

PENERAPAN SISTEM REPLIKASI DAN INTEGRASI BASIS DATA TERDISTRIBUSI PADA PANGKALAN DATA PENDIDIKAN TINGGI (PDPT)

Poetri Lestari Lokapitasari Belluano

Poe3.setiawan@gmail.com

Universitas Muslim Indonesia

Abstrak

Besarnya data pelaporan EPSBED dari Prodi ke PDPT-Universitas hingga ke PDPT-DIKTI membutuhkan sarana integrasi dan komunikasi data yang konsisten, belum mengikuti standarisasi pengelolaan database terpusat pada siklus manajemen yang terkait dengan sistem Pangkalan Data Pendidikan Tinggi (PDPT). Penelitian ini bertujuan membangun sistem informasi yang memiliki integritas, konsistensi dan validitas data antar Program Studi dan PDPT Universitas. Metode yang digunakan meliputi studi lapangan dan kepustakaan untuk mengetahui data pelaporan EPSBED dari Prodi ke PDPT-Universitas hingga ke PDPT-DIKTI, perancangan sistem informasi dan basis data menggunakan *Convention Over Configuration*, pemodelan sistem *Unified Modeling Language*, pengelolaan *Data Base Management System* menggunakan *PostgreSQL*, dan *port TCP-IP* sebagai sarana komunikasi data. Hasil penelitian ini menunjukkan penerapan sistem replikasi yang bersifat *real time* sebagai bentuk alternatif pengiriman data berjumlah besar, menjamin sinkronisasi integrasi basis data dari *server* asal ke *server* tujuan, dan sistem replikasi *DBMS* menggunakan *PostgreSQL* membantu *DBMS* Program Studi terintegrasi dan terkorelasi langsung dengan PDPT-Universitas.

Kata kunci: PDPT, Replikasi, Basis data , PostgreSQL.

1. Pendahuluan

Adanya kebutuhan korelasi data dan informasi antara Dikjen DIKTI dengan entitas lain yang terkait, membutuhkan suatu bentuk pengembangan proses bisnis dan aliran data yang akurat untuk mengembangkan database terpusat dan Modul *Master Data* dalam bentuk aplikasi Pangkalan Data Pendidikan Tinggi (PDPT) sebagai bentuk implementasi integrasi konsistensi validasi data informasi pada Ditjen DIKTI dan entitas lain yang terkait.

Dalam pemrosesan input/output data dibutuhkan teknologi yang *real time* dalam mewujudkan *DBMS (Database Management System)* yang terintegrasi dan aman pada seluruh siklus manajemen yang terkait dengan sistem Pangkalan Data Pendidikan Tinggi (PDPT), baik *DBMS* Program Studi yang terintegrasi dan terkorelasi langsung dengan *DBMS* PDPT dan selanjutnya menjadi data yang otomatis sebagai bahan pertimbangan oleh *DBMS* Dikjen DIKTI untuk menetapkan kebijakan-kebijakan Perguruan Tinggi (PT).

Teknologi *IP-Streaming* yang bersifat *real time* merupakan bentuk alternatif pengiriman atau *share* data dalam jumlah yang besar, dan mampu menjamin sinkronisasi pada replikasi data dari *server* asal ke *server* tujuan. Pengiriman data menggunakan fasilitas dari aplikasi *DBMS* antara *server* asal (yakni PDPT PRODI) dan *server* tujuan (yakni PDPT Universitas).

Pangkalan Data Pendidikan Tinggi (PDPT) merupakan rekaman data menyangkut penyelenggaraan Perguruan Tinggi (PT) baik data akademik maupun data non-akademik, yang memerlukan pengelolaan tersistematis agar dapat menghasilkan informasi yang bermanfaat. Data PT digunakan untuk mendukung pengelolaan data mencakup semua siklus manajemen, validasi, dan kelengkapan data yang terdapat di internal Ditjen DIKTI dan entitas lainnya, seperti BAN-PT, Kopertis, dan PSP (Pusat Statistik Pendidikan) Balitbang. PDPT juga menjamin integritas dan konsistensi antara data, mendefinisikan dan mengklarifikasi proses bisnis, merancang dan mengimplementasikan *database* terpusat, serta menghasilkan informasi yang komprehensif serta menjamin integritas, konsistensi dan validasi data yang umumnya berasal dari database Ditjen DIKTI yang memiliki struktur, *platform*, teknologi dan produk *database* yang berbeda.

Begitu kompleks manfaat PDPT terhadap pengelolaan/manajemen data PT/Universitas, antara lain : untuk mendukung kebutuhan PT dalam memperpanjang ijin penyelenggaraan Program Studi, Serdos, Beasiswa, dan lain sebagainya. Menyediakan pusat penyimpanan data pelaporan akademik dan non akademik PT termasuk publikasi ilmiah, mendukung Sistem Penjaminan Mutu Internal (SPMI) dan Sistem Penjaminan Mutu Eksternal (SPME), mendukung kebutuhan *benchmarking* PT, mengakomodasi pengiriman data (letak geografis), menjamin keamanan data di

level jaringan dan basis data, mendukung integritas dan konsistensi data, serta membantu dalam pengambilan keputusan.

Pentingnya bentuk pemrosesan data dengan menggunakan teknologi *IP Streaming* yang mendukung penuh penerapan dari tujuan dan manfaat PDPT itu sendiri sebagai media integrasi bagi siklus manajemen yang terkait baik internal Ditjen DIKTI dan entitas lainnya, maka dibutuhkan teknologi yang mendukung atau mengakomodasi serta menjamin keamanan data sehingga konsistensi data yang dibutuhkan oleh setiap elemen entitas dalam siklus manajemen tersebut yang terkait terhadap PDPT dapat berjalan dengan efektif dan efisien.

Dalam upaya meneliti kasus pengembangan integrasi PDPT, maka penelitian ini akan berfokus di Program Studi Informatika dan Universitas Muslim Indonesia sebagai *sample* program studi dan Pangkalan Data Pendidikan Tinggi Universitas Muslim Indonesia. Dikarenakan Prodi Informatika masih menyimpan data dalam bentuk *spreadsheet* dan *word processor* untuk keseluruhan kegiatan akademik dan merekapitulasi data tersebut kedalam aplikasi Evaluasi Program Studi Berbasis Evaluasi Diri (EPSBED) secara manual, sehingga proses pelaporan data kegiatan akademik dari Program Studi ke PDPT Universitas sangat memakan waktu, disisi lain jumlah operator yang mengetahui cara menggunakan aplikasi EPSBED di program studi tersebut sangat kurang, hal ini tidak jarang menyulitkan bagi pengelola program studi untuk melakukan kegiatan penjaminan mutu internal maupun eksternal, sehingga menghambat kinerja PDPT dalam kegiatan pelaporan ke Ditjen-DIKTI.

Penelitian ini dimaksudkan untuk meneliti tentang Sistem Replikasi Pangkalan Data Pendidikan Tinggi (PDPT) Terintegrasi Dengan Teknologi IP-Streaming, yang dapat membantu pihak-pihak yang terkait dalam manajemen PDPT khususnya Program Studi dan Universitas untuk dapat mensinkronisasikan ketersediaan data yang dimanfaatkan secara berkelanjutan. Serta bertujuan untuk membangun sistem informasi yang memiliki integritas, konsistensi dan validitas data antar Program Studi dan PDPT Universitas.

2. Metode

Sumber data yang digunakan terdiri data primer yang diperoleh dari aplikasi Evaluasi Program Studi Berbasis Evaluasi Diri (EPSBED) milik program studi teknik elektro (S1) dan teknik informatika (S1) Universitas Muslim Indonesia. Serta data sekunder terdiri dokumen kamus data pendidikan, dokumen pemetaan EPSBED dengan evaluasi, buku manual PDPT versi 1.2.2, dokumen pelatihan pdpt sesi 1, 2, dan 3, dokumen teknis *webservice* push pdpt versi 1.4, data tabel referensi, serta buku, jurnal, dan literature diinternet yang dianggap menunjang pengembangan sistem.

Perancangan sistem meliputi sistem pada Program Studi Teknik Informatika dan PDPT-Universitas Muslim Indonesia, dimana ada beberapa metode yang di gunakan yaitu metode perancangan sistem informasi dan basis data menggunakan paradigma perancangan *Convention Over Configuration (COC)* dan sesuai dengan ketentuan yang diberlakukan oleh Direktorat Jendral Pendidikan Tinggi sehingga menjamin integritas, konsistensi dan validitas data.

Sedangkan pada implementasi sistem untuk penamaan menggunakan tabel data *COC* dengan penerapan pengelolaan aplikasi *DBMS* menggunakan *PostgreSQL*, dan *port TCP-IP* sebagai sarana komunikasi data. Serta pengujian sistem yang dilakukan dengan metode *Blackbox Testing* untuk menguji fungsionalitas aplikasi perangkat lunak dapat berjalan dengan baik, yaitu dengan validasi data dari elemen atribut input sinkron dengan elemen atribut output antara *request* data Program Studi ke PDPT Universitas dan sebaliknya.

Sistem yang akan dikembangkan dalam penelitian ini berbasis *web* dan menggunakan teknologi *RDBMS*, serta melibatkan pengguna dan pengelola sistem. Pengguna sistem dalam hal ini adalah operator data program studi dan pengelola sistem adalah Administrator pangkalan data perguruan tinggi. Perancangan ini menggunakan pemodelan *Unified Modeling Language (UML)* yang merupakan metode pemodelan berbasis objek.

2.1. Analisis Data

Analisa data mengacu pada dokumen kamus data pendidikan, dari analisa tersebut bahwa kategori data dalam skema PDPT dibedakan menjadi tiga kategori [1] yaitu :

- a. Data Referensi adalah data yang bersifat independen (jarang berubah datanya) dimana data ini diakses oleh tabel referensi, tabel master dan transaksi, penamaan tabel ini selalu diawali dengan kata **TREF_***, contoh data referensi adalah : provinsi, kota, status mahasiswa, dan lain sebagainya.
- b. Data Master adalah data yang jarang berubah dan mengakses data referensi serta diakses oleh data transaksi, penamaan table ini diawali dengan kata **TMST_***, contoh data master adalah perguruan tinggi, mahasiswa, dosen, matakuliah dan lain sebagainya.

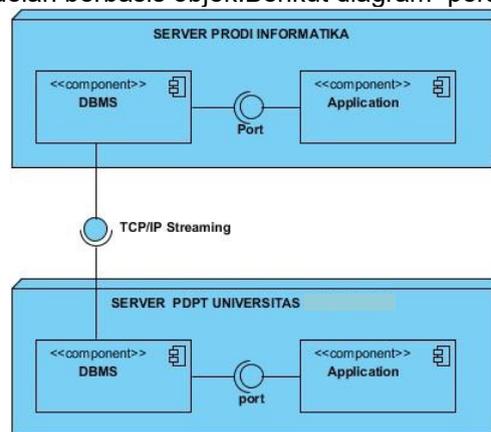
- c. Data Transaksi adalah data yang sering berubah, penamaan tabel ini diawali dengan kata **TRAN_***, contoh data transaksi adalah aktivitas mengajar dosen, aktivitas kuliah mahasiswa, publikasi dosen tetap dan lain sebagainya.

2.2. Perancangan Sistem

Perancangan sistem meliputi sistem pada Program Studi Teknik Informatika dan PDPT-Universitas Muslim Indonesia, dimana ada beberapa metode yang di gunakan yaitu:

- a. Metode perancangan sistem informasi dan basis data menggunakan paradigma perancangan *Convention Over Configuration* dan sesuai dengan ketentuan yang diberlakukan oleh Direktorat Jendral Pendidikan Tinggi sehingga menjamin integritas, konsistensi dan validitas data.
- b. Implementasi sistem untuk penamaan menggunakan tabel data COC dengan penerapan pengelolaan aplikasi DBMS menggunakan *PostgreSQL 9.2*, dan port TCP-IP sebagai sarana komunikasi data.
- c. Pengujian sistem dilakukan dengan metode *Blackbox Testing* untuk menguji fungsionalitas aplikasi perangkat lunak dapat berjalan dengan baik, yaitu dengan validasi data dari elemen atribut input sinkron dengan elemen atribut output antara *request* data Program Studi ke PDPT Universitas dan sebaliknya.

Sistem yang akan dikembangkan dalam penelitian ini berbasis web dan menggunakan teknologi *RDBMS*, serta melibatkan pengguna dan pengelola sistem. Pengguna sistem dalam hal ini adalah operator data program studi dan pengelola sistem adalah Administrator Pangkalan Data Perguruan Tinggi. Perancangan ini menggunakan pemodelan *Unified Modeling Language (UML)* yang merupakan metode pemodelan berbasis objek. Berikut diagram perancangan sistem :



Gambar 1. *Deployment Diagram* Alur kebutuhan Sistem

Alur kebutuhan sistem yang dipaparkan pada Gambar 1 menjelaskan bahwa komponen kerja baik di tingkat Prodi Informatika maupun Universitas, masing-masing bertindak sebagai *server* yang melaksanakan seluruh kegiatan pengelolaan data EPSBED menggunakan aplikasi dan *database*. Sedangkan untuk kebutuhan integrasi dan komunikasi data secara *real time* antara *DBMS* Prodi dengan *DBMS* PDPT Universitas digunakan teknologi replikasi *streaming*, serta memanfaatkan protokol *TCP/IP*.

3. Hasil dan Pembahasan

Arsitektur pengembangan sistem terdiri atas *database management system PostgreSQL* dan sistem dimana aplikasi dan *DBMS* dihubungkan melalui *port 5433* dan sistem berada di *port 80*. Proses *streaming replication DBMS* untuk pengolahan pelaporan data EPSBED dari Prodi ke PDPT-Universitas, diawali dengan memasukkan seluruh data *record* pada setiap modul yang diperlukan melalui aplikasi website iPDPT secara *online*. Berikutnya perlu dipersiapkan pula untuk menginstall *tools PostgreSQL* pada setiap komputer yang digunakan selama proses pelaporan EPSBED.

Instalasi *postgreSQL* dimanfaatkan untuk membuat dan menentukan pengguna replikasi (*user replicate*) terhadap komputer *slave* maupun *master*. Serta mengatur *Login* dan *Password* untuk masing-masing komputer yang akan digunakan, sehingga keamanan dalam proses pengiriman data akan terjaga.

Ketika seluruh data EPSBED telah tersimpan dalam *DBMS* Prodi, maka proses integrasi database ke *DBMS* PDPT-Universitas dipersiapkan dengan cara terlebih dahulu membuat konfigurasi replikasi data antara *master* dan *slave*, seperti pada Gambar 2 yang menunjukkan proses konfigurasi *master* menggunakan *postgresql.conf*. mengkonfigurasi 8 *wal-segments*, masing-masing sebesar 16MB. Untuk proses integrasi dan komunikasi data dalam pelaporan EPSBED antara Prodi dan

ILKOM Jurnal Ilmiah Volume 9 Nomor 1 April 2017

PDPT-Universitas dapat terlaksana dengan cara mengatur *IP-address* pada masing-masing komputer yang bertindak sebagai *master* dan *slave*, sehingga saat komputer Prodi melakukan proses penyimpanan data EPSBED, maka seluruh isi *records* dari atribut *database* Prodi dari hasil pelaporan EPSBED tersebut terkirim dan tersimpan secara *real time* pada database PDPT-Universitas.

```
listen_address = # make sure we're listening as appropriate
wal_level = hot_standby
max_wal_senders = 3
checkpoint_segments = 8
wal_keep_segments = 8
```

Gambar 2. Proses konfigurasi master

Proses menyalin isi database inilah yang dimaksudkan sebagai replikasi *database* yang telah diatur secara konsisten oleh *tools PostgreSQL*. Komunikasi data dalam mengelola pelaporan EPSBED antar Prodi dengan PDPT-Universitas, berlangsung secara satu arah dengan fungsi *slave lock* (penguncian data pada *database Slave*) yang dimaksudkan untuk menjaga konsistensi dan keamanan data dari Prodi sebagai *Server* dan PDPT-Universitas sebagai *Client*.

```
echo Stopping PostgreSQL
sudo service postgresql stop

echo Cleaning up old cluster directory
sudo -u postgres rm -rf /var/lib/postgresql/9.2/main

echo Starting base backup as replicator
sudo -u postgres pg_basebackup -h 1.2.3.4 -D /var/lib/postgresql/9.2/main -U replica

echo Writing recovery.conf file
sudo -u postgres bash -c "cat > /var/lib/postgresql/9.2/main/recovery.conf << _EOF1
standby_mode = 'on'
primary_conninfo = 'host=1.2.3.4 port=5432 user=replicator password=thepassword ssl
trigger_file = '/tmp/postgresql.trigger'
_EOF1"

echo Starting PostgreSQL
sudo service postgresql start
```

Gambar 3. Proses replikasi basisdata

Setelah proses replikasi basisdata (*database*) seperti yang ditunjukkan Gambar 3 berlangsung, maka komputer Prodi sebagai *Server* dapat secara efektif melaksanakan verifikasi data pelaporan EPSBED sehingga PDPT-Universitas sebagai *Client* dapat pula secara konsisten dan otomatis menerima setiap pelaporan EPSBED tersebut tanpa perlu terkendala oleh waktu.

Permasalahan yang dihadapi saat ini untuk pelaporan data EPSBED Prodi ke PDPT-Universitas sebagai pokok pembahasan yang perlu diselesaikan. Salah satu contoh kasus pada pengelolaan EPSBED terkait dengan data dosen yang terkadang berubah-ubah ataupun kelengkapan dokumen yang terhambat karena suatu hal sesuai kondisi masing-masing dosen yang bersangkutan. Kasus tersebut tidak jarang menjadi permasalahan yang mengakibatkan tertundanya pelaporan EPSBED sesuai target yang diharapkan oleh PDPT-Universitas yang diteruskan ke PDPT-DIKTI, sehingga dampak terburuk dari pelaporan yang tidak maksimal akan mempengaruhi penilaian terhadap dosen yang bersangkutan serta penilaian terhadap kinerja Prodi.

Terdapat beberapa pelaporan data dosen yang ditemukan tidak sesuai, contoh sederhana pada data kepangkatan dosen yang sering tidak ter-*update* secara teratur.

Dengan implementasi untuk replikasi *database* dengan teknologi *ip-streaming* menggunakan *sample* modul *master* untuk data Dosen akan membantu Prodi memaksimalkan proses pelaporan EPSBED secara *real-time*. Seperti antarmuka pada Gambar 4 yang menunjukkan proses *entry* modul data dosen untuk atribut "pangkat", dimana *record* pangkat C akan menjadi pangkat B.

Biodata dan Kontak Dosen		
Nomor Ktp	Tempat Lahir	Tanggal Lahir
20080023047210	PARE-PARE	23-04-1972
Alamat	Kota	Kode Pos
	Kota	Kode Pos
Telepon	Hp	
Telepon	Hp	
pangkat	Nip Baru	
Pangkat B	197204231995121001	
Pangkat C		
Pangkat D		
Pangkat A		
Pangkat C	Golongan	
	Golongan IV	
<input type="button" value="Submit"/>		

Gambar 4. Edit data dosen pada interface Prodi

ILKOM Jurnal Ilmiah Volume 9 Nomor 1 April 2017

Ketika proses “Submit” yang terlihat pada Gambar 5 dilaksanakan, maka data dosen dari Prodi yang telah mengalami perubahan akan dikirim secara otomatis melalui prosedur replikasi *database*, sehingga *database* PDPT-Universitas menerima verifikasi tersebut dan meng-update secara *real-time*. Sehingga keseluruhan atribut dan *record* data sukses tereplikasi seperti Gambar 6. Transaksi data pada proses replikasi *streaming database* dari Prodi ke PDPT-Universitas berlangsung dalam durasi waktu per *millisecond* (ms).

The screenshot shows a web application interface for Prodi. On the left is a sidebar menu with categories like 'Riwayat Pendidikan Dosen', 'Riwayat Status Dosen', 'Agama', 'Akta Dan Ijin Mengajar', 'Golongan', 'Ikatan Kerja Dosen', 'Jabatan Akademik', 'Jabatan Struktural', and 'Jenis Kelamin'. The main content area is titled 'Biodata dan Kontak Dosen' and contains several input fields: 'Nomor Ktp' (20060023047210), 'Tempat Lahir' (PARE-PARE), 'Tanggal Lahir' (23-04-1972), 'Alamat', 'Kota', 'Kode Pos', 'Telepon', and 'Hp'. Below this is the 'Status Pegawai Negeri Sipil' section with fields for 'Nip Lama', 'Nip Baru' (197204231995121001), 'Pangkat' (Pangkat B), and 'Golongan' (Golongan IV). A blue 'Submit' button is at the bottom of the form.

Gambar 5. Verifikasi data dosen pada interface Prodi

The screenshot shows a table of initial data for a lecturer in the PDPT-Universitas interface. The table has two columns: 'Tref' and 'Value'. The data is as follows:

Tref	Value
Tref Kota	
Tref Jabatan Struktural	
Nip Lama	
Nip Baru	197204231995121001
Gelar Akademik Depan	Dr.
Gelar Akademik Belakang	S.Si., M.Eng
Tref Pangkat	Pangkat B
Nira	
Tref Agama	
Tref Golongan	Golongan IV
Created	
Modified	2013-11-04 17:16:29
Created By	1
Modified By	1
Tref Tahun Pelaporan	2008-2

Gambar 6. Data awal pada interface PDPT-Universitas

Penelitian ini menunjukkan bahwa *DBMS* PDPT-Universitas dapat menerima verifikasi data yang baru dari Prodi secara *real-time* dengan mudah yakni melalui tahap *refresh* atau *reload* pada *interface* PDPT. Sehingga keseluruhan atribut dan *record* data sukses tereplikasi.

Dalam tahap perancangan aplikasinya dibutuhkan *design pattern* (pola perancangan) yaitu deskripsi dari *object* dan *class* yang saling berkomunikasi, dimana *object* dan *class* tersebut telah di modifikasi (d disesuaikan) untuk memecahkan permasalahan dalam perancangan (desain) secara umum dalam konteks tertentu. [2]

Convention Over Configuration atau disebut juga *Coding By Convention* adalah sebuah paradigma perancangan perangkat lunak yang bertujuan untuk mengurangi jumlah keputusan yang harus diambil oleh pengembang perangkat lunak, untuk kesederhanaan tanpa harus kehilangan fleksibilitas [3], dan dibutuhkan pusat data (*data center*) sebagai suatu fasilitas yang digunakan untuk

menempatkan sistem komputer dan komponen-komponen terkaitnya, seperti sistem telekomunikasi dan penyimpanan data [4]. *Data center server* adalah Pusat pemrosesan data yang didukung dengan perangkat pengolahan data tersebut. Disebut juga dengan pusat komputerisasi. *Data center server* merupakan *server* data terpusat dari jaringan di suatu jaringan, baik dalam jaringan lokal ataupun global, jaringan instansi ataupun perusahaan. *Data center server* dikelola oleh administrator. Pengelolaan *data center* yang baik mendukung seluruh kinerja dari jaringan, dari pemakaian aplikasi, oleh karena itu aturan dan standar pengukuran merupakan hal yang penting dari administrasi *data center*. Beberapa tahun ini *data center server* menjadi pembahasan yang ramai, yang sebelumnya *data center server* bukan merupakan bahasan yang perlu di telaah lebih dalam secara teori, tetapi kebutuhan akan informasi dari pengolahan data center yang baik membuat pakar-pakar jaringan akhirnya memutuskan untuk membahas *data center server* lebih dalam beserta perancangan *data center* dalam Infrastruktur TI sendiri.

Perlu pula mengenal konsep sistem basis data adalah sistem komputerisasi yang tujuan utamanya adalah memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan suatu *field* terhadap *record-record* yang sejenis, sama besar, sama bentuk dan merupakan sekumpulan *entity* yang seragam. Suatu *record* terdiri dari *file-file* yang berhubungan yang menunjukkan bahwa *file* tersebut berada dalam satu pengertian yang lengkap dan direkam dalam suatu *record* [5]. *Database* yang akan digunakan adalah Basis Data yang disajikan dalam bentuk tabel-tabel yang terdiri atas *field* dan *record*. *Field* adalah bagian terkecil dari *record*, biasanya disediakan dalam bentuk kolom, sedangkan *record* adalah kumpulan *field-field* dalam bentuk baris. Basis data (*database*) merupakan sekumpulan *field* data yang saling berhubungan [6].

Untuk basis data terdistribusi (*distributed database*), data disimpan pada beberapa tempat (*site*), Setiap tempat diatur dengan suatu DBMS (*Database Management System*) yang dapat berjalan secara *independent* [7]. Untuk tren basis data terdistribusi saat ini, pemakai harus mengetahui dimana data ditempatkan, juga harus mengetahui dimana sistem yang tidak mendukung independensi data terdistribusi dan transaksi terdistribusi *atomic*. Kedua properti tersebut harus mendukung sistem secara efisien. Untuk sistem terdistribusi yang bersifat global, properti-properti tersebut kemungkinan tidak tepat karena adanya administrasi yang terlalu berlebihan dalam membuat lokasi data yang transparan.

Protokol komputer adalah sebuah aturan atau standar yang mengatur atau mengijinkan terjadinya hubungan, komunikasi, dan perpindahan data antara dua atau lebih titik komputer. Protokol dapat diterapkan pada perangkat keras, perangkat lunak atau kombinasi dari keduanya. Protokol secara umum digunakan pada komunikasi *real-time* dimana standar digunakan untuk mengatur struktur dari informasi untuk penyimpanan jangka panjang [8].

Dalam mengatur integrasi dan komunikasi jaringan antar komputer digunakan protokol yaitu *TCP/IP*. *TCP (Transmission Control Protocol)* bertugas memastikan bahwa semua hubungan bekerja dengan benar, sedangkan *IP (Internet Protocol)* yang mentransmisikan data dari satu komputer ke komputer lain. *TPC/IP* secara umum berfungsi memilih rute terbaik transmisi data, memilih rute alternatif jika suatu rute tidak dapat di gunakan, mengatur dan mengirimkan paket-paket pengiriman data. *TCP/IP* merupakan *suite* dari gabungan beberapa protokol, yang di dalamnya terdapat protokol *TCP, IP, SMTP, POP*, dan sebagainya.

Salah satu aspek penting untuk dicatat tentang proses Replikasi *Streaming* adalah bahwa saat mentransfer terjadi *asynchronous*. Ini berarti bahwa catatan *WAL (Writes Ahead Logging Records)* di *shipped* setelah transaksi telah dilakukan sehingga ada penundaan kecil antara melakukan transaksi di primer dan untuk perubahan dapat terlihat dalam *standby*. Namun, *streaming replication* memiliki penundaan umumnya di bawah satu detik tidak seperti strategi pengalihan yang telah ada sebelumnya. Pada level kebutuhan yang besar, tantangan pada skala keluaran *database* adalah masalah sinkronisasi. Dengan *database* tunggal, ada satu sumber kebenaran untuk aplikasi. Semua *records* berada di satu tempat yang bertindak sebagai otoritas untuk data. Namun, setelah memulai skala *database*, *user* memiliki beberapa *server* yang bertanggung jawab untuk data yang digunakan *user*. Selama data dibuat (*created*), diperbarui (*updated*) dan dihancurkan (*destroyed*), informasi tentu harus disebar secepat mungkin untuk semua *server* lain. Jika *server* tidak sinkron, data akan *mis-reported* atau langsung hilang. Oleh karena itu, setiap *database* harus dalam keadaan konsisten dengan yang lain dengan *margin* kesalahan yg sedikit. Karena tidak ada pendekatan khusus dalam memecahkan masalah ini dengan sempurna untuk semua kebutuhan, sebaliknya ada banyak strategi untuk *scaling database* [9].

Untuk replikasi *IP-Streaming* dimaksudkan untuk menjalankan proses replikasi data dengan melaksanakan prosedur *streaming* pada *WAL records* pada setiap komputer yang diintegrasikan melalui *Port TCP-IP* sehingga komunikasi dan sinkronisasi *DBMS* dapat berlangsung secara *real-time* [10].

Integrasi Data juga dibutuhkan untuk mengeluarkan atau menangkap, restrukturisasi, pemindahan, dan pengangkutan atau penerbitan data dalam bentuk operasional atau analisa penyimpanan data, secara *real time* atau dalam bentuk *batch mode* [11] [12].

4. Kesimpulan Dan Saran

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan sebelumnya, maka ditarik kesimpulan bahwa sistem informasi yang dibangun untuk mengintegrasikan dan mereplikasikan data antar Program Studi dan PDPT Universitas menggunakan teknologi *IP-streaming* telah bekerja dengan baik, sesuai dengan hasil implementasi yang dipaparkan. Serta data pelaporan yang diterima PDPT-Universitas telah berjalan konsisten dengan data pelaporan yang ada pada Prodi.

Teknologi replikasi melalui *streaming database* ini bermanfaat untuk mengelola data dalam jumlah yang besar dengan durasi *real-time*. Oleh karena itu, disarankan dalam penerapannya dibutuhkan infrastruktur koneksi jaringan yang memadai agar mampu memaksimalkan kinerja replikasi *database* yang mengintegrasikan seluruh pelaporan EPSBED baik dari Prodi ke PDPT-Universitas maupun dari PDPT-Universitas ke PDPT-DIKTI.

Daftar Pustaka

- [1] Leonard Benny E.P. 2013. Thesis: *Integrasi Pangkalan Data Perguruan Tinggi Dengan Teknologi Web Service*, Pascasarjana UNHAS, Makassar.
- [2] Gamma E., Helm R., Ralph J., Vlissides J. 1998. *Design Patterns Element Of Reusable Object Oriented Software*. Addison-Wesley.
- [3] Chen Nicholas. 2012. *Convention Over Configuration*. Diakses tanggal : 12 Desember 2012 Available from: <http://softwareengineering.vazexqi.com/files/pattern.html>.
- [4] Mauricio A., Maurizio P. 2004. *Data Center Fundamentals*. Cisco Systems.
- [5] Kadir Abdul. 2003. *Konsep dan Tuntunan Praktis Basis Data*. Andi. Yogyakarta.
- [6] Oetomo Dharma Sutedjo Budi. 2002. *Perancangan dan Pembangunan Sistem Informasi*. Yogyakarta : Penerbit Andi.
- [7] Ozsu M.Tamer, Valduriez Patrick. 2011. *Principle of Distributed Database System 3rd edition*. Springer.
- [8] Malwin Syafrizal. 2008. *Seri Buku Pintar Pengantar Jaringan Komputer*. Yogyakarta : Penerbit Andi.
- [9] Natha. 2012. *Adventures in Scaling. Part 3: PostgreSQL Streaming Replication*.
- [10] Raghavendra. 2011. *PostgreSQL Streaming Replication on Windows*. Blog Relation Database Technologies.
- [11] Giordano David Anthony. 2011. *Data Integration Blueprint and Modeling*. IBM Press. Upper Saddle River. New Jersey.
- [12] Scheinblum James. 2001. *An Introduction to SOAP*. Diakses tanggal : 12 Desember 2012. Available from: <http://virturl.com/pile/scitech/soap-xml.html>.