

PENERAPAN E-LEARNING OLAT DENGAN WEBSERVER TER-CLUSTER UNTUK PENINGKATAN KAPASITAS AKSES E-LEARNING

Acep Taryana¹, Hari Siswanto²

Program Studi Teknik Elektro, Unsoed

Jl. Kampus No 1 Grendeng Purwokerto

E-mail: acep@unsoed.ac.id¹, hari.siswanto@unsoed.ac.id²

ABSTRAKS

Telah dirancang teknik clustering dan load balancing untuk meningkatkan kapasitas akses sistem e-learning. Struktur cluster webserver 1 node, 2 node dan 3 node di susun untuk meringankan beban akses dari pengguna melalui load balancer Haproxy. Pengujian dilakukan dengan menggunakan aplikasi JMeter yang terhubung ke load balancer Haproxy. Haproxy akan mengatur beban akses server-server yang telah didefinisikan dalam clustering webserver. Untuk mengetahui kapasitas e-learning OLAT, pengujian dilakukan pula terhadap sistem e-learning lain yaitu Moodle.

Hasil penelitian menunjukkan, beban pemrosesan request untuk OLAT lebih handal dari pada Moodle. OLAT mampu menangani request tanpa error sampai dengan 111 ribu akses untuk struktur clustering webserver 3 node, sedangkan Moodle hanya mampu menangani sampai dengan 300 request. Hal ini membuktikan bahwa OLAT lebih unggul dari Moodle dalam hal kestabilan akses, dan penerapan teknik clustering webserver yang dipadukan dengan load balancing dapat meningkatkan kapasitas akses sistem e-learning OLAT.

Kata Kunci : *response time, testing, webserver clustering, load balancing.*

1. PENDAHULUAN

1.1 Latar Belakang Masalah

Era digital yang diikuti dengan pesatnya perkembangan di bidang *Information and Communication Technology* (ICT) telah mempengaruhi hampir semua bidang, dan tentunya bidang pendidikan. Dengan implementasi e-learning ini, kita harapkan dapat:

- Memberikan prioritas bagi pengembangan pendidikan tinggi regional dengan membangun lokal points.
- Meningkatkan akses dan pemerataan pendidikan.
- Memperbaiki dan menyebarkan relevansi dan kualitas pendidikan.
- Meningkatkan kegiatan riset, baik secara kuantitas maupun kualitas.

Teknologi e-learning telah banyak diimplementasikan, namun efektifitas dan efisiensi masih dipertanyakan. Apakah e-learning telah betul-betul menggantikan pembelajaran tatap muka di kelas atau masih sekadar pendukung. Selain mengerti pentingnya penerapan e-learning, perlu diperhatikan pemilihan sistem e-learning yang tepat. Pemilihan sistem e-learning yang tepat dapat meningkatkan skalabilitas, kecepatan akses, dan keamanan.

Sistem e-learning secara fisik akan ditempatkan dalam webserver Apache Tomcat yang terhubung ke akses ribuan pengguna secara simultan. Berbagai cara dapat dilakukan untuk melakukan optimasi sistem e-learning yaitu replikasi webserver menjadi lebih dari satu yang disebut dengan teknik clustering, webserver lebih dari satu dikombinasikan dengan teknik pengaturan

beban akses pengguna yang disebut dengan *load balancing*.

Optimasi sistem e-learning akan diterapkan terhadap sistem e-learning OLAT, dan akan dipilih perbandingan sistem e-learning lain yang banyak digunakan di Indonesia yaitu Moodle.

1.2 Perumusan Masalah

Berdasarkan uraian di atas, permasalahan-permasalahan yang akan diteliti perlu dirumuskan sebagai berikut :

1. bagaimana akses pengguna dapat dilayani oleh banyak *server* sehingga kejenuhan sistem dapat diukur.
2. bagaimana menyusun *clustering* dan *load balancing* untuk sistem e-learning OLAT maupun Moodle sehingga mampu mendistribusikan akses pengguna ke berbagai server yang belum bekerja optimal.
3. bagaimana struktur server-server sistem e-learning yang digunakan untuk penanganan pendistribusian beban akses dari pengguna.
4. bagaimana menyesuaikan web server Apache Tomcat dengan sistem e-learning terhadap perubahan pengaturan Sistem e-learning untuk OLAT maupun Moodle.

1.3 Tujuan dan Manfaat

Tujuan yang akan dicapai dalam penelitian ini adalah :

1. Mengkaji teknologi Clustering dan Load balancing OLAT versi 6.2.2
2. Menganalisis kinerja teknologi OLAT versi 6.2.2 menggunakan teknologi clustering dan load balancing dibandingkan dengan Moodle.

Kegunaan yang dapat diperoleh atau dicapai dari penelitian ini adalah :

1. Memudahkan organisasi untuk mengelola konten pembelajaran.
2. Memberikan kecepatan, kestabilan, ketersediaan akses e-learning kepada pengguna.
3. Menjadikan sumber referensi untuk pemilihan sistem e-learning sesuai dengan kebutuhan.

1.4 Landasan Teori

1.4.1 Sistem e-learning

Produk sistem e-learning telah banyak dikembangkan oleh berbagai institusi internasional, sebagai contoh OLAT dan Moodle (anonim, OLAT, 2007). Kedua produk memiliki karakteristik sebagai berikut :

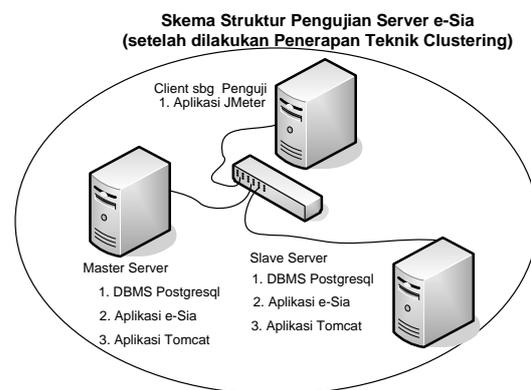
- Moodle
 1. Arsitektur klasik PHP/MySQL
 2. Berkembang secara terorganisir
 3. Data harus selalu dibaca dari basis data / file system
 4. Kesulitan dalam pengembangan jangka panjang
 5. Teroptimisasi untuk pemasangan dalam skala kecil
- OLAT
 1. Versi pertama masih menggunakan PHP
 2. OLAT 3 di desain ulang untuk mendukung berbagai macam konsep pembelajaran dan implementasi skala besar.
 3. Arsitektur dioptimalisasikan untuk pengembangan berskala, terintegrasi, dan memungkinkan perluasan sistem.
 4. Java/J2EE, menggunakan pengembangan framework MVC.
 5. Pengembangan jangka panjang lebih mudah karena arsitekturnya terbagi dalam beberapa modul.
 6. Dioptimisasikan untuk pemasangan e-learning berskala besar.

Memperhatikan pada karakteristik kedua produk e-learning di atas, maka akan terlihat keunggulan atau mana yang paling unggul untuk dapat dikembangkan dalam organisasi yang bersangkutan. Secara umum OLAT lebih unggul dilihat dari sudut pandang pemanfaatan dan pemeliharaan e-learning jangka panjang untuk dapat dimodifikasi, diintegrasikan dengan sistem yang ada dalam organisasi. OLAT dikembangkan dengan teknologi Java dimana pengembangannya menggunakan paradigma berorientasi objek yang memiliki keleluasaan dan kemudahan dalam rekayasa ulang (*re-engineering*).

1.4.2 Clustering Basis Data

Secara umum clustering dirancang untuk mengelompokkan sumber daya (server-server) sehingga memiliki satu kesatuan komputasi.

Clustering basis data merupakan kumpulan server yang dikonfigurasi oleh suatu perangkat lunak DBMS sehingga menjadi satu kesatuan sistem untuk menangani manajemen basis data (Anonim, Postgresql, 2008). Menurut Vishal Batra (2008), salah satu manfaat dari clustering basis data adalah terjaganya aspek “Availability”, artinya akses dari client pada jumlah tertentu secara simultan akan dijamin dapat dilayani oleh server-server dalam lingkungan cluster. Keterjaminan layanan akan menyebabkan kelancaran akses client. Salah satu contoh konfigurasi clustering basis data ditunjukkan dalam Gambar 1. (Taryana A, Cahyono T, Riyadi, 2008).



Gambar 1. Clustering Basis Data

1.4.3 Load balancing

Load balancing atau penyeimbangan beban dalam jaringan sangat penting bila skala dalam jaringan komputer makin besar demikian juga *traffic* data yang ada dalam jaringan komputer makin lama makin tinggi. Layanan *Load Balancing* dimungkinkan pengaksesan sumber daya dalam jaringan didistribusikan ke beberapa *host* lainnya agar tidak terpusat sehingga unjuk kerja jaringan komputer secara keseluruhan bisa stabil (Iwan Rijayana, 2005).

Ketika sebuah server sedang diakses oleh para pengguna, maka sebenarnya server tersebut sebenarnya sedang terbebani karena harus melakukan proses permintaan kepada para penggunanya. Jika penggunanya banyak maka prosesnyapun banyak.

Session-session komunikasi dibuka oleh server tersebut untuk memungkinkan para pengguna menerima servis dari server tersebut. Jika satu server saja terbebani, tentu server tersebut tidak bisa banyak melayani para penggunanya karena kemampuan melakukan processing ada batasnya. Solusi yang paling ideal adalah dengan membagi-bagi beban yang datang ke beberapa server. Jadi yang melayani pengguna tidak hanya terpusat pada satu perangkat saja. Teknik ini disebut Teknik *Load Balancing*.

Adapun manfaat dari *Load Balancing*:

- a). Menjamin *Reliabilitas* layanan, berarti kepercayaan terhadap sebuah sistem untuk dapat terus melayani pengguna dengan sebaik-baiknya.
- b). Skalabilitas dan ketersediaan, jika dalam sebuah jaringan komputer hanya terdapat satu buah server maka kondisi ini menunjukkan terdapat satu titik masalah. Seandainya tiba-tiba server itu mati maka layanan terhadap pengguna akan terganggu. Dengan melakukan penambahan server dan membentuk server farm maka skalabilitas akan meningkat dan selain itu faktor ketersediaan juga akan meningkat.

1.4.4 Metode Load Balancing

Metode load balancing untuk menjamin skalabilitas, ketersediaan akses terdiri dari Load Balancing dengan hardware/switch, Load Balancing dengan Software, Load Balancing dengan perangkat perpaduan Hardware dan Software.

1.4.5 Throughput

Throughput adalah Jumlah kerja yang dapat diselesaikan dalam satu unit waktu. Cara untuk mengekspresikan *throughput* adalah dengan jumlah job pemakai yang dapat dieksekusi dalam satu unit/interval waktu. Dalam sasaran penjadwalan proses, *throughput* ini adalah memaksimalkan jumlah job yang diproses per satu interval waktu. Lebih tinggi angka *throughput*, lebih banyak kerja yang dilakukan system.

1.4.6 Alat Pengukur Kinerja Sistem e-learning

JMeter merupakan sub proyek dari sebuah proyek besar yang disebut Jakarta Meter (Anonim, The Apache Jakarta, 2008), sebagai sebuah proyek besar dari Apache Software Foundation (ASF) yang bertujuan untuk menyediakan alat-alat yang dapat membantu para pengembang, atau siapapun berkolaborasi dan dapat memaksimalkan pengembangan perangkat lunak. JMeter merupakan 100% aplikasi desktop yang berjalan di atas Java VM yang dapat mengukur kinerja sistem dan *load-test* dari aplikasi client/server. Yang dapat diukur antara lain kinerja HTTP (aplikasi web), FTP, JDBC, bahkan EJB, SOAP atau CORBA. Berdasarkan hal tersebut maka aplikasi JMeter akan digunakan untuk mensimulasikan akses pengguna, mengukur *throughput* dan *error request* dari pengguna.

2. METODE PENELITIAN

2.1 Tahap Persiapan

Pada awal penelitian akan dilakukan studi pustaka/literatur tentang Clustering, Load Balancing baik untuk basis data maupun untuk Web Server Tomcat, berupa buku, jurnal. DBMS MySql akan digunakan untuk menguji kinerja sistem e-learning OLAT maupun Moodle. Faktor kriteria perangkat lunak bebas, lisensi, *proprietary* turut menentukan

pemilihan. Perangkat lunak untuk mengukur kinerja sistem dipilih JMeter.

2.2 Perancangan Penelitian

Tujuan penelitian telah ditetapkan, untuk mencapai tujuan tersebut perlu ditetapkan rancangan penelitian yang meliputi rancangan lingkungan uji, rancangan alat, rancangan bahan, rancangan waktu, rancangan sampel uji, rancangan sistem clustering dan load balancing, instalasi sistem operasi, instalasi dan konfigurasi DBMS, instalasi aplikasi server, instalasi sistem e-learning, pengujian sistem e-learning dalam lingkungan *clustering* dan *load balancing*, rancangan dan analisis data.

Waktu yang direncanakan untuk dapat menyelesaikan penelitian ini adalah 2(dua) bulan dengan rincian sebagai berikut : 1) Bulan pertama untuk tahap persiapan dan perancangan penelitian, 2) Bulan kedua implementasi dan pengujian penelitian. Lokasi penelitian adalah laboratorium Komputer Jurusan Teknik Unsoed.

2.2.1 Perancangan Lingkungan Uji

Lingkungan Uji harus didefinisikan terlebih dahulu. Lingkungan uji merupakan laboratorium mini di Lab Komputer Jurusan Teknik yang terdiri dari :

- 3 PC sebagai web server dengan @ prosesor 2 GHz dan memory 1 Gb
- 1 PC sebagai load balancer dengan prosesor 2 Ghz dan memory 512 Mb
- 1 PC sebagai database server dengan prosesor 2 GHz dan memory 512 Mb
- 1 PC sebagai Jmeter test untuk melakukan loading test dengan Jmeter sesuai jumlah request yang direncanakan.
- Peralatan jaringan (Switch Hub, Cable UTP, dll).

Lingkungan uji dibutuhkan untuk menyusun skema/arsitektur server yang dijadikan sebagai infrastruktur untuk arsitektur pengujian dengan clustering dan load balancing. Tiga buah komputer di laboratorium mini tersebut akan difungsikan sebagai client, server yang terdiri dari balancer, web server, application server, database server sesuai skenario uji.

2.2.2 Instalasi Sistem Operasi, DBMS, Web Server Tomcat, OLAT dan Moddle

Sistem Operasi yang digunakan untuk instalasi lingkungan pengujian penelitian adalah Linux untuk komputer server dan client. Cara dan model instalasi sama seperti halnya dalam menyusun jaringan komputer secara umum.

Setelah Sistem Operasi terpasang, maka langkah selanjutnya adalah memasang DBMS dan mengkonfigurasi server untuk menjadi web server, mengkonfigurasi client sehingga dapat mendukung konsep clustering dan load balancing. Perangkat lunak yang digunakan untuk konfigurasi webserver adalah Apache Tomcat.

2.2.3 Perancangan Sistem Clustering dan Load Balancing

Sistem *Clustering* yang digunakan dalam penelitian adalah clustering olat dengan *load balancer* Haproxy. Hirarki cluster terdiri dari lapisan gateway sebagai perantara client dengan web server, lapisan webserver sebagai lapisan untuk menuju sistem OLAT. Lapisan balancer Haproxy berfungsi untuk mengatur akses client pada webserver sehingga setiap webserver akan memproses akses secara bergantian sesuai beban.

2.2.4 Perancangan sampel uji

Pada pelaksanaan pengujian, user diciptakan oleh aplikasi JMeter. Aplikasi JMeter dapat menciptakan ribuan pengguna dalam ukuran detik. Sampel uji untuk pengujian, ditentukan oleh jalannya aplikasi JMeter. Jumlah sampel uji akan dinaikkan terus menerus sampai dengan sistem JMeter jenuh baik untuk pengujian sistem OLAT maupun Moodle.

2.2.5 Pengujian Sistem e-learning OLAT dan Moodle

Pengujian dilakukan terhadap 2 sistem e-learning yaitu sistem OLAT dan Moodle. Skenario pengujian aplikasi tidak berbeda antara OLAT dengan Moodle. Yang masih menjadi pertanyaan sampai rancangan penelitian ini disusun, apakah aplikasi JMeter berfungsi baik untuk seluruh sampel yang diterapkan pada OLAT maupun Moodle ?

2.2.6 Rancangan Analisis Data

Pengujian menggunakan Jmeter akan memunculkan nilai *error*, yang artinya *request* yang ditangani oleh sistem telah *overload*. Pengujian terhadap sistem e-learning Moodle maupun OLAT akan dilakukan. Berdasarkan Tabel 1, didapatkan *throughput*, *error* untuk 1,2,3 node.

Tabel 1. Borang Pengujian Sistem e-learning

	<i>Throughput</i> (dalam detik), <i>Error</i>		
e-learning	1 Node	2 Node	3 Node
Moodle	?	?	?
OLAT	?	?	?

Nilai *error* yang dihasilkan oleh Jmeter menunjukkan beban server, nilai *error* lebih besar dari nol berarti server mulai terbebani untuk memproses permintaan dari client. Semakin kecil nilai *error* maka server tidak terbebani. Terjadinya

nilai *error* berhubungan dengan jumlah user yang akses untuk masing-masing sistem e-learning.

Berdasarkan pada hipotesa penelitian ini, sistem e-learning OLAT lebih unggul dari Moodle dalam hal penanganan request dari user. Apabila penanganan request dari user dapat ditunjukkan dalam pengujian JMeter maka dapat dinyatakan hipotesis benar.

2.2.7 Prosedur Penelitian

Peneliti menyiapkan lingkungan pengujian, harus dipastikan bahwa jaringan telah terbentuk dan seluruh client sudah terkoneksi dengan server. Sistem konfigurasi server sudah ditetapkan sehingga client bisa mengakses melalui alamat IP yang ditentukan.

Peneliti melakukan instalasi dan konfigurasi sistem clustering dan load balancing menggunakan parameter uji yang telah ditetapkan. Pada tahap ini peneliti mencoba untuk menyusun berbagai skema struktur lingkungan pengujian untuk keperluan implementasi teknik *clustering* dan *load balancing*. Konfigurasi sistem e-learning OLAT dan Moodle dilakukan pada tahap ini.

Peneliti melaksanakan pengujian pertama, kedua, ketiga dalam sistem e-learning yang berbeda. Pengujian pertama dilaksanakan untuk menunjukkan kinerja tanpa clustering untuk setiap sistem e-learning. Pengujian kedua digunakan untuk menunjukkan kinerja sistem e-learning untuk 2 node, dan pengujian ketiga untuk menunjukkan sistem e-learning dengan 3 node. Hal yang perlu diperhatikan dalam pelaksanaan pengujian adalah konfigurasi jaringan, konfigurasi Apache Tomcat, jumlah pengguna memiliki satu keseragaman, sehingga mendapatkan hasil penelitian yang signifikan.

Kesimpulan akan ditentukan sesuai aturan berikut : “jika nilai *error* tercapai maka sistem sudah jenuh dan *throughput* akan menurun”. Perlu diperhatikan dan dicatat pula kondisi maupun peralatan yang digunakan untuk setiap pengujian. Kondisi dan peralatan yang digunakan akan menjadi bahan rekomendasi untuk pengembangan dan implementasi sistem e-learning di Jurusan Teknik.

3. HASIL DAN PEMBAHASAN

3.1 Arsitektur Lingkungan Pengujian

Lingkungan uji untuk mengukur kejenuhan sistem e-learning perlu disiapkan sesuai Gambar 3. Lingkungan uji memerlukan komputer yang difungsikan sebagai server dan sebagai client untuk menguji kejenuhan. Pada server akan dipasang DBMS MySQL, Aplikasi Tomcat sebagai webserver, aplikasi e-learning. Sedangkan pada client akan dipasang aplikasi Jmeter untuk mengukur tingkat kejenuhan. Pada lingkungan uji tanpa clustering, keberadaan webserver hanya 1 buah, sedangkan dengan clustering keberadaan webserver ada lebih

dari 1 buah (3 buah). Ada 1 buah server yang difungsikan sebagai load balancer webserver yang disebut dengan Haproxy.

3.1.1 Olat 6.2.2.

Aplikasi Olat 6.2.2 dijalankan di Ubuntu 9.04 di tiga PC web server. Instalasi Olat 6.2.2 membutuhkan aplikasi pendukung antara lain :

- Sun JDK 1.6 sebagai development kit java.
- Apache2 web server .
- Tomcat 6
- Apache Ant -1.7
- Mysql server

Tahapan instalasi Olat 6.2.2

- Instal JDK 1.6 dengan apt-get install
- Install tomcat 6 dengan apt-get install
- Install apache2 web server dengan apt-get install
- Install libapache2-mod-jk untuk menginstal mod-jk konektor apache2 dengan tomcat.
- Mengsetting enable modul jk.load di apache2.
- Mengarahkan java_home dan tomcat home yang ada di wokerproperties libapache2modjk,.
- Membuat db olat di pc database server dengan Mysql server.
- Membuat file directory di /usr/local/opt/olat3 di satu PC web server
- Mengekstrak file source code 6.2.2 ke directory olat3.
- Mencopy file build.properties.default olat3 ke file build.properties.
- Mengubah buil.properties dengan db host diarahkan ke ip PC database server.
- Kemudian lakukan langkah konfigurasi olat dengan ant config-all.
- Membuilding olat dengan ant build.
- Membuilding data base dengan perintah ant dbsetup
- Mendeploy olat dengan perintah ant deploy.
- Di dua web server PC lainnya dibuat file directory /usr/local/opt/olat3, setelah itu file /usr/local/opt/olat3 di webserver yang telah dibuild di mount ke file /usr/local/opt/olat3 lainnya.
- Mengkonfigurasi virtual host di apache2 supaya bias mengarah ke ip balancer PC (yang terdapat Haproxy)
- Menonaktifkan security tomcat 6.
- Mengarahkan directory olat3 di semua pc webserver agar bisa menjadi hak akses user tomcat 6.
- Merestart aplikasi apache2 dan tomcat6 pada masing-masing web server.

3.1.2 Haproxy

Haproxy adalah produk opensource yg mendukung keperluan *loadbalancer* dan *failover webserver*, banyak digunakan untuk keperluan reverse proxy di site2 yang trafik hariannya tinggi.

Setting Haproxy untuk cluster olat adalah sebagai berikut :

- Menginstall Haproxy dengan apt-get haproxy
- Mengubah haproxy.cfg untuk mengkonfigurasi ip webserver olat yang dituju.
- Restart haproxy.

3.1.3 JMeter

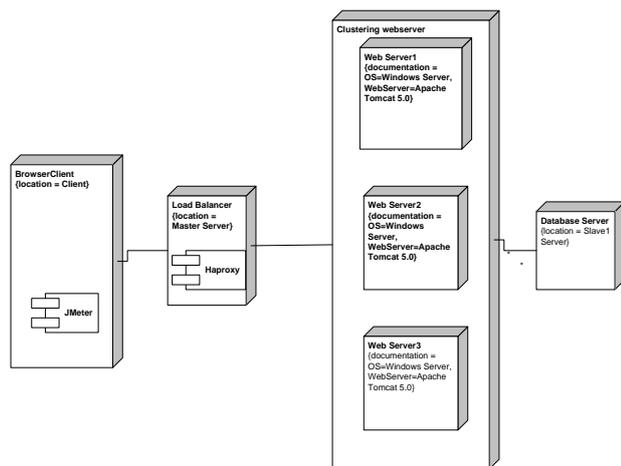
Apache JMeter merupakan *software opensource* yang mempunyai berbagai fitur test web server. Untuk uji beban, JMeter sangat handal karena mempunyai test ke *Get* untuk halaman statis dan test *Post* untuk halaman dinamisnya. Konfigurasi JMeter untuk uji cluster olat adalah sebagai berikut :

- Menginstal JMeter dengan apt-get install jmeter
- Setting banyaknya thread (user)
- Setting Loop (pengulangan)
- Arahkan ke server yang dituju, dalam hal ini ip proxy balancer yang terdapat haproxy.
- Setting http request ke path yang dituju (olat dmz) dan port 80 untuk olat di apache2
- Setting Get type test untuk halaman login dan halaman logout karena merupakan halaman statis
- Setting Post type test untuk login ke halaman dinamis olat.

3.1.4 Manifestasi Fisik Perangkat Lunak saat Pengujian

Perangkat lunak yang digunakan dalam skenario pengujian adalah JMeter. JMeter dipasang di komputer *client* yang melakukan koneksi dengan Haproxy untuk melakukan pengujian akses ke webserver. Didalam node *Web server* terdapat *WebServer Apache Tomcat 6.0* yang memiliki konfigurasi penunjuk akses ke *database server* yang sesuai Gambar 2 .

Manifestasi fisik pengujian dalam lingkungan clustering seperti Gambar 2, memperlihatkan bahwa pengaturan beban akan diatur oleh Haproxy. Haproxy akan mengatur request ke masing-masing webserver sesuai algoritma yang digunakan.



Gambar 2. Diagram Deployment Sistem Pengujian e-learning

3.2 Analisis Data Pengujian untuk Sistem e-learning OLAT

Berdasarkan kajian arsitektur perangkat lunak, sudah pasti dijamin bahwa penerapan Teknik Clustering webservice pada akses sistem e-learning dapat membantu server untuk mengurangi beban pemrosesan sehingga dapat meningkatkan kapasitas pemrosesan request dari user.

Tabel 2 menunjukkan bahwa sistem e-learning OLAT lebih baik dari Moodle, OLAT mampu menangani sejumlah *request* dari pengguna dengan angka lebih besar dari Moodle untuk setiap pengujian. Jumlah request yang ditangani oleh OLAT lebih besar 16 kali dari Moodle pada pengujian 1 node, lebih besar 154 kali pada pengujian 2 node, lebih besar 336 kali pada pengujian 3 node.

Ditinjau dari pengaruh penerapan *clustering* dan *load balancing* terhadap kinerja sistem e-learning itu sendiri, Moodle tidak menunjukkan dampak yang signifikan. Perubahan hanya terjadi pada pengujian 1 node ke 2 node, sedangkan dari 2 node ke 3 node tetap sama. Sebaliknya dengan OLAT, mampu meningkatkan penanganan request cukup drastis.

Tabel 2. Hasil Uji JMeter untuk OLAT dan Moodle

e-learning	Throughput (T) (dalam detik), Error(E)		
	1 Node	2 Node	3 Node
Moodle	T: 7.63 E: Request ke 210	T: 11.45 E: Request ke 330	T: 16.64 E: Request ke 330
OLAT	T: 68,66 E: Request ke 3.500	T: 27.59 E: Request ke 51.000	T: 55.16 E: Request ke lebih dari 111.000

4. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian ini adalah

1. Penelitian menunjukkan bahwa akses pengguna dapat dilayani baik dengan membuat struktur web server terdiri dari 3 node yang terhubung ke sistem OLAT dan Haproxy. Semakin banyak web server maka *error* akan terjadi untuk jumlah request yang lebih besar.
2. Pengujian dengan 3 node menunjukkan bahwa sistem e-learning OLAT mengalami kejenuhan pada request > 111 ribu, sedangkan sistem e-learning Moodle jenuh pada request > 300 user.
3. Sistem Olat 6.2.2. bila di cluster maka akan meningkatkan performasinya sampai 4 kali lipat bila 2 node dan bisa lebih dari 8 kali, bila dengan 3 node.
4. Sistem *Cluster* dan *load balancing* berdampak signifikan pada peningkatan kemampuan penanganan request pengguna dalam sistem Olat 6.2.2.
5. Sistem cluster sangat relevan untuk meningkatkan performa olat 6.2.2
6. Sistem Cluster Moodle bisa mempercepat pengaksesan terhadap request tetapi tidak terlalu signifikan dalam menangani *load request* karena tidak terjadi penambahan yang berarti pada kapasitas maksimum request yang bisa ditangani.

Saran dari penelitian ini adalah

1. Penelitian ini belum menjelaskan ukuran kecepatan akses atau response time dari masing-masing sistem e-learning OLAT dan Moodle. Oleh karena itu perlu ada penelitian yang membahas perbandingan kecepatan akses untuk berbagai konfigurasi. Penelitian lanjutan penting dilaksanakan karena tidak seluruh bidang organisasi memiliki jumlah pengguna yang besar (skala enterprise).
2. standarisasi format konten SCORM.

PUSTAKA

- Anonim. 2008. PostgreSQL 8.2.9 Documentation.URL : <http://www.postgresql.org/docs/8.2> Diakses tgl 15 Juli 2008.
- Anonim. 2008. The Apache Jakarta Project-JMeter. <http://jakarta.apache.org/jmeter/usermanual/>. Diakses tgl 22 Juli 2008.
- Anonim. 2008. Slony-I.URL : <http://slony.info>. Diakses tgl Oktober 2008.
- Anonim. 2007. OLAT-Online Learning And Training : <http://www.frentix.com/de/resources/docu/OLAT-MODDLE-en.pdf>. diakses 5 Juli 2007.
- Fatansyah. 2002. Buku Teks Ilmu Komputer-Basis Data. Cetakan keempat. Informatika Bandung.
- Iwan Rijayana. 2005. Teknologi Load Balancing untuk Mengatasi Beban Server. Prosiding Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005). ISBN: 979-756-061-6
- Love Robert. 2007. Linux System Programming: Talking Directly to the Kernel and C Library. O'Reilly. ISBN 0596009585, 9780596009588.
- Laurie Ben, Laurie Peter. 2002. Apache: The Definitive Guide.O'Reilly.
- Partio Miko. 2007. Final Project-Evaluation Of Postgresql Replication And Load Balancing Implementations. Stadia Helsinki Poliytechnic.
- Vishal Batra, Wen-Syan Li , Sumit Negi. 2008. *J2EE Architecture for Database Cluster-Based High Volume E-Commerce Web Applications*.IEEE Computer Society. pp 697-704.
- Taryana A, Cahyono T, Riyadi. 2008. Laporan Penelitian : Penerapan Teknik Clustering Basis Data dalam Implementasi Sistem Informasi untuk Memperkecil Response Time (Studi Kasus : Sistem Informasi Akademik Unsoed).