuk peningkatan produktifitas kerja fitur audio-visual saja tentunya tidak cukup, karena fitur transfer file, file sharing dan document collaboration tidak dapat dihindari oleh kedua fitur tersebut, fitur collaboration server yang mampu menangani kekurangan pada segi ini [2] [3][4].

Sebuah organisasi, perusahaan, ataupun lembaga apapun pasti setidaknya membutuhkan tiga fungsi tersebut, fungsi yang dapat digunakan individu didalamnya untuk berkomunikasi baik secara audio, visual, dan kolaborasi file dokumen pekerjaan mereka agar pencapaian visi organisasi dapat dengan mudah diperoleh. Namun untuk menerapkan ketiga fitur tersebut diatas tidaklah mudah dan murah. Logikanya tiga fitur tersebut dapat aktif digunakan jika ketiga service tersebut didesain dan dikelola dengan baik. Sebuah Server untuk sebuah fitur diatas menjadi pilihan utama pada beberapa waktu lalu. Artinya akan ada minimal tiga node server secara independen memberi layanannya masing-masing. Namun dengan membuat ketiga layanan berjalan independen akan berdampak pada banyak aspek, diantaranya skalabilitas yang kurang baik, kemudahan menejemen dan maintenace tidak maksimal, serta reliable dan stabilitas performa dari ketiga tidak dapat maksimal.

Zimbra Platform merupakan aplikasi jaringan yang memberikan layanan utama berupa email system dan collaboration system serta memungkinkan untuk die-

*embed*-kan *voice* dan/atau *visual communication services*. Artinya dengan menggunakan sebuah *platform* *Zimbra Collaboration Services* ini kita dapat memiliki ketiga model komunikasi sekaligus dalam sebuah sistem layanan jaringan. Hal ini membuat organisasi yang memiliki kebutuhan dasar akan

Jika ketiga layanan ini bekerja secara bersamaan, tentunya akan sangat membebani performa *node server* tunggal dalam melayani permintaan layanan dari *user*. Maka dari itu, pemanfaatan beberapa *node server* yang terintegrasi sebagai *base system* layanan tersebut diatas harus memiliki fitur skalabilitas yang tinggi, fitur *elastic* yang baik, performa yang *reliable*, stabil serta memberi kemudahan dalam memenejemen dan memaintain *resources* yang dimilikinya [3][5]. Semua fitur tersebut dapat kita temukan pada *Cloud Computing Infrastructure* (IaaS dan SaaS). *Cloud computing* merupakan teknologi yang memungkinkan *resource* IT digunakan untuk beragam *platform*, kode program, dan aplikasi yang berbeda, agar dapat terintegrasi dalam sebuah atau beberapa layanan. Hal ini memungkinkan *cloud computing* menjadi wadah dari ketika macam model layanan komunikasi tersebut diatas agar bekerja pada lingkungan yang sama. Dengan menggunakan *Cloud Environment* sebagai *base system* teknologi Alasan inilah yang menjadi titik awal penelitian ini dibuat.

## 2. Tinjauan Pustaka

### 2.1 Zimbra Collaboration Platform

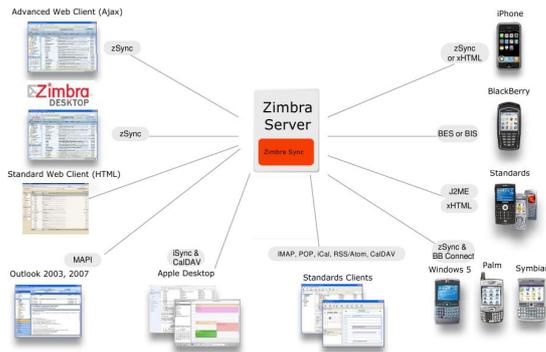
*Zimbra Collaboration Suite (ZCS)* merupakan produk *groupware* dari Zimbra, Inc, California, Amerika Serikat yang kemudian dibeli oleh Yahoo! pada September 2007 lalu diakuisisi oleh VMWare pada 2010 hingga sekarang. VM Ware merilis *Platform* utama Zimbra yang berbasis Open Sources dan versi komersilnya dengan sedikit perbedaan fitur masing-masing. Semua varian *platform* zimbra memiliki fitur utama *Email system* terintegrasi serta mampu berfungsi sebagai SaaS dilingungan *cloud system* yang cocok untuk layanan skala kecilhingga skala enterprise serta dukungan banyak fitur tambahan lainnya yang dapat ditanamkan secara opsional [3] bagi *user* yang menginginkannya, antara lain sebagai berikut [4] :

1. *Search engine* dan *file indexing services* untuk tata kelola pengarsipan *file* secara umum maupun *file* khusus (email dan fitur lainnya yang ditanam ke *zimbra platform*).
2. *Native Security Mail System*, memungkinkan keamanan transfer data antara *sever* dan *client* lebih aman dengan penambahan fitur spam

filter and blocking serta antivirus dan hak akses berjenjang bagi pengelola sistem

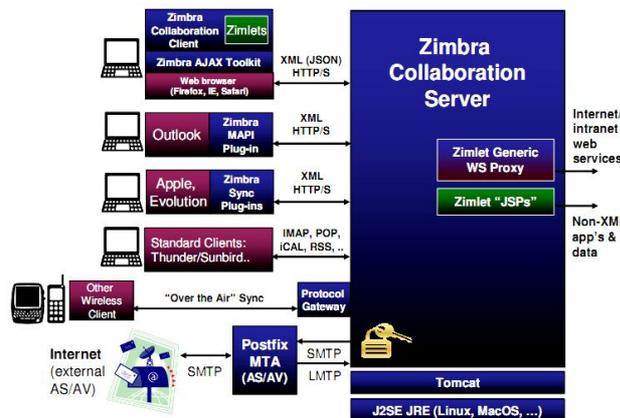
3. *Web Privacy*, fitur ini terdiri dari mode komunikasi dari/ke *server* secara aman (ternkripsi) via SSL, VPN, SSH, atau manual tunneling. Fitur kewanaman yang utama dan terstandarisasi untuk kenyamanan penggunaannya.
4. *Server-Centric Deployment*, merupakan fitur utama yang dapat memudahkan pembangunan sistem secara terpusat serta memudahkan dalam *maintennace* dan *management resources*.
5. *Group Callendar and Organiser*, memudahkan *user* melakukan integrasi jadwal atau *event* ke *email system*.
6. *Address book and contact manager*, fitur ini memudahkan kita menyimpan kontak baik via *email* maupun *sync* ke perangkat *Hand held*.
7. Dukungan *Web 2.0*, *Ajax*, *XML*, *REST*, dan *SOAP* dan sebagainya, meberikan efek positif bagi layanan zimbra yang dapat digunakan pada banyak *platform* disisi *client (interoperability)*.
8. *Collaborative authoring and specific collaboration-oriented over internet protocols*, memberikan performa yang ringan namun kaya akan fitur bagi pengguna kolaborasi *file*, dokumen, agenda, kalender, *Chating/ messaging*, *tele-convrence*, *tele-podcasting*, dan *video on demand*.
9. *Voice communication (VOIP) support system*, juga menambah fitur komunikasi audio antar pengguna layanan dapat mungkin digunakan.
10. *Unified archieving and rapid-elastic storage system* memberikan keamanan pengarsipan data, kemudahan pengindekan dan pencarian, serta memudahkan pengelolaan sumberdaya bagi pemilik/pengelola layanan.
11. *SaaS compatibility*, memungkinkan pengelola layanan dengan mudah mendapatkan manfaat *cloud computing* infrastruktur pada sistem zimbra yang dibuatnya.

Semua layanan yang dapat dimanfaatkan dari *Zimbra platform* dapat dibangun pada minimum sistem requirement yang tidak terlalu tinggi, namun juga dapat digunakan pada spesifikasi yang tinggi untuk skala *enterprise*. Arsitektur zimbra *platform* dapat dilihat pada gambar berikut ini.



Gambar 1. Layanan Zimbra Collaboration Suite

2. Peningkatan performa sistem, karena *cloud computing* bisa digunakan pada beberapa *host* yang terhubung untuk saling melakukan pekerjaan/tugas komputasi bersama-sama.
3. Efisiensi penggunaan *resource*. Fitur virtualisasi, *multi-tenancy* pada *cloud computing* berdampak pada keseluruhan *resources*-nya dapat digunakan bersama-sama (*sharing*), *resource* yang terpakai hanyalah dibutuhkan pengguna saja.
4. Pengurangan biaya operasional karena penggunaan *resource* hanya yang diperlukan saja, serta keuntungan-keuntungan lain yang merupakan efek samping dari keempat keuntungan utama tersebut diatas.



Gambar 2. Zimbra Collaboration Platform

Ada banyak tipe/model komputasi awan (*cloud computing*) yang dapat digunakan, seperti :

- a. *Infrastructure as a service (IaaS)*
- b. *Platform as a service (PaaS)*
- c. *Software as a service (SaaS)*
- d. *Storage as a service (STaaS)*
- e. *Security as a service (SECaaS)*
- f. *Data as a service (DaaS)*
- g. *Database as a service (DBaaS)*
- h. *Test environment as a service (TEaaS)*
- i. *Desktop virtualization*
- j. *API as a service (APIaaS)*
- k. *Backend as a service (BaaS)*

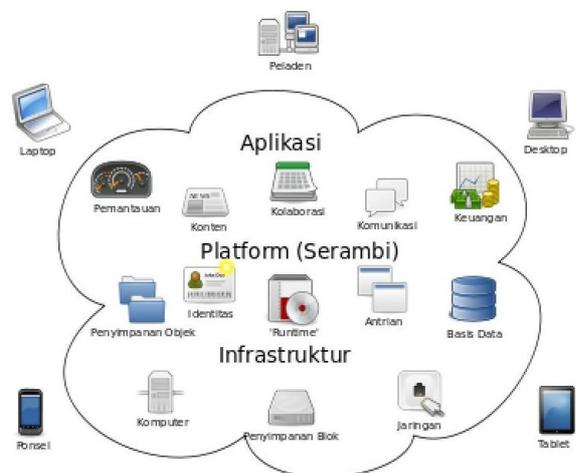
## 2.2 Komputasi Awan (Cloud Computing)

Komputasi awan adalah serangkaian model yang dapat membagi sumberdaya komputasi (seperti *hardware*, *process*, *network*, *storage*, *application*, *services*) kepada *user* dalam infrastruktur jaringan untuk kemudahan penggunaan, dan pelayanan [5][6]. Komputasi awan merupakan abstraksi dari infrastruktur yang kompleks dan tersembunyi, di mana kapabilitas terkait informasi disajikan sebagai layanan (*SaaS*) sehingga pengguna dapat mengaksesnya melalui jaringan komputer tanpa mengetahui apa yang ada didalamnya. *Cloud computing* merupakan sistem virtualisasi yang mampu menampung beragam *platform*, bahasa program (*coding*), aplikasi dan infrastruktur yang berbeda agar dapat bekerja dalam satu *environment* yang terintegrasi, menggunakan *resources* minimal (sesuai penggunaan) namun tetap mempertahankan performa masing-masing sistem yang memiliki perbedaan karakteristik.

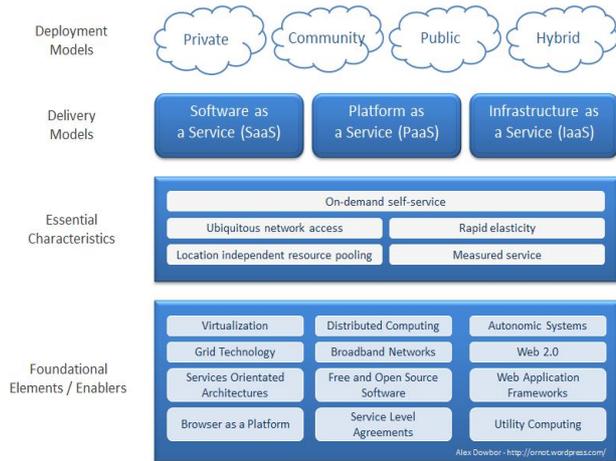
*Cloud Computing* menawarkan beberapa keunggulan, antara lain :

1. *Scalability*. Dengan adanya *cloud system* perpindahan, pengurangan, penambahan dan/ atau pengembangan infrastruktur *hardware* dan *software* menjadi lebih mudah.

*IaaS*, *Paas*, dan *SaaS* merupakan model/tipe layanan utama, karena ketiganya menjadi acuan utama untuk mengaktifkan tipe/fitur yang lain. Dengan kata lain, mengaktifkan tipe *cloud* apapun harus menggunakan tiga tipe utama tersebut sebagai *base* sistemnya. Gambar 3 berikut memvisualisasikan infrastruktur komputasi awan, sedangkan gambar 4 memvisualisasikan diagram penggunaan sistem *cloud computing*.



**Gambar 3. Infrastruktur Komputasi Awan**



**Gambar 4. Visualisasi model penggunaan Cloud Computing**

Dua gambar diatas menjelaskan bahwa *cloud computing system* menggunakan satu atau beberapa *hardware* yang terkoneksi untuk melakukan komputasi kerja sesuai permintaan *user* (layanan) melalui virtualisasi OS dan aplikasi (*software*). Beberapa *user* yang menggunakan layanan yang spesifik akan dilayani oleh OS virtualisasi untuk menggunakan *hardware* yang dibutuhkan masing-masing *user*. Tulisan ini berfokus pada desain integrasi dua buah sistem kedalam lingkungan komputasi awan berbasis IaaS, PaaS dan SaaS.

**3. Metode Penelitian**

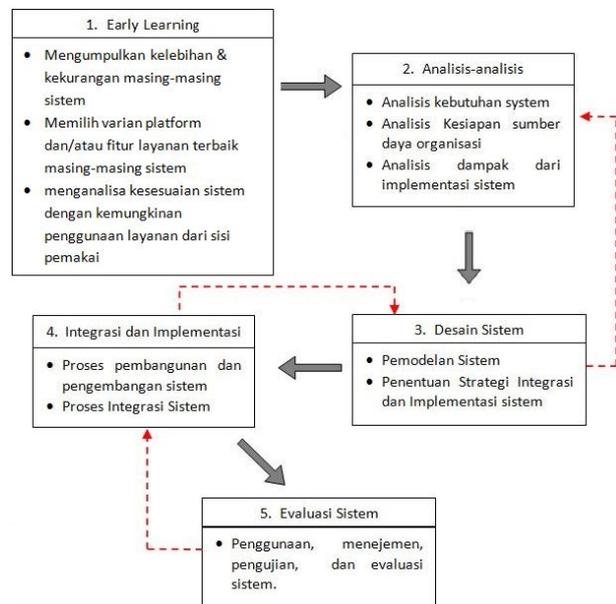
Metodologi yang digunakan dalam penelitian ini tergolong metodologi eksperimental, karena fleksibilitas yang tinggi dari sistem *cloud computing* yang mampu dibangun dari berbagai jenis *platform* dan model maka dapat menghasilkan banyak variasi serta keunikan sistem baru. Beberapa aplikasi sistem yang terbukti baik dan bermanfaat untuk pengguna namun rumit dalam pembangunan infrastruktur dan pengoperasian sistem menjadi modal awal eksperimen integrasi sistem ke dalam *cloud environment*.

Tahapan pembangunan *system* pada penelitian ini mengacu pada *frame work* yang terstandarisasi untuk pembangunan sistem berbasis *cloud*, yakni **ROCCA** (*Roadmap for Cloud Computing Adoption*) [7]. ROCCA merupakan hasil riset yang dilakukan Faith Simba sebagai panduan dalam membangun *Cloud System*. ROCCA menggunakan beberapa standar untuk memastikan tercapainya keberhasilan pembangunan *system*, diantaranya ISO 27001, ISO 27002, COBIT (*Control Framework for Information dan Related Technology*), dan ITIL (*Information Technology Infrastructure Library*). Dokumen ini tidak akan membahas terperinci mengenai *framework* yang

menjadi bahan dasar ROCCA sebagai basis *framework* pembangunan sistem.

Proses desain dan integrasi pada sistem Zimbra dan beberapa fitur tambahan kedalam *cloud environment* terbagi menjadi 5 tahap ekperimental, yakni :

- a. Tahap *Early Learning*, merupakan tahap dimana peneliti dan pihak pengembang perlu memahami kelebihan dan kekurangan sistem yang akan dibangun. Pemilihan *platform*, aplikasi, fitur, accessibility pengguna dan sistem yang baik perlu dikumpulkan dan dianalisa pada tahap ini.
- b. Tahap Analisis. Tahap ini diperlukan untuk menganalisa kebutuhan sistem, kesiapan organisasi yang akan menggunakan sistem, reliabilitas dari pengembangan sistem, serta dampak dari implementasi sitem.
- c. Tahap Desain Sistem. Tahap ini digunakan untuk menentukan strategi terbaik yang memungkinkan untuk digunakannya sistem yang akan dibangun.
- d. Tahap Integrasi dan Implementasi Sistem. Tahapan ini merupakan tahapan proses pembangunan sistem dan integrasi dua model sistem yang berbeda hingga menjadi sebuah sistem terintegrasi seperti yang diharapkan.
- e. Tahap penggunaan, manajemen, pengujian dan evaluasi. Kelima tahapan diatas dapat divisualisasikan seperti pada gambar 5 berikut.



**Gambar 5. Tahapan Penelitian**

#### 4. Hasil dan Pembahasan

Mengacu pada standar *framework* pendukung diatas, maka desain dan pembangunan sistem akan dilakukan dengan beberapa tahap, sama seperti pembahasan pada bab sebelumnya (5 tahap pembangunan sistem). Pembahasan mengenai tahap analisis dan evaluasi (tahap 2 dan 5) tidak akan dibahas karena tahapan ini bergantung pada objek penelitian yang spesifik (organisasi/lembaga). Pembahasan akan difokuskan pada pemilihan alternatif sistem *Zimbra Collaboration Platform*, dan *Cloud Environment* beserta desain dan implementasinya.

##### 4.1. Tahap *Early Learning*

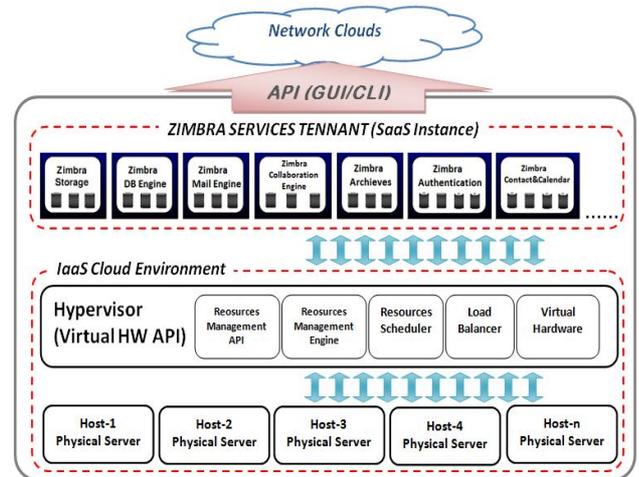
Proses penting yang harus dilakukan pada tahap ini adalah memilih alternatif *platform* IaaS, PaaS, dan SaaS maupun fitur tambahan lainnya yang sesuai dengan keinginan namun tetap cocok untuk digunakan pada *Zimbra platform*. Terdapat beberapa alternatif *cloud virtual environment* yang dapat digunakan sebagai IaaS sistem, diantaranya seperti: *Open Stack*, *Cloud Stack*, dan *Open Nebula*, *Eucalyptus*, dan sebagainya. *Open Stack* dengan kompleksitas dan kapasitas kemampuan sistem yang cukup baik, mungkin menjadi alternatif terbaik diantara yang lain meskipun membutuhkan pemahaman teoritis dan teknikal yang lebih bagi pengembang sistem.

Berbeda dengan IaaS yang fokus menangani manajemen infrastruktur cloud secara hardware (*physical system*) yang akan dikonversi ke mode virtualisasi (*logical system*), beberapa pilihan PaaS dapat dipilih sebagai base system instalasi SaaS sistem *Zimbra*. Misalnya sistem operasi Debian, Ubuntu, Red Hat, Centos, dan sebagainya yang kompatibel terhadap sistem *zimbra*. Sedangkan yang termasuk kedalam SaaS adalah *zimbra platform* itu sendiri beserta fitur tambahan yang dipilih dan disesuaikan dengan kebutuhan penggunaannya.

*Zimbra* menyediakan *system life cycle* (market) untuk menghubungkan para *developer* sistem baik yang non komersial maupun yang komersial ke pengguna sistem baik korporasi ataupun *end user*. Sebagai contoh, fitur VOIP yang dapat diintegrasikan ke lingkungan *zimbra* saja ada beberapa vendor yang menawarkan berbagai varian dengan fitur yang beragam sesuai keinginan. Selama ini pengamatan penulis mengenai kondisi market *zimbra* masih cukup terbuka dan cukup dinamis. Fitur lainnya pun tidak kalah menariknya seperti VOIP diatas. Fitur-fitur yang akan dijadikan layanan bagi pengguna nantinya akan dimasukkan dalam kategori *SaaS Cloud Environment*.

##### 4.2. Desain Sistem

Strategi perancangan sistem integrasi antara SaaS kedalam lingkungan IaaS *Cloud* bisa diilustrasikan sebagai berikut.



Gambar 6. Desain Integrasi LMS-SIS berbasis *Cloud Computing*

*Hardware* pendukung IaaS haruslah memiliki spesifikasi dukungan terhadap teknologi Virtualisasi (VT: *Virtual Technology*). Banyaknya host yang digunakan bergantung pada kemampuan organisasi, minimal 1 host berfungsi sebagai *management-host-controller* sekaligus sebagai *node1, node2, node3, hingga node-n* secara virtual. Jika kebutuhan layanan tidak mampu dikerjakan sebuah host yang berfungsi ganda, maka setiap *node* dapat digantikan oleh host (*pc/server*) lain yang terhubung ke *system* jaringan agar performa *cloud system* meningkat dan tetap stabil. Kondisi ini sering dikenal dengan istilah *scalability*, dimana kemudahan penambahan kemampuan *server* dapat dengan mudah ditambahkan dengan menambah jumlah *node/host physical server*.

##### 4.3. Tahap Implementasi Sistem

Strategi implementasi integrasi sistem *Zimbra Platform* kedalam *Cloud IaaS* dan *PaaS* bisa dilakukan dalam beberapa tahapan, yakni:

- Persiapan *Hardware* sebagai infrastruktur fisik yang utama.
- Desain *Topology* dan *Management Transaction Rule*.
- Instalasi dan interkoneksi fisik (*topology*) antar *physical host*.
- Instalasi dan konfigurasi OS Virtualisasi (IaaS) dan *Logical network, logical node, logical storage*, dan *logical infrastruktur* lainnya.
- Instalasi dan konfigurasi *Platform service (PaaS)* kedalam *IaaS running-system*.
- Instalasi dan konfigurasi *Zimbra* kedalam *PaaS running-system*.
- Instalasi dan konfigurasi aplikasi-aplikasi pendukung lainnya sebagai layanan tambahan .
- Kostumisasi API (*Application Interface*) sebagai sarana interaksi *user* dengan sistem

- i. Testing dan Evaluasi sistem

## 5. Kesimpulan dan Saran

Komunikasi dari waktu ke waktu memiliki style masing-masing. Pada masa kini, banyak sekali model komunikasi yang digunakan individu untuk mempermudah kehidupannya. Hal ini merupakan peluang bagi vendor untuk membangun beragam sistem komunikasi tersebut. Korporasi, dunia pendidikan dan siapapun sebenarnya dapat mengimplementasi layanan komunikasi yang beragam tersebut. Namun pada kenyataannya sangat sulit menyediakan beragam fitur komunikasi yang beragam dan berkualitas kedalam sebuah sistem.

*Zimbra Collaboration Suite (ZCS)* merupakan sistem email dengan kemampuan pengembangan fitur yang luar biasa, mampu diintegrasikan dengan banyak fitur yang berbeda platform dan kegunaan kedalam sebuah email system. Mayoritas model komunikasi dapat diintegrasikan kedalam lingkungan zimbra. Mulai dari layanan server email, kalender, organiser, kontak, voip, tele-convrence, video on demand, podcasting, chat dan messenger, serta kolaborasi dokumen dan berkas kedalam sebuah sistem. Namun dengan banyaknya fitur yang digunakan, maka kebutuhan *resource* juga semakin bertambah. Penambahan *server* independen (*native server*) secara fisik akan membuat sebuah sistem yang seharusnya terintegrasi akan terpisah-pisah, kurang *reliable*, skalabilitas sistem yang rendah, serta tidak mudah untuk dimenajemen dan dirawat (*maintenance*).

Dengan adanya *cloud computing*, perbedaan *platform* bisa diintegrasikan sehingga pembangunan, pengelolaan, penggunaan berasma dan pemeliharaan sistem menjadi lebih mudah, efektif dan efisien. Fitur virtualisasi memberi efek positif dalam hal kemudahan pengembangan sistem (skalabilitas). Fitur *resources sharing* memungkinkan beberapa pengguna dapat saling berbagi sumber daya dengan tetap memiliki *authoritative* yang berbeda dan tetap aman. Kemudahan perbaikan, penggantian dan menejemen sistem juga menambah efektifitas dan efisiensi penggunaan sumber daya manusia. Hal ini menjadikan sistem komunikasi berbasis komputasi awan (*cloud computing*) menjadi kaya fitur tetapi semakin murah, serta dapat efektif untuk meningkatkan kualitas hidup manusia.

Disisi lain, sumberdaya manusia sebagai penggerak sistem komputerisasi ini masih minim, analisa pembangunan sistem ini dalam lingkup nasional menjadi sangat mungkin dan bermanfaat untuk kesejahteraan bangsa jika pemerintah ikut andil dalam hal pengadaan dan penanganannya. Harapannya,

kualitas komunikasi dapat merata dan diraskan oleh banyak individu karena mudah dan murah nya menggunakan teknologi komunikasi. Makin mudah berkomunikasi, tentunya akan lebih mudah manusia mencapai apa yang dicita-citakan.

## Daftar Pustaka

- [1] Firmansyah, Rico Agung, 2012, *Desain Integrasi Learning Content Management System pada Cloud-base Sistem Informasi Sekolah Sebagai Peningkatan Keunggulan Daya Saing*, Prosiding Seminar Nasional Teknologi Informasi dan Multimedia AMIKOM Yogyakarta, 19 Januari 2013, ISSN: 2302-3805
- [2] Azraai, Onn Puade, dkk. *Visualising Collaboration: Qualitative Analysis of an Email Visualisation Case Study*, 11th International Conference Information Visualization-IV'07, Juli 2007
- [3] WANG, Yuan, dkk. *DXEC: Integrating Database, XML and Email for Collaboration*. The 9th Intemational Conference on Computer Supported Cooperative Work in Design, 2008
- [4] Robbins, D., 2011, *Cloud Computing Explained*, Vol. 2011
- [5] Simba, F, 2010, *Cloud Computing:Strategies for Cloud Computing Adoption*, Dublin Institute of Technology.
- [6] P. Mell, T. Grance, 2011, "The NIST Definition of Cloud Computing", NIST Special Publication 800-15.
- [7] Rivera, Ismael, dkk. *Collaborative OpenSocial Network Dataset based Email Ranking and Filtering*. 12th International Conference Information Visualization-XIII'08, Juli 2008

## Biodata Penulis

**Rico Agung Firmansyah**, memperoleh gelar Sarjana Komputer (S.Kom), Program Studi Teknik Informatika STMIK Amikom Yogyakarta, lulus tahun 2006. Tahun 2010 melanjutkan studi di Magister teknik Informatika STMIK Amikom Yogyakarta. Saat ini aktif mengajar sebagai Staf Pengajar program Teknik Informatika STMIK AMIKOM Yogyakarta dan sebagai Kepala Program Keahlian Teknik Komputer dan Jaringan di SMK Syubbanul Wathon Magelang. Aktif melakukan penelitian internal maupun eksternal (seminar nasional).