

MODEL AKSES DATABASE TERDISTRIBUSI BERDASARKAN PENDEKATAN SERVICES ORIENTED ARCHITECTURE

Rizky Tahara Shita
Universitas Budi Luhur
rizky.tahara@gmail.com

ABSTRAK

Bagi perusahaan – perusahaan yang sudah cukup mapan, mereka akan mengembangkan sayapnya dengan membuka perusahaan cabang yang tersebar diberbagai lokasi. Merupakan hal yang penting bagi setiap perusahaan yang memiliki cabang – cabang yang tersebar diberbagai daerah untuk dapat melakukan integrasi data, yang terkadang penyimpanan data dilakukan pada masing – masing cabang tanpa terhubung dengan kantor pusat.

Tidak hanya pada perusahaan, bagi pihak universitas yang sudah mapan juga mempunyai cabang yang tersebar. Seperti pada Universitas Budi Luhur yang memiliki cabang di berbagai lokasi; yang sampai saat ini masih terdapat kekurangan pada sisi integrasi data antara kampus cabang dengan kampus pusat, karena masih menggunakan konsep database terdesentralisasi.

Dengan adanya kekurangan dan keterbatasan dalam integrasi data antara kampus cabang dengan kampus pusat, maka pada penelitian ini akan dikaji pembuatan model akses database terdistribusi berdasarkan pendekatan Service Oriented Architecture yang berbasis web services pada Universitas Budi Luhur. Hal ini dilakukan agar data yang ada pada tiap cabang Universitas Budi Luhur dapat saling terintegrasi dan saling tersinkronisasi dengan kampus pusat serta sebaliknya; yang tentunya akan berdampak positif pada sisi mahasiswa dan akademik untuk tiap kampus cabang dan kampus pusat dalam melakukan kegiatan akademis.

Kata kunci: database terdistribusi, service oriented architecture, web services

Pendahuluan Latar Belakang

Semakin berkembangnya suatu usaha pada sebuah perusahaan, maka akan berdampak pada pengembangan sayap agar perusahaan semakin maju dan mendapatkan *income* yang lebih baik. Pengembangan sayap ini biasanya dilakukan dengan membuka cabang baru pada lokasi yang berbeda dari lokasi pusat.

Dengan semakin banyaknya cabang yang ada dan sistem administrasi yang digunakan juga dapat berbeda – beda, maka koordinasi data antara cabang dengan pusat dapat menjadi suatu kendala baru yang seharusnya dapat dilakukan integrasi dan sinkronisasi. Karena itu perlu dilakukan sebuah cara tersebut hanya terfokus pada mahasiswa yang berada pada kampus pusat saja, hal ini

untuk dapat melakukan integrasi data dari data – data yang tersebar tersebut.

Data yang tersebar tidak saja terjadi pada perusahaan – perusahaan yang sudah mapan. Pada sisi universitas juga mempunyai kendala yang hampir serupa, seperti halnya Universitas Budi Luhur yang memiliki cabang diberbagai lokasi; antara lain: Roxy, Salemba, Cempaka; sedangkan kampus pusat berada di Jakarta Selatan.

Bagi para mahasiswa Universitas Budi Luhur, saat ini sudah dimudahkan dengan adanya fasilitas untuk melihat nilai secara *online* dan berbagai administrasi *online* lainnya melalui *website* yang disediakan khusus untuk mahasiswa dengan klik pada <http://student.bl.ac.id>. Akan tetapi, fasilitas

terjadi karena data yang ada pada tiap cabang tidak dapat langsung terintegrasi dan

tersinkronisasi dengan data yang ada pada kampus pusat. Sedangkan data yang digunakan pada <http://student.bl.ac.id> hanya mengambil data dari *database* yang ada pada kampus pusat. Dan tentunya berdampak pada kesulitan mahasiswa yang datanya terdaftar pada kampus cabang untuk mendapatkan fasilitas yang terdapat pada <http://student.bl.ac.id> tersebut.

Pada sisi akademik; kendala juga terjadi pada kontrol terhadap data yang tersebar di masing – masing cabang, sehingga proses penyampaian informasi kepada mahasiswa menjadi terhambat. Kontrol data yang diperlukan tidak hanya terfokus pada data untuk keperluan mahasiswa saja, data untuk keperluan dosen dan karyawan juga menjadi kurang terkontrol dan dapat berdampak bagi ketua Yayasan Universitas Budi Luhur dalam mengetahui jumlah pegawai dan dosen yang ada pada universitas secara keseluruhan.

Untuk mengatasi masalah tersebut, maka pihak kampus pusat berusaha mencari solusi dengan melakukan penelitian untuk memecahkan masalah diatas. Dari penelitian yang dilakukan, didapatkan bahwa salah satu solusi dari pemecahannya adalah dengan membangun sebuah sistem yang dapat memberikan layanan data kepada kampus pusat dan kampus cabang yang ada maupun sebaliknya. Layanan ini diangkat menjadi sebuah layanan berbasis web (*web services*) agar terdapat arsitektur standar yang dapat mengatur pola distribusi informasi data kepada dan dari tiap kampus cabang dengan kampus pusat.

Masalah Penelitian

Identifikasi Masalah

Permasalahan yang ada pada Universitas Budi Luhur adalah belum adanya arsitektur standar yang mengatur pola distribusi informasi data dari cabang kepada kampus pusat. Sinkronisasi yang dilakukan masih menggunakan cara manual (*transfer* data menggunakan media *flashdisk* atau *compact disc*) yang diantar dari kampus cabang kepada kampus pusat juga menjadi faktor permasalahan yang dihadapi dalam penyampaian data tersebut. Dengan kata lain, bahwa pada tiap kampus masih

menerapkan konsep *database* terdesentralisasi.

Pembatasan Masalah

Mengingat luasnya bahasan tentang *Service Oriented Architecture*, maka tesis ini hanya akan membahas tentang model *prototype* untuk mengakses data mahasiswa dan data nilai mahasiswa berdasarkan teknologi *web services*.

Studi kasus yang diambil adalah pada Universitas Budi Luhur kampus Pusat dan kampus cabang Roxy dengan menggunakan DBMS MySQL, WSO2/WSF PHP *Framework*, CodeIgniter PHP *Framework* dan melakukan integrasi *web services* pada biskitZ CMS untuk keperluan kampus Pusat.

Rumusan Masalah

Rumusan masalah dalam penelitian berdasarkan latar belakang yang ada, dapat dinyatakan sebagai berikut:

- a) Bagaimana bentuk skema arsitektur model akses *database* terdistribusi yang sesuai dengan kebutuhan.
- b) Bagaimana *prototype* model akses *database* terdistribusi untuk dapat melakukan integrasi data tanpa mengubah *legacy system* (sistem yang sudah ada).

Tujuan dan Manfaat Penelitian

Tujuan Penelitian

Tujuan penelitian ini dibagi menjadi 2, yaitu:

- a) Tujuan Umum

Tujuan umum dari penelitian ini adalah agar didapatkan sebuah model akses data pada *database* terdistribusi dengan menerapkan konsep metodologi *Service Oriented Architecture* (SOA) yang berbasis *web services*.

Dimana dengan adanya model akses data ini, maka dapat dilanjutkan untuk pengembangan pada bagian data *warehouse* terdistribusi (*distributed data warehouse*) dan data *mining* terdistribusi (*distributed data mining*). Bagi perusahaan *enterprise*, kebutuhan data *warehouse* dan data *mining* sangat dibutuhkan, apalagi jika data yang diambil pada proses data *warehouse* dan data *mining*, merupakan data yang tersebar dari beberapa data pada lokasi

yang berbeda; karena itu model akses *database* terdistribusi sangat diperlukan dalam pembuatan data *warehouse* dan data *mining* untuk data – data yang tersebar tersebut.

b) Tujuan Khusus

Secara khusus, tujuan penelitian yang dilakukan pada Universitas Budi Luhur adalah agar data mahasiswa dan data nilai matakuliah mahasiswa dapat terintegrasi dari tiap cabang dengan kampus pusat. Sehingga kampus pusat dapat secara *real time* memantau dan memberikan pelayanan akademik yang lebih baik melalui media *website* yang telah disediakan khusus untuk mahasiswa. Hal ini akan mengubah konsep *database* terdesentralisasi yang sekarang masih diterapkan pada Universitas Budi Luhur dengan cabangnya untuk mengarah pada konsep *database* terdistribusi dengan menerapkan metodologi *Service Oriented Architecture*.

Manfaat Penelitian

Manfaat penelitian ini juga dapat dilihat dari sisi umum dan sisi khusus:

a) Manfaat Umum

Secara umum, manfaat penelitian ini akan dapat membantu dalam memberikan model akses *database* terdistribusi berdasarkan metodologi *Service Oriented Architecture* yang berbasis *web services*.

Bagi perusahaan yang sudah mapan dan memiliki beberapa cabang pada lokasi yang berbeda dan membutuhkan integrasi data antar tiap cabang, metodologi *service oriented architecture* berbasis *web services* akan membantu melakukan integrasi data tersebut yang mana pada kelanjutannya dapat memudahkan dalam pembuatan aplikasi data *warehouse* terdistribusi (*distributed data warehouse*) maupun data *mining* terdistribusi (*distributed data mining*).

b) Manfaat Khusus

Manfaat penelitian yang secara khusus, dapat dilihat dari 2 sisi; yaitu:

i. Dari sisi mahasiswa

Dari sisi mahasiswa, manfaat yang akan dapat dirasakan antara lain:

(a) Data nilai matakuliah yang diikuti dapat dilihat dengan lebih cepat melalui *website* khusus untuk mahasiswa.

(b) Efisiensi biaya, karena mahasiswa tidak perlu datang ke kampus (baik pusat maupun cabang) untuk dapat melihat nilai dan mencetak nilai matakuliahnya (mencetak nilai dapat dilakukan sendiri di rumah atau warung internet).

ii. Dari sisi akademik

Pada sisi akademik, manfaat yang akan dapat dirasakan adalah:

(a) Data mahasiswa menjadi mudah untuk dikontrol dan pihak pimpinan dapat mengetahui data mahasiswa secara keseluruhan (baik kampus pusat dan kampus cabang) dengan cepat tanpa harus menunggu terkumpulnya semua data dari kampus cabang.

(b) Data nilai matakuliah mahasiswa menjadi mudah dikontrol oleh pihak kampus pusat dan dapat dengan cepat disajikan melalui *website* khusus untuk mahasiswa.

(c) Efisiensi biaya dapat dilakukan pada pengurangan penggunaan kertas untuk mencetak data mahasiswa maupun pencetakan pada data nilai matakuliah mahasiswa.

Tinjauan Pustaka

Integrasi Data

Integrasi data merupakan penggabungan data dari berbagai sumber yang berbeda dan menyediakannya pada pengguna dalam bentuk yang seragam tentang data tersebut. Integrasi data hadir seiring dengan peningkatan jumlah data dan kebutuhan untuk menyatukan data tersebut menjadi satu kesatuan yang dapat digunakan secara bersama. Sebuah aplikasi yang menampilkan data kepada pengguna, biasanya mengambil data dari satu lokasi saja. Akan tetapi dengan meningkatnya perkembangan perusahaan yang mengembangkan sayap dengan membuka

cabang yang tersebar pada lokasi yang berbeda, maka data tersebut dapat menjadi terpisah – pisah dengan data yang berada pada lokasi pusat. Agar tetap dapat menampilkan data dan memberikan informasi yang tepat kepada pengguna, maka dibutuhkan integrasi data dari data yang tersebar tersebut.

Database Terdistribusi

Pada *database* terdistribusi terdapat *database* logis yang secara fisik tersebar pada beberapa komputer *server* yang berbeda lokasi dengan dihubungkan melalui jaringan komunikasi data.

1. Kelebihan *Database* Terdistribusi

Adapun kelebihan yang didapat dari konsep *database* terdistribusi adalah:

- a) Dapat meningkatkan keandalan dan ketersediaan data.
- b) Desentralisasi pengelolaan data.
- c) Pertumbuhan secara modular.
- d) Menurunkan biaya komunikasi data.
- e) Response time yang lebih cepat terhadap *query – query* tertentu yang hanya melibatkan data lokal.

2. Kekurangan *Database* Terdistribusi

Sedangkan kekurangan yang terdapat pada *database* terdistribusi dapat dilihat pada point – point berikut ini:

- a) Harga dan kompleksitas perangkat lunak yang tinggi.
- b) Biaya overhead pemrosesan.
- c) Resiko integritas data.
- d) Response time lambat untuk *query – query* tertentu yang melibatkan *database – database* tersebar.
- e) Membutuhkan koneksi jaringan yang bersifat *private* (VPN – *Virtual Private Network*) agar dapat mengakses data pada *database – database* yang tersebar.

Pendistribusian Database

Terdapat 2 strategi yang dapat dilakukan dalam melakukan pendistribusian *database*, yaitu:

1. Replikasi Data

Replikasi data merupakan salah satu strategi yang dapat dilakukan dengan cara melakukan duplikasi data yang tersebar di

beberapa *server* yang terpisah secara fisik.

Dengan adanya replikasi data, maka data dapat diduplikasi pada lebih dari satu *database server* dan tipe replikasi data yang dapat digunakan dibagi menjadi 2 cara, yaitu:

a) Replikasi *Push*

Pada tipe replikasi *push*, *server* pusat melakukan sinkronisasi data pada tiap *server* lokal dan *server* pusat akan melakukan sinkronisasi terhadap data yang dianggap sebagai data yang penting untuk diubah pada setiap *server* lokal.

Kontrol sinkronisasi terdapat pada *server* pusat dan *server* lokal hanya dapat menerima perubahan yang dilakukan oleh *server* pusat tersebut.

b) Replikasi *Pull*

Tipe replikasi *pull*, *server* lokal yang menentukan kapan perubahan data akan dilakukan sinkronisasi dengan *server* pusat. Sehingga dengan cara ini, maka *server* lokal yang melakukan kontrol terhadap perubahan data.

Proses sinkronisasi data menjadi tidak terlalu mengganggu dan proses replikasi hanya akan dilakukan pada saat *server* lokal membutuhkannya.

2. Partisi Data

Strategi yang dapat diterapkan berikutnya adalah dengan melakukan partisi terhadap data. Data yang ada pada *database server* pusat disimpan pada lokasi *server* cabang yang saling memiliki hubungan (*relation*) agar nantinya dapat digunakan untuk melakukan penggabungan data melalui perintah – perintah *query*.

Partisi data dapat dilakukan menggunakan 3 metode dalam melakukan partisi tersebut, yaitu:

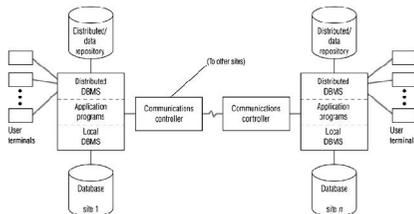
a) Partisi Horizontal

Partisi horizontal merupakan metode partisi data yang digunakan untuk mengambil data yang ada secara baris *record*. Beberapa data yang mempunyai hubungan (*relation*) disimpan dalam bentuk *record* pada *server* pusat dan *server* lokal, dimana data yang mempunyai relasi ini nantinya digunakan untuk menghubungkan dalam melakukan *query*.

- b) Replikasi Vertikal
 Sedangkan pada partisi vertikal merupakan metode partisi data yang mengambil data secara kolom. Hampir sama seperti pada partisi horizontal, bedanya adalah pada partisi vertikal agak sulit dalam melakukan penggabungan data hasil *query*, karena membutuhkan operasi *join*. Berbeda dengan partisi horizontal yang membutuhkan operasi *union* yang lebih mudah dalam melakukan penggabungan data hasil *query*.
- c) Replikasi Hybrid
 Pada metode partisi *hybrid* merupakan metode yang menggabungkan metode partisi horizontal dan partisi vertikal. Sehingga dengan metode hybrid ini, maka penggabungan data hasil *query* dari komputer *server* yang tersebar dapat disesuaikan dengan kebutuhan.

Arsitektur Database Terdistribusi

Arsitektur *database* terdistribusi dapat dilihat pada gambar 1 berikut ini:



Gambar 1: Arsitektur Database Terdistribusi [1]

Services Oriented Architecture

SOA merupakan sebuah pendekatan untuk membangun aplikasi dengan melakukan kolaborasi antara *Consumers* dan *Services Provider* yang didukung dengan adanya kemampuan, prinsip dan standarisasi yang dimiliki oleh SOA dalam membangun aplikasi untuk kelas enterprises dengan adanya fleksibilitas dan waktu reaksi yang singkat dalam memberikan respon; meskipun pada tiap sistem mempunyai arsitektur yang berbeda (*heterogeneous systems*).

Prinsip Dasar SOA antara lain:

1. *Loose Coupling*
Loose Coupling mengacu pada

keleluasaannya hubungan antara modul, komponen, *service consumers* dan *service provider*, sehingga pada *loose coupling*, hubungan tersebut tidak saling ketergantungan satu sama lainnya. Hal ini sangat baik, karena setiap *service* tidak saling tergantung dan dapat meningkatkan fleksibilitas.

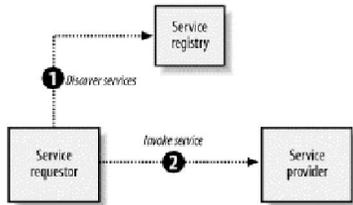
2. *Interoperability*
Interoperability merupakan prinsip SOA untuk dapat saling berinteraksi antara bahasa pemrograman yang satu dengan yang lainnya, sehingga tidak harus selalu menggunakan bahasa pemrograman yang sama dalam menerapkan SOA.
 3. *Reuseability*
 Layanan SOA yang sudah ada dapat dimanfaatkan untuk membangun layanan baru yang memanfaatkan layanan SOA yang sudah ada tersebut tanpa harus membangunnya dari awal, sehingga layanan SOA yang baru dapat berfokus pada proses bisnis yang dibuat.
 4. *Discoverability*
 Prinsip berikutnya adalah SOA harus mudah ditemukan sehingga dapat digunakan oleh *Consumers*. Baik hanya menggunakan saja atau dapat menggunakan kembali layanan SOA tersebut untuk keperluan yang dibutuhkan oleh *Consumers*.
 5. *Governance*
 Pengertian *governance* yang terdapat pada prinsip SOA merujuk pada penyediaan aturan dan *framework* dalam melaksanakan prinsip SOA untuk dapat mengukur dan melakukan identifikasi peluang yang tepat terhadap pengembangan SOA tersebut.
 Prinsip *governance* harus dipertimbangkan dari sisi waktu pembuatan (*design-time*), waktu pengerjaan (*bind-time*) dan sisi waktu pelaksanaan (*run-time*).
- Web services**
Web services merupakan sebuah layanan yang dapat diakses melalui media internet dengan menggunakan sekumpulan standarisasi (XML) dan tidak terikat dengan jenis sistem operasi tertentu maupun bahasa pemrograman tertentu untuk dapat saling

berkolaborasi antar data yang diperlukan. Arsitektur *web services* dapat dilihat dari:

1. *Web services Role*

Terdapat 3 tugas *web services* dalam arsitektur *web services*, yaitu:

- a) *Service Provider*
- b) *Service Requestor*
- c) *Service Registry*



Gambar 2: *Web Service Roles* [2]

2. *Web Service Protocol Stack*

- a) *Service Transport Layer*
- b) *XML Messaging*
- c) *Service Description*
- d) *Service Discovery*

Discovery	UDDI
Description	WSDL
XML messaging	XML-RPC, SOAP, XML
Transport	HTTP, SMTP, FTP, BEEP

Gambar 3: *Web Service Protocol Stack* [2]

WSO2 WSF/PHP

WSO2 *Web Services Framework* PHP merupakan sebuah framework yang dapat membantu implementasi *web services*, sehingga dapat menyediakan dan memanfaatkan *web services* pada bahasa pemrograman PHP.

Systems Development Life Cycle (SDLC)

merupakan sebuah proses tentang bagaimana sebuah sistem informasi dapat mendukung kebutuhan bisnis, melakukan desain sistem, membangunnya dan melakukan implementasi sistem yang dibutuhkan tersebut untuk dapat digunakan oleh pengguna.

Fase – fase SDLC:

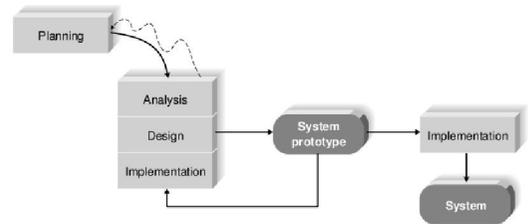
- 1. *Planning*
- 2. *Analysis*

3. *Design*

4. *Implementation*

Prototype Development Methodology

Dalam menerapkan SDLC, metodologi yang dimanfaatkan adalah *Prototype Development Methodology*. Dengan *prototype development*, maka tiap tahapan SDLC dapat dilakukan secara bersamaan dan tiga tahap (*analysis, desain, implementation*) dilakukan berulang – ulang sampai lengkapnya sebuah sistem.



Gambar 4: *Prototype Development Methodology* [3]

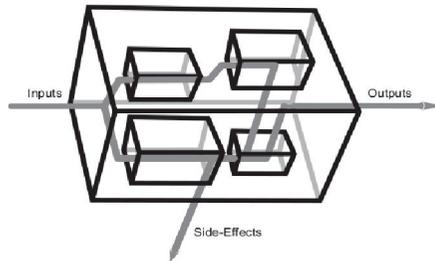
Keunggulan yang didapat dari *prototype methodology* adalah waktu yang dibutuhkan untuk menyediakan sistem yang sangat cepat; sehingga pengguna dapat langsung berinteraksi untuk menggunakan sistem tersebut, meskipun belum sepenuhnya berfungsi. Dan pengguna juga dapat memberikan masukan (*feedback*) terhadap hal apa saja yang dibutuhkannya dengan mencoba menggunakan *prototype* yang diberikan.

COTS

COTS (*Commercial Off The Shelf*) merupakan penggunaan kembali (*reuse*) terhadap aplikasi/komponen yang sudah ada untuk dimanfaatkan pada keperluan aplikasi lain; sehingga komponen yang digunakan pada aplikasi yang baru tidak dibuat dari awal, tapi dengan memanfaatkan kembali aplikasi/komponen yang sudah ada.

White Box Testing

White Box Testing merupakan metodologi pengujian sistem yang mengacu pada teknik untuk menguji sistem dengan pengetahuan yang cukup pada sisi internal dari sistem tersebut. *White Box Testing* dilakukan dengan menganalisa *source code* dari sistem terhadap *error* yang kemungkinan terjadi pada *source code* tersebut.



Gambar 5: White Box Testing [4]

Tinjauan Obyek Penelitian

Universitas Budi Luhur adalah sebuah universitas swasta yang merupakan penggabungan dari Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK), Sekolah Tinggi Ilmu Ekonomi (STIE), Sekolah Tinggi Teknik (STT) dan Sekolah Tinggi Ilmu Sosial dan Ilmu Politik (STISIP), diresmikan pendiriannya pada tanggal 7 Juni 2002, berdasarkan surat keputusan Menteri Pendidikan Nasional Republik Indonesia Nomor 114/D/O/2002. Universitas Budi Luhur yang merupakan suatu perwujudan karya nyata Yayasan Pendidikan Budi Luhur bagi bangsa dan negara Republik Indonesia, dirancang untuk ikut mencerdaskan anak bangsa dengan dilandasi budi pekerti yang luhur. Universitas Budi Luhur adalah sebuah perguruan tinggi swasta yang mengemban misi dari Yayasan Pendidikan Budi Luhur, dengan berlandaskan pada motto Asah-Asih-Asuh. Universitas Budi Luhur menyelenggarakan pendidikan, mengembangkan dan menyebarkan serta mengabdikan ilmu pengetahuan, teknologi, kesenian dan ilmu kemanusiaan serta ilmu sosial untuk mengupayakan agar bangsa Indonesia menjadi bangsa yang cerdas dan berbudi luhur.

Kerangka Konsep

Berdasarkan hasil pengamatan sementara dan kajian teori yang telah disusun serta tujuan penelitian yang telah ditetapkan, maka kerangka konsep tentang model akses *database* terdistribusi berdasarkan pendekatan *services oriented architecture* yang berbasis *web services* diharapkan dapat diimplementasikannya model akses *database* terdistribusi untuk dapat meningkatkan performa akses data pada *database* terdistribusi antara kampus cabang dengan kampus pusat tanpa membebani *server* kampus cabang.

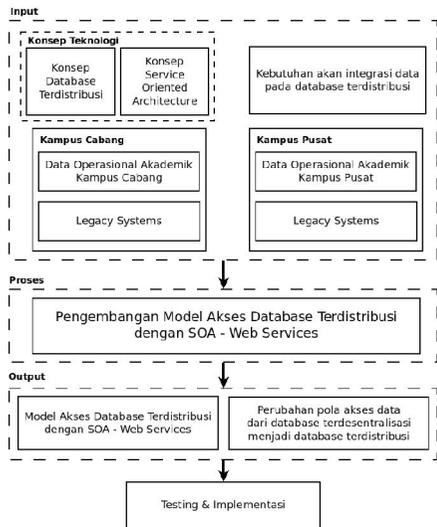
Agar dapat membuat kerangka konsep tersebut, maka perlu diketahui kondisi yang ada saat ini. Kondisi saat ini dapat dilihat pada gambar berikut ini:



Gambar 6: Kondisi Saat Ini

Seperti yang terlihat pada gambar, bahwa kondisi saat ini adalah sudah terdapat *legacy systems* pada kampus cabang dan kampus pusat dimana terdapat data operasional akademik untuk menyimpan data operasional yang terjadi. Akan tetapi untuk dapat melakukan integrasi data, maka kampus cabang harus melakukan *transfer* data operasional akademiknya pada kampus pusat. *transfer database* operasional ini masih dilakukan secara manual dengan mengantarkannya menggunakan media *flashdisk* maupun *compact disc* dari kampus cabang kepada kampus pusat.

Setelah mengetahui kondisi yang ada, maka kerangka konsep yang sudah dijelaskan sebelumnya dapat digambarkan sebagai berikut:



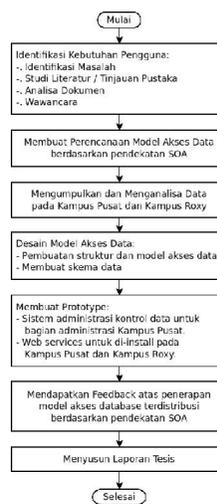
Gambar 7: Kerangka Pemikiran

Dengan adanya kampus cabang dan kampus pusat yang sudah memiliki *legacy systems* untuk melakukan operasional akademik pada masing – masing kampus dan adanya kebutuhan untuk melakukan integrasi data dari kampus cabang dengan kampus pusat, maka diperlukan sebuah cara yang lebih baik dari kondisi yang ada sebelumnya untuk melakukan *transfer database* dari kampus cabang kepada kampus pusat. Hal tersebut didukung dengan hadirnya teknologi tentang *database terdistribusi* dan konsep *services oriented architecture* untuk melakukan integrasi *database* yang dibutuhkan dan menjadi *input* untuk mengembangkan model akses *database terdistribusi* berdasarkan pendekatan *services oriented architecture* yang berbasis *web services*.

Pengembangan ini merupakan proses dari *input* yang ada sebelumnya untuk mendapatkan *output* yang berupa model akses *database terdistribusi* berdasarkan pendekatan *services oriented architecture* yang berbasis *web services* serta mengubah pola akses data dari *database terdesentralisasi* yang selama ini masih diterapkan oleh Universitas Budi Luhur menjadi *database terdistribusi*.

Adapun dalam mencapai kerangka konsep tersebut, maka diperlukan langkah – langkah yang dilakukan dalam penelitian. Langkah –

langkah penelitian dapat dilihat pada gambar berikut ini:



Gambar 8: Langkah – langkah penelitian

Hipotesis

Berdasarkan kajian teoritis dan kerangka konsep yang telah dikemukakan, maka dapat dirumuskan hipotesa penelitian sebagai berikut:

1. Diduga pembangunan model akses *database terdistribusi* berdasarkan pendekatan *service oriented architecture* akan dapat meningkatkan performa akses data yang lebih baik terhadap data yang tersebar.
2. Diduga pembangunan model akses *database terdistribusi* berdasarkan pendekatan *service oriented architecture* tidak mengubah *legacy systems*; dimana pada *web service* yang akan diterapkan, menggunakan model TEL (*Transfer, Extract, Load*) yang mana:
 - a. Data tersebar yang terdapat pada *database* kampus cabang, secara otomatis dan periodik dikirimkan (*Transfer*); sehingga memberikan kemudahan (pengguna tidak merasakan adanya proses *transfer data* tersebut) dan tidak membebani *server* pada kampus cabang, karena proses pada kampus cabang hanya melakukan pengiriman data yang berbentuk XML.
 - b. Data yang berbentuk XML yang diterima pada kampus pusat akan

dilakukan proses *Extract* untuk dapat di-*Load* pada *database* kampus pusat, sehingga pemrosesan data sepenuhnya terdapat pada kampus pusat.

Metodologi Penelitian

Metode Penelitian

Penelitian ini menggunakan metode Deskriptif Kualitatif dengan metode studi kasus yang bertujuan untuk mendapatkan gambaran yang lebih mendalam dan lengkap dari subyek penelitian dengan melakukan pengamatan (observasi), wawancara dan mendapatkan *feedback*.

Metode Pengumpulan Data

Pengumpulan data yang dilakukan pada penelitian ini berasal dari data primer dan data sekunder. Dimana pada data primer, teknik pengumpulan data dilakukan dengan wawancara, observasi lapangan, uji coba serta mendapatkan *feedback*. Sedangkan data sekunder didapat dari studi literatur, obyek penelitian, tulisan ilmiah tentang *services oriented architecture* dan *web services*.

Pengumpulan Data

Data yang dibutuhkan dalam penelitian ini dikumpulkan melalui wawancara dan mendapatkan *feedback* dengan *key person* yang diwawancarai:

1. Pihak Manajemen Eksekutif dan Operasional:
 - a. Kepala Biro Administrasi Akademik Dan Kemahasiswaan.
 - b. Koordinator Unit Cabang Roxy.
2. Pihak Pengembang Sistem Informasi:
 - a. Kepala Biro Sistem Informasi pada Kampus Pusat.
 - b. IT Support pada Kampus Roxy.
3. Beberapa mahasiswa kampus Roxy yang mengalami kendala dalam mengakses *website* khusus untuk mahasiswa yang disebabkan oleh belum di-*transfer*-nya data mahasiswa dan data nilai mahasiswa kepada kampus pusat.

Observasi Lapangan

Untuk observasi lapangan, dilakukan metode pengumpulan data yang digunakan untuk menghimpun data penelitian melalui pengamatan.

Studi Literatur

Studi literatur dilakukan dengan

membaca struktur XML tersebut agar mempelajari dari berbagai referensi yang terkait dengan model *database* terdistribusi dan *services oriented architecture* berbasis *web services*.

Instrumentasi

Instrumentasi utama pada penelitian ini adalah peneliti sendiri dengan melakukan:

1. Identifikasi sistem komputerisasi yang ada saat ini (*as-is systems*).
Dalam melakukan identifikasi sistem komputerisasi yang sedang berjalan (*as-is systems*) pada Kampus Pusat dan Kampus Roxy, maka dilakukan observasi pada kampus tersebut untuk mendapatkan arsitektur data, arsitektur infrastruktur dan sistem *database* yang sedang berjalan.
2. Melakukan wawancara pada *key person* yang terkait untuk mendapatkan kebutuhan SOA yang akan dibangun.
3. Mendapatkan *feedback* dari *key person* yang terkait setelah diterapkannya SOA.

Rancang Bangun Dan Uji Model

Project Management and Requirement

Mendefinisikan Project

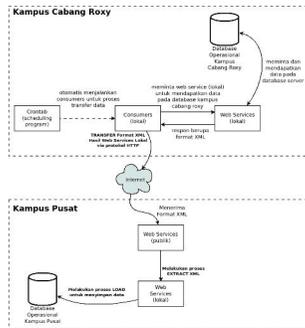
Model akses *database* terdistribusi yang akan dibangun adalah dengan melakukan integrasi data mahasiswa dan data nilai matakuliah mahasiswa yang ada pada kampus cabang Roxy dan kampus Pusat Universitas Budi Luhur. Hal ini dilakukan karena adanya kebutuhan akan integrasi data tersebut agar mahasiswa kampus cabang dapat melihat data mahasiswanya dan data nilai matakuliahnya secara *online* melalui *website* khusus untuk mahasiswa yang telah disediakan oleh pihak Universitas Budi Luhur.

Analisa Kebutuhan

Analisa kebutuhan (*requirement definition*) dilakukan berdasarkan pengumpulan data, yaitu:

1. Dibutuhkan sebuah model akses *database* terdistribusi yang mampu untuk melakukan integrasi data mahasiswa dan data nilai mahasiswa dari kampus cabang kepada kampus pusat.
2. Dibutuhkan integrasi data mahasiswa dan data nilai mahasiswa yang dilakukan

kampus cabang Roxy kepada *web services*



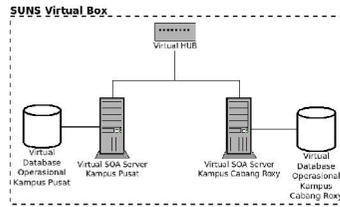
Gambar 13: Proses *Transfer, Extract* dan *Load*

Berikutnya, format XML yang diterima oleh *web services* (publik) pada kampus pusat akan meneruskan proses pada *web services* (lokal) untuk dilakukan proses *EXTRACT* dari format XML tersebut menjadi format yang dapat dimengerti oleh *database server* untuk kemudian dilakukan *LOAD* sehingga data dapat disimpan pada *database server* kampus pusat.

Pengujian

Pengujian akan dilakukan dengan menerapkan metodologi *White Box Testing* pada skala lab. Pengujian dilakukan dari sisi internal sistem dengan menganalisa *source code* dari sistem terhadap *error* yang kemungkinan terjadi. Sedangkan skala lab yang dimaksud adalah bahwa pengujian dilakukan dengan membuat arsitektur yang mirip dengan keadaan pada lapangan. Skala lab yang digunakan pada pengujian, menggunakan aplikasi *SUNS Virtual Box* dengan melakukan simulasi 2 komputer yang terdapat dalam 1 komputer utama. Komputer *server* kampus Pusat dan komputer *server* kampus cabang Roxy dibuat *virtual* yang seolah – olah terpisah. Berikut adalah gambar pengujian skala lab yang dimaksud:

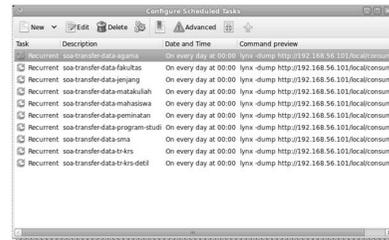
kampus pusat.



Gambar 14: *Testing Skala Lab. Secara Virtual*

Uji Fungsi Model

1. Kampus Cabang Roxy
 - a) Uji Fungsi Aplikasi *Crontab*

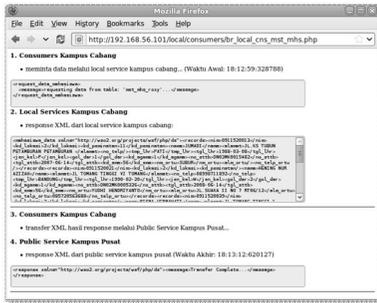


Gambar 15: *Crontab / Schedule Task Interface*

- b) Uji Fungsi Proses TEL (*Transfer, Extract* dan *Load*)

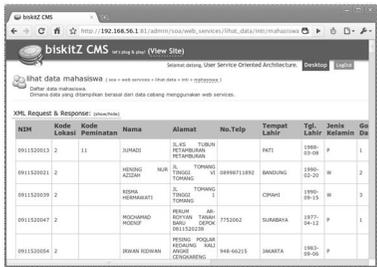
Secara umum, proses TEL yang terjadi adalah sebagai berikut:

- (1) Aplikasi *Crontab* secara otomatis dan periodik memanggil *consumers* yang telah disiapkan untuk melakukan request pada *web service* lokal yang ada pada kampus cabang.
- (2) *Request* yang berupa format XML ini diterima oleh *web service* lokal untuk membaca data yang ada pada *database server* kampus cabang dan memberikan *respon* kepada *consumers* berupa data yang telah diambil dari *database server* dalam bentuk XML.
- (3) Berikutnya, *consumers* akan melakukan *TRANSFER* format XML ini pada *web service* publik kampus Pusat.
- (4) Kampus Pusat selanjutnya akan melakukan proses *EXTRACT* sehingga data dengan format XML yang diterima dapat di *LOAD* pada *database* kampus Pusat untuk disimpan.



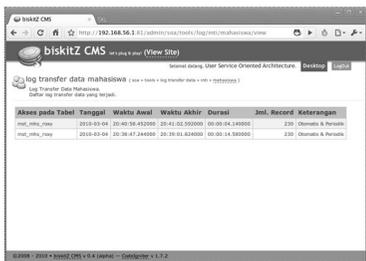
Gambar 16: Uji Fungsi Consumers Data Mahasiswa

2. Kampus Pusat
 - a) Uji Fungsi Daftar Data Mahasiswa



Gambar 17: Daftar Data Mahasiswa Kampus Roxy via Web Services

- b) Uji Fungsi Log Transfer Data Mahasiswa



Gambar 18: Log TEL Data Mahasiswa

3. Pemanfaatan Model Akses Database Terdistribusi



Gambar 19: Laporan Data Mahasiswa Kampus Roxy via Web Services

User Acceptance Test

Pengujian terhadap penerimaan dari model akses database terdistribusi berdasarkan pendekatan *services oriented architecture* didapat dengan mendapatkan *feedback* dari *key person* terkait.

Key person yang terkait memberikan tanggapan (*feedback*) terhadap model akses database terdistribusi dalam menjawab kebutuhan akan integrasi data antara kampus cabang Roxy dengan kampus Pusat Universitas Budi Luhur. Tanggapan – tanggapan dari *key person* ini digunakan sebagai pertimbangan akan penerimaan model atau penolakan terhadap model akses database terdistribusi yang diterapkan.

Dengan mendapatkan *feedback*, maka model akses database terdistribusi yang telah dilakukan dapat disimpulkan bahwa penerimaannya sudah sesuai dengan kebutuhan.

Pengaruh Perubahan

Pemanfaatan model akses database terdistribusi dengan berdasarkan pendekatan *service oriented architecture* memberikan beberapa perubahan proses bisnis terutama dalam hal transfer data mahasiswa dan data nilai mahasiswa dari kampus cabang Roxy kepada kampus Pusat Universitas Budi Luhur.

Pada model akses database terdistribusi yang dikembangkan memiliki beberapa keuntungan dari sisi kampus cabang Roxy, yaitu pihak administrasi tidak perlu lagi mengantarkan data mahasiswa dan data nilai mahasiswa kepada kampus Pusat secara

manual. Proses *transfer* data ini dilakukan secara otomatis dan periodik oleh sistem, sehingga pihak administrasi kampus cabang Roxy pun tetap dapat berkonsentrasi pada kegiatan akademik yang terjadi pada kampus cabang Roxy untuk melayani mahasiswa dengan sistem yang sudah ada saat ini (*legacy systems*). Hal ini dapat terjadi karena *web service* hanya di-*install* pada komputer *server* kampus Roxy sebagai penambahan dan tidak mengubah *legacy systems* yang ada saat ini.

Selain itu, pihak kampus Pusat juga dapat melihat *log* dari proses *transfer*, *extract* dan *load* (TEL) yang terjadi beserta durasi yang dibutuhkan untuk proses TEL tersebut. Dan penyajian informasi pada *website* khusus untuk mahasiswa menjadi lebih cepat untuk disajikan kepada mahasiswa.

Dan dari pihak mahasiswa, terutama mahasiswa cabang Roxy dapat melihat biodata dan data nilai matakuliah yang diikutinya melalui *website* khusus mahasiswa dengan lebih cepat (karena akibat dari proses integrasi data yang dilakukan secara otomatis dan periodik pada kampus Roxy kepada kampus Pusat), dibandingkan dengan proses *transfer* yang dilakukan secara manual sebelumnya.

Implikasi Penelitian

Aspek Manajerial

Implikasi penelitian terhadap aspek manajerial adalah sebagai berikut:

1. Model akses *database* terdistribusi dapat dimanfaatkan untuk keperluan manajerial dalam hal penyampaian informasi yang bersifat laporan. Dengan mengolah data yang sudah didapat dari hasil integrasi data, maka dapat dibuatkan laporan yang disesuaikan dengan kebutuhan dari sisi manajerial.
2. Pada aspek manajerial, ketersediaan informasi yang dibutuhkan dari data yang tersebar menjadi lebih cepat disajikan untuk kebutuhan manajerial, karena integrasi data dilakukan secara otomatis dan periodik.

Aspek Sistem

Pada aspek sistem, implikasi penelitian yang terjadi antara lain adalah:

1. Dengan model akses *database*

terdistribusi berdasarkan pendekatan *services oriented architecture*, maka penerapan *web services* yang dilakukan tidak mengubah sistem yang sudah ada (*legacy systems*).

2. Pada aspek sistem, model akses *database* terdistribusi dapat mengubah proses bisnis untuk melakukan integrasi data dari data yang tersebar yang sebelumnya dilakukan secara manual menjadi lebih mudah karena dilakukan secara otomatis dan periodik.

Aspek Penelitian Lanjut

Aspek penelitian lanjut yang dapat dilakukan yaitu:

1. Penelitian ini dapat dilanjutkan untuk dikembangkan pada penelitian data *warehouse* terdistribusi (*distributed data warehouse*) berdasarkan pendekatan *services oriented architecture*.
2. Pengembangan juga dapat dilanjutkan untuk penelitian data *mining* terdistribusi (*distributed data mining*) berdasarkan pendekatan *services oriented architecture*.
3. Integrasi data yang dilakukan tidak terbatas pada integrasi data mahasiswa dan data nilai mahasiswa saja. Lebih lanjut, integrasi data dapat dilakukan untuk semua data yang ada pada kampus cabang dan kampus Pusat.

Penutup

Kesimpulan

Berdasarkan penelitian pengembangan yang telah dilakukan, bahwa akses *database* terdistribusi berdasarkan pendekatan *service oriented architecture* berbasis *web service* dapat memberikan kelebihan, antara lain:

1. Secara Umum
 - a) Dengan model akses *database* terdistribusi berdasarkan pendekatan *services oriented architecture* yang dapat melakukan integrasi data dari beberapa lokasi yang berbeda, maka hal ini dapat dimanfaatkan bagi perusahaan – perusahaan yang sudah mapan dan mempunyai lokasi cabang yang berbeda untuk dapat melakukan integrasi data dari *database* yang tersebar.

- b) Dengan dapat melakukan integrasi data dari beberapa lokasi yang berbeda, maka pengembangan dapat dilakukan untuk keperluan *distributed data warehouse* maupun *distributed data mining* berdasarkan pendekatan *services oriented architecture*.
- 2. Secara Khusus
 - a) Skema arsitektur model akses *database* terdistribusi menjadi lebih baik yang disebabkan oleh adanya *intermediary* yang berupa *web service* untuk melakukan integrasi *database* antara kampus Roxy dengan kampus Pusat berdasarkan pendekatan *services oriented architecture*.
 - b) *Web service* yang diterapkan pada kampus Roxy tidak mengganggu *legacy systems* yang ada, karena yang dilakukan oleh *web service* hanya melakukan proses *transfer* data dari kampus Roxy kepada kampus Pusat.
 - c) Proses *transfer* data yang dilakukan dari kampus Roxy kepada kampus Pusat tidak mengganggu kinerja pihak administrasi kampus Roxy, karena proses tersebut dilakukan secara otomatis dan periodik.
 - d) Mengubah konsep *database* terdesentralisasi yang selama ini diterapkan oleh Universitas Budi Luhur menjadi konsep *database* terdistribusi.
 - e) Peningkatan performa akses *database* yang lebih baik dan dapat dilihat pada hasil *log* yang terjadi pada saat proses TEL dilakukan dibandingkan dengan melakukan *transfer* secara manual (dikirim via media *flashdisk* maupun *compact disc*).

Saran

Dengan adanya model akses *database* terdistribusi dengan berdasarkan pendekatan *service oriented architecture* yang ada, dapat dikembangkan lagi lebih lanjut untuk kebutuhan *transfer* tidak hanya dari kampus Roxy, akan tetapi dari kampus cabang lainnya yang dimiliki oleh Universitas Budi Luhur untuk saling dapat melakukan integrasi data agar dapat memudahkan mahasiswa kampus cabang lainnya dalam hal melihat data mahasiswanya dan data

nilai matakuliah yang diikutinya.

Selain itu, pengembangan dapat dilakukan tidak hanya pada data mahasiswa dan data nilai mahasiswa, data yang terdapat pada *database* terdistribusi yang ada pada cabang – cabang Universitas Budi Luhur dapat diintegrasikan dengan menggunakan model akses *database* terdistribusi berdasarkan pendekatan *service oriented architecture* ini. Penyediaan laporan untuk pimpinan terhadap informasi yang penting dapat disajikan dengan lebih baik tidak saja hanya terbatas pada penyediaan laporan data mahasiswa dan data nilai mahasiswa saja. Pengembangan kedepan dari model akses *database* terdistribusi dapat diteruskan oleh peneliti lain untuk membuat model data *warehouse* terdistribusi (*distributed data warehouse*) maupun model data *mining* terdistribusi (*distributed data mining*). Hal ini tentu dapat sangat bermanfaat bagi pengelola bisnis dalam mendapatkan informasi eksekutif dari cabang tersebar yang dimilikinya dan dapat digunakan secara *mobile*.

Sedangkan kelemahan yang ada pada *service oriented architecture* adalah dapat terjadinya waktu *transfer* yang lambat dikarenakan infrastruktur yang kurang memadai. Hal ini dapat diatasi dengan memperbaiki pada sisi infrastruktur terutama pada penyediaan bandwidth yang tersedia pada kampus cabang untuk ditingkatkan.

Daftar Pustaka

- [1] Hoffer, Jeffrey A., et.al, *Modern Database Management*, 8th ed., Pearson Prentice Hall, New Jersey, 2007.
- [2] Cerami, Ethan, *Web Services Essentials*, 1st ed., O'Reilly & Associates, Inc, Canada, 2002.
- [3] Dennis, Alan, et.al, *Systems Analysis and Design with UML*, 3rd ed., John Wiley & Sons, Ltd, Indianapolis, 2009.
- [4] O'Dochtery, Mike, *Object-Oriented Analysis and Design Understanding System Development with UML 2.0*, John Wiley & Sons, Ltd, England, 2005.
- [5] Bean, James, *SOA and Web Services*

- Interface Design – Principles, Techniques, and Standards*, Elsevier Inc, Massachusetts, 2010.
- [6] Cohen, Frank, *FastSOA*, Morgan Kaufmann Publishers, San Francisco, 2007.
- [7] Durbin, Jason, et.al, *Distributed Database Systems*, Oracle Corporation, California, 1999.
- [8] Deviana, Hartati, 2007, *Penerapan XML Web Service Untuk Sistem Distribusi Barang Studi Kasus: PT. Apotik Plus Palembang*, Tesis, Yogyakarta: Universitas Gajah Mada.
- [9] Dye, Charles, “Oracle Distributed Systems”, 1st ed., O’Reilly & Associates, Inc., Canada, 1999.
- [10] Josuttis, Nicolai M., “SOA in Practice: The Art of Distributed System Design”, 1st ed., O’Reilly & Associates, Inc, Canada, 2007.
- [11] Pungus, Stenly. R., 2008, “Penerapan Service Oriented Architecture Untuk Pengintegrasian Sistem Informasi Perguruan Tinggi”, Tesis, Bandung: Institut Teknologi Bandung.
- [12] Rittinghouse, John W. and James F. Ransome, “Cloud Computing: Implementation, Management, and Security”, CRC Press, Boca Raton, 2010.
- [13] Rosen, Mike, et.al, “Applied SOA: Service-Oriented Architecture and Design Strategies”, Wiley Publishing, Inc., Indianapolis, 2008.
- [14] Universitas Budi Luhur, “Menuju Masyarakat Cerdas Berbudi Luhur”, Universitas Budi Luhur, 2009, Jakarta.
- [15] Yunus, Mamoon and Rizwan Mallal, “SOA Testing using Black, White and Gray Box Techniques”, Crosscheck Networks, 2006.