COP 4710: Database Systems
Spring 2006

CHAPTER 25 – Data Warehousing – Part 2

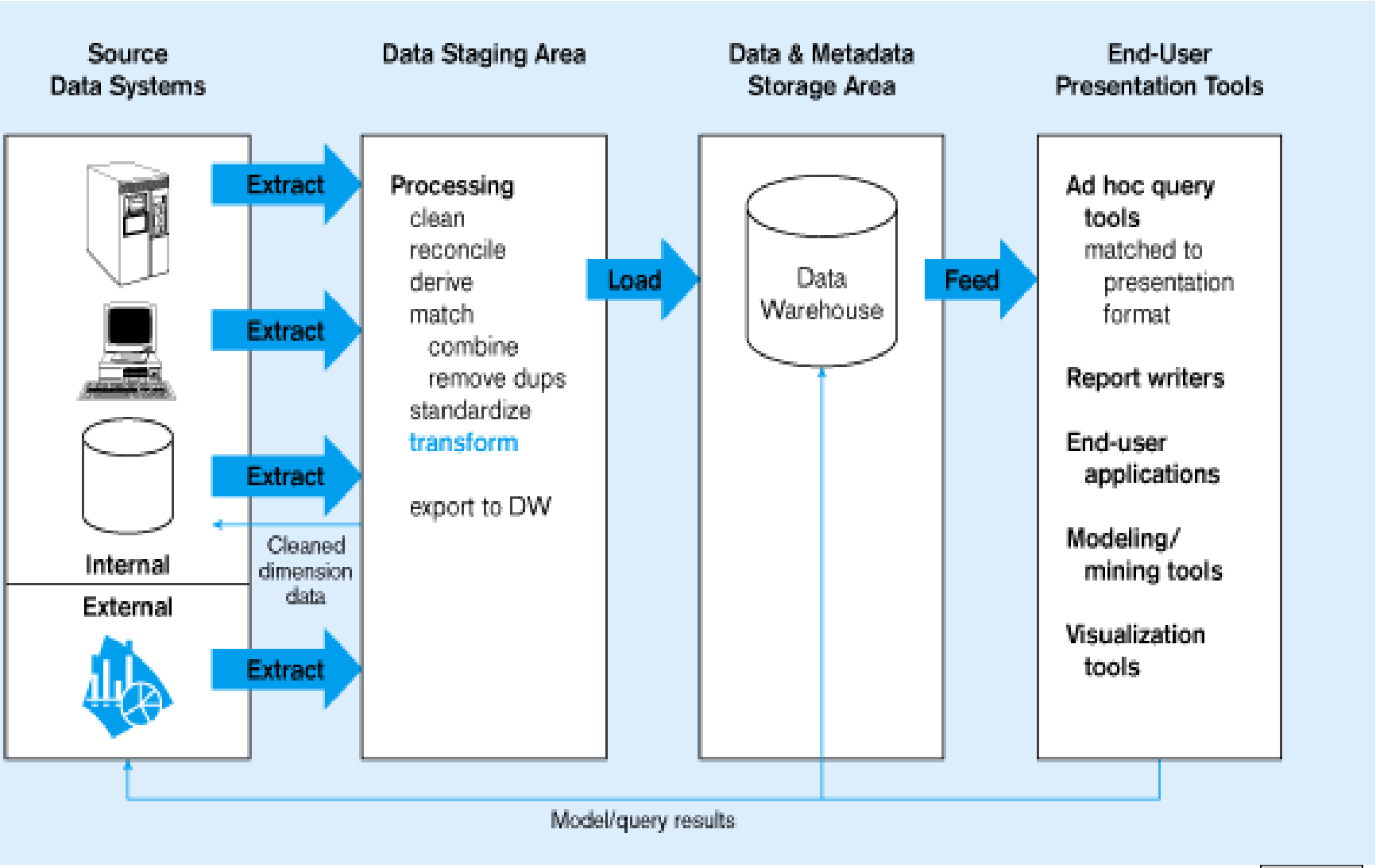
Instructor : Mark Llewellyn
markl@cs.ucf.edu
CSB 242, 823-2790
http://www.cs.ucf.edu/courses/cop4710/spr2006

School of Electrical Engineering and Computer Science
University of Central Florida

Summary of Differences in Operational Databases and Data Warehouses

Characteristic	Operational DB	Data Warehouse
Primary purpose	Run the business on a real-time basis (current basis)	Support managerial decision making
Type of data	Current representation of the state of the business	Historical point in time (snapshots) and predictions
Primary users	Clerks, salespersons, administrators	Managers, business analysts, customers
Scope of usage	Narrow, planned, simple updates and queries	Broad, ad hoc, complex queries and analysis
Design goal	Performance, throughput, availability	Ease of flexible access and use
Volume	Many, constant updates and queries on one or a few table rows	Periodic batch updates and queries involving many or all rows

Generic Two-level Data Warehouse



Generic Two-level Data Warehouse (cont.)

- Building a data warehouse, like that shown in the previous slide requires four basic steps (moving left to right in the picture):

1. Data are extracted from the various internal and external source files and databases. In large organizations there may be dozens or hundreds of such sources.
2. The data from the various sources are transformed and integrated before being loaded into the warehouse. Transactions may be sent to source systems to correct errors discovered in data staging.
3. The data warehouse is organized for decision support. It contains both detailed and summary data.
4. Users access the warehouse by means of a variety of query languages and analytical tools. Results (e.g., predictions, forecasts) may be fed back into the warehouse and operational databases.

Introduction to OnLine Analytical Processing

- The need for more intensive decision support prompted the introduction of a new generation of tools. These new tools, called online analytical processing (OLAP), create an advanced data analysis environment that supports decision making, business modeling, and operations research.

- OLAP systems share four main characteristics:

  1. Use multidimensional data analysis techniques.
  2. Provide advanced database support.
  3. Provide easy-to-use end-user interfaces.
  4. Support client/server architectures.

Multidimensional Data Analysis Techniques

- The most distinct characteristic of OLAP tools is their capacity for multidimensional analysis. In multidimensional analysis, data are processed and viewed as part of a multidimensional structure. This view of data analysis is particularly attractive to business decision makers because they tend to view business data as data that are related to other business data.

- Multidimensional analysis techniques are augmented by:

  – Advanced data presentation functions: 3D graphics, pivot tables, crosstabs, data rotation, three-dimensional cubes, and so on.

  – Advanced data aggregation, consolidation, and classification functions that all the business data analyst to create multiple data aggregation levels, slice and dice, and drill down and roll up data across different dimensions and aggregation level.s. For example aggregating data across the time dimension (by day, week, month, quarter, year) allows the analyst to drill down and roll up across time dimensions.

  – Advanced computational functions: business-oriented variables (market share, period comparisons, sales margins), financial and accounting ratios (profitability, overhead, cost allocations, returns, etc.).

  – Advanced data modeling functions: support for "what-if" scenarios, variable assessment, linear programming, variable contributions to outcome, etc.

Advanced Database Support

- OLAP tools must have many advanced data access features. These features include:
  - Access to many different kinds of DBMSs, flat files, and internal and external data sources.
  - Access to aggregated data warehouse data as well as to the detailed data found in operational databases.
  - Rapid and consistent query response times.
  - The ability to map end-user requests, expressed in either business or model terms, to the appropriate data source and then to the proper data access language (typically SQL). The query code must be optimized to match the data source, regardless of whether the source is operational or warehouse data.
  - Support for VLDBs (Very Large Databases).

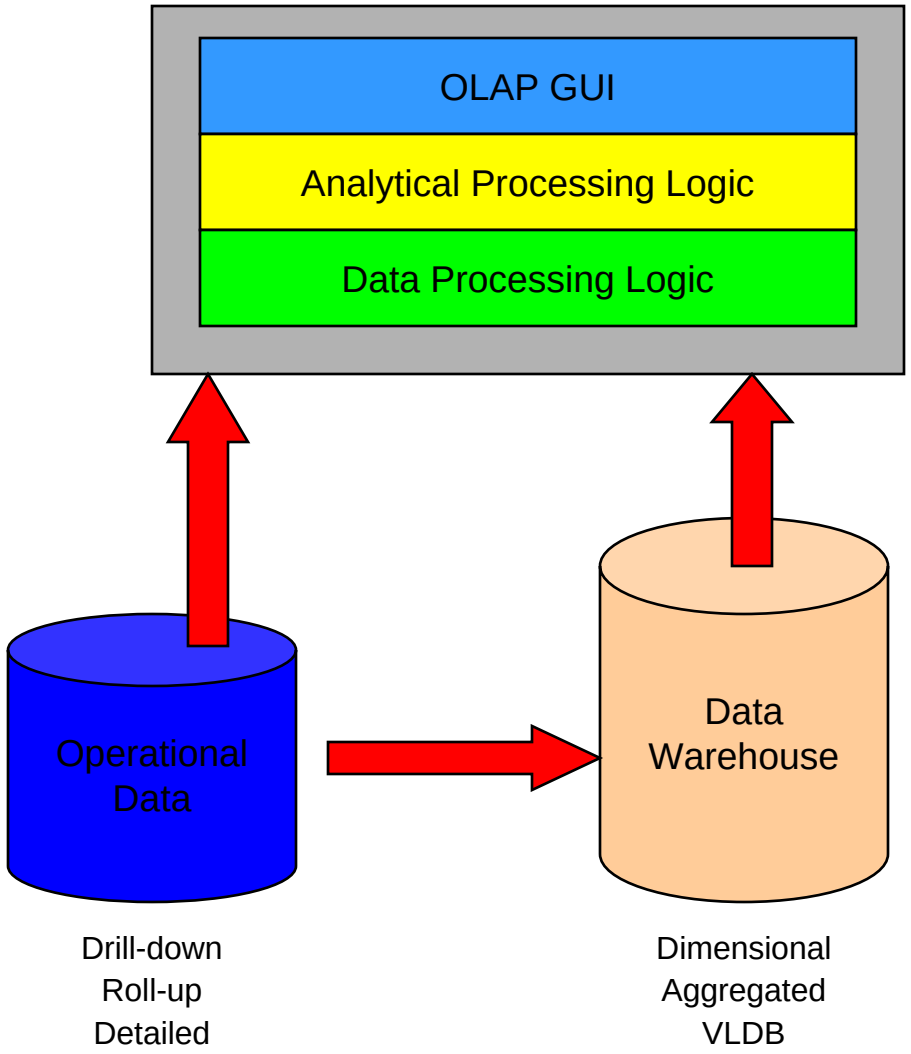
Easy to Use End User Interface

- Developers of OLAP tools learned very early in the game that OLAP tools are much more useful if access to them is kept simple.
- Most of the commercially available OLAP tools have easy to user GUIs and many of the their features have been borrowed from previous generations of data analysis tools that are already familiar to end users.
- More information about various OLAP tools can be obtained from www.olapreport.com. (This is a subscription site, but you can see many details without a subscription.)

Client/Server Architecture

- Client/server architecture provides a framework within which new systems can be designed, developed, and implemented.
- The client/server environment allows us to look at an OLAP system as if it consists of several components that define its architecture.
- The components of the OLAP can be placed on a single computer system or distributed among several computers.
- The OLAP operational characteristics can be divided into three main modules:
  - GUI (graphical user interface).
  - Analytical processing logic.
  - Data-processing logic.

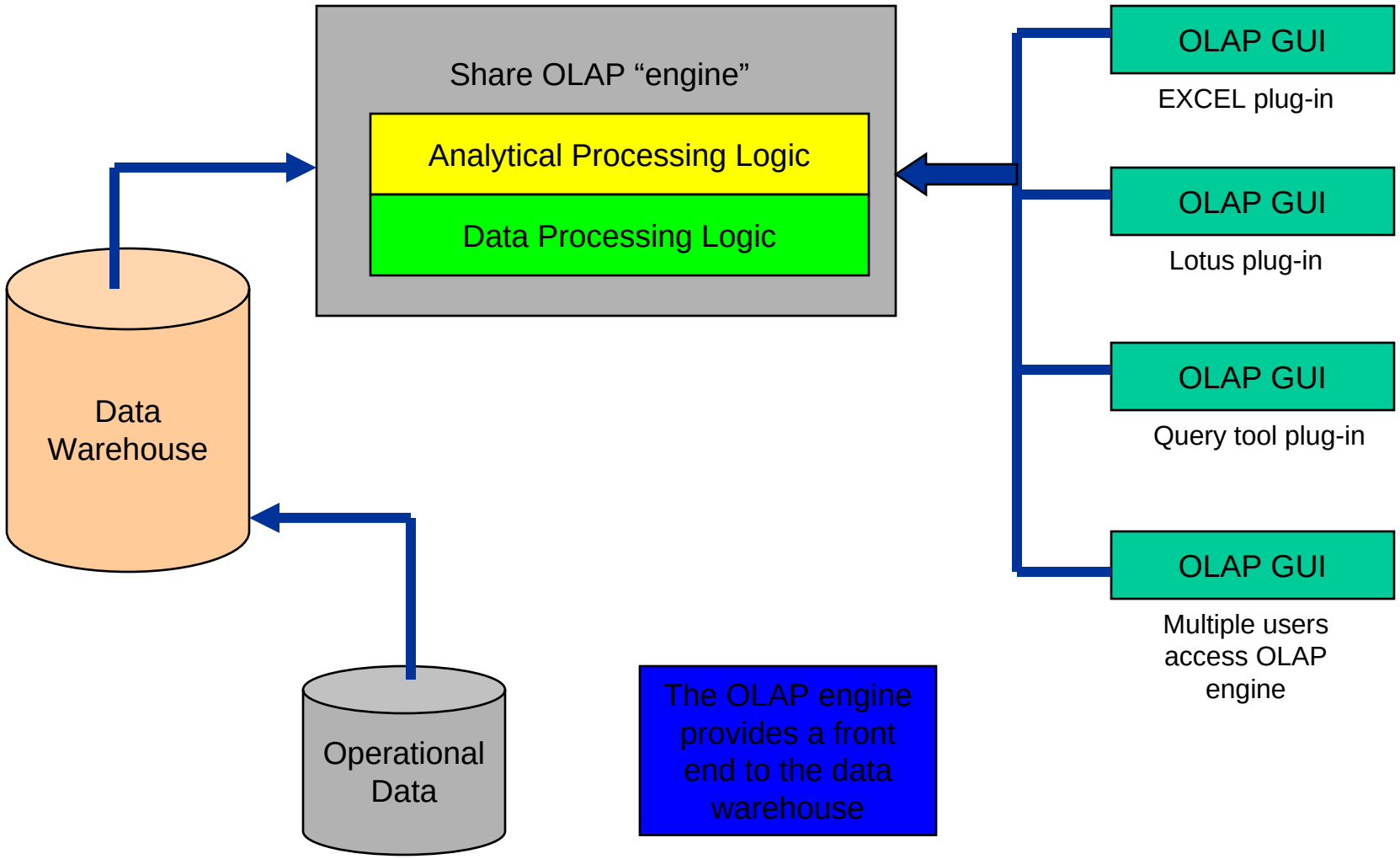
OLAP Client/Server Architecture



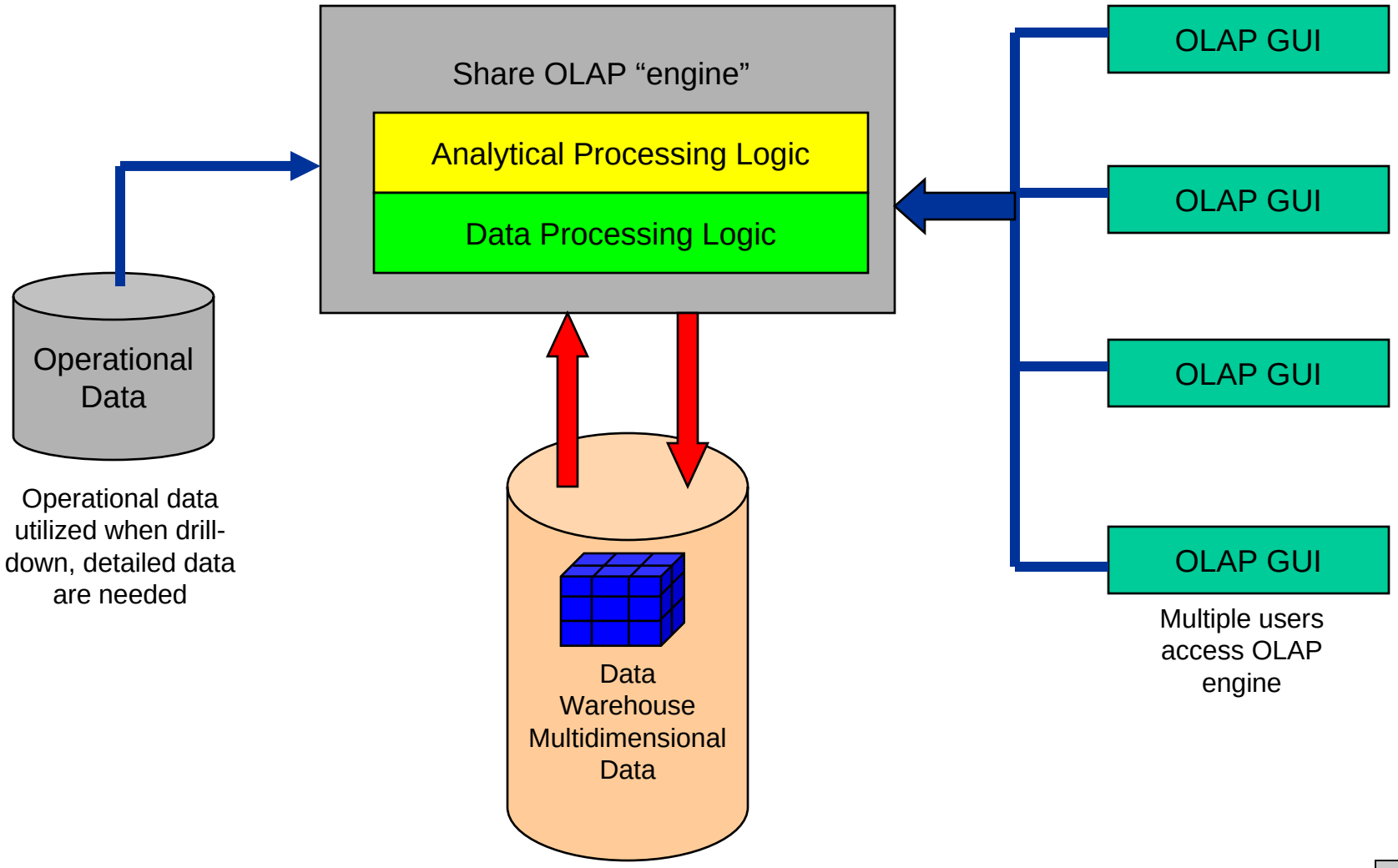
OLAP System exhibits

- Client/Server Architecture
- Easy-to-use GUI
  - Dimensional presentation
  - Dimensional modeling
  - Dimensional analysis
- Multidimensional data
  - Analysis
  - Manipulation
  - Structure
- Database support
  - Data warehouse
  - Operational database
  - Relational
  - Multidimensional

OLAP Server Arrangement



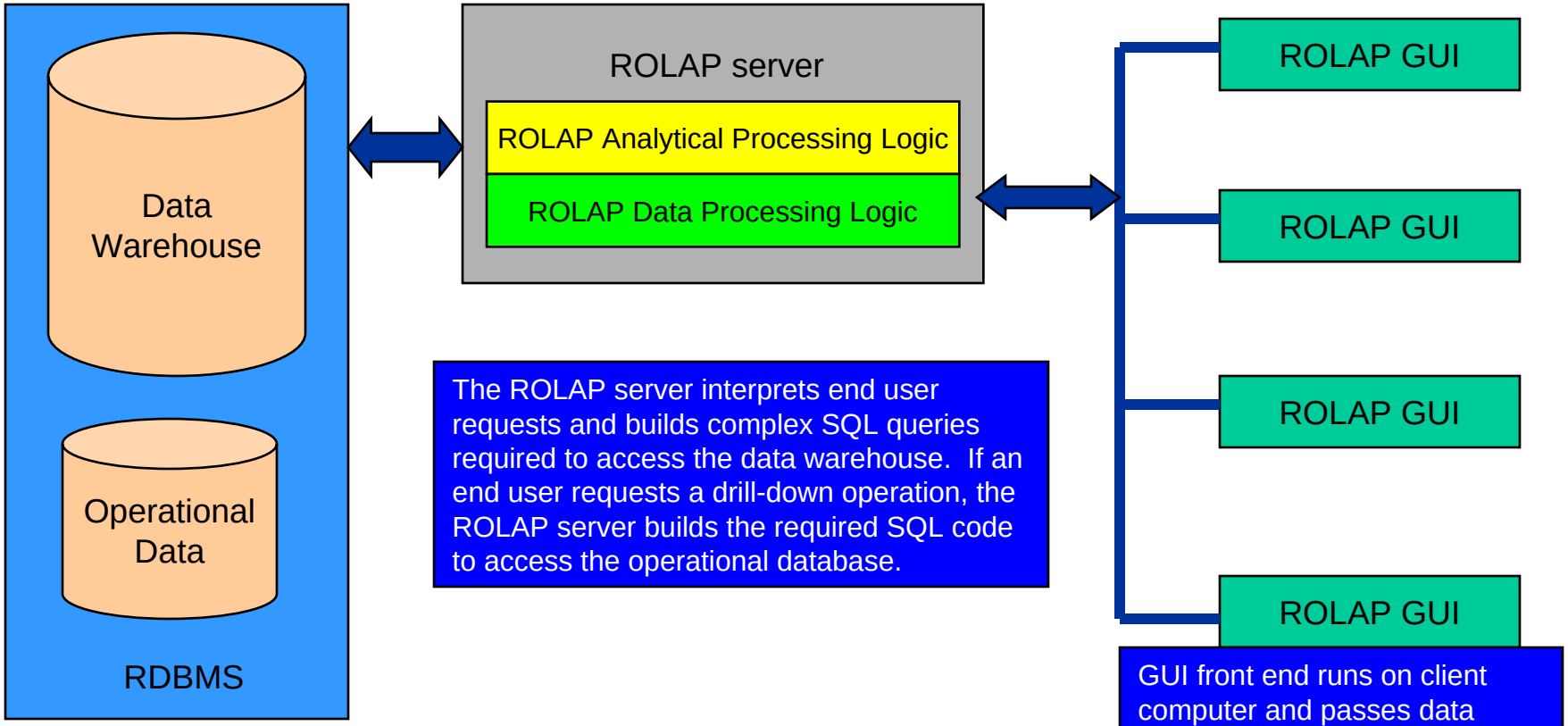
OLAP Server with Multidimensional Data Store Arrangement



Relational OnLine Analytical Processing (ROLAP)

- Relational OnLine Analytical Processing (ROLAP) provides OLAP functionality by using relational databases and familiar relational query tools to store and analyze multidimensional data.
- This approach builds on existing relational technologies and represents a natural extension for relational database vendors.
- ROLAP adds the following extensions to traditional RDBMS technology:
  - Multidimensional data schema support within the RDBMS.
  - Data access language and query performance optimized for multidimensional data.
  - Support for VLDBs.

ROLAP System



The ROLAP server interprets end user requests and builds complex SQL queries required to access the data warehouse. If an end user requests a drill-down operation, the ROLAP server builds the required SQL code to access the operational database.

An RDBMS is used to manage both the data warehouse and the operational data. The RDBMS executes data requests and sends back the data sets.

GUI front end runs on client computer and passes data analysis to the ROLAP server. The GUI receives data replies from the ROLAP server and formats them according to the end user's presentation needs.

Relational OnLine Analytical Processing (ROLAP)

- Relational technology utilizes normalized tables to store data. This reliance on normalized data, while a benefit to the normal relational system, is viewed as a stumbling block in OLAP systems.
- As you will recall, normalization divides tables into smaller pieces to produce the normalized tables. Normalization is useful for reducing redundancies and eliminating certain types of data anomalies.
- Unfortunately, for decision support purposes, it is easier to understand data when they are seen with respect to other data. Normalization tends to preclude this possibility.
- Fortunately, particularly for those businesses which are heavily invested in relational technology, ROLAP uses a special design technique to enable RDBMS technology to support multidimensional data representations. This technique is called the star schema.

An Aside On The Star Schema

- The star schema is a data modeling technique used to map multidimensional decision support data into a relational database. In effect, the star schema creates the near equivalent of a multidimensional database schema from the existing relational database.
- Star schemas yield an easily implemented model for multidimensional data analysis, while still preserving the relational structures on which the operational database is built.
- The basic star schema has four components:
  - facts
  - dimensions
  - attributes
  - attribute hierarchies.

An Aside On The Star Schema (cont.)

- Facts are numeric measurements (values) that represent a specific business aspect or activity. For example, sales figures. Facts are normally stored in a fact table that is the center of the star schema. The fact table contains facts that are linked through their dimensions.

- Dimensions are qualifying characteristics that provide additional perspectives to a given fact. Dimensional data is stored in dimension tables. Recall that DSS data are almost always viewed in relation to other data. For instance, sales might be compared by product from region to region, and from one time period to the next.

  – In effect, dimensions are the magnifying glass through which the facts are studied.

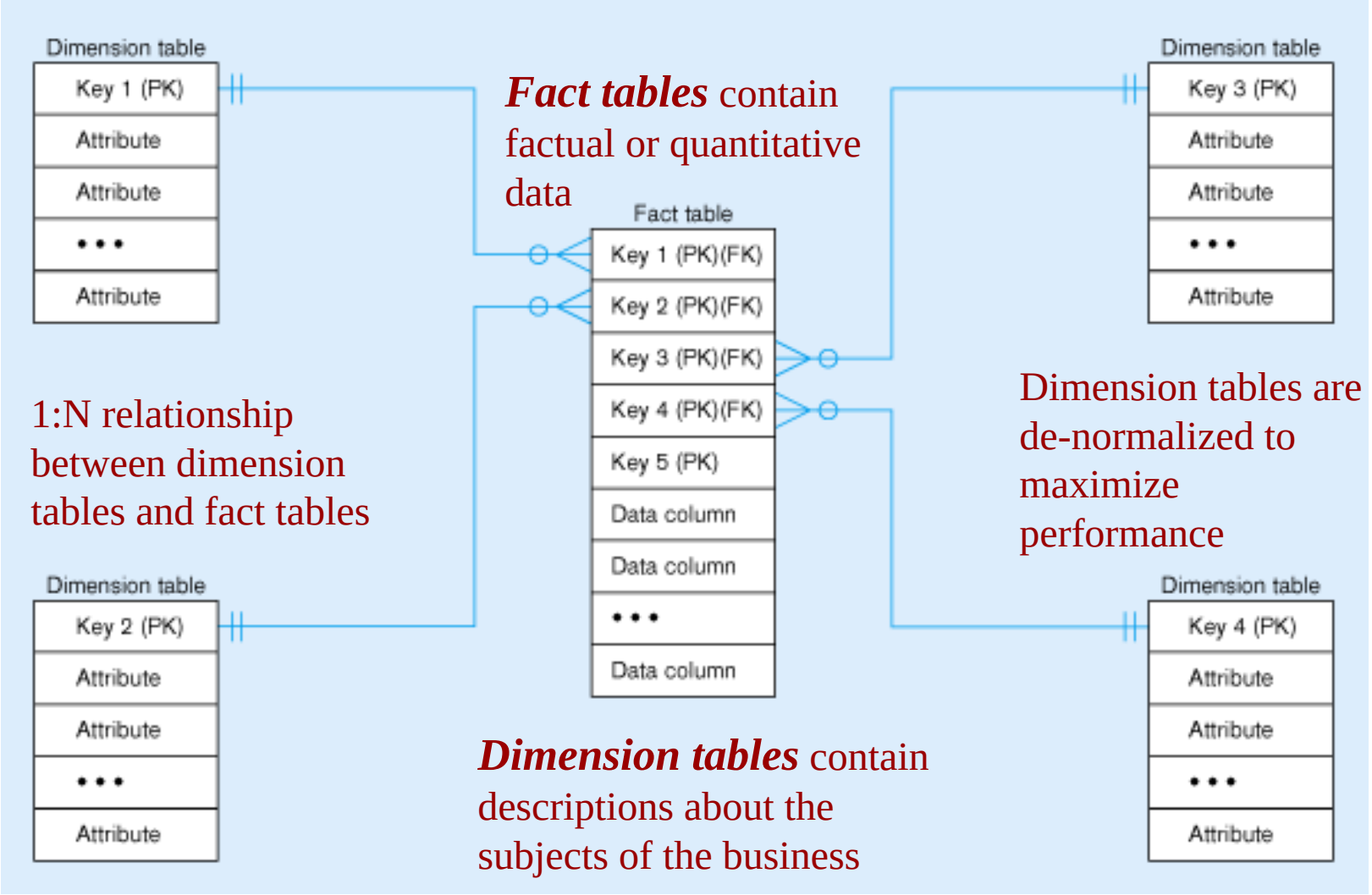
An Aside On The Star Schema (cont.)

- Attributes are often used to search, filter, or classify facts. Dimensions provide descriptive characteristics about the facts through their attributes. The data warehouse designer must define common business attributes that will be used by the data analyst to narrow a search, group information, or describe dimensions.
  - Example: Consider sales. Some possible attributes for the dimensions of sales might be: location, product, and time. These attributes add a business perspective to the sales facts. The data analyst can now group the sales figures for a given product, in a give region, and at a given time.
- The star schema, through its facts and dimensions, can provide the data when needed and in the required format. It can do this without imposing the burden of the additional and unnecessary data (such as order number, purchase order number, status, etc.) that commonly exist in the operational database.

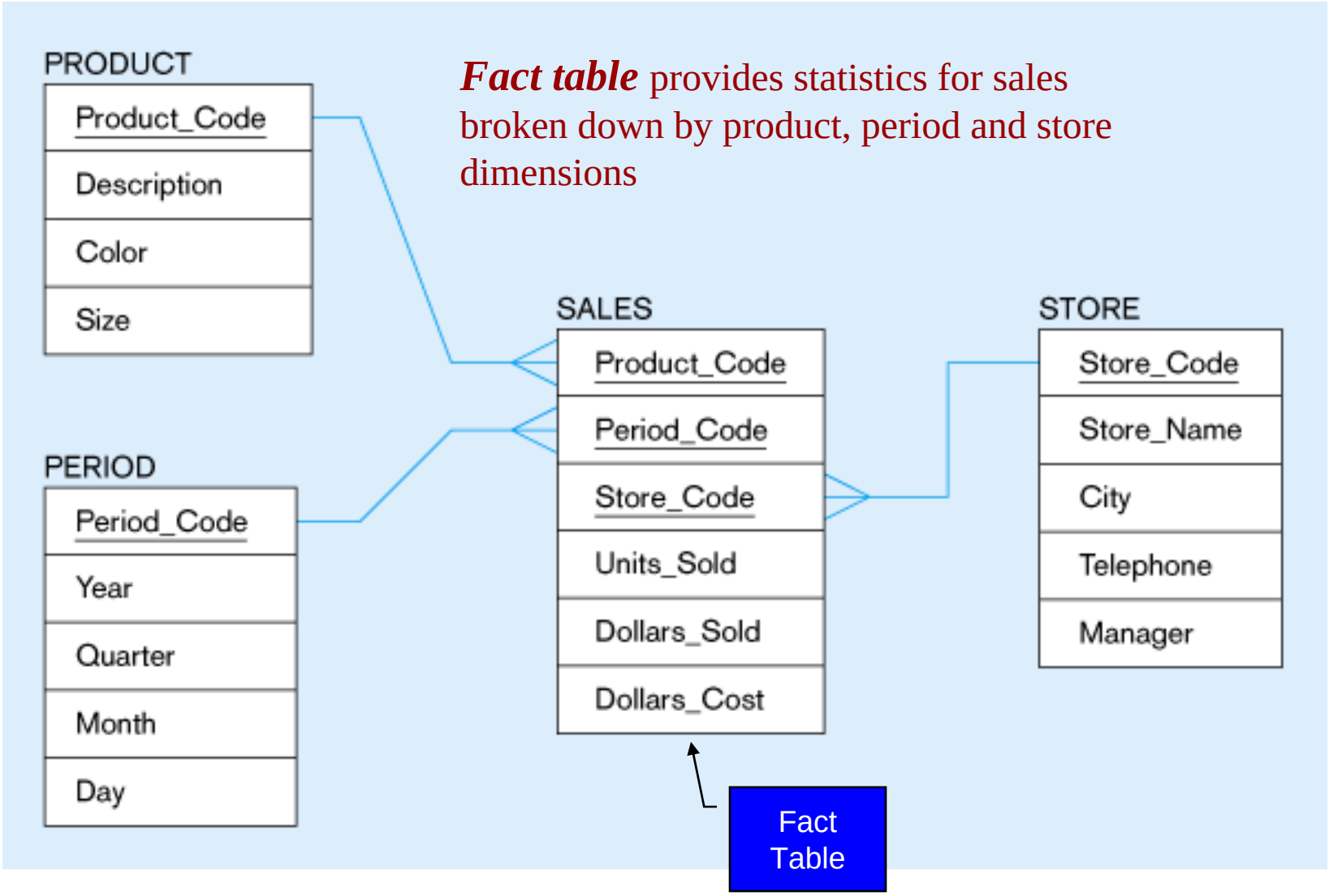
An Aside on the Star Schema (cont.)

- The star schema is a database design which is especially well-suited to ad-hoc queries in which dimensional data (describing how data are commonly aggregated) are separated from fact or event data (describing individual transactions).
- The star schema is not well-suited to on-line transaction processing and therefore is not typically used in operational databases.

An Aside on the Star Schema (cont.)

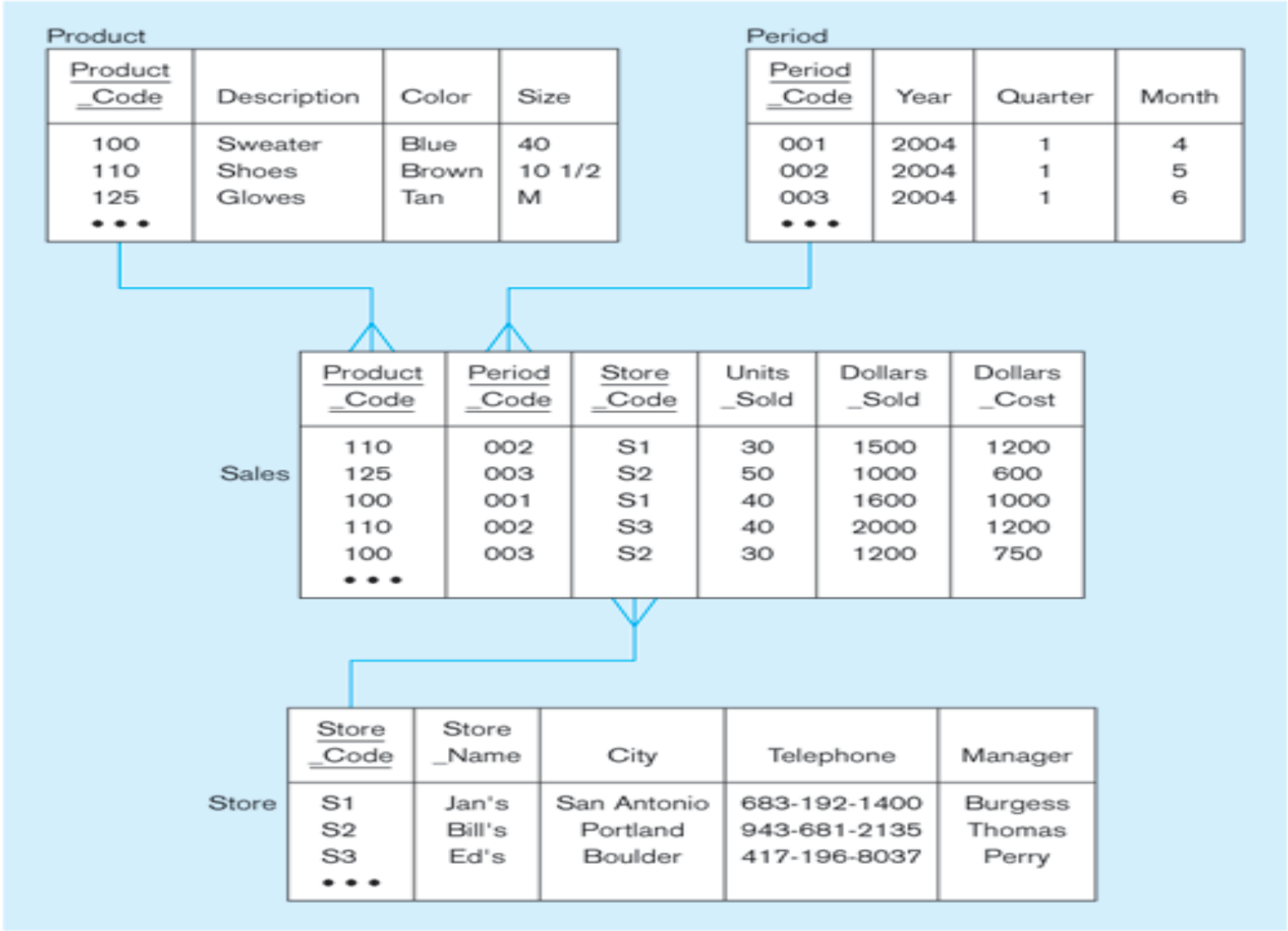


An Aside on the Star Schema (cont.)



Fact table provides statistics for sales broken down by product, period and store dimensions

An Aside on the Star Schema (cont.)



Product

Product_Code	Description	Color	Size
100	Sweater	Blue	40
110	Shoes	Brown	10 1/2
125	Gloves	Tan	M
• • •			

Period

Period_Code	Year	Quarter	Month
001	2004	1	4
002	2004	1	5
003	2004	1	6
• • •			

Sales

Product_Code	Period_Code	Store_Code	Units_Sold	Dollars_Sold	Dollars_Cost
110	002	S1	30	1500	1200
125	003	S2	50	1000	600
100	001	S1	40	1600	1000
110	002	S3	40	2000	1200
100	003	S2	30	1200	750
• • •					

Store

Store_Code	Store_Name	City	Telephone	Manager
S1	Jan's	San Antonio	683-192-1400	Burgess
S2	Bill's	Portland	943-681-2135	Thomas
S3	Ed's	Boulder	417-196-8037	Perry
• • •				

An Aside on the Star Schema (cont.)

- Dimension table keys must be ***surrogate*** (non-intelligent and non-business related), because:
  - Keys may change over time.
  - Length/format consistency.
- Granularity of Fact Table – what level of detail do you want?
  - Transactional grain – finest level.
  - Aggregated grain – more summarized.
  - Finer grain implies a better ***market basket analysis*** capability.
  - Finer grain implies more dimension tables, more rows in fact table.
- Duration of the database – how much history should be kept?
  - Natural duration – 13 months or 5 quarters.
  - Financial institutions may need longer duration.
  - Older data is more difficult to source and cleanse.

Relational OnLine Analytical Processing (ROLAP)

- The star schema is designed to optimize data query operations rather than data update operations. Naturally, changing the data design foundation means that the tools used to access such data will have to change. End users familiar with the traditional relational query tools will discover that these tools will not work efficiently with the star schema.
- ROLAP, however, saves the day by adding support for the star schema to use familiar query tools.
- ROLAP provides advanced data analysis functions, and improves query optimization and data visualization methods.
- Another criticism of RDBMs is that SQL is not suited for performing advanced data analysis. Most of the decision support data requests require the use of multiple-pass SQL queries or multiple nested SQL statements.

Relational OnLine Analytical Processing (ROLAP)

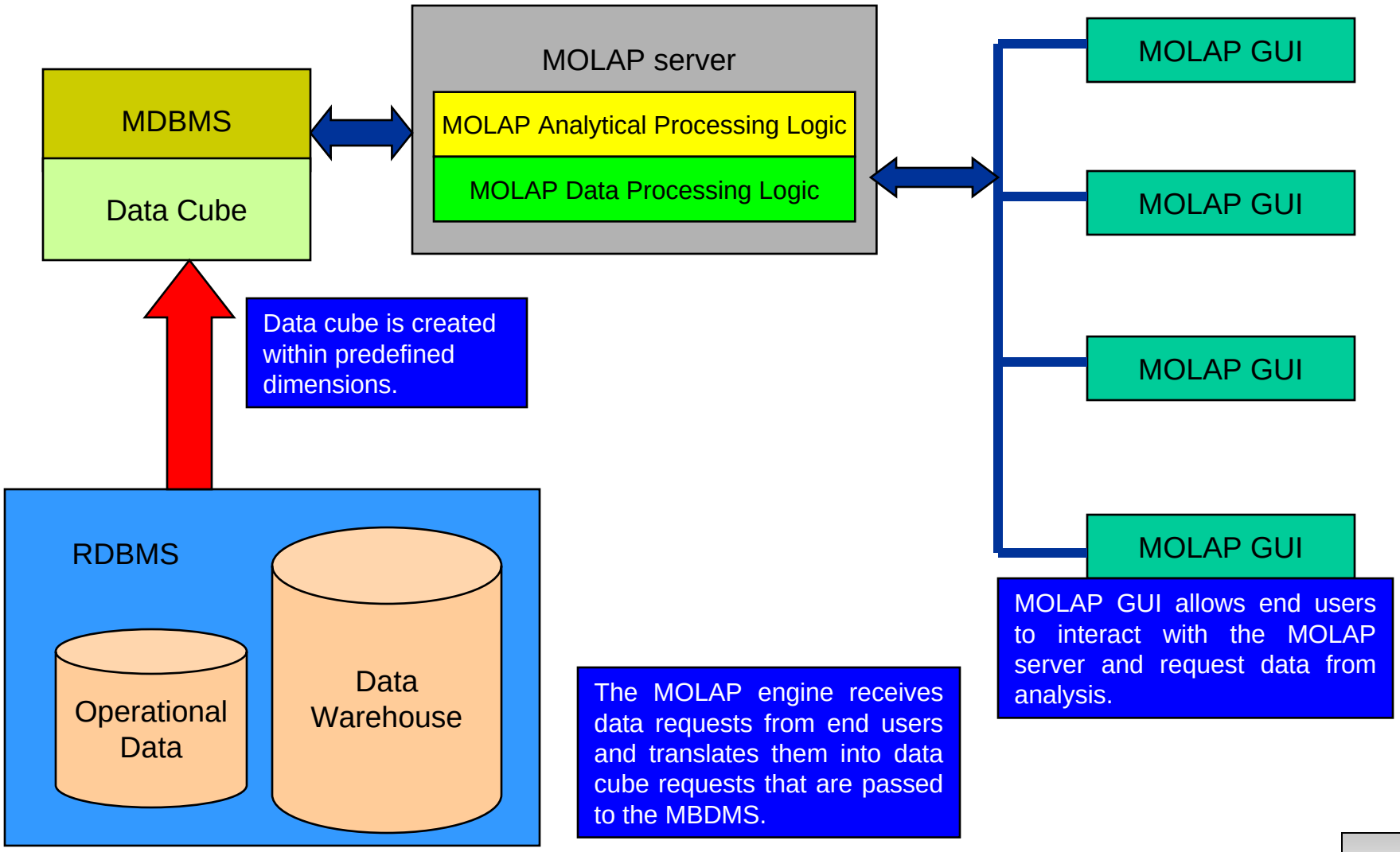
- To answer this criticism, ROLAP extends SQL so that it can differentiate between access requirements for data warehouse data (based on the star schema) and operational data (based on normalized tables). In this fashion, a ROLAP system can properly generate the SQL code required to access the star schema data.

- Query performance is also enhanced because the query optimizer is modified so that it can identify the SQL-code's intended query targets. For example, if the query target is the data warehouse, the optimizer passes the request to the data warehouse. However, if the end user performs drill-down queries against operational data, the query optimizer identifies this operation and properly optimizes the SQL request before passing them through to the operational DBMS.

Multidimensional OnLine Analytical Processing (MOLAP)

- Multidimensional OnLine Analytical Processing (MOLAP) extends OLAP functionality to multidimensional database management systems (MDBMSs).
- An MDBMS typically employs proprietary techniques to store data in matrix-like *n*-dimensional arrays.
- Many of the techniques in MDBMS are derived from CAD/CAM techniques and GIS (Geographic Information Systems).
- Conceptually, MDBMS end users visualize the stored data as a three-dimensional cube known as a data cube. The location of each data value in the data cube is a function of the x, y, and z axes in three-dimensional space.
- The data cubes can grow to *n*-dimensions, thus becoming hypercubes.
- Data cubes are created by extracting data from operational databases or from the data warehouse. An important characteristic of a data cube is that it is static. They are not subject to change and must be created before use. They cannot be created by ad hoc queries.

MOLAP System



Relational vs. Multidimensional OLAP

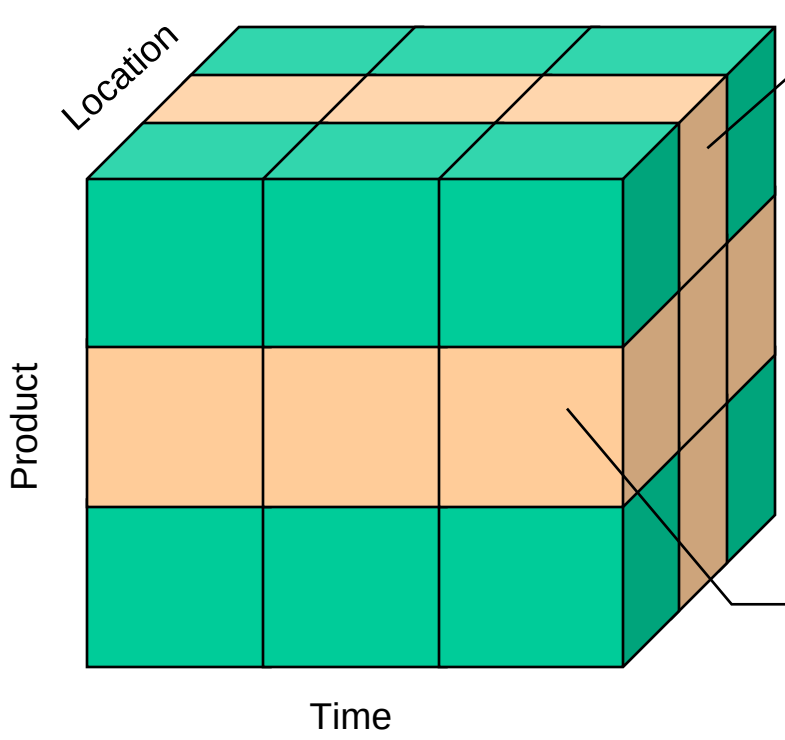
Characteristic	ROLAP	MOLAP
Schema	Uses star schema. Additional dimensions added dynamically	Uses data cubes Additional dimensions require re-creation of the data cube
Database Size	Medium to large	Small to medium
Architecture	Client/server Standards based Open	Client/server Proprietary
Access	Supports ad hoc requests Unlimited dimensions	Limited to pre-defined dimensions
Resources	High	Very high
Flexibility	High	Low
Scalability	High	Low
Speed	Good with small data data sets; average for medium to large data sets	Faster for small to medium data sets; average for large data sets.

Three Dimensional View of Data

Location: possible attributes – region, state, city, store, etc.

Product: possible attributes – product type, id, brand, color, size.

Time: possible attributes – year, quarter, month, week, day, time of day, etc.



Sales manager's view of sales data

Product manager's view of sales data

Slice and Dice Operation