

LAMPIRAN

1. STAGE-ODS

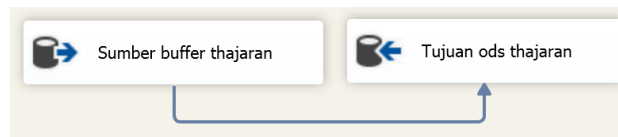
a. Ods.faculty



Available Input Columns		Available Destination Columns	
Name		Name	
FACULTY_ID		faculty_id	
NAME_OF_FACULTY		name_of_faculty	
FACULTY_ID_HURUF		faculty_id_huruf	
NAMA_FAKULTAS_UJAZAH			
NAMA_FAKULTAS_UJAZAH_ENG			

Input Column	Destination Column
FACULTY_ID	faculty_id
NAME_OF_FACULTY	name_of_faculty
FACULTY_ID_HURUF	faculty_id_huruf

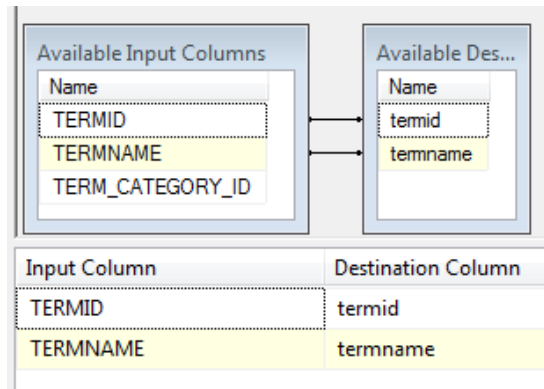
b. Ods.thajaran



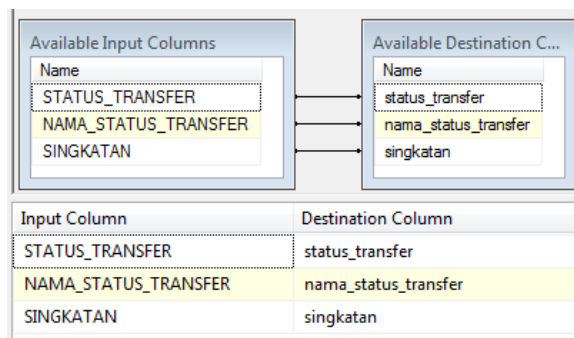
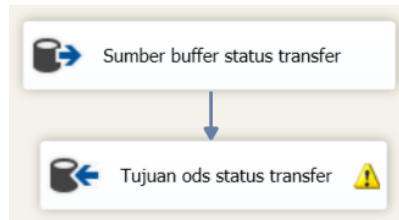
Available Input Columns		Available Destination Columns	
Name		Name	
THAJARANID		thajaranid	
STARTDATE		startdate	
ENDATE		enddate	
DESCRIPTION		description	

Input Column	Destination Column
THAJARANID	thajaranid
STARTDATE	startdate
ENDATE	enddate
DESCRIPTION	description

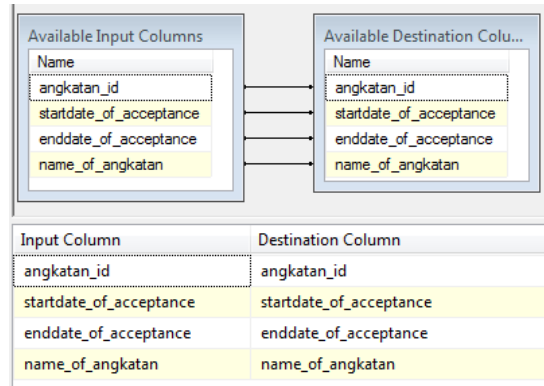
c. Ods.term



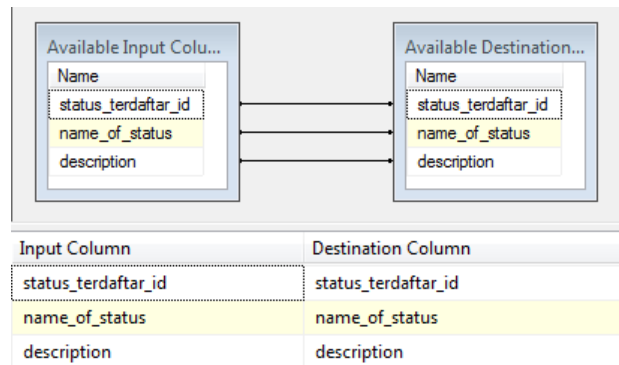
d. Ods.status_transfer



e. Ods.angkatan

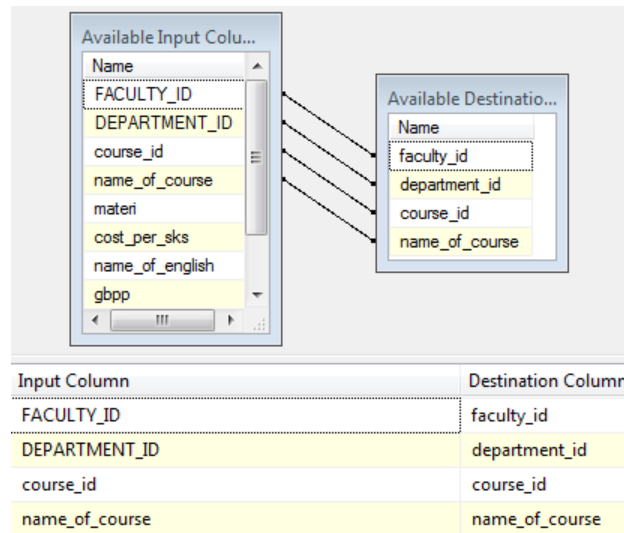


f. Ods.status_terdaftar

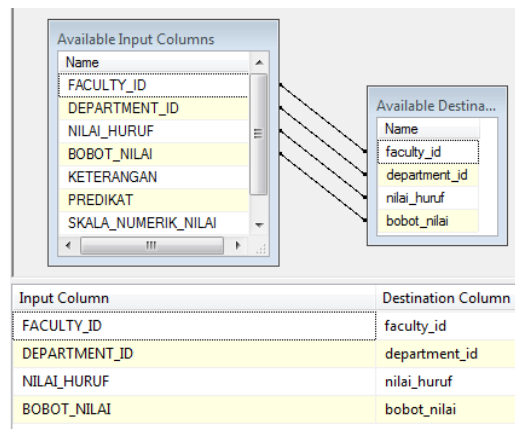


g. Ods.course



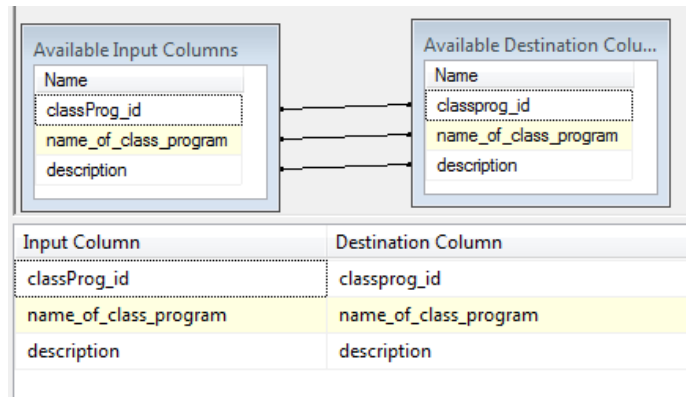


h. Ods.nilai_huruf

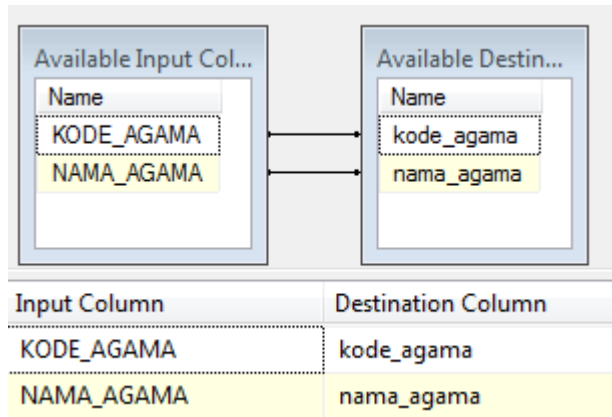


i. Ods.class_program

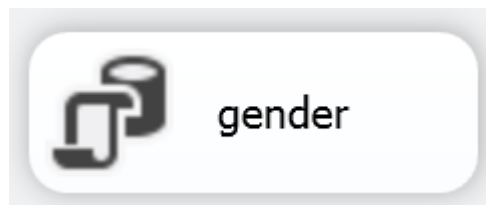




j. Ods.Agama



k. Ods.gender



```
insert into [ods.gender] values(0,'-');
insert into [ods.gender] values(1,'Perempuan');
insert into [ods.gender] values(2,'Laki-Laki');
```

2. ODS-DDS

```

CREATE TABLE [dim_status_terdaftar] (
  status_terdaftar_key SMALLINT NOT NULL IDENTITY(0,1),
  status_terdaftar_id TINYINT NOT NULL,
  name_of_status VARCHAR(38) NULL,
  description VARCHAR(2) NULL,
  PRIMARY KEY(status_terdaftar_key)
);

CREATE TABLE [dim_status_mahasiswa] (
  status_mahasiswa_key SMALLINT NOT NULL IDENTITY(0,1),
  status_mahasiswa_id TINYINT NOT NULL,
  status_mahasiswa VARCHAR(5) NULL,
  PRIMARY KEY(status_mahasiswa_key)
);

CREATE TABLE [dim_nilai_huruf] (
  nilai_huruf_key SMALLINT NOT NULL IDENTITY(0,1),
  faculty_id TINYINT NOT NULL,
  department_id SMALLINT NOT NULL,
  nilai_huruf VARCHAR(2) NOT NULL,
  bobot_nilai DECIMAL(5,2) NULL,
  PRIMARY KEY(nilai_huruf_key)
);

CREATE TABLE [dim_thajaran] (
  thajaran_key smallint NOT NULL IDENTITY(0,1),
  thajaranid SMALLINT NOT NULL,
  startdate DATETIME NULL,
  enddate DATETIME NULL,
  description VARCHAR(10) NULL,
  PRIMARY KEY(thajaran_key)
);

CREATE TABLE [dim_term] (
  term_key SMALLINT NOT NULL IDENTITY(0,1),
  termid TINYINT NOT NULL,
  termname VARCHAR(15) NULL,
  PRIMARY KEY(term_key)
);

CREATE TABLE [dim_status_transfer] (
  status_transfer_key SMALLINT NOT NULL IDENTITY(0,1),
  status_transfer TINYINT NOT NULL,
  nama_status_transfer VARCHAR(27) NULL,
  singkatan VARCHAR(10) NULL,
  PRIMARY KEY(status_transfer_key)
);

```

```
CREATE TABLE [dim_class_program] (  
    classprog_key SMALLINT NOT NULL IDENTITY(0,1),  
    classprog_id TINYINT NOT NULL,  
    name_of_class_program VARCHAR(21) NULL,  
    description VARCHAR(1) NULL,  
    PRIMARY KEY(classprog_key)  
);  
  
CREATE TABLE [dim_angkatan] (  
    angkatan_key SMALLINT NOT NULL IDENTITY(0,1),  
    angkatan_id SMALLINT NOT NULL,  
    startdate_of_acceptance DATETIME NULL,  
    enddate_of_acceptance DATETIME NULL,  
    name_of_angkatan VARCHAR(9) NULL,  
    PRIMARY KEY(angkatan_key)  
);  
  
CREATE TABLE [dim_mahasiswa] (  
    student_key INTEGER NOT NULL IDENTITY(0,1),  
    studentid VARCHAR(15) NOT NULL,  
    fullname VARCHAR(50) NULL,  
    placeofbirth VARCHAR(50) NULL,  
    dateofbirth DATETIME NULL,  
    pembayar VARCHAR(15) NULL,  
    telp VARCHAR(30) NULL,  
    email VARCHAR(60) NULL,  
    nama_gender VARCHAR(9) NULL,  
    nama_agama VARCHAR(7) NULL,  
    PRIMARY KEY(student_key)  
);
```

```

CREATE TABLE [dim_faculty] (
    faculty_key SMALLINT NOT NULL IDENTITY(0,1),
    faculty_id TINYINT NOT NULL,
    name_of_faculty VARCHAR(17) NULL,
    faculty_id_huruf VARCHAR(4) NULL,
    PRIMARY KEY(faculty_key)
);

CREATE TABLE [dim_department] (
    department_key SMALLINT NOT NULL IDENTITY(0,1),
    department_id SMALLINT NOT NULL,
    name_of_department VARCHAR(38) NULL,
    department_id_huruf VARCHAR(4) NULL,
    PRIMARY KEY(department_key)
);

CREATE TABLE [dim_course] (
    course_key INTEGER NOT NULL IDENTITY(0,1),
    course_id VARCHAR(12) NOT NULL,
    faculty_id TINYINT NOT NULL,
    department_id SMALLINT NOT NULL,
    name_of_course VARCHAR(82) NULL,
    PRIMARY KEY(course_key)
);

CREATE TABLE [fact_perkuliahan] (
    student_key INTEGER NOT NULL,
    angkatan_key SMALLINT NOT NULL,
    thajaran_key smallint NOT NULL,
    term_key SMALLINT NOT NULL,
    faculty_key SMALLINT NOT NULL,
    department_key SMALLINT NOT NULL,
    classprog_key SMALLINT NOT NULL,
    status_terdaftar_key SMALLINT NOT NULL,
    status_transfer_key SMALLINT NOT NULL,
    status_mahasiswa_key SMALLINT NOT NULL,
    course_key INTEGER NOT NULL,
    nilai_huruf_key SMALLINT NOT NULL,
    sks DECIMAL(3,1) NULL,
    bobot_nilai DECIMAL(5,2) NULL,
    IP DECIMAL(5,2) NULL,
    IPK DECIMAL(5,2) NULL,
    constraint fk_mahasiswa foreign key (student_key) references dim_mahasiswa(student_key)
    ,constraint fk_angkatan foreign key (angkatan_key) references dim_angkatan(angkatan_key)
    ,constraint fk_thajaran foreign key (thajaran_key) references dim_thajaran(thajaran_key)
    ,constraint fk_term foreign key (term_key) references dim_term(term_key)
    ,constraint fk_faculty foreign key (faculty_key) references dim_faculty(faculty_key)
    ,constraint fk_department foreign key (department_key) references dim_department(department_key)
    ,constraint fk_classprog foreign key (classprog_key) references dim_class_program(classprog_key)
    ,constraint fk_status_terdaftar foreign key (status_terdaftar_key) references dim_status_terdaftar(status_terdaftar_key)
    ,constraint fk_status_transfer foreign key (status_transfer_key) references dim_status_transfer(status_transfer_key)
    ,constraint fk_status_mahasiswa foreign key (status_mahasiswa_key) references dim_status_mahasiswa(status_mahasiswa_key)
    ,constraint fk_course foreign key (course_key) references dim_course(course_key)
    ,constraint fk_nilai_huruf foreign key (nilai_huruf_key) references dim_nilai_huruf(nilai_huruf_key)
);

```


3. Penggunaan kolom dari tahap stage hingga DDS

a. Tabel faculty

STAGE	ODS	DDS
-	-	Faculty_key
OBJECT_CATEGORY_ID	-	-
FACULTY_ID	faculty_id	faculty_id
HIRARKI_ID	-	-
NAME_OF_FACULTY	name_of_faculty	name_of_faculty
FACULTY_ID_HURUF	faculty_id_huruf	faculty_id_huruf
NAMA_FAKULTAS_IJAZAH	-	-
NAMA_FAKULTAS_IJAZAH_ENG	-	-

b. Tabel department

STAGE	ODS	DDS
-	-	department_key
OBJECT_CATEGORY_ID	-	-
FACULTY_ID	-	-
DEPARTMENT_ID	department_id	department_id
HIRARKI_ID	-	-
NAME_OF_DEPARTMENT	name_of_department	name_of_department
DEPARTMENT_ID_HURUF	department_id_huruf	department_id_huruf

c. Tabel term

STAGE	ODS	DDS
-	-	term_key
TERM_CATEGORY_ID	-	-
TERMID	termid	termid
TERMNAME	termname	termname

d. Tabel thajaran

STAGE	ODS	DDS
-	-	thajaran_key
THAJARANID	thajaranid	thajaranid
STARTDATE	startdate	startdate
ENDATE	enddate	enddate
DESCRIPTION	description	description

e. Tabel angkatan

STAGE	ODS	DDS
-	-	angkatan_key
angkatan_id	angkatan_id	angkatan_id
thajaranid	-	-
Startdate_of_acceptance	Startdate_of_acceptance	Startdate_of_acceptance
Endate_of_acceptance	Endate_of_acceptance	Endate_of_acceptance
Name_of_angkatan	Name_of_angkatan	Name_of_angkatan

f. Tabel class program

STAGE	ODS	DDS
-	-	classprog_key
classProg_id	classProg_id	classprog_id
name_of_class_program	name_of_class_program	name_of_class_program
description	description	description

g. Tabel status_transfer

STAGE	ODS	DDS
-	-	status_transfer_key
STATUS_TRANSFER	status_transfer	status_transfer
NAMA_STATUS_TRANSFER	nama_status_transfer	nama_status_transfer
SINGKATAN	singkatan	singkatan

h. Tabel status_terdaftar

STAGE	ODS	DDS
-	-	status_terdaftar_key
status_terdaftar_id	status_terdaftar_id	status_terdaftar_id
name_of_status	name_of_status	name_of_status
description	description	description

i. Tabel agama

STAGE	ODS	DDS
-	-	Join dengan tabel dim_mahasiswa
KODE_AGAMA	kode_agama	
NAMA_AGAMA	nama_agama	
Lain	-	

j. Tabel gender

STAGE	ODS	DDS
-	-	Join dengan tabel dim_mahasiswa
GENDER	gender	
DESKRIPSI	deskripsi	
Lain	-	

k. Tabel course

STAGE	ODS	DDS
-	-	course_key
FACULTY_ID	faculty_id	faculty_id
DEPARTMENT_ID	department_id	department_id
course_id	course_id	course_id
name_of_course	name_of_course	name_of_course
Materi	-	-
cost_per_sks	-	-
name_of_english	-	-
Gbpp	-	-
course_type_id	-	-

l. Tabel nilai huruf

STAGE	ODS	DDS
-	-	Nilai_huruf_key
FACULTY_ID	faculty_id	faculty_id
DEPARTMENT_ID	department_id	department_id
NILAI_HURUF	nilai_huruf	nilai_huruf
BOBOT_NILAI	bobot_nilai	bobot_nilai
KETERANGAN	-	-
PREDIKAT	-	-
SKALA_NUMERIK_NILAI	-	-
SKALA_NUMERIK_NILAI_LB	-	-

m. Tabel mahasiswa

STAGE	ODS	DDS
-	-	mahasiswa_key
FACULTY_ID	faculty_id	faculty_id
DEPARTMENT_ID	department_id	department_id
STUDENTID	Studentid	Studentid
CAMARUID	-	-
ANGKATAN_ID	Angkatan_id	-
THAJARANID	Thajaranid	-
TERMID	Termid	

STATUS_TERDAFTAR_ID	Status_terdaftar_id	-
AVAILABLE_EDU_ID	-	-
TOTAL_SCORE	-	-
ENTRY_PERIOD_ID	-	-
SPONSOR_ID	-	-
FULLNAME	Fullname	Fullname
NICKNAME	-	-
GENDER	Gender	Nama_gender
PLACEOFBIRTH	Placeofbirth	Placeofbirth
DATEOFBIRTH	Dateofbirth	Dateofbirth
KODE_AGAMA	Kode_agama	Nama_agama
MARITALSTATUSID	-	-
NBSPOUSE	-	-
NBCHILDREN	-	-
DATE_OF_ACCEPTANCE	-	-
DATE_OF_FIRST_REGISTERED	-	-
DATE_OF_COMPLETION	-	-
LENGTH_OF_STUDY	-	-
MODIFIED_BY	-	-
MODIFIED_DATE	-	-
PICTUREFILE	-	-
JOB_ID	-	-
BLOOD_ID	-	-
CLASSPROG_ID	Classprog_id	-
CONCENTRATION_ID	-	-
KEWARGANEGARAAN_ID	-	-
STRATA	-	-
JURUSAN_SMA	-	-
IS_AKTIF	-	-
Pembayar	pembayar	-
Date_of_do	-	-
Donation	-	-
Entry_period_type	-	-
Kota_lahir	-	-
Prop_lahir	-	-
Neg_lahir	-	-
REG_DATE	-	-
REG_TIME	-	-
REG_OFFICER	-	-
TELP	Telp	Telp
STATUS_TRANSFER	Status_transfer	-
KURIKULUM_ID	-	-
EMAIL	Email	Email

INDEX_CLASS_ID	-	-
-	Status_mahasiswa_id	-

n. Tabel student_course_krs

STAGE	ODS	DDS
Studentid	faculty_id	Join menjadi fact_perkuliahan
Termid	Termid	
Thajaranid	Thajaranid	
course_id	course_id	
Faculty_id	Faculty_id	
Department_id	Department_id	
Available_edu_id	-	
Index_class_id	-	
IsBaru	-	
Nb_pengambilan	-	
Course_grade	-	
Nilai_huruf	-	
IsNilaiTetap	-	
Date_KRS	-	
Classprog_id	-	
Modified_by	-	
Modified_date	-	
Sks	-	
Amount	-	
isRequired	-	
Due_date	-	
Biaya_id	-	

o. Tabel student_payment

STAGE	ODS	DDS
Thajaranid	thajaranid	Join menjadi fact_perkuliahan
Termid	Termid	
faculty_id	-	
department_id	-	
Pembayar	Pembayar	
Transaction_ke	-	
Faculty_id	-	
Department_id	-	
available_edu_id	-	
Cost_id	-	
Payment_amount	-	
Ispembayaran	-	
isangsuran	-	

Islunas	-	
Description	-	
Cashier_id	-	
Angsuran_ke	-	
Amount_due	-	
Discount	-	
Quantity	-	
Satuan	-	
Date_of_payment	-	
Modified_date	-	
Modified_by	-	
isdispensasi	-	
Isuang	-	
Dispensasi_id	-	
Amount_dispensasi	-	
Disc_type	-	
Start_dispensasi	-	
End_dispensasi	-	
Branch_id	-	
Idbank	-	
Iscetak	-	
Print_date	-	
Name_of_cost	-	
REFP_PAYMENT_ID	-	
Persen_angsuran	-	
Accountnumber	-	
Thajaranasal	-	
Term_asal	-	
JENIS	-	
ADJUSTMENT_AMOUNT	-	
Date_copy	-	
Retur_id	-	
Jml_keyin	-	
Jml_telah_retur	-	
Persen_retur	-	

p. Tabel transkrip

STAGE	ODS	DDS
faculty_id	-	Join menjadi fact_perkuliahan
department_id	-	
available_edu_id		
Studentid		
Course_id		
Nilai_huruf		

Nilai_angka		
thajaranid	thajaranid	
Termid	Termid	
Hasil_ujian_ke	-	
Modified_by	-	
Modified_date	-	
Sks	-	
Istransfer	-	
Bobot_nilai	-	
isPakai	-	
Keterangan	-	
isRequired	-	
Prasyarat	-	

4. Jumlah data

Tabel	Stage	ODS	DDS
Faculty	11	11	11
Department	45	45	45
Mahasiswa	52732	52710	52710
Status_terdaftar	7	7	7
Course	5725	5695	5696
nilai_huruf	240	240	241
Term	6	6	6
Thajaran	16	16	16
Status_transfer	5	5	5
Class_program	5	5	5
Angkatan	30	30	30
student_payment	1434472	150386	1067469
student_course_krs	2128744	1033974	
Transkrip	1439846	607202	

5. Analisis basis data

a. *Trailing Space*

Dalam pelaksanaan *JOIN* data didapatkan suatu kendala yaitu hilangnya data yang memiliki nilai kolom sama antar tabel. Saat dilakukan *query* secara manual dengan cara mengetikkan sendiri nilai dari suatu kolom, maka data yang hilang dapat muncul. Namun ketika dilakukan secara otomatis menggunakan SSIS data menghilang. Setelah dikaji ulang ternyata

ditemukan data yang memiliki *trailing space*. Adapun cara pencarian *trailing space* ditunjukkan pada gambar pertama dan hasil di gambar kedua. Dari hasil gambar kedua dapat dilihat terdapat jarak antara digit terakhir dan tanda petik. Hal inilah yang menyebabkan data hilang.

```
select * from [buffer_transkrip]
  where CHARINDEX(' ', studentid) > 0;


select * from [buffer_student_course_krs]
  where CHARINDEX(' ', studentid) > 0;

select * from [buffer_student_payment]
  where CHARINDEX(' ', pembayar) > 0;

select * from [buffer_transkrip]
  where CHARINDEX(' ', nilai_huruf) > 0;
```

Gambar 1

```
'20070530178 '
'1997120053 '
'0600310033 '
'0899820137 '
'1299320180 '
' A '
' B - '
' A '
```



Gambar 2

b. *Constraint*


Terdapat kasus bahwa nilai suatu kolom pada tabel transaksional tidak tersedia pada tabel master. Hal ini menyebabkan masalah ketika menerapkan *foreign key*. Adapun hasilnya sebagai berikut:

faculty_id	department_id	nilai_huruf
4	41	T
4	41	T
4	41	T




FACULTY_ID	DEPARTMENT_ID	NILAI_HURUF
4	41	A
4	41	B
4	41	C
4	41	D
4	41	E
4	41	TL

faculty_id	department_id	nilai_huruf
20	201	C
20	201	C
20	201	C



FACULTY_ID	DEPARTMENT_ID	NILAI_HURUF
20	201	A
20	201	A-
20	201	B
20	201	B+
20	201	B-

faculty_id	department_id	nilai_huruf
8	81	4



FACULTY_ID	DEPARTMENT_ID	NILAI_HURUF
8	81	A
8	81	A-
8	81	B
8	81	B+
8	81	B-
8	81	C
8	81	C+
8	81	D
8	81	E
8	81	K

Pada bagian kiri adalah hasil dari tabel transaksional dan sebelah kanan merupakan tabel master. Dapat dilihat bahwa nilai pada kolom nilai_huruf pada tabel transaksional tidak dimiliki oleh kolom nilai_huruf pada tabel master.



AMERICAN
SCIENTIFIC
PUBLISHERS

Copyright © 2011 American Scientific Publishers
All rights reserved
Printed in the United States of America

Advanced Science Letters
Vol. 4, 400–407, 2011

Building a Data Mart Using Single Dimensional Data Store Architecture with Student Subject: Case Study at Muhammadiyah University of Yogyakarta

Fajar Rianda, Asroni, Ronald Adrian
Muhammadiyah University of Yogyakarta, Indonesia.

Muhammadiyah University of Yogyakarta have an academic information system. This system is an Online Transaction Processing (OLTP). System have data which related with student and can used for accreditation. But, system have not capability to give information quickly and accurately. Data warehouse is a system which have capability to give information quickly and accurately. With this ability to easily get related information students. This research using Single Dimensional Data Store (DDS) architecture. The goal research is build student data mart on data warehouse of Muhammadiyah University of Yogyakarta using single DDS architecture and resulting information quickly and accurately for supporting borang accreditation. Data mart has been built give ease to get information about student and the information can display in tabular.

Keywords: Data Warehouse, Data Mart, Borang Accreditation, Student, Single DDS

1. INTRODUCTION

A. Background

Every college have to accredited by Badan Akreditasi Nasional – Perguruan Tinggi (BAN-PT). Accreditation is used to determine eligibility and quality of college. BAN-PT will accrediting using borang accreditation form.

Muhammadiyah University of Yogyakarta (UMY) as one of the college who have to fill in the accreditation. UMY have an academic information system. This system is an Online Transaction Processing (OLTP). System have data which related with student and can used for accreditation. But, system have not capability to give information quickly and accurately.

The solution for that case is data warehouse. Data warehouse is a system which have capability to give information quickly and accurately. With this ability to easily get related information students.

B. Formulation of the problem

Formulation of the problem based on the background above are how to building data mart on data warehouse with single dimensional data store architecture.

C. Goal

The goal of this research is to build student data mart at of Muhammadiyah University of Yogyakarta using single DDS architecture and resulting information quickly and accurately for supporting borang accreditation.

2. THEORITICAL

A. Literature Review

The Research related about data warehouse has been done several times. Some research as a reference for this research is:

- Windarto has done research that entitle Pemanfaatan Data Warehouse sebagai sarana Penunjang Penyusunan Borang Akreditasi Standar 3 pada Fakultas Teknologi Informasi Universitas Budi Luhur [1]. His conclusion research is using data warehouse give ease to get information more effective and increase data security.
- Mukhlis Febriady and Bayu Adhi Tama have done research that entitle Rancang Bangun Data Warehouse untuk Menunjang Evalusai Akademik di Fakultas [2]. Their research resulting integrated database and make report faster.
- Other research from Armadiyah Amborowati that entitle Perancangan dan Pembuatan Data Warehouse pada Perpustakaan STMIK AMIKOM Yogyakarta [3]. Her research made load process periodic automatically and help administrator.

B. Data Warehouse

Data Warehouse is a subject-oriented, integrated, time-variant, nonupdatable collection of data used in support of management decision-making processes and business

intelligence [4]. The meaning of each of the key terms in this definition follows [5]:

- Subject-oriented, a data warehouse is organized around the key subjects [5]. Major subjects may include students, patients, and products [5].
- Integrated, the source data from different source systems. The source data usually inconsistent. The integrated source system must be made consistent [6].
- Time-Variant, Data in the warehouse is only accurate and valid at some point in time or over some time interval [6].
- Nonupdatable, Data in the data warehouse are loaded and refreshed from operational system, but cannot be updated by end users [5].

C. Online Transaction Processing (OLTP)

Online Transaction Processing is a system whose main purpose is to capture and store the business transactions [7]. OLTP system contain the data that will load into the data warehouse [7].

D. Difference of OLTP and Data Warehouse

Table 1. Difference of OLTP and Data Warehouse [6]

OLTP	Data Warehouse
Holds current data	Hold historical data
Data is dynamic	Data is largely static
Transaction-driven	Analysis-driven
Application-oriented	Subject-oriented
Supports day-to-day decision	Supports strategic decisions

E. Data Mart

Data mart is a subset of a data warehouse that supports the requirements of a particular department or business function [6].

F. Extract, Transform, and Load (ETL)

ETL is a system that has the capability to connect to the source systems, read the data, transform data, and load it into a target system [7].

G. Dimensional Data Store

Dimensional Data Store is a database that stores the data warehouse data in a different format than OLTP [7]. DDS is a better format to store data in the warehouse for the purpose of querying, analyzing data and gives better query performance [7].

H. Dimensional Modelling (Star Schema)

A Star schema is a simple database design in which dimensional data are separated from fact or event data [5]. A star schema consist of two types of tables: fact tables and dimension tables. Star schema is simpler than other

schema, and making easier for ETL process data into DDS [7].

I. Single Dimensional Data Store (DDS) architecture

In Single DDS, consist of only two data stores: stage and DDS. Single DDS architecture use two data store, which are [7]:

- A stage, is a place where you store the data you extracted from the store system temporarily, before processing it further or load into to other data store.
- Dimensional Data Store (DDS), is a user-facing data store, where the data is arranged in dimensional format for purpose of supporting analytical queries [7].

Single DDS is simpler because the data from the stage is loaded straight into the dimensional data store, without going to any kind of normalized store first [7].

3. METHOD

In this research there are some steps. The steps are display in Figure 1:



Figure 1. Research Procedure

A. Requirement

Requirement is first step in this research. Requirement will determined as basis for the building a data mart.

B. Building

In this research using single DDS architecture. In single DDS there are some step to build. Single DDS architecture display in Figure 2[7].

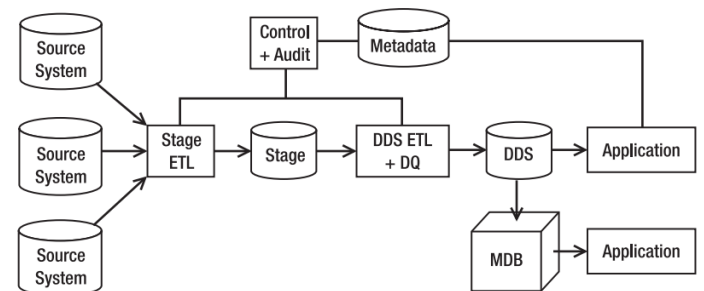


Figure 2. Single DDS

In Figure 2, the first step is determine source system which will use for building data mart. After source system have determined, the next step is ETL process for load data from source system into stage data store. After data have been loaded in stage data store, the next step is ETL process again and data quality (DQ). In second ETL process data will stored in dimensional data store with star schema structure. Star schema display in Figure 3:

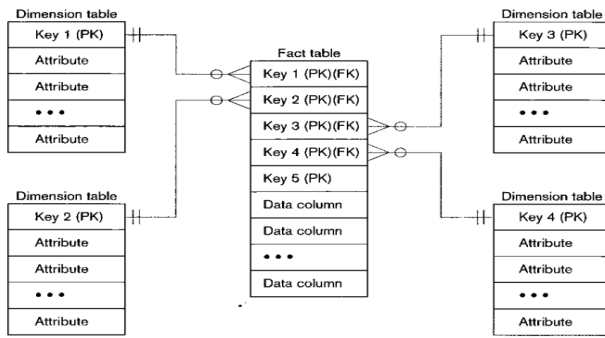


Figure 3. Star Schema

C. Testing

There are two testing using in this research, which are:

- ETL testing, in this testing make sure that appropriate changes in the source system are loaded correctly into data mart.
- Functional testing, in the functional testing verify make sure all requirements are fulfilled by data mart.

4. RESULT AND EXPLANATION

A. Requirement

The requirement data in this research obtained from borang accreditation document standard 3.

B. Building

1. Source system

The source system that used is master tabel database. Data that taken from the students of 2000 to 2014. In master tabel database there are some tables that used for this research. The following tables are:

- dbo.MAHASISWA
- dbo.STATUS_TERDAFTAR
- dbo.STATUS_TRANSFER
- dbo.THAJARAN
- dbo.FACULTY
- dbo.DEPARTMENT
- dbo.ANGKATAN
- dbo.CLASS_PROGRAM

as for the figure relation between tables above as follows :

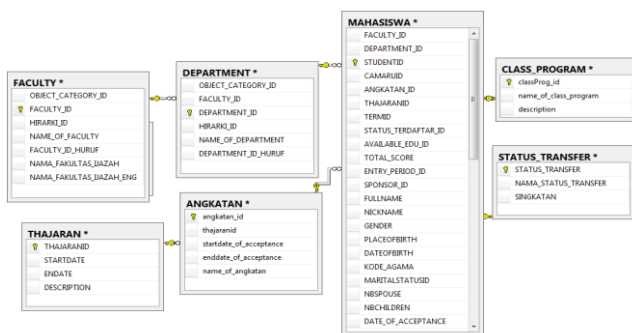


Figure 4. Source system

2. Extract, Transform, and Load (ETL)

3

In first ETL process, source system will stored in stage data store and change the name of tables. Table 2 displaying tables name changes in stage data store.

Table 2. Name Changes from Source system to Stage

Source system	Stage Data Store
dbo.MAHASISWA	dbo.stage.mahasiswa
dbo.STATUS_TERDAFTAR	dbo.stage.status_terdaftar
dbo.STATUS_TRANSFER	dbo.stage.status_transfer
dbo.THAJARAN	dbo.stage.thnajaran
dbo.DEPARTMENT	dbo.stage.department
dbo.FACULTY	dbo.stage.faculty
dbo.ANGKATAN	dbo.stage.angkatan
dbo.CLASS_PROGRAM	dbo.stage.class_program

After data have stored in stage data store, the next step is ETL process. In second ETL process, data will stored from stage to dimension data store. Beside store data, in this step will do data quality. Data quality make sure data that stored in dimensional data store are not dirty data or noise. Dirty data or noise can include null data, duplicate data. In this step, name of tables will change. For dbo.stage.faculty table and dbo.stage.department table merged and became dbo.stage.dim_faculty_department. Table 3 displaying tables name changes in DDS:

Table 3. Name Changes from Stage Store to DDS

Stage Data Store	Dimensional Data Store
dbo.stage.mahasiswa	dbo.dim_mahasiswa
dbo.stage.status_terdaftar	dbo.dim_status_terdaftar
dbo.stage.status_transfer	dbo.dim_status_transfer
dbo.stage.thnajaran	dbo.dim_thnajaran
dbo.stage.angkatan	dbo.dim_angkatan
dbo.stage.class_program	dbo.dim_class_program

That tables are dimension table. In DDS, there are one fact table. The table is dbo.fact_jumlah_mahasiswa. Dimension table and fact table made with star schema structure. Figure star schema display in Figure 5 :

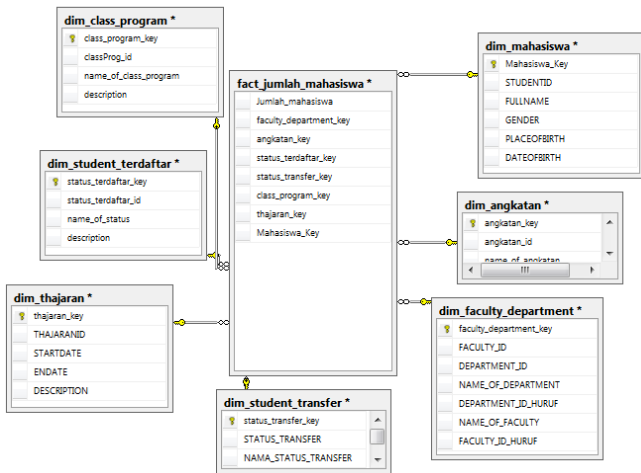


Figure 5. Star Schema

As the result of database architecture that have made from source system to dimensional data store is shown in the following figure :

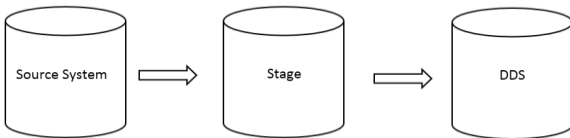


Figure 6. Database Architecture

C. Testing

1. ETL Testing

In this testing will make sure that data in ETL process have stored to data store. The results of testing display in Table 4.

Table 4. ETL Testing

Data store \ Year	Stage Store	DDS
2014	6380	6380
2013	7003	7003
2012	5391	5391
2011	3912	3912
2010	2714	2714
2009	2174	2174
2008	1878	1878
2007	1916	1916
2006	1615	1615
2005	2150	2150
2004	2467	2467
2003	2515	2515
2002	3174	3174

2001	3061	3061
2000	2946	2946

2. Functional Testing

This testing make sure that data mart have capability to fulfill requirement of borang accreditation document. Table 5 display result of testing.

Table 5. Student Count

Academic periodic	Student Count	
	Regular Non Transfer	Transfer
TS-4	2587	0
TS-3	3719	0
TS-2	5211	0
TS-1	6775	0
TS	6072	0

5. CONCLUSION

The conclusion that can taken from this research is building data mart give ease to get information about student, can used by Muhammadiyah University of Yogyakarta to fill in borang accreditation and the information can display in tabular.

REFERENCES

- [1] Windarto, "Pemanfaatan Data Warehouse Sebagai Sarana Penunjang Penyusunan Borang Akreditasi Standar 3 Pada Fakultas Teknologi Informasi Universitas Budi Luhur," *TELEMATIKA MKOM*, vol. 3, no. 2, 2011.
- [2] M. Febriady and B. A. Tama, "Rancang Bangun Data Warehouse Untuk Menunjang Evaluasi Akademik di Fakultas," in *KNTIA*, Palembang, 2011.
- [3] A. Amborowati, "Perancangan dan Pembuatan Data Warehouse pada Perpustakaan STMIK AMIKOM Yogyakarta," *Seminar Nasional Aplikasi Sains dan Teknologi*, 2008.
- [4] W. H. Inmon, *Building the Data Warehouse Fourth Edition*, John Wiley & sons, 2005.
- [5] J. A. Hoffer, M. B. Prescott and F. R. McFaden, *Modern Database Management Eighth Edition*, Pearson Education Inc, 2007.
- [6] T. M. Connolly and Carolyn E. Begg, *Database System A Practical Approach to Design, Implementation, and Management fourth edition*, United States of America: Addison Wesley, 2005.
- [7] V. Rainardi, *Building a Data Warehouse: With Example in SQL Server*, 2008.