BINUS UNIVERSITY GRADUATE PROGRAM

People
Innovation
Excellence

LECTURE NOTES

7023T Advanced Database System

Session 08
Introducting Extract, Transform, Load

People
Innovation
Excellence

LEARNING OUTCOMES

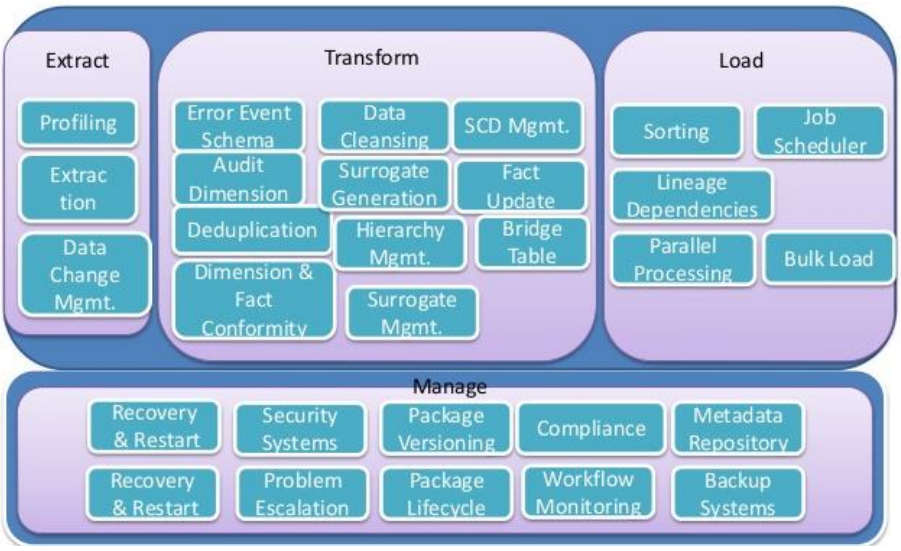
- Peserta diharapkan mampu memahami konsep dasar proses ETL (*extract – transform – load*).
- Peserta diharapkan dapat menjelaskan persyaratan dan kondisi (*requirements* dan *constraint*) apa saja yang perlu dipenuhi oleh ETL *tools.*
- Peserta diharapkan mampu mengidentifikasi kemampuan dari ETL *tools* yang saat ini beredar.
- Peserta diharapkan mampu memahami proses perancangan ETL.

OUTLINE MATERI (Sub-Topic):

1. *Introduction to* ETL
2. ETL *Requirements and Constraints*
3. ETL *Tools*
4. ETL *design*

Introduction to ETL

Seperti yang sudah dibahas pada bagian sebelumnya, proses ETL terdiri dari tiga operasi utama, yaitu: (E) ekstraksi data dari sumber data, (T) transformasi data untuk melakukan *cleansing* dan *conforming*, dan (L) *load* – atau memuat data ke *presentation server*. Jika operasi pengelolaan terhadap tiga operasi tersebut (ETL) juga dihitung, maka terdapat empat proses utama. Sebagian besar perangkat lunak data warehouse memasukkan ETL *tools* sebagai bagian utama dari sistem. Satu hal yang penting untuk diperhatikan adalah sekitar 70% waktu dan usaha dalam pengembangan sistem DW/BI akan difokuskan pada proses ETL. Hal lain yang perlu diperhatikan adalah harus diyakinkan bahwa ETL *tools* yang digunakan dapat bekerja dengan semua sumber data yang akan digunakan pada sistem DW/BI, jika tidak maka proses ETL harus dilakukan dengan cara menuliskan *script* SQL secara manual. Kimball membagi fungsionalitas ETL menjadi 34 subsistem seperti diperlihatkan pada gambar 1. Keseluruhan subsistem tersebut dapat dibagi menjadi 4 kelompok, yakni *Extract, Transform, Load, dan Manage.* Secara umum DW/BI *tools* mungkin saja menerapkan sedikit perbedaan fungsionalitas dari apa yang disarankan oleh Kimball.



Gambar 1. 34 ETL subsistem

ETL Requirements and Constraints

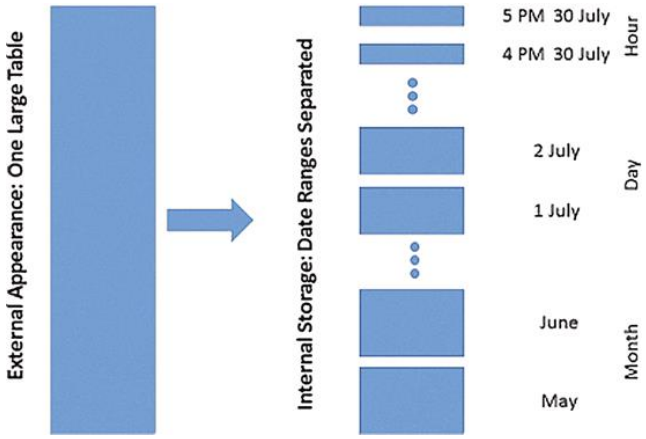
Tujuan utama dari komponen ETL, seperti komponen lain dari sebuah data warehouse adalah untuk memenuhi kebutuhan pengguna dari kalangan bisnis. Sebagian kebutuhan

tersebut dikumpulkan saat tahap *business requirement definition*, sementara lainnya dikumpulkan saat dilakukan investigasi terhadap sumber data pada tahap *data modelling*. Tim ETL akan mencocokkan keseluruhan kebutuhan *high level* terhadap sumber data yang tersedia. Pada saat ditemukan ketidakcocokkan, perlu dilakukan penyesuaian kebutuhan agar sesuai dengan ketersediaan data.

Sistem ETL juga perlu untuk mengikuti beberapa persyaratan (*requirements and constraints*), seperti *legal compliance, security,* dan *archiving.* Hal tersebut ditangani oleh subsistem ke 32 (*security system*) dan subsistem 33 (*compliance manager*). Beberapa aktifitas yang perlu dilakukan dalam rangka memenuhi persyaratan tersebut, seperti membuat salinan *backup*, memenuhi protokol yang sudah ditentukan, serta mendokumentasikan algoritma dan proses. Fakta di lapangan menunjukkan bahwa pelanggaran terhadap *security* lebih banyak berasal dari dalam organisasi dibanding dari luar, oleh karena itu sangat dianjurkan untuk menerapkan *role-based security* pada semua data dan metadata ETL. *Compliance* dalam hal ini memiliki makna mendokumentasikan seluruh proses yang terkait dengan data, sehingga setiap data dapat diketahui dari mana asalnya, proses transformasi apa yang sudah terjadi, keasliannya, serta salinannya.

Persyaratan ETL yang terkait dengan penyampaian data ke komponen-komponen lainnya termasuk data *latency* dan *formatting*. Data *latency* adalah *throughput dari* sistem ETL sebagai bagian dari keseluruhan sistem DW/BI. Pemilihan arsitektur perangkat keras yang tepat dan lingkungan perangkat lunak termasuk kegiatan yang dilakukan dalam rangka memenuhi persyaratan *data latency*. Persyaratan lain, seperti *real-time processing* adalah salah satu persyaratan yang umum dalam area ini. *Real-time processing* bermakna kebutuhan pengguna DW agar data selalu diperbaharui secara kontinu sepanjang hari. Pembaharuan tersebut dapat saja secara *instant*, *frequent*, atau *daily*. Pembaharuan data secara *instant* berarti informasi yang ditampilkan pada layar pengguna DW/BI diperbaharui segera setelah terjadi perubahan pada sumber data. Pembaharuan *instant* biasa ditemui dalam sistem *Enterprise Information Integration* (EII). Salah satu pendekatan yang dapat digunakan untuk mengoptimalkan proses pembaharuan data secara *real-time* adalah dengan menerapkan teknik *real-time partition* yang membagi tabel fakta secara fisik berdasarkan rentang waktu tertentu (misal: setiap jam, setiap hari, setiap minggu, setiap bulan, dan seterusnya). Gambar 2 memperlihatkan ilustrasi penerapan teknik *real-time partition.* Pembaharuan secara *frequent* berarti informasi yang ditampilkan pada layar pengguna DW/BI diperbaharui

beberapa kali dalam sehari, misalnya 15 menit sekali. Pembaharuan ini biasanya diimplementasikan sebagai *micro-batches* dalam arsitektur ETL.



Gambar 2. Ilustrasi teknik *real-time partition*

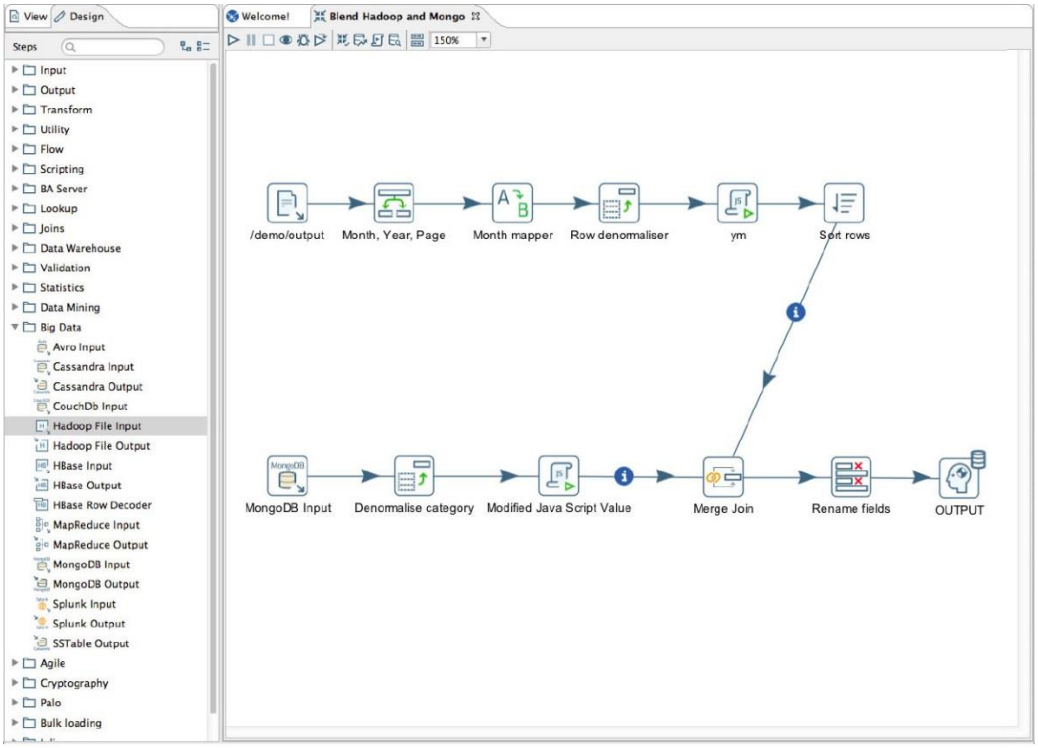
ETL *Tools*

Sebelum tersedia perangkat ETL *tools*, ETL *developer* perlu menghabiskan waktu banyak dalam menyiapkan *script* untuk melakukan proses ekstraksi, transformasi, dan memuat data kedalam data warehouse. ETL *tools* yang saat ini tersedia di pasaran sangat membantu ETL *developer* dalam melaksanakan tanggung-jawabnya. ETL *tools* menyediakan antarmuka grafis dengan kemampuan *drag-and-drop* untuk merancang proses ETL yang akan menghasilkan *script* untuk proses ekstraksi, transformasi, maupun pemuatan data ke dalam data warehouse. Script tersebut nantinya dapat dieksekusi secara manual maupun otomatis sesuai dengan jadwal yang sudah ditentukan. ETL *tools* dapat mengurangi waktu pengembangan secara signifikan. Beberapa ETL *tools* juga dilengkapi dengan opsi *template* yang memungkinkan sebuah program ETL disimpan dan digunakan untuk kebutuhan lain di kemudian hari. Gambar 3 memperlihatkan antarmuka grafis dari Pentaho Data Integration. Generasi terbaru dari ETL *tools* dilengkapi dengan dukungan yang lebih luas terhadap sistem database (Oracle, SQL Server, Teradata, DB2, MySQL, PostgreSQL, dll.), sistem ERP (SAP, PeopleSoft, dll.), perangkat lunak keuangan (NetSuite, QuickBook, Myob, dll.), maupun *tools reconciliation* seperti Hiperion, dan lain-lain.

Beberapa ETL *tools* menyediakan kemampuan *version control* dan pengembangan dalam lingkungan *multi-users*. *Tools* tersebut juga menyediakan catatan berupa *log* yang memberikan laporan dari proses ETL yang telah dijalankan. Catatan ini sangat bermanfaat untuk keperluan *troubleshooting* dan memungkinkan analisis trend untuk membandingkan

waktu yang dibutuhkan dalam mengolah data pada periode waktu berbeda. Selain itu. Catatan berupa *log* dapat dimanfaatkan untuk keperluan pelaporan.

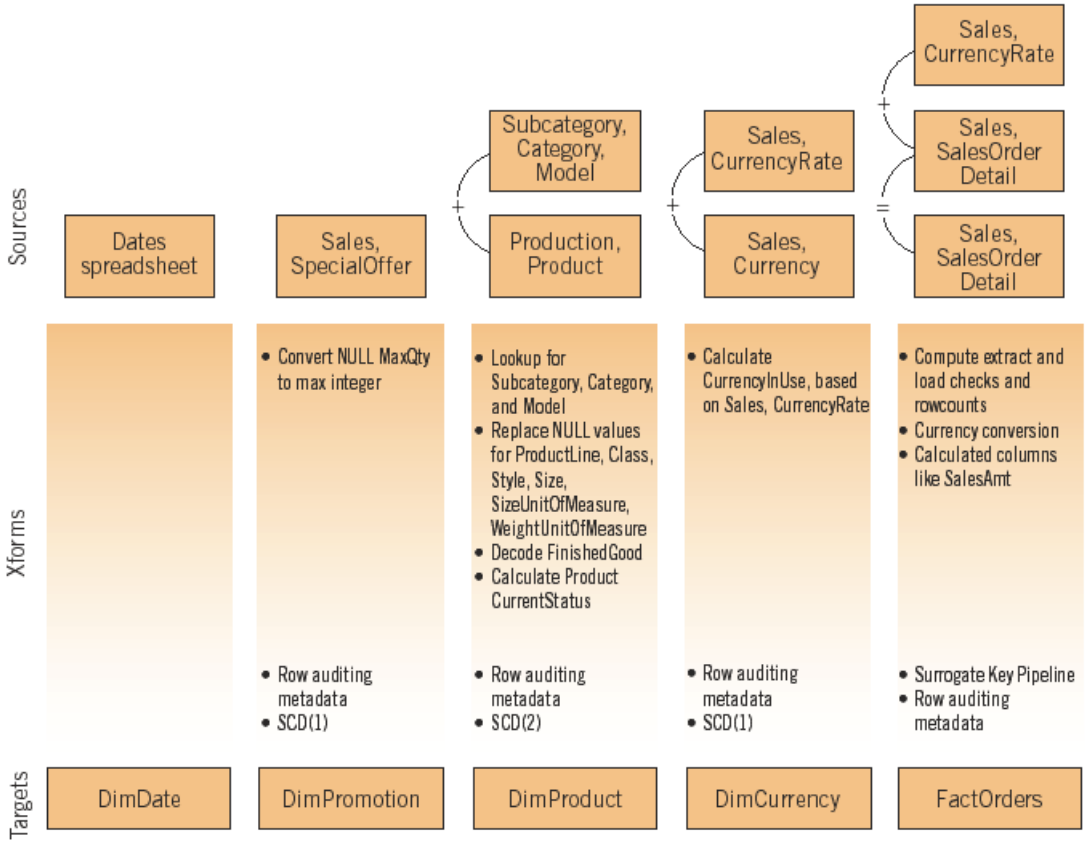
ETL *tools* juga menyediakan pilihan apakah proses pemuatan data akan dilakukan secara paralel atau sekuansial. Pemrosesan secara paralel memungkinkan pemuatan beberapa tabel dapat dilakulan secara simultan sehingga mengurangi waktu proses secara keseluruhan. ETL *tools* juga dilengkapi dengan kemampuan penjadwalan yang memungkinkan proses pemuatan data dilakukan pada waktu tertentu sesuai dengan jadwal yang ditentukan. Komponen lain yang penting dari perangkat lunak ETL adalah penanganan kesalahan dan notifikasi. Dengan adanya komponen ini, proses ETL dapat dikonfigurasi agar membatalkan pemuatan data segera setelah ditemukan kesalahan pertama kali, atau meneruskan proses pemuatan ke *record* berikutnya setelah mengalami kesalahan pada saat mengolah *record* tertentu. Komponen notifikasi dapat diatur agar memberikan informasi terkait proses ETL yang sudah dijalankan melalui email, baik proses ETL yang berhasil dilakukan maupun pada saat menemui kesalahan yang mengakibatkan kegagalan proses.



Gambar 3. Antarmuka grafis dari Pentaho Data Integration

ETL *Design*

Untuk proses desain ETL, arsitek ETL dan *developer* perlu memahami dan melakukan review terhadap model data dan arsitektur data. Model dimensional membantu proses review terhadap tabel-tabel dimensi dan tabel fakta. Jika tersedia *data dictionary*, maka hal tersebut dapat digunakan sebagai langkah awal untuk mendokumentasikan informasi mengenai sumber data untuk tabel-tabel dimensi dan tabel fakta, beserta atribut-atributnya. Jika tidak tersedia *data dictionary,* maka daftar sumber data dapat dibuat berdasarkan model dimensional. Model data juga bermanfaat dalam mempersiapkan dokumen pemetaan dari sumber ke tujuan (*source to target mapping* - STM) yang merupakan hal penting dalam proses ETL. STM membantu untuk mendokumentasikan aturan transformasi atau aturan pemetaan yang akan diterapkan pada saat memindahkan data dari satu layer ke layer lainnya pada DW/BI (sumber data → *staging database*, *staging database* → *operational data source*, *operational data source* → DW, DW → *data mining*). Gambar 4 memperlihatkan diagram *high level source to target mapping* untuk memetakan beberapa data/tabel sumber (Dates spreadsheet, Sales, SpecialOffer, SubCategory, Cagegory, Model, Production, Product, Sales, CurrencyRate, Currency, SalesOrder Detail) menjadi empat tabel dimensi (DimDate, DimPromotion, DimProduct, DimCurrency, dan FactOrder), serta proses tranformasi yang perlu dilakukan pada masing-masing data/tabel sumber untuk menjadi tabel tujuan (dimensi dan fakta).



Gambar 4. Contoh *high level source to target mapping*

Aktifitas dalam desain ETL juga termasuk mengidentifikasi waktu yang tepat untuk melakukan ekstraksi masing-masing sumber data, memuatnya ke beberapa layer DW/BI, urutan pemuatan dari setiap tabel, dan lain lain. Ketersediaan sumber data yang berasal dari berbagai sistem yang berbeda mungkin saja pada waktu yang berbeda. Sebagai contoh, informasi penjualan dari kantor cabang di negara amerika latin mungkin saja tersedia saat tengah malam, sementara informasi yang sama dari kantor cabang di negara asia tersedia saat tengah hari. Oleh karena itu, proses ETL terhadap kantor cabang yang terletak di benua yang berbeda tersebut perlu dilakukan pada saat yang berbeda. Jadwal pelaksanaan ekstraksi data juga perlu mempertimbangkan beban yang paling minimum ke sistem operasional agar tidak tidak terlalu mengganggu kegiatan bisnis sehari-hari. Apabila hanya terdapat sedikit atau tidak ada tranformasi yang terlibat pada saat pemindahan data dari sumber data ke *staging area*, maka kinerja dari sistem operasional tidak akan terganggu secara signifikan.

Pada umumnya, proses pemuatan data harus dilakukan dengan urutan tertentu karena adanya dependensi antara tabel-tabel sumber. Sebagai contoh, tabel-tabel dimensi biasanya dimuat terlebih dahulu sebelum tabel fakta. Hal ini akan menghindarkan pelanggaran terhadap *referential integrity constraints*. ETL *tools* kini telah dilengkapi dengan opsi untuk

mengkonfigurasi *workflow* atau *sequences* untuk melakukan pemuatan tabel-tabel dimensi dan tabel fakta sesuai dengan urutan yang diinginkan.

SIMPULAN

- Sekitar 70% waktu dan usaha dalam pengembangan sistem DW/BI akan difokuskan pada proses ETL.
- Kimball membagi fungsionalitas ETL menjadi 34 subsistem yang dapat dikelompokkan menjadi 4 kategori yakni *Extract, Transform, Load,* dan *Manage*.
- ETL *tools* saat ini sudah dilengkapi dengan antarmuka grafis yang sangat membantu ETL *developer* dalam melakukan proses ETL, sehingga mereka tidak perlu lagi menghabiskan waktu untuk membuat *script* ETL.
- Dokumen STM (*source to target mapping*) sangat penting dalam desain ETL, dokumen ini membantu mendokumentasikan aturan transformasi atau aturan pemetaan yang akan diterapkan pada saat memindahkan data dari satu layer ke layer lainnya pada DW/BI.

BINUS UNIVERSITY GRADUATE PROGRAM

People
Innovation
Excellence

DAFTAR PUSTAKA

1. Kimball, R. (2008). *The Data Warehouse Lifecycle Toolkit*. John Wiley & Sons.
2. Kimball, R., & Ross, M. (2011). *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. John Wiley & Sons.
3. Inmon, W. H. (2005). *Building the Data Warehouse*. John wiley & sons.