CSCI 4144

Introduction

Gao

Alex Safatli

Tuesday, January 8, 2013

Contents

Introduction

Administration

Bluenose has class notes and assignment handouts located at directory *prof4144*. Other documents may also be located there such as the Syllabus and web links associated with each lecture (for example, note1.1).

A midterm test will be given before the reading week that will cover the first three assignments, and the final exam will cover an additional two assignments. Assignments will be worth 50% (10% each), the midterm will be worth 20%, and the final exam will be worth 30%.

SQL Server 2008 will be used with focus on data analysis. Appropriate packages will be used to create a data warehouse and the information will be retrieved utilizing OLAP. This is what is necessary for the second assignment (DW/OLAP). The remaining three assignments are implementation-based and involve solely data mining.

Two tutorials on SQL Server will be available on Jan. 25 and Jan. 29 from 4:00 to 5:00 pm in Teaching Lab 2.

A General Overview

Data warehousing involves the organization and categorization of data through given models. Many electronic applications make use of databases in order to store and keep track of data. Grouping and summary of results that aid in decision-making is the foundation of data warehousing. Data in general may be stored in these, but knowledge has to be acquired from these warehouses in order to make decisions. These may involve regularities or patterns in the data.

For example, Walmart has a very well-established online data warehousing system in place. All of the information is available for all of the departments involving customer information, inventory, etc. All of the information necessary for decision-making is there.

This sort of information involves small queries that have to be done simultaneously. Relational databases were not enough – they only involved simple groupings and aggregations. These conventional databases do not support the same power that data warehouses are capable of – patterns and regularities being grouped together through models are not part of their specifications. Retail stores, for example, need to know summaries of every day and have reports for every month, etc. Constant summarization has to be done and a different sort of aggregation is necessary in order to measure business performance. Information that may span multiple databases also involve partitioning tasks and is a complicated procedure. Records may split into different databases and therefore summarization may become very difficult and time-consuming.

This is where DW/OLAP technology is useful. Information can be prepared, and are integrated, so reports can be produced quickly and efficiently on summarized data. This is the difference between traditional databases, which are data rich but information poor, and data warehousing, new strategies for Decision Supporting Systems (DSS) or BI (Business Intelligence) systems.

Technologies Are Evolving

Database technologies (relational database, data warehouses, hadoop) and data analysis & information retrieval technologies (SQL, data indexing & pattern matching, statistics, machine learning, data mining) are numerous.

Technologies Are Evolving continued on next page...

Database technologies are associated with *storing* data. Relational databases are mainly used to support operational processing or systems, which are not designed to be optimal for supporting complicated aggregations. **Data analysis & information retrieval technologies** revolves around how to retrieve the data and information within these databases. For example, SQL is intrinsically used for manipulating and retrieving data.

The types of information that business users want is about aggregation. What is happening about the business? Higher-level information used to explain why things are happening, and information that allows one to predict what will happen based on regularities. A **database** is an information system of organizations for *storing/managing* data and *querying information* from the data.

Data types can be retrieved from data warehouses, and they want to be stored because they represent business actions, etc. Once you attach values with metadata, they become information – you do not know what they stand for, otherwise. Questions can be answered about individual objects, but business applications are more concerned with aggregation of objects – total sales of given products, etc.

Questions for review are available on the slide. What is important is to notice that there are three types of information processes desired to have by all organizations. The advantage of data warehouses is that even with a large number of databases, information can be retrieved. This is unlike SQL which does not have this capacity to span multiple databases as such. For example, when accessing our account details for university, grouping and aggregation is not necessarily required. However, for the dean, this is the case. Decision-making is required.

> **NOTE** Remember that JOINing tables requires the most effort and has the greatest cost in a relational database. Therefore, because aggregation necessarily requires looking at multiple tables, a new sort of data model is necessary. This is where data warehousing comes in. This is the limitation of conventional information systems (DBMS technologies) in terms of information queries.

The types of information necessary by organizations are as follows:

1. *Information about individual objects (or records)*. What, when, where, etc. Search for existing value(s) of records; no calculations involved. Operational.
2. *Information about aggregations*. What happened to the business. Define groupings and apply simple statistics functions. Can be single-attribute based, or combinations of attributes.
3. *Information about patterns.* Why it happened, and what to happen next. Patterns are regularities or knowledge of a given data set. Discover hidden patterns about a concept (or called target), or relationships, or abstractive representation of categories. May involve multiple aggregations.

An example on the slides given is a relational DB for an electronics retail business, such as Futureshop. Different types of objects are stored, and each table possesses the scheme indicated on the slide by metadata. The queries given on slide 9 of *note1.1.pdf* can be classified as follows:

1. Type 1: Operational
2. Type 1: Operational
3. Type 2: Aggregation
4. Type 2: Aggregation
5. Type 3: Pattern
6. Type 3: Pattern

Decision Support Systems (DSS) (vs. Information Process) have three different types of information directly associated to the above types:

1. *Online Transactional (information) Processes (OLTP)*: Operational databases; relational DB and SQL. Track/record/retrieve original data records of everyday business operations.
2. *Online Analytical (information) Processes (OLAP)*: data warehouses and OLAP. Store and manipulate summaries of various groupings of original data records for answering what happened to the business.
3. *Knowledge discovery from data*: Data mining. Discover/analyze hidden patterns of abstractive information for answering what will happen next.

NOTE Note that **data** is raw measures or unprocessed facts, **information** has *metadata*, and **knowledge** involves regularities and patterns.

A view of DSS (Business Intelligence) is available on slide 21 as a triangle hierarchy. With greater complexity there is an increased potential to support business decisions.

More About Data and Information

The differences between *data*, *information*, and *knowledge* can be expressed with examples.

Data is what is expressed in operational databases – symbols, values, measures, and effects. Columns in these databases will express features or attributes and each cell contains data. An example is the number 41.44.

Attaching metadata, or context, to data enables it to become **information**. Defining the measure enables us to see it as something beyond merely a measure. You can answer questions about it. An example is the price $41.44. With a defined *schema*, a set of attributes across one or more tables, the types of objects are defined and thus you can store a collection of data, and consequently information (they have metadata).

Information can be divided into **low-level** and **aggregation-level**. Low-level information is about operational databases. They are simply stored in the database and not processed. Aggregation-level information involves *grouping* of information and attributes. Low-level information typically involves itself with one single object and can be retrieved by going to its location; operational information. On the other hand, *aggregated information* involves statistical functions (such as total, count) and are calculated. Data warehouses is a new type of database model meant to store this sort of information.

Information is measures, descriptions of these measures, and could involve groupings of them. **Knowledge** *summarizes* higher-level information and is established by analyzing information. It is general laws, rules, or patterns generalized from a large group of factual information. It looks at the relationships among other attributes. It is used to explain the nature of the information and can be used to predict. An example of this is determining that it is hot outside by looking at the temperature, or the risk with a given subject. Knowledge acquisition can be done through machine learning and data mining.

A *knowledge-based system* does automatic problem solving by applying built-in knowledge to the sensed environment (i.e. input data). Examples of this include robot navigation, engine diagnostic systems, and bank load approval systems. A *business intelligent system*, for example, uses built-in knowledge to perform problem solving. This could be an automatic risk detection classification system and makes inferences.

NOTE Queries can be categorized and link to each of these concepts. See last section.

More About Data and Information continued on next page...

For a retail store like Walmart, they would like to separate queries by the nature of individual customers. Technology is necessary to support this. If aggregation is necessary, data warehousing will be needed to support this. The simple rules for choosing solution tools for getting different types of business information thus involves knowing what sort of information you need.

NOTE Slide 31 on *note1.2.pdf* has a list of SQL questions.

The conventional information system (DBMS) has limitations. Using a relational DB model involves a structure of tables and links. A very strong mathematical basis is present here. Operations involve basic relational algebra operators: SELECTion (retrieving rows), PROJECTion (retrieving columns), UNION (combination of tuples from different tables that are *union-compatible*), INTERSECTION (*union-compatible*), DIFFERENCE (*union-compatible*), and JOIN. However, when tables get large, calculations begin to take time. JOIN is expensive. Furthermore, information across multiple databases cannot be queried using SQL here.

In a DBMS, information is queried in SQL (structured query language). Each query is described by an *SQL statement* which specifies the sequence of the relational algebra operations for retrieving a defined information in terms of what has been stored. Overall, a **DBMS is for operation databases**.

DBMSs are **transaction-based** and often designed using the *relational database model*. Such a database will contain several normalized tables. The objectives are to reduce redundancy and promote *quick access to individual records*. It is not efficient for dynamically changed grouping operations from large data sets, in particular if the queried information is stored in different tales. It is, also, difficult to use SQL to define complex queries. Analyzing data and exploring relationships are not part of the vocabulary in SQL. Overall, it is constrained to retrieve information from a single database.

A **data warehouse** is a historical database designed for decision support rather than transaction processes. DW is subject-oriented (data are organized around a business focus), and the data stored is aggregated information data. OLAP tools are used to quickly generate reports for answering business ad hoc queries, etc.

When it comes down to it, there are three simple rules when it comes to acquiring information or knowledge:

1. If you know exactly what you are looking for, the information you are looking for is *explicitly stored*, or you want some simple grouping and aggregations, use **SQL**.
2. If you are dealing with business decision support information (subject-oriented) which may involve *complex groupings and aggregations* (or historical information), you need **DW/OLAP** solutions.
3. If you are looking for high-level information (*knowledge*) which may provide explanations for current behaviour, or to predict future data, turn to **Data Mining**. It is the process of finding *unknown, valid, and actionable patterns* from large data.

The merit of data warehousing is that it provides "...a single repository for [a] completely integrated, 360-degree view of your business – one version of the truth."

Knowledge Discovery Process

The steps involved in the **knowledge discovery** process include:

1. *Understand the application domain*. Relevant prior knowledge and goals of application.
2. *Create a target data set.* Data selection.
3. *Data cleaning.*

4. *Data reduction and transformation.* Find useful features, dimensionality/variable reduction, etc.
5. *Choosing functions of data mining.* Classification, regression, association, clustering, summarization, etc.
6. *Choosing the mining algorithm(s).*
7. *Data mining.* Induction and search for patterns of interest. Core part of the knowledge discovery process.
8. *Pattern evaluation and knowledge presentation.* Visualization, transformation, removing redundant patterns, etc.
9. *Use of discovered knowledge.*

What are the basic knowledge discovery tasks? There are four basic tasks: **classification**, finding the targeted concept model, finding **association** rules, **clustering** objects into groups, and **summarization** or characterization.

Classification involves having a **target concept** or business concept in mind and predicting values for this concept — determine the model. A classifier is determined here and it is used as a tool for determining knowledge. An example here is determining the method for assessing risk. This has many applications.

Association rule mining is a very important part of data mining. It is used to represent statistical importance of togetherness of different items. Finding association rules has an analogy: *a shopping basket*. When each customer goes to the checkout, they will have a basket of items purchased. Many customers will checkout. There will be some regularities of what items are purchased most likely together. Essentially, this means you use some items, or their grouping, to predict other items and their layout. The rule is defined as $X \Rightarrow Y$ where either the *sup* or *conf* can be evaluated (e.g. $sup(X \Rightarrow Y)$) to get either the **support** or **confidence rate** respectively.

Clustering involves looking for high similarities within a data set. *Finding groups of objects such that the objects in a group are similar (or related) to one another and different from (or unrelated to) the objects in other groups.* In clusters, intra-cluster distances are minimized and inter-cluster distances are maximized.

Summarization involves how to summarize data quickly. Looking for problem areas, bigger items of impact, etc. in each cluster; this is very involved.

Data Mining Functions

Or **task areas**. Data mining can be split into two types of tasks: (1) **predictive** and (2) **descriptive**. Furthermore, predictive tasks include:

- classification,
- regression (the result of fitting a set of data points with a [quadratic] function),
- time series analysis,
- and prediction.

Descriptive tasks include:

- clustering,
- summarization,
- association rules,

Data Mining Functions continued on next page...

- and sequence discovery.

Traditional statistical methods are similar to data mining. However, there are key differences. First of all, DM is a hypothesis-free approach: it is relatively easy to gain new insight. DM also makes fewer assumptions or no assumptions at all. Finally, the data input to DM can be any data type. It is fair to say that statistics traditionally has been used for many of the analyses now done with data mining (such as building predictive models or discovering associations in databases).

Remember that data mining is different than **information retrieval** (IR), which is to retrieve desired information from an information store: such as a database, or the Internet. SQL is the conventional tool for retrieving information from a relational database. The Internet is a new type of database dominated by unstructured textual data which require new information retrieval technology — a search engine.

Similarity (used in clustering), precious and recall measures (for describing accuracy of predictive modelling techniques) from IR have a great impact on the development of data mining. DM supports for establishing indexes for IR (text clustering for categorization).

Data mining also incorporates a great deal of technologies including **machine learning**. ML focuses on learning methodology, but DM is an application and extension of ML for discovering knowledge from a large data set. *What is learning*? The operational definition is that it involves a certain task to carried out either well or badly, and a subject that is to carry out the task. This would he *how to determine when someone has learned something*. People learn how to carry out a task by making a transition from a situation the person could not do it to one where they can. The process of this transition is called learning or training. ML methods are developed to obtain knowledge automatically from data for carrying on new unknown tasks. This has a strong relationship with the methodology of science (they both share the process of knowledge discovery).

NOTE Recall that the two general purposes of data mining results are for: (1) explanation and (2) prediction of upcoming events. Ultimately, these both encompass the more general purpose: **problem-solving**.

The Empirical Cycle Model (ECM)

What is the Empirical Cycle Model (ECM) of scientific research? It is a very important model: all scientific study features and use this model. It encompasses four steps: (1) observation, (2) analysis, (3) theory, and (4) prediction.

In the model, you start with a number of **observations**, you try to find patterns in the observations (**analysis**), formulate a **theory** for explaining the data, and **predict** new phenomena that can be potentially verified. This final step has two possibilities:

- Our predictions are correct; theory is correct.
- The predictions are wrong.

If it is the second, the new observations must be analyzed and changes must be made to the theory; or a new one must be made. The whole process starts again. Note, though, that while we may make hypotheses to explain observations, they can never be proved true.

How are machine learning and data mining associated with the ECM? Both are a part of the scientific discovery (knowledge discovery) process. This general model applies to all scientific research regarding finding knowledge or general properties. In extension, *a discovered knowledge only has temporary value*: once new data is observed, the cycle process repeats and the original hypothesis is put into question.

The Empirical Cycle Model (ECM) continued on next page...

Why does discovered knowledge need to be corroborated by statistics? The notion of statistics enables us to support our hypotheses and allows us to validate them.

What are the main differences between IR, statistics analysis, and DM? Data mining does not necessarily have a hypothesis, whereas this is necessary for statistics analysis. The data collected is so large, the conventional statistical methods that can be used are limited. IR looks for existing facts while DM looks for summarized knowledge.

Data Warehouses and OLAP

Note that before any task of data mining, data warehousing, etc. **data-preprocessing** has to occur first. Optimal schemas have to be developed, data has to be cleaned, etc.

Objectives

Note that the data stored in data warehouses are aggregated; put into groups of data. They are already sorted. OLAP operations are really intended for quick response times. When investigating DSS, data warehouses and OLAP comprises two tiers: **data exploration** (OLAP), and **data preprocessing/integration, data warehouses**. Below these two tiers is the origin of the data: OLTP — this comprises a great deal of hardware. They are designed to be optimal for supporting a great deal of users. If the queries are small, SQL is the best.

> **NOTE** **DW Enablers** refer to the technology. **DW Drivers** are the motivations of providing answers to various ad hoc queries of large organizations.

How do you quickly provide answers to various ad hoc queries of large organizations? Note there is a problem here: conventional DB solutions are data rich but information poor. Corporate data assets are increasingly dispersed among hundreds or even thousands of different platforms throughout the enterprise.

Distributed DBMS (DDBMS) solutions fail to achieve a single enterprise-wide data management layer which could provide various types of transparency (transparencies of location, platform, and data formats) and treat these as if they were a single logically centralized, and homogeneous database.

The DBMS technology is mainly designed to handle day-to-day based operations which have very limited power of retrieving decision-oriented information. It possesses noise and redundant data, lack of integration, structure limitations of viewing at multiple levels, and is difficult for discovering hidden knowledge. We need analytical information: statistical summaries (information data) of various groupings about business subjects.

There are two approaches:

- Database query-driven (lazy).
- Data Warehouse/OLAP (eager).

The **database query-driven** approach involves a **mediator** which, going through wrappers to various sources, provides the results of queries to clients. **Warehouse architecture** involves a **warehouse** which, paired with metadata, provide results of queries and analyses to clients through not wrappers but **integration** with sources.

The objectives of data warehouses is that we want:

Objectives continued on next page...

- quick analytical information on ad hoc business queries,
- analytical information: statistical summaries of various groupings about business subjects,
- information data: consolidated, cleansed, staged, ready for use,
- efficient management on enterprise-wide analytical data: single centralized and homogeneous database,
- focus on the usage of information/analytical side: quickly generate aggregation-based reports, analyze trends, etc.

NOTE Assignment #2 is due February 5; a tutorial will be held tomorrow from 4:00 to 5:00pm and on January 29 from 4:00 to 5:00pm.

Why is it said that businesses are the drivers of DW/OLAP *technology*? The need to make decisions based on data reports, etc. drives a way to aggregate such data. *There are two general approaches for integrating information different data sources for a large enterprise; can you describe them*? A lazy method by database queries, or an eager method using DW/OLAP. Relational databases remain the best for online transactional operations, acquired very quickly.

Point out the general properties of DW, and state how they fit the main DW's objectives. See above.

Enterprise DW & Data Marts

There are two types of DW:

- Enterprise DW: Contains subject-oriented data spanning the entire organization. Provides corporate-wide data integration.
- Data Mart: Contains a single subset of department-wide data.

DW/OLAP provides a good solution to DSS in that *an ad hoc query can be translated into a series of* OLAP *operations* for forming a meaningful answer. For example: "Why are the sales for this year not meeting the targets?" This question may be translated into a sequence of queries for factual information, such as:

1. For each product or each category of products, what are the cumulative sales for the year?
2. Identify those products for which actual sales are less than the targets?
3. Where are the turning points for those sales which do not match the targets?

When put together, this information can allow one to answer the original question. In general, there is a linear increase in demand and payoff of new OLAP technologies by businesses.

Defining a Data Warehouse

A **data warehouse** can be defined in many different ways, but not rigorously: they are *decision support databases* that are maintained *separately from the organization's operational database*. They typically *support aggregated information processing* by providing a solid platform of consolidated, historical data for analysis.

Ultimately, a **data warehouse** is a **subject-oriented, integrated, time-variant**, and **nonvolatile** collection of **information data** in support of management's decision-making process.

Subject-oriented drives home the point that business data is reorganized around its major subjects: for example, "sales", "supplies" and "customers" for retailer stores or "grades" for CS student data analyses. This excludes data not relevant to the decision-making process.

Integrated refers to the fact that a DW is constructed by integrating multiple, heterogeneous data sources: relational databases, flat files, online transaction records, etc. Data cleaning, transformation, integration, and techniques are applied to: ensure accurate data with consistency in conventions, encoding structures, attribute measures, etc. among data sources, and to convert data as it is moved to the warehouse.

Time variant refers to the fact that the time horizon for the data warehouse is significantly larger than that of operational systems: they provide information from a historical perspective (past years) rather than current values (e.g. in an operational database). Factual elements would contain time elements (explicitly or implicitly).

Non-volatile refers to how a DW is physically separated from any operational databases. Does not require transaction processing, recovery, concurrency mechanisms.

How do we get business analytical information easily? Store them in a **multi-dimensional space** (MDS) and use MDS manipulators (**OLAP operators**) to retrieve/view them in real-time. Multi-dimensional software, unlike spreadsheets or SQL databases, is specifically designed to facilitate the definition and computation of sophisticated multi-level aggregations and analyses via OLAP operations.

Note that OLAP applications are dominated by ad hoc, complex queries. In SQL terms, these are queries that involve grouped-by and aggregation operations which are poorly handled in most DBMSs. The natural way to think about OLAP queries is in terms of a **multi-dimensional data model** (MDM). This is a data model using **logical dimensions** to define an information space of business events.

This **logical space** is also known as a **hypercube** (data cube). Each dimension of the cube represents an aspect of the possible business events divided into discrete values representing attribute domains of the dimension.

In data warehouse literature, a *n-D* base cube is called a **base cuboid**. The topmost 0-D cuboid, which holds the highest-level of summarization, is called the **apex cuboid**. The lattice of cuboids forms a data cube. *How many cuboids are in a data cube of n-dimensions*? 2^n. A data cube is a structured space of cuboids.

For representing summarizations at different levels of the same dimension, we can also talk about **concept hierarchies**. See slide. The number of dimensions is based on the query: how general the aggregation is that is being looked for. For each dimension, you have a concept level and summarization level.

The total number of cuboids of a data cube with n-dimensions and concept hierarchies is defined by T:

$$T = \Pi_{i=1}^{n}(L_i + 1)$$

What is multi-dimensional space (MDS) for DW, and how is it described by the data cube lattice? It allows information to be easier to view – it provides structure and ensures it is not chaotic. It provides convenience of addressing and locating given information. Lattices provide dimensionality to the structure: in a way, looking at how rooms are organized – features, descriptors, on the information wanted to be retrieved.

What are "cuboids" in a MDS space? How do you estimate the total cuboids contained by a DW space? These cuboids represent, in a way, rooms for information to be held in. This is related to the above question. The number of dimensions (n) and concept hierarchies/levels in each of these cuboids can give us the exact number of cuboids of a data cube. See the above formula.

How different does a DW model compare with a conventional database model? Many differences exist and have been outlined before: OLAP *v.* OLTP, cuboids, aggregation, capacity for prediction; a collection of all fact measures into aggregations (cuboids).

Database Models: DW Schemas

The basic structure of a data warehouse is comprised of *dimensions* and *measures*.

- A **star schema** refers to a *fact table* (collection of all cuboids) in the middle connected to a set of *dimension tables* (used to define each particular dimension — dimension name, concept levels).
- A **snowflake schema** refers to a refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to a snowflake. Removes redundancy of information.
- **Fact constellations** are comprised of *multiple* fact tables (multiple data cubes) that share dimension tables, viewed as a collection of stars. These are also known as *galaxy schemas*. Can be used at the enterprise level for very complicated warehouses: multiple sub-spaces and subjects.

Star schema is the most commonly used schema for OLAP applications and used as a data mart for department-level data warehouses. It is simple *but some* ***redundancy*** *may occur*. See slides for an examples of all three types (*note3.3.pdf*). Note that much of the data amongst rooms, etc. may not fully materialize because all combinations may not be practically possible.

OLAP Query Language

Data warehouse query language involves the definition of cubes and the definitions of dimensions. See slides. OLAP operations include:

1. **Roll up** (drill-up): summarize (aggregate) data; by climbing up the hierarchy or by dimension reduction.
2. **Drill down** (roll down): reverse of roll-up; from higher-level summary to lower-level summary or detailed data, or introducing new dimensions.
3. **Slice and dice**: project and select.
4. **Pivot** (rotate): reorient the cube, visualization, 3D to series of 2D planes.
5. Other operations include:
    - **Drill across**: involving (across) more than one fact table.
    - **Drill through**: through the bottom level of the cube to its back-end relational tables (using SQL).

An illustration can be made for a **star-net query model** which is a way to represent OLAP queries. Each axis indicate dimension tables and each circle is a level of abstraction. Data cubes are tools in order to answer complex queries.

What is the difference between logic dimensional space (MDS) and the conventional geometry (or physical) space? Cuboids are the notions of rooms or cells and can be at different levels of concepts and comprise the MDS (are sub-spaces). Physical conventional geometry is also constricted to 3-dimensional space. Furthermore, there are a great deal number of rooms (there is a formula to calculate how many).

OLAP Query Language continued on next page...

As distinguished from physical dimensions, which are based on angles and limited to tree, logical dimensions have no such limits. There are two types of dimensions of a data cube:

- **Identifier dimensions**: logical factors or identifying attributes of measurable events or things that we track.
- **Variable dimension**: identify what we track in a situation.

MDS software enables multi-dimensions of information to be combined onto each row, column, and page axis of a screen, thus making it possible to visualize and understand the data set in terms of information present on a flat screen. The ability of MDS software to model multidimensional information and to handle the user representation of the information makes it better suited for working with complex datasets than either SQL databases or traditional spreadsheets.

How is it one can define OLAP queries? Recall the star-net query model. The concept of various dimensions and the concepts associated with various levels allow one to zero in using OLAP tools to get a result.

Presenting Cuboids on a 2-Dimensional Screen

What is the visualization metaphor for displaying OLAP results of multiple logic space (data cube), and how do you best use it? Look at the slides from last lecture (*note3.3.pdf*). Once the information is obtained, it has to be presented. The limitation here is you typically only have a 2-dimensional screen. How do you display greater than 3 dimensions? You can still use 2 dimensions, but you have to follow the idea of 3 dimensions: a **column**, a **row**, and **page**.

How does one map multiple logical dimensions onto a single computer screen or image? There are different metaphors for this:

- Physical dimension metaphor (for 3D graphics): *virtual camera*.
- Logic dimension metaphor (for table-based report): *analytical screen*.

For example, let's say you have a six-dimensional MDS (a data warehouse). These dimensions are chosen because of the information needed for this particular application. To combine multiple logical dimensions within the same dimensions, the row, column, and page (as above) are employed as analytical screen dimensions — each dimension of the vertical bar can be connected to either a row, column, or page axes.

There are different ways that the same model dimensions can be mapped onto row, column, and page axes. The ability to easily view data by reconfiguring how dimensions are displayed is one of the great benefits of MDS – separation of data structure as represented in vertical logical dimensions from data display as represented in the multi-dimensional grid.

Issues present of using the analytical screen include:

- The more screen space is consumed for displaying dimension members, the less space is left for displaying data.
- The less space left for displaying data, the more scrolling you need to do between screens to see the same data.
- The more scrolling you need to perform, the harder it is to understand what you are looking for.

Therefore, *make optimal use* of the analytical screen. To maximize the degree to which everything on the screen is relevant, try keeping dimensions along pages unless you know you need to see more than one member at a time. Ask yourself "What do I want to look at?" or "What am I trying to compare?" before deciding how to display information on the screen.

Summarization: Aggregation Measures

What do I want to look at? What am I trying to compare? We can define a grouping (determine a cuboid of the data cube) and choose measures about the cuboid (for pre-calculated values, or invoke online aggregate functions to the grouping). An OLAP query can be defined as a cuboid-value pair (or dimension-value pair).

For example: "What is the total sales of computers in Vancouver for the first quarter?" The cuboid here is:

```
<time = "Q1", location = "Vancouver", item = "Computer">
```

And the value is:

```
sales = sum(the data set of the cuboid)
```

Aggregation functions are statistical models of data summarization and include: SUM, COUNT, AVERAGE, MAXIMUM, MINIMUM, VARIANCE, STANDARD DEVIATION, MEDIAN, MODE, RANK, etc. Higher-level summarization does not need to be pre-calculated but can be done online – very easy to calculate and done real-time. These functions are not equally valued in terms of computation necessary.

Categories of aggregate functions include the following. Distributive (particularly) and algebraic functions are the most frequently used and can be calculated efficiently (online). Holistic functions are not done online.

- **Distributive**: if the result derived by applying function to n aggregate values is the same as that derived by applying the function on all the data without partitioning (e.g., $count()$, $sum()$, $min()$, $max()$, ...).
- **Algebraic**: if it can be computed by an algebraic function with the arguments which are obtained by applying distributive aggregate functions (e.g., $avg() = sum()/count()$, $variance()$, $standard_deviation()$, ...).
- **Holistic**: if it needs repeated search and comparison on the selected data set (e.g., $rank()$, $median()$, ...).

OLAP Server Architectures

There are three architectures of implementation:

1. **Multidimensional OLAP** (MOLAP): implemented as a large multidimensional array; fast indexing to pre-computed summarized data (built-in indexing), and a fully materialized MOLAP array can contain an enormous number of *empty cells* — results in *unacceptable* storage requirements.
2. **Relational OLAP** (ROLAP): implemented as a collection of relational tables; can be processed and queries with traditional RDBMS technology (indexing, grouping, JOIN), has greater scalability, but no built-in indexing.
3. **Hybrid OLAP** (HOLAP): has user flexibility depending on what level, e.g. low-level: relational, high-level: array. This is what is used by MS SQL Server 2008.

See slide on multi-tiered architecture. There are three kinds of data warehouse applications:

1. *Directed Information Processing*: deals directly with stored aggregate information; supports simple ad hoc queries, basic statistical analysis, and reporting using crosstabs, tables, charts, graphs.
2. *Analytical Processing*: supports more sophisticated ad hoc queries basic on a set of OLAP operations.

OLAP Server Architectures continued on next page...

3. *Data Mining*: Support user interactive exploration for finding hidden patterns, find hidden patterns from a defined data set in DW; supports associations, constructing analytical models, performing classification and prediction, and presenting mining results using visualization tools.

Data Preprocessing

A Major Part of DW and Knowledge Discovery

Why is data preprocessing important? The typical tasks of data preprocessing will be investigated, a case study will be looked at, and DP examples will be followed.

Data preprocessing occurs in the very first parts of the multi-tiered architecture of an OLAP server: when data is retrieved from other sources and operational databases and the data is extracted, transformed, loaded, and refreshed. The more accurate and quality of the data, the better the results of data mining – in terms of effort that is necessary for each data mining process step, the most effort should be allocated to the data preparation step.

In the main phases of Knowledge Discovery (KDD), transforming, data cleaning, data integration, and selection from databases to the data warehouse to task-ready data comprises data preprocessing. Again, this comprises the very beginning of the process – retrieving the original data.

Why preprocess data? If you do not possess quality data, you do not have a chance to retrieve quality DW/DM results. Noise and irrelevant data will make the process more complicated. Quality decisions must be based on quality analytical information and knowledge which can only be derived from quality data.

What properties should quality data have? They should be:

- relevant,
- clean,
- consistent.
- enriched (integrated),
- and in the right format and type.

Major tasks can include *data cleaning*, *data integration*, *data transformation* (can include normalization or scaling), and *data reduction*. Data in the real world is *dirty*: it can be...

- *incomplete*: lacking attribute values, lacking certain attribtues of interest, or containing only aggregate data,
- *noisy*: containing errors or outliers,
- *inconsistent*: containing discrepancies in codes or names.

Data type and form must be ready; for example, text words are transformed into numerical weights for clustering mining or age values are discretized into categorical labels.

Data cleaning or *data scrubbing* is the act of detecting and correcting (or removing) corrupt or inaccurate data records from a data set, table, or database. This may include finding and filling in missing values, identifying and correcting errors, finding and removing duplications, identifying and removing outliers, and finding and resolving inconsistencies.

Data transformation involves transforming data types and form into appropriate ones for the task. For instance, this could include normalization which is necessary if you want to compare different attributes equally. **Data normalization** is involves scaling all attributes to the *same order of magnitude* so we can obtain reliable distance measures between different records.

Data integration means you integrate the data of multiple data sets, databases, data cubes, or other data files into one location or accessible form.

Data reduction is the process of reducing data size and making the prepared data *more relevant* – this includes choosing proper subsets from the original data, removing irrelevant attributes and object records, etc.

Give three reasons why data needs to be processed for DW and DM. Reasons can include: they are from different sources and must be integrated into one, data could be incomplete, and may need to be normalized in order to scale all attributes equally.

What properties should quality data have? They should be relevant, clean, consistent, enriched (integrated), and in the right format and type.

What are the typical data pre-processing tasks? Data cleaning, data transformation (normalization), data integration, and data reduction.

A Case Study: Data Preparation for Customer Profiles Discovery

The business background here involves a publishing company that sells magazines on cars, houses, sports, music, and comics. Typical information queries include:

- *What is the typical profile of a reader of a car magazine*?
- *Is there any correlation between an interest in cars and interest in comics*?

The business DSS model is to find clusters of clients and the profiles in order to setup a marketing exercise. The data mining task is to mine clusters of clients and deal with association analysis. To prepare the data for clusters, the above tasks have to be done (data cleaning, data integration, data reduction, data transformation, etc.).

The company may have multiple data sources but the relevant databases may include: the subscription invoice database and the customer database. The relevant data needs to be determined including the tables and attributes. De-duplication algorithms using pattern analysis techniques could identify situations of duplication or inconsistent data and present it to a user to make a decision.