

# Penerapan *Slowly Changing Dimensions* untuk Mendukung Pembentukan Dimensi Dinamis pada *Data Warehouse*

(Studi Kasus: Dinas Kependudukan dan Pencatatan Sipil Kabupaten XYZ)

Gadis Pujiningtyas Rahayu  
Jurusan Teknik Informatika  
Universitas Sanata Dharma  
Yogyakarta, Indonesia  
gadis.anis@yahoo.com

Ridowati Gunawan  
Jurusan Teknik Informatika  
Universitas Sanata Dharma  
Yogyakarta, Indonesia  
rido@usd.ac.id

**Abstrak**—Dalam pengelolaan data, utamanya bagi Pemerintah Daerah yang memiliki data penduduk dengan *volume* besar dan tersebar ke dalam *database* terpisah, dewasa ini telah ada teknologi data *warehouse*. Data penduduk yang tersimpan dapat mengalami perubahan, hal ini merefleksikan aspek dinamis dari data *warehouse* sehingga memerlukan *updating* beberapa dimensi. Tabel dimensi memiliki peran yang sangat penting dalam setiap *Data Warehouse*. Peran utama tabel dimensi adalah untuk mendukung catatan tabel fakta dengan deskripsi dan informasi lain tentang entitas yang terlibat pada catatan ini. Untuk mendukung pembentukan dimensi yang dinamis dalam sebuah data *warehouse* diimplementasikan sebuah kriteria dimensional yaitu *Slowly Changing Dimensions* (SCD) sehingga setiap perubahan data dapat terpelihara dengan baik.

**Kata kunci**—*Slowly Changing Dimensions* (SCD); dimensi dinamis; data *warehouse*

## I. PENDAHULUAN

### A. Latar Belakang

Informasi dan data yang dikelola atau dimiliki oleh Pemerintahan Daerah kini telah berubah menjadi aset berharga untuk menentukan keputusan dalam perencanaan pembangunan dan pengembangan daerah guna pengambilan kebijakan Pemerintah dalam rangka menyejahterakan penduduk. Selain itu untuk memutuskan kebijakan yang cepat dan tepat, data dan informasi tersebut perlu dipelihara untuk kemudahan akses jika sewaktu-waktu dibutuhkan misalnya untuk keperluan analisis pertumbuhan penduduk, pendukung keputusan kebijakan Pemerintah dan lain sebagainya. Kemudahan akses data-data operasional yang bersifat *historical* dapat dikembangkan menjadi informasi guna kebutuhan perencanaan atau kebutuhan strategis ke depan dengan tujuan memperoleh informasi yang relevan bagi kebutuhan Pemerintah yang dipakai untuk pengambilan keputusan.

Untuk mendukung pengelolaan data yang baik, khususnya bagi Pemerintah Daerah yang memiliki data penduduk dengan

*volume* besar dan tersebar ke dalam *database* terpisah, dewasa ini telah ada teknologi *Data Warehouse* yang dapat menggabungkan data dari berbagai sumber data operasional dan sinkronisasi datanya dapat dilakukan secara periodik maupun *real time*, disesuaikan dengan kebutuhan yang ada. “A *Data Warehouse is a Subject Oriented, Integrated, Non-Volatile, and Time-variant collection of data in support of management’s decisions*” [2]. Penerapan *Data Warehouse* yang baik dan sesuai dengan kebutuhan secara otomatis memudahkan pengelolaan data dan proses pengambilan informasi dari sumber data.

Admin dari masing-masing Kecamatan mengentrikan data penduduk yang disimpan kedalam *database* terpusat di Kabupaten dan dapat diambil dengan mudah rangkuman datanya ketika digudangkan dalam suatu basisdata besar (*Data Warehouse*) terlebih dahulu. Data yang telah disimpan dapat mengalami perubahan, hal ini menunjukkan aspek dinamis dari *Data Warehouse* sehingga memerlukan *updating* beberapa dimensi. Perubahan-perubahan dalam catatan dimensi dapat menyebabkan situasi tidak teratur jika tidak diperlakukan dengan baik. Untuk mendukung pembentukan dimensi yang dinamis dalam sebuah *Data Warehouse* diimplementasikan sebuah kriteria *Dimensional* yaitu *Slowly Changing Dimensions* (SCD) sehingga data histori dapat tersimpan dan diakses sewaktu-waktu.

Hasil dari sebuah *Data Warehouse* dapat dimanfaatkan untuk kepentingan OLAP (*Online Analytical Processing*) yang memungkinkan dilakukan *query* dengan cepat dan menghasilkan informasi secara multidimensi.

Kebutuhan akan *Data Warehouse* dan juga penyajian informasi dengan multidimensi sangat diperlukan oleh Pemerintah Daerah agar informasi dapat diperoleh dengan cepat dan pengambilan keputusan dapat segera dilakukan.

### B. Rumusan Masalah

Bagaimana membangun sistem *database* OLAP (*Online Analytical Processing*) dengan dengan mengimplementasikan SCD *type 2* yang mendukung pembentukan dimensi dinamis

pada *Data Warehouse* untuk membantu DISDUKCAPIL (Dinas Kependudukan dan Pencatatan Sipil) Kabupaten XYZ dalam mengelola data penduduk.

C. Tujuan

Menetapkan SCD *type 2* dalam pembentukan dimensi dinamis pada *Data Warehouse* untuk membangun sistem *database OLAP (Online Analytical Processing)*.

II. TINJAUAN PUSTAKA

A. Data Warehouse

Menurut W.H. Inmon dalam [1], *Data Warehouse* didefinisikan sebagai sekumpulan data yang terintegrasi, basis data berorientasi subyek yang di desain untuk mendukung fungsi sistem pengambilan keputusan, dimana setiap unit dari data adalah *non-volatile* dan relevan untuk waktu tertentu.

B. Slowly Changing Dimensions (SCD)

Dapat dipastikan bahwa pada suatu saat sumber data pembentuk dimensi akan berubah. Perubahan tersebut dinamakan *Slowly Changing Dimensions (SCD)* atau dimensi yang berubah secara perlahan. Ada 3 macam tipe respon / penanganan SCD yaitu Tipe 1, 2, dan 3. Bagian Berikut menerangkan apa perbedaan dari tiap tipe SCD [2] :

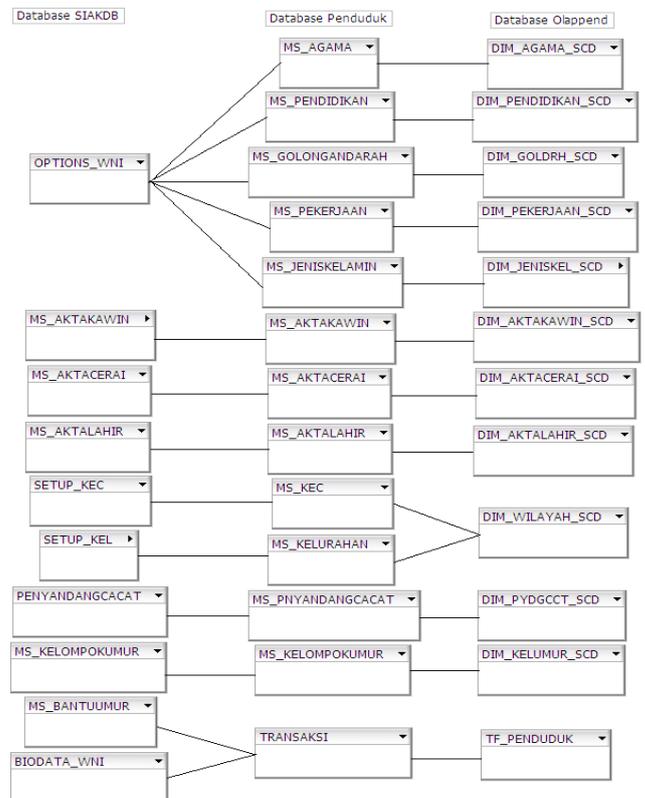
- a. *Type 1 SCD*, menggunakan pendekatan menimpa data yang berubah (*overwrite*) [3]. Ini dilakukan jika memang tidak ada kepentingan menyimpan data historis atau pergerakan isi data dari dimensi tersebut.
- b. *Type 2 SCD*, menyimpan semua historis data dengan *Surrogate Key* berbeda (*partitioning history*). Dapat menambahkan masa berlaku dari dimensi bersangkutan. Ini dilakukan jika memang pergerakan data historis tetap berpengaruh kepada analisis data seperti perpindahan salesman dari satu regional ke regional lain.
- c. *Type 3 SCD*, menyimpan satu historis data pada kolom lain (*alternate realities*). Sama seperti *Type 2* hanya saja lebih terbatas kepada jumlah perubahan data (*update*) yang diijinkan.

III. METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan untuk perancangan *data warehouse* adalah *Nine-Step Methodology* menurut Kimball dalam buku Connolly dan Begg [1] dengan langkah sebagai berikut:

- a. Memilih Proses.  
Proses yang dipilih adalah pembuatan laporan.
- b. Memilih Sumber.  
Setelah menentukan proses pengolahan data yang dibutuhkan dalam *data warehouse*, maka ditentukan *grain* yang menjelaskan tiap fakta.
- c. Identifikasi dan Konfirmasi Dimensi.  
Menentukan dimensi apa saja yang diinginkan.
- d. Pemilihan Fakta.  
Fakta yang akan diambil adalah fakta tentang kependudukan yang secara fisik disimpan dalam tabel

- e. Penyimpanan Kalkulasi Awal dalam Tabel Fakta.  
Untuk langkah ini tidak terjadi kalkulasi awal. Karena yang diukur hanyalah jumlah penduduk saja.
- f. Melengkapi tabel dimensi.  
Pada tahap ini dilakukan pemeriksaan ulang pada tabel dimensi dan menambahkan deskripsi teks terhadap dimensi, serta menentukan hirarki atribut dimensi untuk mempermudah proses analisis. Proses pemetaan yang dilakukan yaitu dari *database SIAKDB* ke *database penduduk* dan *database olappend*. Gambar 1 merupakan alur proses pembentukan tabel dan proses transformasi data.



Gambar 1. Alur proses pembentukan tabel dan proses transformasi data

Langkah-langkah proses integrasi data dilakukan dengan:

1. Pengintegrasian data transaksional dan data master dari *database* sumber (SIAKDB) ke *database* target (penduduk).
  2. Pemecahan tabel master *Data Warehouse* menjadi tabel dimensi dan tabel fakta.
  3. Pengintegrasian data transaksional dan data master untuk membentuk tabel dimensi dan tabel fakta ke *database OLAP*.
- g. Menentukan durasi *database*.  
Pada langkah ini menentukan lamanya sumber data yang akan digunakan pada *data warehouse*. Untuk *data*

warehouse penduduk ini digunakan sumber data dari database OLTP yang diambil dari tiga tahun yang lalu.

h. Melacak SCD

Untuk memantau perubahan yang terjadi dalam dimensi, digunakan tipe *Slowly Changing Dimensions* (SCD) yang dikemukakan oleh Kimball [2]. Hal ini diterapkan pada semua tabel dimensi menggunakan SCD tipe 2

i. Menentukan prioritas query dan jenis query.

Pada tahap ini dilakukan pertimbangan perancangan fisik, seperti keberadaan dari *summary* (ringkasan) dan *aggregate* (penjumlahan). Pada bagian ini, rancangan desain fisik yang disarankan hanya urutan fisik dari tabel fakta, dimana urutan data pada sebuah tabel fakta telah terurut berdasarkan waktu transaksi masing-masing. Untuk pendesainan *indexing*, setiap *primary key* pada tiap dimensi menggunakan *index*.

IV. IMPLEMENTASI DAN ANALISIS HASIL

A. Implementasi Hasil

1) Implementasi Integrasi Data

Setelah melewati tahap perancangan struktur *database Data Warehouse*, pembangunan *Data Warehouse* selanjutnya proses implementasi integrasi data. Integrasi data transaksional dan *data master* dari database SIAKDB ke database penduduk terdiri dari beberapa langkah. Setiap langkah memiliki aktivitas dan tujuan tersendiri. Aktivitas dan tujuan dari proses integrasi data dapat dilihat pada Tabel I.

TABEL I. AKTIVITAS DAN TUJUAN DARI INTEGRASI DATA TRANSAKSIONAL DAN DATA MASTER DARI DATABASE SIAKDB KE DATABASE PENDUDUK

No	Langkah	Aktivitas	Tujuan
1	Integrasi Data transaksi dari biodata_wni dan ms_bantu umur dari biodata_wni dan ms_bantu umur	Transformasi sumber data dari tabel biodata_wni dan ms_bantuumur di database siakdb ke tabel transaksi di database penduduk.	pembentukan tabel transaksi
2.	Integrasi Data ms_agama dari options_wni	Transformasi sumber data dari tabel options_wni di database siakdb ke tabel ms_agama di database penduduk	pembentukan tabel ms_agama
3.	Integrasi Data ms_pekerjaan dari options_wni	Transformasi sumber data dari tabel options_wni di database siakdb ke tabel ms_pekerjaan di database penduduk.	pembentukan tabel ms_pekerjaan
4.	Integrasi Data ms_pendidikan dari	Transformasi sumber data dari tabel options_wni di database siakdb ke tabel ms_pendidikan di database penduduk.	pembentukan tabel ms_pendidikan

	options_wni		
5.	Integrasi Data ms_golongan darah dari Options_wni	Transformasi sumber data dari tabel options_wni di database siakdb ke tabel ms_golongandarah di database penduduk.	pembentukan tabel ms_golongandarah
6.	Integrasi Data ms_jenis kelamin dari options_wni	Transformasi sumber data dari tabel options_wni di database siakdb ke tabel ms_jeniskelamin di database penduduk.	pembentukan tabel ms_jeniskelamin
7.	Integrasi Data ms_aktakawin dari ms_aktakawin	Transformasi sumber data dari tabel ms_aktakawin di database siakdb ke tabel ms_aktakawin di database penduduk.	pembentukan tabel ms_aktakawin
8.	Integrasi Data Ms_aktaceraai dari Ms_aktaceraai	Transformasi sumber data dari tabel ms_aktaceraai di database siakdb ke tabel ms_aktaceraai di database penduduk.	pembentukan tabel ms_aktaceraai
9.	Integrasi Data Ms_aktalahir dari Ms_aktalahir	Transformasi sumber data dari tabel ms_aktalahir di database siakdb ke tabel ms_aktalahir di database penduduk.	pembentukan tabel ms_aktalahir
10.	Integrasi Data Ms_kec dari Ms_Setup_kec	Transformasi sumber data dari tabel ms_kec di database siakdb ke tabel ms_kec di database penduduk.	pembentukan tabel ms_kec
11.	Integrasi Data Ms_kel dari Ms_Setup_kel	Transformasi sumber data dari tabel ms_kel di database siakdb ke tabel ms_kelurahan di database penduduk.	pembentukan tabel ms_kelurahan
12.	Integrasi Data ms_pnyandangcacat dari pnyandangcacat	Transformasi sumber data dari tabel ms_pnyandangcacat di database siakdb ke tabel ms_pnyandangcacat di database penduduk.	pembentukan tabel ms_pnyandangcacat
13.	Integrasi Data Ms_kelompokum dari Ms_kelompokum	Transformasi sumber data dari tabel ms_kelompokum di database siakdb ke tabel ms_kelompokum di database penduduk.	pembentukan tabel ms_kelompokum
14.	Integrasi Data ms_bantu umur dari ms_bantu umur	Transformasi sumber data dari tabel ms_bantu umur di database siakdb ke tabel ms_bantu umur di database penduduk.	pembentukan tabel ms_bantu umur

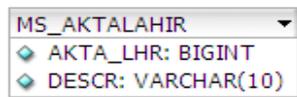
2) Implementasi Memecah *Data Warehouse* Membentuk Tabel Fakta dan Tabel Dimensi.

Dimensi yang akan dibentuk adalah dimensi jenis kelamin, dimensi agama, dimensi pekerjaan, dimensi penyandang cacat, dimensi akta kawin, dimensi akta lahir, dimensi akta cerai, dimensi pendidikan, dimensi kecamatan, dimensi usia, dimensi wilayah, dan dimensi golongan darah.

3) Pembentukan Dimensi Dinamis

Cara yang ditempuh untuk menguji SCD *type 2* agar pergerakan data dimensi terlihat yaitu dengan menambah atau *update* data. Data yang tersimpan dapat dilihat dari berbagai dimensi, yaitu dimensi jenis kelamin, dimensi agama, dimensi pekerjaan, dimensi penyandang cacat, dimensi akta kawin, dimensi akta lahir, dimensi akta cerai, dimensi pendidikan, dimensi kecamatan, dimensi usia, dimensi wilayah dan dimensi golongan darah. Dalam pengujian ini akan dibuat lebih fokus yaitu pergerakan data yang dilihat dari sudut pandang akta lahir. Berikut pergerakan data yang dapat dilihat dari beberapa tabel yaitu *dim\_aktalahir\_scd* dan transaksi.

Gambar 2 menunjukkan struktur tabel yang dimiliki oleh tabel *ms\_aktalahir*. Sedangkan Gambar 3 adalah semua data yang ada pada *ms\_aktalahir*. Pada tabel *ms\_aktalahir* terdapat 2 rows yang akan ditransformasi.

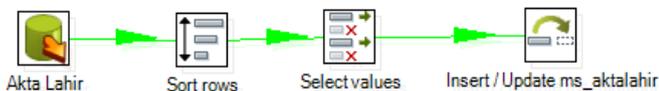


Gambar 2. Struktur Tabel Ms\_aktalahir

AKTA_LHR	DESCR
1	Tidak Ada
2	Ada

Gambar 3. Data ms\_aktalahir

Tabel master akta lahir ini yang akan ditransformasikan membentuk dimensi SCD *type 2* untuk mendukung dimensi dinamis. Adapun proses transformasi yang dilakukan dapat dilihat pada Gambar 4.

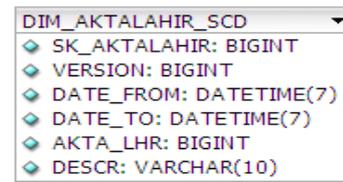


Gambar 4. ms\_aktalahir.ktr

*Step insert/update* tabel *ms\_aktalahir* adalah step yang akan menambah/memperbaharui data dari proses transformasi ke tabel *ms\_aktalahir* di *database* penduduk berdasarkan nilai data kode *akta\_lhr* pada tabel *ms\_aktalahir*.

Ketentuan proses *Insert/Update*: jika *akta\_lhr* tidak sama dengan *akta\_lhr* pada tabel *ms\_aktalahir* maka dilakukan proses menambahkan/*insert* data baru. Jika *akta\_lhr* sama dengan *akta\_lhr* pada tabel *ms\_aktalahir* maka dilakukan proses mengubah/*update* data yang lama dengan data yang baru.

Hasil transformasi dari tabel *ms\_aktalahir* adalah *dim\_aktalahir\_scd*. Struktur *dim\_aktalahir\_scd* dapat dilihat pada Gambar 5.



Gambar 5. Struktur Tabel Ms\_aktalahir\_SCD

Sedangkan Gambar 6 adalah data dari *dim\_aktalahir\_scd* hasil transformasi.

SK_AKTALAHIR	VERSION	DATE_FROM	DATE_TO	AKTA_LHR	DESCR
0	1				
1	1	01/01/1900	31/12/2199 23:59:59		1 Tidak Ada
2	1	01/01/1900	31/12/2199 23:59:59		2 Ada

Gambar 6. Data dim\_aktalahir\_scd dengan menerapkan SCD 2.

*Query* untuk *insert* dan *update* *ms\_aktalahir* yang telah disiapkan untuk menguji SCD *type 2* agar pergerakan data dimensi akta lahir terlihat adalah :

```
UPDATE MS_AKTALAHIR SET DESCR='Tidak' WHERE AKTA_LHR=1;

INSERT INTO MS_AKTALAHIR ( AKTA_LHR, DESCR ) VALUES (3, 'Proses');
```

Setelah dilakukan *update* data pada *ms\_aktalahir* data pada tabel *ms\_aktalahir* terdapat 3 record seperti yang terlihat pada gambar 7.

AKTA_L...	DESCR
1	Tidak
2	Ada
3	Proses

Gambar 7. Data setelah ms\_aktalahir diinsert dan di-update

Proses pembentukan dimensi dinamis telah dilakukan, langkah berikutnya adalah kembali melakukan proses transformasi data dengan menggunakan dimensi dinamis. Maka data akan ter-*update* secara otomatis. Gambar 8 berisi data *dim\_aktalahir\_scd* hasil dari transformasi *ms\_aktalahir*.

SK_AKTALAHIR	VERSION	DATE_FROM	DATE_TO	AKTA_LHR	DESCR
0	1				
1	1	01/01/1900	13/02/2013 8:24:43		1 Tidak Ada
2	1	01/01/1900	31/12/2199 23:59:59		2 Ada
3	2	13/02/2013 8:24:43	31/12/2199 23:59:59		1 Tidak
4	1	01/01/1900	31/12/2199 23:59:59		3 Proses

Gambar 8. Data hasil pergerakan dimensi dinamis

4) Implementasi *Star Schema* pada *Database OLAP*

*Star schema* merupakan desain *database* yang akan digunakan untuk kepentingan analisis OLAP. Kubus yang akan dibuat adalah kubus Pendudukan yang akan membaca tabel *tf\_penduduk* yang ada dalam *database* *olappend*. Gambar 9 merupakan *star schema* kubus penduduk.



Gambar 9. Star Schema dari Kubus Penduduk

B. Analisis Hasil

1) Data Dimensi

Seluruh dimensi yang ada dibuat model pergerakan dimensinya. Gambar 6 merupakan contoh dimensi akta lahir yang belum mengalami pergerakan. Sedangkan gambar 8 adalah dimensi akta lahir yang mengalami perubahan pada data masternya. Proses transformasi yang dilakukan terhadap perubahan master data menggunakan teknik *SCD type 2* yang menyimpan seluruh histori data. Ketika data *ms\_aktalahir* mengalami perubahan dilakukan transformasi data ke *dim\_aktalahir\_scd* (dimensi hasil perubahan master akta lahir) maka *value* dari *sk\_aktalahir* bertambah otomatis.

Gambar 8 merupakan hasil transformasi *dim\_aktalahir\_scd* yang telah mengalami pergerakan dimensi. Untuk akta\_lhr=1 telah *di-update* *descr* menjadi 'Tidak' sehingga data baru yang dipunyai ini akan disimpan dengan *version 2*. Pada data yang lama dengan *descr* 'Tidak Ada' untuk masa *validnya* berubah pada saat dilakukan transformasi yaitu pada kolom *date\_to*.

Pergerakan ataupun perubahan data pada master data akan selalu dipertahankan dengan menerapkan *SCD type 2* ini. Hal ini akan membawa dampak pada data faktanya, tidak akan ada data yang hilang historinya. Tercatat perubahan yang terjadi yang dilihat dari waktu perubahan dan kapan data tersebut valid.

2) Hasil Pengujian *Historical Data*

Data yang akan dipilih untuk *di-update* adalah sebanyak 3 NIK (Karena alasan tertentu NIK diblok merah). Gambar 10 menunjukkan data yang akan *di-update*.

3510100000000002	Semua Akta Lahir	1
3510100000000002	Semua Akta Lahir	1
3510100000000002	Semua Akta Lahir	1
3510100000000002	Semua Akta Lahir	1
	Tidak Ada	1
3510100000000002	Semua Akta Lahir	1
3510100000000002	Semua Akta Lahir	1
3510100000000002	Semua Akta Lahir	1

Gambar 10. Data NIK yang akan diupdate

Data tersebut akan dilakukan proses *update*. Kemudian akan dilakukan tranformasi dengan teknik *SCD type 2*. Ketika data sudah *ter-update* maka *view* data yang terbaru akan terlihat seperti Gambar 11.

3510047112470012	Semua Akta Lahir	1
3510000000000001	Semua Akta Lahir	1
3510000000000001	Semua Akta Lahir	1
3510000000000001	Semua Akta Lahir	1
3510000000000001	Semua Akta Lahir	1
3510000000000001	Semua Akta Lahir	1
3510000000000001	Semua Akta Lahir	1
	Proses	1

Gambar 11. Data NIK yang telah diupdate

*View* data yang ditampilkan adalah data yang terbaru, meskipun demikian data lama tidak akan hilang karena data tersebut tersimpan pada masing-masing dimensi. Data dimensi yang sudah *ter-update* dapat diakses dengan melihat histori penduduk. Gambar 12 adalah *form* untuk melihat histori data

Data History Penduduk

Masukkan NIK penduduk:

Kategori Dimensi:

- Dimensi-
- Jenis Kelamin
- Akta Lahir
- Akta Kawin
- Akta Cerai
- Penyandang Cacat
- Agama
- Pendidikan
- Golongan Darah
- Pekerjaan
- Wilayah

Gambar 12. Form untuk melihat histori data

NIK penduduk yang dimasukkan akan dibaca dan dilakukan proses pembacaan data dari dimensi sesuai dengan kategori dimensi yang dipilih. Hasil pembacaan histori data penduduk dilihat dari dimensi akta lahir dapat dilihat pada Gambar 13. Pada Gambar 13 terlihat untuk setiap NIK pergerakan dan perubahan data tetap dipertahankan dengan diterapkannya *SCD type 2*.

## Data History Penduduk

Masukkan NIK penduduk:

Kategori Dimensi:

**NIK** ████████████████████  
**Nama** ████████████████████  
**Data History**

Akta Lahir	Versi	Tgl Berlaku	Tgl Kadaluarsa
Proses	2	1900-01-01 00:00:00.0	2199-12-31 23:59:59.0
Tidak Ada	1	1900-01-01 00:00:00.0	2013-02-13 08:24:43.0
Tidak	1	2013-02-13 08:24:43.0	2199-12-31 23:59:59.0

Gambar 13. Setelah dimensi mengalami pergerakan.

## V. SIMPULAN DAN SARAN

### A. Simpulan

Simpulan yang dapat diberikan dari penelitian ini adalah:

- Proses Pembangunan sistem *database* OLAP (*Online Analytical Processing*) dengan Mondrian sebagai OLAP server dengan mengimplementasikan SCD *type* 2 yang mendukung pembentukan dimensi dinamis pada *Data Warehouse* telah berhasil dilakukan.
- Proses transformasi dilakukan secara manual dengan menjalankan *file* otomatisasi.bat

- Perhitungan OLAP akan bernilai *valid* jika data sumber memiliki kelengkapan informasi yang dibutuhkan terutama data yang akan ditransformasi dengan data dimensi.
- Histori penduduk dapat terekap dengan baik.
- Pemecahan masalah yang dialami DISDUKCAPIL Kabupaten XYZ dalam pembuatan laporan dengan menerapkan teknik *Data Warehouse* untuk mengelola data Penduduk adalah tepat.

### B. Saran

Dalam penelitian ini proses *Load* data yang besar untuk transformasi membutuhkan waktu yang agak lama. Untuk itu diperlukan teknik tranformasi yang lebih efisien. Atau dapat mencoba menggunakan *co>operating system* [4] sebuah *platform* untuk komputasi *enterprise* secara paralel dan terdistribusi berbasis *dataflow*.

## DAFTAR PUSTAKA

- T. Connolly, C.Begg, "Database Systems: A Practical Approach to Design, Implementation, and Management", 5<sup>th</sup> ed., Addison-Wesley, 2010.
- R. Kimball, J. Caserta., "The data warehouse ETL toolkit : practical techniques for extracting, cleaning, conforming, and delivering data", Wiley, 2004.
- K. Srikanth, M.E.V.S. Murthy, and J. Anitha, "Data Warehouse Concept Using ETL Process For SCD-Type 1," The International Journal of Computer Science & Applications (TIJCSA), volume 1, no. 10, December 2012.
- C.W. Stanfill, "Type 2 slowly changing dimensions: a case study using the *co>operating system*," DOLAP'12: Proceedings of the fifteenth international workshop on Data warehousing and OLAP, pp. 81-88