

STRATEGY TO INTEGRATE ACADEMIC DATA USING WEB SERVICES AND PL SQL (CASE STUDY TELKOM UNIVERSITY)

¹TORA FAHRUDIN

¹Computer Engineering, Telkom Applied Science School, Telkom University

E-mail: torafahrudin@telkomuniversity.ac.id

ABSTRACT

Established in 17 July 2013, Telkom University was formed from various faculties from Telkom Education Foundation, such as Faculty of Engineering (TES), Faculty of Management (TEBS), Faculty of Applied Sciences (TASS) and the Faculty of Creative Industries (TCIS). Each faculty has the Academic Information Systems. When it became a university, problems faced is how to do the integration strategy of academic data from the different data structure and data sources of academic information system at each faculty into one new scheme of University database. The challenge is the old academic system is still running for one semester, which means the data will still moving. Therefore, this study focuses on both strategic and technical concepts to integrate academic data base on that condition. A major strategy is using web services for data interchange and PL SQL Scheduler to process transaction data which still moving. Meanwhile, master data is loaded only once with flat files format. The conclusions obtained from this research are 1) to integrate academic data, logic data needed to be split into master data and transaction data, 2) the strategy for integrate master data are loaded only once with flat files format, meanwhile the transaction data is loaded many times via web services (JSON), 3) transaction data from web services will be loaded and mapping into destination database using JSON PL SQL package which run periodically.

Keywords: *Academic Data Integration; Web Services; JSON; PL SQL; DBMS Scheduler*

1. INTRODUCTION

According to a decree of the Minister of Education no 270/E/O/2013, 4 institutions under Telkom Education Foundation are allowed to merge as a Telkom University. The four institutions were Telkom Institute of Technology, Telkom Institute of Management, Telkom Polytechnics and Telkom Arts School [1]. Conceptually each institution will be a faculty, so Telkom University has four faculties, which are the Faculty of Engineering (TES), Faculty of Business Management (TEBS), Faculty of Applied Science (TASS) and the Faculty of Creative Industries (TCIS).

Each faculty has had academic information system, respectively. Each academic information system at the faculty has different data structure and application environment. When all institutions merge as Telkom University, data must be integrated, especially academic data. Based on the research of Jihon Chengxiu Zeng and Tang, there are three advantages of integrating data in the context of enterprise-level planning and decision making:

- (1) improved managerial information for organization-wide communication;
- (2) Improved operational coordination across sub-units or divisions of an organization; and
- (3) Improved organization-wide strategic planning and decision making [2]

Data integration is necessary for data to serve as a common language for communication within an organization [2].

The differences in the data environment and data structure of each faculty, the challenge is what strategy to integrate an academic data with the constraint that ongoing transaction should still run and not stopped by integration process.

Data need to be grouped logically into 2 main sections before integrated. Each group has its own different characteristics.

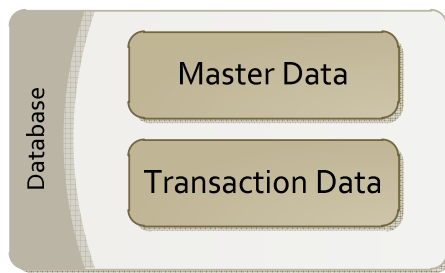


Figure 1: Data logic concept

Table 1: Different Characteristic of Master Data and Transaction Data

Characteristic	Master Data	Transaction Data
Record	Fix	Increased
Value	Not Change	Allow change
Size of data	small	Medium to big

Master data integration is performed once at the beginning of integration process. Meanwhile, active transaction data was loaded by scheduler any time periodically. But for non-active transaction data, integration process is performed only once time at the beginning same as master data. Data will be requested from each web services of faculty periodically by PL SQL and DBMS Scheduler and was processed to enter a University Database Scheme.

2. THEORITICAL BACKGROUND

2.1 Data Integration

Data integration is the process of standardization of data definitions and data structures by using common conceptual schema across a collection of data sources [3].

The Advanced Forest Technologies in Canada (AFT, 1997) identified the following factors which must be addressed to integrate data properly. Some of important factor [4]:

- Identification of an optimal subset of the available data sources for integration
- The formats of the data, the archive systems, and the data storage and retrieval
- The computational efficiency of the integrated data sets to achieve the goals of the users

2.2 Web Services

A web service is service that accepts a request and returns data or carries out a processing task [5]. The data returned a normally formatted in

a-machine-readable form [5]. In order to make these web-services available, there must be standards so that everyone knows what information can be requested, how to request it, and what form the response will take [5].

Web Service is an interface which implements business logic [6]. Performance is a one of the major problem in Web Services. [6]. Several standard formats for data document exchanged are JSON and XML. The results of research conducted by Nurzhan Nurseitov from Montana University, find that JSON is significantly faster than XML [7]. Therefore, by considering the speed of JSON, JSON more chosen as a data exchange format then XML.

2.2.2 JSON

JSON = JavaScript Object Notation [8]. JSON is designed to be data exchange language which is human readable and easy for computers to parse and use [8].

Example of JSON format

```
[{"ROOMID":"1","ROOMNAME":"A1","CAPACITY":"40"}, {"ROOMID":"2","ROOMNAME":"A2","CAPACITY":"20"}]
```

Above shows that there are 2 pieces of data (in the form of an array) where each data consists of three attributes : RoomID, roomname and capacity.

```
[
  - {
    ROOMID: "1",
    ROOMNAME: "A1",
    CAPACITY: "40"
  },
  - {
    ROOMID: "2",
    ROOMNAME: "A2",
    CAPACITY: "20"
  }
]
```

Figure 2: JSON View from Google Chrome with JSON Add On View Module

2.3 PL SQL

PL SQL can be used to implement business rules through the creation of stored procedures, functions and packages, triggers to trigger events and add programming logic to execution of SQL commands [9]. PL SQL divided into [10]:

- Anonymous Blocks

- Stored Program Units (procedures, functions, and packages)
- Triggers

2.4 DBMS Scheduler

The DBMS_SCHEDULER package provides a collection of scheduling functions and procedures that are callable from any PL/SQL program [11]. Scheduler has 3 important parts:

- Program : what action should be execute
- Schedule: when program should be execute
- Jobs : unique name from schedule job

```
BEGIN
  DBMS_SCHEDULER.CREATE_JOB (
    job_name => 'JOB_LOAD_STUDENT_TASS',
    job_type => 'PLSQL_BLOCK',
    job_action => 'BEGIN LOAD_STUDENT; END;',
    start_date => SYSTIMESTAMP,
    repeat_interval=>'FREQUENCY=DAILY',
    enabled => TRUE);
END;
```

Figure 3: DBMS Scheduler example

2.5 JSON Package

JSON Package is Free package from Jonas Krogsboell and Lewis R Cunningha (2009), that has been released under MIT License which supply package for parsing JSON (decode or encode) on Oracle environments [12].

```
set serveroutput on format wrapped;
declare
  obj json;
begin
  obj := json('{"a": true }');
  obj.print;
end;
```

Figure 4: JSON Package example (ex1.sql) [11]

3. INTEGRATION CONCEPT

3.1 Master Data and Transaction Data on Academic Environment

Master Data is conceptualized in the core entities of the enterprise [13]. The main characteristics of the data master are:

- Data change slowly
- Many applications or business processes that refer to the data
- Core element of business processes

While the transaction data, has the opposite characteristics.

Table 2: Grouping of Academic Entities

Entities	Master Data	Transaction Data
Student	V	
Lecturer	V	
Curriculum	V	
Rooms	V	
Shift	V	
Education Bill	V	
Faculty	V	
Study Program	V	
Student Registration Status		V
Student Payment		V
Student Study Card		V
Schedules		V
Lecturer Presence		V
Student Presence		V
Student Test Score		V

3.2 Data Integration Strategy

The possibility of insert, update and delete between master data and transaction data can be seen in the table below.

Table 3: Master Data versus Transaction Data

Data Type	Type Transaction	The Change of Business Process or the entry of new master data	Teaching and Learning Activities
Master Data	Insert	V	-
	Update	V	-
	Delete	V	-
Transaction Data	Insert	V	V
	Update	V	V
	Delete	V	V

From the table, it can be seen that the insert, update and delete on the master data only happens if there are any changes in business processes such as changes in curriculum or new entry data of master data such as new admissions, acceptance of new lecturer, and so on. Meanwhile in the data transactions, insert, update and delete will occur for all process in one semester.

Table 4: Insert Update and Delete of each Academic Process for one semester (in months)

Academic Process	Active Semester					
	1	2	3	4	5	6
Student Registration Status						
Student Payment						
Student Study Card						
Schedules						
Lecturer Presence						
Student Presence						
Student Test Score						

From the table above, it can be seen that the characteristics of academic transaction data is changing in a semester for each Academic Processes. The implication of the theory of master data and transaction of data is the difference in the strategy of integrating the data. For data that are not moving, the data is loaded only once. While the transaction data that still changing must be loaded by PL SQL Scheduler to process data with JSON data format.

Table 5: The different strategies of Master Data and Transaction Data

Data Type	Past Academic Calendars		2013-2014 (Active Academic Calendar)	
	Odd	Even	Odd	Even (Active Semester)
Master Data	- Once Load - Use Flat File Format			
Transaction Data	- Once Load - Use Flat File Format		- Data was Load via JSON format from each faculty which scheduled periodically	

Before the integration, it must create a database schema of Telkom University first. Data will be Load and mapped from each faculty to the new database schema of Telkom University.

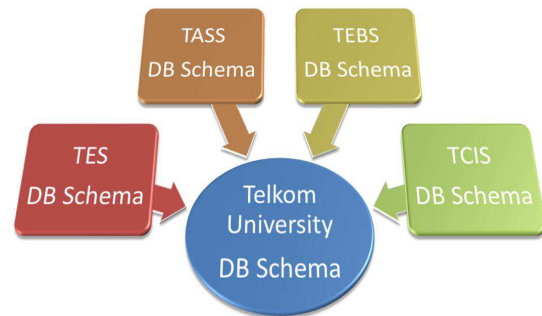


Figure 5: Data Load and Mapping into Telkom University Schema

3.3 Limitation Study

This research has some limitations and assumptions.

- Study did not discuss details of the mapping tables from the source database to the destination database
- It is assumed in the transaction data load is fixed, unchanged after migrate. If the data is already loaded in the faculty was changed, an adjustment at the end of the process is necessary to ensure valid data.
- Assumed no network failure during integration with PL SQL scheduler is running. If the networks fail, PL SQL can't process the transaction data that day. So the data need to be adjusting again.
- Accuracy of the data migration results are not discussed in this paper

4. IMPLEMENTATION RESULT

4.1 Load Data Master

Master data is loaded only one time using Flat File format. Each faculty must export master data on Flat File format. After that, data are mapped into a Telkom University Schema.

4.2 Web Services for Transaction Data

Each faculty creates a Web Services for each transaction type according to Telkom University Schema. Web Services can be accessed from the IP server only. Web services have different URL address to obtain a variety of different transaction data as needed.

Table 6: URL Web Services for different transaction

Transaction	URL
Student Registration Status	http://facultydomain/api/type=registration
Student Payment	http://facultydomain/api/type=payment
Student Study Card	http://facultydomain/api/type=studentstudyocard
Schedules	http://facultydomain/api/type=schedules
Lecturer Presence	http://facultydomain/api/type=lecturespresence
Student Presence	http://facultydomain/api/type=studentpresence
Student Test Score	http://facultydomain/api/type=studenttestscore

4.3 PL SQL

Transaction data from web services each faculty, were processed by PL SQL scripts which are embedded in Telkom University Database Schema. 2 Package were needed, 1) Package from Jonas Krogsboell and Lewis R Cunningha (2009) for parsing JSON data, 2) UTL HTTP package to get web page content from Web services.

Table 7: Procedure PL SQL Each Transaction

Transaction	Procedure PL SQL
Student Registration Status	PROC_LOAD_STU_REG
Student Payment	PROC_LOAD_TRANSACTION
Student Study Card	PROC_LOAD_STUDY_CARD
Schedules	PROC_LOAD_SCHED
Lecturer Presence	PROC_LOAD_LEC
Student Presence	PROC_LOAD_STU_PRE
Student Test Score	PROC_LOAD_STU_SCORE

For each PL SQL Procedure, there are four procedures representing each faculty (TES, TEBS, TASS, and TCIS).

```
CREATE OR REPLACE PROCEDURE PROC_LOAD_STU_REG
IS
BEGIN
    PROC_LOAD_STU_REG_TES;
    PROC_LOAD_STU_REG_TEBS;
    PROC_LOAD_STU_REG_TASS;
    PROC_LOAD_STU_REG_TCIS;
    COMMIT;
END;
```

Figure 6: Load Student Registration Procedure PL SQL

4.4 SCHEDULER

Scheduler is used to run a PL SQL procedure in a particular period. Sequences of PL SQL procedure will be adjusted manually based on sequence of process. Examples, Student Presence will be processed after the PL SQL procedure process the Lecturer Presence.

Table 8: Schedule of PL SQL procedure each transaction

Transaction	Loaded Time
Student Registration Status	One Time After Registration Change Schedules Finish
Student Payment	One Time After Registration Change Schedules Finish
Student Study Card	One Time After Registration Change Schedules Finish
Schedules	One Time After Registration Change Schedules Finish
Lecturer Presence	Every day at 22:00
Student Presence	Every day at 23:00
Student Test Score	One Time After Final grades Submission Deadline

5. SUMMARY

5.1 Conclusions

The conclusions obtained from this research is

1. To integrate academic data, logic data needed to be split into master data and transaction data
2. The strategy for integrate master data are loaded only once with flat files format, meanwhile the transaction data is loaded many times via web services (JSON)
3. Transaction data from web services will be loaded and mapping into new scheme database using JSON PL SQL package which run periodically.

REFERENCES:

- [1]<http://www.telkomuniversity.ac.id/page/history>
(accessed on April, 2014)
- [2]Zeng, Jihong; Tan, Chengxiu. “*Overview of Research and Practices in Information Sharing for Enterprise Resource Planning*”. Journal of Applied Business and Economics.
- [3]Heimbigner, D. & McLeod, D. (1985). “*A Federated Architecture for Information Management, ACM Transactions on Office Information Systems*”, 3, (3), 253-278.
- [4] Research and Practical Experiences in the Use of Multiple Data Sources. www.ctg.albany.edu. (2003). (accessed from http://www.ctg.albany.edu/publications/reports/multiple_data_sources?chapter=5&PrintVersion=2, on April, 2014)
- [5]Hunter, David, et al. (2007). “*Beginning XML. 4th Edition*”. Indiana: Wiley Publishing, Inc.
- [6] Raedy, Mohan Ch Ram; Geetha, D Evanglin; Srinivasa, KG. “*Early Performance Prediction Of Web Services*”. International Journal on Web Service Computing (IJWSC), Vol. 2, No. 3, September. (2011)
- [7]Nurseitov Nurzhan, Paulson Michael, Reynolds Randall, Izurieta Clemente. (2009). “*Comparison of JSON and XML Data Interchange Formats: A case study*”. Department of Computer Science Montana State University – Bozeman, Montana, USA.
- [8]JSON. JSON. [json.org](http://www.json.org). <http://www.json.org> (accessed on April, 2014)
- [9]http://docs.oracle.com/cd/B19306_01/appdev.102/b14261.pdf (accessed on June, 2014)
- [10]http://docs.oracle.com/cd/B19306_01/appdev.102/b14251/adfns_packages.htm (accessed on June, 2014)
- [11]http://docs.oracle.com/cd/B19306_01/appdev.102/b14258/d_sched.htm (accessed on June, 2014)
- [12]Krogsboll Jonas, Lewis R Cunningham. (2009). “*PL / JSON Reference Guide (version 1.0.4) For Oracle 10g and 11g*”
- [13]<http://www.semarchy.com/en/overview/what-is-master-data/> (accessed on June, 2014)