

MODEL DATA WAREHOUSE DENGAN SERVICE ORIENTED ARCHITECTURE UNTUK MENUNJANG SISTEM INFORMASI EKSEKUTIF

Joko Christian

Fakultas Teknologi Informasi Universitas Budi Luhur, joko.christian@budiluhur.ac.id

Abstrak

Tuntutan untuk meningkatkan kualitas pendidikan merupakan tantangan bagi pihak universitas untuk terus mengembangkan diri dan memperbaiki kekurangan yang ada. Dalam penyempurnaan kualitas pendidikan dan peningkatan daya saing, pelaksanaan kebijakan yang tepat merupakan salah satu pengaruh yang penting. Pihak manajemen berkewajiban untuk menghasilkan kebijakan dan solusi tepat berdasarkan informasi akurat dan strategis yang bersumber dari *data warehouse* yang baik.

Secara umum sebagian besar *data warehouse* dikembangkan dengan teknologi lama, yaitu bentuk *direct data access*. Dewasa ini *data warehouse* sudah menggunakan arsitektur *Service Oriented Architecture* (SOA) yang menyediakan akses data melalui *middleware* (akses tidak langsung). Penelitian ini bertujuan tidak hanya mengoptimalkan SOA dalam *data warehouse*, tapi juga mengintegrasikan kebutuhan sistem informasi eksekutif dalam struktur *data warehouse*. Sebagai model yang digunakan, digunakan data-data dari Fakultas Teknologi Informasi Universitas Budi Luhur yang saat ini belum memiliki *data warehouse* untuk sistem informasi eksekutifnya. Metode pengembangan *data warehouse* ini akan menggunakan pendekatan *Business Life Cycle*.

Kata Kunci : *data warehouse*, *service oriented architecture*, informasi strategis, sistem informasi eksekutif

1. Pendahuluan

Latar Belakang

Salah satu faktor yang mempengaruhi arah pengembangan pendidikan di masing-masing lembaga pendidikan ialah proses pembuatan kebijakan. Kebijakan yang tepat dan sesuai dengan strategi bisnis yang telah ditentukan akan mendorong masing-masing elemen pendidikan untuk bekerja optimal. Untuk menentukan kebijakan tersebut diperlukan manajemen yang bertanggung-jawab, cepat dan tanggap atas perubahan, dan ketersediaan informasi strategis yang akurat.

Tipe informasi strategis merupakan informasi yang terintegrasi dan bersifat strategis karena dibutuhkan untuk keputusan manajemen yang berdampak pada keberlangsungan suatu lembaga pendidikan.

Fakultas Teknologi Informasi sebagai salah satu fakultas pada Universitas Budi Luhur,

telah memiliki data operasional yang lengkap dari kegiatan perkuliahan. Hal ini dimungkinkan karena hampir seluruh proses bisnis akademis telah terkomputerisasi.

Penggunaan data operasional harian sebagai sumber informasi eksekutif tidak memberikan nilai yang memadai, hal ini dikarenakan data operasional memiliki volume yang besar dan tidak memiliki format atau struktur yang sesuai dengan kebutuhan informasi eksekutif yang ingin digunakan.

Pihak manajemen eksekutif telah memiliki sistem informasi eksekutif yang membantu untuk menentukan kebijakan, sistem informasi eksekutif yang dimaksud masih menggunakan bentuk akses data secara langsung (*direct access*) dengan DBMS sehingga memiliki tingkat modularitas yang rendah, terikat pada *legacy system* dan memiliki keamanan yang bergantung penuh pada software DBMS.

Sebagai solusi dari permasalahan tersebut, diperlukan sebuah model *data warehouse*

akademik yang memungkinkan manajemen data yang lebih efektif sebagai komponen sistem informasi eksekutif yang menghasilkan informasi strategis dengan cepat dan akurat.

Untuk mengantisipasi kebutuhan akses informasi oleh *end application* yang berbeda-beda, dan independen terhadap jenis DBMS, model distribusi akan menggunakan pendekatan *service oriented architecture* (SOA), secara khususnya dalam bentuk *web services*. Dengan pendekatan SOA, layanan penyediaan informasi akan memiliki skalabilitas, keamanan dan modularitas yang lebih baik.

Masalah penelitian

Identifikasi masalah

Berdasarkan latar belakang yang telah dinyatakan, ditemukan beberapa permasalahan yang menjadi dasar dari penelitian ini, yaitu:

1. Tidak adanya *data warehouse* yang khusus dikembangkan untuk keperluan analisa strategis.
2. Tidak adanya model distribusi *data warehouse* yang sifatnya independen dan tidak dipengaruhi oleh *legacy DBMS system*.

Batasan masalah

Penelitian ini hanya membahas proses pengembangan model prototipe *data warehouse* dengan SOA untuk keperluan sistem informasi eksekutif Fakultas Teknologi Informasi Universitas Budi Luhur yang berkaitan dengan evaluasi akademik perkuliahan.

Evaluasi yang dimaksud ialah informasi akademik mahasiswa dan kinerja dosen. *Data warehouse* yang dibangun akan menggunakan DBMS MySQL 5 dengan SOA framework WSF/PHP. Penelitian ini tidak membangun sistem informasi eksekutif yang memanfaatkan *data warehouse* dimaksud

Rumusan masalah

Berdasarkan identifikasi masalah yang ada, perumusan masalah penelitian dapat dinyatakan sebagai berikut:

1. Bagaimana bentuk struktur dan arsitektur model *data warehouse* yang sesuai dengan kebutuhan pihak eksekutif.
2. Bagaimanakah model distribusi *data warehouse* yang meningkatkan skalabilitas dan fleksibilitas penggunaan *data warehouse* tanpa mengubah *legacy system*.

Tujuan dan manfaat penelitian

Tujuan Penelitian

Penelitian ini bertujuan untuk menghasilkan model prototipe *data warehouse* dengan SOA yang dapat digunakan pihak manajemen Fakultas Teknologi Informasi untuk mendapatkan informasi strategis sebagai dasar pembuatan kebijakan.

Manfaat Penelitian

Secara umum, hasil penelitian sebagai *pilot project* untuk mengubah arsitektur sistem informasi pada Yayasan Budi Luhur menjadi sistem informasi yang memiliki *data warehouse* yang baik dengan arsitektur SOA, sehingga pengembangan sistem di masa mendatang dapat lebih modular dengan skalabilitas yang lebih baik.

Secara khusus, hasil penelitian diharapkan dapat membantu pihak manajemen dan eksekutif Fakultas Teknologi Informasi Universitas Budi Luhur untuk mendapatkan informasi eksekutif yang lebih akurat dan cepat sebagai dasar analisa untuk pengambilan keputusan dan pembuatan kebijakan. Manfaat lainnya ialah peningkatan skalabilitas *data warehouse* tersebut, sehingga pihak pengembang sistem (Biro Sistem Informasi) dimudahkan untuk pengembangan sistem informasi terkait.

2.Landasan Teori

Tinjauan pustaka

Eksekutif

Dari sudut pandang organisasi, eksekutif ialah orang atau kelompok orang yang memiliki kewenangan administratif atau pengawasan dalam suatu organisasi [1]. Umumnya merupakan manajer senior yang membuat perencanaan dan kebijakan perusahaan.

Eksekutif memiliki 5 fungsi utama, yaitu merencanakan (*planning*), mengorganisasikan (*organizing*), menyusun staf (*staffing*), mengarahkan (*directing*), dan mengendalikan (*controlling*).

Executive Information System

Disebut juga sebagai Executive Support System (ESS) merupakan salah satu bentuk sistem informasi yang disusun dari banyak sumber data dalam bentuk summary yang dipergunakan oleh pihak manajemen senior untuk melakukan monitor *performance*, assessment dan pengembangan strategi bisnis [2].

Untuk memenuhi fungsinya, EIS memiliki karakteristik sebagai berikut :

1. dibuat secara spesifik untuk seorang eksekutif.
2. akan digunakan langsung oleh eksekutif tanpa perantara.
3. mampu melakukan proses ekstrak, menyaring (*filter*), meningkatkan dan melacak "*critical data*".
4. mengakses dan mengintegrasikan data internal dan eksternal.
5. bersifat *user friendly*.

Format data yang disediakan oleh EIS harus memenuhi kebutuhan pihak eksekutif. Berikut adalah karakteristik data yang harus didukung oleh EIS:

1. *Highly summarized data*
2. *Drill down dan drill up*.
3. Integrasi data dari basis data yang berbeda - beda.
4. trend jangka panjang,
5. Dapat mengakses data eksternal
6. Informasi yang disampaikan dalam bentuk faktor penentu kesuksesan (*critical success factors*)

Data warehouse

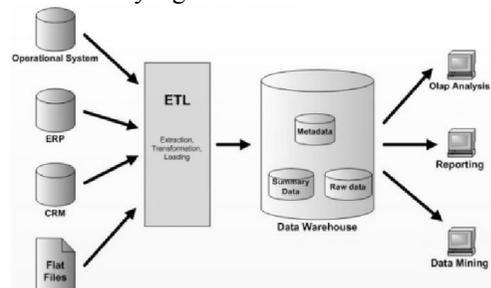
Data warehouse merupakan jantung dan pondasi dari semua proses EIS karena memiliki satu sumber data terintegrasi dengan tingkat granularitas yang tepat [3].

Data warehouse juga dapat didefinisikan dengan dua asumsi implisit, yaitu :

1. Sebuah database yang mendukung sistem pendukung keputusan dan di-maintain secara terpisah dari database operasional perusahaan

2. Sebuah database yang mendukung pemrosesan informasi dengan menyediakan platform yang terintegrasi dan data historis untuk melakukan analisis.

Proses-proses dalam pengembangan *data warehouse* ialah ETL (*Extract-Transform-Loading*). *Extract* ialah seluruh proses yang berkaitan dengan pengambilan dan pengumpulan data (termasuk data dari luar yang dibutuhkan, seperti kode prefix telepon daerah). *Transform* ialah proses penyiapan data agar sesuai dengan kebutuhan seperti pembersihan, *decoding* dan integrasi. *Loading* adalah proses penyimpanan dan pengorganisasian data pada struktur *data warehouse* yang telah ada.



Gambar 1. Proses pengembangan dan pemanfaatan *data warehouse* [4]

Data warehouse memiliki karakteristik sebagai berikut:

1. *Subject oriented*
2. *Integrated*
3. *Time variant*
4. *non volatile*

SOA (Service Oriented Architecture)

SOA merupakan hasil evolusi dari arsitektur berbasis komponen, distributed object protocol, dan framework seperti .NET atau COM. SOA telah memungkinkan proses *rapid deployment*, *business agility* dan *reuse*.

SOA bukanlah secara eksklusif berbicara teknologi saja, dan tidak hanya terbatas pada *web service*, namun dapat dianggap sebagai kombinasi dari consumer dan service yang berkolaborasi dan didukung oleh kumpulan

kaidah batasan, prinsip dan standar [5]. SOA memiliki 3 komponen dasar yaitu :

1. *Consumer*
2. *Service*
3. *Intermediaries*

Hubungan dari komponen ini dapat dilihat pada gambar berikut:



Gambar 2. Hubungan komponen dalam SOA [5]

SOA yang dikembangkan dengan baik harus memenuhi beberapa prinsip dasar, yaitu:

1. *Loose Coupling*
2. *Interoperability*
3. *Reusability*
4. *Discoverability*
5. *Governance*

Web Service

Web service merupakan salah satu implementasi teknologi SOA. Sebagai *intermediate* antara *consumer* dan *service*, digunakan teknologi akses web baik berupa HTTP, SMTP, FTP atau BEEP (*Blocks Extensible Exchange Protocol*). Arsitektur web service dapat dilihat dalam 4 *layer service protocol stack* [6].

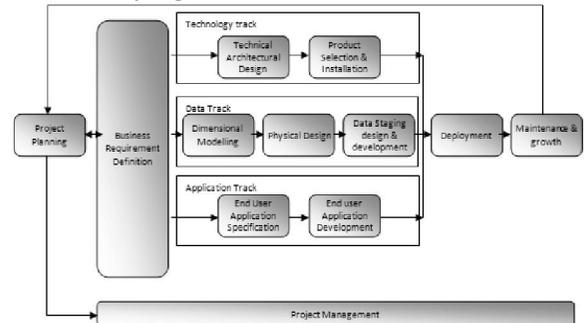
Discovery	UDDI
Description	WSDL
XML messaging	XML-RPC, SOAP, XML
Transport	HTTP, SMTP, FTP, BEEP

Gambar 3. *Web service protocol stack* [6]

Metodologi *Business Dimensional lifecycle*

Pengembangan *data warehouse* yang disesuaikan untuk kebutuhan bisnis dapat menggunakan pendekatan *business life cycle*. Salah satu pendekatan yang diusulkan oleh Ralph Kimball, ialah mengintegrasikan pengembangan dari 3 sudut pandang berbeda, yaitu teknologi, data dan aplikasi dari pendekatan bisnis.[7]. Diagram berikut menggambarkan urutan dari kegiatan yang

akan dilakukan untuk mendesain, membuat dan mengimplementasikan sebuah *data warehouse* yang efektif.



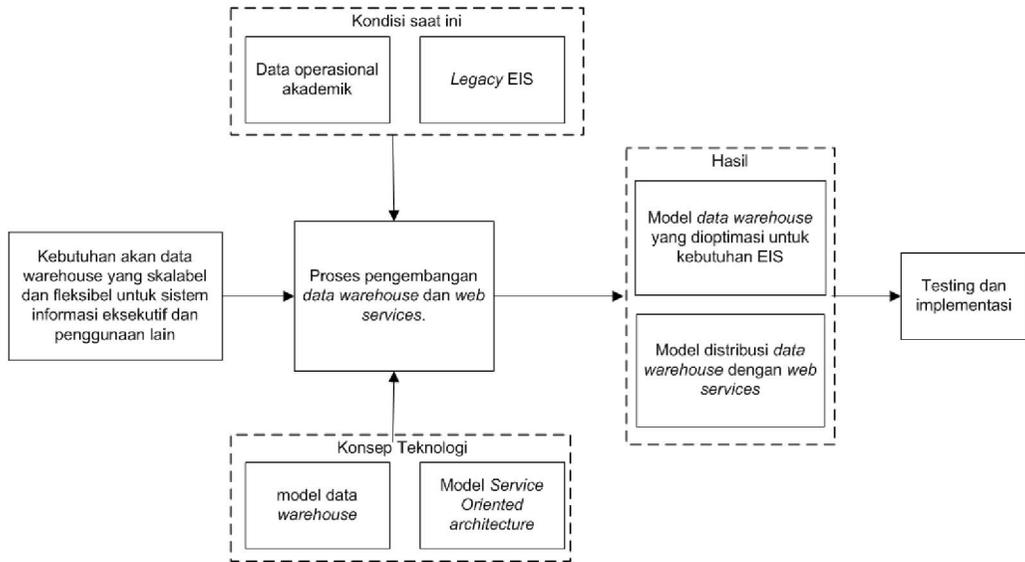
Gambar 4. *Business dimensional lifecycle*

Tinjauan obyek penelitian

Fakultas Teknologi Informasi (FTI) sebagai salah satu fakultas pada Universitas Budi Luhur merupakan fakultas yang lahir dari AMIK Budi Luhur semenjak 1979. Fakultas ini memiliki 5 program studi, yaitu Teknik Informatika, Sistem Informasi, Sistem Komputer, Manajemen Informatika (D3) dan Komputerisasi Akuntansi (D3). FTI telah melakukan komputerisasi dari seluruh proses bisnis akademiknya, diantaranya yaitu : penentuan mata kuliah, penentuan dosen pengajar, kartu rencana studi, penilaian, hasil studi kumulatif, bimbingan PA, penjadwalan, dan *assessment* dosen. Semua data operasional telah dimanajemen dengan DBMS Oracle 6.

Kerangka konsep

Sesuai dengan tujuan penelitian yang telah ditetapkan, peneliti ingin melakukan analisa dan pengembangan dari sistem *data warehouse* yang ditujukan sebagai sumber data bagi sistem informasi eksekutif. Model *data warehouse* yang digunakan ialah model multi dimensi dengan teknik distribusi SOA. Hasil yang diharapkan ialah terimplementasi nya model *data warehouse* yang lebih efektif, efisien, fleksibel, dengan skalabilitas dan keamanan tinggi dalam memenuhi kebutuhan analisa data eksekutif. Kerangka konsep di atas dapat digambarkan dengan diagram sebagai berikut :



Gambar 5. Kerangka konsep

Kerangka konsep diatas dilandasi atas kebutuhan akan *data warehouse* yang akan digunakan sebagai sumber data untuk sistem informasi eksekutif dan penggunaan lain seperti informasi akademik untuk mahasiswa. Model *data warehouse* yang dimaksud harus skalabel (mudah dikembangkan dan diperluas) serta memudahkan pihak pengembang aplikasi. Berdasarkan kondisi riil, dimana data operasional telah disimpan dalam database, dan ketersediaan *legacy EIS*, penelitian ini akan mengkorporasikan teknologi *data warehouse* dan model *service oriented architecture* berupa *web service* dalam proses pengembangan prototipe.

Model yang dimaksud merupakan sebuah *data warehouse* yang strukturnya dikembangkan secara spesifik untuk keperluan analisa. model tersebut akan menggunakan bentuk distribusi *web service* yang akan meningkatkan skalabilitas dan fleksibilitas penyebaran informasi.

Hasil akhir berupa prototipe model *data warehouse* dengan metode distribusi *web service* akan melalui tahap *testing* (*user acceptance*) dan implementasi untuk digunakan.

Hipotesis

Berdasarkan kajian teoritis dan kerangka berpikir yang telah dikemukakan peneliti menduga akan hasil yang didapat dari penelitian, yaitu:

Hipotesis pertama:

H0: Diduga model *data warehouse dimensional* dengan *star join approach* merupakan struktur yang sesuai untuk penelitian ini.

H1: model *data warehouse star join* memiliki kelebihan dalam respon query yang lebih baik dan berorientasi pada kebutuhan informasi.

Hipotesis kedua:

H0: Diduga penggunaan *Service Oriented Architecture* merupakan model distribusi *data warehouse* yang meningkatkan skalabilitas dan fleksibilitas pemanfaatan *data warehouse*.

H1: model *data warehouse* ini memanfaatkan *middleware* berupa *web service* sebagai penengah antara sumber data dengan aplikasi pengguna, sehingga tidak ada kontak langsung dengan sumber data. Ini berarti aplikasi pengguna tidak perlu mengetahui arsitektur *warehouse* yang digunakan. Perubahan teknologi atau arsitektur *data warehouse* tidak

mempengaruhi aplikasi. Format data yang standar (XML) mempermudah desain dan pengembangan aplikasi yang akan mengakses *data warehouse*.

3. Metodologi Penelitian

Metode Penelitian

Penelitian ini menggunakan metode penelitian Deskriptif Kualitatif yang bertujuan untuk mendapatkan gambaran lengkap dari proses dan subyek penelitian. Pelaksanaannya menggunakan pengamatan dan wawancara yang bersifat analitis deduktif dari peneliti untuk mengumpulkan data yang lengkap sebagai dasar pengembangan hipotesis. ([8], dengan perubahan).

Untuk menguji fungsionalitas dari prototipe, digunakan uji *white box testing* dan *feedback* dari *user*.

Metode Pengumpulan Data

Data untuk penelitian ini berasal dari data primer dan sekunder. Data primer didapat dari wawancara, observasi lapangan, dan uji coba. Data sekunder didapatkan dari data obyek penelitian (*database*), studi literatur dan tulisan ilmiah tentang *data warehouse* dan SOA.

Wawancara dan tes kesiapan

Wawancara dilakukan dengan beberapa *key person* yang terkait dengan ruang lingkup penelitian. Tujuan wawancara ini untuk mendapatkan gambaran kondisi *real*, *requirement* dan harapan pengguna. Pihak yang terkait yaitu :

1. Pihak manajemen eksekutif dan operasional. Termasuk dalam kategori ini ialah Dekan Fakultas Teknologi Informasi, dan ketua program studi.
2. Pihak pengembang Sistem informasi. Termasuk dalam kategori ini ialah Biro Sistem Informasi (BSI) Yayasan Budi Luhur, yaitu Kabag. BSI dan Kasubag. Pengelolaan Data.

Observasi lapangan

Pada observasi lapangan, peneliti melakukan analisa terhadap *best practices*

yang dilakukan di lapangan untuk mencapai tujuan.

Studi Literatur

Studi literatur dilakukan peneliti dengan mempelajari berbagai referensi dan tulisan ilmiah tentang *data warehouse* dan SOA.

Instrumentasi

Instrumentasi utama dari penelitian deskriptif kualitatif ialah sang peneliti sendiri. Dalam hal ini peneliti akan melakukan langkah-langkah berikut:

1. Wawancara dengan semua *key person* terkait.
2. Identifikasi sistem terkomputerisasi yang tersedia, yaitu Infrastruktur, sistem informasi eksekutif, dan sistem database operasional.

Untuk pelaksanaan pengembangan *prototype* dan *testing*, akan digunakan alat bantu dan komponen berupa DBMS MySQL, Apache, SqlYog, WSF/PHP, Ms Office, Toad, Oracle *server*, SQL architect, Mondrian-Wabbit dan Geany.

Teknik pengembangan

Teknik pengembangan dilandasi metodologi *Business Life Cycle* dengan urutan sebagai berikut.

1. Perencanaan Proyek *data warehouse* (*Project planning*)
2. Mendefinisikan kebutuhan (*business requirement definition*)

Kebutuhan yang telah dikumpulkan kemudian diterjemahkan ke dalam suatu rancangan.

Kebutuhan bisnis tersebut dapat dipisahkan menjadi 3 bagian yang dapat dikerjakan secara paralel, yaitu:

1. Teknologi yang akan digunakan.
2. Data yang akan disimpan.
3. Aplikasi *end user* yang menggunakan.

Technology track : Technical architecture design

Tahap ini mengacu pada proses desain arsitektur *data warehouse* yang meliputi 3 faktor utama yaitu analisa kebutuhan arsitektur, arsitektur yang sedang berjalan

dan arah pengembangan arsitektur masa depan.

Technology track: product selection and installation.

Setelah didapatkan desain arsitektur, selanjutnya ditentukan komponen arsitektur seperti : spesifikasi hardware, DBMS, tool data staging atau *tool* akses data. Setelah produk dipilih, dapat dilakukan instalasi dan testing untuk memastikan terintegrasinya *data warehouse*.

Data track : Dimensional modelling

Model data ini berorientasi pada bentuk model yang telah dioptimasi untuk sebuah kepentingan bisnis yang spesifik.

Data track : Physical design

Mendesain database secara fisik difokuskan pada definisi struktur fisik untuk mendukung desain logical database. Pada tahap ini, ditentukan definisi standar penamaan dan *setting* untuk lingkungan *database*. Tabel dan relasi akan dibuat sesuai hasil *dimensional modeling*.

Data Track : data staging and development

Proses data staging mempunyai 3 langkah yang sering juga disebut dengan ETL yaitu : *Extract, Transformation, Loading*. Pada proses ini sumber data akan dikumpulkan, lalu diubah agar sesuai dengan format *data warehouse*, lalu di masukkan ke dalam *data warehouse*.

Application track: end user application specification

Spesifikasi kerja dari aplikasi yang memanfaatkan *data warehouse* akan dibuat. Pada penelitian ini, aplikasi yang dibuat bukanlah aplikasi *end user*, namun aplikasi *web service* yang akan bertindak sebagai *middleware*. Spesifikasi *web service* akan menjelaskan bentuk komunikasi, *template* pengiriman data, dan *parameter* yang digunakan untuk interaksi. Spesifikasi ini dibuat dengan tujuan untuk menyamakan persepsi tim pengembang *end user application* dengan pengembang *web service*.

Application track: end user application development

Tahapan selanjutnya adalah membangun aplikasi *web service* berdasarkan spesifikasi yang telah dibuat pada tahap sebelumnya. Pada tahap ini, akan dimanfaatkan COTS (*Component OF The Shelf*). COTS mengacu pada produk software atau hardware yang telah ada dan tersedia untuk digunakan pada aplikasi yang dikembangkan ([Webopedia 2001], dengan perubahan). Pada penelitian ini akan dimanfaatkan komponen berupa WSF/PHP (*Web Service Framework/PHP*) yang bertindak sebagai *framework* dasar untuk *web service*.

Deployment

Tahapan ini menggabungkan teknologi , data dan aplikasi yang dapat diakses oleh *end user*. Pada penelitian ini dilakukan implementasi prototipe yang bisa dikembangkan lebih lanjut.

Maintenance and growth

Agar fungsi *data warehouse* dapat digunakan *user*, perlu diberikan pelatihan. Untuk menjamin *data warehouse* tetap optimal, performansi dari *data warehouse* harus terus diukur, dan *maintenance* fisik maupun data dapat dilakukan secara berkala.

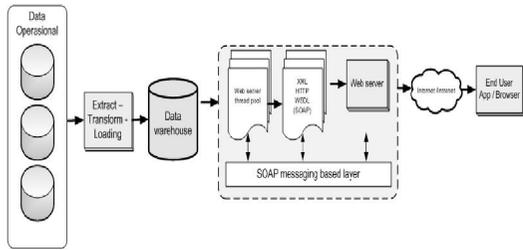
Project management

Untuk menjamin tahapan pelaksanaan *lifecycle* ini berlangsung optimal, *project management* diterapkan di keseluruhan proses. Termasuk proses *monitoring* dan komunikasi yang memungkinkan *feedback* dan *capture requirement* yang lebih cepat.

Arsitektur sistem dan testing

Arsitektur data

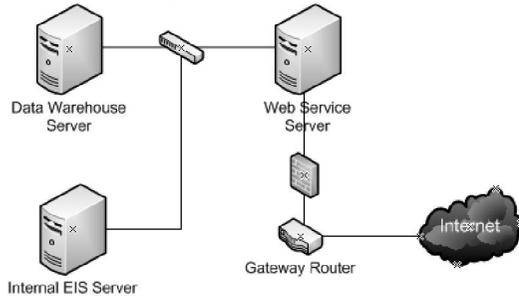
Arsitektur berikut menggambarkan stuktur, hubungan, dan arus data dalam sistem *data warehouse* yang akan dikembangkan.



Gambar 6. Rancangan arsitektur data

Arsitektur Infrastruktur

Arsitektur berikut menggambarkan hubungan infrastruktur untuk sistem *data warehouse* yang akan dikembangkan.



Gambar 7. Rancangan arsitektur infrastruktur.

Testing

Testing yang akan dilakukan ialah *white box testing*. Teknik ini mengacu pada pelaksanaan uji coba sistem dengan pengetahuan internal dari sistem itu sendiri. Analisa akan dilakukan dengan melakukan pemeriksaan code, struktur, dan *test case*.

Testing struktur *data warehouse* akan memanfaatkan Mondrian-Wabbit, sedangkan testing arsitektur distribusi data akan dilakukan dengan pemanggilan fungsi *web service* dari script PHP dan aplikasi vb.net.

4. Rancang Bangun Dan Pengujian Model

Project Management and requirement

Data warehouse yang dikembangkan adalah *data warehouse* yang dioptimasi untuk kepentingan evaluasi akademik Fakultas Teknologi Informasi. Fungsi evaluasi mencakup evaluasi nilai mahasiswa, absensi dan hasil studi yang dapat dianalisa

berdasarkan program studi, tahun ajaran, lokasi kuliah, dan angkatan mahasiswa.

Analisa kebutuhan

Berdasarkan pengumpulan data yang dilakukan, dilakukan analisa kebutuhan (*requirement definition*) yaitu:

1. Dibutuhkan sebuah model data warehouse yang mampu digunakan oleh EIS maupun digunakan untuk pemanfaatan non transaksional lainnya.
2. Dibutuhkan sebuah metode distribusi yang bersifat independen dan dapat digunakan oleh berbagai jenis aplikasi yang mungkin digunakan, baik berupa *web* maupun *desktop*.
3. Dibutuhkan sumber data yang mampu menampilkan laporan dan analisa berikut:
 - a) Informasi perkuliahan yang diambil oleh mahasiswa, berupa data nilai dan data absensi.
 - b) Informasi persebaran dan rata-rata prestasi akademik yang dapat dikategorikan berdasarkan program studi, lokasi, dan periode perkuliahan.
 - c) Informasi penunjang seperti persebaran jumlah mahasiswa dan jumlah pelaksanaan perkuliahan yang bisa dikategorikan berdasarkan program studi, lokasi, dan periode perkuliahan.

Permodelan Data Warehouse Dimensional

Sebagai sumber data untuk *data warehouse*, digunakan sumber data operasional yang menyimpan data harian dari sistem akademik. Database harian yang digunakan ialah oracle DBMS 6 yang menyimpan data – data akademik yaitu: mahasiswa, rencana studi, nilai, absensi, mata kuliah, jadwal, kurikulum, pembimbing akademik, dan lainnya. Dari kumpulan data tersebut hanya akan dipilih data yang sesuai dengan informasi yang ingin didapat.

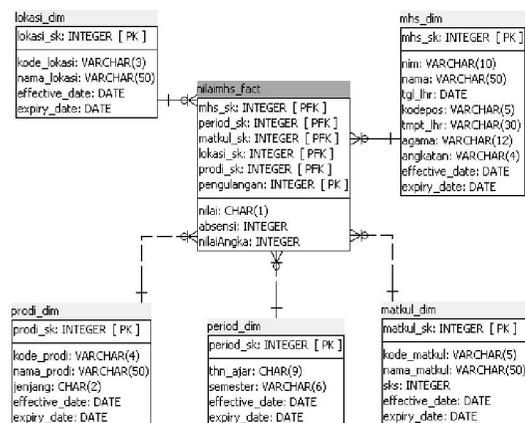
Sesuai dengan hasil analisa kebutuhan, disimpulkan bahwa model *data warehouse* yang dikembangkan harus memiliki data berikut:

1. Mahasiswa
2. Program studi

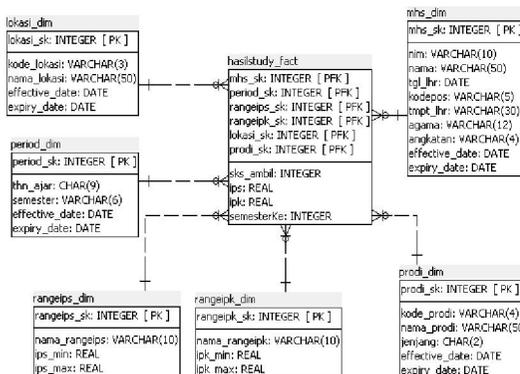
3. Lokasi
4. Tahun ajaran dan semester
5. Mata kuliah
6. Nilai mahasiswa atomik (tiap mata kuliah)
7. Nilai mahasiswa agregat (tiap semester)

Untuk model skema yang digunakan ialah *star schema*, dimana satu tabel fakta dikelilingi oleh beberapa tabel dimensi. Penggunaan skema ini dilandasi atas kemudahan *query* dan akses terhadap tabel dimensi yang lebih mudah.

Sesuai dengan hasil analisa yang dilakukan, didapatkan 2 star model dengan 2 tabel fact dan beberapa tabel dimensi. Star schema berikut merupakan model yang dioptimasi untuk penampilan nilai mahasiswa dan absensi mahasiswa (data atomik) dan korelasinya dengan periode, program studi dan lokasi.



Gambar 8. Star Schema Nilai Mahasiswa



Gambar 9. Star Schema hasil study

Berikut ialah penjelasan dari tabel yang terbentuk.

1. Nilaimhs_fact

Tabel ini berisi semua data yang berhubungan dengan nilai dan absensi dari mahasiswa untuk tiap mata kuliah, juga data pengambilan ulang (jika mahasiswa mengambil mata kuliah tersebut lebih dari sekali). Analisa pengukuran dasar yang bisa dilakukan dengan OLAP pada tabel ini ialah rata-rata nilai absensi dan nilai angka (basis 4).

2. Hasilstudy_fact

Tabel ini berisi semua data yang berhubungan dengan hasil study mahasiswa tiap semester, yaitu sks yang diambil, ips, ipk, dan posisi semester. Analisa pengukuran yang bisa dilakukan OLAP pada tabel ini mencakup rata-rata ips, ipk, dan jumlah sks ambil.

3. Mhs_dim

Pada tabel dimensi ini, data yang termasuk ialah nim, nama, tanggal lahir, tempat lahir, kodepos, agama dan angkatan masuk. Umumnya field ini dapat dipakai untuk mengkategorikan data secara demografi (kodepos).

4. Prodi_dim

Pada tabel dimensi ini, data yang termasuk ialah data nama program studi dan jenjang.

5. Matkul_dim

Pada tabel dimensi ini, data yang termasuk ialah data kode mata kuliah, nama mata kuliah, dan besar sks.

6. Period_dim

Pada tabel dimensi ini, data yang termasuk ialah data tahun ajaran, dan semester.

7. Lokasi_dim

Pada tabel dimensi ini, data yang termasuk ialah nama lokasi. Data lokasi tidak digabung pada mahasiswa, karena mahasiswa dapat berpindah kuliah ke cabang selama masa studinya.

8. Rangeips_dim

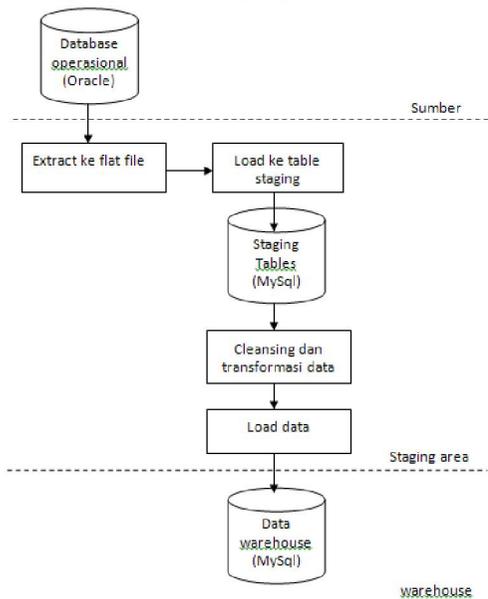
Pada tabel dimensi ini, data yang termasuk ialah kategori kelompok ips yang dapat digunakan sebagai filter pencarian.

9. Rangeipk_dim

Pada tabel dimensi ini, data yang termasuk ialah kategori kelompok IPK yang dapat digunakan sebagai filter pencarian.

Proses ETL Data Warehouse

Data warehouse yang telah dikembangkan akan dipopulasi dengan data. Untuk keperluan penelitian ini, akan dilakukan populasi dengan sample data akademik dari mahasiswa Fakultas Teknologi Informasi angkatan 2005 dan 2006. Berikut ialah kerangka proses ETL yang dilakukan:



Gambar 10. Proses ETL data warehouse

Testing dan representasi data warehouse

Data warehouse dalam penelitian ini dikembangkan dengan orientasi untuk kepentingan sistem informasi eksekutif, namun tetap dapat digunakan sebagai sumber data untuk OLAP. Contoh representasi data berikut merupakan

sebagian dari jenis informasi yang dapat dihadirkan.

1. Informasi dapat dimunculkan untuk tahun ajaran dan semester tertentu.

period_dim	Measures	
	rata_absen	rata_nilai
All period_dim.period_hirarkis	91.281	2.641
+ 20052006	91.295	2.611
+ 20062007	91.555	2.557
+ 20072008	93.1	2.734
+ 20082009	89.701	2.754
+ 20092010	81.76	2.058

Gambar 11. Informasi nilai mahasiswa berdasarkan period

2. Informasi rata –rata IPS dan IPK dengan filter periode.

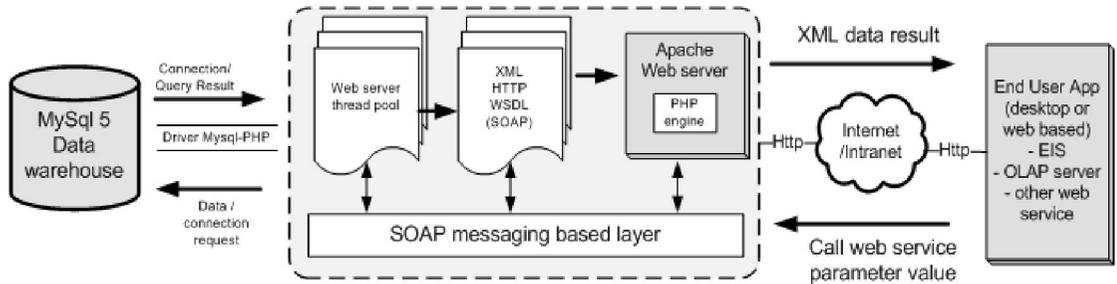
prodi_dim	Measures		
	rata_ips	rata_ipk	rata_sksAmbil
D3	3.316	3.037	3.229
KOMPUTERISASI AKUNTANSI-Diploma 3	3.379	3.17	2.889
MANAJEMEN INFORMATIKA-Diploma 3	3.248	2.897	3.588
SISTEM KOMPUTER-Diploma 3			
S1	3.017	2.917	22.656
KOMPUTERISASI AKUNTANSI-Strata 1	3.377	3.211	23.366
SISTEM INFORMASI-Strata 1	2.895	2.782	22.567
SISTEM KOMPUTER-Strata 1	2.513	2.622	19.261
TEKNIK INFORMATIKA-Strata 1	3.085	3.005	22.746

hirarki > semester > All period dim.period hirarkis > 20072008 > Genap

Gambar 12. Informasi hasil study berdasarkan prodi dan periode

Pengembangan Model Distribusi Web Service.

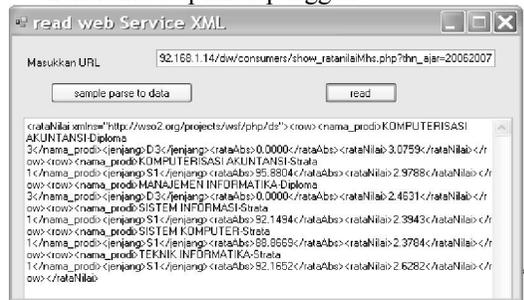
Sesuai dengan rancangan sistem yang telah dinyatakan pada bab III, model distribusi data warehouse yang akan dikembangkan berbasis web service dengan framework WSF/PHP. Pola kerja arsitektur dapat dilihat pada gambar berikut:



Gambar 13. Arsitektur *web service* dan *data warehouse*

Web service dikembangkan dalam bentuk file PHP dengan format yang disesuaikan dengan *framework*. Untuk keperluan prototipe ini, dikembangkan contoh 6 *service*. Jumlah dan isi *service* ditentukan oleh kebutuhan informasi yang dibutuhkan oleh EIS yang dapat ditambahkan kemudian. *Web service* pada penelitian ini dapat diakses menggunakan SOAP (*Simple Object Access Protocol*) dengan *transport layer* http. Tiap client yang mengakses harus membuat instance dari kelas WSCClient dan mengirim XML *format request*. Untuk dapat digunakan aplikasi desktop, sebuah consumer bertindak sebagai *intermediate* untuk mengembalikan format XML.

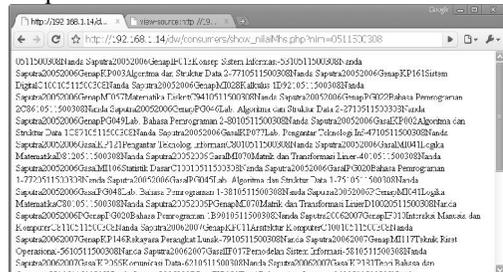
desktop (VB.NET) untuk menunjukkan fleksibilitas aplikasi pengguna.



Gambar 15. pemanfaatan *web service* menggunakan aplikasi *desktop*

Pengujian hasil proses *web service*

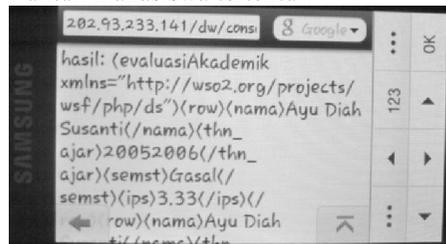
1. Daftar nilai dan absensi mahasiswa berdasarkan nim Menampilkan data nilai dan absensi dari tiap mata kuliah yang pernah diambil oleh mahasiswa tersebut.



Gambar 14. Tampilan data hasil pemanggilan *web service*

2. Rata-rata nilai Mahasiswa per program studi berdasarkan periode. Menampilkan rata – rata nilai yang diagregasi berdasarkan program studi. Contoh berikut menggunakan program

3. Data IPS per semester untuk tiap mahasiswa. Menampilkan data IPS tiap semester untuk mahasiswa tertentu



Gambar 16. data IPS per semester untuk tiap mahasiswa diakses mobile

Implikasi Penelitian

Aspek manajerial

Dari sisi manajerial, pemanfaatan data warehouse dengan distribusi *web service* memberikan beberapa perubahan pada proses bisnis analisa data. Pihak manajemen dapat langsung menganalisa informasi yang disediakan *data warehouse*, ini memberikan peningkatan kualitas informasi yang dijadikan basis pengambilan keputusan.

Pihak manajemen juga dapat mengurangi campur tangan pengembang EIS dari akses skema data yang tidak berhak. Tanggung jawab pengembangan dan *web service* menjadi terpisah dari EIS.

Aspek sistem

Dari sisi sistem, keuntungan utama ialah adanya *data warehouse* yang terpisah dari data operasional, sehingga meningkatkan kesederhanaan proses *retrieve* data dan peningkatan kinerja sistem. Skalabilitas dari sistem menjadi lebih baik, karena jika ada perubahan teknologi atau perubahan struktur pada sisi *data warehouse*, tidak akan mempengaruhi aplikasi pengguna (tidak perlu dilakukan perubahan).

Penggunaan *web service* sebagai *intermediate* antara *data warehouse* dan aplikasi pengguna meningkatkan modularitas dan fleksibilitas dari data yang disajikan. Aplikasi yang memanfaatkan data tidak perlu terkoneksi langsung dengan database, dan tidak terikat pada bentuk *legacy* dari koneksi yang ada.

Jenis aplikasi yang memanfaatkan data tidak tertutup hanya pada EIS, namun pada aplikasi lain yang perlu menampilkan informasi yang terkait (misalnya informasi HSK *online* untuk mahasiswa). Lebih lanjut, penggunaan *web service* yang bertindak sebagai *logic layer* (atau *business layer*) memudahkan pengembangan aplikasi, karena aplikasi hanya perlu memanggil *service* yang telah disediakan.

Aturan dan ketentuan untuk mengakses data menjadi berubah, dimana aplikasi EIS tidak boleh mengakses *data warehouse* secara langsung.

Penelitian lanjut

Penelitian lanjut dari penelitian ini dapat diteruskan dari sisi pengembangan struktur *data warehouse*, atau pengembangan *web service* untuk *distributed data warehouse*, dimana *web service* bertindak sebagai *engine* untuk *Extract*, *Transform*, dan *Loading*, dan distribusi dari sumber *data warehouse* yang kompleks dan berbeda lokasi.

Penutup

Kesimpulan

Berdasarkan penelitian pengembangan yang telah dilakukan, dapat disimpulkan bahwa sebuah *data warehouse* yang dikembangkan secara khusus untuk kepentingan analisa akan memberikan beberapa kelebihan berikut:

1. Model *data warehouse* dengan *star join approach* merupakan struktur data warehouse yang relatif mudah diadaptasi sesuai dengan kebutuhan analisa data.
2. Peningkatan skalabilitas dan fleksibilitas. Dengan menggunakan *web service* sebagai perantara antara *data warehouse* dengan aplikasi pengguna, terjadi peningkatan skalabilitas dan fleksibilitas (Pengembang EIS tidak perlu memiliki akses langsung *data warehouse*).
3. Penyusunan laporan yang lebih efektif. Dengan adanya *data warehouse*, pihak pengembang EIS tidak perlu mengakses langsung data operasional dan menarik data dari puluhan tabel relasional.
4. Memudahkan penemuan “pola” baru. Dengan model *data warehouse* dimensional, *end user* dapat dimudahkan untuk menemukan pendekatan baru melakukan analisis yang sebelumnya tidak mudah ditemukan jika data masih dalam bentuk relasi operasional.

Saran

Setelah melaksanakan penelitian, peneliti memiliki beberapa saran yang berkaitan dengan sisi manajemen, sistem, dan studi lanjutan.

1. Dari sisi manajemen, *data warehouse* yang terbentuk dapat di pergunakan secara optimal bila ada maintenance dan pengembangan, sesuai dengan siklus hidup sistem, oleh karena itu diperlukan susunan dan pembagian kewenangan yang jelas untuk tugas itu.
2. Dari sisi sistem, untuk meningkatkan kualitas dari kebijakan pihak eksekutif, model *data warehouse* yang ada dapat dikembangkan, baik dari sisi jumlah laporan, maupun penambahan *star join* baru yang dapat mengikut sertakan aspek-aspek tambahan dalam evaluasi pendidikan.

3. Peningkatan infrastruktur sistem, khususnya *bandwith* jaringan, karena kelemahan utama dari pemanfaatan SOA ialah meningkatnya waktu *delay* antara permintaan data dengan penerimaan data. Hal ini memang wajar mengingat sifat kerjanya sebagai penghubung antara data dan aplikasi.
4. Dilakukan penelitian lanjutan yang berkaitan dengan peningkatan performa dan fungsi dari *data warehouse* dengan *web service* sebagai *engine* ETL maupun sebagai bentuk distribusi data.

Daftar Pustaka

- [1] Dictionary.com, "*executive* in The American Heritage® Dictionary of the English Language, Fourth Edition", 2004, <http://dictionary.reference.com/browse/executive> (diakses 10 Januari 2010).
- [2] Britannica.com "*Executive Information System*", 2010, <http://www.britannica.com/EBchecked/topic/287895/information-system/218062/Executive-information-systems>. (diakses 12 Januari 2010).
- [3] Immon, Bill, "*Building The Data warehouse*", Wiley Computer publishing, 4th edition, New York, 2005.
- [4] Vamsi, Khrisna, "*Data warehouse explained*", 2008, <http://thepcweb.com/?p=232> (diakses 13 Januari 2010).
- [5] Bean, "SOA and Web Services Interface Design", Elsevier Publishing, Massachusetts, 2010.
- [6] Cerami, Ethan, "Web Service Essentials: Distributed Application with XML-RPC, SOAP, UDDI & WSDL", Wiley Publishing, 2002.
- [7] Kimball, Ralph., *The Data warehouse Lifecycle Toolkit*, Wiley Publishing, USA, 1998.
- [8] Key, James.P, "Qualitative Research", Source location <http://www.okstate.edu/ag/agedcm4h/academic/aged5980a/5980/newpage21.htm>. (Diakses: 24 Januari 2010).
- [9] Darmawikarta, Djoni., "Dimensional Data Warehouseing with Mysql: A Tutorial", Brainy Software Corp., 2007.
- [10] Dictionary.com, "*business intelligence* in *Dictionary.com's 21st Century Lexicon*", 2010, http://dictionary.reference.com/browse/business_intelligence(diakses 10 Januari 2010).
- [11] Josuttis, Nicolai.M., "SOA In Practice: The Art Of Distributes System Design", O'Reilly, California, 2007.
- [12] Mulyati, Sri "Pembangunan Model *Data Warehouse* untuk Menunjang Sistem Informasi Eksekutif Studi Kasus : Penerimaan Mahasiswa Baru Pada Universitas Budi Luhur", tesis, MKOM UBL, Jakarta, 2009.
- [13] Sucahyo, Yudho G., Aximah, Ariana., 2007, "Implementasi Data Warehouse untuk menunjang kegiatan akademik", Seminar nasional sistem dan informatika, Bali 2007.
- [14] Webopedia.com, "COTS", 2001, <http://www.webopedia.com/TERM/C/COTS.html>(diakses 2 Februari 2010).
- [15] Zhong, Luo., et all., 2005. "The Data Warehouse Model Based on Web Service Technology", *Journal of Communication and Computer*, ISSN1548-7709, Volume 2-1:26-31,USA.