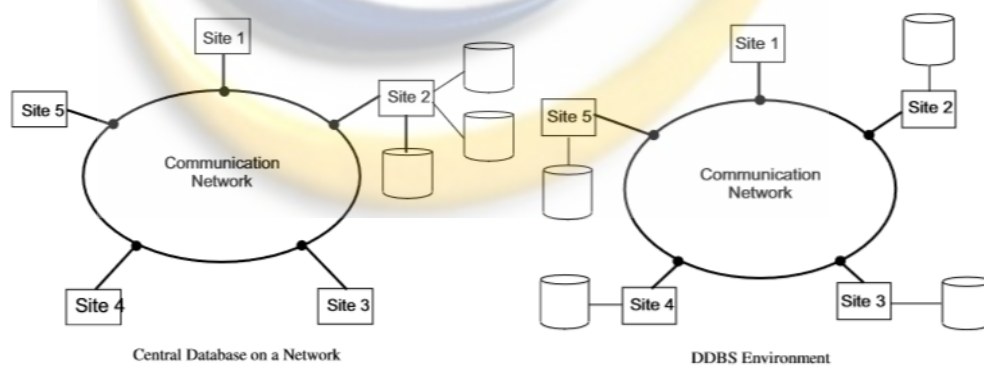


BAB II

LANDASAN TEORI

2.1 Basis Data Terdistribusi

Basis Data Terdistribusi didefinisikan sebagai sebuah *collection of multiple*, database yang saling berkaitan secara logik yang didistribusikan melalui sebuah jaringan komputer. *Distributed Database Management System (Distributed DBMS)* didefinisikan sebagai sistem perangkat lunak yang mengizinkan suatu manajemen *database* terdistribusi dan membuat transparansi distribusi kepada *user*. Teknologi sistem Basis Data Terdistribusi merupakan gabungan antara sistem *database* dan teknologi jaringan komputer.



Gambar 2.1. Perbedaan database terpusat dan database terdistribusi

Beberapa teknik pada sistem Basis Data Terdistribusi dalam hal penyimpanan data atau *table*, yaitu:

- Replikasi

Suatu teknik dalam sistem Basis Data Terdistribusi dalam melakukan duplikasi dan distribusi data dan objek-objek *database* dari satu *database* ke *database* lain serta melakukan sinkronisasi data, sehingga konsistensi data akan terjamin.

- Fragmentasi

Merupakan suatu teknik dalam mempartisi data dalam bentuk vertikal atau horizontal. Fragmentasi terdiri dari pemecahan suatu relasi kedalam relasi atau fragmen yang lebih kecil, dan memungkinkan menyimpan fragmen ke beberapa situs yang berbeda.

- Replikasi dan Fragmentasi

Merupakan kombinasi dari replikasi dan fragmentasi. Relasi dipecah menjadi beberapa bagian lalu diduplikasi ke beberapa situs yang berbeda.

2.2 Enterprise Architecture

Dalam bahasa Indonesia yaitu Arsitektur Enterprise. Terbentuk dari kata arsitektur dan enterprise. Arsitektur menyiratkan suatu perencanaan yang diwujudkan dengan model dan gambar dari komponen dari sesuatu dengan berbagai sudut pandang. Untuk definisi Enterprise mengandung arti keseluruhan komponen pada suatu organisasi dibawah kepemilikan dan kontrol organisasi tunggal (Lankhorst et al. 2005).

Dapat disimpulkan, Arsitektur Enterprise merupakan kegiatan pengorganisasian data yang dihasilkan oleh organisasi yang kemudian dipergunakan untuk mencapai tujuan proses bisnis dari organisasi tersebut (Mutyarini & Sembiring, 2006).

2.3 Integrasi Data

Integrasi merupakan proses penggabungan unsur-unsur dari dua proses yang mirip dengan membuat satu proses yang dapat digunakan untuk menggantikan proses aslinya. Proses integrasi bertujuan untuk menyelidiki hubungan ringkasan bisnis untuk menghasilkan klasifikasi dan menggabungkan kegiatan ke dalam sistem standar. Kegiatan ini melibatkan metode pencocokan dan penggabungan. (Morrison et al, 2009). Integrasi data menurut Giordano (2011) adalah serangkaian prosedur, teknik, dan teknologi yang digunakan untuk merancang dan membangun proses yang mengekstrak, merestrukturisasi, merubah, dan memuat data secara operasional atau melakukan analisis penyimpanan data baik secara *real time* atau dalam modus *batch*. Dengan demikian integrasi data merupakan proses mengkombinasikan dua atau lebih set data agar mempermudah dalam berbagi dan analisis, dalam rangka mendukung manajemen informasi di dalam sebuah lingkungan kerja. Integrasi data menggabungkan data dari berbagai sumber database yang berbeda ke dalam sebuah penyimpanan seperti gudang data (*data warehouse*).

Metode integrasi data berdasarkan jenis pengolahannya yaitu :

- a) *Transactional data integration* : berfokus pada bagaimana transaksi dibuat, diperbarui, dimodifikasi, dan dihapus
- b) *Business intelligence data integration* : berfokus pada pengumpulan transaksi dan membentuknya ke dalam struktur database untuk kegiatan analisis.

Jenis metode diatas terefleksikan dalam model arsitektur sebagai berikut :

- a) *Enterprise Application Integration (EAI)*, menyediakan *transactional data integration* untuk beberapa sistem sumber yang berbeda baik secara custom oleh perusahaan maupun paket. Paket aplikasi perusahaan yang sudah ada adalah aplikasi *Enterprise Resource Planning (ERP)* dari vendor misalnya SAP dan Oracle.
- b) *Service-Oriented Architecture (SOA)*, merupakan *transactional data integration* yang merute-kan pesan objek yang akan tampil pada tingkat yang berbeda pada antarmuka jaringan yang umum disebut bus layanan.
- c) *Federation*. Federasi data menyediakan pandangan nyata dari data yang terintegrasi (seperti jika semua dalam satu database) tanpa membawa semua data menjadi satu bentuk, sentralisasi database.
- d) *Extract, Transform, Load (ETL)*

Ekstraksi Data yaitu proses dimana data diambil atau diekstrak dari berbagai sistem operasional, baik menggunakan query, atau aplikasi ETL. Transformasi Data adalah proses dimana data mentah (*raw data*) hasil ekstraksi disaring dan diubah sesuai dengan kaidah bisnis yang berlaku. Pengisian Data merupakan proses terakhir yang melakukan

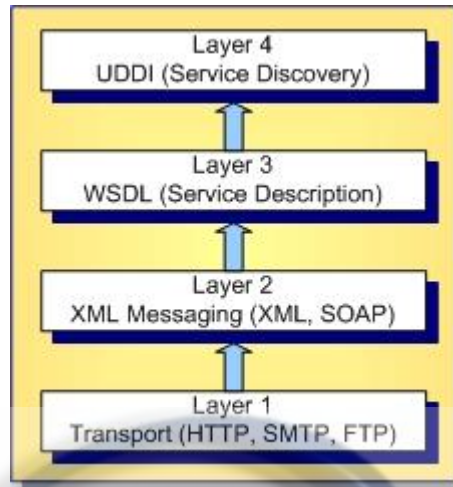
proses pemuatan data yang didapatkan dari hasil transformasi ke dalam *data warehouse*.

2.4 Web Service

Web Services adalah antar muka yang dapat diakses jaringan untuk fungsionalitas aplikasi, dibangun menggunakan teknologi internet standar. *Web services* harus bisa mengirim dan menerima pesan menggunakan beberapa kombinasi standar *internet protocols*. *Web Service* merupakan suatu komponen *software* yang merupakan *self-containing*, aplikasi modular *self-describing* yang dapat dipublikasikan, dialokasikan dan dilaksanakan pada *web*.

Gambar 2.1 merupakan blok bangunan *web service* yang mana menyediakan komunikasi jarak jauh antara dua aplikasi yang merupakan layer arsitektur *web service*.

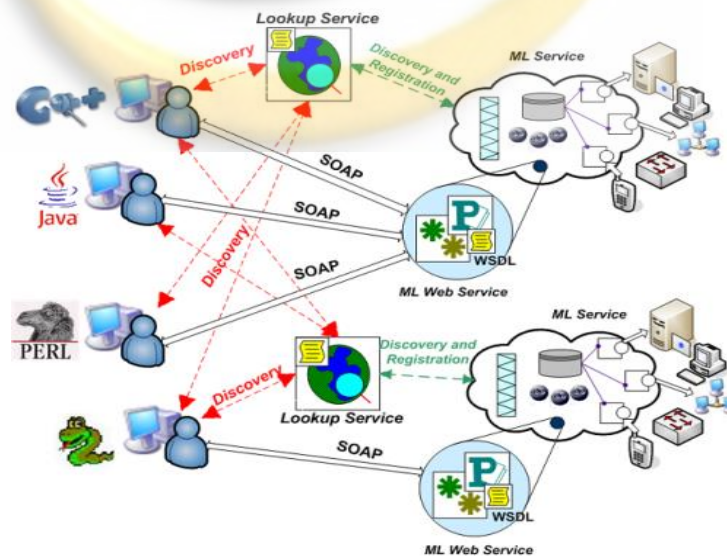
- Layer 1: protocol internet standar digunakan sebagai sarana transportasi adalah HTTP dan TCP/IP.
- Layer 2: *Simple Object Access Protocol* (SOAP) berbasiskan XML dan digunakan untuk pertukaran informasi antar sekelompok layanan.
- Layer 3: *Web Service Description Language* (WSDL) digunakan untuk mendeskripsikan atribut layanan.
- Layer 4: *Universal Description, Discovery, and Integration*, yang mana merupakan direktori pusat untuk deskripsi layanan.



Gambar 2.2 Blok Bangunan Web Service

Sarana komunikasi *web service* yang menggunakan protocol TCP/IP dan HTTP menjadikan kelebihan tersendiri yaitu kemudahan dalam pengiriman data, karena protocol HTTP merupakan protocol yang umum digunakan di internet.

Berikut ini adalah gambaran konsep *web service* yang bisa diterapkan untuk pengintegrasian berbagai sistem dengan perbedaan platform:



Gambar 2.3 SOAP Web Service

2.5 ESB (Enterprise Service Bus)

Enterprise Service Bus adalah sebuah platform integrasi berbasis standar yang menggabungkan pesan, layanan web, transformasi data, dan routing cerdas untuk menghubungkan dan mengkoordinasikan interaksi sejumlah aplikasi yang beragam di suatu perusahaan dengan integritas transaksional (Chappell, 2004).

ESB mendukung SOA melalui pengurangan jumlah, ukuran dan kompleksitas interface antara aplikasi dan layanan-layanan. Pendekatan *services bus* untuk integrasi adalah menggunakan teknologi yang menyediakan *bus* untuk integrasi aplikasi. Aplikasi-aplikasi yang berbeda tidak berkomunikasi satu sama lain secara langsung melainkan berkomunikasi melalui *backbone middleware* SOA. Fitur arsitektur ESB yang paling membedakan yaitu sifat terdistribusi dari topologi integrasi. ESB merupakan sekumpulan *middleware* layanan-layanan yang menyediakan kemampuan integrasi. *Middleware* layanan-layanan ini merupakan jantung arsitektur ESB yang menempatkan pesan untuk dapat dirutekan dan ditransformasikan (Juric, 2007; Andary-Sage, 2010).

ESB diperlukan untuk melakukan koneksi ke beberapa sumberdaya teknologi informasi. ESB harus fleksibel untuk menggabungkan dan memasang ulang komponen sesuai dengan perubahan kebutuhan bisnis. ESB melakukan koneksi komponen yang terikat longgar, sehingga menyediakan kemampuan untuk mengintegrasikan sistem ke dalam SOA dan men-*deploy* secara bertahap (Juric, 2007; Andary-Sage, 2010).

Pendekatan *services bus* untuk integrasi adalah menggunakan teknologi yang menyediakan *bus* untuk integrasi aplikasi. Aplikasi-aplikasi yang berbeda tidak

berkomunikasi satu sama lain secara langsung melainkan berkomunikasi melalui *backbone middleware* SOA. Fitur arsitektur ESB yang paling membedakan adalah sifat terdistribusi dari topologi integrasi. ESB merupakan sekumpulan *middleware* layanan-layanan yang menyediakan kemampuan integrasi. *Middleware* layanan-layanan ini merupakan jantung arsitektur ESB yang menempatkan pesan untuk dapat diroutekan dan ditransformasikan (Juric, 2007; Andary-Sage, 2010). Komponen dapat mengambil peran penghasil layanan atau pemakai layanan. Layanan-layanan dapat berupa komponen spesial seperti mesin orkestrasi, adapter untuk sumberdaya data atau adapter untuk sistem eksternal dengan transformasi pesan atau konversi transport protokol. ESB melakukan mediasi pesan antar komponen, memutuskan lokasi untuk rute pesan, dan transformasi pesan. ESB memerlukan memori persisten seperti terkoneksi dengan basis data (Juric, 2007; Andary-Sage, 2010). Menurut Juric (2007) dan Andary-Sage (2010), satu pendekatan dalam mendefinisikan arsitektur umum ESB adalah spesifikasi Java Business Integration. JBI merupakan standard untuk ESB, sedangkan ESB sendiri merupakan sebuah pola arsitektural untuk SOA. Spesifikasi JBI mendeskripsikan arsitektur *pluggable* bagi container untuk penyedia layanan dan pemakai komponen. Layanan melakukan koneksi melalui *Binding Component* (BC) atau dapat di-*host* kedalam kontainer sebagai bagian dari *Service Engine* (SE). Layanan-layanan dideskripsikan menggunakan WSDL. Pesan selalu diterjemahkan ke dalam format pesan umum dan dirutekan oleh *Normalized Message Router* (NMR).

2.6 SOAP (Simple Object Access Protocol)

Simple Object Access Protocol (SOAP) adalah sebuah standarisasi dasar dari protokol komunikasi untuk saling bertukar informasi terstruktur di antara aplikasi-aplikasi yang berjalan. SOAP adalah sebuah aplikasi pesan XML yang sangat bergantung pada skema XML dan XML *namespaces* untuk mendefinisikannya. SOAP juga sangat bergantung dengan beberapa protokol *layer* lainnya seperti HTTP, JMS, RPC untuk transmisi pesan.

2.7 XML (eXtended Markup Language)

XML merupakan dasar terbentuknya *web service* yang digunakan untuk mendeskripsikan data. Pada level paling detail *web service* secara keseluruhan dibentuk diatas XML. Fungsi utama dari XML adalah komunikasi antar aplikasi, integrasi data, dan komunikasi aplikasi eksternal dengan partner luaran. Dengan standarisasi XML, aplikasi-aplikasi yang berbeda dapat dengan mudah berkomunikasi antar satu dengan yang lain.