

Integrasi Enterprise (Studi Kasus: Yayasan Pendidikan “X”)

Tanti Kristanti

Jurusan Teknik Informatika, Fakultas Teknologi Informasi
Universitas Kristen Maranatha
Jl. Surya Sumantri No. 65, Bandung 40164
e-mail: tantikristanti02@yahoo.com

Abstract

Ownership of information system at one particular company is expected to support their totally business process, which is not limited to just certain business function but can give benefit to a number of functional areas. This tendency claims to the requirements to integrate the function oriented system in order to make them in line with business process. Yayasan Pendidikan “X” as educational institution has also a number of business function supported by information system. But in its growth, management of the system is conducted independently by a number of organizational unit regardless of business requirement so that emerging “islands” of information system as impact from the system that having stovepipe character and unable to communicate one with another. Enterprise integration is the answer for the problems resulted from the system that has differ platform which cannot give optimal benefit to company.

To be able to obtain holistic benefit for business, hence integration project will preceded by analysis activities to capture enterprise present condition. Analysis is aimed to understand “X” current condition (as-is), started with business modeling, defines what is in place today for application system and supporting technology that result information resource catalog. Business modeling by using Porter Chain Value as well as Four Stage Life Cycle Business System Planning is expected to understand functional areas owned by “X” so that yielded architecture that is not impressionable by internal and external changing environment. Information resource catalog analysis constitute integration project that can be manage in an optimal fashion without depending on certain technology. Pursuant to result of analysis from current condition, hence major kinds of data, technology and application for future needs can be determined (to-be). As-is analysis and to-be definition will result some gaps that constituting policies for integration project.

Integration is also triggered by business drivers and requirements. The business drivers and requirements which are triggering the needing of integration project in “X” are the needs to increase business efficiency and competitiveness and to improve customer satisfaction for both internal and external customer of “X”. Integration architecture development covers technical, service, information and business process integration. The results from this architecture then are used as basis for integration implementation strategy.

Keywords: enterprise integration, enterprise architecture planning.

1. Pendahuluan

Informasi telah menjadi salah satu aset yang penting dalam perusahaan, oleh karenanya perusahaan melakukan berbagai upaya untuk dapat mengumpulkan serta mengolah data menjadi informasi yang bermanfaat bagi kegiatan bisnisnya. Salah satu upaya tersebut adalah dengan membangun sistem informasi yang didukung

oleh teknologi informasi. Namun, perusahaan pada umumnya membangun sistem informasi/teknologi informasi (SI/TI) hanya untuk mendukung fungsi bisnis tertentu, sedangkan suatu perusahaan terdiri atas banyak fungsi bisnis, yang masing-masing fungsi tersebut juga ternyata memerlukan dukungan SI/TI. Tidak jarang, sistem informasi yang dibangun tidak selaras dengan tujuan bisnis secara menyeluruh karena sistem informasi dalam perusahaan seringkali hanya dipandang sekedar pengadaan teknologi yang mampu memenuhi kebutuhan jangka pendek dan untuk mengikuti *trend* teknologi. Pengembangan sistem yang hanya memperhatikan kebutuhan fungsi tertentu untuk jangka waktu tertentu berakibat pada kepemilikan sejumlah sistem dengan *platform* yang berbeda-beda baik dari sisi perangkat keras maupun perangkat lunak serta hanya mampu menunjang area fungsional tertentu.

Kondisi bisnis yang kompetitif saat ini memerlukan SI/TI yang dapat mendukung proses bisnis suatu perusahaan secara menyeluruh, yang bukan terbatas pada fungsi bisnis tertentu saja namun dapat bermanfaat bagi sejumlah area fungsional (*across functional areas*). Kecenderungan ini menuntut pada kebutuhan untuk melakukan integrasi terhadap sistem yang berorientasi fungsi agar dapat sejalan dengan proses bisnis. Kebutuhan akan integrasi perusahaan/*enterprise integration (EI)* dikendalikan oleh sejumlah faktor kunci. Pertama, tekanan lingkungan bisnis yang kompetitif yang mengendalikan para manajemen SI/TI untuk dapat memperpendek siklus hidup pengembangan aplikasi dengan cara guna ulang (*reuse*) layanan aplikasi dan informasi yang telah ada dan bukan dengan menciptakan proses bisnis, layanan aplikasi serta penyimpanan data yang sama secara berulang-ulang. Kedua, integrasi aplikasi menyediakan manfaat kompetitif bagi perusahaan yang ingin saling berbagi informasi aplikasi baik di dalam perusahaan maupun dengan pihak di luar perusahaan [9].

Berdasarkan faktor-faktor kunci itulah, perusahaan mulai melirik integrasi sebagai solusi atas permasalahan seputar sistem yang mereka miliki. Namun, setiap perusahaan memiliki berbagai kepentingan yang terkait dengan integrasi. Perusahaan yang memerlukan integrasi internal terhadap berbagai sistem yang mendukung area-area fungsional yang berbeda dari suatu bisnis akan melakukan integrasi intra-organisasi secara horizontal. Perusahaan yang memerlukan integrasi antara sistem pada tingkatan kontrol dan manajerial yang berbeda dari suatu organisasi akan melakukan integrasi intra-organisasi secara vertikal. Sedangkan perusahaan yang memerlukan integrasi antar sistem dengan perusahaan lain akan melakukan integrasi inter-organisasi.

Kecenderungan pengembangan sistem yang hanya diperuntukkan bagi fungsi bisnis tertentu juga terjadi di Yayasan Pendidikan “X”. “X” sebagai lembaga yang bergerak dalam bidang pendidikan memiliki sejumlah fungsi bisnis yang ditunjang oleh SI/TI. Namun dalam perkembangannya, pengelolaan SI/TI ini dilakukan secara independen oleh masing-masing unit organisasi. Setiap unit organisasi melakukan pembelian perangkat keras dan pengembangan perangkat lunak pada waktu yang berbeda dan dari vendor yang berbeda. Sebagai contoh adalah aplikasi-aplikasi yang mendukung fungsi akademik dan keuangan yang dibuat oleh sejumlah vendor dan dalam waktu yang berlainan yaitu berkisar dari tahun 1999 sampai dengan tahun 2006. Perbedaan pengembang dan tahun pembuatan jelas berdampak pada kepemilikan sejumlah sistem dengan *platform* yang berbeda-beda,

mulai dari perbedaan teknologi perangkat keras, bahasa pemrograman, sistem pengelola basis data, sistem operasi sampai dengan sistem aplikasi penunjang lainnya. "X" sebagai satu kesatuan perusahaan memerlukan integritas data dan informasi dari seluruh fungsi baik fungsi utama yaitu pendidikan maupun fungsi-fungsi pendukung. Namun integritas data dan informasi ini tidak dapat diperoleh melalui sistem yang dimiliki "X" saat ini karena adanya perbedaan *platform*. Perbedaan *platform* menyebabkan terjadinya "pulau-pulau" informasi karena setiap sistem tidak dapat berkomunikasi untuk saling berbagi pakai data dan informasi.

Enterprise integration merupakan jawaban terhadap permasalahan yang timbul sebagai akibat munculnya "pulau-pulau" informasi. Selama beberapa generasi, pengembangan sistem di "X" diarahkan hanya untuk melayani fungsi bisnis tertentu yang berakibat pada adanya "*stovepipe system*" di dalam perusahaan. Sistem yang dibangun secara *custom built* dengan menggunakan *data storage* serta teknologi pengembangan aplikasi yang tidak standar berdampak luas bagi "X". Sistem menjadi tidak mampu memberikan landasan bagi *business agility* dan tidak mampu menyesuaikan diri terhadap perubahan secara cepat. Penerapan SI/TI ternyata juga tidak dapat memenuhi kebutuhan bisnis (*business requirements*) secara maksimal dan hal ini baru disadari oleh "X" setelah sistem tersebut diimplementasikan dengan munculnya banyak keluhan terhadap ketidakmampuan sistem dalam memenuhi kebutuhan para pelaku organisasi.

Berdasarkan sejumlah permasalahan yang timbul di "X" saat ini sebagai akibat adanya "pulau-pulau" informasi/"*stovepipe system*" dan mengingat pentingnya integrasi sebagai solusi atas permasalahan tersebut, maka pembahasan tesis ini akan fokus pada pembentukan arsitektur integrasi sistem internal yang bersifat *enterprise-wide*. Pengembangan arsitektur akan menghasilkan cetak biru integrasi yang dapat mendukung proses bisnis secara menyeluruh.

2. Tinjauan Pustaka

2.1 Definisi Integrasi Enterprise

Bernstein dan Ruh mendefinisikan *enterprise integration* yaitu :

"Unrestricted sharing of information, services, and business processes among any connected applications or data sources in the enterprise." [4].

Hohpe, Gregor dalam bukunya *Enterprise Integration Patterns* mendefinisikan *enterprise integration* sebagai :

"Enterprise integration has to deal with multiple applications running on multiple platforms in different locations, making the term simple integration pretty much an oxymoron" [5].

Lam, Wing dalam tulisan yang berjudul *Technical Risk Management on Enterprise Integration Projects* mendefinisikan *enterprise integration* sebagai :

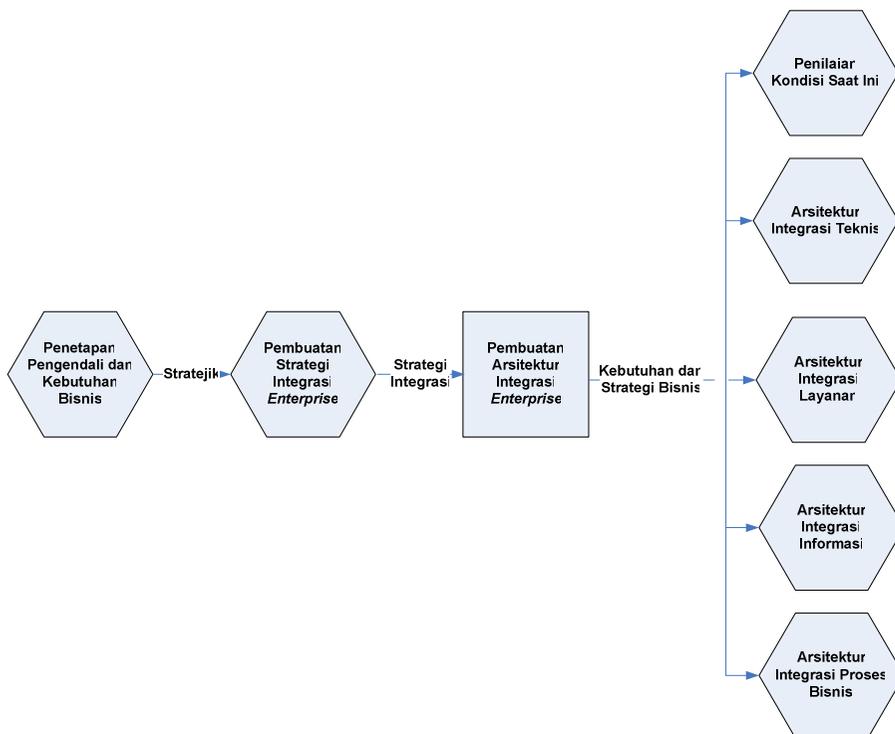
"A general term that refers to the integration of IT systems and business processes both within the enterprise and between different enterprises" [8].

Dari beberapa definisi di atas dapat ditarik kesimpulan bahwa *enterprise integration* (EI) merupakan tugas untuk membuat agar aplikasi-aplikasi yang bekerja pada berbagai *platform* di lokasi yang berbeda dapat bekerja sama guna

menghasilkan suatu kesatuan fungsionalitas, sehingga dapat saling berbagi informasi, layanan dan proses bisnis baik di dalam *enterprise* maupun antar *enterprise*.

2.2 Road Map Integrasi

Road map menjelaskan keseluruhan langkah yang menuntun kegiatan integrasi *enterprise*, secara lengkap dapat dilihat pada Gambar 1. Dari Gambar 1 dapat dilihat bahwa langkah pertama kegiatan integrasi *enterprise* adalah **penetapan pengendali dan kebutuhan bisnis**, langkah ini akan menentukan ruang lingkup integrasi. Langkah berikutnya setelah pendefinisian pengendali dan kebutuhan bisnis adalah **pembuatan strategi integrasi** dan **pembuatan arsitektur integrasi**.



Gambar 1 *Road Map* Integrasi [4]

2.2.1 Pengendali dan Kebutuhan Bisnis

Spesifikasi pengendali dan kebutuhan bisnis merupakan dokumentasi yang menggambarkan apa yang sedang bisnis coba untuk capai [4]. Spesifikasi ini akan menjadi tuntunan bagi proyek dan juga akan digunakan sampai sistem dapat beroperasi untuk menilai dampak pada bisnis.

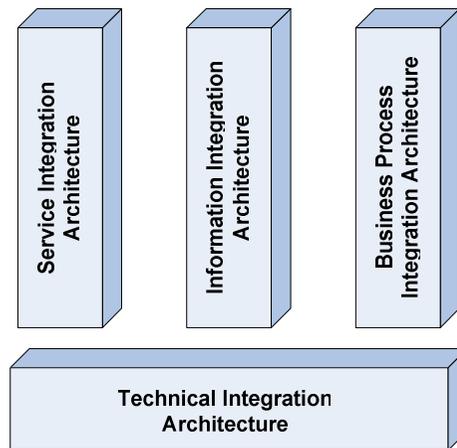
2.2.2 Strategi Integrasi *Enterprise*

Spesifikasi strategi integrasi bisnis merupakan dokumen yang memetakan kebutuhan, strategi dan inisiatif bisnis menjadi strategi dan proyek integrasi. Spesifikasi strategi integrasi dapat dibuat baik pada tingkatan *enterprise* maupun proyek [4].

2.2.3 Arsitektur Integrasi Enterprise

Arsitektur integrasi *enterprise* menyediakan *blueprint* untuk proyek integrasi baik strategik maupun taktis [4], menggambarkan keseluruhan komponen dari arsitektur. Arsitektur integrasi *enterprise* strategik meliputi tata kelola untuk memastikan bahwa proyek memenuhi standar-standar yang telah ditetapkan dan terdapat proses untuk pengecualian. Pendekatan-pendekatan taktis untuk mengembangkan infrastruktur teknis ternyata memiliki biaya pemeliharaan yang tinggi dan menghambat *business agility*. Dikarenakan hal tersebut, maka organisasi-organisasi besar dan juga lembaga-lembaga pemerintahan telah menetapkan *framework* arsitektur *enterprise* (*enterprise architecture/EA*). Arsitektur integrasi *enterprise* haruslah cocok dengan seluruh *framework* arsitektur *enterprise*. Prioritas dari pengembangan arsitektur dikendalikan oleh kebutuhan dan strategi bisnis.

Gambar 2 menunjukkan empat domain arsitektur yaitu arsitektur integrasi teknis, arsitektur integrasi layanan, arsitektur integrasi informasi dan arsitektur integrasi proses bisnis.



Gambar 2 Domain Arsitektur Integrasi Enterprise [4]

Penjelasan untuk masing-masing domain arsitektur integrasi *enterprise* adalah :

1. Arsitektur integrasi teknis
Arsitektur integrasi teknis mendefinisikan teknologi untuk seluruh solusi integrasi. Domain ini menjadi dasar guna mendukung komponen arsitektur integrasi *enterprise* yang lain.
2. Arsitektur integrasi layanan
Arsitektur integrasi layanan merupakan *subset* dari arsitektur aplikasi *enterprise*. Didefinisikan sebagai *loosely coupled, reusable business services*, arsitektur aplikasi ini paling fleksibel dan dapat beradaptasi terhadap perubahan bisnis, sehingga memungkinkan integrasi aplikasi secara cepat.
3. Arsitektur integrasi informasi
Arsitektur integrasi informasi menyediakan pandangan secara *enterprise-wide* mengenai data yang terdapat pada sistem yang terpisah. Nilai dari data itu sendiri bergantung pada pemeliharaan integritas data antar sistem. Solusi untuk pemeliharaan nilai, makna dan integritas data antara aplikasi adalah metadata.

Metadata merupakan informasi mengenai data. Semakin deskriptif, akurat dan lengkap metadata, maka akan semakin baik integrasinya. Untuk kepentingan integrasi, metadata dipresentasikan dalam format kanonik sehingga mempermudah pemetaan kembali ke sistem sumber.

4. Arsitektur integrasi proses bisnis

Arsitektur integrasi proses bisnis memodelkan proses bisnis yang melintasi batasan-batasan organisasi. Tujuan dari integrasi adalah untuk meningkatkan proses bisnis dan efisiensi. Arsitektur proses bisnis memaksimalkan *business agility* karena memungkinkan perubahan terhadap proses bisnis diimplementasikan secara cepat.

2.3 Enterprise Architecture Planning

Menurut Spewak, *enterprise architecture planning* (EAP) merupakan “*proses mendefinisikan arsitektur untuk menggunakan informasi guna mendukung bisnis dan rencana untuk mengimplementasikan arsitektur tersebut.*”[11]

EAP merupakan proses untuk mendefinisikan kedua *top layer* dari *framework* arsitektur sistem informasi Zachman. EAP menghasilkan *blueprint* mengenai data, aplikasi dan teknologi yang menghasilkan solusi jangka panjang yang *cost-effective*, bukan hanya perbaikan secara cepat. Gambar 3 menunjukkan 7 komponen atau fase EAP, yang menjelaskan bagaimana mendefinisikan arsitektur dan perencanaan. Komponen-komponen tersebut terbentuk sebagai *layer*, dimana tiap *layer* merepresentasikan fokus tugas yang berbeda, yaitu :

1. Layer 1-Where We Start

Planning initiation. Memulai EAP pada jalur yang benar, termasuk menentukan metodologi yang digunakan, siapa yang harus dilibatkan dan *toolset* apa yang digunakan.

2. Layer 2-Where We Are Today

- a. *Business modeling*. Menyusun *knowledge base* mengenai bisnis dan informasi yang digunakan untuk melaksanakan bisnis.
- b. *Current systems & technology*. Mendefinisikan sistem aplikasi apa yang terdapat saat ini dan *platform* teknologi yang mendukung.

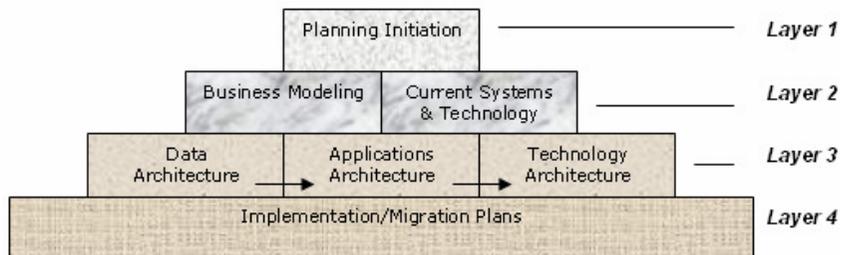
3. Layer 3-Where We Want to Be in the Future

- a. *Data architecture*. Mendefinisikan data yang diperlukan untuk mendukung bisnis.
- b. *Application architecture*. Mendefinisikan aplikasi yang diperlukan untuk mengelola data dan mendukung fungsi-fungsi bisnis.
- c. *Technology architecture*. Mendefinisikan *platform* teknologi yang diperlukan untuk menyediakan lingkungan bagi aplikasi yang mengelola data dan mendukung fungsi-fungsi bisnis.

Panah pada *layer* ini menunjukkan bahwa *data architecture* didefinisikan terlebih dahulu, lalu berturut-turut mendefinisikan *application architecture* dan *technology architecture*. Hal ini berbeda dengan metoda perencanaan sistem tradisional yang melakukan sebaliknya, dimana pertama-tama menentukan *hardware*, kemudian aplikasi yang berjalan pada *hardware*, dan terakhir data yang perlu diproses.

4. Layer 4-How We Get There

Implementation/migration plans. Mendefinisikan urutan langkah untuk mengimplementasikan aplikasi, jadwal implementasi, analisis manfaat/biaya dan mengajukan jalur yang jelas untuk melakukan migrasi dari *where we are today* ke *where we want to be*.



Gambar 3 Komponen *Enterprise Architecture Planning* [11]

2.4 Value Chain

Kunci analisis *value chain* adalah memahami aktivitas di dalam institusi yang menciptakan manfaat kompetitif serta pengaturan aktivitas tersebut lebih baik dari institusi lain pada industri. Porter (1985) mengemukakan bahwa aktivitas bisnis dapat dikelompokkan menjadi dua :

1. *Primary activities*, yang secara langsung berkaitan dengan produksi dan pengiriman produk atau layanan; serta
2. *Support activities*, yang mendukung *primary activities*, tidak terlibat langsung dalam produksi, namun memiliki potensi meningkatkan efisiensi dan efektivitas.

2.10 Four Stage Life Cycle Business System Planning (BSP)

Four stage life cycle [7] adalah *tool* yang digunakan untuk menemukan turunan dari fungsi bisnis yang terkait dengan produk atau layanan yang diberikan oleh fungsi bisnis tersebut. *Four stage life cycle* pada BSP digunakan pada tahap pendefinisian proses bisnis. Keempat siklus yang digunakan, yaitu :

1. Tahap I, *Requirement, Planning, Measurement and Control*.
Aktivitas yang menentukan berapa banyak produk atau layanan yang dibutuhkan, rencana untuk mendapatkannya dan pengukuran serta kontrol yang terkait dengan rencana.
2. Tahap II, *Acquisition*.
Aktivitas yang dilakukan untuk mengembangkan produk atau layanan atau untuk mendapatkan sumber daya yang akan dipergunakan untuk kegiatan pengembangan.
3. Tahap III, *Stewardships*.
Aktivitas untuk membentuk, mempertajam, memodifikasi atau merawat dukungan sumber daya dan untuk menyimpan atau menelusuri produk atau layanan.
4. Tahap IV, *Retirement/Disposition*.
Aktivitas dan keputusan yang mengakhiri tanggung jawab organisasi terhadap suatu produk/layanan atau isyarat terhadap berakhirnya penggunaan suatu sumber daya.

3. Analisis Kondisi Enterprise

3.1 Analisis Kondisi Saat Ini (As-is)

Analisis kondisi *enterprise* saat ini (*as-is*) bertujuan untuk melihat kondisi “X” saat ini dengan pendekatan EAP yang terdiri atas tahapan inisiasi perencanaan, pemodelan bisnis serta penilaian kondisi sistem dan teknologi.

3.1.1 Inisiasi Perencanaan

Inisiasi perencanaan dilakukan agar proyek dapat diproses secara cepat dalam arahan yang benar semenjak awal. Pendekatan *Enterprise Architecture Planning* (EAP) dalam studi kasus ini digunakan untuk memberikan landasan dalam mengatasi berbagai permasalahan terpisahnya aplikasi *legacy* sehingga timbulnya “pulau-pulau” data yang berdampak pada kurang optimalnya dukungan sistem informasi terhadap bisnis serta menjadi arahan bagi pengembangan sistem.

Langkah-langkah yang dilakukan pada fase inisiasi perencanaan adalah :

1. Menentukan ruang lingkup dan sasaran perencanaan arsitektur enterprise.

Berdasarkan analisa terhadap profil dan kegiatan “X”, dapat ditarik kesimpulan bahwa fungsi bisnis utama “X” adalah pendidikan, baik pendidikan non formal maupun formal. Oleh karenanya, pembentukan arsitektur integrasi didasarkan pada kebutuhan fungsi bisnis utama yaitu pendidikan dan fungsi-fungsi pendukungnya. Area-area yang akan dikaji, yang kemudian akan menjadi ruang lingkup dalam pembentukan arsitektur integrasi adalah area penerimaan siswa/mahasiswa baru, area pengelolaan kegiatan akademik, area pengelolaan wisuda, alumni dan bursa kerja serta area pengelolaan pembayaran biaya pendidikan. Sasaran dari pembentukan arsitektur integrasi adalah menyediakan artifak-artifak berupa daftar fungsi/proses bisnis, unit organisasi, entitas data, aplikasi dan landasan teknologi yang dapat dijadikan dasar pengembangan sistem informasi terintegrasi.

2. Menentukan visi.

Visi SI/TI “X” disesuaikan dengan visi organisasi “X”.

3. Menentukan metodologi.

Metodologi yang digunakan untuk pembentukan arsitektur integrasi adalah :

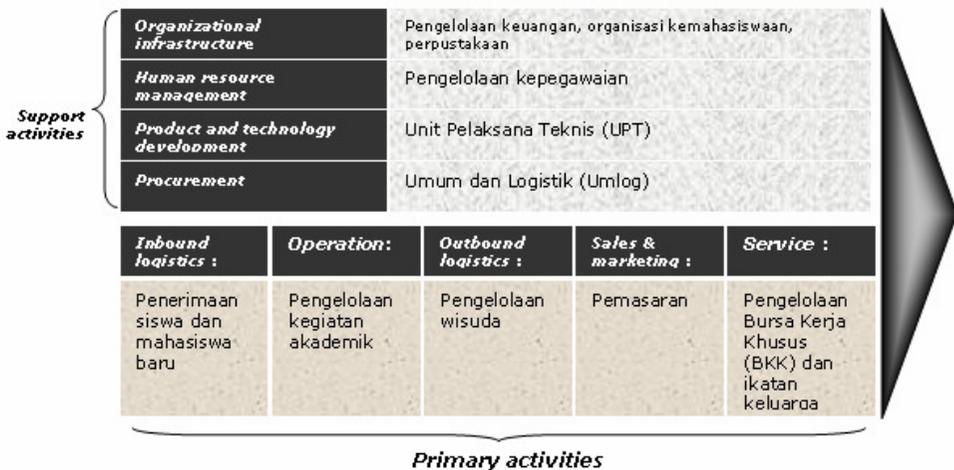
- a. Inisiasi perencanaan (*planning initiation*).
- b. Pemodelan bisnis (*business modeling*).
- c. Analisis arsitektur sistem dan teknologi saat ini (*current systems and technology architecture*).
- d. Arsitektur data, aplikasi dan teknologi (*data architecture, application architecture, technology architecture*).
- e. Pengembangan arsitektur integrasi.

3.1.2 Pemodelan Bisnis

Fungsi merupakan sekumpulan aktivitas yang dilakukan dalam bisnis dan fungsi didefinisikan berdasarkan bagian-bagiannya. Definisi fungsi bisnis hanyalah didasarkan pada aksi-aksi yang dilakukan, bukan pada organisasinya maupun orang yang bertanggung jawab untuk melaksanakan suatu fungsi. Untuk mengidentifikasi dan mendefinisikan fungsi-fungsi bisnis yang terdapat di “X”, langkah-langkah yang harus dilakukan adalah :

1. Mendefinisikan area-area fungsional utama menggunakan konsep “*value-added*” Michael Porter.
2. Memecah/mendekomposisi tiap area fungsional menjadi sub fungsi sampai fungsi tersebut menjadi *single-action*, dapat dilakukan secara berulang, memiliki *outcome* dan dapat dikaitkan dengan unit organisasi tertentu menggunakan *Four Stage Life Cycle Business System Planning*.
3. Membuat relasi antara fungsi-fungsi yang telah terinci dengan unit-unit organisasi yang melaksanakannya dalam bentuk matriks.

Pendefinisian aktivitas area-area fungsional utama di “X” menggunakan rantai nilai (*value chain*) Michael Porter, seperti yang terdapat pada Gambar 4. Dalam Gambar 4 tersebut, fungsi-fungsi bisnis di “X” dikelompokkan menjadi 2 yaitu *primary activities* dan *support activities*.



Gambar 4 Rantai Nilai “X”

3.1.3 Arsitektur Sistem dan Teknologi Saat Ini

Tujuan dari tahapan ini adalah untuk mendokumentasikan dan mendefinisikan seluruh *platform* sistem dan teknologi yang dimiliki, dikelola serta digunakan *enterprise* saat ini. *Deliverable* dari tahapan ini adalah *Information Resource Catalog* (IRC), disebut juga *System Encyclopedia* atau *System Inventory*. Langkah-langkah untuk membangun IRC adalah :

1. Mempersiapkan koleksi data aplikasi dan teknologi.
2. Mengumpulkan data IRC.

Tujuan dari tahapan ini adalah untuk menentukan macam-macam data yang disertakan dalam IRC. Langkah-langkah dalam pengumpulan data yaitu menentukan data mengenai aplikasi dan mengidentifikasi *platform* teknologi. Dokumentasi aplikasi bertujuan mengidentifikasi aplikasi apa saja yang telah dimiliki, dikelola serta digunakan oleh masing-masing unit organisasi di “X”. Saat ini data yang dihasilkan oleh proses bisnis di “X” disimpan dalam basis data aplikasi-aplikasi yang berbeda dan tidak terintegrasi. Fungsi bisnis yang telah didukung oleh aplikasi adalah fungsi akademik dan keuangan. Aplikasi-aplikasi tersebut dianggap telah mampu mendukung suatu fungsi bisnis tertentu namun

tidak dapat saling mendukung fungsi bisnis lain, karena tidak terhubung satu sama lain dan memiliki *platform* yang berbeda.

Identifikasi *platform* teknologi merupakan definisi dekomposisi secara hirarkis mengenai jenis-jenis *platform* teknologi yang terdapat dalam suatu *enterprise*. *Platform* teknologi di “X” terbagi ke dalam 3 kelompok besar yaitu perangkat keras (*hardware*), perangkat lunak (*software*) dan perangkat komunikasi (*communication*).

3.1.4 Hasil Analisis Kondisi “X” Saat Ini

Core business “X” adalah pendidikan, hal ini dapat dilihat pada rantai nilai “X” dengan menggunakan rantai nilai Porter (Gambar 4). Dalam melaksanakan aktivitas-aktivitas utamanya, “X” melakukan pemisahan pengelolaan antara kegiatan akademik untuk pendidikan non formal dengan pendidikan formal.

Pemisahan pengelolaan dari sisi bisnis, berdampak pada pengelolaan sistem informasi/teknologi informasi (SI/TI), dimana mulai dari pengadaan, pengoperasian sampai dengan pemeliharaan SI/TI dilakukan secara independen oleh masing-masing bagian pada unit-unit organisasi untuk memenuhi kebutuhan suatu fungsi bisnis yang mendesak saat itu (*temporer*). Pengelolaan SI/TI yang dilakukan secara independen ini menyebabkan perbedaan pada spesifikasi perangkat keras dan *platform* perangkat lunak di masing-masing bagian yang mengelola fungsi bisnis.

Berdasarkan hasil analisis dan informasi mengenai aplikasi pada IRC, terdapat 4 kelompok aplikasi yang masing-masing adalah aplikasi untuk mendukung fungsi akademik dan aktivitas pendukung keuangan untuk dua unit organisasi yang berbeda yaitu “A” dan “B”. Aplikasi-aplikasi tersebut dibuat oleh beberapa vendor dalam waktu yang berbeda sehingga tidak mengherankan jika aplikasi-aplikasi tersebut dibuat dalam berbagai teknologi bahasa pemrograman dan *database management system* (DBMS) yang berbeda-beda.

Perbedaan bahasa pemrograman dan DBMS atau perbedaan *platform* dan fungsionalitas aplikasi menjadikan aplikasi-aplikasi berdiri sendiri-sendiri (*stovepipe*) untuk melayani suatu fungsi bisnis akademik dan juga keuangan pada satu unit organisasi dan tidak dapat saling mempertukarkan data serta fungsionalitas antar fungsi dan juga antar unit organisasi sebagai satu kesatuan (*enterprise-wide*).

3.2 Menentukan Kebutuhan Arsitektur Mendatang (*To-be*)

Menurut Spewak, tahapan-tahapan perencanaan arsitektur *enterprise* dikelompokkan ke dalam 4 layer yaitu layer 1 (*where we start*), layer 2 (*where we are today*), layer 3 (*where we want to be in the future*) dan layer 4 (*how we get there*). Pada bagian sebelumnya telah dilakukan analisis terhadap kondisi sistem informasi “X” saat ini (*as is*). Tahapan selanjutnya adalah menentukan kebutuhan sistem informasi “X” di masa mendatang (*to be*).

3.2.1 Arsitektur Data

Arsitektur data merupakan arsitektur pertama dari 3 arsitektur yang harus didefinisikan karena kualitas data merupakan produk dasar dalam fungsi sistem

informasi. Arsitektur data berisi entitas-entitas data dimana masing-masing entitas tersebut memiliki atribut dan membentuk relasi dengan entitas data lain.

Langkah-langkah dalam membentuk arsitektur data adalah :

1. Mendaftar kandidat entitas-entitas data.
2. Mendefinisikan entitas, atribut dan relasi.
3. Merelasikan entitas dengan fungsi bisnis.

3.2.2 Arsitektur Aplikasi

Arsitektur aplikasi merupakan definisi mengenai apa yang harus dilakukan aplikasi untuk mengelola data dan menyediakan informasi bagi pelaksana fungsi-fungsi bisnis. Tahapan-tahapan untuk menghasilkan arsitektur aplikasi adalah :

1. Mendaftar kandidat aplikasi.
2. Mendefinisikan aplikasi.
3. Merelasikan aplikasi dengan fungsi.

3.2.3 Arsitektur Teknologi

Arsitektur teknologi dibuat untuk mendefinisikan teknologi yang diperlukan untuk dapat menyediakan lingkungan bagi aplikasi dalam pengelolaan data. Sama dengan arsitektur data dan aplikasi, arsitektur teknologi juga merupakan model konseptual yang mendefinisikan *platform*. Tahapan-tahapan dalam pembentukan arsitektur teknologi adalah :

1. Mengidentifikasi prinsip dan *platform* teknologi.
2. Mendefinisikan *platform*.
3. Merelasikan *platform* teknologi dengan fungsi-fungsi bisnis.
4. Merelasikan *platform* teknologi dengan aplikasi.

3.3 Pengendali dan Requirement Bisnis untuk Integrasi Enterprise

Terdapat sejumlah *business initiative* yang mengendalikan *requirement* integrasi, diantaranya adalah untuk mengurangi *business cycle time* sehingga dapat meningkatkan efisiensi dan daya saing, meningkatkan kepuasan konsumen, *merger* dan akuisisi serta untuk mematuhi suatu regulasi. Adanya perbedaan *requirement* bisnis akan berdampak pada teknologi integrasi yang digunakan.

4. Pembentukan Arsitektur Integrasi

4.1 Mengevaluasi Gap Diantara Kondisi As-is dan To-be

Pada bagian sebelumnya telah dianalisis kondisi *enterprise* saat ini (*as-is*) serta kebutuhan sistem mendatang (*to-be*). Penilaian terhadap kondisi saat ini menunjukkan kapabilitas sistem yang sedang berjalan dan tentu saja terdapat kesenjangan (*gap*) diantara kondisi saat ini dengan kebutuhan untuk dapat mencapai kondisi ideal. Melalui hasil evaluasi kesenjangan inilah nantinya akan dibuat kebijakan penyelesaian permasalahan integrasi.

4.1.1 Perbandingan Data

Entitas data yang dihasilkan dan digunakan pada sistem saat ini adalah pada fungsi penerimaan siswa/mahasiswa baru, kegiatan akademik, pengelolaan wisuda dan pembayaran biaya pendidikan. Aliran data yang disimbolkan dengan garis

beranak panah menandakan bahwa suatu area sistem menggunakan data pada area sistem lainnya, sebagai contoh area sistem kedua yaitu akademik memiliki aliran data dari area sistem pertama yaitu penerimaan siswa/mahasiswa baru. Hal tersebut berarti bahwa sistem akademik menggunakan data pendaftar, jadwal usm dan hasil usm yang diciptakan pada sistem penerimaan siswa/mahasiswa baru.

Melalui pemetaan yang telah dilakukan terhadap data pada aplikasi *legacy* dengan arsitektur ideal, ditemukan bahwa terdapat 4 entitas data dari keseluruhan 30 entitas data ideal atau 13.33 % yang belum tersedia pada aplikasi *legacy*. Jadi 86.67 % entitas data ideal sebenarnya telah dihasilkan dari aplikasi *legacy*.

Permasalahan lain dari hasil pemetaan adalah adanya 5 atau 16.67 % entitas data acuan (master) yang dihasilkan secara berulang oleh sejumlah aplikasi *legacy* yaitu entitas pendaftar, mata kuliah, siswa, mahasiswa dan dosen. Sedangkan 83.33 % entitas-entitas lainnya dikelola secara mandiri oleh unit-unit organisasi sehingga memiliki beragam format dan tidak terintegrasi. Hal ini tentu saja merepotkan pengelolaan sistem karena seharusnya isi data yang sama dibuat berulang-ulang (terjadi redundansi). Berdasarkan hal inilah maka diperlukan integrasi data yang dikelola oleh aplikasi-aplikasi *legacy*.

4.1.2 Perbandingan Aplikasi

Terdapat 4 aplikasi dari total 29 aplikasi atau 13.79 % yang termasuk ke dalam pengembangan baru, yaitu aplikasi yang berhubungan dengan aplikasi promosi dan pengelolaan BKK. Sedangkan aplikasi lain sebesar 86.21 % dipertahankan atau dimodifikasi dari aplikasi lama (*legacy*) dengan melakukan integrasi.

4.1.3 Perbandingan Teknologi

Perbandingan dokumentasi teknologi yang ada dan digunakan saat ini dengan arsitektur teknologi ideal, diperoleh beberapa kesimpulan sebagai berikut :

1. Teknologi jaringan lokal (LAN) saat ini masih dapat digunakan untuk mendukung aplikasi dan data, namun perlu dipertimbangkan untuk meningkatkan konfigurasi jaringan sehingga dapat mendukung aplikasi berbasis internet. Saat ini teknologi jaringan internet telah tersedia namun hanya digunakan untuk kegiatan belajar mengajar dan belum dimanfaatkan untuk mendukung fungsi bisnis.
2. Distribusi data dan file saat ini sebagian besar terpusat di *server*, hal ini sangat membebani kerja *server*. Oleh karenanya perlu adanya pemisahan fungsi *server* sebagai penyedia layanan jaringan dengan penyedia data dan aplikasi.
3. Diperlukan *middleware* yang dapat mengkomunikasikan data dari basis data berbasis bahasa *Clipper* ke basis data SQL Server dan begitu juga sebaliknya.
4. Setiap aplikasi tidak menyediakan *interface* agar dapat berkomunikasi dengan aplikasi lain sehingga tidak dapat dilakukan integrasi antar aplikasi pada level *interface*.

4.2 Pembentukan Arsitektur Integrasi Teknis

Arsitektur integrasi teknis menyajikan *building code* bagi proyek integrasi karena berisi spesifikasi yang akan diacu oleh proyek saat memilih teknologi integrasi. Arsitektur ini terdiri atas tuntunan dan batasan rancangan mengenai

bagaimana seharusnya aplikasi dikembangkan. Oleh karenanya, spesifikasi harus dapat mendefinisikan seluruh aspek arsitektur integrasi dan mudah diakses sehingga informasi dapat mudah ditemukan dan dipahami. Arsitektur teknis haruslah dikendalikan oleh *business requirements* dan mampu memenuhi kebutuhan di masa mendatang.

Berdasarkan hasil analisa terhadap proses bisnis, kondisi sistem dan teknologi saat ini, maka kebutuhan "X" adalah melakukan integrasi terhadap aplikasi yang ada (*legacy*), integrasi data dari berbagai unit organisasi yang akan menghasilkan informasi terintegrasi, serta integrasi gabungan dengan aplikasi baru yang akan dikembangkan. Integrasi aplikasi menjadi solusi bagi permasalahan di "X" karena adanya kebutuhan untuk tetap dapat menggunakan basis data dan layanan pada aplikasi yang ada (*legacy*) semaksimal mungkin, sehingga setiap unit organisasi dapat saling berbagi data dan proses tanpa membuat perubahan terhadap aplikasi maupun struktur data secara terus-menerus untuk mengikuti kebutuhan bisnis.

Berdasarkan kebutuhan untuk menggabungkan aplikasi yang tidak merubah logik aplikasi serta memperhatikan skema relasi basis data dan skema aplikasi yang memerlukan pertukaran data antar aplikasi dari tabel-tabel master yang tidak memiliki keterkaitan logik secara erat antar basis data maka disimpulkan bahwa keempat kelompok aplikasi memerlukan adanya integrasi pada tingkatan data.

Integrasi pada level data dipilih karena tidak adanya akses terhadap logik atau *source code* dari masing-masing aplikasi. Integrasi dapat dilakukan pada level data dengan menempatkan *software* diantara basis data dari keempat aplikasi. *Software* ini bertugas untuk melakukan ekstraksi informasi dari suatu basis data, melakukan *reformat* terhadap data dengan merubah isi dan skema jika diperlukan serta melakukan *update* basis data. Data direplikasi diantara basis data saat terjadi *update* dari sisi basis data manapun ke tabel yang bersesuaian.

4.3 Pembentukan Arsitektur Integrasi Layanan

Arsitektur integrasi layanan mendefinisikan aplikasi bisnis sebagai komponen fungsionalitas bisnis yang dapat diguna ulang dan mudah dirubah serta bagaimana komponen-komponen tersebut saling terkait. Konsep ini merupakan konsep arsitektur yang berbasis layanan (*service-oriented architecture/SOA*). Tahapan-tahapan dalam membuat arsitektur integrasi layanan adalah :

1. Menentukan *business events*.
2. Menentukan layanan.

4.4 Pembentukan Arsitektur Integrasi Informasi

Informasi dan data merupakan hal yang penting dalam proyek integrasi karena permasalahan utama pada seluruh proyek integrasi adalah bagaimana memungkinkan *interoperability* diantara sistem yang memiliki data dalam berbagai struktur dan format. Arsitektur integrasi informasi mendefinisikan infrastruktur dan proses yang memungkinkan informasi diakses pada sistem.

Dalam integrasi, aliran informasi dan juga data perlu digambarkan untuk memperoleh kejelasan mengenai data dan informasi apa saja yang sebenarnya dihasilkan atau diperlukan dari atau ke sistem. Penggambaran aliran informasi dalam sistem tersebut akan menggunakan *data flow diagram* (DFD).

4.5 Pembentukan Arsitektur Integrasi Proses Bisnis

Tujuan integrasi adalah untuk mendukung peningkatan proses bisnis dalam rangka meningkatkan efisiensi bisnis. Integrasi level proses mendefinisikan interaksi diantara sistem melalui definisi *business workflow*. Peran arsitektur integrasi proses adalah untuk menciptakan model dan definisi proses sebagai entitas yang dikelola sehingga mudah beradaptasi terhadap perubahan bisnis. Arsitektur integrasi proses mendefinisikan *end-to-end business process* yang diotomatisasi pada sistem dengan *platform* yang berbeda-beda.

4.6 Strategi Integrasi

Arsitektur integrasi yang telah dibangun merupakan *blueprint* integrasi teknologi, layanan, informasi dan proses bisnis yang menjadi dasar bagi pengembangan dan pengelolaan sistem informasi sehingga selaras dengan bisnis *enterprise*. Arsitektur integrasi dibangun dengan didasarkan pada dorongan bisnis kemudian pada kebutuhan data, aplikasi dan teknologi. Untuk menerapkan hasil pengembangan arsitektur, maka diperlukan strategi sehingga dapat menerapkan integrasi. Hasil dari penerapan strategi integrasi diharapkan menjadi acuan dalam implementasi kegiatan integrasi.

5. Kesimpulan

Berdasarkan hasil pembahasan mengenai pembentukan integrasi *enterprise* dengan studi kasus Yayasan Pendidikan “X”, maka diperoleh beberapa kesimpulan sebagai berikut :

1. Berdasarkan hasil analisis kondisi *as-is* yang diterapkan pada “X” menunjukkan bahwa kepemilikan sistem informasi saat ini tidak dilandaskan pada pemahaman terhadap fungsi-fungsi bisnis secara menyeluruh sehingga muncul “pulau-pulau” informasi. “Pulau-pulau” informasi tersebut ditunjukkan dengan didapatinya 83.33 % data yang dikelola secara independen yang berdampak pada isolasi data dan menghasilkan 16.67 % redundansi data.
2. Kebijakan SI/TI yang dibuat oleh “X” juga hanya fokus pada pemenuhan kebutuhan yang bersifat temporer sehingga mewarisi berbagai teknologi *obsolete* yang tidak mampu beradaptasi dengan teknologi baru. Permasalahan terjadi ketika adanya kebutuhan untuk mengintegrasikan data dan informasi, namun tidak dapat dipenuhi karena ternyata setiap sistem berbeda *platform*.
3. Berdasarkan hasil penentuan kebutuhan data, aplikasi dan teknologi ideal (*to-be*) menunjukkan bahwa teridentifikasi 30 entitas data dan 29 aplikasi yang harus tersedia di “X”. Sebenarnya data dan fungsionalitas sebagian besar telah tersedia pada aplikasi *legacy* namun data dan aplikasi tersebut redundan, tidak konsisten dan terisolasi pada unit-unit organisasi “A” dan “B”.
4. Kesenjangan antara kondisi *as-is* dan *to-be* disertai dengan pengendali dan *requirement* bisnis menunjukkan bahwa “X” memerlukan integrasi yang tidak hanya melandaskan pada penggunaan teknologi tertentu namun memerlukan hasil integrasi yang stabil terhadap perubahan *trend* teknologi. Hal ini diatasi dengan pembentukan arsitektur integrasi teknis, layanan, informasi dan proses yang bersifat strategik.
5. Integrasi *enterprise* bagi “X” telah terbukti dapat memperpendek siklus hidup pengembangan sistem karena dapat melakukan guna ulang (*reuse*) layanan

aplikasi dan informasi yang telah ada dan tidak perlu membuat proses bisnis, layanan aplikasi serta simpanan data yang sama secara berulang-ulang. Hal ini ditunjukkan melalui pembahasan integrasi keempat kelompok aplikasi *legacy* di "X" dengan cara melakukan translasi dan transformasi data antar simpanan data tanpa perlu membuat basis data baru.

6. Integrasi *enterprise* mampu mengkomunikasikan sistem berbeda *platform* di "X" sehingga dapat mengatasi permasalahan adanya pulau-pulau informasi sebagai akibat isolasi pengelolaan data. Hal ini ditunjukkan melalui pembahasan pembentukan arsitektur integrasi teknis, layanan, informasi dan proses yang memberikan kemampuan integrasi sistem yang tidak tergantung pada teknologi tertentu.
7. Kegiatan integrasi yang selama ini dilakukan oleh banyak perusahaan seringkali hanya memandang bahwa integrasi hanya sekedar penggunaan teknologi dan infrastruktur. Berdasarkan pembahasan tesis, ditunjukkan bahwa ketergantungan pada teknologi tertentu untuk kegiatan integrasi membuat sistem yang dihasilkan tidak mampu memberi landasaan bagi *business agility*. Integrasi *enterprise* harus dimulai dengan pemahaman terhadap permasalahan bisnis dan mengetahui bagaimana bisnis berhubungan dengan konsumen dan rekanan bisnisnya.

Daftar Pustaka

- [1] Beynon, Paul-Davies, *Database Systems*, third edition, Palgrave MacMillan.
- [2] Cummins, Fred A. (2002), *Enterprise Integration: An Architecture for Enterprise Application and Systems Integration*, Wiley, USA.
- [3] Erl, Thomas (2004), *Service Oriented Architecture : A Field Guide to Integrating XML and Web Services*, Pearson Education, Inc., USA.
- [4] Gold-Bernstein, Beth (2005), *Enterprise Integration : The Essential Guide to Integration Solutions*, Pearson Education, Inc., USA.
- [5] Hohpe, Gregor (2003), *Enterprise Integration Patterns : Designing, Building, and Deploying Messaging Solutions*, Addison Wesley.
- [6] Husein, Inne Gartina (2004), *Model Enterprise Application Integration Sistem Manajemen Billing di PT TELKOM Divisi Regional III*, Tesis Program Magister, Institut Teknologi Bandung.
- [7] International Business Machine (IBM) Corporation (July 1981), *Business Systems Planning*, 3rd edition.
- [8] Lam, Wing (2004), *Technical Risk Management on Enterprise Integration Projects*, vol. 13, Communications of the Association for Information Systems.
- [9] Linthicum, David S. (2000), *Enterprise Application Integration*, Addison Wesley Longman.
- [10] Mitchell, Victoria L. (December 2006), *Knowledge Integration and Information Technology Project Performance*, vol.30 no.4, pp.919-939, MIS Quarterly.
- [11] Spewak, Steven H. (1992), *Enterprise Architecture Planning : Developing a Blueprint for Data, Applications and Technology*, John Wiley & Sons, Inc.
- [12] Triloka, Joko (2007), *Pemodelan Arsitektur Enterprise untuk Mendukung Sistem Informasi Terintegrasi di Bidang Akademik Menggunakan Enterprise*

- Architecture Planning (Studi Kasus : Universitas Islam Negeri Sunan Kalijaga Yogyakarta)*, Tesis Program Magister, Institut Teknologi Bandung.
- [13] Ward, John & Peppard, Joe (2002), *Strategic Planning for Information Systems*, 3rd edition, John Wiley & Sons, Ltd.
- [14] www.3ht.com, diakses tanggal 31-01-2008 jam 15.39.
- [15] www.bea.com, diakses tanggal 22-09-2007 jam 11.49.
- [16] www.dciexpo.com, diakses tanggal 31-01-2008 jam 14.48.
- [17] www.ilmukomputer.com, diakses tanggal 25-09-2007 jam 17.40.
- [18] www.javaworld.com, diakses tanggal 22-09-2007 jam 12.05.
- [19] www.zifa.com, diakses tanggal 27-04-2007 jam 10.04.