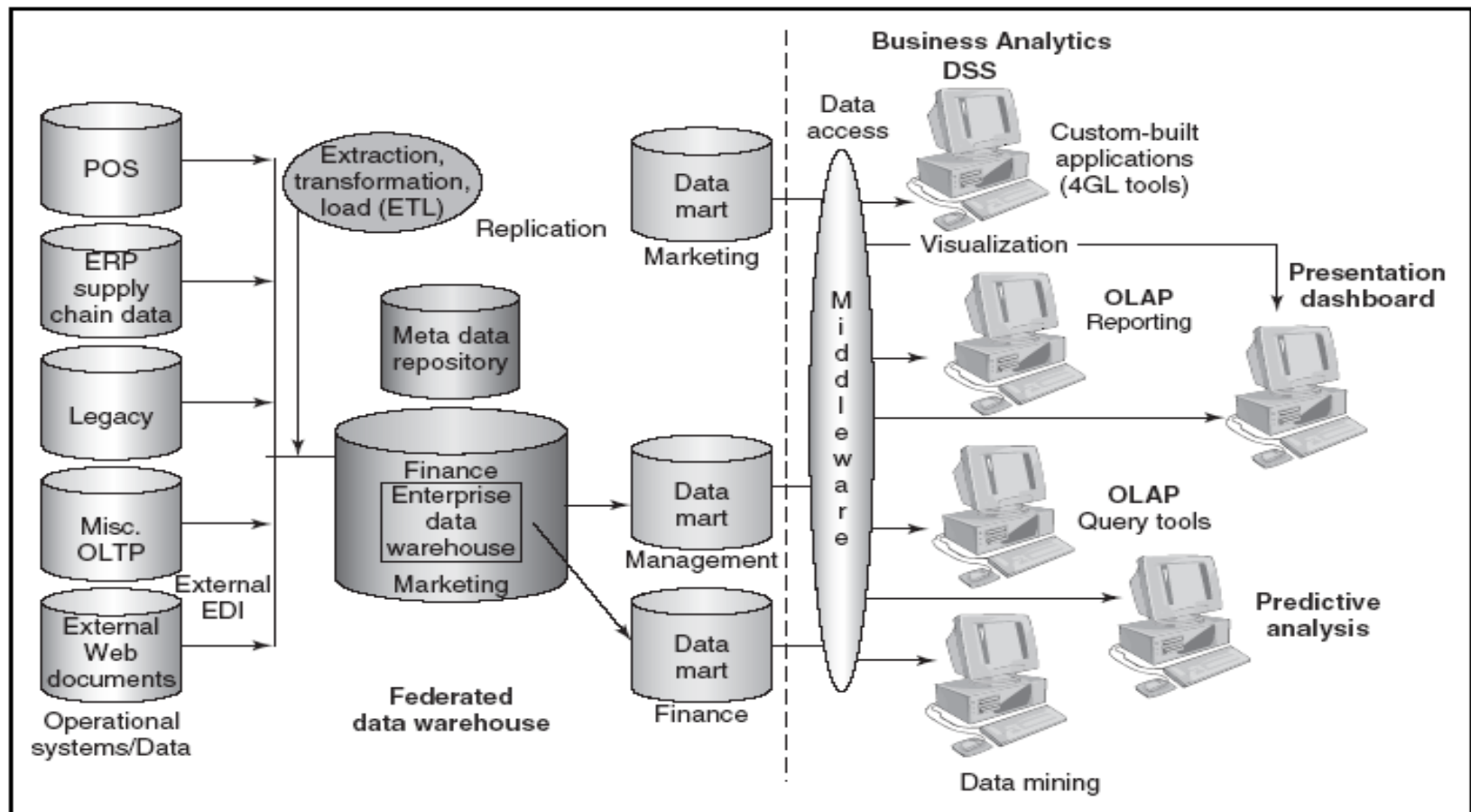


ARSITEKTUR DATA WAREHOUSE (2)

Arsitektur *Business Intelligence*

Menurut Turban :



Komponen *Data Warehouse*

Menurut Turban :

- *Data Sources*

Sumber data dapat berasal dari *OLTP*, sistem *ERP*, sistem operasional yang ada , data eksternal, atau sumber data lainnya.

- *Data Extraction*

Data diekstrak menggunakan software *ETL*.

- *Data Loading*

Data dimasukkan dalam *staging area* untuk dilakukan proses transformasi dan pembersihan. Data yang telah siap selanjutnya, dimasukkan ke dalam *Data Warehouse*.

Komponen *Data Warehouse*

Menurut Turban :

- *Comprehensive Database*

Data Warehouse memberikan analisa penunjang keputusan dengan menyediakan hasil ringkasan dan informasi rinci yang relevan berasal dari berbagai sumber.

- *Metadata*

Metadata merupakan proses identifikasi atribut dan struktur dari sebuah data atau informasi untuk menjelaskan *database* secara rinci yaitu dari nama, tipe, dan ukuran atribut. Hal ini berguna untuk membantu memperjelas analisis dan pembuatan laporan.

- *Middleware*

Middleware adalah software penghubung yang berisi sekumpulan layanan yang memungkinkan beberapa proses dapat berjalan pada satu atau lebih mesin untuk saling berinteraksi pada suatu jaringan. Dalam hal ini *software Middleware* ini akan menghubungkan *Data Warehouse* dan berbagai *tool* analisa seperti *OLAP*, *Data Mining*, *Business Report*, dan *Dashboard*.

Data Mart

- Data Mart adalah subyek spesifik atau aplikasi spesifik dari Data Warehouse yang berisi data untuk satu line bisnis tertentu semisal penjualan atau pemasaran. Perbedaan antara Data Mart dan Data Warehouse adalah pada ruang lingkup informasi yang disediakan. Ruang lingkup Data Mart lebih kecil, data didapatkan dari sumber yang lebih sedikit dan waktu implementasinya juga lebih pendek.

Data Mart

- Data Mart dapat bersifat dependent atau independent, tergantung pada sumber informasinya.
- Sumber informasi untuk Data Mart bersifat dependent apabila berasal dari Data Warehouse itu sendiri.
- Sumber informasi untuk Data Mart bersifat independent apabila tidak berasal dari Data Warehouse, dan data diekstrak secara langsung dari sistem operasional.

Alternative Arsitektur Data Warehouse

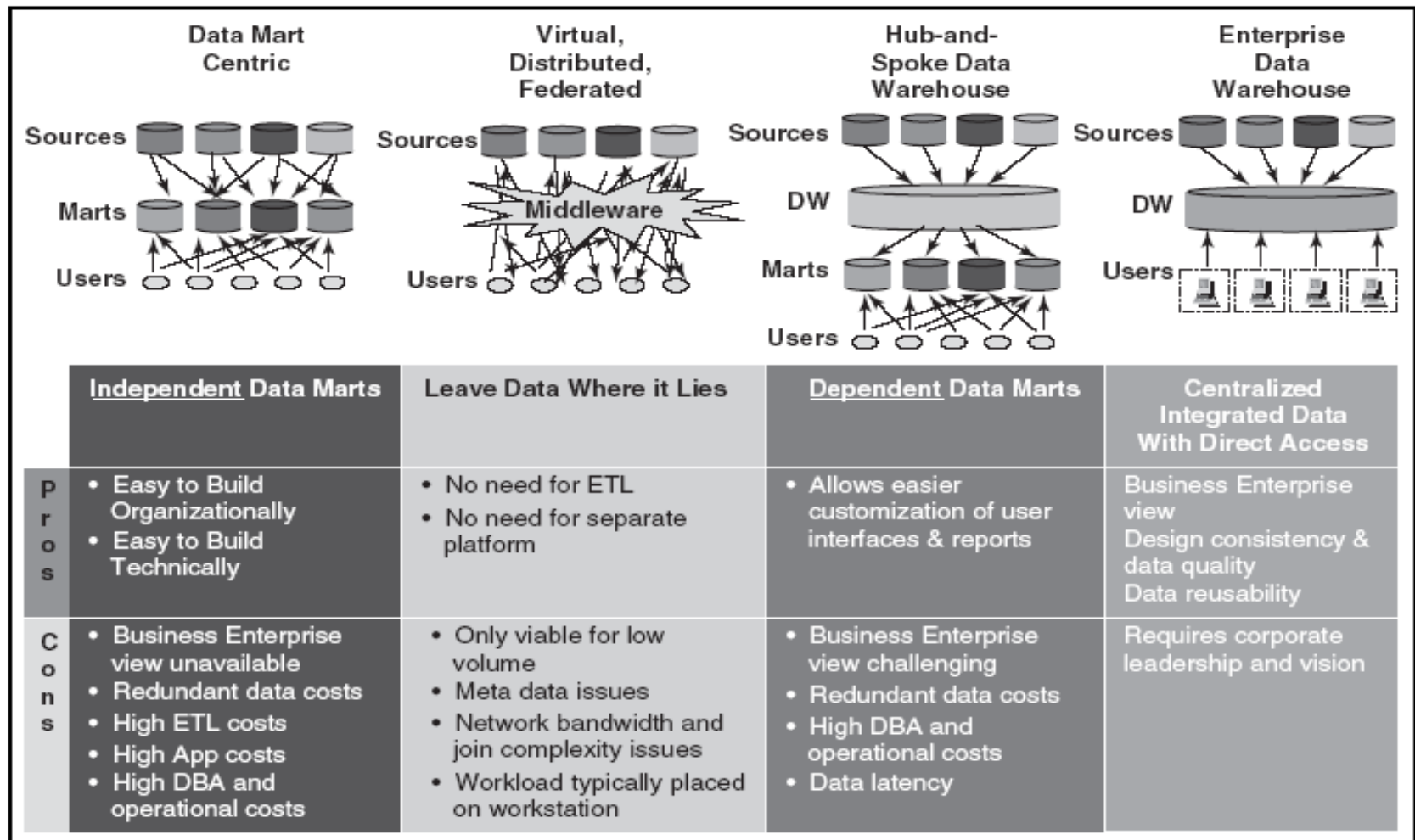


FIGURE 5.6 Alternative Architectures for Data Warehousing Efforts

Mengapa Sistem Data Warehouse harus dipisah dari Sistem Operasional?

- Untuk mendapatkan proses dengan High Performance dalam melakukan analisa
- Pelaksanaan proses analisa queri pada server operasional dapat menurunkan kinerja server.
- Pada server operasional dibutuhkan kontrol mekanisme untuk menjamin konsistensi transaksi
- Data Warehouse hanya mempunyai fungsi membaca (read)
- Data pada server operasional belum bisa mendukung untuk analisa dalam pengambilan keputusan.
- Data pada server operasional belum terkonsolidasi sedangkan Data Warehouse merupakan data konsolidasi hasil integrasi dari berbagai sumber data.
- Data Warehouse memerlukan adanya mekanisme rekonsiliasi atas kode dan format data.

Mengapa Sistem Data Warehouse harus dipisah dari Sistem Operasional?

Menurut Carlo Vercelis, terdapat beberapa alasan mengapa sistem Data Warehouse harus dipisah dari Sistem Operasional :

- Integration

Untuk melakukan analisa bisnis yang menggunakan data/informasi dari berbagai sumber data maka diperlukan standarisasi dan integrasi data.

- Quality

Untuk mendapatkan hasil analisa bisnis yang tepat diperlukan kualitas data yang benar. Hal ini didapatkan di sistem *Data Warehouse* yang mempunyai data yang sudah konsisten hasil dari proses cleansing dan transformasi dari sistem operasional.

Mengapa Sistem Data Warehouse harus dipisah dari Sistem Operasional?

- Efficiency

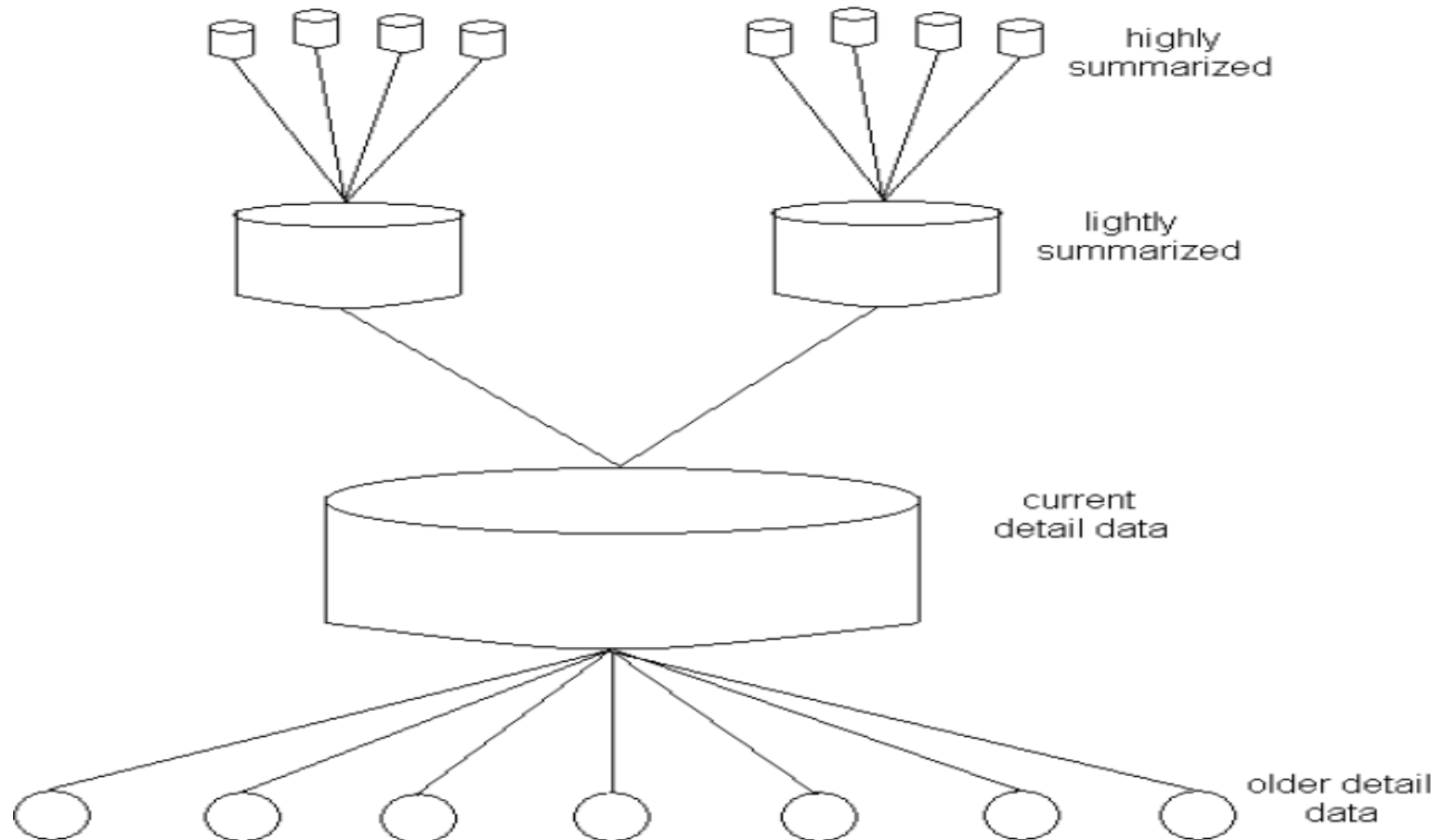
Pekerjaan analisa bisnis di sistem Data Warehouse dapat dilakukan lebih efisien. Hal ini dikarenakan proses query data pada sistem Data Warehouse dikerjakan pada data yang sudah terintegrasi, lengkap, terstruktur dan sudah merupakan agregasi data.

- Extendability

Pekerjaan analisa membutuhkan data historikal yang dapat disediakan oleh sistem Data Warehouse secara lengkap baik data masa lampau atau masa kini.

Struktur Data Warehouse

- Menurut Poe



Komponen Struktur Data Warehouse

Current detail data

- *Current detail data* merupakan data detil yang aktif saat ini, mencerminkan keadaan yang sedang berjalan dan merupakan level terendah dalam *data warehouse*. Didalam area ini warehouse menyimpan seluruh detail data yang terdapat pada skema basis data. Jumlah data sangat besar sehingga memerlukan *storage* yang besar pula dan dapat diakses secara cepat. Dampak negatif yang ditimbulkan adalah kerumitan untuk mengatur data menjadi meningkat dan biaya yang diperlukan menjadi mahal.

Komponen Struktur Data Warehouse

- Berikut ini beberapa alasan mengapa *current detail data* menjadi perhatian utama :
 - Menggambarkan kejadian yang baru terjadi dan selalu menjadi perhatian utama
 - Sangat banyak jumlahnya dan disimpan pada tingkat penyimpanan terendah.
 - Hampir selalu disimpan dalam *storage* karena cepat di akses tetapi mahal dan kompleks dalam pengaturannya.
 - Bisa digunakan dalam membuat rekapitulasi sehingga *current detail data* harus akurat.

Komponen Struktur Data Warehouse

Older detail data

- Data ini merupakan data historis dari *current detail data*, dapat berupa hasil cadangan atau archive data yang disimpan dalam *storage* terpisah. Karena bersifat *back-up*(cadangan), maka biasanya data disimpan dalam *storage* alternatif seperti *tape-desk*.
- Data ini biasanya memiliki tingkat frekuensi akses yang rendah. Penyusunan file atau *directory* dari data ini di susun berdasarkan umur dari data yang bertujuan mempermudah untuk pencarian atau pengaksesan kembali.

Komponen Struktur Data Warehouse

Lightly summarized data

- Data ini merupakan ringkasan atau rangkuman dari *current detail data*. Data ini dirangkum berdasar periode atau dimensi lainnya sesuai dengan kebutuhan.
- Ringkasan dari *current detail data* belum bersifat total *summary*. Data-data ini memiliki detail tingkatan yang lebih tinggi dan mendukung kebutuhan *warehouse* pada tingkat departemen. Tingkatan data ini di sebut juga dengan *data mart*. Akses terhadap data jenis ini banyak digunakan untuk view suatu kondisi yang sedang atau sudah berjalan.

Komponen Struktur Data Warehouse

Highly summarized data

- Data ini merupakan tingkat lanjutan dari *Lightly summarized data*, merupakan hasil ringkasan yang bersifat totalitas, dapat diakses misal untuk melakukan analisis perbandingan data berdasarkan urutan waktu tertentu dan analisis menggunakan data multidimensi.

Komponen Struktur Data Warehouse

Metadata

- Metadata bukan merupakan data hasil kegiatan seperti keempat jenis data diatas. Menurut Poe, metadata adalah 'data tentang data' dan menyediakan informasi tentang struktur data dan hubungan antara struktur data di dalam atau antara *storage* (tempat penyimpanan data).
- Metadata berisikan data yang menyimpan proses perpindahan data meliputi database structure, contents, detail data dan summary data, matrices, versioning, aging criteria, versioning, transformation criteria. Metadata khusus dan memegang peranan yang sangat penting dalam data warehouse.

Komponen Struktur Data Warehouse

Metadata berisi :

- Struktur data

Sebuah direktori yang membantu user untuk melakukan analisis *Decision Support System* dalam pencarian letak/lokasi dalam data warehouse.

- Algoritma

Algoritma digunakan untuk summary data. Metadata sendiri merupakan panduan untuk algoritma dalam melakukan pemrosesan *summary* data antara *current detail data* dengan *lightly summarized data* dan antara *lightly summarized data* dengan *highly summarized data*.

- *Mapping*

Sebagai panduan pemetaan (*mapping*) data pada saat data di transform/diubah dari lingkup operasional menjadi lingkup data warehouse.