

**ANALISIS DAN PERANCANGAN SISTEM INFORMASI EKSEKUTIF DAN DATA WAREHOUSE  
MONITORING PERMOHONAN DAN PERSURATAN MEREK PADA DIREKTORAT MEREK  
DAN INDIKASI GEOGRAFIS DIREKTORAT JENDERAL KEKAYAAN**

**Urim Carry Wilson Sitio<sup>1</sup>, Yudi Harianto<sup>2</sup>**

Sekolah Tinggi Manajemen Informatika dan Komputer Jayakarta  
urimo.sitio@gmail.com<sup>1</sup>, yudi.harianto@gmail.com<sup>2</sup>

**Abstrak**

Direktorat Jenderal Kekayaan Intelektual (DJKI) adalah sebuah unit eselon 1 pada Kementerian Hukum dan Hak Asasi Manusia, kesulitan dalam merekap data dalam jumlah besar dan ditampilkan dalam bentuk yang mudah dimengerti sehingga DJKI membutuhkan sebuah Sistem Informasi Eksekutif atau *Executive Information System* (EIS) dan *data warehouse* untuk menunjang pengambilan keputusan terkait permohonan merek dan persuratan, dengan dibuatnya EIS yang berlingkup pada Direktorat Merek dan Indikasi Geografis dan digunakan para Eksekutif Direktorat Merek dan Indikasi Geografis selaku pengambil keputusan Direktorat Merek dan Indikasi Geografis, agar dapat mempercepat pembuatan keputusan yang terbaik untuk Direktorat Merek dan Indikasi Geografis dan publik. Perancangan EIS dan *data warehouse* dilakukan dengan *Software Life Development Cycle* (SDLC), dimana menggunakan metode *waterfall* dalam perancangan EIS tersebut. Untuk *database* akan digunakan Oracle 11g dimana sebagian besar aplikasi pada DJKI menggunakan oracle 11g dalam menyimpan data, selain itu untuk bahasa pemrograman akan menggunakan VB.NET dan template *DevExpress*. Setelah melalui pengujian *blackbox testing* EIS dapat digunakan untuk membantu para eksekutif dalam pengambilan keputusan.

**Kata Kunci:** *Data warehouse*, Sistem Informasi Eksekutif, Direktorat Jenderal Kekayaan Intelektual, Direktorat Merek dan Indikasi Geografis

**I. PENDAHULUAN**

Direktorat Jenderal Kekayaan Intelektual (DJKI) adalah sebuah Unit Eselon 1 pada Kementerian Hukum dan Hak Asasi Manusia, bergerak pada bidang pencatatan dan pendaftaran Kekayaan Intelektual di Indonesia, baik berupa Merek, Paten, Desain Industri, Indikasi Geografis, Hak Cipta dan lain-lain. Pendaftaran Kekayaan Intelektual yang terus meningkat setiap tahun tentu berpengaruh pada pertambahan data pada basis data Aplikasi yang digunakan DJKI yang bernama *Intellectual Property Automation System* (IPAS) yang melakukan pencatatan pada Merek, Desain Industri, Paten, selain itu DJKI mempunyai beberapa basis data dari aplikasi lain yang digunakan, seperti e-Hak-Cipta untuk pendaftaran Hak Cipta secara *online* dan IPAS Hak Cipta yang penggunaannya berbeda dengan IPAS yang digunakan untuk Merek, Desain Industri dan Paten.

Pada Direktorat Merek permohonan yang didapatkan pada tahun 2015 sebanyak 15.763 untuk merek Jasa, 24.328 untuk merek dagang dan sekitar 25.000 dokumen yang diajukan baik untuk perubahan nama pemilik, pengalihan hak, perpanjangan merek dan dokumen-dokumen lainnya.

Direktorat Merek dibawah oleh 1 orang Direktur dan juga 5 orang Sub Direktorat yang dipimpin oleh para Kasubdit yang membawahi 10 seksi yang dipimpin oleh para Kepala Seksi dan terdapat pula para pemeriksa merek yang bertugas untuk memeriksa permohonan merek dan dokumen-dokumen yang telah diajukan oleh para Pemohon.

Dikarenakan banyaknya permohonan merek dan dokumen baik untuk pengalihan hak, perpanjangan merek, perubahan nama dan dokumen-dokumen lainnya, Direktorat Merek dan Indikasi Geografis dituntut untuk memproses seluruh permohonan sesuai dengan undang-undang Merek yang berlaku di Indonesia yaitu Undang-Undang nomor 15 tahun 2001, juga untuk memantau kinerja dari Direktorat Merek dan Indikasi Geografis, sehingga dibutuhkan suatu alat untuk memantau pengerjaan permohonan dan persuratan dari Direktorat Merek dengan para pemohon, juga suatu alat untuk dapat melihat terjadinya penumpukan pekerjaan, sehingga dapat dicari solusi untuk memecahkan penumpukan tersebut, bila alat untuk memantau tersebut dihubungkan langsung pada *database* aplikasi IPAS, tentu akan memberatkan kinerja *database* tersebut, dikarenakan *database* aplikasi IPAS digunakan oleh banyak pengguna dan berbagai aplikasi lainnya.

Untuk mengatasi masalah tersebut, maka dibutuhkan sebuah rancangan berupa rancangan *data warehouse* untuk Direktorat Merek, yang kelanjutannya *data warehouse* tersebut dapat diolah menjadi laporan baik berbentuk tabel atau grafik, yang dapat dilihat dan mudah dipahami oleh para eksekutif untuk menunjang keputusan yang akan dibuat. *Data warehouse* tersebut terus terbaharui 1 (satu) kali sehari tanpa mengganggu kinerja dari *database* aplikasi IPAS, dengan jadwal setiap malam hari agar tidak terlalu memberatkan *database* aplikasi IPAS. Sehingga informasi yang ditampilkan selalu terbaru.

### 1.1 Definisi Masalah

Berdasarkan hasil pengamatan dan latar belakang masalah pada Direktorat Jenderal Kekayaan Intelektual, diperoleh beberapa masalah, seperti yang diuraikan dibawah ini.

Masih belum adanya rancangan *data warehouse* yang berupa ringkasan dari *database* aplikasi IPAS tentang monitoring permohonan Merek dan persuratan Direktorat Merek dan Indikasi Geografis.

Masih belum adanya rancangan sebuah Sistem Informasi Eksekutif yang terhubung dengan sebuah *data warehouse* untuk menampilkan data tentang monitoring permohonan dan persuratan yang mudah dipahami dan dianalisa, oleh para pengambil keputusan Direktorat Merek dan Indikasi Geografis.

## II. KAJIAN TEORI

### 2.1 Data warehouse

*Data warehouse* adalah koleksi data yang mempunyai sifat *subject-oriented*, *integrated*, *nonvolatile*, dan *time-variant* untuk mendukung proses pengambilan keputusan dalam manajemen.

### 2.2 Arsitektur data warehouse

Arsitektur *data warehouse* terbagi dalam beberapa komponen.

### 2.3 Back End Tier

*Back End Tier* tersusun dari beberapa proses yaitu:

1. *Extraction*, mengumpulkan data dari beberapa sumber, sumber bisa dari *database*, file dan dalam format-format lainnya.
2. *Transform*, mengubah data yang berasal dari berbagai sumber menjadi sesuai dengan format data pada *data warehouse*, dan terdiri dari beberapa aspek yaitu:

- *Cleaning*, yaitu menghapus kesalahan dan ketidak konsistenan data dan merubah sesuai dengan format yang standar.
  - *Integration*, menyatukan data dari sumber-sumber data yang berbeda baik pada skema maupun data level.
  - *Aggregation*, menyimpulkan data dari sumber data menurut detail level atau rincian pada *data warehouse*.
3. *Loading*, mengisi *data warehouse* dengan data yang sudah melewati *Transform*, termasuk juga pembaruan *data warehouse* dengan spesifikasi waktu untuk menjaga agar data pada *data warehouse* merupakan data terbaru. Pembaruan bisa dalam hitungan bulan, hari bahkan jam.

### 2.4 Skema Dalam Data Warehouse

#### Skema Bintang

Skema bintang adalah suatu jenis spesifik dari perancangan basis data yang digunakan untuk mendukung proses analitis serta memiliki secara spesifik satuan tabel normalisasi. Skema bintang memiliki dua macam tabel yaitu:

1. Tabel fakta, disebut juga tabel utama (*major table*), merupakan inti skema bintang dan berisi data aktual yang akan dianalisis (data kuantitatif dan transaksi). *Field-field* table fakta sering disebut dengan *measure* dan biasanya dalam bentuk numerik. Selalu berisi *foreign key* dari masing-masing tabel dimensi. Tabel dapat terdiri atas banyak kolom dari ribuan baris data.
2. Tabel dimensi, disebut juga tabel kecil (*minor tabel*), biasanya lebih kecil dan memegang data deskriptif yang mencerminkan dimensi suatu bisnis. Tabel dimensi berisi data yang merupakan deskripsi lebih lanjut dari data yang ada pada tabel fakta.

### 2.5 Skema Snowflake

Skema *snowflake* merupakan skema yang berbeda dengan skema bintang karena tabel dimensi tidak berisi data denormalisasi. *Snowflake* merupakan variasi lain dari skema bintang, di mana tabel dimensi dari skema bintang diorganisasi menjadi suatu hirarki dengan melakukan normalisasi. Prinsip dasar skema ini tidak jauh berbeda dari skema bintang. Penggunaan tabel dimensi sangat menonjol, karena itulah perbedaan mendasar dari skema bintang dan skema *snowflake*.

### 2.6 Skema Fact Constellation

Skema ini lebih kompleks dibanding skema *snowflake* atau skema bintang karena skema ini berisi banyak *fact table*. Skema *fact constellation* memungkinkan suatu *dimension table* berhubungan dengan banyak *fact table*. Skema *fact constellation* sangat fleksibel, namun terkadang menjadi susah dalam pengaturan dan *support*. Kerugian utama skema ini adalah desain lebih rumit karena banyak varian agregasi yang harus dipertimbangkan, selain itu juga untuk menjawab *single query* mungkin butuh *multiple SQL statement*.

### 2.7 Agregasi

Agregasi merupakan proses perhitungan data fakta selama pendefinisian atribut. Definisi agregasi biasanya mengandung makna penghitungan, perumusan informasi yang mendasari hubungan antar data yang terdapat dalam sebuah tabel. Selain itu, agregasi juga dibuat selama proses transformasi dan pemuatan data ke dalam *data warehouse*.

### 2.8 Denormalisasi

Merupakan suatu proses penggabungan tabel yang dilakukan dengan cermat dan hati-hati yang bertujuan untuk meningkatkan kinerja. Sebenarnya ini merupakan proses yang melanggar aturan dari *third normal form*. Denormalisasi merupakan kebalikan dari normalisasi. Denormalisasi berguna untuk meningkatkan kinerja *query* untuk mengabaikan adanya *dependency* yang tidak menimbulkan anomali *storage* yang signifikan.

### 2.9 Sistem Informasi Eksekutif

Menurut Kadir (2013). Sistem Informasi Eksekutif atau *Executive Information System* (EIS) terkadang disebut sebagai sistem pendukung eksekutif atau *Executive Support System* (ESS). Sistem ini merupakan sistem informasi yang menyediakan fasilitas yang fleksibel bagi manajer dan eksekutif dalam mengakses informasi eksternal dan internal yang berguna untuk mengidentifikasi masalah atau mengenali peluang. Pemakai yang awam dengan komputer pun tidak sulit mengoperasikannya karena sistem dilengkapi dengan antarmuka yang sangat memudahkan pemakai untuk menggunakannya (*user friendly*).

Informasi di EIS biasa disajikan dalam suatu bentuk yang dinamakan digital *dashboard* yang mencakup berbagai indikator yang penting untuk diperhatikan oleh eksekutif sehingga dapat bereaksi kalau ada hal-hal yang perlu ditindaklanjuti.

EIS secara pokok tidak dirancang untuk menyelesaikan masalah tertentu. EIS dirancang untuk membantu eksekutif mencari informasi yang diperlukan manakala mereka membutuhkannya. Kemampuan *drill-down* yang tersedia pada EIS juga memungkinkan eksekutif dapat melihat rinci suatu informasi.

### 2.10 Oracle

Oracle merupakan program basis data yang memungkinkan perusahaan-perusahaan besar dan pemerintah-pemerintah mengelola, mengakses dan menyimpan sejumlah informasi pada berbagai sistem komputer.

*Database Links* adalah sebuah konsep dari hubungan antar sistem *database*, sebuah *database links* mengkoneksikan antara 2 *database server* yang mengizinkan *database* pengguna *database links* untuk mengakses *database* lain sebagai suatu kesatuan sistem *database*.

*Transparent Network Substrate* (TNS) adalah arsitektur dari jaringan dari Sistem *database* Oracle

*Materialized view* adalah objek *database* yang merupakan hasil *query*, yang meng-copy data kedalam tabel *local database* dan dapat di-update secara berkala. Data dari *materialized view* dapat ditampilkan sebagaimana tabel atau *view* pada *database*. Untuk *data warehouse*, *materialized view* biasa digunakan untuk agregasi dari satu atau banyak tabel.

### 2.11 Visual Basic.Net

*Visual Basic .NET* adalah sebuah alat untuk mengembangkan dan membangun aplikasi yang bergerak di atas sistem *.NET framework*, dengan menggunakan bahasa BASIC. Dengan menggunakan alat ini, programmer dapat membangun aplikasi *Windows Forms*, Aplikasi web berbasis ASP.NET, dan juga aplikasi *command line*. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti Microsoft Visual C++, Visual C#, atau Visual J#), atau juga dapat diperoleh secara terpadu dalam Microsoft Visual Studio .NET. Bahasa Visual Basic .NET sendiri menganut paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari Microsoft Visual Basic versi sebelumnya yang diimplementasikan di atas *.NET framework*. Peluncurannya mengundang kontroversi, mengingat banyak sekali perubahan yang dilakukan oleh Microsoft, dan versi baru ini tidak kompatibel dengan versi terdahulu.

### 2.12 Kekayaan Intelektual

Kekayaan Intelektual didefinisikan sebagai segala bentuk kekayaan materi maupun non materi yang berasal dari hasil kecerdasan dan kepandaian manusia (intelektual), sebagai hasil olah pikir otak manusia (*brainware*) dan proses pembelajaran. Kekayaan intelektual akan menghasilkan produk berupa barang dan jasa yang bermanfaat bagi umat manusia.

### 2.13 Bidang Kekayaan Intelektual

Secara garis besar Kekayaan Intelektual dibagi dalam 2 (dua) bagian, yaitu

1. Hak Cipta (*Copyright*)
2. Hak kekayaan industri (*Industrial Property Rights*), yang mencakup:
  - a. Paten
  - b. Merek
  - c. Indikasi Geografis
  - d. Desain Tata Letak Sirkuit Terpadu
  - e. Rahasia Dagang

### 2.14 Industrial Property Automation System

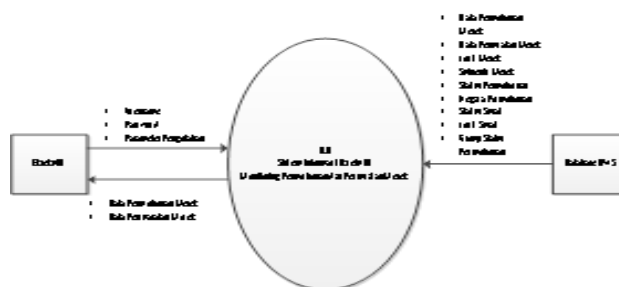
*Industrial Property Automation System* (IPAS) merupakan aplikasi yang dibuat oleh *World Intellectual Property Organization* (WIPO) yang berdiri dibawah naungan PBB, sebagai badan induk dunia dalam hal Kekayaan Intelektual, IPAS yang dikembangkan dengan bahasa pemrograman Java telah dikembangkan hingga versi 3.2 hingga saat ini. Untuk di Indonesia IPAS yang digunakan adalah IPAS 2.7 dengan Oracle sebagai aplikasi *database* yang digunakan

## III. PEMBAHASAN

### 3.1 Desain

#### DFD

Pemodelan EIS dan *data warehouse* dengan menggunakan DFD

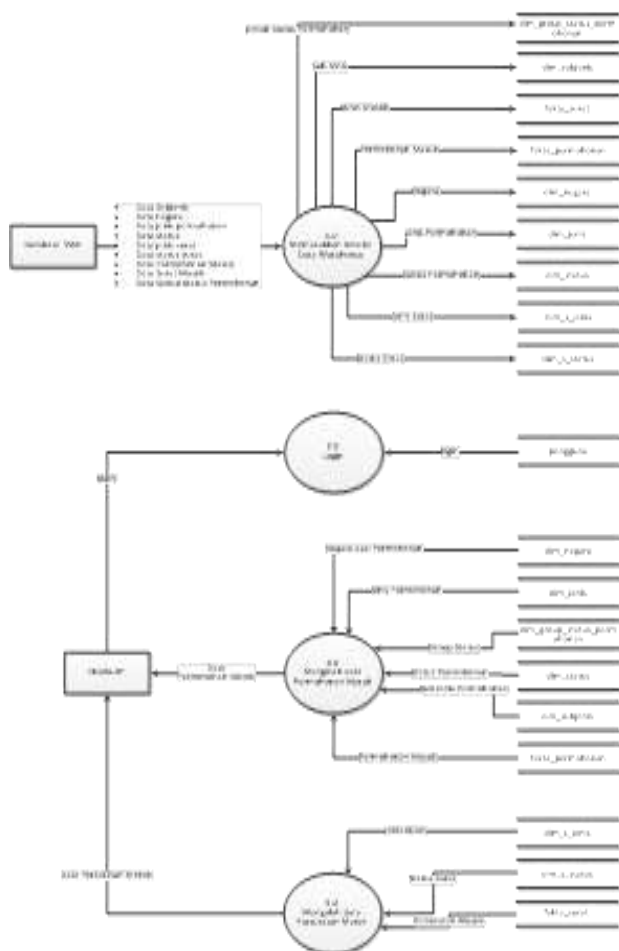


Gambar 1 Data Flow Diagram

Gambar 1 merupakan DFD EIS Monitoring Permohonan dan Persuratan Merek yang terdiri dari 12 input dan 2 Output. Entitas luar yaitu Eksekutif atau pengguna juga *database* IPAS sebagai sumber data.

#### Diagram Level 1

Pemodelan EIS dan *data warehouse* dengan menggunakan Diagram Level 1.



Gambar 2 Diagram Level 1

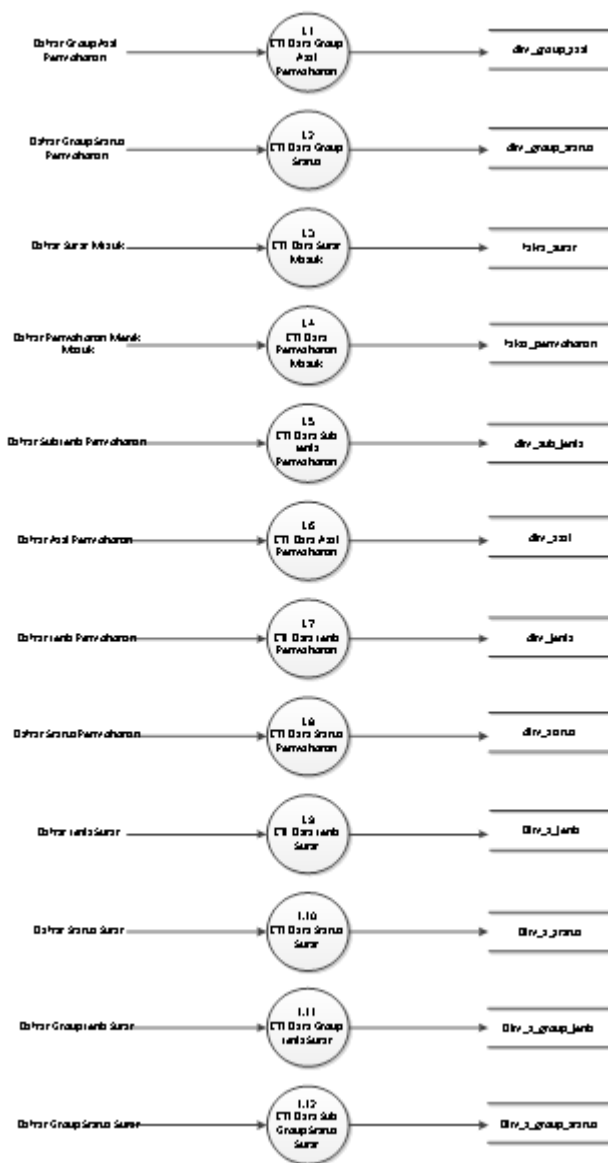
Gambar 2 Merupakan Diagram Level 1 dari EIS *Monitoring*. Permohonan dan Persuratan Merek yang dibagi menjadi proses-proses kecil untuk menjelaskan fungsi-fungsi dan arus data yang mengalir pada Sistem Informasi Eksekutif dan proses ETL berikut adalah proses-prosesnya:

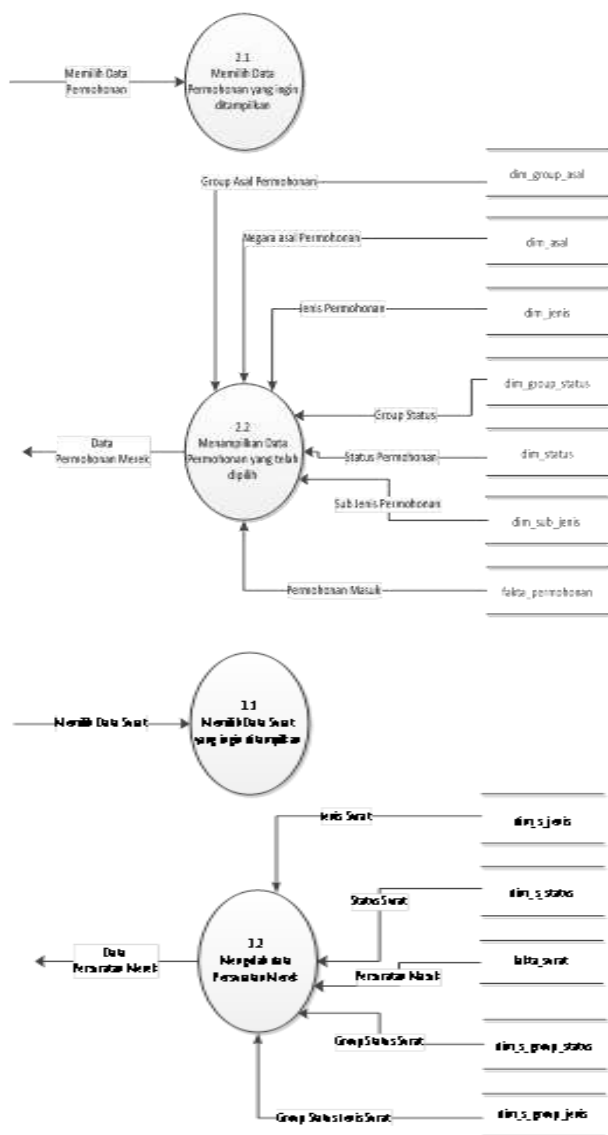
1. Memasukkan data kedalam *data warehouse*, yaitu proses memasukkan data ke dalam *data warehouse* dimana berlangsung proses ETL pada proses ini.
2. Login, yaitu proses untuk masuk ke dalam EIS.
3. Mengolah data Permohonan Merek, yaitu proses untuk mengolah data Permohonan Merek berupa tabel dan grafik dimana pengguna dapat melakukan fungsi *drag drop* atau *drill down* pada dimensi permohonan merek.

4. Mengolah data Persuratan Merek, yaitu proses untuk mengolah data Persuratan Merek berupa tabel dan grafik dimana pengguna dapat melakukan fungsi *drag drop* atau *drill down* pada dimensi persuratan merek.

**Diagram Level 2**

Pemodelan EIS dan *data warehouse* dengan menggunakan Diagram Level 2.





Gambar 3 Diagram Level 2

Pada Gambar 3 Memodelkan EIS dan *data warehouse* untuk monitoring Permohonan Merek dan Persuratan Merek pada bagian 1 digambarkan proses ETL yang dibagi menjadi beberapa proses yaitu:

1. ETL Group Asal Permohonan, merupakan proses menarik data Group Asal Permohonan dari sumber data yaitu database IPAS, kemudian merubah struktur data agar dapat dimasukkan ke database yang akan memuat data kemudian memasukkan ke dalam data

warehouse yaitu tabel dimensi group asal permohonan, dengan nama tabel *dim\_group\_asal*.

2. ETL Data Group Status, merupakan proses menarik data Group Status dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi *group status*, dengan nama tabel *dim\_group\_status*.
3. ETL Data Surat Masuk, merupakan proses menarik data surat masuk dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel fakta Surat Masuk, dengan nama tabel *fakta\_surat*.
4. ETL Data Permohonan Masuk , merupakan proses menarik data permohonan masuk dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel fakta permohonan, dengan nama *fakta\_permohonan*.
5. ETL Data Sub Jenis Permohonan, merupakan proses menarik data sub jenis permohonan dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi sub jenis permohonan dengan nama tabel *dim\_sub\_jenis*.
6. ETL Data Asal Permohonan, merupakan proses menarik data asal permohonan dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi asal permohonan dengan nama tabel *dim\_asal*.
7. ETL Data Jenis Permohonan, merupakan proses menarik data Group jenis permohonan dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi jenis permohonan dengan nama tabel *dim\_jenis*.
8. ETL Data Status Permohonan, merupakan proses menarik data status permohonan dari sumber data yaitu *database IPAS*, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data*

*warehouse* yaitu tabel dimensi status permohonan, dengan nama tabel *dim\_status*.

9. ETL Data Jenis Surat, merupakan proses menarik data jenis surat dari sumber data yaitu *database* IPAS, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi jenis surat dengan nama tabel *dim\_s\_jenis*.
10. ETL Data Status Surat, merupakan proses menarik data status surat dari sumber data yaitu *database* IPAS, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi status surat dengan nama tabel *dim\_s\_status*.
11. ETL Data Group Jenis Surat, merupakan proses menarik data group jenis surat dari sumber data yaitu *database* IPAS, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi jenis surat dengan nama tabel *dim\_s\_group\_jenis*.
12. ETL Data Group Status Surat, merupakan proses menarik data group status surat dari sumber data yaitu *database* IPAS, kemudian merubah struktur data agar dapat dimasukkan ke *database* yang akan memuat data kemudian memasukkan ke dalam *data warehouse* yaitu tabel dimensi group status surat dengan nama tabel *dim\_s\_group\_status*.

Sedangkan pada bagian 2, merupakan permodelan untuk menampilkan data permohonan dan dimensi-dimensinya dari *data warehouse*, dimana terdapat 2 proses yaitu:

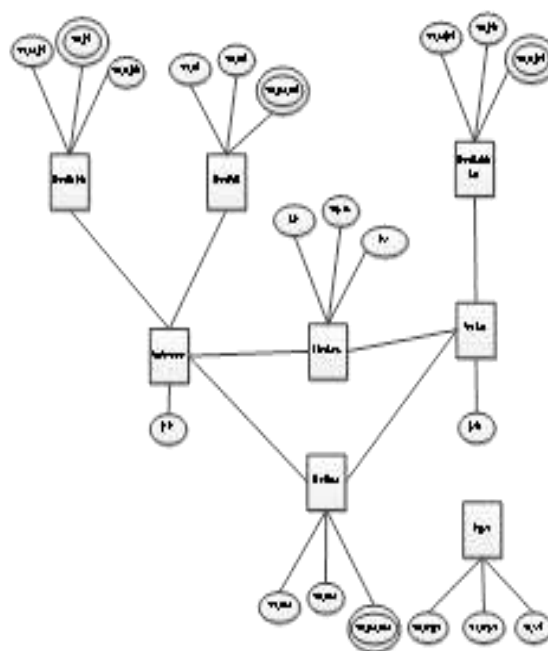
1. Memilih data permohonan yang ingin ditampilkan, dimana proses ini pengguna dapat melakukan fungsi *drill-down*, *drag-drop* untuk menampilkan data yang diinginkan, sehingga sesuai dengan keinginan dari pengguna.
2. Menampilkan data permohonan yang telah dipilih, dimana proses ini pengguna dapat melihat hasil dari pengolahan proses pertama dan menampilkan dalam bentuk grafik dan melakukan *export* ke Microsoft Excel.

Kemudian pada bagian 3, merupakan permodelan untuk menampilkan data persuratan masuk dan dimensi-dimensinya dari *data warehouse*, dimana terdapat 2 proses yaitu:

1. Memilih data persuratan masuk yang ingin ditampilkan, dimana proses ini pengguna dapat melakukan fungsi *drill-down*, *drag-drop* untuk menampilkan data yang diinginkan, sehingga sesuai dengan keinginan dari pengguna.
2. Menampilkan data persuratan masuk yang telah dipilih, dimana proses ini pengguna dapat melihat hasil dari pengolahan proses pertama dan menampilkan dalam bentuk grafik dan melakukan *export* ke Microsoft Excel

#### Entity Relationship

Berikut adalah perancangan *Entity Relationship* dari *data warehouse* dari EIS.



Gambar 4 Entity Relationship data warehouse

#### Conceptual Data Modeling

Berikut adalah perancangan *Conceptual Data Modelling* dari *data warehouse* dari EIS.



Gambar 5 Conceptual Data Modeling data warehouse

**Physical Data Model**

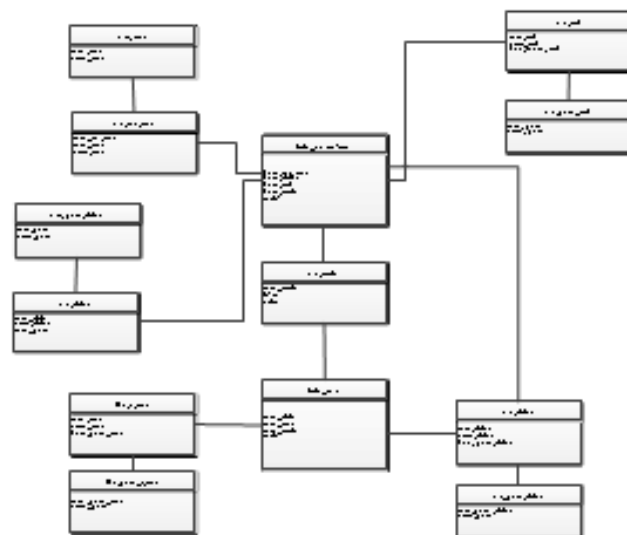
Berikut pada gambar 6 adalah perancangan Physical Data model dari data warehouse dari EIS.



Gambar 6 Physical Data Model data warehouse

**Skema Fact Constellation data warehouse**

Skema *Fact Constellation* memiliki beberapa tabel fakta berelasi dengan tabel dimensi dan beberapa diantaranya tabel dimensi yang sama.



Gambar 7 Skema Fact Constellation data warehouse

**Implementasi Rancangan AntarMuka**

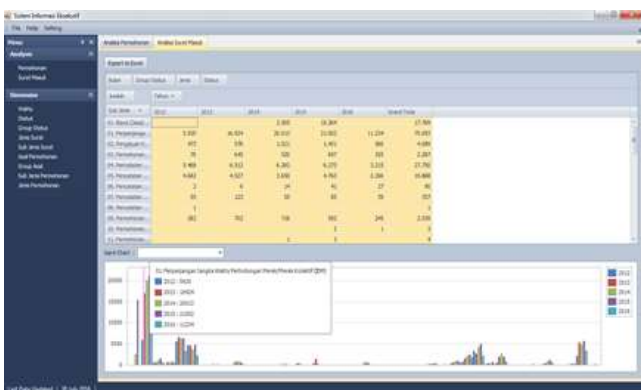
Form Analisa Permohonan pada gambar 8 berisi Tabel, dimana terdapat measures berupa jumlah dan dimensi-dimensi dimana pengguna dapat melakukan fungsi *drag-drop* dan *drill down* pada form. Pengguna juga dapat menghasilkan grafik dengan cara memblok nilai-nilai pada tabel sehingga akan akan membentuk grafik yang juga dapat diganti pada *combo box* ganti *chart*. Pengguna juga dapat mencetak kedalam bentuk Microsoft Excel dengan menekan tombol *Export To Excel* pada bagian atas form.



Gambar 8 Form Analisa Permohonan



Sedangkan Form Analisa Surat Masuk berisi Tabel, dimana terdapat *measures* berupa jumlah dan dimensi-dimensi dimana pengguna dapat melakukan fungsi *drag-drop* dan *drill down* pada *form*. Pengguna juga dapat menghasilkan grafik dengan cara memblok nilai-nilai pada tabel sehingga akan membentuk grafik yang juga dapat diganti pada *combo box* ganti *chart*. Pengguna juga dapat mencetak kedalam bentuk Microsoft Excel dengan menekan tombol *Export To Excel* pada bagian atas form.



Gambar 9 Form Analisa Persuratan

Selain Form Fakta juga terdapat Form dimensi-dimensi berisi daftar dimensi-dimensi yang digunakan, seperti gambar 10, gambar 11,

Gambar 10 List Dimensi Waktu

Gambar 11 List Dimensi Status

#### IV. KESIMPULAN

Kesimpulan yang dapat diambil, dengan adanya EIS dan *data warehouse* untuk memonitoring permohonan dan surat masuk ke Direktorat Merek dan Indikasi Geografis dapat membantu pengambilan keputusan dengan efektif dan efisien, dengan kecepatan dalam memproses data, dan kesesuaian fungsi untuk monitoring permohonan dan persuratan Direktorat Merek dan Indikasi Geografis. Hal tersebut dapat dibuktikan dengan penilaian kepada para responden.

Kesesuaian fungsi-fungsi yang dibutuhkan oleh Direktorat Merek dan Indikasi Geografis seperti Monitoring Permohonan dan Surat Masuk yang sebelumnya membutuhkan waktu lama dapat dilakukan semakin efektif dan efisien. Juga ditunjang oleh tampilan yang sesuai dan menarik serta kemudahan dalam pemahaman penggunaan akan mempermudah dalam penggunaan EIS.

Hasil dari EIS dan *data warehouse* ini dapat dipergunakan sebagai pertimbangan dalam pengambilan keputusan para eksekutif terkait dengan memonitor permohonan dan surat yang masuk pada Direktorat Merek dan Indikasi Geografis pada Direktorat Jenderal Kekayaan Intelektual.

#### REFERENSI

- [1] Direktorat Jenderal Kekayaan Intelektual, (2015), *Buku Panduan Kekayaan Intelektual*
- [2] Indrajani, (2014), *Database Systems Case Study All In One*. Elex Media Komputindo
- [3] Kadir A, (2013), *Pengenalan Sistem Informasi Edisi Revisi*, Andi Offset
- [4] *Materialized View* (t.thn.). Dipetik 9 21, 2016, dari <http://www.databasejournal.com/features>.
- [5] Nugroho. A, (2014), *Sistem Basis Data Oracle 10g*, Andi Offset



- [6] Pratama, (2014), *Komputer dan Masyarakat*, Informatika
- [7] Rosa A.S, (2013), *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Informatika
- [8] *Visual Basic .NET*, (t.thn.). Dipetik 8 4, 2016 <http://www.wokusoft.com/2015/02/apa-itu-vbnet-visual-basic-net.html>
- [9] Zimanyi E, Vaisman A, (2014), *Data Warehouse Systems Design and Implementation*, Springer