

RANCANG BANGUN *HETEROGENOUS DISTRIBUTED DATABASE SYSTEM* UNTUK MENINGKATKAN KAPASITAS ORACLE XE 10G PADA STUDI KASUS SISTEM INFORMASI AKADEMIK

M. Didik R. Wahyudi¹⁾, Anas Azhimi Qalban²⁾

¹⁾²⁾ Teknik Informatika FST UIN Sunan Kalijaga
Email : m.didik@uin-suka.ac.id

ABSTRACT

Oracle XE 10g (Express Edition) is an Oracle edition which purposed to the requirement of the data management for small enterprise to the middle enterprise which freeware licence. Unfortunately, Oracle XE 10g database capacity limited 4 GB. One of the methods in a distributed database system is heterogenous distributed database systems use partialy connected network topology to solve the problem of the capacity limit at Oracle XE 10g. To increase the data storage capacity, the oracle XE database connected to the MySQL database with the help of the simulation database ODBC link academic information system

Keyword: *database, distributed database, heterogenous distributed database system, academic information system.*

A. PENDAHULUAN

Oracle XE 10g (*Express Edition*) adalah edisi Oracle yang ditujukan untuk keperluan pengelolaan data pada instansi atau perusahaan kecil sampai menengah yang berlisensi *freeware* / gratis. Para pemakai Oracle XE 10g dapat menggunakannya tidak hanya untuk percobaan, tapi juga dapat digunakan untuk pengembangan dan *deployment* sistem. Oracle XE dapat dikatakan sebagai sebuah produk *database server portabel* yang menyediakan teknologi inti *database server* yang diambil dari inti (*kernel*) dan digunakan dalam produk Oracle versi *Professional* ataupun *Enterprise*. Oracle XE 10g ini memiliki keterbatasan yaitu kapasitas penyimpanan data hanya sebesar 4 GB.

Keterbatasan penyimpanan data pada Oracle XE 10g ini dapat ditingkatkan

dengan cara diterapkannya sistem basis data terdistribusi, dimana penyimpanan data tidak hanya pada satu *database*, tetapi memiliki lebih dari satu *database*. Terdapat dua macam metode dalam *database* terdistribusi, yaitu: *homogenous distributed database system* dan *heterogenous distributed database system*. *Homogenous distributed database system* adalah suatu sistem yang menjalankan tipe DBMS yang sejenis di setiap unit distribusi data. Misal: MySQL didistribusikan dengan MySQL, Oracle dengan Oracle. Sedangkan *heterogenous distributed database system* adalah suatu sistem yang menjalankan tipe DBMS yang berbeda di setiap unit distribusi datanya, baik secara relational DBMS ataupun non relational DBMS. Misalkan di suatu unit memakai DBMS Oracle, dan di unit lain

memakai DBMS MYSQL, dimana kedua DBMS ini saling bisa berkomunikasi.

Melihat kemampuan Sistem Basis Data Terdistribusi ini, maka keterbatasan kapasitas pada Oracle XE dapat ditingkatkan dengan cara menambahkan DBMS lain untuk meningkatkan kapasitas Oracle XE 10g dengan mempergunakan metode *heterogenous distributed database system*.

B. TINJAUAN PUSTAKA

Distributed Database (DDB)

Basis Data terdistribusi adalah sebuah *database* yang diatur oleh sebuah *Database Management System (DBMS)* yang tersimpan secara fisik pada beberapa komputer yang terdapat pada beberapa lokasi dengan terkoneksi pada sebuah jaringan. Jaringan tersebut harus bisa menyediakan akses bagi user untuk melakukan sharing data. Terdapat dua macam sifat dari *database* terdistribusi yaitu *Heterogenous* dan *Homogenous*.

1. *Homogenous* artinya suatu *database* terdistribusi dimana data di distribusikan pada beberapa komputer dengan menggunakan DBMS (*database management system*) yang sama. DBMS digunakan pada *database* terdistribusi untuk melakukan koordinasi data pada beberapa node. Sedangkan,
2. *Heterogenous* adalah kebalikan dari *Homogenous* dimana data di sebarakan dengan menggunakan DBMS yang berbeda.

Basis data terdistribusi (*distributed database*) adalah suatu basis data yang berada di bawah kendali sistem manajemen basis data (DBMS) terpusat dengan peranti penyimpanan (*storage devices*) yang terpisah-pisah satu dari yang lainnya. Tempat penyimpanan ini dapat berada di satu lokasi yang secara

fisik berdekatan (misal: dalam satu bangunan) atau terpisah oleh jarak yang jauh dan terhubung melalui jaringan internet. Penggunaan basis data terdistribusi dapat dilakukan di *server internet*, *intranet* atau ekstranet kantor, atau di jaringan perusahaan.

Distributed Database Management System (DDBMS)

Sebuah sistem perangkat lunak yang mengatur basis data terdistribusi dan membuat pendistribusian data secara transparan sistem manajemen *database* terdistribusi (*Distributed DBMS*) merupakan sistem software yang dapat memelihara DDBS dan transparan ke user. DDBS bukan merupakan kumpulan dari file yang dapat disimpan tersendiri di setiap node dari jaringan komputer. Untuk membentuk DDBS, file tidak seharusnya berelasi secara logika saja, tetapi perlu ada struktur di antara file dan akses data bukan merupakan hal yang khusus.

DDBMS memiliki satu logikal basis data yang dibagi ke dalam beberapa fragment. Setiap fragment disimpan pada satu atau lebih komputer dibawah kontrol dari DBMS yang terpisah, dengan mengkoneksi komputer menggunakan jaringan komunikasi. Masing- masing site memiliki kemampuan untuk mengakses permintaan pengguna pada data lokal dan juga mampu untuk memproses data yang disimpan pada komputer lain yang terhubung dengan jaringan.

Pengguna mengakses basis data terdistribusi dengan menggunakan dua aplikasi yaitu aplikasi lokal dan aplikasi global, sehingga DDBMS memiliki karakteristik yaitu :

- Kumpulan dari data logik yang digunakan bersama-sama
- Fragment mungkin mempunyai copy (replika)

- Setiap site berhubungan dengan jaringan komunikasi
- Data pada masing-masing site dibawah pengawasan DBMS
- DBMS pada masing-masing site dapat mengatasi aplikasi lokal, secara otonomi
- Masing-masing DBMS berpartisipasi paling tidak satu global aplikasi.

Database Link

Konsep utama dari sistem basisdata distribusi adalah *database link*. *Database link* adalah suatu koneksi antara dua *database* fisik yang memungkinkan *client* untuk mengakses datanya sebagai satu *database logic*. *Database link* merupakan sebuah *pointer* yang didefinisikan sebagai satu-arah jalur komunikasi dari *server Oracle database* kepada *server database* lainnya. Koneksi *database link* memungkinkan *user-user* lokal untuk mengakses data pada *database remote*. Agar koneksi ini berhasil, setiap *database* pada sistem distribusi harus memiliki *global database name* yang unik pada domain jaringan.

Database link bisa bersifat *private* (pribadi) atau *public* (umum). Jika bersifat *private*, maka hanya *user* yang telah membuatnya memiliki akses dan jika *link* adalah *public* maka semua *user database* bisa menggunakannya untuk mengakses. Satu perbedaan signifikan mengenai *database link* adalah bagaimana caranya koneksi dengan *database remote*.

Database MySQL

MySQL merupakan sebuah *database server* yang mampu untuk memanajemen *database* dengan baik, MySQL terhitung merupakan *database* yang paling digemari dan paling banyak digunakan dibanding *database* lainnya. MySQL merupakan *software database* yang bersifat *free*

karena MySQL dilisensi dibawah GNU *General Public License (GPL)*. MySQL memiliki *query* yang telah distandardkan oleh ANSI/ISO yaitu menggunakan bahas SQL sebagai bahasa permintaannya, hal tersebut juga telah dimiliki oleh *database server* seperti Oracle, PostgreSQL, MS SQL, SQL Server maupun bentuk – bentuk *database* yang berjalan pada mode grafis (sifatnya visual) seperti Interbase yang diproduksi oleh Borland.

Kemampuan lain yang dimiliki MySQL adalah mampu mendukung Relasional *Database Manajemen Sistem (RDBMS)*, sehingga dengan kemampuan ini MySQL akan mampu menangani data – data sebuah perusahaan yang berukuran sangat besar hingga berukuran Giga Byte.

ODBC (Open Database Connectivity)

Open Database Connectivity (ODBC) adalah sebuah standar terbuka untuk konektivitas antar mesin basis data. Standar ini menyediakan API yang dapat digunakan untuk menjalankan dan mengoneksikan sebuah aplikasi dengan sebuah sistem manajemen basis data (SMBD). Para desainer ODBC membuatnya dengan tujuan agar ODBC terbebas dari penggunaan bahasa pemrograman tertentu, sistem manajemen basis data tertentu, dan sistem operasi tertentu. ODBC adalah bagian *database Microsoft Windows Open Services Architecture (WOSA)*, sebuah interface yang memungkinkan aplikasi *desktop* berbasis Windows yang terhubung ke lingkungan komputasi tanpa harus menulis ulang aplikasi untuk setiap *platform*. Pada gambar 2.8 dapat dilihat bagaimana arsitektur cara kerja ODBC. ODBC memiliki beberapa komponen utama, yaitu :

- **ODBC API:** sekumpulan panggilan fungsi, kode-kode kesalahan dan sintaksis SQL yang

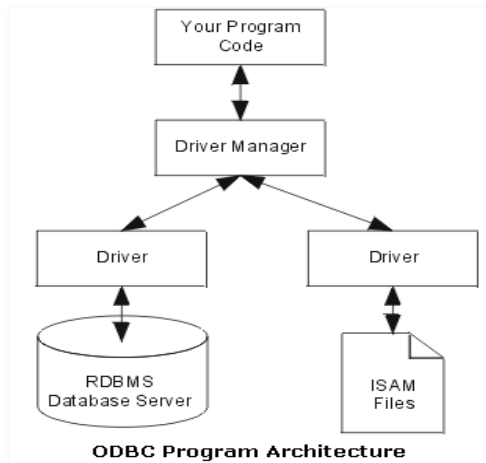
mendefinisikan bagaimana data dalam sebuah DBMS diakses.

- **Driver basis data ODBC:** *driver* (yang berupa *dynamic link library*) yang mampu memproses panggilan fungsi ODBC untuk sebuah DBMS tertentu.
- **ODBC Driver Manager:** yang bertugas untuk memuat *driver* basis data ODBC yang dibutuhkan oleh aplikasi.

Beberapa keuntungan dari ODBC adalah :

- ODBC menyediakan antarmuka yang konsisten tanpa memperhatikan jenis *database server* yang digunakan.
- Dapat diakses lebih dari satu tempat secara konkuren.
- Aplikasi tidak perlu terikat untuk setiap *database* yang akan berjalan.

ODBC *driver* mendukung sebagian besar DBMS, termasuk Oracle, PostgreSQL, MySQL, Microsoft SQL Server (tetapi tidak untuk Compact alias edisi CE), Sybase ASE, dan DB2.



Gambar 1: Arsitektur ODBC

C. METODE PENELITIAN

Penelitian ini menggunakan metode pengembangan sistem yang terdiri atas tiga tahap pendahuluan, yaitu :

1. Tahap Inisiasi

Pada tahap ini dilakukan penelusuran dan pengkajian literatur-literatur yang berhubungan dengan database Oracle XE 10g, MySQL dan basis data terdistribusi

2. Tahap Investigasi

Pada tahap ini dilakukan investigasi terhadap kebutuhan sistem yang akan dipergunakan untuk membangun database terdistribusi dengan database Oracle XE 10g dan MySQL.

3. Tahap Prototype

Pada tahap ini mulai dibangun basis data terdistribusi dan membangun suatu aplikasi yang akan menunjukkan bahwa konsep ini dapat diimplementasikan pada dunia nyata. Studi kasus yang dipergunakan adalah simulasi pada suatu Sistem Informasi Akademik.

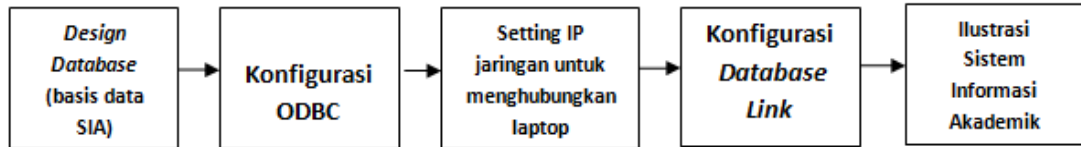
4. Tahap Validasi/Pengujian

Pada tahap ini dilakukan pengujian terhadap konsep basis data terdistribusi yang dirancang dan pada simulasi sistem informasi akademik.

D. HASIL DAN PEMBAHASAN

Arsitektur Sistem

Pada penelitian ini, metode basis data terdistribusi yang digunakan adalah metode *Heterogenous Distributed Database System* dan menggunakan topologi *Partially Connected Network*. Alur kerja dari penelitian ini sendiri dapat dilihat pada gambar 2 berikut ini.

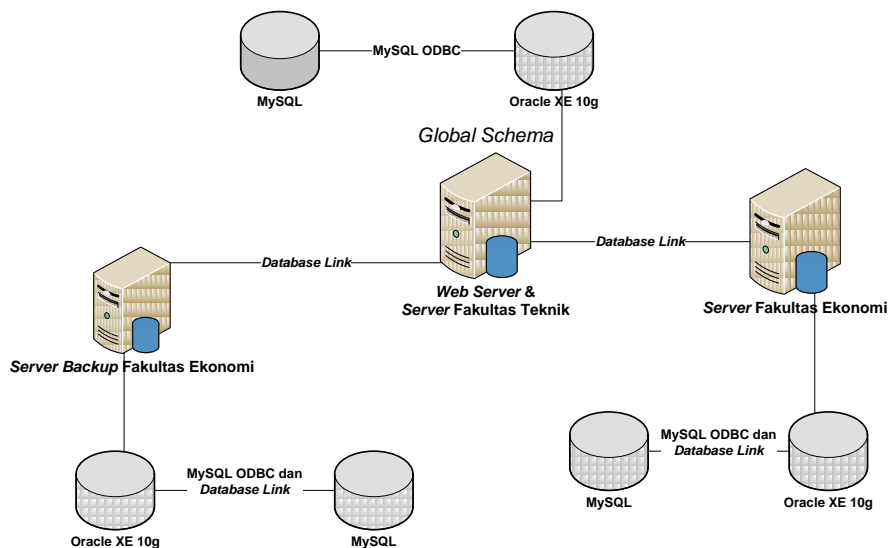


Gambar 2: Diagram Balok Alur Kerja Penelitian

Pada penelitian ini dilakukan implementasi *Heterogenous Distributed Database System* menggunakan dua jenis *software* basis data, yaitu Oracle XE 10g dan MySQL. Implementasi menggunakan tiga buah laptop sebagai perangkat utama, setiap laptop di-*install* dengan *software* Oracle XE 10g dan MySQL, salah satu laptop juga di fungsikan sebagai *server* utama pengolah sistem berbasis web. Implementasi *Heterogenous Distributed Database System* diilustrasikan dengan prototipe sistem informasi akademik berbasis web dengan bahasa pemrograman PHP. Sebagai gambaran disediakan dua *server* fakultas dan satu sebagai *server backup*, yaitu fakultas teknik dan fakultas ekonomi. *Server* fakultas teknik difungsikan sebagai *server* basis data dan juga sebagai web *server*. Selain itu server fakultas teknik ini juga berperan sebagai

global schema. Ketiga laptop dihubungkan dengan jaringan *hotspot* yang dikontrol oleh laptop *server* fakultas ekonomi dengan menggunakan aplikasi Connectify, dimana pengalaman komputer menggunakan IP kelas C dengan alamat *network* yaitu 192.168.125.0. Gambaran umum dari Implementasi metode *Heterogenous Distributed Database System* dengan dua buah laptop dapat dilihat dari gambar 3.

Penelitian ini menitik-beratkan kepada implementasi *Heterogenous Distributed Database System* dengan diilustrasikan menggunakan prototipe sistem informasi akademik. Tujuannya untuk membuktikan apakah metode ini dapat membantu meningkatkan kapasitas Oracle XE 10g yang hanya mampu menyimpan data sebesar 4GB.



Gambar 3 : Rancangan Sistem Informasi Akademik

dengan metode *Heterogenous Distributed Database System*

Dalam gambar 3 diatas, server fakultas Teknik berisi web server Apache, Oracle XE 10g dan MySQL. Komputer ini berisi juga berperan sebagai global schema. Server fakultas Ekonomi dan server backup fakultas Ekonomi memiliki isi sama yaitu OracleXE dan MySQL.

Rancangan Database

Rancangan tabel database pada ketiga komputer sama. Rancangan tabel basis data pada DBMS Oracle XE 10g diinisialisasikan dengan skema SIA yang terdiri dari 7 (tujuh) tabel yaitu tabel ADMIN, DB, DOSEN, FAKULTAS, MHS, PRODI dan SMT_AKTIF. Rancangan basis data pada MySQL diinisialisasikan dengan skema db_sia. Pada skema db_sia terdapat tiga tabel, yaitu tabel mhs, makul, dan krs. Tabel mhs di MySQL ini berguna sebagai penampung semua data mahasiswa yang disimpan di tabel mhs di Oracle dengan cara membuat duplikat semua data mahasiswa di Oracle, kemudian tabel mhs di Oracle dikosongkan untuk diisikan data mahasiswa angkatan baru, sehingga kapasitas Oracle yang terbatas tidak penuh.

Data mahasiswa tiap Fakultas, disimpan pada *server database* Fakultas masing-masing, sehingga data yang disimpan dalam Oracle XE 10g hanya sebagian kecil saja, yang berisi selain data transaksi. Data yang lain, yang berisi transaksi, disimpan pada database MySQL. Berikut ini rancangan tabel database Oracle XE :

Tabel 4.1 Struktur Tabel ADMIN

Field	Type	Field Size	Constraint
ID_ADMIN	Number	22	Primary Key
USERNAME	Varchar2	20	
PASSWORD	Varchar2	20	

LVL	Varchar2	10	
AKSES	Varchar2	20	

Tabel 4.2 Struktur Tabel DB

Field	Type	Field Size	Constraint
STAT_SIMPAN	Varchar2	1	Primary Key
DB_NAMA	Varchar2	7	

Tabel 4.3 Struktur Tabel DOSEN

Field	Type	Field Size	Constraint
ID_DOSEN	Number	22	Primary Key
NIP	Varchar2	20	
NAMA_DOSEN	Varchar2	25	
ALAMAT	Varchar2	30	
JK	Varchar2	1	
TELP	Varchar2	15	

Tabel 4.4 Struktur Tabel MHS

Field	Type	Field Size	Constraint
NIM	Varchar2	8	Primary Key
NAMA	Varchar2	25	
PASSWORD	Varchar2	20	
ALAMAT	Varchar2	30	
TGL_LAHIR	Varchar2	15	
JK	Varchar2	1	
ID_DOSEN	Number	22	
ID_FAK	Varchar2	3	
ID_PRODI	Varchar2	2	
ANGKATAN	Varchar2	4	
TELP	Varchar2	15	
STAT_SIMPAN	Varchar2	1	
LVL	Varchar2	10	

Tabel 4.4 Struktur Tabel FAKULTAS

Field	Type	Field Size	Constraint
ID_FAK	Varchar2	3	Primary Key
NAMA_FAK	Varchar2	30	
TELP	Varchar2	15	

Tabel 4.6 Struktur Tabel PRODI

Field	Type	Field Size	Constraint
ID_PRODI	Varchar2	2	Primary Key

NAMA_PRODI	Varchar2	25	
TELP	Varchar2	15	

Tabel 4.7 Struktur Tabel SMT_AKTIF

Field	Type	Field Size	Constraint
ID_SMT_AKTIF	Varchar 2	3	Primary Key
TA	Varchar 2	12	
TAHUN	Varchar 2	4	
TGL_MULAI	Date		
TGL_AKHIR	Date		
STATUS_AKTIF	Varchar 2	1	
KET	Varchar 2	6	
MULAI_PINDA H DATA	Date		
AKHIR_PINDA H DATA	Date		

Berikut ini rancangan tabel database MySQL :

Tabel 4.8 Struktur Tabel Krs

Field	Type	Field Size	Constraint
ID_KRS	Int	6	Primary Key
NIM	Char	8	
KD_MAKUL	Varchar	6	
ID_DOSEN	int	3	
ID_SMT_AKTIF	Varchar	3	
UTS	Varchar	2	
UAS	Varchar	2	
BOBOT	Varchar	1	
ANG_BOBOT	Varchar	4	
NILAI	Char	1	

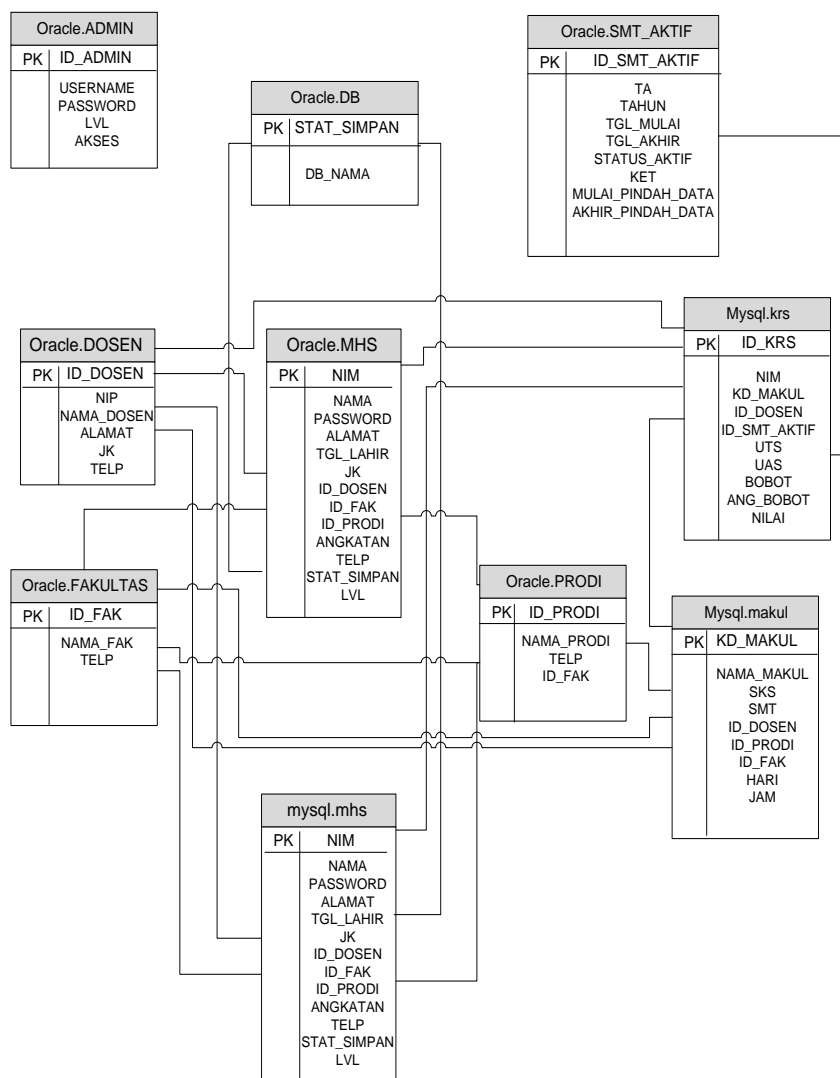
Tabel 4.9 Struktur Tabel Makul

Field	Type	Field Size	Constraint
KD_MAKUL	Varchar	6	Primary Key
NAMA_MAKUL	Varchar	35	
SKS	Varchar	6	
SMT	Int	2	
ID_DOSEN	Int	3	
ID_PRODI	Varchar	2	
ID_FAK	Varchar	3	
HARI	Varchar	10	
JAM	Varchar	15	

Tabel 4.10 Struktur Tabel Mhs

Field	Type	Field Size	Constraint
NIM	Varchar	8	Primary Key
NAMA	Varchar	25	
PASSWORD	Varchar	20	
ALAMAT	Varchar	30	
TGL_LAHIR	Varchar	15	
JK	Varchar	1	
ID_DOSEN	Int	3	
ID_FAK	Varchar	3	
ID_PRODI	Varchar	2	
ANGKATAN	Varchar	4	
TELP	Varchar	15	
STAT_SIMPAN	Varchar	1	
LVL	Varchar	10	

Tabel database yang berada pada DBMS Oracle dan MySQL, saling berhubungan sebagai suatu kesatuan. Berikut ini relasi antar tabel database yang berada pada DBMS Oracle dan MySQL.



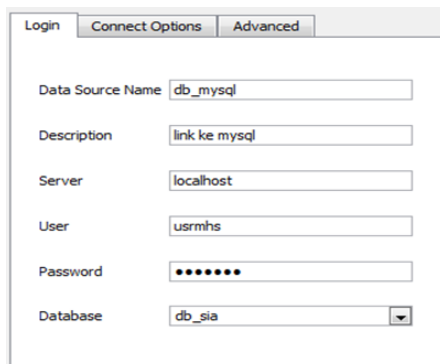
Gambar 4 : Relasi Antar Tabel

Konfigurasi ODBC MySQL

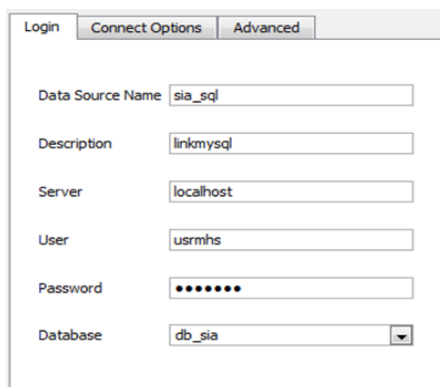
Langkah berikutnya setelah merancang basis data prototipe sistem informasi akademik yang akan digunakan untuk implementasi *Heterogenous Distributed Database System* adalah melakukan peng-install-an ODBC mysql-connector-odbc-5.1.0-alpha-win32 ke setiap laptop *server*, kemudian melakukan konfigurasi ODBC *driver* MySQL sebagai

gerbang penghubung antara *database* MySQL dan Oracle.

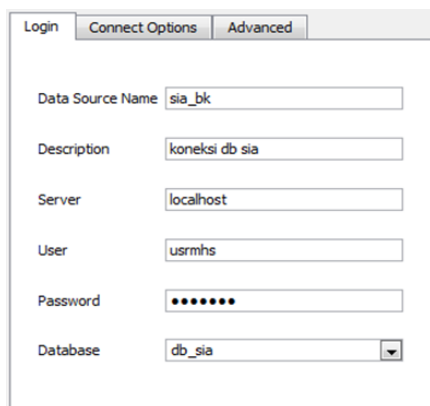
Berikut ini konfigurasi ODBC MySQL yang dibuat pada masing masing komputer :



Gambar 5 : Konfigurasi MySQL ODBC driver pada laptop pertama



Gambar 6 : Konfigurasi MySQL ODBC driver pada laptop kedua



Gambar 7 : Konfigurasi MySQL ODBC driver pada laptop ketiga

Database link

Pada tahap berikutnya setelah MySQL ODBC Driver telah di

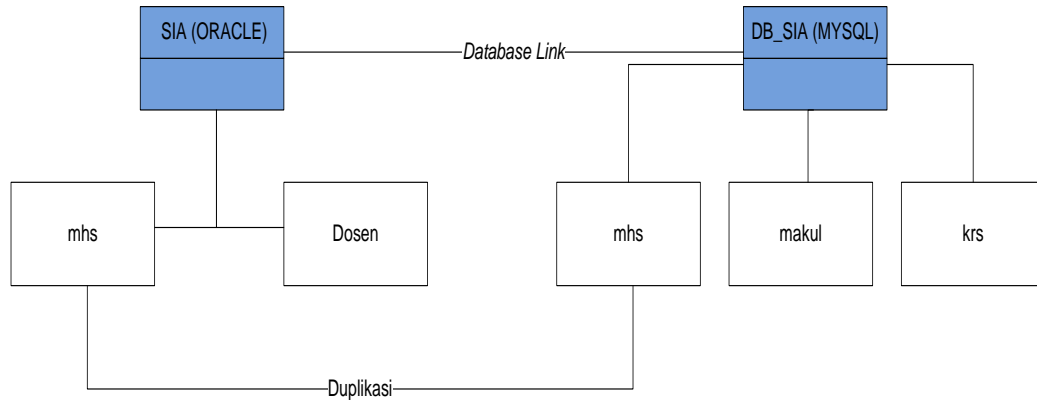
konfigurasi adalah mengkonfigurasi *database link* di tiga laptop yang ditugaskan sebagai *server database* untuk penghubung antara *database Oracle* dan *database MySQL*. *Database link* merupakan sebuah fungsi yang disediakan oleh *database Oracle*, untuk menghubungkan antara dua *database* fisik yang memungkinkan *client* untuk mengakses datanya sebagai satu *database logic*.

Dengan memanfaatkan MySQL ODBC Drive, *database link* dapat dibangun dengan cara melakukan konfigurasi pada Oracle, tahapan konfigurasi *database link* adalah sebagai berikut:

1. Melakukan konfigurasi pada file **listener.ora** yang terletak di C:\oracle\app\oracle\product\10.2.0\server\NETWORK\ADMIN
2. Lakukan konfigurasi pada file **tnsname.ora** yang terletak di C:\oracle\app\oracle\product\10.2.0\server\NETWORK\ADMIN
3. Kemudian buat sebuah *file* Net Service dengan nama *file* disesuaikan dengan nama pengenalan MySQL ODBC yang telah dibuat, pada penelitian ini nama *file*-nya **initdb_mysql.ora** yang terletak pada C:\oracle\app\oracle\product\10.2.0\server\hs\admin
4. Restart *service* OracleXETNSListener dengan cara masuk ke Run → ketikkan service.msc → cari *service* OracleXETNSListener → kemudian pilih restart
5. Kemudian *create database link* ke MySQL melalui SQL Command Line, masuk ke skema prototipe sistem informasi akademik, yaitu skema SIA.

6. Buat *database link* untuk menghubungkan *database* pada laptop pertama dengan *database* laptop kedua. Dengan cara masuk ke *interface* Oracle Application

Express. Login ke skema SIA, kemudian pada menu Object Browser pilih *create* → *database link*.



Gambar 8 : Gambaran Alur Kerja *Database Link*

Konfigurasi *File Listener.ora*

Konfigurasi *file* listener.ora harus dilakukan pada ketiga laptop yang dipakai pada penelitian ini. Konfigurasi *file* listener.ora pada ke tiga laptop ditunjukkan pada gambar berikut ini.

```
(SID_DESC =
(SID_NAME = db_mysql)
(ORACLE_HOME = C:\oracle\app\oracle\product\10.2.0\server)
(PROGRAM = hsodbc)
)
(SID_DESC =
(SID_NAME = sia_sql)
(ORACLE_HOME = C:\oracle\app\oracle\product\10.2.0\server)
(PROGRAM = hsodbc)
)
```

Gambar 9 : Konfigurasi Listener.ora Laptop Pertama

```
(SID_DESC =
(SID_NAME = db_mysql)
(ORACLE_HOME = C:\oracle\app\oracle\product\10.2.0\server)
(PROGRAM = hsodbc)
)
(SID_DESC =
(SID_NAME = sia_sql)
(ORACLE_HOME = C:\oracle\app\oracle\product\10.2.0\server)
(PROGRAM = hsodbc)
)
```

Gambar 10 : Konfigurasi Listener.ora Laptop Kedua

```
(SID_DESC =
(SID_NAME = sia_bk)
(ORACLE_HOME = C:\oracle\app\oracle\product\10.2.0\server)
(PROGRAM = hsodbc)
)
```

Gambar 11 : Konfigurasi Listener.ora Laptop Ketiga

Konfigurasi *File Tnsname.ora*

Sama seperti konfigurasi *file* listener.ora, konfigurasi *file* tnsname.ora harus dilakukan juga pada ketiga laptop yang dipakai pada penelitian ini. Konfigurasi pada ketiga laptop dilakukan memanfaatkan dua MySQL ODBC driver, pertama ODBC pada lokal laptop yang bertugas sebagai *server database* fakultas teknik dan *web server* dan yang kedua ODBC pada laptop kedua *server database* fakultas ekonomi. Berikut ini konfigurasi *file* listener.ora pada ketiga laptop.

```
db_mysql=
(DESCRIPTION=
(AADDRESS=(PROTOCOL=tcp)(HOST=localhost)(PORT=1521))
(CONNECT_DATA=(SID=db_mysql))
(HS=OK)
)

sia_sql=
(DESCRIPTION=
(AADDRESS=(PROTOCOL=tcp)(HOST=192.168.125.1)(PORT=1521))
(CONNECT_DATA=(SID=sia_sql))
(HS=OK)
)
```

Gambar 12 : Konfigurasi Tnsname.ora Laptop Pertama

```
db_mysql=
(DESCRIPTION= |
(AADDRESS=(PROTOCOL=tcp)(HOST=192.168.125.60)(PORT=1521))
(CONNECT_DATA=(SID=db_mysql))
(HS=OK)
)

sia_sql=
(DESCRIPTION=
(AADDRESS=(PROTOCOL=tcp)(HOST=localhost)(PORT=1521))
(CONNECT_DATA=(SID=sia_sql))
(HS=OK)
)
```

Gambar 13 : Konfigurasi Tnsname.ora Laptop Kedua

```
sia_bk=
(DESCRIPTION=
(AADDRESS=(PROTOCOL=tcp)(HOST=localhost)(PORT=1521))
(CONNECT_DATA=(SID=sia_bk))
(HS=OK)
)
```

Gambar 14 : Konfigurasi Tnsname.ora Laptop Ketiga

Buat File Net Service

Tahap selanjutnya adalah membuat sebuah *file net service*. *File* tersebut dibuat dengan notepad yang dijalankan dengan akses administrator dan nama *file net service* tersebut harus diawali dengan kata *init* kemudian diikuti nama *data source* MySQL ODBC. Dalam penelitian ini nama *file net service* yang dibuat adalah *initdb_mysql.ora* dan *initsia_sql.ora*. Setting *file net service* ini harus dilakukan pada ketiga laptop *server* yang dipakai dalam penelitian ini. Didalam *file net service* yang dibuat ditambahkan *script* konfigurasi yang dapat dilihat pada gambar berikut ini.

```
HS_FDS_CONNECT_INFO=db_mysql
HS_FDS_TRACE_LEVEL=OFF
```

Gambar 15 : Script File Net Service *initdb_mysql.ora*

```
HS_FDS_CONNECT_INFO=sia_sql
HS_FDS_TRACE_LEVEL=OFF
```

Gambar 16 : Script File Net Service *initsia_sql.ora*

```
HS_FDS_CONNECT_INFO=sia_bk
HS_FDS_TRACE_LEVEL=OFF
```

Gambar 17 : Script File Net Service *initsia_bk.ora*

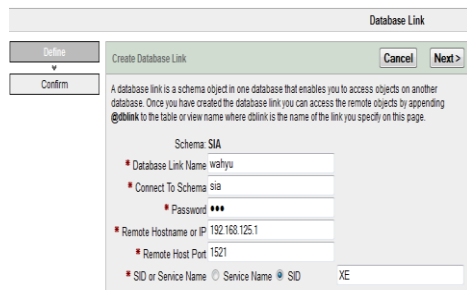
Restart Service OracleXETNSListener

Restart service OracleXETNSListener setelah konfigurasi *listener.ora*, *tnsname.ora*, dan *create net service* dengan cara masuk ke Run → ketikkan *service.msc* → cari *service OracleXETNSListener* → kemudian pilih *restart*.

Create Database Link

Langkah selanjutnya adalah membuat *database link* melalui *SQL Command Line* yang disediakan oleh Oracle. Cara membuat *database link* ke MySQL dan *database link* untuk menghubungkan Oracle laptop pertama dan Oracle laptop kedua tidaklah sama, untuk membuat *database link* Oracle MySQL harus melakukan konfigurasi file-file *listener*, *tnsname*, *net service* dan MySQL ODBC. Untuk membuat *database link* yang menghubungkan Oracle laptop pertama dan Oracle laptop kedua dengan cara login ke skema SIA melalui Oracle Application Express, sebagaimana pada gambar 18. Berikut ini *script* untuk membuat *database link* pada laptop pertama dengan perintah perintah *sql*.

```
Create Database Link linkmysql
Connect to "usrmhs" identified
by "passmhs" using 'db_mysql';
```



Gambar 18 : Create Database Link ke Oracle Laptop Kedua dengan Application Express

Proses backup data dari fakultas Ekonomi, dilakukan dengan menggunakan PL/SQL dari komputer server fakultas Ekonomi. Berikut ini script PL/SQL tersebut :

```
DECLARE
  CURSOR remote_insert IS
  SELECT * FROM MHS@whmysql;
BEGIN
  FOR rec IN remote_insert
  LOOP
    INSERT INTO mhs@ekobksql
    (NIM, NAMA, PASSWORD,
    ALAMAT, TGL_LAHIR, JK,
    ID_DOSEN, ID_FAK, ID_PRODI,
    ANGKATAN, TELP,
    STAT_SIMPAN, LVL) VALUES
    (rec.NIM, rec.NAMA, rec.PASSW
    ORD, rec.ALAMAT,
    rec.TGL_LAHIR, rec.JK, rec.ID
    _DOSEN, rec.ID_FAK, rec.ID_PR
    ODI, rec.ANGKATAN, rec.TELP, r
    ec.STAT_SIMPAN, rec.LVL);
  END loop;
END;
```

E. KESIMPULAN

Implementasi *Heterogenous Distributed Database System* pada prototipe sistem informasi akedemik yang dilakukan menunjukkan dapat meningkatkan kapasitas Oracle XE 10g dengan memanfaatkan MySQL ODBC dan *Database Link* sebagai pintu gerbang penghubung basis data Oracle XE 10g dengan basis data MySQL dalam

membangun sebuah jaringan basis data terdistribusi.

Dengan mengimplementasikan *Heterogenous Distributed Database System*, kapasitas Oracle XE 10g yang hanya sebesar 4 GB, dapat ditingkatkan kapasitasnya dengan bantuan MySQL. Hal ini dapat dilihat dari percobaan yang dilakukan dengan menggunakan studi kasus sistem informasi akademik. Data akademik yang besar dapat disimpan secara bersama-sama dalam database Oracle XE 10g dan MySQL. Oracle XE 10g berperan sebagai *global schema* untuk mengendalikan akses data ke seluruh database.

DAFTAR PUSTAKA

- [1] Andjani, Jean Andrea Terry. 2013. *Data & Basis Data*. <http://www.slideshare.net/cascana1/data-basisdata-19076014>. Diakses: 18 Juni 2013 08:00 WIB.
- [2] Anggraeni, Dwi. 2009. *Open Database Connectivity (ODBC)*. http://blog.unsri.ac.id/userfiles/59081003042_4.doc. Diakses: 9 Juni 2013 23:50 WIB.
- [3] Bramantya, Alit Mahendra. 2009. *Arsitektur Basis Data Terdistribusi*. <http://oke.or.id/wp-content/plugins/downloadsmanager/upload/DDBM S2>.
- [4] pdf. Diakses : 24 September 2012 19:30 WIB.
- [5] D.Y. Ye, dkk. 2002. *Mobile Agents for Distributed Transactions of a Distributed Heterogeneous Database System*. <http://ftp10.us.freebsd.org/users/azhang>
- [6] /disc/springer/0558/papers/2453/24530403.pdf. Diakses : 16 Mei 2013 11:50 WIB.
- [7] Jogiyanto. 2001. *Analisis & Desain Sistem Informasi : pendekatan terstruktur teori dan praktek aplikasi bisnis*. Andi, Yogyakarta.
- [8] John Miles Smith, dkk. 1981. *Multibase-Integrating Heterogenous Distributed Database Systems*. http://www.eecs.berkeley.edu/~wong/wong_pubs/wong71.pdf. Diakses: 3 juni 2013 23:42 WIB.
- [9] Mulyanto, Agus. 2009. *Sistem Informasi Konsep dan Aplikasi*. Yogyakarta:Pustaka Pelajar.
- [10] Nugroho, Bunafit. 2009. *Aplikasi Pemrograman Web Dinamis dengan PHP dan MySQL*. Yogyakarta: Gava Media.
- [11] Oracle. 2006. *Oracle Database Administrator's Guide, 10g Release 2 (10.2) B14231-02*. USA: Oracle.
- [12] Prasetyo, Eko. 2008. *Pemrograman Web PHP & MySQL untuk Sistem Informasi Perpustakaan*. Yogyakarta: Graha Ilmu.

- [13] Pratama, Mudafiq Riyan. 2010. Database Link pada Oracle 10g. <http://ilmukomputer.org/wpcontent/uploads/2010/12/MudafiqDatabaseLinkOracle.pdf>. Diakses: 5 Maret 2013 14:23 WIB.
- [14] Putranta, H.D. 2004. Pengantar Sistem dan Teknologi Informasi. AMUS. Yogyakarta.
- [15] Raharjo, Budi. 2011. Pemrograman Web dengan PHP + Oracle. Bandung: Informatika.
- [16] Reza Ghaemi, dkk. 2008. Evolutionary Query Optimization for Heterogeneous Distributed Database Systems. <http://www.waset.org/journals/waset/v19/v19-9.pdf>. Diakses: 16 Mei 2013 11:58 WIB.
- [17] Setiyadi, Didik. 2010. Sistem Basis Data. <http://files.edithadiansyah.com/SistemBasisData/Bab3BahasaBasisData.pdf>. Diakses: 6 Juni 2013 23:00 WIB.
- [18] Susanto, Budi. 2012. Membangun Sistem Basis Data dengan OracleXE. Yogyakarta: Andi.
- [19]