

## Integrasi Data dengan *Framework* OGSA-DAI

Wiranto H. Utomo

Fakultas Teknologi Informasi  
Universitas Kristen Satya Wacana Salatiga  
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia  
Email: wiranto.uksw@gmail.com

Ahmad Ashari

Electronics and Instrumentation Lab. Physics Department  
Universitas Gajah Mada, Yogyakarta 55281, Indonesia  
Email: Eashari@ugm.ac.id

### Abstract

Data integration refers to the activity a person, or a computer program, performs to combine two or more data set together. The aim of the combination is to create a single data set that another computer program is able to analyze and generate information used for scientific research. OGSA-DAI is a grid middleware that facilitates the sharing of data and allowing users to construct workflows that commonly consists of database query, dataset transformations and dataset delivery. Using OGSA-DAI framework, this research has been successfully deploy data service, data resource or MySQL database and Microsoft SQL Server database, expose data resource, and build client toolkit to access data resource.

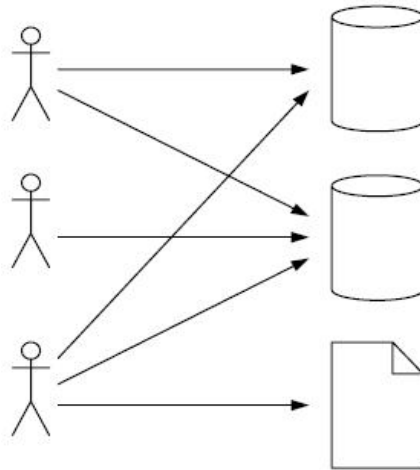
**Keywords :** OGSA-DAI, Data Integration, Workflow, Database.

### 1. Pendahuluan

Komputasi Grid menekankan isu kolaborasi, sharing data dan sumberdaya. Layanan grid merupakan middleware yang mendukung sharing dan alokasi sumberdaya tersebut. Saat ini basisdata yang digunakan oleh pengguna grid bervariasi dan sharing data diantara pengguna membutuhkan layanan yang memungkinkan terjadinya kolaborasi data.

OGSA-DAI (*Open Grid Services Architecture – Data Access and Integration*) dapat divisualisasikan sebagai serangkaian layanan grid yang saling bekerjasama yang memungkinkan basisdata diakses melalui *grid service interface*. Framework OGSA-DAI merupakan framework yang dapat diperluas, diakses melalui web services, mengeksekusi *data-centric workflows*, mencakup data resources yang heterogen untuk tujuan akses data, integrasi, transformasi dan delivery di dalam sebuah grid, dan dimaksudkan sebagai toolkit untuk mengembangkan *application-specific data services* pada level yang lebih tinggi. Grid merupakan

sharing sumber daya, sedangkan OGSA-DAI merupakan sharing sumber daya data terstruktur (Lihat Gambar 1).



**Gambar 1** *Sharing* Data melalui OGSA-DAI merupakan *Sharing* Sumberdaya Data Terstruktur

Implementasi OGSA-DAI dirancang dengan tujuan untuk memungkinkan sumberdaya data eksternal seperti basisdata untuk bergabung dalam framework OGSA dan dari OGSA ini dapat diakses melalui *interface* layanan grid yang standard. Dengan menggunakan OGSA-DAI, sumberdaya yang bermacam-macam yang terpisah dapat diakses secara menyatu (*uniform*). Hal ini mencakup dukungan untuk registrasi dan *discovery* basisdata dan interaksi dengan basisdata tersebut. Struktur hasil pengembalian, *method* dan lokasi dapat diatur oleh *client* yang menggunakan. Adapun tujuan penulisan artikel ini adalah 1) Melakukan *deploy data service* dengan nama **ogsadai/Data Service**; 2) Melakukan *deploy* data resource dari dua basisdata yaitu basisdata **MySQL** dan basisdata **SQL Server**; 3) Melakukan expose data resource yang sudah di-*deploy* tersebut; 4) Melakukan akses data resource tersebut menggunakan data browser; 5) Membuat aplikasi *client toolkit* guna mengakses *data resource* tersebut dari sisi *client*.

## 2. Framework OGSA-DAI

OGSA-DAI merupakan framework yang diperluas untuk akses data dan integrasi data. OGSA-DAI digunakan untuk mengekspose sumber daya yang heterogen untuk *grid* melalui *web services*. OGSA-DAI berinteraksi dengan sumber daya data dengan *query & update*, transformasi/kompresi data, *delivery* data dan fungsionalitas aplikasi tertentu. OGSA-DAI merupakan dasar untuk *services* pada level yang lebih tinggi yang mencakup federasi, mining, visualisasi, dan sebagainya.

Proyek OGSA-DAI termasuk dalam *Open Grid Forum DAIS Working Group* dengan menggunakan spesifikasi DAIS (*Database Access and Integration*). Dalam hal ini OGSA-DAI menjadi referensi implementasi dari DAIS. Kelompok

spesifikasi OGSI. Implementasi OGSA-DAI dikeluarkan dengan mengimplementasikan interface yang didefinisikan dalam *framework* OGSA-DAI yang didasarkan pada GT3. Tujuan OGSA-DAI adalah memungkinkan sumber daya eksternal, seperti database untuk digabungkan didalam *framework* OGSA, dan dari sini dapat diakses melalui *interface Grid services standard*. Dengan menggunakan OGSA-DAI, berbagai sumber daya yang berbeda-beda dan terpisah dapat diakses dalam sebuah kesatuan. OGSA-DAI mendukung registrasi/*discovery database* dan interaksi dengan *database* tersebut.

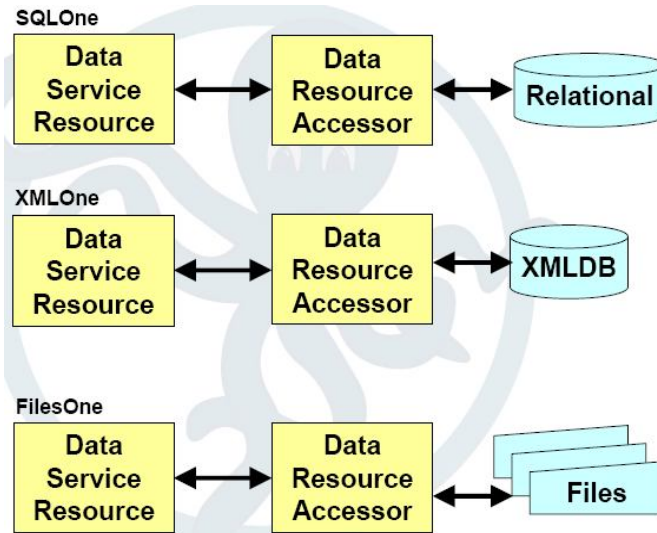
Proyek OGSA-DAI dibangun dalam beberapa tahap :

- Tahap pertama dimulai pada bulan Pebruari 2002 dan berakhir pada Juli 2003 (18 bulan) dengan melibatkan EPCC, NeSC, IBM, Oracle, ReSC, eSNW. Pada tahap ini dikeluarkan 3 *release* utama dan 3 *release intern*. Proyek ini menghabiskan dana £3 juta yang didanai oleh DTI/EPSC melalui *UK e-Science Grid Core Programme*.
- Tahap kedua dinamakan DAIT (DAI-Two) dimulai pada bulan Oktober 2003 dan berakhir pada bulan Oktober 2005 (24 bulan) dengan melibatkan EPCC, NeSC, IBM, NEReSC, eSNW. Pada tahap kedua ini dihasilkan empat *release* utama. Proyek tahap kedua ini menghabiskan dana £1.5 juta dari DTI/EPSC melalui *UK e-Science Grid Core Programme 2* sebagai bagian dari proyek OMII-UK.
- Tahap ketiga dinamakan proyek OMII UK Node yang dimulai pada bulan November 2005 dan berakhir pada bulan Oktober 2008 nanti (36 bulan). Proyek ini melibatkan EPCC dan NeSC dan masih menghasilkan satu *release*. Proyek ini menghabiskan dana £1.9 juta dari EPSC.

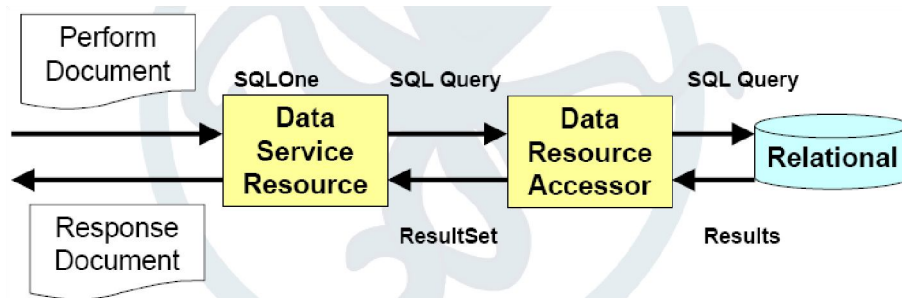
OGSA-DAI digunakan secara meluas dalam proyek-proyek sebagai berikut AstroGrid ([www.astrogrid.org](http://www.astrogrid.org)) - *query* terdistribusi untuk *database* astronomi yang sangat besar; Automed and ISpider ([www.doc.ic.ac.uk/automed](http://www.doc.ic.ac.uk/automed)) dan ([www.ispider.man.ac.uk](http://www.ispider.man.ac.uk)) – Integrasi data berbasis model dan platform informatika berbasis Grid untuk *proteomic* ; CancerGrid ([www.cancergrid.org](http://www.cancergrid.org)) – penyimpanan dan analisis data terdistribusi yang berisi data laboratorium dan klinik; ESSC ([www.nerc-essc.ac.uk](http://www.nerc-essc.ac.uk))- simulasi atmosfer dan lingkungan; Gold ([www.goldproject.ac.uk](http://www.goldproject.ac.uk)) – menyediakan infrastruktur untuk organisasi virtual ; NTRAC ([www.ntrac.org.uk](http://www.ntrac.org.uk)) – mirip dengan proyek CancerGrid.

Ada beberapa alasan mengapa OGSA-DAI cocok untuk mengakses data dan kebutuhan integrasi dalam lingkungan grid. Alasan tersebut diantaranya adalah mencakup a) cocok dengan model *grid* atau *web service*; b) adanya *workflows* yang membungkus *interaksi web service* yang banyak ke dalam interaksi tunggal; c) adanya solusi untuk mengakses berbagai data, mentransformasi, menghantar (*delivery*) dan skenario integrasi tanpa perlu untuk melakukan *develop* aplikasi tertentu; d) dapat diperluas berdasarkan fungsionalitas untuk melakukan data *query*, *update*, transformasi dan *delivery*; e) *framework* yang dapat diperluas dan serbaguna, karena developer dapat menambahkan atau melakukan pengaturan kapabilitas; f) *platform independen* berjalan pada berbagai platform yang mendukung *Java*; g) adanya lapisan *security* tambahan dapat disediakan jika dibutuhkan, misalnya otorisasi dapat dilakukan pada level *web service* dan atau

*Request* merupakan sekumpulan aktivitas terkoneksi yang seharusnya dieksekusi oleh *data resource*. Pada saat *request* ini XML menampilkan dokumen yang diminta oleh *client*. Sedangkan aktivitas merupakan operasi yang berkaitan dengan data individual yang dapat berupa 0 atau lebih input dan 0 atau lebih output. *Response* merupakan status eksekusi dari *request* yang mungkin dengan data hasil. Pada saat *response* ini XML menanggapi dokumen yang dikembalikan ke *client*. Pada proses *request* dan *response* ini mesin OGSA-DAI melakukan beberapa pekerjaan yaitu melakukan *parse request*, mengeksekusi aktivitas, dan membangun *respons*.

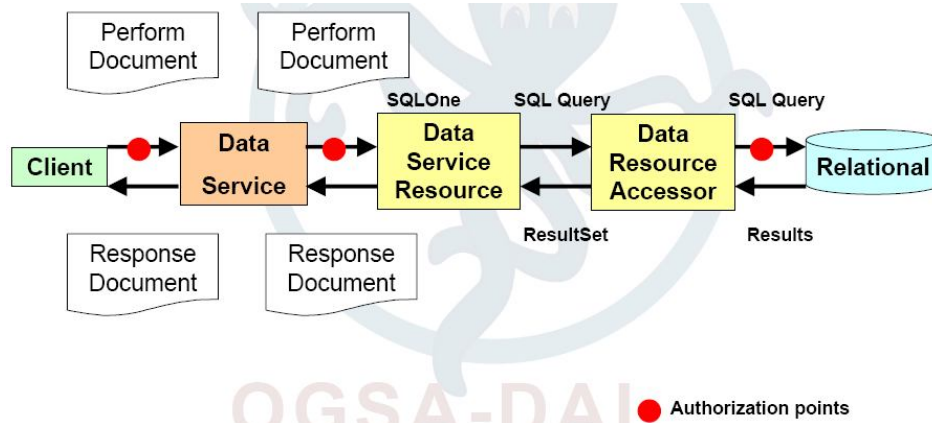


Gambar 4 Data Service Resource



Gambar 5 Request dan Response

Aktivitas merupakan operasi yang berkaitan dengan data individual yang dapat berupa 0 atau lebih *input* dan 0 atau lebih *output*. Aktivitas untuk sumberdaya data relational adalah a) melakukan *query* SQL, *update*, *stored procedure*, *bulk load*, *extract logical* dan skema fisik; b) melakukan konversi *ResultSet* ke *WebRowSet*, *ResultSet* ke CSV, *ResultSet* ke *bytes*, skema *database* relational ke XML ; c) proyek *ResultSet* atau *WebRowSet* ke dalam kolom; d) melakukan *ekstrak byte* dari *ResultSet*. Aktivitas untuk sumberdaya XMLDB adalah melakukan



**Gambar 7** Interaksi *Client* dengan Data Service

Dalam berinteraksi dengan *data service* tersebut dilakukan dengan menggunakan *client toolkit* yang bertugas sebagai berikut: 1) Membangun dan melakukan submit *request* dalam *Java* dan bukan *XML*; 2) *Toolkit* menangani konstruksi *request SOAP* dan melakukan *response parsing*; 3) Melakukan render layanan OGSA-DAI yang berjenis *transparent*; 4) Melakukan abstraksi *Java* dari *Data services*, *Data service resource ID* dan *session ID*, *Requests* dan *responses*, serta aktivitas.

### 3. Studi Kasus dan Pembahasan

Dari berbagai penggunaan OGSA-DAI yang telah dibahas diatas, maka akan diimplementasikan beberapa penggunaan saja sebagai studi kasus. Dalam studi kasus ini akan dikerjakan beberapa hal yaitu :

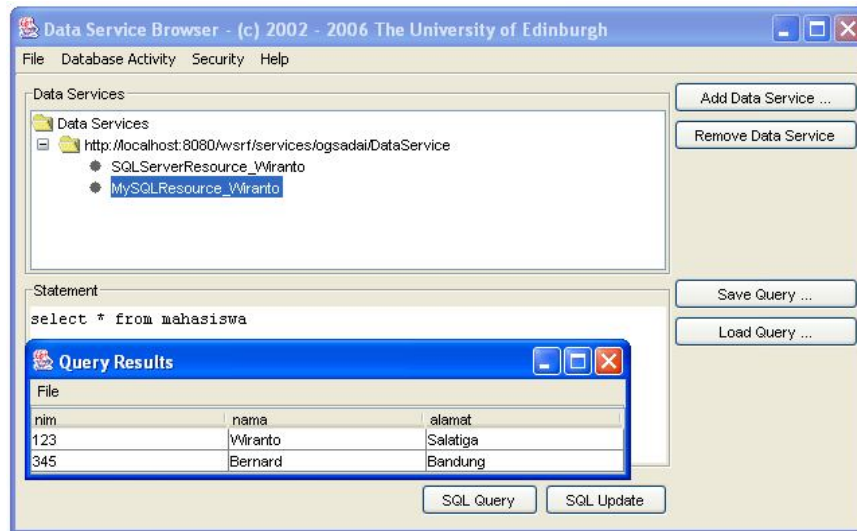
1. Melakukan *deploy data service* dengan nama **ogsadai/Data Service**.
2. Melakukan *deploy data resource* dari dua basisdata yaitu basisdata *MySQL* dan basisdata *SQL Server*.
3. Melakukan *expose data resource* yang sudah *dideploy* tersebut.
4. Melakukan akses *data resource* tersebut menggunakan *data browser*.
5. Membuat aplikasi *client toolkit* guna mengakses *data resource* tersebut dari sisi *client*. *Software* yang digunakan untuk melakukan uji coba kasus ini adalah *J2SDK 1.4*, *Apache Ant 1.6*, *Tomcat 5.0*, OGSA-DAI WSRF versi 2.2, serta *Globus Toolkit* dengan *WS Core* versi 4.02.

*Deploy data service* dapat dilakukan dengan dua cara, yaitu 1) Menggunakan *command line* atau menggunakan *Graphical User Interface (GUI)*. Untuk uji coba ini, *deploy data service* dilakukan dengan menggunakan *GUI*. *Data service* yang di *deploy* diberi nama **ogsadai/DataService** (Lihat Gambar 8). *Data service ogsadai/DataService* berperan sebagai *container web service* dari *data resource*.

*Deploy data resource* juga dapat dilakukan dengan dua cara yaitu 1) menggunakan *command line* atau menggunakan *Graphical User Interface (GUI)*. Untuk studi kasus ini *deploy data resource* menggunakan *GUI*. *Data resource* yang

Ekspose *data resource MySQLResource\_Wiranto* dilakukan dengan menggunakan GUI (Lihat Gambar 10).

Setelah data resource di-*expose* ke kontainer *web service*, maka data-data yang di-*expose* dalam *data resource* tersebut dapat diakses, diintegrasikan, dan dimanipulasi (*insert, delete dan update*) dari berbagai tempat secara online. Untuk uji coba dalam tulisan ini hanya dilakukan akses data dengan menggunakan data *browser* dan *client toolkit*. Uji coba dilakukan dengan melakukan *retrieve* basisdata dengan perintah *select \* from mahasiswa* menggunakan data *browser* dan dengan hasil *output* yang dapat dilihat pada Gambar 11.



**Gambar 11** Akses *Data Service Resource MySQLResource\_Wiranto* menggunakan *Data Browser*

Selain menggunakan data *browser*, akses data juga dapat dilakukan dengan membuat *interface* di sisi *client* dengan menggunakan bahasa pemrograman apapun. Dalam studi kasus ini digunakan bahasa pemrograman *Java*. Adapun kode selengkapnya dapat dilihat pada *listing* dibawah ini.

```
package uk.org.ogsadai.examples.clienttoolkit;

import uk.org.ogsadai.client.toolkit.GenericServiceFetcher;
import uk.org.ogsadai.client.toolkit.Response;
import uk.org.ogsadai.client.toolkit.activity.ActivityRequest;
import uk.org.ogsadai.client.toolkit.activity.sql.SQLQuery;
import uk.org.ogsadai.client.toolkit.activity.sql.WebRowSet;
import uk.org.ogsadai.client.toolkit.service.DataService;

public class ClientMySQL {

    private static final String COPYRIGHT_NOTICE =
        "(c) Wiranto Herry Utomo, 2008. (c) Universitas Kristen
        Satya Wacana.";
```

#### 4. Simpulan

Pada tulisan ini dibahas implementasi aplikasi *grid* basis data dengan menggunakan framework OGSA-DAI. Beberapa pengujian telah dilakukan diantaranya adalah instalasi OGSA-DAI, *deploy data service*, *deploy data resource*, *expose data resource*, serta pengaksesan data melalui data *browser*. Selain itu sebuah aplikasi interface di sisi *client* telah dibuat untuk mengakses data dari *container web service* yang berisi *data resource*.

#### 5. Daftar Pustaka

- [1] Anonim.2006. *OGSA-DAI WSRF 2.2 User Guide*. The University of Edinburgh.
- [2] <http://www.ogsadai.org.uk>.