ANALYZING AND DEVELOPING TECHNIQUE FOR MINING VERY LARGE DATABASES TO SUPPORT KNOWLEDGE EXPLORATION

A THESIS SUBMITTED TO

SAURASHTRA UNIVERSITY, RAJKOT

FOR THE AWARD OF

DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE

IN THE FACULTY OF SCIENCE

SUBMITTED BY

THUMAR SATISH GOBARBHAI

DEPARTMENT OF COMPUTER SCIENCE

SAURASHTRA UNIVERSITY, RAJKOT

UNDER THE GUIDANCE OF

DR. N. N. JANI

PROF. & HEAD

DEPARTMENT OF COMPUTER SCIENCE

SAURATHRA UNIVERSITY, RAJKOT

March 2006

CERTIFICATE

I hereby certify that **Mr. Thumar Satish Gobarbhai** has completed his thesis for doctorate degree entitled **"ANALYZING AND DEVELOPING TECHNIQUE FOR MINING VERY LARGE DATABASES TO SUPPORT KNOWLEDGE EXPLORATION".** I further certify that the research work done by him is of his own and original and is carried out under my guidance and supervision. For the thesis that he is submitting, he has not been conferred any degree, diploma or distinction by either the Saurashtra University or any other University according to best of my knowledge.

Place: Rajkot
Date:

Dr. N.N. Jani
Prof. & Head
Department of Computer Science
Saurashtra University, Rajkot.

CERTIFICATE

I certify that the developed model for Data Warehousing, Data Mining, Web Mining, Text Mining and results derived by analysis and described in the thesis has been based on the literature survey, bibliographical references and through study of the web sites in respect of related areas.

Apart from these, all the analysis, hypothesis, inferences and interpretation of data and strategy have been my own and original creation. The model has been prototyped to a domain, which is my own and original creation. Moreover, I declare that the work done in the thesis, either the Saurashtra University or any other university has not conferred any degree, diploma or distinction on me before.

Place: Rajkot
Date:

Thumar Satish Gobarbhai

"IT SHOULD NEVER FORGET
THAT TECHNOLOGY MEANS TO AN END
AND NOT AN END IN ITSELF"

E.F.Codd

To my father, my mother and my wife

ACKNOWLEDGEMENT

I express my profound sense of gratitude to Dr. N.N. Jani - my research guide, who provides me undeviating encouragement, indefatigable guidance and valuable suggestions throughout the research project.

I take opportunity to express my deep sense of gratitude to Dr. K. P. Joshipura, Vice-Chancellor of the Saurashtra University for his consistent encouragement to the research and development.

I express my gratitude to all those officials in Computer Centre of Saurashtra University and Proseon Infolabs at Baroda, who spared their precious time to me and thus provided me valuable information and insight into various important issues related to the study.

I also give my sincere thanks to faculty members and of Department of Computer Science, Saurashtra University, who debated few key issues and offered critical comments on several aspect of the study. I am also thankful to the administrative staff of the department, who has always been a support of inspiration during my entire work.

My deepest thanks to the reviewer of my thesis. I am highly indebted to my parents, my wife and sister, all my relatives and friends who constantly inspired me.

Thumar Satish Gobarbhai
Rajkot.

ABSTRACT

There is a massive increase of information available on electronic media. This profusion of resources on the electronic media and storage capacity increase gave rise to considerable interest in the research community. Traditional information retrieval techniques have been applied to the document collection on the Internet, and panoply of search engines and tools have been proposed and implemented. However, the effectiveness of these tools is not satisfactory. None of them is capable of discovering knowledge from the Internet. The Web is still evolving at an alarming rate. In a recent report on the future of database research known as the Asilomar Report, it has been predicted that in ten years from now, the majority of human information will be available on the World-Wide Web or at some central location.

In this work we propose a data warehouse model of Saurashtra University on top of the existing infrastructure. We also work to developed a data-mining model for University using the data warehouse. We also gave the model for text mining and web mining for any kind of web pages or text data.

Large collections of Saurashtra University data being gathered for a myriad of applications. The use of student result is being used for the data mining. Data mining from such a data corpus can lead to interesting discoveries. For the web mining part We have used many different types of web sites. And for text mining We have taken the data from SEC EDGAR.

LIST OF FIGURES

"Analyzing and Developing Technique for Mining Very Large Databases to Support Knowledge Exploration"

Content

1.1. Selection of Research Title

Data mining has appeared as one of the tools of choice to better explore software engineering data. The constant increase on software and hardware infrastructures will only increase the availability of data in software organizations. The recent boom on data mining research will more than likely increase the number and quality of tools available for data analysis. One does not need to be a visionary to predict that during the next few years the use of data mining techniques will increase sharply among software engineering practitioners and data analysts.

Software engineering professionals are trying to meet the availability of new data analysis methods and tools with caution. First of all, these techniques can only be as good as the data one collects. Having good data is the first requirement for good data exploration. There can be no knowledge discovery on bad data.

Good data is, however, just the first step. The second requirement for successful data mining is to understand what is being analyzed. One has to understand what is being sifted through a data mining technique in order to recognize (and use) it as "new knowledge." Data analysts and domain experts must understand the semantics of the data they propose to mine. If the data is being extracted from an external source, one must make sure that he knows what he is getting. One should also recognize the data limitations, treat missing values, and identify noisy information.

Now, assuming that one has good data and has successfully extracted and preprocessed it. What next? Well, this will depend on

the problem at hand and the task one wants to perform. Most of the time, the most attractive task to perform is to build a model that solves the problem at hand. Model building is indeed the most common application of data mining nowadays. In software engineering, reports on model building are almost as old as the discipline itself. It just happens that building good models is very hard. This is especially true in software engineering, a discipline that is very complex, human intensive, and many times abstract. It is our belief that model building should be the last step of the ladder. Model building tasks should be grounded on data exploration. Software engineers should try to understand their object of analysis be it a resource, a product, or a process before trying to model it.

Data analysts need to consider that every data mining technique has strengths and weaknesses. The effectiveness of a technique is very dependent on the type of data at hand. Some techniques such as neural networks are well suited to analyzing numeric data, others such as classification trees are well suited to analyzing categorical data. A technique's effectiveness also depends on the number of data points available for analysis. Whenever possible, the data analyst should compare or even combine available techniques in order to obtain the best possible results. Domain experts are striving to acquire new domain knowledge whenever possible. Domain knowledge is the basis for software process improvement. Techniques that produce interpretable models like classification trees and Bayesian belief networks can be a valuable source of new domain knowledge. These techniques allow domain experts to use their background domain knowledge to interpret the models built by the automated algorithms.

1.2. Survey of the research

As recently as three years ago, data mining was a new concept for many people. Data mining products were new and marred by unpolished interfaces. Only the most innovative or daring early adopters were trying to apply these emerging tools. Today's products have matured, and data mining is accessible to a much wider audience. We are even seeing the emergence of specialized vertical market data mining products.

But what kinds of business problems can use data mining technology to solve their business problem and what must users understand to apply these tools effectively? Data mining extracts new information from data. Data mining tools do more than query and analysis tools, OLAP tools, or statistical techniques like an analysis of variance to name just a few examples. Understanding the kinds of questions data mining tools can answer is the best way to appreciate how they differ from other approaches.

Other query and analysis tools can respond to questions such as, "Do sales of Product X increase in November," or "Do sales of Product X decrease when there is a promotion on Product Y?" In contrast, you can use a data mining tool to ask, "What are the factors that determine sales of Product X?"

The traditional approach is much more painstaking for the analyst. A critical distinction has to do with who drives the bus. With traditional tools, the analyst starts with a question, or an assumption, or perhaps just a hunch and explores the data and builds a model, step by step, working to prove or disprove a theory. It is the analyst's

responsibility to propose each hypothesis, test it, propose an additional or substitute hypothesis, test it, and so on, and in this iterative way, build a model. While this responsibility does not disappear entirely with data mining, data mining shifts much of the work of finding an appropriate model from the analyst to the computer. This has the following potential benefits:

- Generating a model requires less manual effort. (It's more efficient.)
- You can evaluate a larger number of models, and this increases the odds of finding a better model.
- The analyst needs less technical expertise because more of the step-by-step procedure is automated.

1.3. Research Motivation

Modern organizations are under enormous pressure to respond quickly to changes in the market. Clearly, in order to do this one needs rapid access to varieties of information before one can frame a logical decision. To assist one in making the right choices for one's organization, it is essential to analyze the past and identify relevant trends. Obviously, in order to perform any trend analysis one must have access to entire information, and this information is mainly stored in very large/huge databases. The easiest approach to gain access to this huge data and extract patterns to support effective decision-making in business environment needs to set up a data warehouse. Once a data warehouse is organized and the be housing historical data as well as a data mart for operational data, one can access information from operational data and some patterns from data warehouse by way of data mining. These patterns will be analyzed to organize and summarize as business intelligence. In this competitive age it is the business intelligence with quality information at backbone can give the power of with standing and growth.

General objective of my research

The objective is to explore how the data mining interprets database information and "how this information is used to organize actions". This has created a special interested in making comparison of different algorithms of data mining with effort to develop experimental paradigms that allow testing the mining algorithms.

The data mining strategies do not follow the same track but show different pictures in different situations. The variability may make the implementation complex and success rate not to the anticipated level. To smoothen the track, the efforts are made to model the track of data mining that is expected to cover many different situations.

With fast increase in the number of companies using data warehouse and data mining, the model that is to be proposed under this research initiative will help the user to establish data warehouse structure with data mining solution.

1.4. Major Characteristics of the Research

Design Strategies

1. Naturalistic inquiry—Studying real-world situations as they unfold naturally; no manipulative and no controlling; openness to whatever emerges (lack of predetermined constraints on findings).

2. Emergent design flexibility—Openness to adapting inquiry as understanding deepens and/or situations to change; the researcher avoids getting locked into rigid designs that eliminate responsiveness and pursues new paths of discovery as they emerge.

Data-Collection and Fieldwork Strategies

3. Qualitative data

4. Personal experience and engagement

5. Empathic neutrality and mindfulness

6. Dynamic systems

Analysis Strategies

7. Unique case orientation

8. Inductive analysis and creative synthesis— Immersion in the details and specifics of the data to discover important patterns, themes, and interrelationships; begins by exploring, then confirming, guided by analytical principles rather than rules, ends with a creative synthesis.

9. Holistic perspective—The whole phenomenon under study is understood as a complex system that is more than the sum of its parts; focus on complex interdependencies and system dynamics that cannot meaningfully be reduced to a few discrete variables and linear, cause effect relationships.

10. Context sensitivity

1.5. Overview of KDD

The term Knowledge Discovery in Databases abbreviated as KDD refers to the broad process of extracts knowledge from huge data, and emphasizes the "high-level" application of particular data mining methods. It is of interest to researchers in machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition for expert systems, and data visualization. The unifying goal of the KDD process is to extract knowledge from data in the context of large databases.

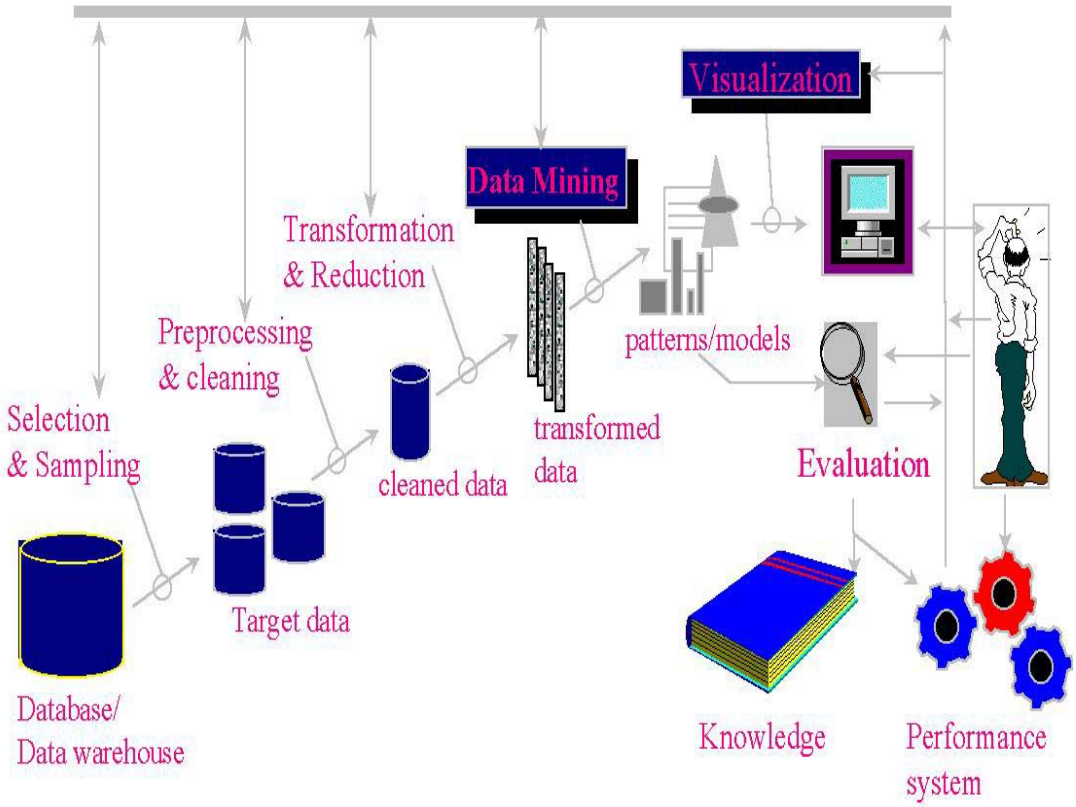


Fig. 1.1 KDD

Reference: Fayyad, Piatetsky-Shapiro, Smyth, "From Data Mining to Knowledge Discovery: An Overview", in Fayyad, Piatetsky-Shapiro, Smyth, Uthurusamy, Advances in Knowledge Discovery and Data Mining, AAAI Press / The MIT Press, Menlo Park, CA, 1996, pp.1-34

The overall process of finding and interpreting patterns from data involves the repeated application of the following steps:

1. Developing an understanding of
    - the application domain
    - the relevant prior knowledge
    - the goals of the end-user .
2. Creating a target data set: selecting a data set, or focusing on a subset of variables, or data samples, on which discovery is to be performed.
3. Data cleaning and preprocessing.
    - Removal of noise or outliers.
    - Collecting necessary information to model or account for noise.
    - Strategies for handling missing data fields.
    - Accounting for time sequence information and known changes.
4. Data reduction and projection.
    - Finding useful features to represent the data depending on the goal of the task.
    - Using dimensionality reduction or transformation methods to reduce the effective number of variables under consideration or to find invariant representations for the data.
5. Choosing the data mining task.
    - Deciding whether the goal of the KDD process is classification, regression, clustering, etc.

6. Choosing the data mining algorithm(s).
    - Selecting method(s) to be used for searching for patterns in the data.
    - Deciding which models and parameters may be appropriate.
    - Matching a particular data mining method with the overall criteria of the KDD process.
7. Data mining.
    - Searching for patterns of interest in a particular representational form or a set of such representations as classification rules or trees, regression, clustering, and so forth.
8. Interpreting mined patterns.
9. Consolidating discovered knowledge.

KDD Process Definitions

Knowledge discovery in databases is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.

Data	A set of facts, F.
Pattern	An expression E in a language L describing facts in a subset F_E of F.
Process	KDD is a multi-step process involving data preparation, pattern searching, knowledge evaluation, and refinement with iteration after modification.
Valid	Discovered patterns should be true on new data with some degree of certainty. Generalize to the future (other data).
Novel	Patterns must be novel (should not be previously known).
Useful	Actionable; patterns should potentially lead to some useful actions.
Understandable	The process should lead to human insight. Patterns must be made understandable in order to facilitate a better understanding of the underlying data.

1.6. Contributions

The major contributions of this thesis are summarized as under:

- Proposal of a framework and model, for hierarchical organization and management of Data for resource and implicit knowledge discovery.

- Presentation of strategies for mediating between different data views.

- Proposal of automatic transferring the different OLTP data to Data Warehouse.

- Proposal of architecture for a data mining and OLAP system from Data Warehousing cubes.

- Comparisons of different Data Mining Algorithms and their uses.

- Proposal of architecture for Text Mining and Web Mining.

1.7. Organization of the thesis

This thesis consists of five chapters. In chapter 1, researcher introduces overview of KDD and the motivation for this work.

In chapter 2, researcher tries to cover the theory behind this research. This part gives a brief introduction to data warehouse and data mining, its history, functionality and some of its technical aspects, however, that part merely serves as an extended introduction, for the technology presented has left its mark on markets around the world for over a decade.

In chapter 3, researcher proposed a data warehouse model for a Saurashtra University. Here researchers have given the step-by-step process for implementing the data warehouse solution and also discuss the out comes of the system.

In chapter 4, researcher proposed a data-mining model. Here researcher has performed data mining using various available data mining algorithm and try to compare the results of those algorithms. Here researchers have also given a common model for web mining and text mining.

Finally, in chapter 5, researcher summarizes and discusses future directions for research.

At last researcher has given the appendix where researcher has discussed the various available tools and its features for data mining. Researcher has also given the comparison between tools. You can also found the some important source code of the research.

2.1 Introduction to Data Warehouse

2.1.1 Types of Systems

Perhaps the most important concept that has come out of the Data Warehouse movement is the recognition that there are two fundamentally different types of information systems in all organizations: operational systems and informational systems.

"Operational systems" are just what their name implies; they are the systems that help us run the enterprise operation day-to-day. These are the backbone systems of any enterprise, our "order entry', "inventory", "manufacturing", "payroll" and "accounting" systems. Because of their importance to the organization, operational systems were almost always the first parts of the enterprise to be computerized. Over the years, these operational systems have been extended and rewritten, enhanced and maintained to the point that they are completely integrated into the organization. Indeed, most large organizations around the world today couldn't operate without their operational systems and the data that these systems maintain.

On the other hand, there are other functions that go on within the enterprise that have to do with planning, forecasting and managing the organization. These functions are also critical to the survival of the organization, especially in our current fast-paced world. Functions like "marketing planning", "engineering planning" and "financial analysis" also require information systems to support them. But these functions are different from operational ones, and the types of

systems and information required are also different. The knowledge-based functions are informational systems.

"Informational systems" have to do with analyzing data and making decisions, often major decisions, about how the enterprise will operate, now and in the future. And not only do informational systems have a different focus from operational ones, they often have a different scope. Where operational data needs are normally focused upon a single area, informational data needs often span a number of different areas and need large amounts of related operational data.

In the last few years, Data Warehousing has grown rapidly from a set of related ideas into architecture for data delivery for enterprise end-user computing.

2.1.2 OLTP & DSS Systems

One of the interesting differences between the operational environment and the data warehouse environment is that of the transaction that is executed in each environment. In the operational environment when a transaction executes, the execution entails very little data. As few as two or three rows of data may be required for the execution of an operational transaction. A really large operational transaction may access up to twenty-five rows of data. But the number of rows that is accessed is modest. It is necessary to keep the row size small in the operational environment if consistent, good online response time is to be maintained.

The transaction profile in the DSS data warehouse environment is very different. The transactions run in the DSS environment may access thousands and even hundreds of thousands of rows of data. Depending on what the DSS analyst is after, the data warehouse transaction may access huge amounts of data.

The response time in the DSS environment is very different from the response time found in the OLTP environment. Depending on what is being done in the DSS data warehouse environment, response time may vary from a few seconds all the way up to several hours.

There is then a marked difference in the transaction profile found in the DSS data warehouse environment and the operational transaction processing environment.

One by product of this extreme difference in transaction profiles is that the definition of response time differs from one environment to another. In the operational environment, transaction response time is the length of time from the initiation of the transaction until the moment in time when results are FIRST returned to the end user.

In the DSS data warehouse environment, there are two response times. One response time is the length of time from the moment when the transaction is initiated until the first of the results are returned. And the second measurable response time is the length of time from the moment of the initiation of the transaction until the moment when the LAST of the results are returned. The difference between these two variables can be considerable.

Both sets of response time are needed in order to effectively measure system performance in the DSS data warehouse environment.

2.1.3 what is Data Warehousing

An enterprise data warehouse - an EDW - is an architectural component that is:

- subject oriented,
- integrated,
- non volatile, and
- time variant.

The data warehouse exists to enhance management's ability to make decisions.

Subject oriented data is data that is organized along the lines of the subjects of the corporation. Typically the subjects of the corporation consist of entities such as:

- customer
- product
- vendor
- transaction.

In years past applications were built around the functions of the corporation. The functions of the corporation (in this case a bank) might include such things as:

- Demand deposit
- Commercial savings
- Home loans
- Private trust, and so forth.

In order to understand the difference between a subject orientation and a functional orientation, examine the differences between subjects and functions. Each function will have some data that relates to each subject. Demand deposit processing has customer data, as do commercial savings, home loans, and private trust. By the same token, commercial savings has product data, as do demand deposit, home loans, and private trust, and so on.

Mapping the data from each function to each subject area shows that there is a fundamental restructuring and realignment of data that must be done in order to create a data warehouse. Data must be read in a functional format and written in a subject-oriented format.

Once the data is physically restructured in the form of subject areas, the data arrives in the data warehouse.

Integration of data refers to the transformation of data from an application into an integrated data warehouse. Each application has its own way of looking at data. But when the data is placed in the data warehouse, the different forms of the applications must be intertwined into a single form that resides in the warehouse. This means that there is a single key structure and a single structure of data to be found in the warehouse where there might have been many forms of the same data in the applications. In the data warehouse there is a single structure for customer. There is a single structure for product. There is a single structure for transaction, and so on.

As an example of the integration that might occur, suppose that there are different designations for gender in the application environment. Application A specifies gender as "M" and "F". Application B specifies gender as "1" and "0". Application C specifies gender as "X" and "Y". Application D specifies gender as "male" and "female". There needs to be a single understanding of what gender means in the data warehouse. The data warehouse designer picks the designation of "M" and "F". Wherever the application data does not exist as "M" and "F", a conversion needs to be done.

As another example of integration, suppose that different applications measure pipeline in different ways. Application A measures pipeline in centimeters. Application B measures pipeline in inches. Application C measures pipeline in yards. And pipeline D measures pipeline in thousand cubic feet per minute. The data base designer needs to pick

one measurement of pipeline - in this case the designer chooses centimeters. A conversion must be done in order to move the data into the warehouse and achieve consistency.

Non-volatility of data refers to the inability of data to be updated. Every record in the data warehouse is time stamped in one form or another. This means that every record in the warehouse is a snapshot of data as of some moment in time. If their need to be changes made to the warehouse environment, a new snapshot as of a later moment in time is made. The net result is that the data warehouse contains a historical record of snapshots of data. Once accurately created, a data warehouse record is not changed.

Time variant records are records that are created as of some moment in time. Every record in the data warehouse has some form of time valiancy attached to it. The easiest way to understand time variant records is to contrast time variant records against standard data base records. Consider a standard data base record. As the world changes, so change the values inside the database record. Data is updated, deleted, and inserted inside the standard database record. Now contrast the data warehouse record with the standard database record. Data is loaded into the data warehouse record. The moment when the data is loaded into the warehouse is usually a part of the warehouse record. And data is accessed inside the data warehouse record. But once placed inside the data warehouse, data is not changed there. Data inside the warehouse becomes an environment where the environment can be typified as load and access.

There are some other important characteristics of data in the data warehouse. Data in the warehouse is granular. This means that data

is carried in the data warehouse at the lowest level of granularity. As some examples of granularity:

- telephone data is at the call level, where an individual record of a phone call is made,
- banking data is at the record of the transaction, where individual checks and individual ATM activities are tracked,
- retail data is at the sale level, where individual items are sold,
- transportation, where the incidence of a leg of travel is sold and tracked, and so forth.

The low level of granularity found at the data warehouse allows the data to be reshaped and reformed in the many different ways that the end user needs to see the data.

2.1.4 Who provide Data Warehousing

A short amount of time the world of data warehousing has gone from maverick thinking to mainstream. A short ten years ago only the avant garde shakers and movers were thinking about and building data warehouses in our industry. Those early data warehouses were the foundation of today's important systems that address churn analysis, CRM, and a whole host of other important applications. Those early mavericks in corporate America are now known as vice presidents and CIO's.

But there has occurred stratification in today's market place. In the early 1990's there was much discussion as to what DBMS technology best fit data warehousing. Many companies spent a serious amount of resources studying the possibilities to determine the best technology for their environment. But there has today occurred a consolidation in the market place and surprisingly the consolidation has not been

along the lines of what DBMS technology is the best fit. Instead the technology decision is what processing environment best fits.

And the long-term prospects for data warehousing are not along the lines of DBMS products but along the lines of solutions.

The DBMS choices that are found in the market place today are Oracle, DB2 (in several flavors), SQL Server, Informix, and in a specialized niche - Teradata. There may be a few other pretenders for DBMS choices for a data warehouse, but in truth, these choices make up the marketplace. And surprisingly these choices do not overlap to any great extent. The way the market place has shaken out is by operating system, not DBMS.

Looking at the different DBMS, the environments that are served by the different DBMS are:

- Oracle – the general purpose Unix environment,
- DB2/UDB – the OS390 and the AIX environment,
- SQL Server – the NT environment,
- Informix – the high end Unix environment,
- Teradata – the large-scale parallel environment.

There is very little, if any, overlap among the different market places. For example, within NT, DB2 is unknown. Within OS390, SQL Server is unknown. Within Teradata there are no other DBMS players, and so forth. In other words, the market has shaken out along the lines of different operating system technologies, not in terms of different DBMS.

This comes as a surprise to those of us who fought the data warehouse battles of an earlier day and age. A decade ago there was serious debate as to the different features of the various DBMS and as to the different capabilities and capacities they have for data warehousing. Today there is just not much to say about the DBMS technologies. Instead the debate goes on about the different operating environments.

And even across the different operating environments there is starting to appear a blur. NT keeps getting to be more powerful and capable of handling more data. Teradata keeps getting to be more flexible and less brittle. DB2/UDB keeps adding functionality and operating performance. Informix keeps adding products either directly or indirectly through its spin off essential. If we wait long enough there will be very little to choose from when it comes to selecting a data warehouse environment from the standpoint of the DBMS.

The result is that within each of the operating environments the different DBMS appear to be a monopoly. This is important to the buyers in each of these spaces and to the third party vendors in these spaces. The fact that there is a monopoly within an environment has many implications.

If the differences between the different DBMS no longer matter, then what does matter? What are the differences between the different operating environments? The differences are many, and include such considerations as:

- third party software that is available,

- volumes of data that can be managed,
- the hardware platforms that the software can operate on,
- vendor support
- near line support
- the availability of trained staff for hire
- economic considerations, and so forth.

But perhaps the biggest market place that is developing for data warehouse is not that of DBMS technology at all but that of total solutions. When it comes to total solutions for data warehousing there are three prominent vendors:

- SAP with its BW offering,
- SAS with its integrated suite of products
- PeopleSoft with its EPM offering
- Computer Associates with its broad product line

Each of these vendors offers a solution rather than a point solution. Sure, there are other point solutions in the market place. There are OLAP products. There are ETL products. There are analytic offerings. But no one has offered a holistic approach other than these vendors.

The SAS product line is perhaps the earliest solution in the market place. It is robust and has special strengths when it comes to data mining and exploration. But in truth, SAS has done a good job of covering the other bases for such needs as ETL, analytics, data marts, and so forth. SAS is a player when it comes to solutions.

The most pleasantly surprising solution is that of SAP. Three years ago SAP had R/3. Then they appeared on the scene with BW. And in

a few short releases BW has come a long way – a very long way in a short amount of time. SAP's BW looks very much like the Corporate Information Factory framework. And the future looks to hold exciting promise for SAP as they start to go into arenas that the DBMS vendors have been reluctant to go into. The DBMS vendors are going to wake up and find that it is SAP that is calling the shots, not the DBMS vendors. The SAP solution is architecturally a very good solution with promise of even getting better.

Another pleasant surprise is that of PeopleSoft. PeopleSoft started off with a rendition of the Corporate Information Factory that appeared to be data mart centric (much as SAP started off.) But new releases of the EPM product have greatly enhanced the data warehouse aspect of the product so that PeopleSoft is to the point where they – like SAS and SAP – strongly resemble the Corporate Information Factory.

When it comes to solutions, Computer Associates must be mentioned. Computer Associates has undoubtedly the largest collection of software in the world. And from this large collection comes the many pieces needed for a true solution. The different components of the Corporate Information Factory can be pieced together to form a cohesive infrastructure. In this regard, CA surely qualifies as a data warehouse/ Corporate Information Factory provider of a solution, not just point products.

And why are solutions the wave of the future? It is because organizations have had a hard time building their own data warehouse. They would much prefer for a vendor to come in and present a full and integrated solution. Today's modern organization is one that installs and implements, not builds from scratch.

Accordingly, the solutions vendors are in a position to call the shots in data warehousing for years to come. Unlike other companies, the solutions vendors understand that data warehouse is about architecture and infrastructure, not technology. Try as they may, the DBMS vendors simply have not been able to twist data warehouse into a technology.

In short, the world of data warehousing is a changing world. Once a fragmented point solution world, data warehouse has turned into an architectural world where the Corporate Information Factory is at the center of the table and where solutions, not technologies, are the driving force.

2.2 Data Warehouse H/W and S/W Architecture

The architecture of a data warehouse by necessity is complex, and includes many elements. The reason for this is that a data warehouse is an amalgamation of many different systems. Integration of diverse elements is its primary concern, and to accomplish this integration, many different systems and processes are necessary.

Most software development projects require selection of the technical infrastructure, and this is true for the warehouse as well. Basic technical infrastructure includes operating system, hardware platform, database management system, and network. The DBMS selection becomes a little more complicated than a straightforward operational system because of the unusual challenges of the data warehouse, especially in its capability to support very complex queries that cannot be predicted in advance.

How does the data get into the data warehouse? The warehouse requires ongoing processes to feed it; these processes require their own infrastructure. Many times, IS shops overlook this aspect when they plan for the data warehouse. Data layers need to be understood and planned for. Data cleansing usually involves several steps; where will the "staging area" be stored? How will ongoing data loads, cleansing, and summarizing be accomplished?

Backup and recovery are interesting challenges in the data warehouse, mainly because data warehouses are usually so large.

How will users get information out of the warehouse? The choice of query tool becomes very important, and depends upon a multiplicity of factors.

Six Steps to Develop the Architecture

It is important to discuss the steps you must follow to develop this architecture. Each and every one of these steps needs to be performed in order to have the best opportunity of succeeding. The six important steps to effective data warehouse architecture development are as follows:

1. The most important step in developing effective data warehouse architecture is to enlist the full support/commitment (project sponsor) of an executive of the company.
2. Next, you must staff an architecture team with strong personnel. It is not necessarily the technology you choose for your architecture, it is the personnel you have designing and developing the architecture that makes the project successful.

3. Prototype/benchmark all the technologies you are interested in using. Design and develop a prototype that can be used to test all of the different technologies that are being considered.
4. Give the architecture team enough time to build the architecture infrastructure before development begins. For a large organization, this can be anywhere from six months to a year or more.
5. Make sure you train the development staff on the use of the architecture before development begins. Spend time letting the

development team get full exposure to the capabilities and components of the architecture.

6. Provide the architecture team an opportunity to enhance and improve the architecture as the project moves forward. No matter how much time is spent up front developing architecture, it will not be perfect the first time around.

The Data Warehouse Infrastructure

The data warehouse consists of the following architectural components, which compose the data warehouse infrastructure:

- System infrastructure: Hardware, software, network, database management system, and personnel components of the infrastructure.
- Metadata layer: Data about data. This includes, but is not limited to, definitions and descriptions of data items and business rules.
- Data discovery: The process of understanding the current environment so it can be integrated into the warehouse.
- Data acquisition: The process of loading data from the various sources. This is described in more detail in the ongoing maintenance section later in this chapter.
- Data distribution: The dissemination/replication of data to distributed data marts for specific segmented groups.
- User analysis: Includes the infrastructure required to support user queries and analysis. This is described in more detail in the "User Access" section, later in this chapter.

Data Warehouse System Infrastructure

The technical architecture of a data warehouse is one of, if not the most, important component. The reason for this is that the technical architecture is used as the base for building all the other data warehouse components. This is why the technical architecture is called the infrastructure.

The infrastructure foundation upon which the data warehouse is built is often called the platform. It is made up of the following components:

- Hardware, including operating system: Should be open, meaning that a variety of tools are able to run on the platform, and data is able to flow to/from the platform with a minimal amount of effort required. Most of the hardware of a data warehouse will consist of a number of large machines. Large machines are 6 to 8 or even 12 CPUs with a gigabyte(s) of memory and many gigabytes or even a terabyte of disk space.
- Network: Should minimize complexity, maximize bandwidth. Should connect (directly) all components and locations of the corporate enterprise that need access to the data warehouse.

- Software: Of course, the most important software component of a data warehouse is the Database Management System (DBMS) (see the next section). However, there are other important software components as well: the monitoring, administration, and network management tools used to maintain the database; the software used to support user access; and data modeling tools used by the development staff

to design, implement, and maintain the data warehouse (i.e. ER/Win, Oracle Enterprise Manager, ClearCase [Configuration Management], and third-party utilities).

- Personnel: This may seem like an odd component of data warehouse architecture, but it is the most important (and the most expensive!). There are a number of technology choices on the market today, and each of these technologies has good features and bad features. Therefore, choosing the right components is not an exact science. The key factor in whether or not the technology will work is the skill level of the individuals designing and developing the architecture. Good components and experienced architects/developers will make the difference in the end.

System Architecture

The system architecture is the overall blueprint that you will follow when building your data warehouse platform. It is the underlying foundation that governs many of the decisions you will need to make when building and managing your data warehouse platform. Given this, it is no surprise that there are probably as many different data warehouse architectures as there are data warehouses. However, they can usually be grouped into one of two main categories: a three-tiered architecture or a two-tiered architecture.

Three-Tiered Architectures

In a three-tiered design, the first tier is comprised of your operational systems that are already in place. These are the transaction processing systems that collect the data about all the events that

occur within your company. The data collected by these systems is fed into your data warehouse. The second and third tiers of this architecture are the data warehouse and the data marts, respectively.

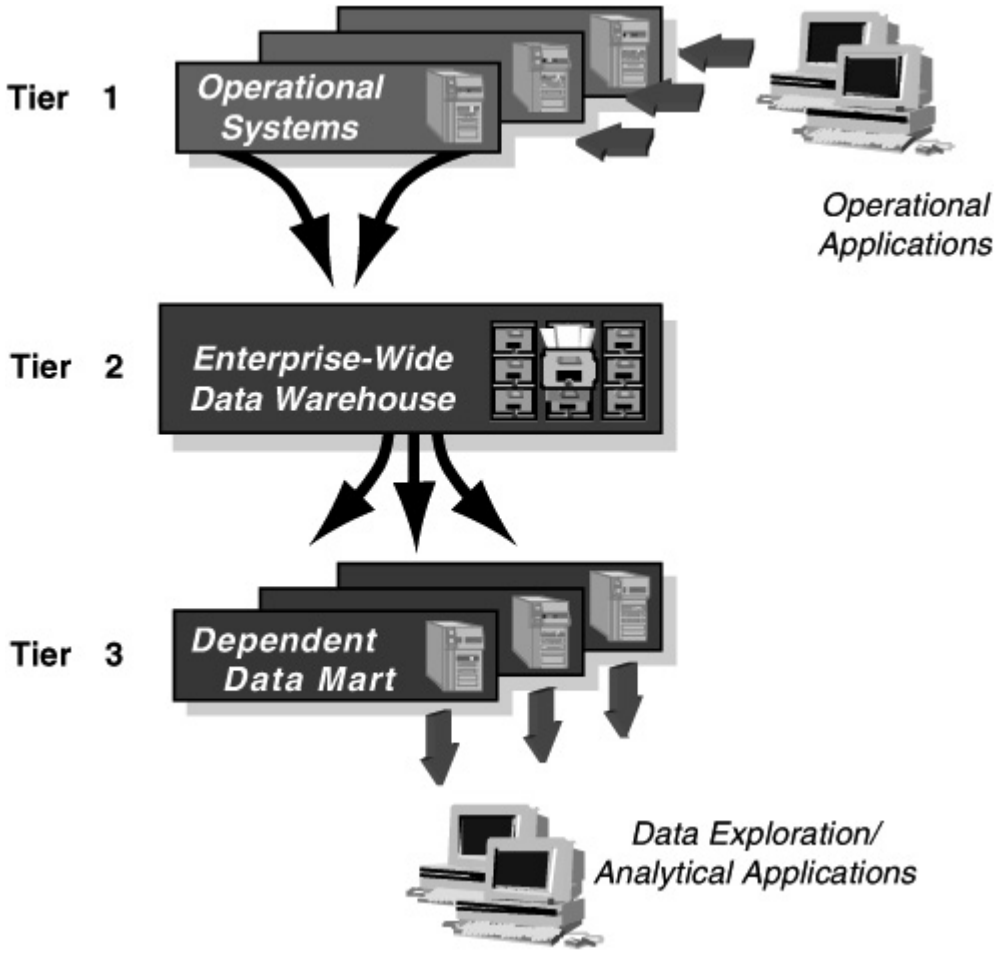


Fig. 2.1 Three-Tiered Architectures

Why do we recommend separating the data warehouse and the data marts into different tiers instead of combining them both into the second tier? Because there are two different types of functions that must be addressed when building a large-scale data warehouse

environment: data consolidation (i.e., getting data in) and data analysis (i.e., getting information out).

Data consolidation refers to the process of extracting, cleaning, and transforming the data from disparate, unconsolidated operational systems into one consolidated repository. Data analysis refers to the process of having end-users access, manipulate, and generally analyze the data looking for useful insights. With two different needs, it is much more scalable to split those functions into two different tiers. In essence, we're using a form of functional parallelism here to improve scalability. We're assigning different tasks to different computers. Also, it is simply easier to address these different needs if we have one tier focused on optimally solving data consolidation issues and another tier focused on optimally solving data exploration issues.

With a three-tier architecture, the data warehouse tier is responsible for the consolidation activities. Its goal is to take data from the operational systems, consolidate it, and then feed portions of the consolidated data to the various data marts that are responsible for the exploration activities. Since these data marts get their data from the data warehouse, we refer to them as "dependent data marts"—they are dependent on the existence of an enterprise data warehouse.

Highlighting the differences between data warehouses and data marts, the data warehouse is fed by multiple operational systems (with additional externally purchased data occasionally included as well), so it must therefore perform the required extractions and transformations. The data marts, on the other hand, only need to

extract data from a single source that is already consolidated, the data warehouse. The marts will also occasionally include additional external data, but the amount of consolidation that must be performed is still far less than for the data warehouse.

Also, the data warehouse needs to store its information in a form that is "application generic," since it will be feeding multiple data marts, each of which is focused on a different set of business problems. As a data warehouse designer, you therefore want to keep the data stored in the data warehouse tier in its most flexible form, which is the unsummarized, detail level form. This means you should design a database schema for the second tier that has much of the flavor of a traditional third-normal form schema. In contrast, data marts need to store their information in a form that is "application specific," tailored to the needs of the explorer or farmer. This means that you will want summarizations, subsets, and/or samples in your data mart that are specific to the particular business unit that is using the data mart.

You also want the data in your data marts to be easily accessible by the end-user community. Traditional third-normal form schemas, though they excel in minimizing data redundancy, are fairly poor models for end users to try to understand and analyze, and is, therefore, usually a poor choice for data marts. Instead, data marts should use dimensional models and star schema designs, which make heavy use of redundancy to make it far easier for end users to navigate their way through large volumes of data without getting lost.

Even the choice of optimal hardware and DBMS is different for the data warehouse and the data marts. The data warehouse tier has to

act as a repository for enormous amounts of data that span many different organizations and subject areas. In addition, more data from both existing and new subject areas is constantly being added to the data warehouse. And the data warehouse tier must be able to feed an ever-growing number of application-specific data marts. This means that the data warehouse tier must be an industrial-strength, highly scalable, enterprise-class hardware and DBMS.

The data marts, however, need only focus on a single business problem. To be sure, they will see growth in their particular subject area, because new transactions relating to that subject area are continually being collected from the operational systems, but the magnitude of this growth is far less. Therefore the data mart tier is usually a smaller (though still scalable), department level hardware and DBMS solution.

BENEFITS OF A THREE-TIERED ARCHITECTURE

The main benefits of three-tiered data warehouse architecture are high performance and scalability. The high performance comes from the fact that the inclusion of data marts allows us to partition different query workloads across different data marts. This means that the workload levels of users of other data marts will not affect users of one data mart. For example, if users of the sales data mart are executing complex, long-running queries that are highly resource intensive, this will have no effect on the performance seen by users of the completely separate finance data mart. The resulting increase in end-user satisfaction levels is enormous. In our experience, end users get frustrated by their own queries slowing down their machine, but nothing infuriates end users more than having someone

else in some other department bring a machine that everyone must share to its knees. By separating the workloads into physically separate marts, you can prevent different sets of users from adversely affecting each other.

This architecture is also very scalable. As we stated above, we're assigning extraction and consolidation functions to one tier and end-user query and data analysis functions to another tier. This is nothing more than a straightforward use of the concept of functional parallelism, which we described in the previous chapter as one effective method for improving scalability. Both the second and third tiers can be individually scaled up as well. Since the data warehouse tier is a large and highly scalable, it can be scaled up by adding more resources to it (such as processors, disks, I/O controllers, and so on). Scaling up the data-mart tier is done by simply adding more data marts to service new user populations, address new subject areas, or focus on providing a new type of functionality such as data mining. Since data marts are usually far cheaper to build than data warehouses, it is fairly easy to add another data mart when you need to explore a new business area.

COSTS OF A THREE-TIERED ARCHITECTURE

However, the cost issue raises a much larger issue concerning three-tiered architecture. The main issue is the multisubject, enterprise-wide data warehouse that is at the center of this architecture. To design this data warehouse is a complex process. You must spend time consolidating various subject areas, and this can involve many long meetings (and many long debates) with representatives from various organizations. In addition, there is quite a large time

investment involved with building an enterprise-wide data warehouse. Defining what business problems you want to solve, finding where all the data required to solve those problems is located, writing all the required extraction, cleansing, and transformation routines, loading all the data into the database, and then tuning the resulting system is no trivial task. In fact, the average enterprise-wide data warehouse takes about 18 months to build. Finally, the cost of such a system is not trivial, often reaching into the many millions of dollars.

For many organizations, the complexity of the project and the required time and cost investments are prohibitive. And even if they weren't, having the first step in a data warehouse development project be the development of a centralized, enterprise-wide data warehouse is a risky proposition. The dynamic, organic nature of a data warehouse environment means that it doesn't really make sense to build something that large as your very first step. Your organization's needs will likely change by the time you finally deliver your data warehouse. You will have spent a large amount of effort building a wonderful system that helps give answers to questions that are no longer the most important questions that need answering.

Because of these issues, many organizations have dropped the notion of a three-tiered architecture and moved to a simpler two-tiered architecture. We'll first describe the advantages of this approach, but then we'll highlight the serious flaws with this approach. To avoid leaving you with the feeling that there's no approach that doesn't have significant problems, we'll present a simple solution that gives you the performance and scalability advantages of three-tiered

architecture, but doesn't incur the initial up-front cost and time investments.

Two-Tiered Architectures

There are two ways to build a two-tiered solution. The first involves just building the enterprise-wide data warehouse without the data marts, and having all the end-users directly access it. With this architecture, you may save some money because you don't have to buy separate data mart hardware to store copies of the data that already exists in the central data warehouse, and you may save some time by not having to build data marts. But data marts are generally not as complicated to build, so the timesavings won't be dramatic. Ultimately there are two main problems with this approach. First, you are starting by building the central data warehouse, so the majority of the complexity, time, cost, and risk are still factors. Second, since all departments and all users will be sharing a single database, you lose the ability to separate workloads among different user groups. Since there really aren't many advantages to this version of the two-tiered approach, we won't discuss it further.

The other far more common approach is to build the data marts without building the centralized data warehouse. Since these data marts do not depend on the existence of a consolidated data warehouse, we refer to them as "independent data mart"

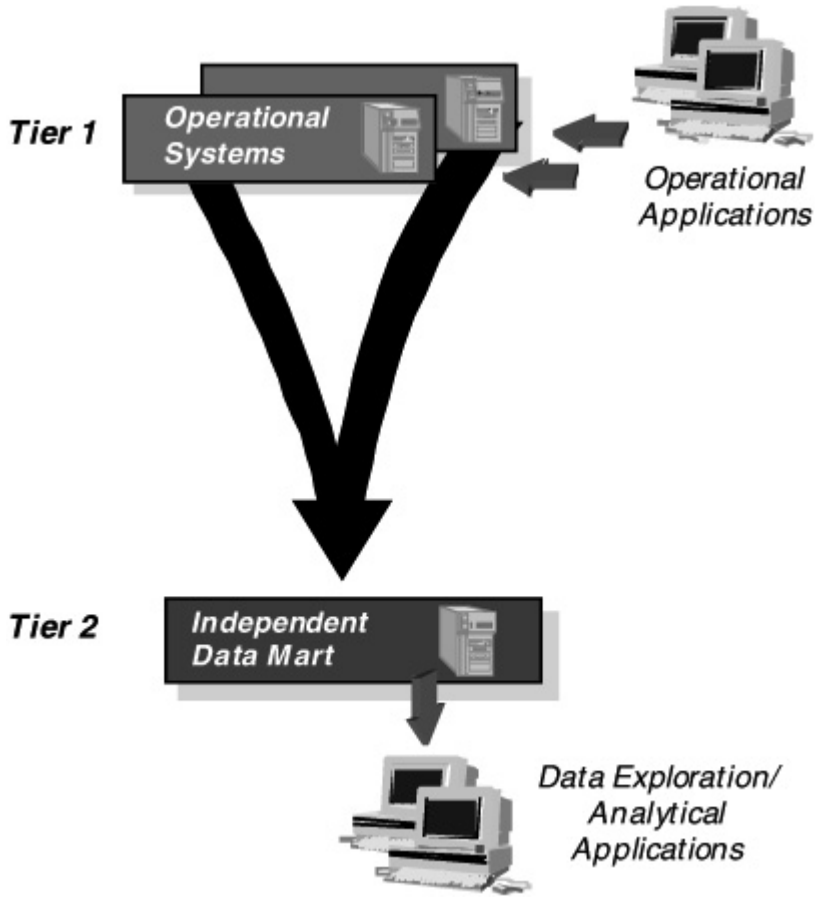


Fig. 2.2 Two-Tiered Architectures

The advantages of building a two-tiered environment using the independent data mart approach are fairly enticing:

- The data mart traditionally will only have data pertaining to one or two subject areas, so there is much less complexity involved in the design and implementation of this architecture.
- Since you are dealing with fewer data sources and less data, the time required to build a data mart can be on the order of 3 or 4 months, not 18 months.

- As we discussed earlier, the hardware required for the data mart, though still scalable, are generally much smaller departmental machines, not enterprise-class machines.

The costs will be far lower as well. Essentially, since few organizations have the multiple millions of dollars and 18 months it takes (on average) to build an enterprise data warehouse all at once, the only option is to be less ambitious and start with a data mart.

2.3 Data Marts

The data mart is the structure that is fed from the enterprise data warehouse. The data mart is where the end user has the most interaction with the enterprise data warehouse environment. The data marts are shaped after the requirements of the different departments that own the data mart. Because a different department owns each data mart, the data marts all look different from each other.

Very little detailed data is found in the data mart. Instead the detailed data that is found in the enterprise data warehouse becomes the source of data inside the data mart. The enterprise data warehouse detailed data is reshaped according to the departmental understanding of the data before the data is entered into the data mart.

The data found in the data mart can be described as residing in star schemas (star joins) or snowflake structures. These star joins reflect the different ways that the departments look at their data.

Data is often summarized and/or otherwise aggregated as it moves into the data mart.

The world of data mart revolves around technology and structures that are designed for end user access and analysis. The leading of these structures is the cube or the multi dimensional structure.

The person most involved with the data mart is the farmer. Most of the queries are well structured before the query is submitted.

Many reports emanate from the data mart. In addition, the data mart can spawn smaller desktop versions of the data mart that can be fed and maintained at an individual's workstation.

The data mart is reconcilable back to the enterprise data warehouse. The only legitimate source of data for the data mart other than the enterprise data warehouse is external data.

Distributed metadata resides at the data mart. With distributed metadata in the data mart, the end user can see:

- The metadata that applies to the local data mart,
- The metadata that resides elsewhere but which is relevant to the local data mart,
- Locally protected metadata that is not open and available to other departments,

- Metadata residing at architectural entities other than data marts, such as ODS, enterprise data warehouse, and so forth.

In addition, with distributed metadata the data mart user can look at both technical and business metadata.

Data Mart Structure

The data structure found at the data mart is best described as a star join or a snowflake structure. A star join structure has two basic components - a fact table and supporting dimension tables. A fact table represents the data that that is the most populous in the data mart. In a telephone company, call usage data is typically the most populous. In a bank, checking and ATM activity are typically the most populous data. In a retail environment, sales and stocking are the most populous entities, and so forth.

The fact table is a composite of many types of data that have been prejoined together. The fact table contains:

- a primary key reflecting the entity the table has been built for, such as an order, a sale, a phone call, and so forth,
- Information about the primary key,
- Foreign keys relating the fact table to the dimensions,
- Non-key foreign data that is carried with the foreign key. This non-key foreign data is included if it is regularly used in the analysis of data found in the fact table.

The fact table is highly indexed. It is not unusual for there to be as many as 30 to 40 indexes on the fact table. In some cases every column in the fact table is indexed. The net result of all of the indexing is that data in a fact table is highly accessible. However, the

amount of resources required for the loading of the indexes must be factored into the equation.

As a rule, fact tables are not updated. They have data loaded into them, but once a record is loaded properly, there is no need to go into the record and alter any of its contents.

The dimension tables surround the fact tables. The dimension tables contain the data that is non-populace. The dimension tables relate to the fact tables through a foreign key relationship. Typical dimension tables might be product list, customer lists, vendor lists, and so forth, depending of course on the data mart being represented.

The source of the data found in the data mart is the enterprise data warehouse. All data - with one exception - should pass through the enterprise data warehouse before finding its way into the data mart. The one exception is data that is specific only to the data mart that is used nowhere else in the environment. External data often fits this category. If however, the data is used anywhere else in the DSS environment, then it must pass through the enterprise data warehouse.

The data mart contains two kinds of data, generally speaking - detailed data and summary data. The detailed data in the data mart is contained in the star join, as previously described. It is noteworthy that the star join may well represent a summarization as it passes out of the enterprise data warehouse. In that sense, the enterprise data warehouse contains the must elemental data while the data mart contains a higher level of granularity. However, in the eyes of the data mart user, the star join data is as detailed as data gets.

The second kind of data the data mart contains is summary data. It is very common for the farmer to create summaries from the data found in the star join. A typical summary might be monthly sales totals by sales territory. Because summaries are kept on an ongoing basis, history is stored in the data mart. But the preponderance of history that is kept in the data mart is stored at the summary level. Very little history is kept at the star join level.

The data mart is refreshed from the enterprise data warehouse on an as needed basis. It is common to have a data mart refreshed every week or so. However, refreshment can be done either much more or much less frequently, depending on the needs of the department that owns the data mart.

Data Mart Uses

The data mart is the most versatile of the data structures. The data mart allows data to be examined from the standpoint of many perspectives - from a detailed perspective, from a summarized perspective, across much data, across few occurrences of data.

The primary user of the data is a user that can be called a farmer. The farmer knows what he/she is looking for when the query is submitted. The farmer does the same activity repeatedly against different occurrences of data. The data is structured inside the data mart so that it is optimal for the access of the farmer. The farmer spends a fair amount of time with the DWA gathering and synthesizing requirements before the data mart is built.

The farmer looks at summarized data, at exception based data, at data created on a periodic basis, and other types of data. The farmer

looks upon the data mart as a mission critical component of the environment.

The other use of the data mart is as a spawning ground for insightful analysis by the explorer community. While the data mart is not designed to support exploration, many of the insights, which merit deeper exploration, are initiated at the data mart. The explorer has the inspiration at the data mart, does some cursory exploration there, and then moves down to the exploration warehouse for the detailed analysis that is required for exploration. The data mart lacks the foundation for exploration because data is structured along the lines of a department, because data is usually summarized and the explorer needs detail, and because the data mart has a limited amount of historical data. Never the less, the data mart is a fertile breeding ground for insight.

2.4 OLAP

In the OLAP world, there are mainly two different types: Multidimensional OLAP (MOLAP) and Relational OLAP (ROLAP). Hybrid OLAP (HOLAP) refers to technologies that combine MOLAP and ROLAP.

2.4.1 MOLAP

This is the more traditional way of OLAP analysis. In MOLAP, data is stored in a multidimensional cube. The storage is not in the relational database, but in proprietary formats.

Advantages:

- Excellent performance: MOLAP cubes are built for fast data retrieval, and are optimal for slicing and dicing operations.
- Can perform complex calculations: All calculations have been pre-generated when the cube is created. Hence, complex calculations are not only doable, but they return quickly.

Disadvantages:

- Limited in the amount of data it can handle: Because all calculations are performed when the cube is built, it is not possible to include a large amount of data in the cube itself. This is not to say that the data in the cube cannot be derived from a large amount of data. Indeed, this is possible. But in this case, only summary-level information will be included in the cube itself.

- Requires additional investment: Cube technologies are often proprietary and do not already exists in the organization. Therefore, to adopt MOLAP technology, chances are additional investments in human and capital resources are needed.

2.4.2 ROLAP

This methodology relies on manipulating the data stored in the relational database to give the appearance of traditional OLAP's slicing and dicing functionality. In essence, each action of slicing and dicing is equivalent to adding a "WHERE" clause in the SQL statement.

Advantages:

- Can handle large amounts of data: The data size limitation of ROLAP technology is the limitation on data size of the underlying relational database. In other words, ROLAP itself places no limitation on data amount.
- Can leverage functionalities inherent in the relational database: Often, relational database already comes with a host of functionalities. ROLAP technologies, since they sit on top of the relational database, can therefore leverage these functionalities.

Disadvantages:

- Performance can be slow: Because each ROLAP report is essentially a SQL query (or multiple SQL queries) in the relational database, the query time can be long if the underlying data size is large.

- Limited by SQL functionalities: Because ROLAP technology mainly relies on generating SQL statements to query the relational database, and SQL statements do not fit all needs (for example, it is difficult to perform complex calculations using SQL), ROLAP technologies are therefore traditionally limited by what SQL can do. ROLAP vendors have mitigated this risk by building into the tool out-of-the-box complex functions as well as the ability to allow users to define their own functions.

2.4.3 HOLAP

HOLAP technologies attempt to combine the advantages of MOLAP and ROLAP. For summary-type information, HOLAP leverages cube technology for faster performance. When detail information is needed, HOLAP can "drill through" from the cube into the underlying relational data.

Popular Tools

- Business Objects
- Cognos
- Hyperion
- Microsoft Analysis Services
- Micro Strategy

2.5 ETL

The **ETL** (Extraction, Transformation, Loading) process typically takes the longest to develop, and this can easily take up to 50% of the data warehouse implementation cycle or longer. The reason for this is that it takes time to get the source data, understand the necessary columns, understand the business rules, and understand the logical and physical data models.

When it comes to ETL tool selection, it is not always necessary to purchase a third-party tool. This determination largely depends on three things:

- Complexity of the data transformation: The more complex the data transformation is, the more suitable it is to purchase an ETL tool.
- Data cleansing needs: Does the data need to go through a thorough cleansing exercise before it is suitable to be stored in the data warehouse? If so, it is best to purchase a tool with strong data cleansing functionalities. Otherwise, it may be sufficient to simply build the ETL routine from scratch.
- Data volume. Available commercial tools typically have features that can speed up data movement. Therefore, buying a commercial product is a better approach if the volume of data transferred is large.

ETL Tool Functionalities

While the selection of a database and a hardware platform is a must, the selection of an ETL tool is highly recommended, but it's not a

must. When you evaluate ETL tools, it pays to look for the following characteristics:

- **Functional capability**: This includes both the 'transformation' piece and the 'cleansing' piece. In general, the typical ETL tools are either geared towards having strong transformation capabilities or having strong cleansing capabilities, but they are seldom very strong in both. As a result, if you know your data is going to be dirty coming in, make sure your ETL tool has a strong cleansing capability. If you know there are going to be a lot of different data transformations, it then makes sense to pick a tool that is strong in transformation.
- **Ability to read directly from your data source**: For each organization, there is a different set of data sources. Make sure the ETL tool you select can connect directly to your source data.
- **Metadata support**: The ETL tool plays a key role in your metadata because it maps the source data to the destination, which is an important piece of the metadata. In fact, some organizations have come to rely on the documentation of their ETL tool as their metadata source. As a result, it is very important to select a metadata tool that works with your overall metadata strategy.

Popular Tools

- Data Junction
- Essential Data Stage
- Ab Initio
- Informatica

2.5.1 Meta Data

Metadata exists throughout the information systems environment and has existed for many years. There are many kinds of metadata. This conversation is about the kinds of metadata that exist in the DSS data warehouse environment. To understand metadata, as it exists in the world of data warehouse, consider the dilemma of a DSS analyst as the DSS analyst is assigned the task of producing a new report. Assume that the DSS analyst is someone who has just come to work at your company.

The DSS analyst is overwhelmed with the assignment because the DSS analyst does not know where to start. Someone suggests to the analyst that there might be some useful data in the data warehouse. So the DSS analyst heads for the data warehouse.

The first obstacle waiting for the DSS analyst is that the analyst has no idea what is in the warehouse. The analyst turns to the metadata that describes the data warehouse. The metadata that the analyst finds describes what tables and what attributes are there in the data warehouse. The analyst is grateful to find out what data is available as a source for the new report.

But by finding out what is in the data warehouse, the analyst has a new problem. There are ten tables in the warehouse that appear to be likely candidates for the report the analyst is doing. The DSS analyst does not know where to start. Once again the DSS analyst turns to metadata.

The next type of metadata that the DSS analyst uses is source information about where the data came from that resides in the data warehouse. The metadata that tells what the source is for each of the tables and fields in the warehouse greatly helps the DSS analyst to understand which table is the best source for the report.

The next type of DSS metadata the DSS analyst finds useful is that of a description of the logic that is used to integrate the data into the data warehouse. In most cases the data that arrives in the data warehouse has undergone a serious conversion and transformation process. The DSS analyst needs to understand that process in order to create the best possible report. The logic of conversion helps the DSS analyst to explain to management why there are differences in data values and how those values can be resolved. That logic is explained in metadata.

The next type of metadata the DSS analyst finds of interest in the building of the report is that of the refreshment tracking of the loading of the metadata. Different data is of different

freshness inside the warehouse. Some data has been loaded today. Some data was loaded last night. Other data is a month old. Merely looking at the data in the data warehouse does not tell the DSS analyst which is the freshest. Yet the DSS analyst needs to know just how fresh data is inside the warehouse in order to determine which data is the best source for the report that must be written. Refreshment tracking of metadata then becomes an important type of metadata.

There are in fact many different kinds of metadata ion the DSS environment, all of which make the job of the DSS analyst easier and more effective.

Meta Data Architecture

There are two basic architectures for metadata in the DSS data warehouse environment. Those architectures are a centralized architecture and a distributed architecture.

The classical metadata architecture is a centralized architecture. A centralized architecture is one where metadata is stored and managed centrally. The rights of creation and update are invested in a central administrator. The great appeal of the centralized approach is that data can be uniformly defined and used over the corporation. Once defined centrally, the metadata will have no conflict in its definition.

The centralized approach to metadata architecture has the problem of not accommodating the need for local autonomy of metadata. Local autonomy of metadata refers to the need to create and manage metadata entirely within the confines of a department. There are other difficulties with the centralized approach to metadata as well. Some of these difficulties are:

- most or all of the metadata in the enterprise must be accommodated and captured before any of the metadata is useful,
- the administrator of the centralized metadata must be taught the business of the department before the metadata becomes useful,
- the centralized metadata infrastructure cannot be built incrementally,
- it is very difficult for the centralized metadata to be kept up to date once captured, and so forth.

In a word, there are many problems with centralized metadata.

A variant form of the centralized metadata architecture is the centralized replicated form of architecture. In the centralized replicated form of metadata architecture, the metadata is gathered centrally, as in the case of the classical centralized metadata architecture. But once

gathered there, the metadata can be copied out to any other person or environment that requests. Once copied, the person or organization that has requested the metadata can alter or otherwise manipulate the metadata. There are no controls or conditions on the metadata once copied.

Yet another alternative is that of the distributed metadata architecture. In the distributed metadata mode, metadata resides independently at the many different locales in the DSS environment, such as at the data mart environment, the ODS environment, the enterprise data warehouse environment, and so forth. Metadata resides and is managed locally. This means that a data mart, for example, creates, updates, and deletes its own metadata. There is local control and ownership of metadata. However, the metadata that is owned and managed locally can be shared. Other corporate entities - other data marts, other ODS, other enterprise data warehouse, and so forth - can access the metadata as it is stored locally. In doing so, careful record is made of who the owner of the metadata is. If an organization is sharing metadata, it cannot alter the metadata that does not belong to it. The preservation of the rights of ownership of metadata is called the system of record across all participating corporate entities. The system of record is the backbone of the distributed metadata environment.

2.6 Introduction to Data mining

Large amount of data generated by organizations worldwide is mostly unorganized. If data is organized one can generate/extract meaningful and useful information to convert unorganized data into organized data. Normally the concept of DBMS is implemented though a database in management systems is embedded with a query language popularly known as SQL server. The use of SQL particularly in unorganized large databank is not always adequate to meat the end user requirements.

Data mining is the technique of abstracting meaningful information form large and unorganized databanks. It involves the process of performing automated abstraction and generating predictive information from large databanks. The abstraction of meaningful large databanks can also be known as knowledge discovery. The data mining process uses of varieties of analysis tools to determine the

relationship between data and the databank and to use the same to make valid prediction. Data mining techniques are a result of integration of various techniques forms multiple disciplines such as statistic, machine learning, pattern recognition, neural networks, image processing and so on.

Data Mining Process

The general phases in the data mining process to abstract knowledge are outlined as under

1. Problem definition: this phase is to understand the problem and the domain environment in which the problem occurs. You need to clearly define the problem before you proceed further. Problem definition specifies the limits within which the problem needs to be solved. It also specifies the cost limitations to solve the problem.
2. Creating a database for data mining: this phase is to create a database where the data to be mined are stored for knowledge acquisition. Creating a database does not require you to create a specialized database management system. You can even use a flat file or a spreadsheet to store data. Data warehouse is also a kind of data storage where large amount of data is stored for data mining. The creation of data mining database consumes about 50% to 90% of the overall data mining process.
3. Exploring the database: this phase is to select and examine important data sets of a data mining database in order to determine their feasibility to solve the problem. Exploring the database is a time-consuming process and requires a good user interface and computer system with good processing speed.

4. Preparation for creating a data mining model: this phase is to select variables to act as predictors. New variables are also built depending upon the existing variables along with defining the range of variables in order to support imprecise information.
5. Building a data-mining model: this phase is to create multiple data mining models and to select the best of these models. Building a data-mining model is an interactive process. At times, you need to go back to the problem definition phase in order to change the problem definition itself. The data-mining model that you select can be a decision tree, an artificial neural network, or an association rule model.
6. Evaluating the data-mining model: this phase is to evaluate the accuracy of the selected data-mining model. In data mining, the evaluating parameter is data accuracy in order to test the working of the model. This is because the information generated in the simulated environment varies from the external environment. The errors that occur during the evaluation phase needs to be recorded and the cost and time involved in rectifying the error needs to be estimated. External validation is also needs to be performed in order to check whether the selected model performs correctly when provided real world values.
7. Deploying the data-mining model: this phase is to deploy the built and the evaluated data-mining model in the external working environment. A monitoring system should monitor the working of the model and generate reports about its performance. The information in the report helps enhance the performance of selected data mining model.

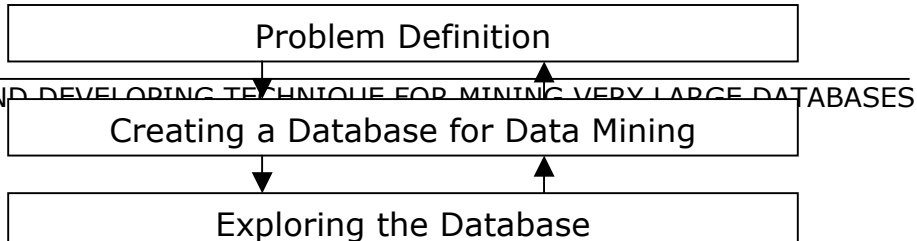


Fig. 2.3 General Phases of Data Mining Process

2.7 An Architecture for Data Mining

To best apply these advanced techniques, they must be fully integrated with a data warehouse as well as flexible interactive business analysis tools. Many data mining tools currently operate outside of the warehouse, requiring extra steps for extracting, importing, and analyzing the data. Furthermore, when new insights require operational implementation, integration with the warehouse simplifies the application of results from data mining. The resulting analytic data warehouse can be applied to improve business processes throughout the organization, in areas such as promotional campaign management, fraud detection, new product rollout, and so on. Figure 1 illustrates architecture for advanced analysis in a large data warehouse.

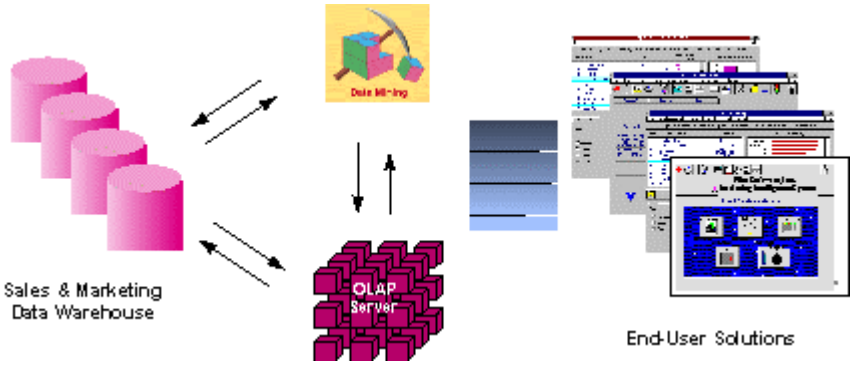


Fig. 2.4 - Integrated Data Mining Architecture

The ideal starting point is a data warehouse containing a combination of internal data tracking all customer contact coupled with external market data about competitor activity. Background information on potential customers also provides an excellent basis for prospecting. This warehouse can be implemented in a variety of relational database systems: Sybase, Oracle, Redbrick, and so on, and should be optimized for flexible and fast data access.

An OLAP (On-Line Analytical Processing) server enables a more sophisticated end-user business model to be applied when navigating the data warehouse. The multidimensional structures allow the user to analyze the data as they want to view their business – summarizing by product line, region, and other key perspectives of their business. The Data Mining Server must be integrated with the data warehouse and the OLAP server to embed ROI-focused business analysis directly into this infrastructure. An advanced, process-centric metadata template defines the data mining objectives for specific business issues like campaign management, prospecting, and promotion optimization. Integration with the data warehouse enables operational decisions to be directly implemented and tracked. As the

warehouse grows with new decisions and results, the organization can continually mine the best practices and apply them to future decisions.

This design represents a fundamental shift from conventional decision support systems. Rather than simply delivering data to the end user through query and reporting software, the Advanced Analysis Server applies users' business models directly to the warehouse and returns a proactive analysis of the most relevant information. These results enhance the metadata in the OLAP Server by providing a dynamic metadata layer that represents a distilled view of the data. Reporting, visualization, and other analysis tools can then be applied to plan future actions and confirm the impact of those plans.

2.8. Data Mining Techniques

Data mining techniques provide a way to use data mining tasks in order to predict solution sets for a problem and a level of confidence about the predicted solution interns of consistency of prediction and interns of frequency of correct predictions. Data mining techniques include:

- 2.8.1 Statistics
- 2.8.2 Machine learning
- 2.8.3 Decision Tree
- 2.8.4 Hidden Markov Model
- 2.8.5 Artificial Neural Network
- 2.8.6 Genetic Algorithms
- 2.8.7 Meta Learning
- 2.8.8 Association Rules

2.8.9 Clustering Techniques

2.8.1 Statistics can be used in various stages of the data mining

- Data cleansing stage
- Data collection & sampling stage
- Data analysis stage

2.8.2 Machine learning is a process capable of independently acquiring data and integrating that data to generate useful knowledge. The concept of machine learning is implemented by way of computing software system that act as human being who learns from experience, analyses the observations made and self-improves providing increased efficiency and effectiveness.

2.8.3 Decision Trees is a tree-shaped structure, which represents a predictive model. In a decision tree, each branch of the tree represents a classification question while the leaves of the tree represents the partition of the classified information. You need to follow the branches of a decision tree that reflects the data elements you have such that when you reach the bottom-most leaf of the tree, you have the answer to your question in that leaf node. Decision trees are generally suitable for tasks related to clustering and classification. It helps generate rules that can be used to explain the decision being taken. Here are some of the decision tree algorithms

- CART
- ID3
- C4.5
- CHAID
- Rainforest
- Approximate Methods
- CLOUDS

- BOAT

2.8.4 Hidden Markov Models is a model that enables you to predict future actions action to be taken in time series. The models provide the probability of a future event, when provided with the present and previous events. Hidden Markov models are constructed using the data collected from a series of events such that the series of present and past events enables you to determine future events to a certain degree. In simple terms, Hidden Markov models are a kind of time series model where the outcome is directly related to the time. One of the constraints of Hidden Markov models is that the future actions of the series of the events are calculated entirely by the present and past events.

2.8.5 Artificial Neural Networks are non-linear predictive model that uses the concept of learning and closely resembles the structure of the biological neural networks. In artificial neural network, a large set of historical data is trained or analyzed in order to predict the output of a particular future situation or a problem. You can use artificial neural network in a wide variety of applications such as the fraud detection in online credit card transactions, secret military operations, and for operation, which requires biological simulations.

2.8.6 Genetic Algorithms is a search algorithm that enables you to locate optimal binary strings by processing an initial random population of binary strings by performing operations such as artificial mutation, crossover, and selection. The process of genetic algorithm is in an analogy with the process of natural selection. You can use genetic algorithms to perform tasks such as clustering and association rules. If you have a certain set of sample data, then genetic algorithms enables you to determine the best possible model out of a set of models in order to represent the sample data. While

using the genetic algorithm technique, first you arbitrarily select a model to represent the sample data. Then you perform a set of iterations on the selected model to generate new models. Out of the many generated models, you select the most feasible model to represent the sample data using the defined fitness function.

2.8.7 Meta Learning is the concept of combining the predictions made from multiple models of data mining and analyzing those predictions to formulate a new and previously unknown prediction. The concept of Meta learning is most useful when the models that you use very widely and are very different form each other. You can predictions made by different data mining techniques as an input to a Meta learner, which would combine each of the predictions in order to create the best-possible model or solution. For example, if you use the predictions of a decision tree, an artificial neural network, and a genetic algorithm as an input to a neural Meta learner, it will try to implement the learning methodology to combine the input prediction to generate the maximum classification accuracy.

2.8.8 Association Rules

Association rules is an example of data mining task that fits in the descriptive model of data mining. The use of association rules enables you to establish association and relationships between large unclassified data items based on certain attributes and characteristics. Association rules define certain rules of associativity between data items and then use those rules to establish relationships.

Association rules are largely used by those organizations that are into retail sales business. For example, the study of purchasing patterns

of various consumers enables the retail business organizations to consolidate their market share by better planning and implementing the established association rules. You can also use association rules to describe the associativity of telecommunication failures, satellite link failures, or the overseas failure of the Indian cricket team with certain attributes. The result of these association rules can help prevent such failures by taking appropriate measures. Here are some of the Association Rules algorithms

- A Priori Algorithm
- Partition Algorithm
- Pincer-Search Algorithm
- Dynamic Item set Counting Algorithm
- FP-tree Growth Algorithm
- Incremental Algorithm
- Border Algorithm

2.8.9 Clustering Techniques

Clustering is an example of data mining task that fits in the descriptive model of data mining. The use of clustering enables you to create new groups and classes based on the study of patterns and relationship between values of data in a data bank. It is similar to classification but does not require you to predefine the groups or classes. Clustering technique is otherwise known as unsupervised learning or segmentation. All those data items that resembles more closely with each other are clubbed together in a single group, also known as clusters. Here are some of the Clustering Techniques algorithms

- K-Medoid Algorithms
- CLARA
- CLARANS

- DBSCAN
- BIRCH
- CURE
- STIRR
- ROCK
- CACTUS

2.9 Text Mining

Text format is a common and natural form of storing information. Text data are inherently unstructured and fuzzy. Thus text mining is more complicated process than general data mining. Domain of text Mining overlaps with several other fields, such as computational linguistics, and machine learning.

The study of the computerized applications and techniques, such as automatic machine translation and text analysis in processing and analyzing a language is commonly known as computational linguistics. In other words, computational linguistics is simply a branch of linguistics studies that applies computers for the research of linguistics. Text mining is widely used in computational linguistics.

A text-mining tool can be used to explore and analyze the content of textual documents and to visually display the extracted information with a graphical interface commonly known as a conceptual map or concept map, or sometimes document map. A concept map provides a close view of the material, representing the main concepts within the text and how they are related to each other. A content map also displays the conceptual structure of the extracted information. You can search for number of occurrences of concepts and their interrelations in the text documents. Text mining tools offers both quantifying and displaying the conceptual structure of a document set.

The steps to generate a conceptual map are:

1. Extracting keywords from the text indicating important concepts
2. Calculating the strength of associative relationships between the keywords using statistical formulae or computational mathematics.
3. Displaying the associative structures of keywords.

If two keywords appear closely in a text, the relationship between two words is considered close. For example, in a text of 10 sentences two keywords appear in the same sentence 7 times. Thus these keywords are closely related.

A concept map is visualized as a graph. In a concept map keywords are usually shown as a node. In the conceptual map, the keywords having stronger relationships are placed closer together.

Usage of Text Mining

Text mining tools, such as SAS Text miner and Leximancer are used for various text-mining tasks. For example, you are creating an index of keywords used in a book. You need to know number of occurrences of a particular term along with the page numbers. Using a text-mining tool you can count number of occurrences of keywords. Text mining can be used for:

- Text analyzing: Enable you to exploring large text data and analyzing it using charts and graphs.
- Creating indexes of documents: Enables you to prepare document index
- Detecting plagiarism: Enable you to detect plagiarism. You can explore the content of a book or a large text and check for plagiarism.

- Statistical analysis of documents: Provides you various statistics regarding texts. For example, you can use a text-mining tool to count number of occurrences of a keyword in a single or across several documents.
- Document mapping: Can read a large text for you and create a map of the document collection. This does not necessarily mean that a text-mining tool understands the details of a text as well as you do. Example of a document mapping tasks is arranging parts of texts under appropriate heading.
- Processing customer letters: Enables you to summarize and categorize the customer queries and complain letters according to types of query or complain.
- Building archives of electronic data: Indicates creating a digital library, or news achieves.

Text Mining Tasks

There are several text-mining tasks performed for analyzing text. The text mining tasks are

- Clustering
- Factor Analysis
- Text classification
- Text purification
- Text summarization
- Distributed storage and retrieval
- Find similar documents
- Find association between terms
- Find commonly occurring terms
- Answer queries about the document

2.10 Web Mining

Web mining is a specialized application of data mining. In simple words, Web mining is a technique to process information available on Web and search for useful data. Web mining enables you to discover Web pages, text documents, multimedia files, images and other types of resources from web. Pattern extraction is a Web mining process to monitor the original or uploaded web pages, extract information from them and generate matches of a specific pattern with necessary information specified by a user. The pattern extraction process enables you to efficiently surf and access data available on the Web:

Web mining is widely used in several fields. The various fields where Web mining is applied are:

- E-commerce
- Information filtering
- Fraud detection
- Plagiarism detection
- Education and research

In e-commerce, Web mining helps in generating user profiles by customizing the choice of users. For example, Web mining enables a user to search for an advertisement and information regarding a product of his interest. Internet advertising is one of the major fields in e-commerce, where Web mining is widely used. Advertising in a specific domain of an e-commerce Web site or a general Web site and is considered as one of major application area of Web mining.

Information filtering is the method to identify the most important results forms a list of discovered frequent set of data items for which you can make use of Web mining. Fraud detection can be performed using Web mining by maintaining a list of signatures of all the users. Web mining is also applied for plagiarism detection and research work.

The problems and difficulties in Web mining are:

- The size and span of Word Wide Web (WWW) is huge and very wide. Thus it takes time to explore such a large volume of data.
- The Web is widely distributed so that any break in communication links results in break in the Web mining process.
- The information available in the Web is highly dynamic and your Web mining tools need to locate a specific part of the available information.
- The online Web documents contain several unstructured and semi structure data.
- Interconnection of Web pages may create difficulties in Web mining. While searching for a topic, if you navigate between the linked Web sites referred to as hyperlink, you may fall in an infinite loop.
- Hidden Web resources cannot be located easily and are difficult targets for retrieving information.
- Scope to present customized data to individual users is limited.

Web Mining Tasks and Characteristics

The general techniques and algorithms of data mining are also applicable in Web mining Web mining tasks can be decomposed into four subtasks:

- Resource searching: Indicates the task of retrieving documents from the Web.
- Information selection: Denotes automatic extraction of information from the Web documents. Several Web mining tools such as Web Miner are available to perform this task.
- Generalization of patterns: Denotes automatic discovery of patterns across multiple web sites.
- Analysis of Web documents: Denotes validation and analysis of the extracted patterns.

Web Mining Issues

Several software's are available to perform the various tasks of Web mining. For example, you can download a complete Web site with its structure using software such as Teleport Pro, Back Street Browser and Grab-a-Site. This software is called offline browsers. Offline browsers enable you to download a copy of Web pages of a Web site and explore the Web pages offline. While browsing the downloaded Web pages offline, you need not worry about communication link and cost of using the Web.

An offline Web browser can save all the files of a Web site in a single folder. In this case lose the path information of a file. If there are two files with the same name, the previous file will be overwrite\ten. Otherwise, you can save all the files with the original directory structure. This is called saving a mirror of a Web site. In such a case, the home page of the Web site is saved at the topmost level of the hierarchy of the downloaded files and directories. The directory containing the home page of the Web site is called virtual root directory.

Various Web analyzing software are available to analyze the statistics regarding the range of factors of a Web site such as average number of Web pages viewed by the visitors. You can also analyze the behavior of the visitors of the Web site by using the various Web analyzing software. The visitor information is usually stored in a log file in the server of a Web site. The structure of a log file varies depending upon the type of Web server. The structure of log file of Internet information Server (IIS) is different form Apache Web server. The Web analyzing software read data regarding a Web site from the log file.

The technique to retrieve visitor based information fro Web servers based log files and apply this information to analyze data is know as Web log mining. For example, consider that a Web site consists of four Web pages. Whenever you visit the Web site, the home page is shown displayed by default. You can access the remaining three pages only by clicking the specific link provided in the home page.

There is two major types of log files used in Web log mining: access log and agent log files. An access log file maintains a list of all the Web pages that the visitors have requested. These Web pages includes the HTML files and their imbedded graphic images and any other associated files such as texts. An agent log file consists of information about the browser that was used to explore the various Web pages.

You can use a Web analyzing software to analyze, which the visitors have accessed Web page the most. Thus, you can verify the behavior

of the visitors that whether the visitors are more interested in chatting, or reading news or email checking. The various examples of Web analyzing software's are Nihuo Web Log Analyzer, One Stat, Webalizer and Click Tracks Analyzer. The Webalizer works on Linux platform while other Web analyzing software work on Windows platform.

Creating the maintaining a community-specific Web site is another important issue in Web mining. People with the same interest can form a community for cooperation and sharing of Web documents. For example, some universities and research institutes forms a community. In the Web site of each of the community, you can find hyperlinks to other members of the community. This helps users to navigate between different community-specific Web sites to share information.

In terms of mathematic, a community is a set of Web pages, where each page has multiple links to other members in the community, rather than links outside the community. You can visualize a community in terms of a graph. There can be two types of entities in a community: hub and authority. Hubs do not provide any information, rather hubs points to the sources of information. In words, hubs specify the location where you can search for information. Examples of a hub are directories and online yellow pages. A single person or a group of persons are authorized to provide information about the concerned subject. These authorized persons are also responsible to update the information at different intervals. A good hub points to numerous authorities. On the other

hand, several hubs link to a good authority. Following figure shows the graph representing a community:

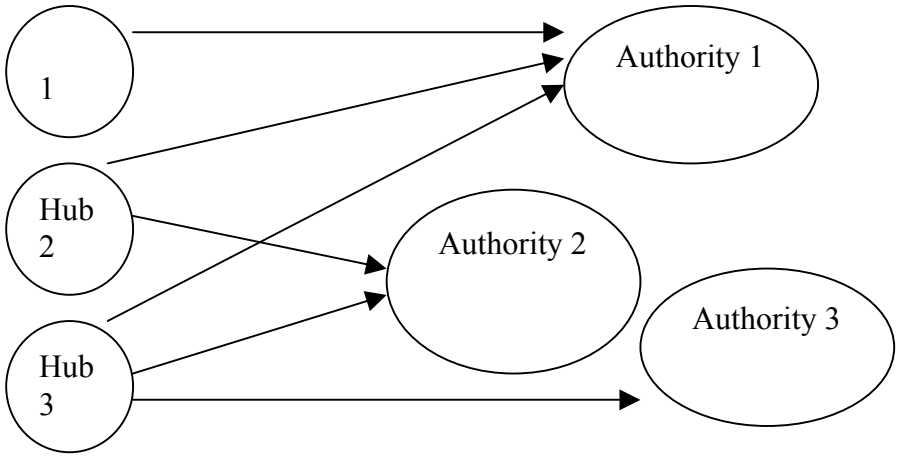


Fig. 2.5 Graph Representing a Community page

There are tow types of nodes in a community graph. A node represented as a small circle in the graph denotes a hub while a large circle in the graph represents an authority. An arrow indicates a link from a hub and an authority. For example, in above figure the arrow form Hub 2 to Authority 1 denotes that there exists a link between Hub 2 and Authority1. Hub 3 is the best hub simply because it has links to three authorities, while other hubs have less number of links to authorities. On the other hand, authority 1 is the best authority because three hubs are linked to it. No other authority is linked by three or more than three hubs.

Web Mining Software

- Synopses Summarizer
- Teleport Pro
- Click tracks

2.11 Spatial Data Mining

Spatial data mining is to mine high-level spatial information form large spatial databases. Examples of spatial data are images used in medical science, commonly known as medical imaging, images regarding design of electronic circuits, navigation charts, information regarding air traffic control, multiple image sequences such as video, and map of any location.

Any data is associated with a location is referred to as spatial data. You can plot spatial data in a co-ordinate system. For example, a medical image of an x-ray plate is basically a set of points plotted in a two dimensional graph. The coordinates indicate location of a point in the image. In a map, any specific location is identified by latitude and longitude.

Spatial Data Mining Overview

Spatial data mining means extracting undiscovered and implied spatial information, such as spatial patterns, which is not visibly stored in a spatial database. Spatial pattern matching is a process, where actual spatial data stored in a spatial database is extracted, analyzed and matches of a specific pattern or type of data specified by a user is generated.

Applications of Spatial Data Mining

Spatial data mining is applied in various fields for discovering unexplored information such as patterns, clusters, and classes from spatial databases. The nature of the spatial mining is so dynamic that

boundary of domains of application of spatial data mining overlap. Examples of the fields, where spatial data mining is used are:

- Geography
- Geology
- Medical imaging
- Robotics
- Video processing
- Navigation
- Traffic control

Geographical and geological data mining is a subset or part of spatial data mining. Geographical data mining is to mine geographical data such as maps. Geographic data mining is the process of exploring necessary and earlier unknown information from large spatial databases. Example of a spatial mining task in the field of geography and geology is analyzing a map that shows the earthquake prone areas. The further an area from the earthquake prone zone the safer it is.

Explosive increase in metrological data and the appearance of new spatial technologies highlight the need for the automated discovery of geographical and geological knowledge. Examples of information, which is explored in geographical and geological data mining, are census data, road maps, topographic maps, and maps regarding soil, climate, and ecology.

Spatial data mining is used in various analytical tasks of medical imaging. The study of graphical representation of information regarding medical science is known as medical imaging. Examples of

spatial data that is used in medical science is nerve signals from the human retina or skin. Example of application of spatial data mining in medical imaging is analyzing X-ray plates, images of brain scanning, and various graphs such as an audiogram, and output form a cardiograph.

Audiogram is basically a graph that shows a person's hearing ability, decided from certain medical examinations of hearing acuity of different sound frequencies. A cardiograph is a device, which continuously monitors cardiac output of a person.

Brain Workbench is software that applies spatial data mining in the field of medical imaging. Brain Workbench displays digital image of a brain and enables you to analyses various parts of brain.

Traditionally, analyzing brains involves a few sequential steps. Neuroscientists perform histological tests and analysis of a brain and cut the brain into hundreds to thousands of slices. The study of tissues of a living creature is called histology. Those slices are photographed and scanned into a sequence of two-dimensional images.

Assembly the two-dimensional image of the slices of a brain is very difficult, if you do it manually be arranging them in order they exist in the brain. Some images may be faulty and hazy. It is also difficult to mark the sequence of slices.

Brain Workbench automates the procedure of analyzing brain. Brain Workbench automatically builds a digital map of a brain and uses the

map for visualization, guiding neuroscience experiment, and also serving as a spatial database front end. Brain workbench uses compression techniques for storing high-resolution medical images.

There are four modules in Brain Workbench: the visualization module, the spatial and time series database engine, the relational database engine and the data-mining module. The visualization module enables you to locate and view any specific part of a brain. The spatial and time series database engine module and the relational database engine module enable you to store images and other information of brain in spatial and relational databases. The data-mining module enables you to apply mining the information stored in the databases used by Brain Workbench. Spatial data mining is also used in other fields such as video processing, navigation, traffic control and robotics.

Characteristics of Spatial Data

Spatial data are different from other sort of data such as plain text, and integers. Spatial data consists of cartographic information. The science and technique of creating maps is known as cartography. Spatial data usually contains the graphical elements such as lines, and arcs. For example, in a map showing running water supply of a city. A green line may be used to denote water supply lines.

In addition to graphical elements, there may exist some descriptive element that describes a feature. This descriptive element that describes a feature. These descriptive elements are commonly referred to as non-spatial data. For example, in a database, temperature of a city is stored in the form of text. If the temperature

is within the range 15º to 30º the text string, pleasant, is stored as temperature. If the temperature is below 15º, the text string, cold, is stored as temperature. If the temperature is above 30º, the text string, hot, is stored as temperature.

Any spatial information usually consists of three common types of elements:

- Points: Used for denoting discrete locations in a map and too tiny to be displayed as lines or areas. Points are stored as a single pair of X, and Y coordinates. This means there exists a coordinate system and every point is referred to as an address in the coordinate system.
- Lines, Arcs, and Curves: used for denoting objects with length but no area examples of the objects those are represented by lines and arcs are roads, rivers, property boundaries, railway lines, water supply lines, canals, and air transport route. There exists a scale which denotes the ratio between the length of an objects displayed in the map and their actual ratio. These objects are stored as a series of X, and Y coordinates.
- Areas and polygons: Used for denoting the objects that can be displayed as closed regions that represent the shape and location of homogeneous features. Example of the features that can be displayed as area and polygons are land, soil type, and buildings. Size of an areas or polygon is also scaled into spatial information.

In any spatial data such as a map, some features and characteristics are implicit. However, while storing these data electronically and

analyzing these data using a computer, you need to express these features explicitly. Features of any spatial data are:

- Position: Marked and referenced by a coordinate system.
- Spatial relationships: Indicates any relation between two cities. The cities are represented in a map as points. Cities are spatial objects here, and the distance between two cities is a spatial relationship between two spatial objects. Manually you can judge distance and proximity while viewing the map. But in a computer representation of a map, you must define the distance between two cities explicitly.
- Size and shape: Means that you can approximate the shape, region and boundary of a spatial object; but you must define size and shape with coordinate values, for spatial data mining.
- Legend: Denotes the interpretation of the symbols used in any spatial information. You need to define the meanings of the legends explicitly in computerized version of the spatial data.

Source of spatial data are

- Maps
- Aerial photographs
- Navigation charts
- Geographic Information System (GIS)
- Satellites images

A map is set of points, lines, and areas all defined both by positional reference to a coordinate system, usually latitude and longitude. Maps are stored and viewed as images. Several legends are used for marking any specific objects. For example, applying blue curves for

denoting rivers and streams, a thick black curve for denoting broad gauge railway line, and thin black curve for denoting meter gauge railway line is very common.

In each map, there exists a scale, which is basically the ratio of the distance between two points in the map and the actual distance between the places represented by these points. For example if the scale is 1:15000, this means if two places have been shown 1 unit after from each other, in reality these places are 15000 units afar.

Maps are used for:

- Providing any specific information such as weather, population and demography of a specific location. For example, a topological map of India displays that latitude of Delhi is 28º28” North and longitude of Delhi is 77º15” East.
- Providing general information about spatial patterns and comparing spatial patterns. For example, northeastern sates of India are less populated in comparisons with western states.

Aerial photographs are images taken in mid-air from any aircraft. Aerial photographs are commonly used to study the changes on the earth surface over time. Aerial photographs are useful for generating maps. Sometimes, aerial photographs contain distorted information and these are rectified before using them for spatial data mining.

A navigation chart is similar to a road map for the water that displays routes between ports. A compass symbol on the navigation chart shows north direction. Navigation charts also show depth of the water level. Navigation charts helps sailors to identify and locate several

objects, such as lighthouses. An air navigation chart shows the air routes.

Satellite images are photographs take form high in space by governmental satellites. Satellite image are commonly used for analyzing various weather related issues, such as movement of cloud. India has recently launched INSAT3E satellite for gathering information regarding various fields including telecommunications, television broadcasting, meteorological imaging, disaster warning and satellite-aided search and rescue services

There can be several types of spatial relationship between two spatial objects. In a query, you can combine two or more relationships using logical operators: AND, OR, and NOT. Relationships can be classified into four categories:

- Topological: Denotes positional relationships of spatial objects. For example, a spatial object, river P, stored as a curve, interests another spatial object, country Q, stored as region. This relationship is written in a query as river_P intersects country_Q.
- Distance: Indicates the distance between two spatial objects. Usually the shortest distance is counted. For example, (city_M and city_N) = 10 means, the distance between the spatial objects, city_M and city_N are 10 units.
- Direction: Indicates the direction between two spatial objects. Usually nine values to denote directions: east, north, west, south, south_east, south_west, north_east, north_west, and any direction. For example, city_M east city_N means, the spatial object city_M is situated east of the spatial object, city_N.

3.1 Data Warehousing Architecture

Systems in the data warehouse environment are not built under the system development life cycle (SDLC). There are two major components to building a data warehouse.

1. The design of the interface from operational systems.
2. Design of the data warehouse.

The requirements for the data warehouse cannot be known until it is partially populated and in use and design approaches that have worked in the past will not necessarily suffice in subsequent data warehouse. Data warehouse are constructed in a heuristics manner, where one phase of development depends entirely on the results attained in the previous phase.

3.1.1 Saurashtra University Model

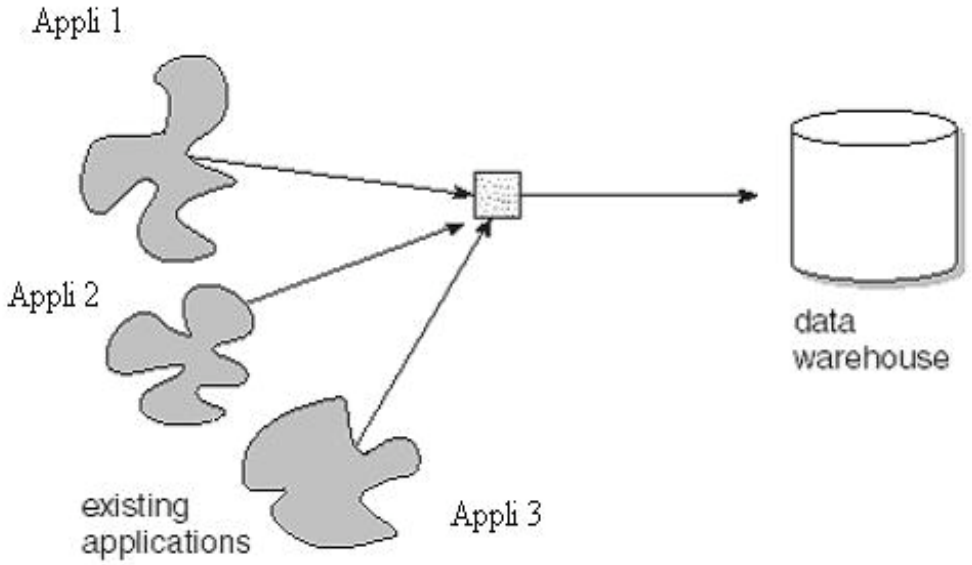


Fig. 3.1 Saurashtra University Model

As shown in Fig. 3.1, you can see that the there are many application exciting in Saurashtra University, and researcher want to put all the data to central location.

3.2. Requirement: Explain each module of Architecture

3.2.1 Beginning with operational data

Merely pulling data out of the legacy environment and placing it in the data warehouse achieves very little of the potential of data warehousing. When the existing applications were constructed, no thought was given to possible future integration. Each application had its own set of unique and private requirements. It is no surprise, then that some of the same data exist in various places with different names, some data is labeled the same way in different places, some data is all in the same place with the same way in different places, some data is all in the same place with the same name but different measurement and so on.

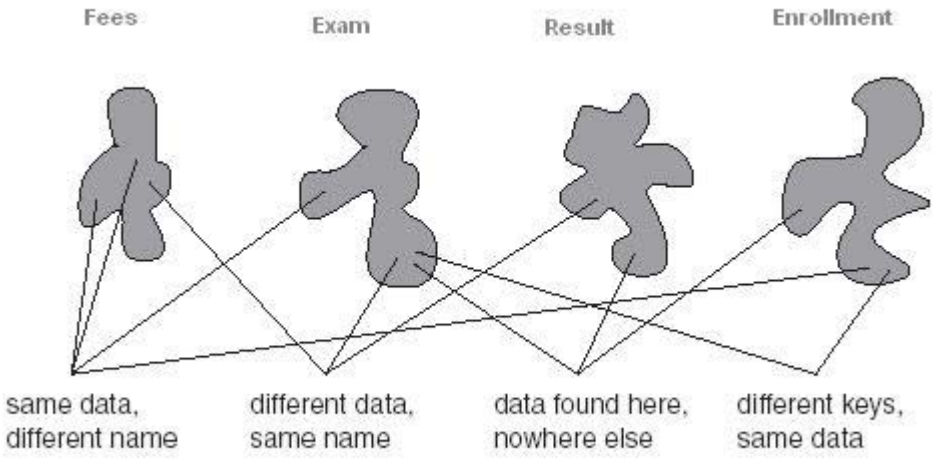


Fig. 3.2. Saurashtra University Application

In the above application it may be possible that

- In Fees and Exam same data have a different name.
- In Exam and Result different data have same name.
- Similarly Result and Enrollment have no common data.
- And many more possibility.

Yet another issue is that legacy data exists in many different formats under many different DBMSs. Some legacy data is under IMS, some legacy data is under DB2, and still other legacy data is under VSAM. But all of these technologies must have the data they protect brought forward into a single technology. Such a translation of technology is not always straightforward.

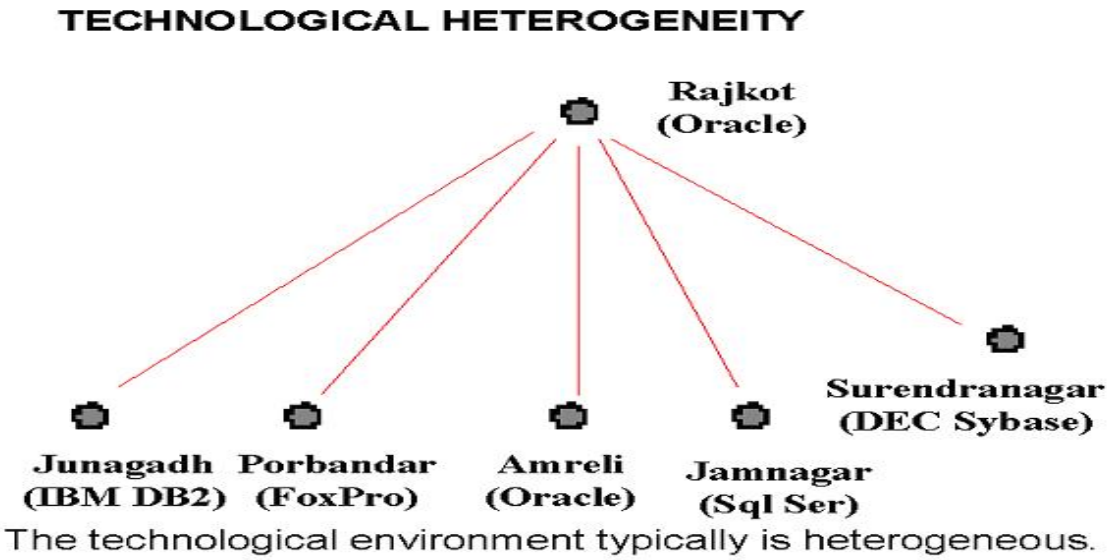


Fig. 3.3. Technological heterogeneity for Saurashtra University colleagues

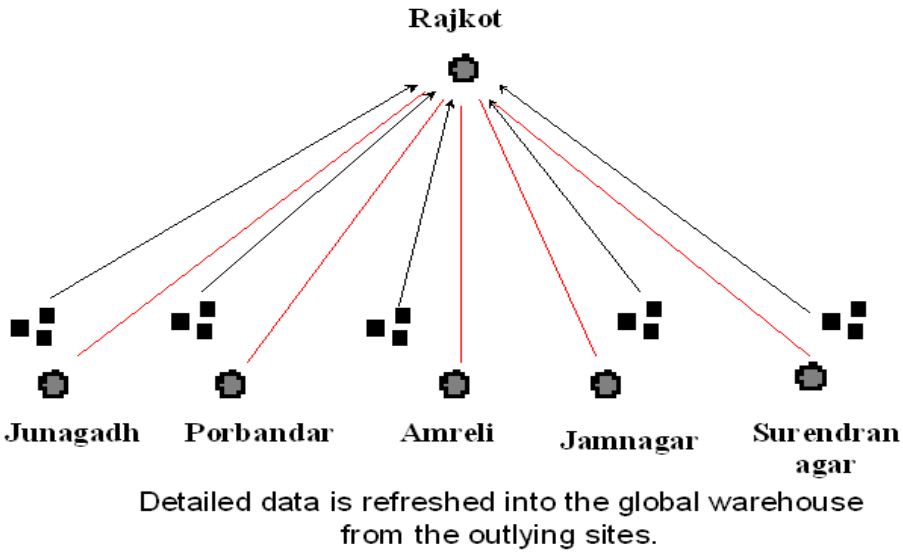


Fig. 3.4. Global Data Warehouse for Saurashtra University

There are five techniques, which are used when data warehouse is refreshed.

- Time stamped
- Delta files
- Log/audit files
- Modify application code
- Rubbing of 'before' & 'after' image of the operational file

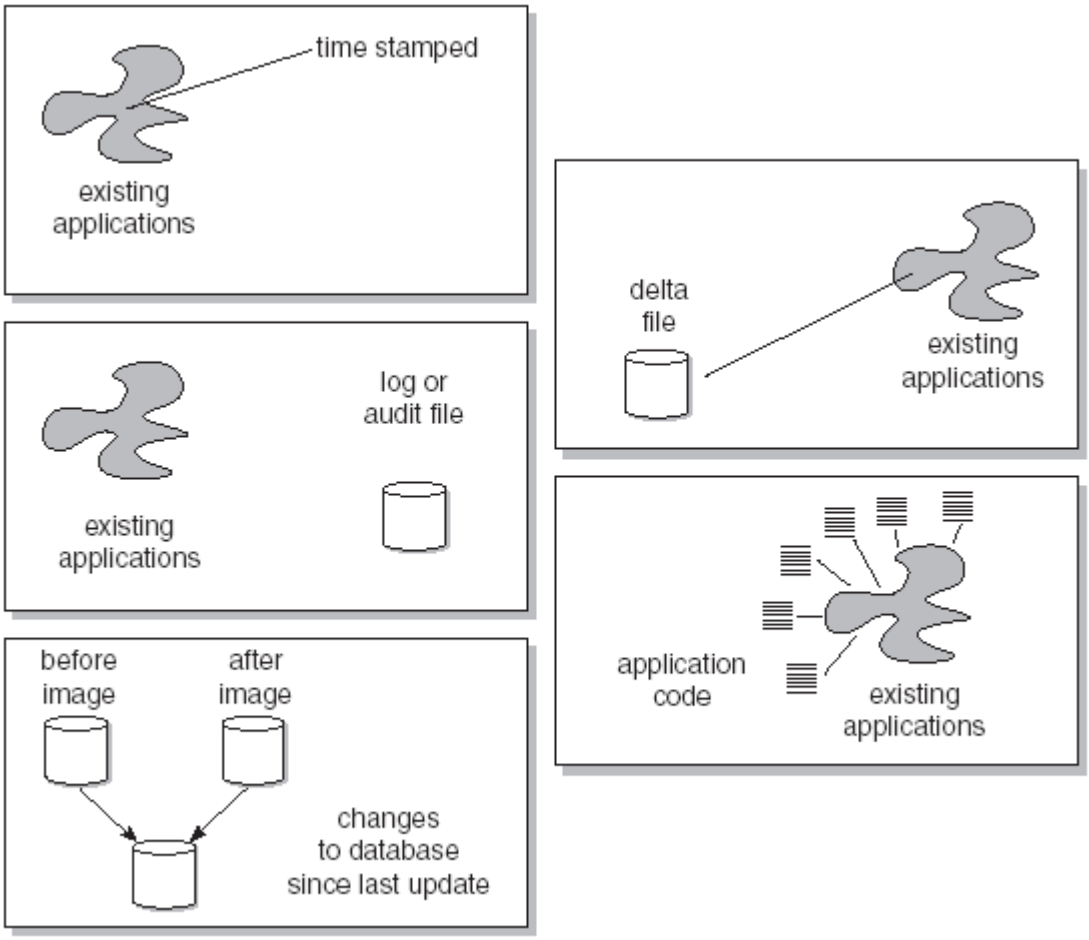


Fig. 3.5 Different Refreshing styles for data warehouse

3.2.2 Process models and architect environment

Before attempting to apply conventional database design techniques, the designer must understand the applicability and the limitations of those techniques. Figure 3.6 below shows the relationship among the levels of the architecture and the disciplines of process modeling and data modeling. The process model applies only to the operational environment. The data model applies to both the operational environment and the data warehouse environment. Trying to use a process or data model in the wrong place produces nothing but frustration.

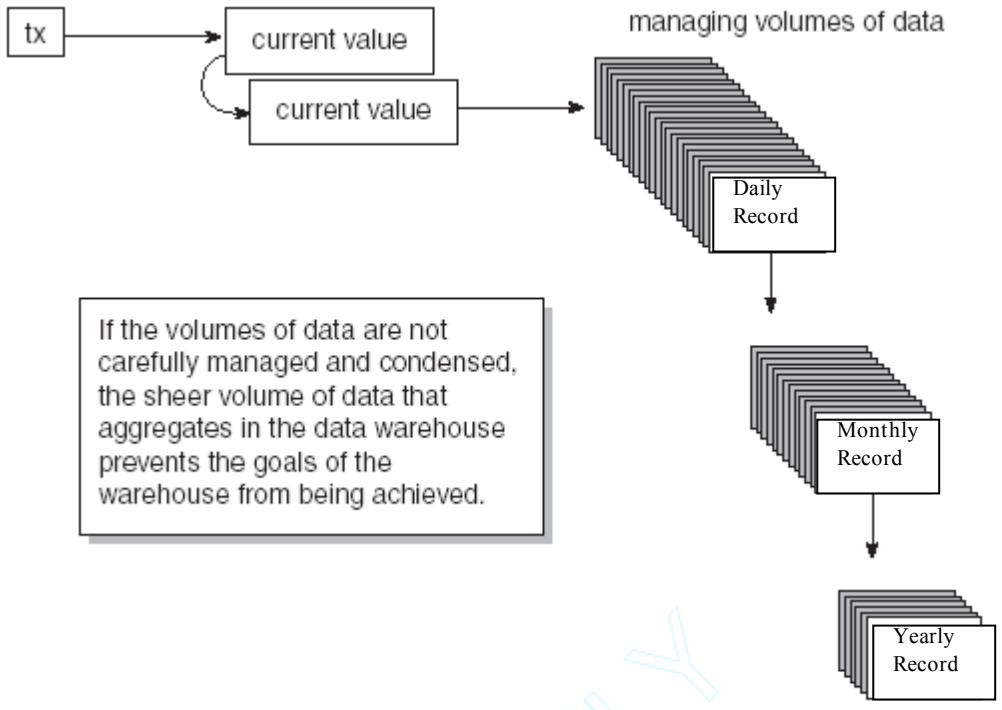


Fig. 3.6. Volume Data Management

A process model typically consists of the following (in whole or in part):

- Functional decomposition
- Context-level zero diagram
- data flow diagram
- Structure chart
- State transition diagram
- HIPO chart
- Pseudocode

There are many contexts and environments in which a process model is invaluable —for instance, when building the data mart. However, because the process model is requirements-based, it is not suitable for the data warehouse. The process model assumes that a set of known processing requirements exists—a priori—before the details of the design are established. With processes, such an assumption can be made. But those assumptions do not hold for the data warehouse. Many development tools, such as CASE tools, have the same orientation and as such are not applicable to the data warehouse environment.

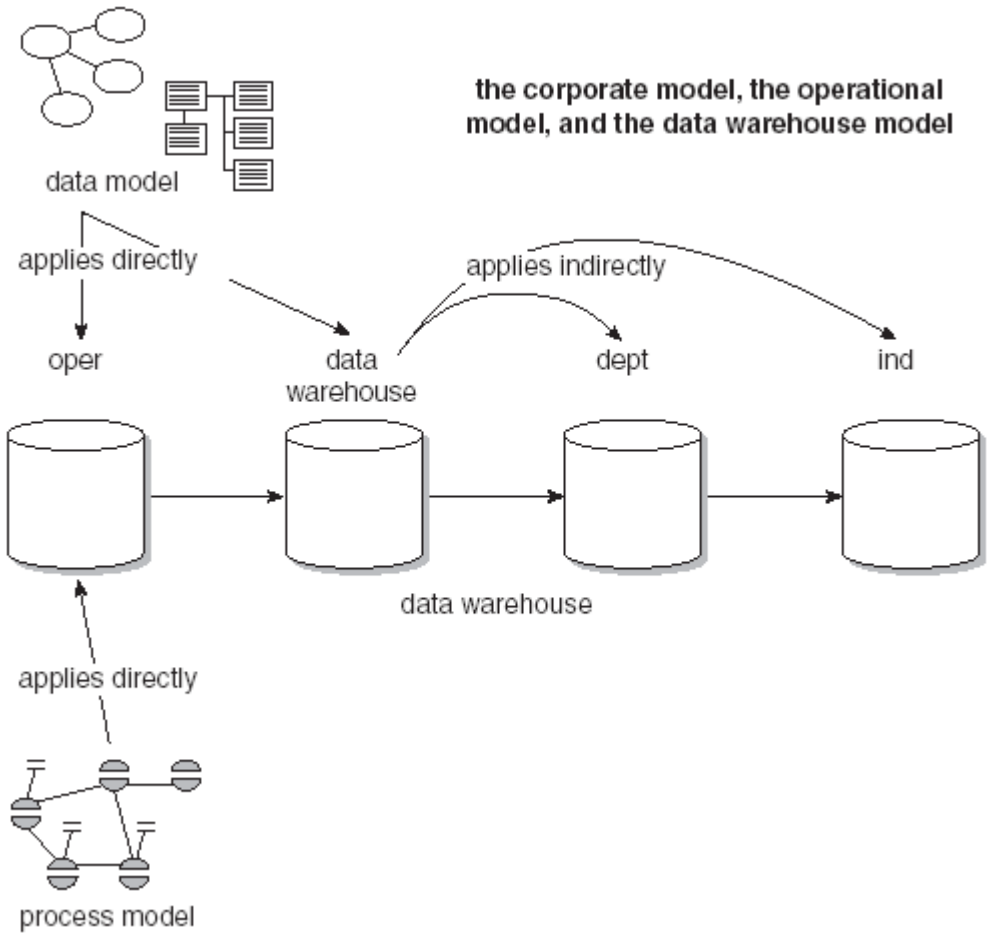


Fig. 3.7. Different Models for Data

3.2.3.1 A Data Model

As shown in below Figure, the data model is applicable to both the existing systems environment and the data warehouse environment. Here, an overall corporate data model has been constructed with no regard for a distinction between existing operational systems and the data warehouse. The corporate data model focuses on and represents only primitive data. To construct a separate existing data model, the beginning point is the corporate model, as shown. Performance factors are added into the corporate data model as the model is transported to the existing systems environment. All in all, very few

changes are made to the corporate data model as it is used operationally.

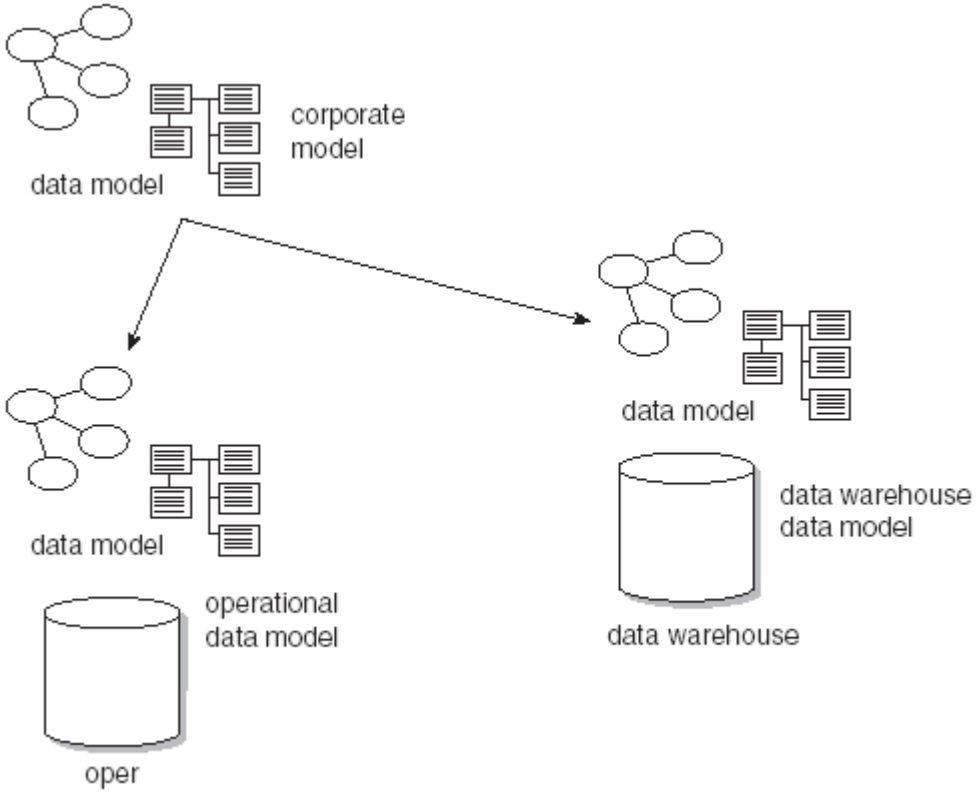


Fig. 3.8. Different Models for Data

However, a fair number of changes are made to the corporate data model as it is applied to the data warehouse. First, data that is used purely in the operational environment is removed. Next, the key structures of the corporate data model are enhanced with an element of time. Derived data is added to the corporate data model where the derived data is publicly used and calculated once, not repeatedly. Finally, data relationships in the operational environment are turned into "artifacts" in the data warehouse.

- Remove pure operational data
- Add element of time to key
- Add derived data where appropriate

- Create artifacts of relationships
- Operational data model equals corporate data model
- Performance factors are added prior to database design

There are three level of data modeling:

- High-level modeling (called the ERD, entity relationship level),
- Midlevel modeling (called the data item set, or DIS),
- Low-level modeling (called the physical model).

High-level modeling

The high level of modeling features entities and relationships, as shown in below Figure. The name of the entity is surrounded by an oval. Relationships among entities are depicted with arrows. The direction and number of the arrowheads indicate the cardinality of the relationship, and only direct relationships are indicated. In doing so, transitive dependencies are minimized.

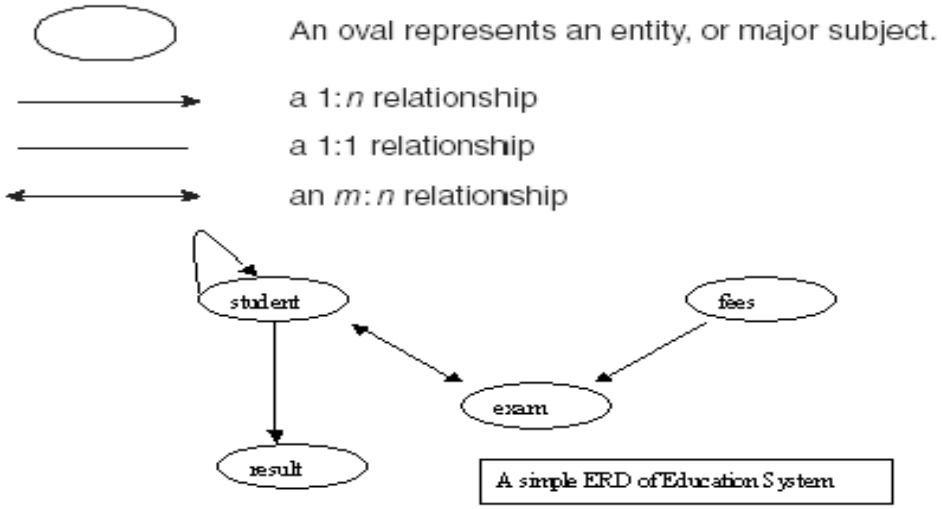


Fig. 3.9. ERD for Education System

The entities that are shown in the ERD level are at the highest level of abstraction. What entities belong in the scope of the model and what entities do not are determined by what is termed the "scope of integration,". The scope of integration defines the boundaries of the data model and needs to be defined before the modeling process commences. The modeler, the management, and the ultimate user of the system agree the scope on. If the scope is not predetermined, there is the great chance that the modeling process will continue forever. The definition of the scope of integration should be written in no more than five pages and in language understandable to the businessperson.

The ERDs representing the known requirements of the DSS community are created by means of user view sessions, which are interview sessions with the appropriate personnel in the various departments.

3.2.3.2 The Midlevel Data Model

After the high-level data model is created, the next level is established—the midlevel model, or the DIS. For each major subject area, or entity, identified in the high-level data model, a midlevel model is created, as seen in below Figure. The high-level data model has identified four entities, or major subject areas. Each area is subsequently developed into its own midlevel model.

Interestingly, only very rarely are all of the midlevel models developed at once. The midlevel data model for one major subject area is expanded, then a portion of the model is fleshed out while other parts remain static, and so forth.

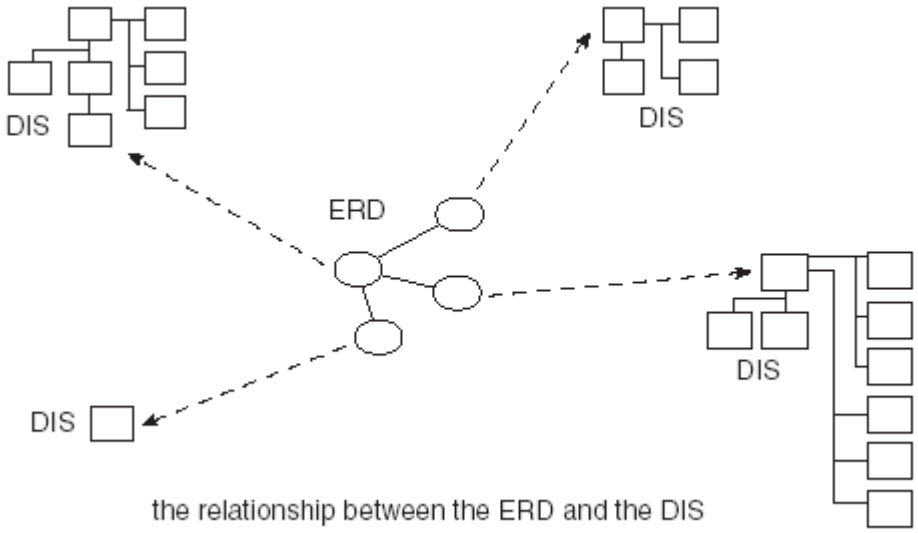


Fig. 3.10.

The four constructs that make up the midlevel data model

- A primary grouping of data
- A secondary grouping of data
- A connector, signifying the relationships of data between major subject areas
- "Type of" data.

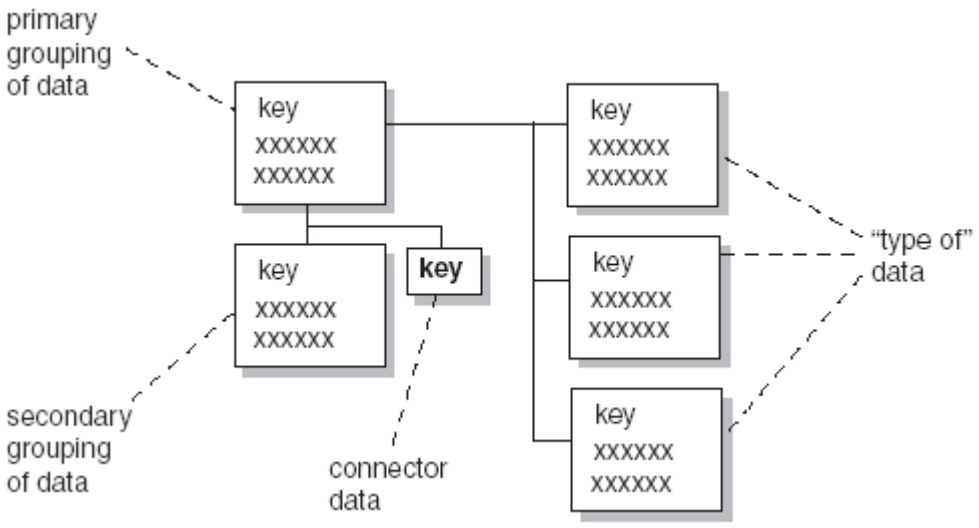


Fig. 3.11. Grouping Data

The primary grouping exists once, and only once, for each major subject area. It holds attributes that exist only once for each major subject area. As with all groupings of data, the primary grouping contains attributes and keys for each major subject area.

The secondary grouping holds data attributes that can exist multiple times for each major subject area. A line emanating downward from the primary grouping of data indicates this grouping. There may be as many secondary groupings as there are distinct groups of data that can occur multiple times.

The third construct is the connector. The connector relates data from one grouping to another. A relationship identified at the ERD level results in an acknowledgment at the DIS level. The convention used to indicate a connector is an underlining of a foreign key.

The fourth construct in the data model is "type of" data. "Type of" data is indicated by a line leading to the right of a grouping of data. The grouping of data to the left is the super type. The grouping of data to the right is the sub type of data.

3.2.3.3 The Physical Data Model

The physical data model is created from the midlevel data model merely by extending the midlevel data model to include keys and physical characteristics of the model. At this point, the physical data model looks like a series of tables, sometimes called relational tables. Although it is tempting to say that the tables are ready to be cast into the concrete of physical database design, one last design step remains—factoring in the performance characteristics. With the data warehouse, the first step in doing so is deciding on the granularity and partitioning of the data

After granularity and partitioning are factored in, a variety of other physical design activities are embedded into the design. At the heart of the physical design considerations is the usage of physical I/O (input/output). Physical I/O is the activity that brings data into the computer from storage or sends data to storage from the computer.

Data is transferred to and from the computer to storage in blocks. The I/O event is vital to performance because the transfer of data to and from storage to the computer occurs roughly two to three orders of magnitude slower than the speeds at which the computer runs. The computer runs internally in terms of nanosecond speed. Transfer of data to and from storage occurs in terms of milliseconds. Thus, physical I/O is the main impediment to performance.

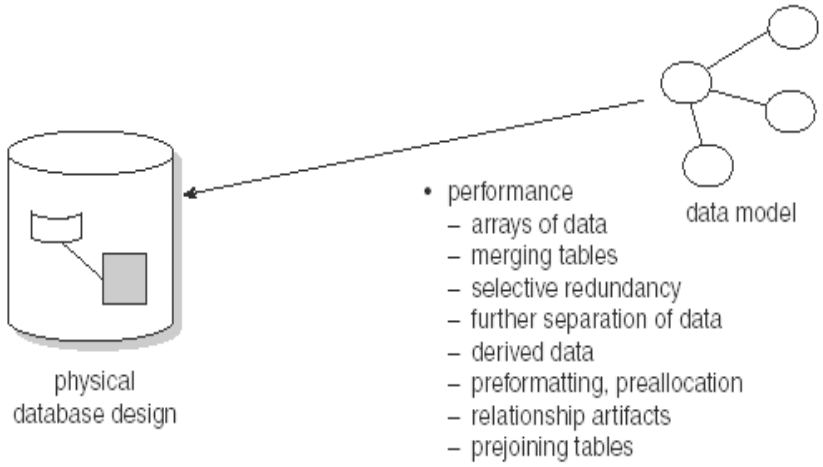


Fig. 3.12. Physical Database Design

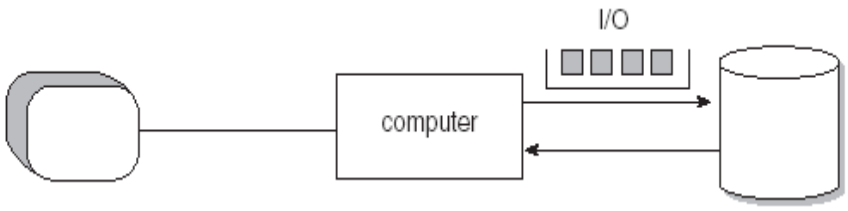


Fig. 3.13. Data Warehouse Interface

The job of the data warehouse designer is to organize data physically for the return of the maximum number of records from the execution of a physical I/O. (Note: This is not an issue of blindly transferring a large number of records from DASD to main storage; instead, it is a more sophisticated issue of transferring a bulk of records that have a high probability of being accessed.)

For example, suppose a programmer must fetch five records. If those records are organized into different blocks of data on storage, then five I/Os will be required. But if the designer can anticipate that the

records will be needed as a group and can physically juxtapose those records into the same block, then only one I/O will be required, thus making the program run much more efficiently.

There is another mitigating factor regarding physical placement of data in the data warehouse: Data in the warehouse normally is not updated. This frees the designer to use physical design techniques that otherwise would not be acceptable if it were regularly updated.

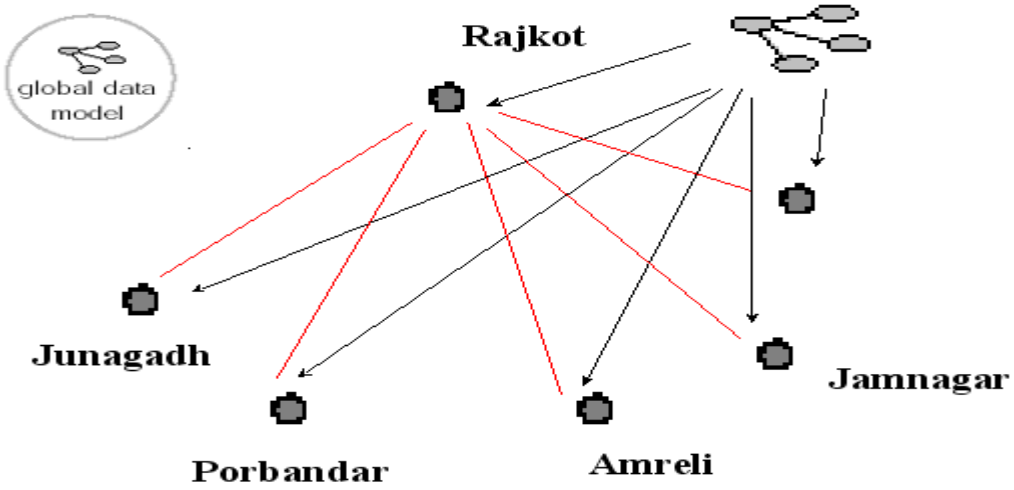


Fig. 3.14.

3.2.4 The Data Model and Iterative Development

In all cases, the data warehouse is best built iteratively. The following are some of the many reasons why iterative development is important:

- The industry track record of success strongly suggests it.
- The end user is unable to articulate requirements until the first iteration is done.

- ❖ Management will not make a full commitment until actual results are tangible and obvious.
- ❖ Visible results must be seen quickly.

What may not be obvious is the role of the data model in iterative development. To understand the role of the data model during this type of development, consider the typical iterative development suggested by Figure 3.15. First, one development effort is undertaken, then another, and so forth. The data warehouse serves as a roadmap for each of the development efforts, as seen in Figure 3.16.

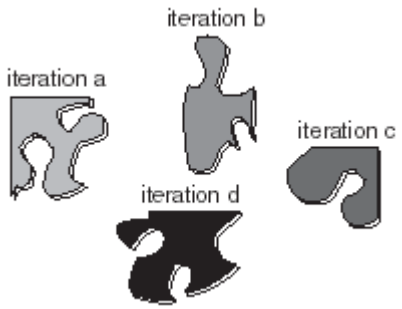


Fig 3 15 Iteration

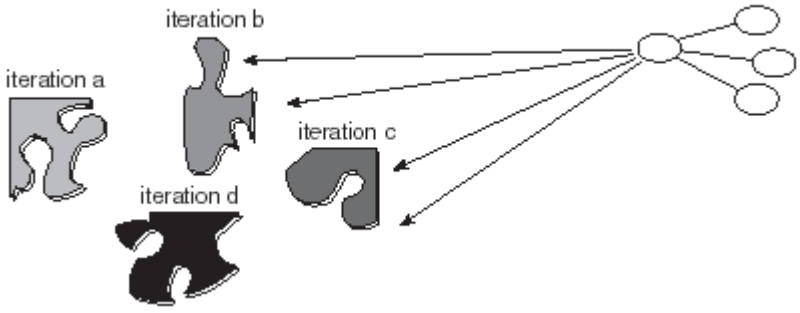


Fig 3 16 Iteration with Data Warehouse

When the second development effort ensues, the developer is confident that he or she will intersect his or her effort with the first development effort because all development efforts are being driven from the data model. Each succeeding development effort builds on the preceding one. The result is that the different development efforts are done under a unifying data model. And because they are built under a single data model, the individual iterative efforts produce a cohesive and tightly orchestrated whole, as seen in Figure 3.17.

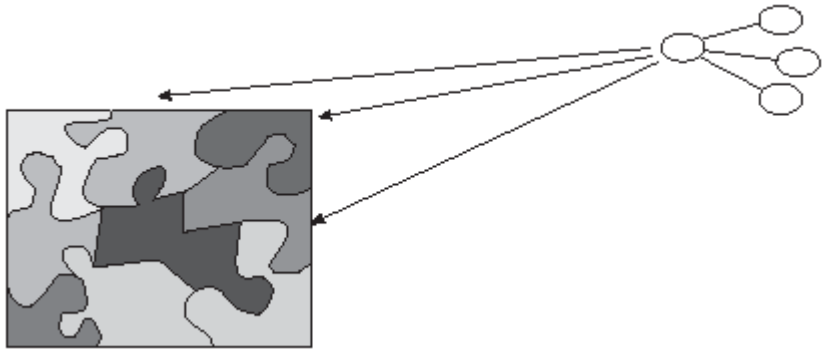


Fig 3 17 All Iteration in one group

When the different iterations of development are done with no unifying data model, there is much overlap of effort and much separate, disjoint development. Figure 3.18 suggests this cacophonous result. There is, then, an indirect, yet important, correlation between the data model and the ability to achieve long-term integration and a harmonious effort in the incremental and iterative development of a data warehouse.

Fig. 3.18 Iterative View

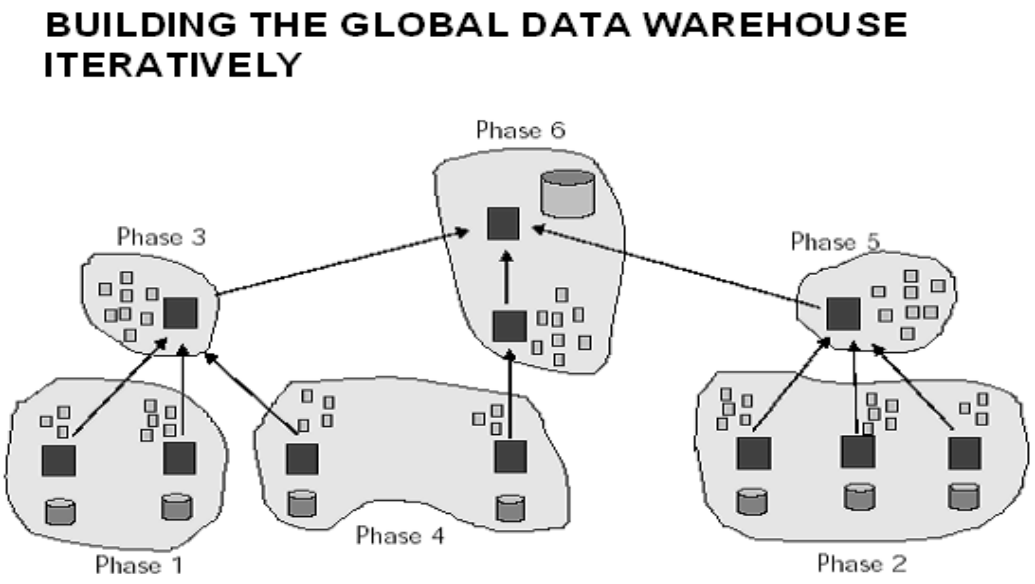


Fig. 3.19.

3.2.5 Metadata

Metadata is data about data. Metadata has been around as long as there have been programs and data that the programs operate on. Figure 1 shows metadata in a simple form.

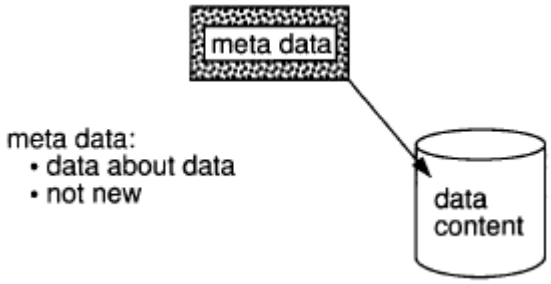


Fig. 3.20. Meta Data

While metadata is not new, the role of metadata and its importance in the face of the data warehouse certainly is new. For years the information technology professional has worked in the same environment as metadata, but in many ways has paid little attention to metadata. The information professional has spent a life dedicated to process and functional analysis, user requirements, maintenance, architectures, and the like. The role of metadata has been passive at best in this milieu.

But metadata plays a very different role in data warehouse. Relegating metadata to a backwater, passive role in the data warehouse environment is to defeat the purpose of data warehouse. Metadata plays a very active and important part in the data warehouse environment.

The reason why metadata plays such an important and active role in the data warehouse environment is apparent when contrasting the operational environment to the data warehouse environment insofar as the user community is concerned.

Figure U shows the difference in the communities served by data warehouse and operational systems.

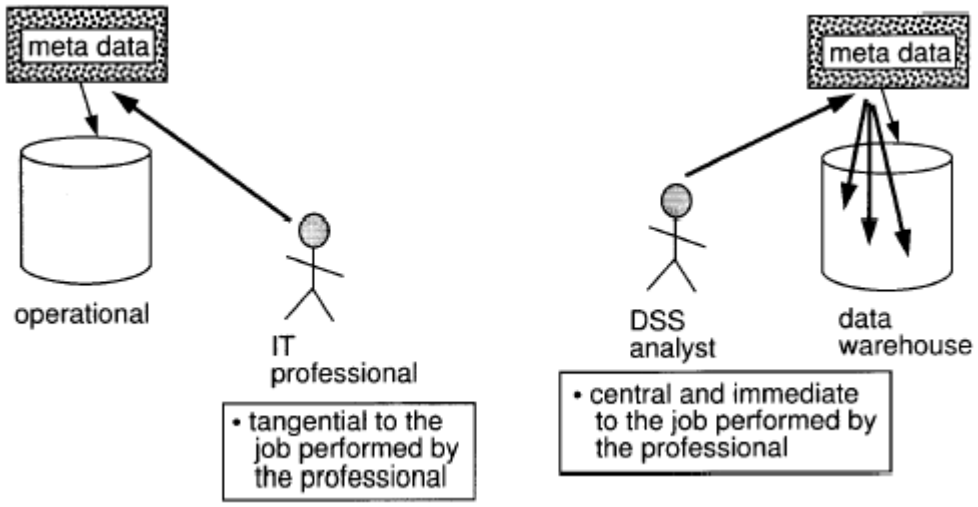


Fig. 3.21. Meta Data in Operation

The information technology professional is the primary community involved in the usage of operational development and maintenance facilities. It is expected that the information technology community is computer literate, and able to find his/her way around systems. The community served by the data warehouse is a very different community. The data warehouse serves the DSS analysis community. It is anticipated that the DSS analysis community is not computer literate. Instead the expectation is that the DSS analysis community is a businessperson community first, and a technology community second.

Simply from the standpoint of who needs help the most in terms of finding one's way around data and systems, it is assumed the DSS analysis community requires a much more formal and intensive level of support than the information technology community. For this

reason alone, the formal establishment of and ongoing support of metadata becomes important in the data warehouse environment.

But there is a secondary, yet important, reason why metadata plays an important role in the data warehouse environment. In the data warehouse environment, the first thing the DSS analyst needs to know in order to do his/her job is what data is available and where it is in the data warehouse. In other words, when the DSS analyst receives an assignment, the first thing the DSS analyst needs to know is what data there is that might be useful in fulfilling the assignment. To this end the metadata for the warehouse is vital to the preparatory work done by the DSS analyst.

Contrast the importance of the metadata to the DSS analyst to the importance of metadata to the information technology professional. The information technology professional has been doing his/her job for many years while treating metadata passively.

The community served and the importance to that community is the primary reasons why metadata is so important in the data warehouse environment. But there are other powerful reasons as well.

- **MAPPING**
- **MANAGING DATA OVER TIME**
- **VERSIONING OF DATA**
- **EXTRACT HISTORY**
- **SUMMARIZATION ALGORITHMS**
- **RELATIONSHIP ARTIFACTS**
- **RELATIONSHIP HISTORY**

- **OWNERSHIP/STEWARDSHIP**
- **ACCESS PATTERNS**
- **REFERENCE TABLES/ENCODED DATA**
- **DATA MODEL - DESIGN REFERENCE**

Here is a simple example of metadata for a university

Let us explain the role of metadata in the ETL process with the help of an example table shown below which contains information about an university students.

Student Name	Student Age	Branch	Class
ABC	20	BCom.	FY

In the above table, the second row, containing information like ABC, 20, BCom., Class are known as Data. Whereas the first row, (i.e.) table header containing headings like Student Name, Student Age, Branch, Class are called as **Metadata** for the above said data.

3.2.6 Granularity in the Data Warehouse

The single most important design issue facing the data warehouse developer is determining the granularity. When the granularity is properly set, the remaining aspects of design and implementation flow smoothly; when it is not properly set, every other aspect is awkward.

Granularity is also important to the warehouse architect because it affects all of the environments that depend on the warehouse for data. Granularity affects how efficiently data can be shipped to the different environments determine the types of analysis that can be done.

The primary issue of granularity is that of getting it at the right level. The level of granularity needs to be neither too high nor too low. The trade-off in choosing the right levels of granularity centers around managing the volume of data and storing data at too high a level of granularity, to the point that detailed data is so voluminous that it is unusable. In addition, if there is to be a truly large amount of data, consideration must be given to putting the inactive portion of the data into overflow storage.

Choosing the proper levels of granularity for the architected environment is vital to success. The normal way the levels of granularity are chosen is to use common sense, create a small part of the warehouse, and let the user access the data. Then listen very carefully to the user, take the feedback he or she gives, and adjust the levels of granularity appropriately.

The worst stance that can be taken is to design all the levels of granularity a priori, then build the data warehouse. Even in the best of circumstances, if 50 percent of the design is done correctly, the design is a good one. The nature of the data warehouse environment is such that the DSS analyst cannot envision what is really needed until he or she actually sees the reports.

The process of granularity design begins with a raw estimate of how large the warehouse will be on the one-year and the five-year horizon. Once the raw estimate is made, then the estimate tells the designer just how fine the granularity should be. In addition, the estimate tells whether overflow storage should be considered.

There is an important feedback loop for the data warehouse environment. Upon building the data warehouse's first iteration, the data architect listens very carefully to the feedback from the end user. Adjustments are made based on the user's input.

Another important consideration is the levels of granularity needed by the different architectural components that will be fed from the data warehouse. When data goes into overflow—away from disk storage to a form of alternate storage—the granularity can be as low as desired. When overflow storage is not used, the designer will be constrained in the selection of the level of granularity when there is a significant amount of data.

For overflow storage to operate properly two pieces of software are necessary —a cross-media storage manager that manages the traffic to and from the disk environment to the alternate storage

environment and an activity monitor. The activity monitor is needed to determine what data should be in overflow and what data should be on disk.

3.2.7 Different Approach to develop the Data Warehouse

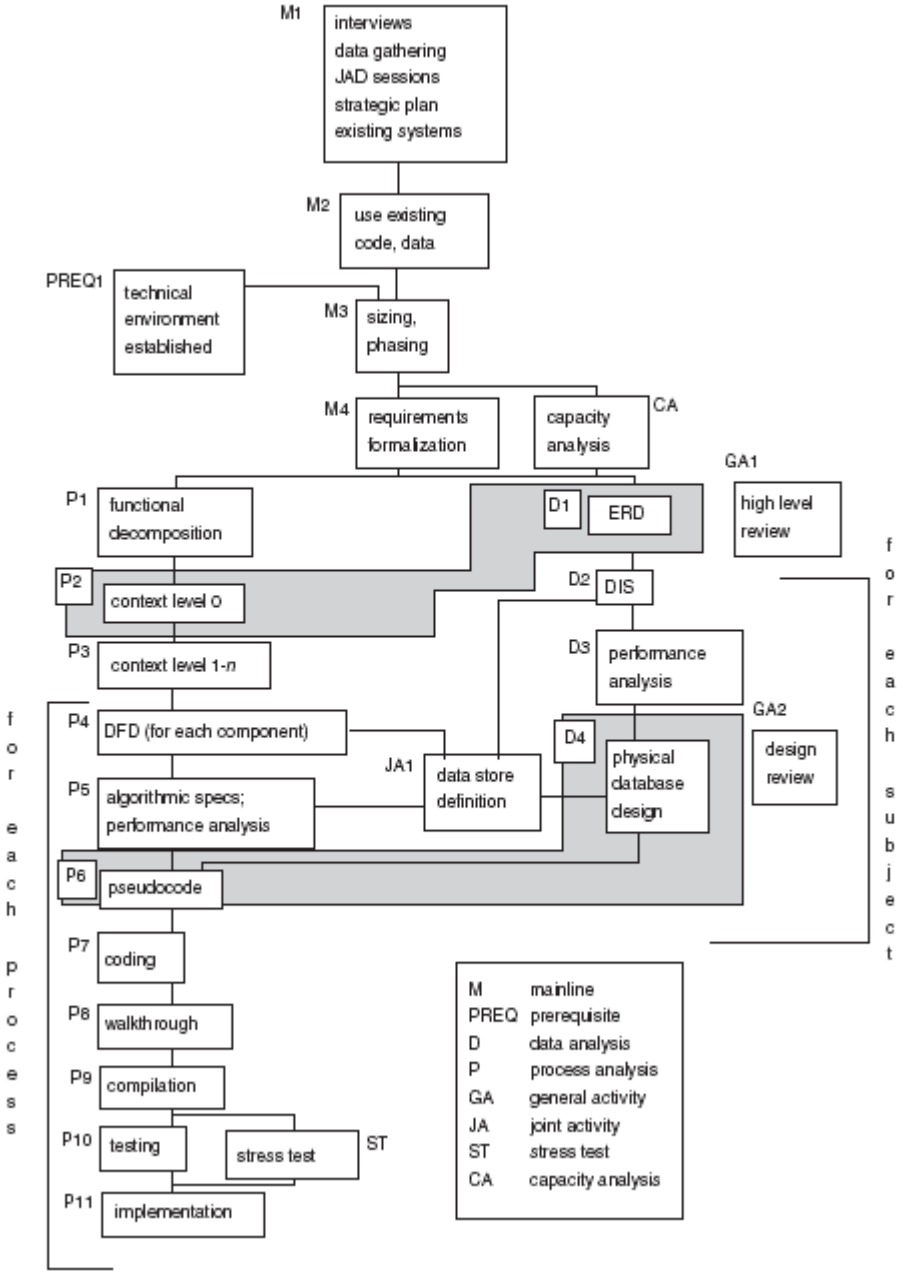


Fig. 3.22.

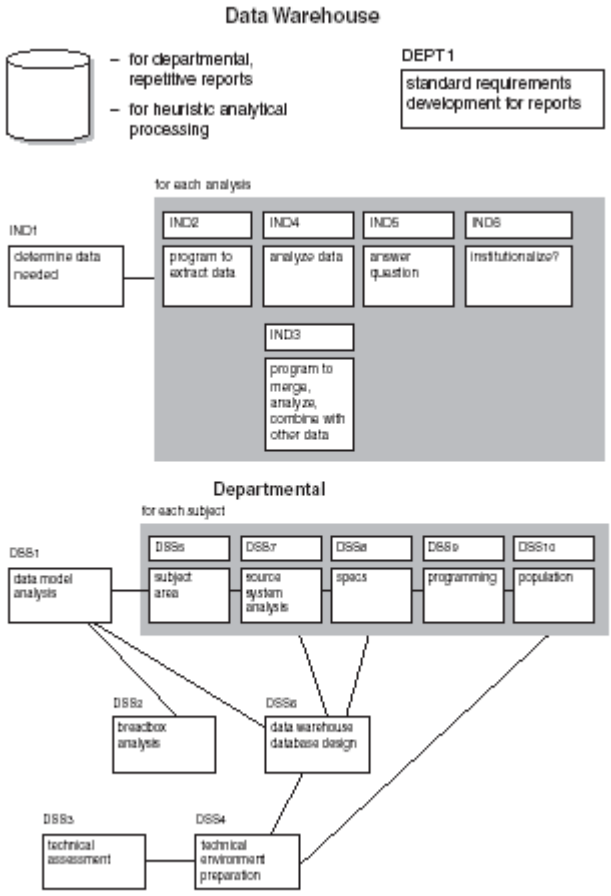


Fig. 3.23.

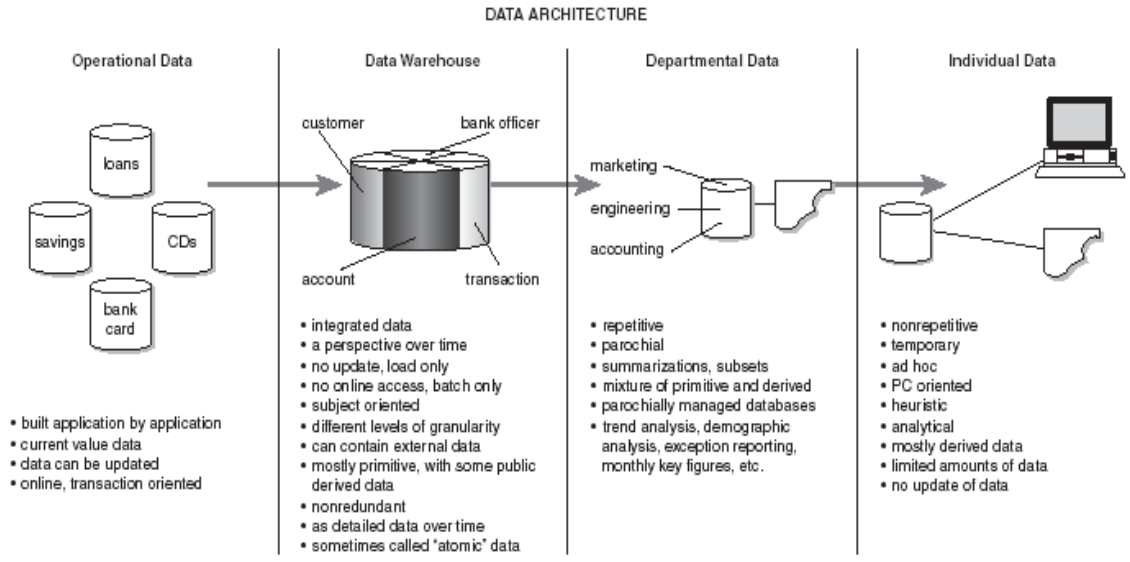


Fig. 3.24.

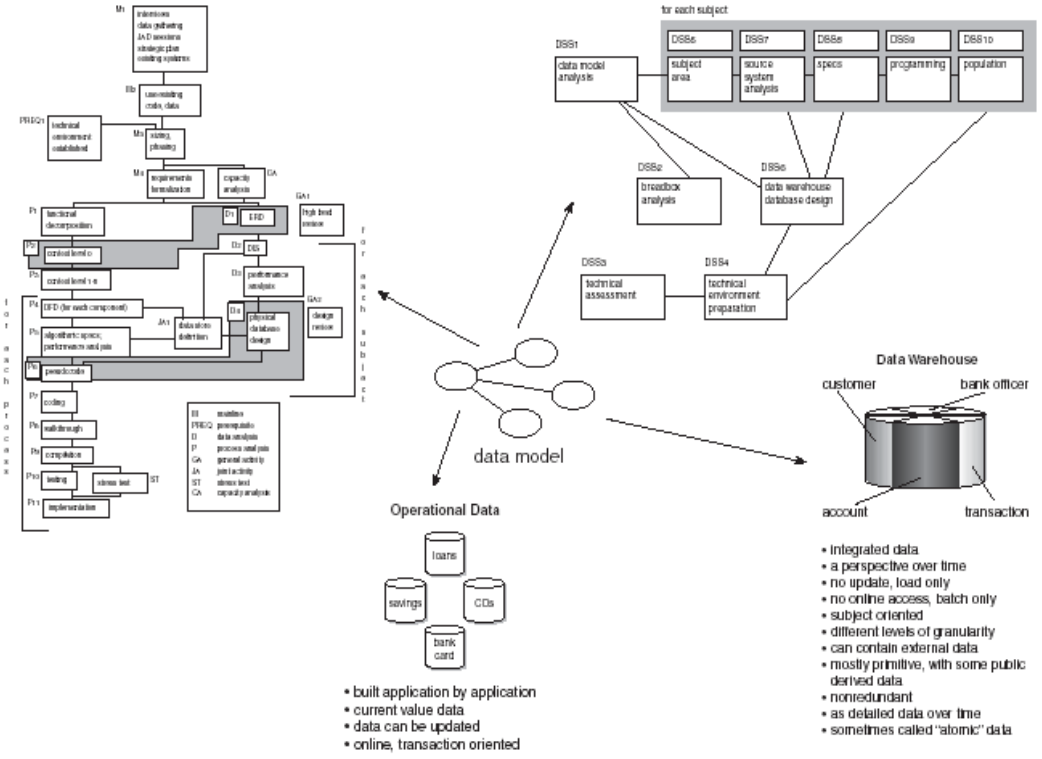

for each subject
DSS1
data model analysis
DSS6
DSS7
DSS8
DSS9
DSS10
subject area
source system analysis
specs
programming
population
DSS2
breadbox analysis
DSS5
data warehouse database design
DSS3
technical assessment
DSS4
technical environment preparation
data model
Operational Data
loans
savings
CDs
bank card
• built application by application
• current value data
• data can be updated
• online, transaction oriented
Data Warehouse
customer
bank officer
account
transaction
• integrated data
• a perspective over time
• no update, load only
• no online access, batch only
• subject oriented
• different levels of granularity
• can contain external data
• mostly primitive, with some public derived data
• nonredundant
• as detailed data over time
• sometimes called "atomic" data

Fig. 3.25.

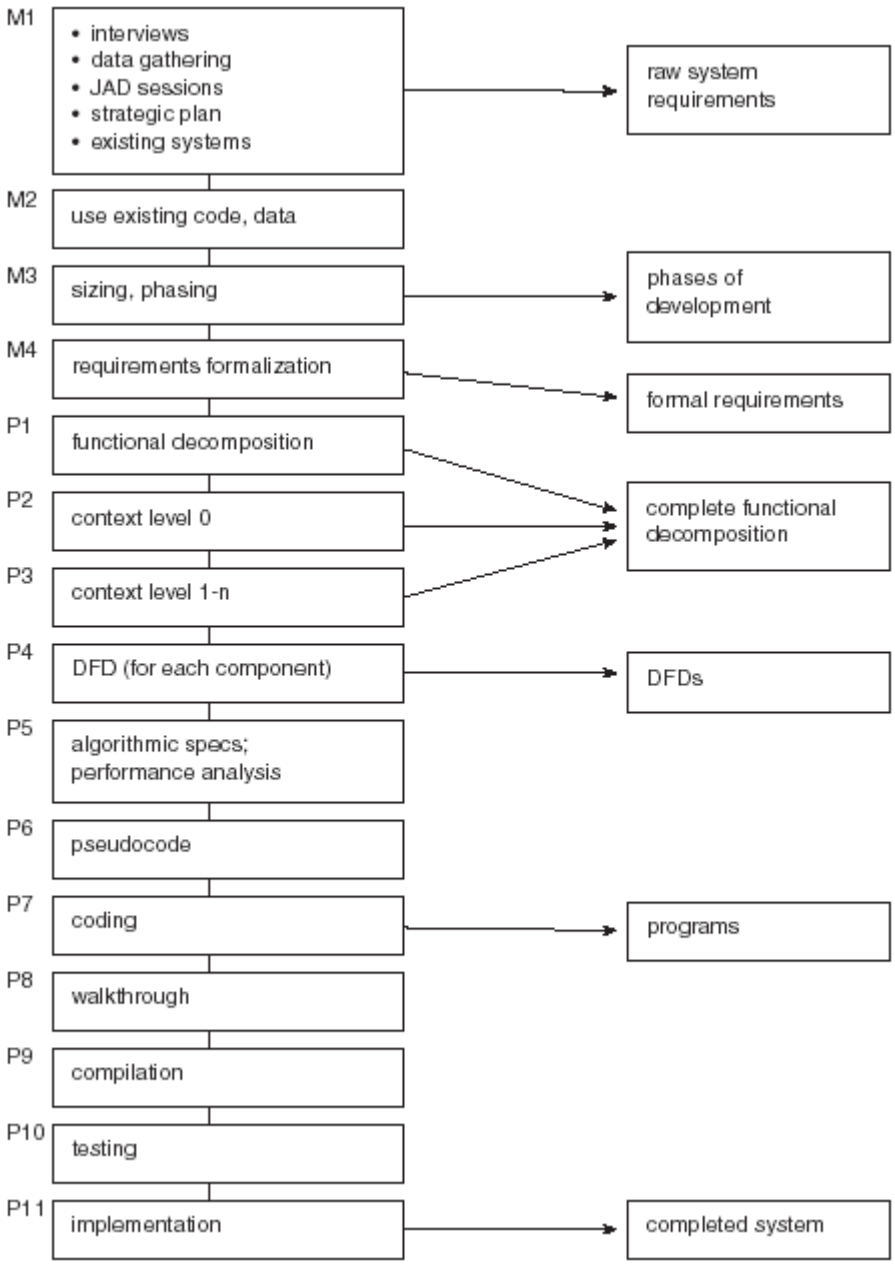

M1
• interviews
• data gathering
• JAD sessions
• strategic plan
• existing systems
raw system requirements
M2
use existing code, data
M3
sizing, phasing
phases of development
M4
requirements formalization
formal requirements
P1
functional decomposition
P2
context level 0
P3
context level 1-n
complete functional decomposition
P4
DFD (for each component)
DFDs
P5
algorithmic specs; performance analysis
P6
pseudocode
P7
coding
programs
P8
walkthrough
P9
compilation
P10
testing
P11
implementation
completed system

Fig. 3.26.

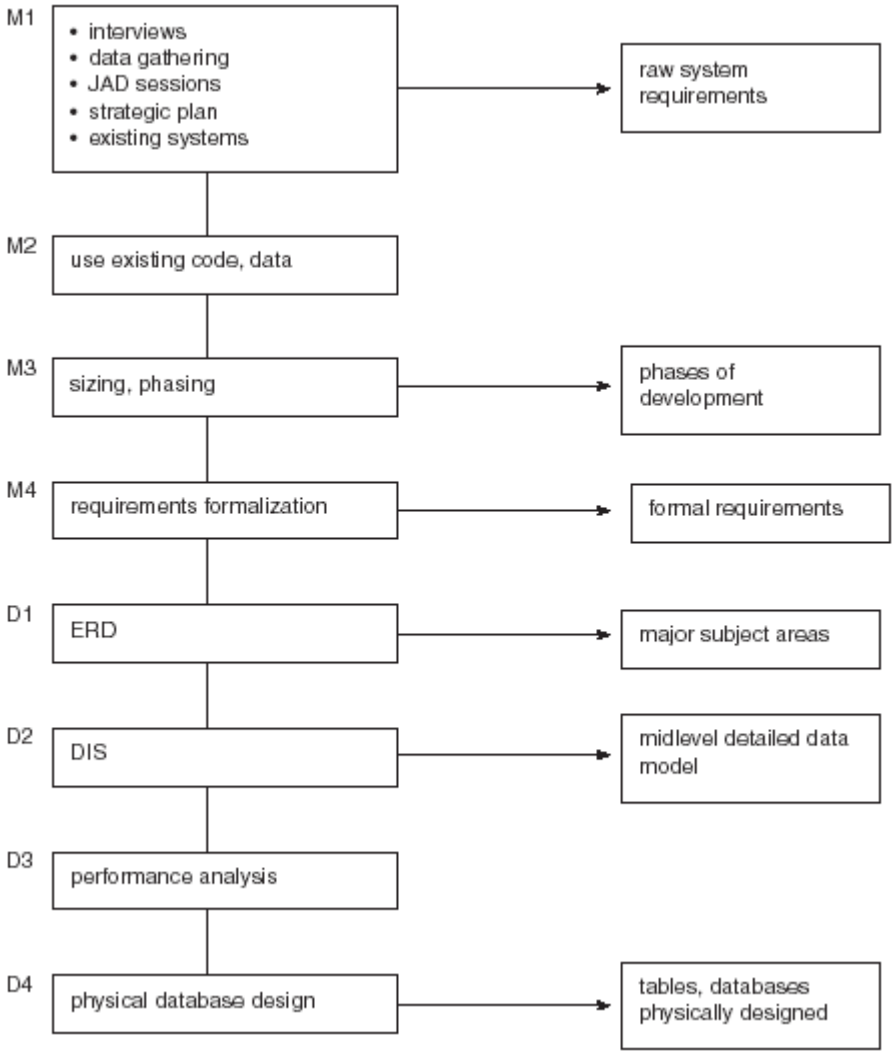

M1
• interviews
• data gathering
• JAD sessions
• strategic plan
• existing systems
raw system requirements
M2
use existing code, data
M3
sizing, phasing
phases of development
M4
requirements formalization
formal requirements
D1
ERD
major subject areas
D2
DIS
midlevel detailed data model
D3
performance analysis
D4
physical database design
tables, databases physically designed

Fig. 3.27.

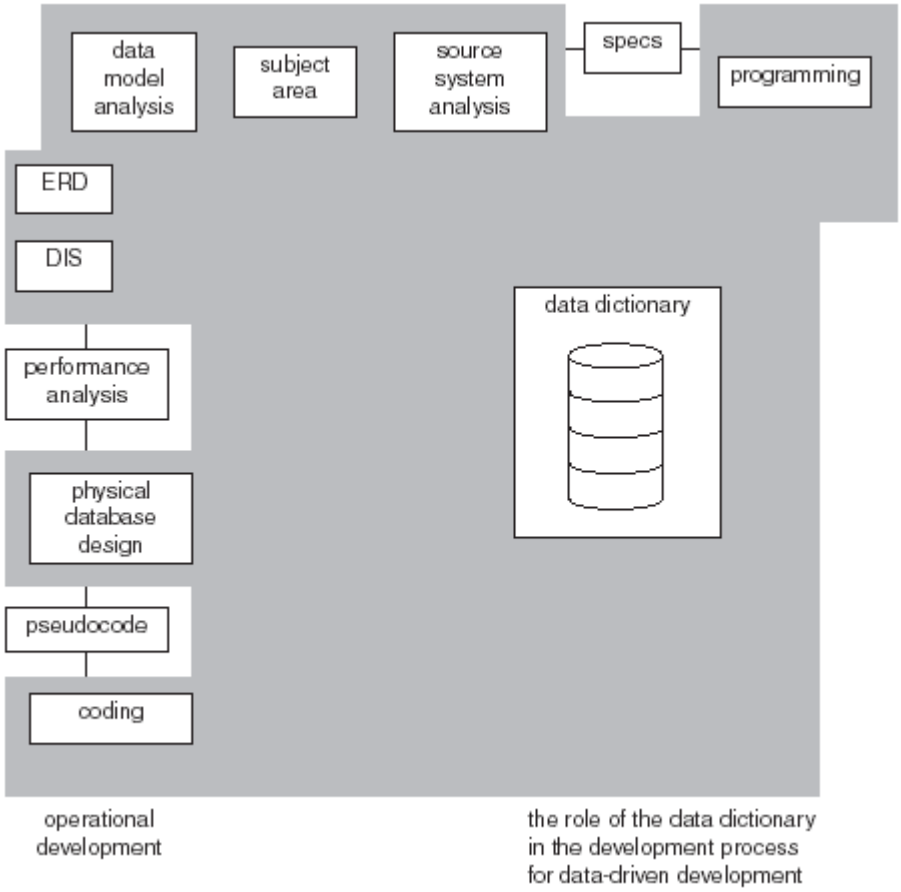


Fig. 3.28.

Fig. 3.22 to Fig. 3.28 shows the different data warehouse architecture can be used to develop a data warehouse solution.

3.2.8 A case study outcomes

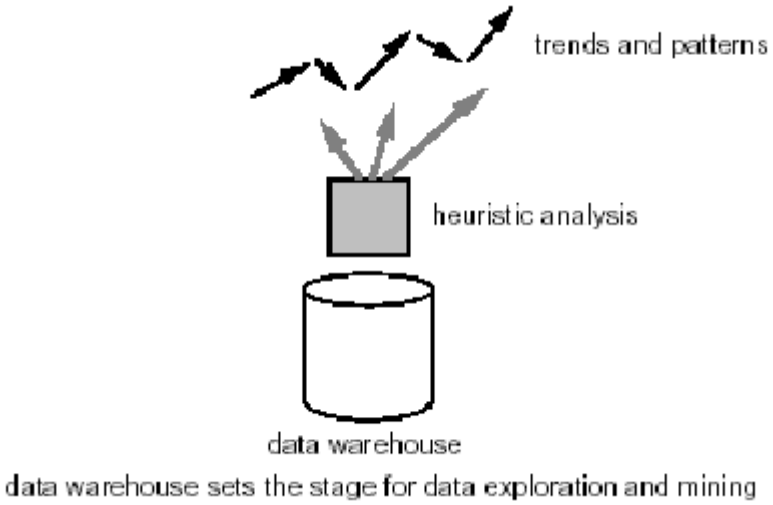


Fig. 3.29.

Out comes from the System

- Discovering the stages and status of teaching and research work undertaken by faculty members.
- To access information regarding learning environment for student and optimize for specific needs.
- Status of allotment of work, monitoring of allotted work, follow up of work for increasing effectivity and productivity.
- To optimize time for conduct of exam of university.
- To monitor activities of departments teaching, research and administration.
- To promote collaborative work and establish communication to increase effectiveness of collaborative work.
- The data keeping and data mining will generate data that will assist for quality maintains, quality improvement leads to

assessment for ISO 9000 and accreditation for NAAC, NBA (AICTE).

- Account intelligence.
- Budgeting analysis and setting budget target.
- Budget monitoring on time scale.
- Student feedback data warehousing and generating intelligent conclusion.
- Creating syllabus information for course program for individual subject and deriving conclusion for accommodation of required subject based on analysis and to certain extend course contain detail. This approach will help to update curriculum keeping pace with emerging technology and quick implementation by industry.

4.1 Steps for Data Mining

Data mining is often described as "the process of extracting valid, authentic, and actionable information from large databases." In other words, data mining derives patterns and trends that exist in data. These patterns and trends can be collected together defined as a mining model. The mining models can be applied to specific business scenarios, such as:

- Forecasting sales
- Determining which products are likely to be sold together
- Finding sequences in the order that customers add products to a shopping cart
- To get some rule for future direction

An important concept is that building a mining model is part of a larger process that includes everything from defining the basic problem the model will solve, to deploying it into a working environment. This process can be defined using the following six basic steps:

1. Defining the Problem
2. Preparing Data
3. Exploring Data
4. Building Models

5. Exploring and Validating Models

6. Deploying and Updating Models

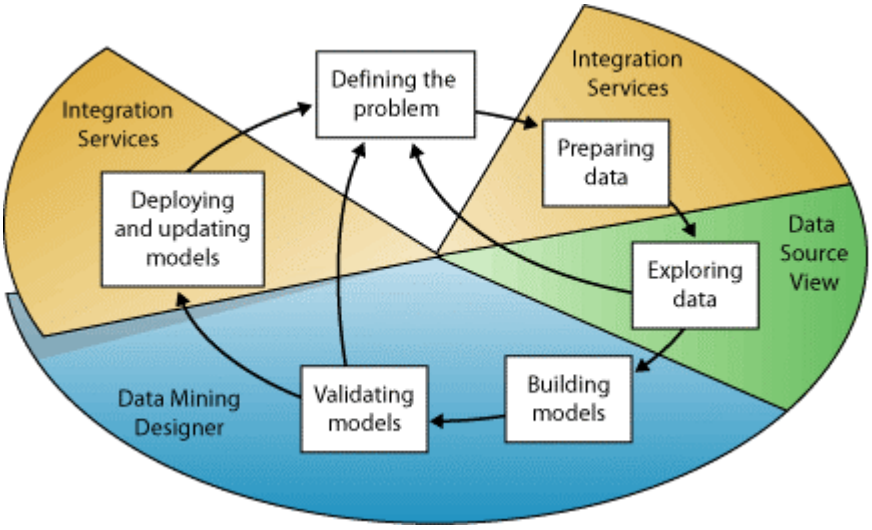


Fig. 4.1

Though the process displayed in the diagram is circular, each step does not necessarily directly lead to the next. Creating a data mining model is a dynamic and iterative process. After exploring the data, you may find that the data is insufficient to create the appropriate mining models, and thus need to look for more data. You may build several models and realize that they do not answer the problem posed in the problem definition and thus redefine the problem. You may need to update the models after they have been deployed because more data becomes available. It is therefore important to understand that creating a data mining model is a process, and that each step in the process may be repeated as many times as necessary to create a good mining model.

Defining the Problem

The first step in the data mining process, as highlighted in the following diagram, is to clearly define the business problem.

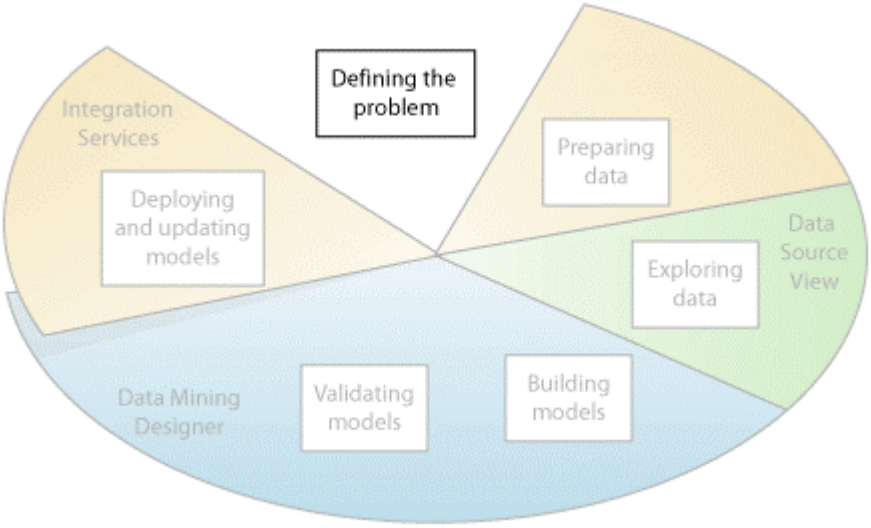


Fig. 4.2

This step includes analyzing the business requirements, defining the scope of the problem, defining the metrics by which the model will be evaluated, and defining the final objective for the data mining project. These tasks translate into questions like:

- What are you looking for?
- Which attribute of the dataset do you want to try to predict?
- What types of relationships are you trying to find?
- Do you want to make predictions from the data mining model or just look for interesting patterns and associations?

- How is the data distributed?
- How are the columns related, or if there are multiple tables, how are the tables related?

To answer the questions you may need to conduct a data availability study, investigating the needs of the business users with respect to the available data. If the data does not support needs of the users need, you may need to redefine the project.

Preparing Data

The second step in the data mining process, as highlighted in the following diagram, is to consolidate and clean the data identified in Defining the Problem.

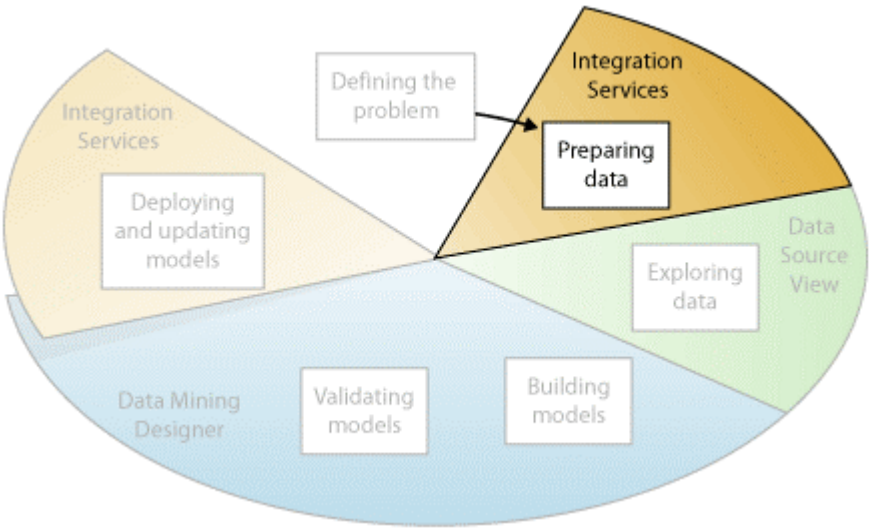


Fig. 4.3

Data can be scattered across a company and stored in different formats or contain inconsistencies such as flawed or missing entries.

For example, the data might show that a customer bought a product before she was born or shops regularly at a store 2,000 miles from her home. Before you begin to build the models, you need to fix these problems. Usually, you are working with a very large dataset and can not look through every transaction. Therefore, you need to use some form of automation (such as in Integration Services) to explore the data and find the inconsistencies.

Exploring Data

The third step in the process, as highlighted in the following diagram, is to explore the prepared data.

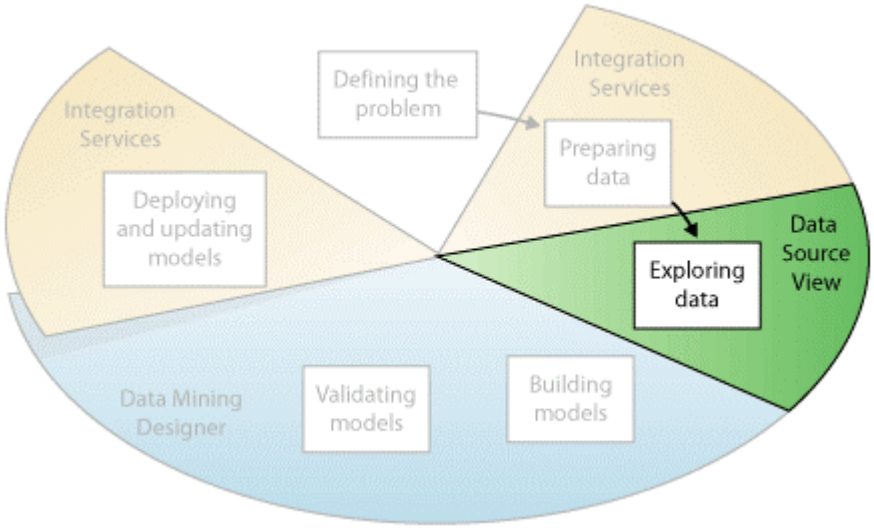


Fig. 4.4

You need to understand the data so that you can make appropriate decisions when you create the models. Exploration techniques include calculating the minimum and maximum values, calculating mean and standard deviations, and looking at the distribution of the data. After

exploring the data, you can decide if the dataset contains flawed data, and devise a strategy for fixing the problems.

Building Models

The fourth step in the process, as highlighted in the following diagram, is to build the mining models.

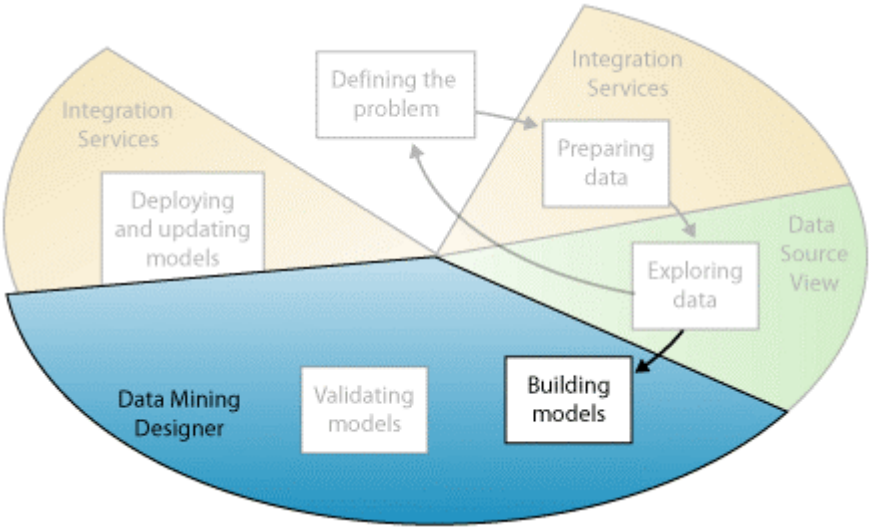


Fig. 4.5

Before building the model, you need to randomly separate the prepared data into separate training (model-building) and testing (validation) datasets. You use the training dataset to build the model and testing dataset to test the accuracy of the model by creating prediction queries. You can use the Percentage Sampling Transformation in Integration Services to split the dataset.

Exploring and Validating Models

The fifth step of the process, as highlighted in the following diagram, is to explore the models that you have built, and test their effectiveness.

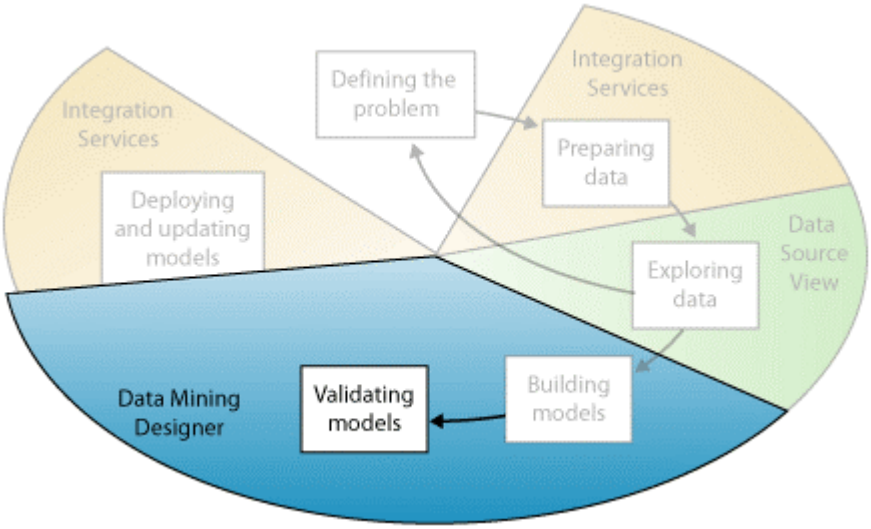


Fig. 4.6

You do not want to deploy a model into a production environment without first testing how well it performs. Also, you may have created several models and will need to decide which will perform the best. If none of the models you created in the Building Models step perform well, you may need to return to a previous step in the process either by redefining the problem (Defining the Problem) or reinvestigating the data in the original dataset (Preparing Data).

Deploying and Updating Models

The last step of the data mining process, as highlighted in the following diagram, is to deploy the models that performed the best to a production environment.

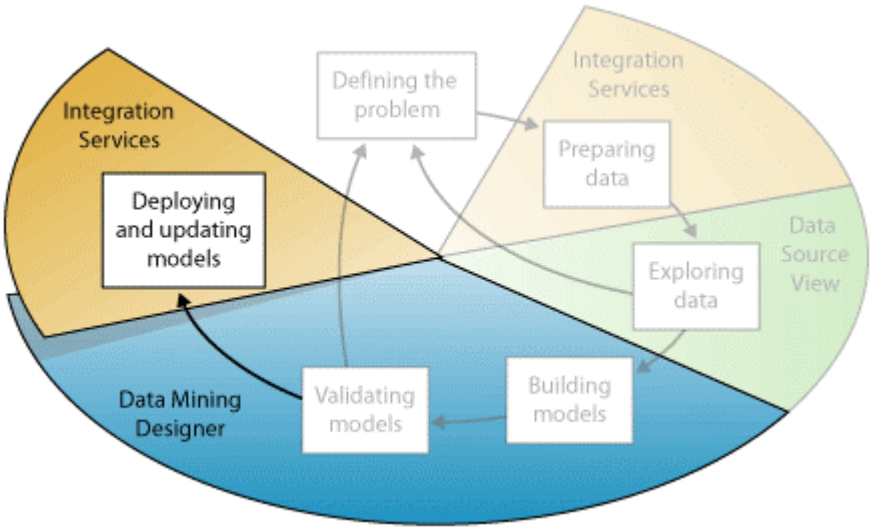


Fig. 4.7

Once the mining models exist in a production environment you can perform many tasks, depending on your needs. Tasks can include:

- Using the models to create predictions, which you can then use to make business decisions. SQL Server provides the DMX language that you can use to create prediction queries and a prediction query builder to help build the queries.

- Embedding data mining functionality directly into an application. You can either include Analysis Management Objects (AMO), an assembly containing a set of objects your application can use to create, alter, process, and delete mining structures and mining models, or you can send XML for Analysis messages directly to an Analysis Services instance.

- Using Integration services to create a package where a mining model is used to intelligently separate incoming data into multiple tables. For example, if a database is continually updated with potential customers, you could use a mining model in conjunction with Integration Services to split the incoming into customers that are likely to purchase a product, and customers that are likely to not purchase a product.

- Create a report allowing users to directly query against an existing mining model

Updating the model is part of the deployment strategy. As more data comes into the organization, you need to develop a process to reprocess the models, thereby improving their effectiveness.

4.2. Converting data to data warehouse from legacy system

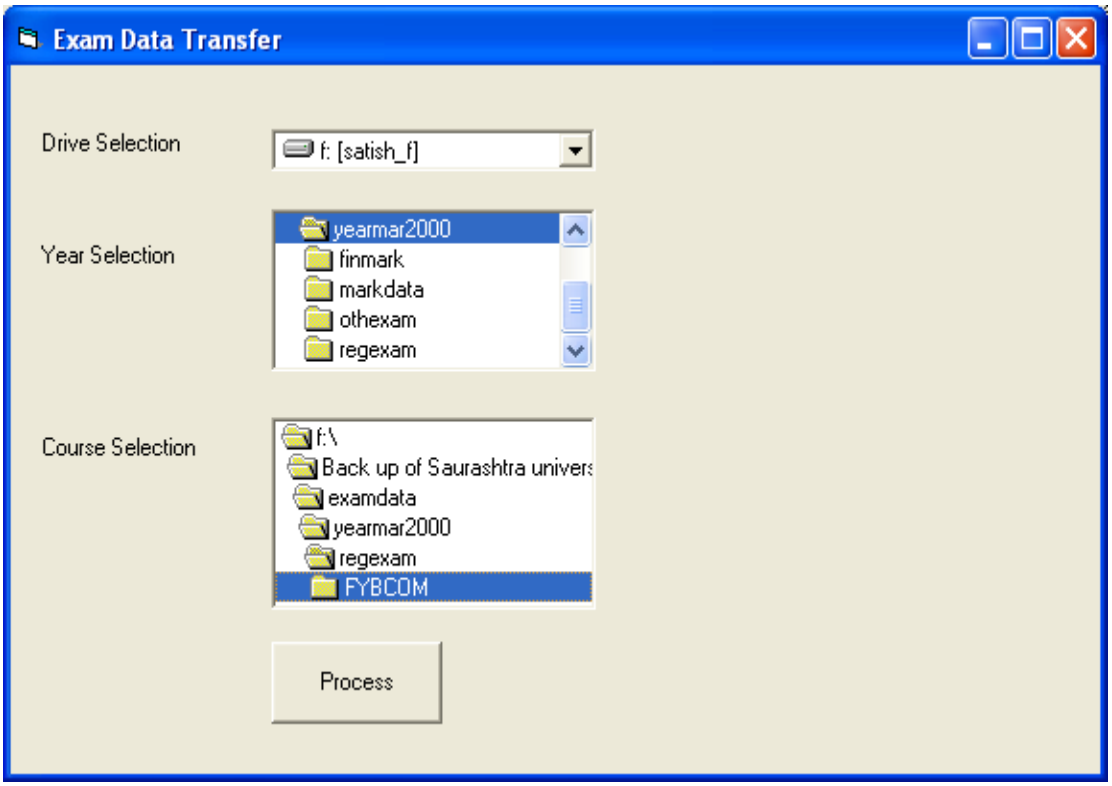


Fig. 4.8

The above application is developed to transfer the data from legacy system to the data warehouse. Here the legacy system is in the dbase form and the data warehouse is using the SQL server 2000 database sever. In the legacy system there are nearly 15 tables but for the data warehouse only three tables are useful and that data is transfer in to a single table in SQL Server 2000. There is no way to find out the year and course from the table so that fields are taken from the folder name to the table.

To do the transformation one has to select the path in the above application, and then has to select the year and course. After that one has to press the Process button.

For the above application, the DTS of SQL Server 2000 and Visual basic 6.0 is used. At end of the Thesis you can find the source code for this

4.3. Applying different mining algorithms to data

Now we have a data in data warehouse, so we can do the data mining. Here are the some of the data mining result obtain after applying the all the steps of topic 4.1.

For the below application the SQL server 2000 Analysis services is used.

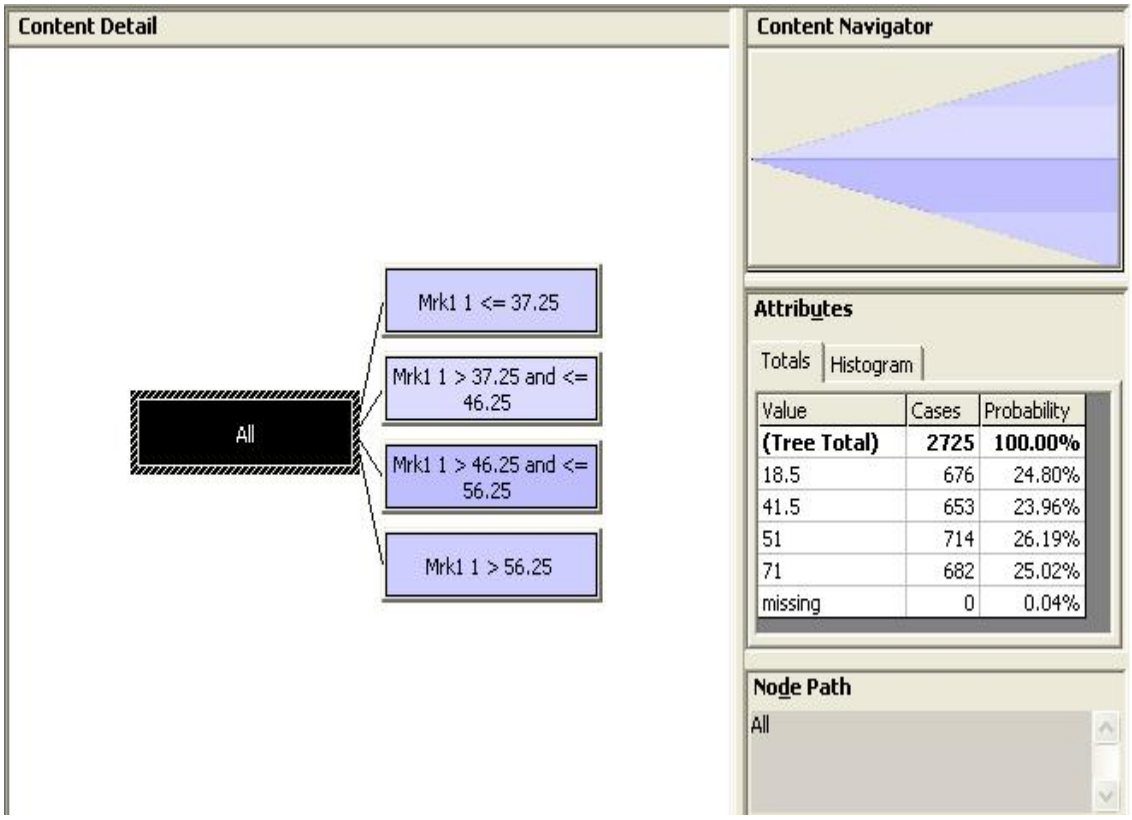


Fig 4.9

The Fig. 4.9 shows the decision tree for the student of F.Y. B.Com of year 2000. Here it has generated 4-leaf node.

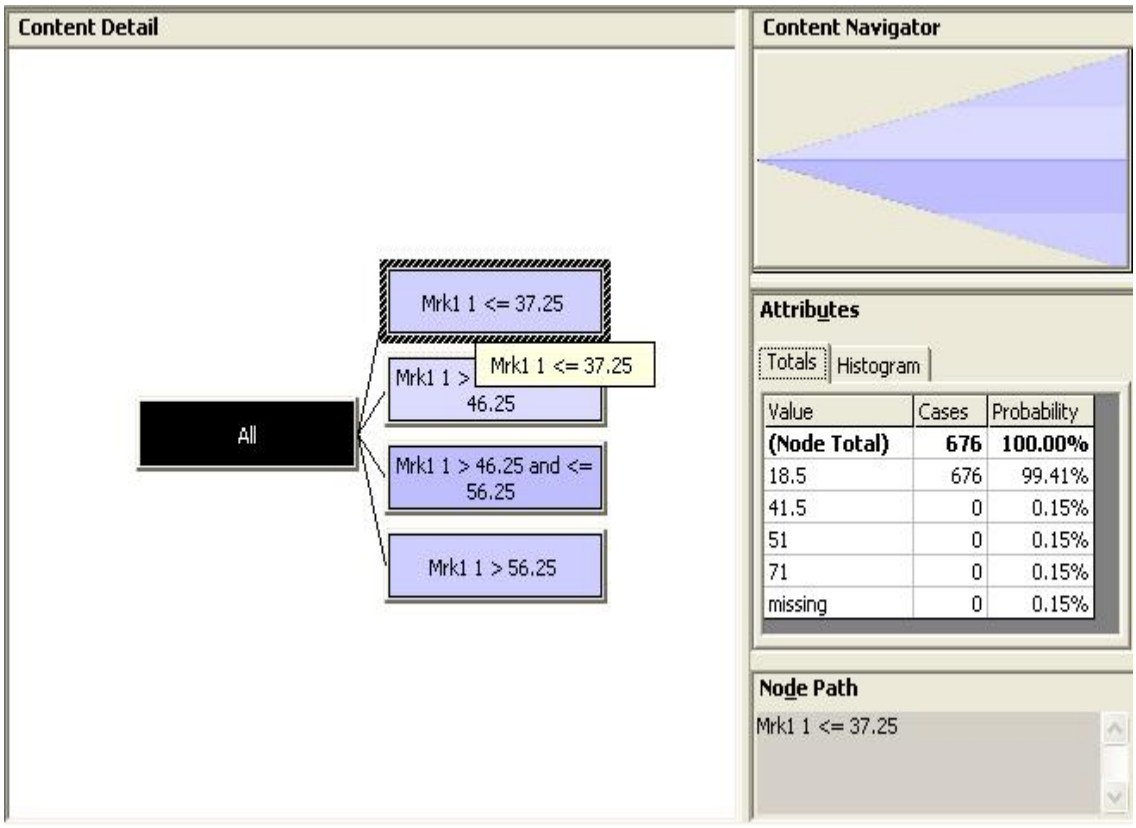


Fig. 4.10

The Fig. 4.10 shows the first leaf node of the decision tree having the 676 cases with the condition of subject marks of mrk1 is less than equal to 37.25. If you observe that the result has the probability of 99.41%.

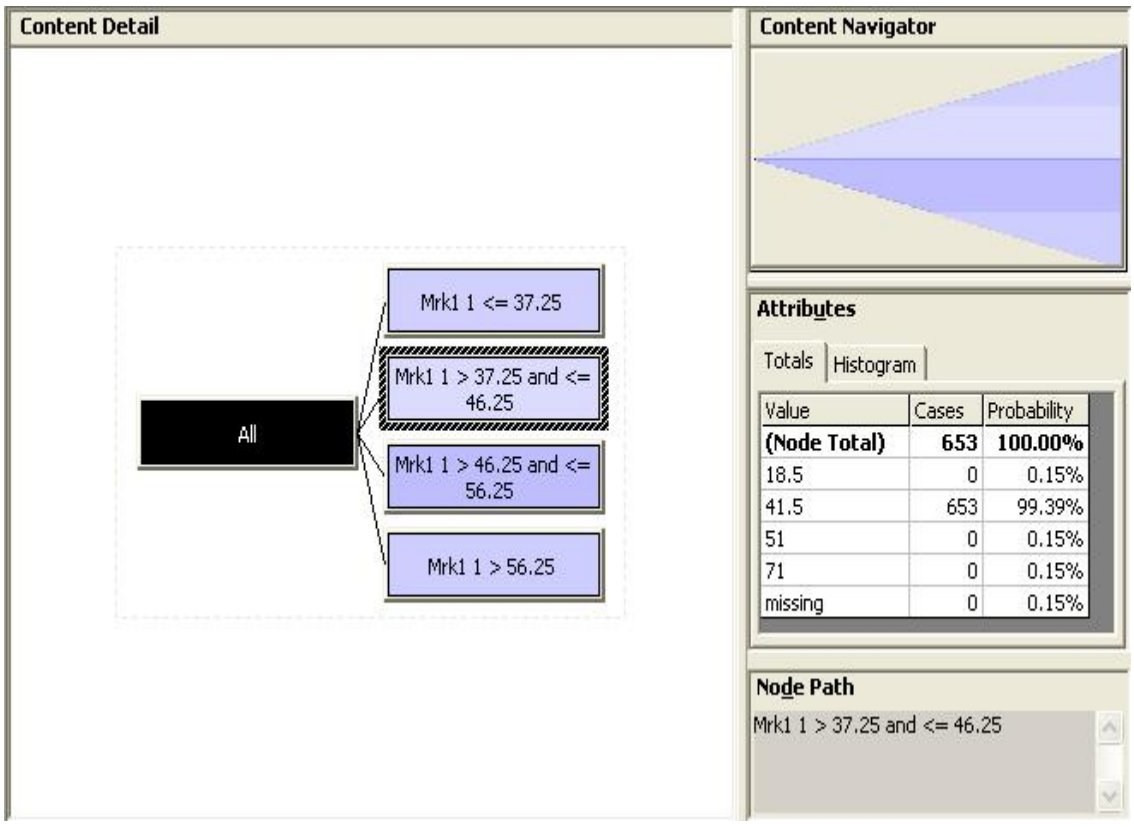


Fig. 4.11

The Fig. 4.11 shows the second leaf node of the decision tree having the 653 cases with the condition of subject marks of mrk1 is greater than 37.25 and less then equal to 46.25. If you observe that the result has the probability of 99.39%.

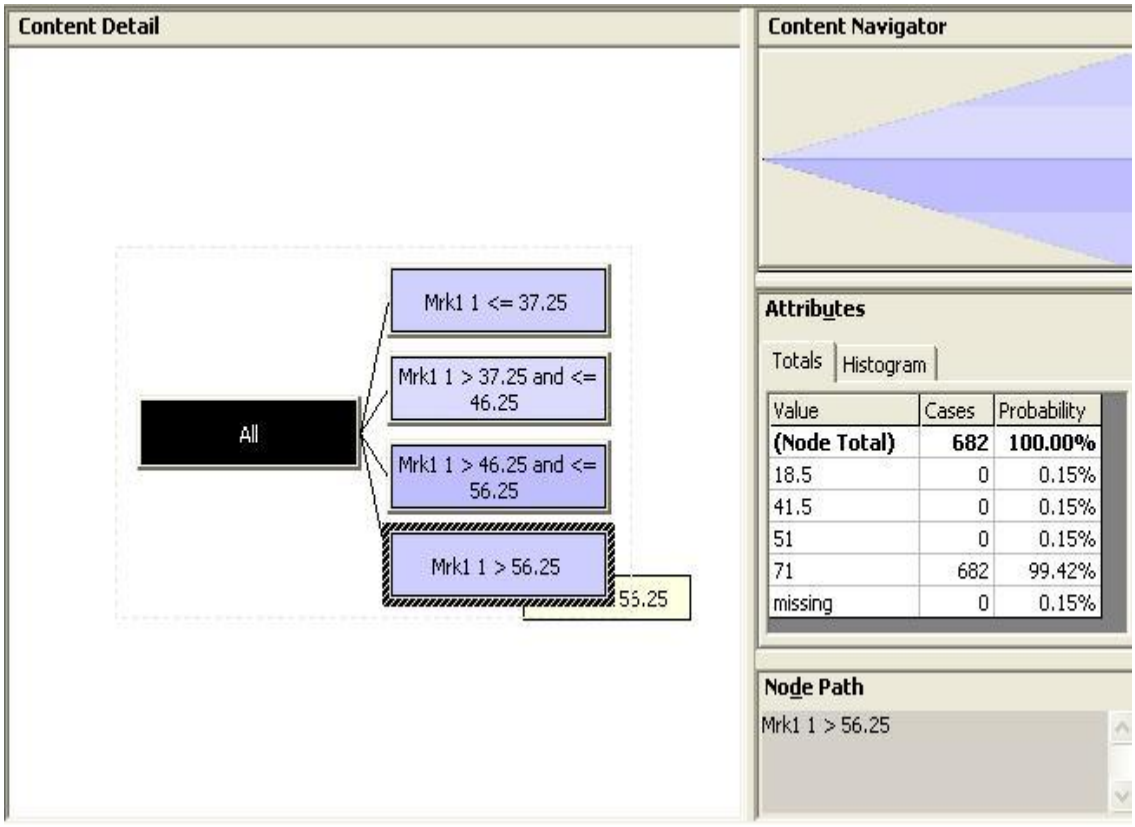


Fig. 4.12

The Fig. 4.12 shows the fourth leaf node of the decision tree having the 682 cases with the condition of subject marks of mrk1 is greater than 56.25. If you observe that the result has the probability of 99.42%.

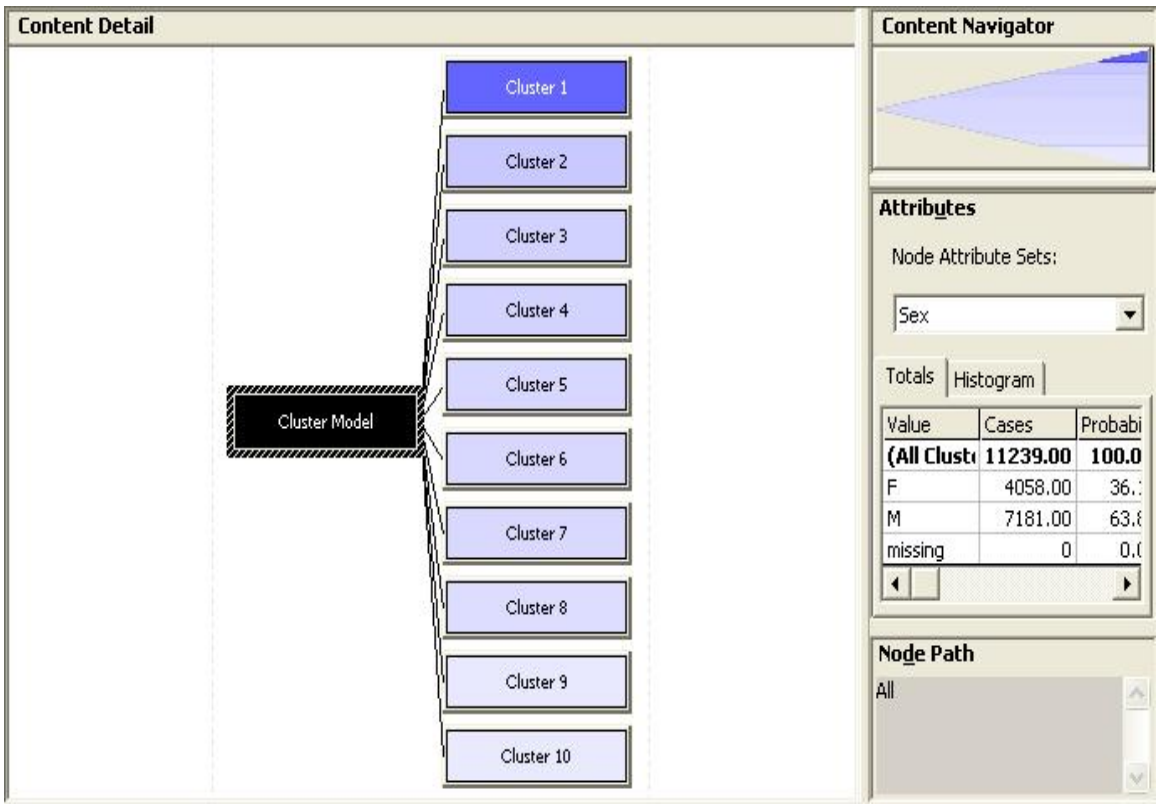


Fig. 4.13

Fig. 4.13 shows the data cluster having the 11239 records of F.Y. B.Com of year 2000 students. From the above fig. you can see that there are 4059 female and 7181 male student. Here we have generated 10 different clusters all having the different rule.

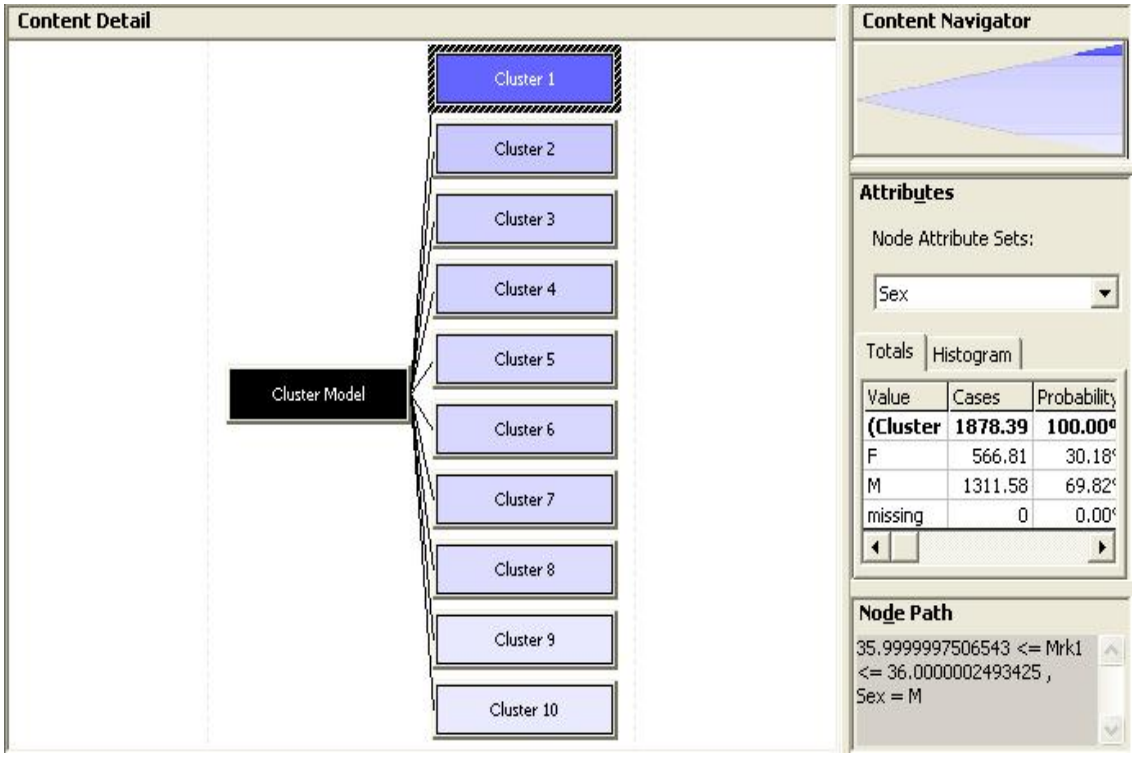


Fig. 4.14

From the above figure one can see that if the subject mark of mrk1 is 36 then 69.82% chance is of a male student and 30.18% is of female student.

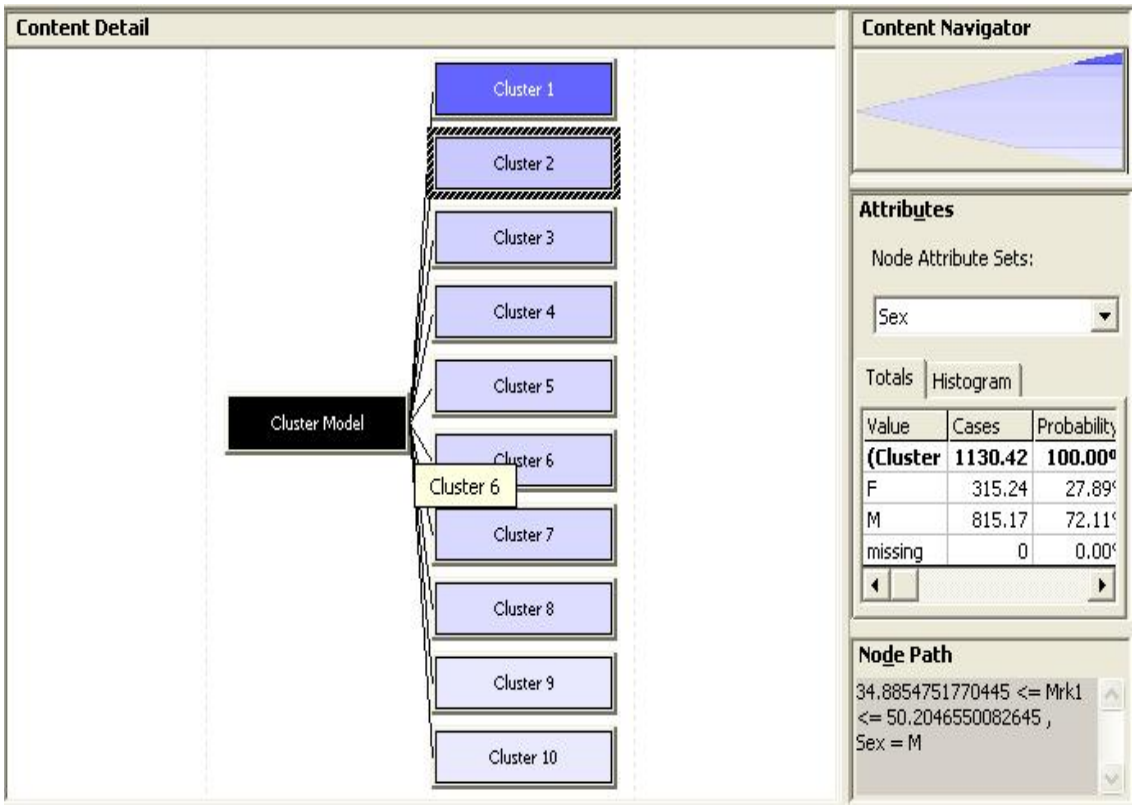

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Sex
Totals
Histogram
Value
Cases
Probability
(Cluster
1130.42
100.00
F
315.24
27.89
M
815.17
72.11
missing
0
0.00
Node Path
34.8854751770445 <= Mrk1 <= 50.2046550082645 , Sex = M

Fig. 4.15

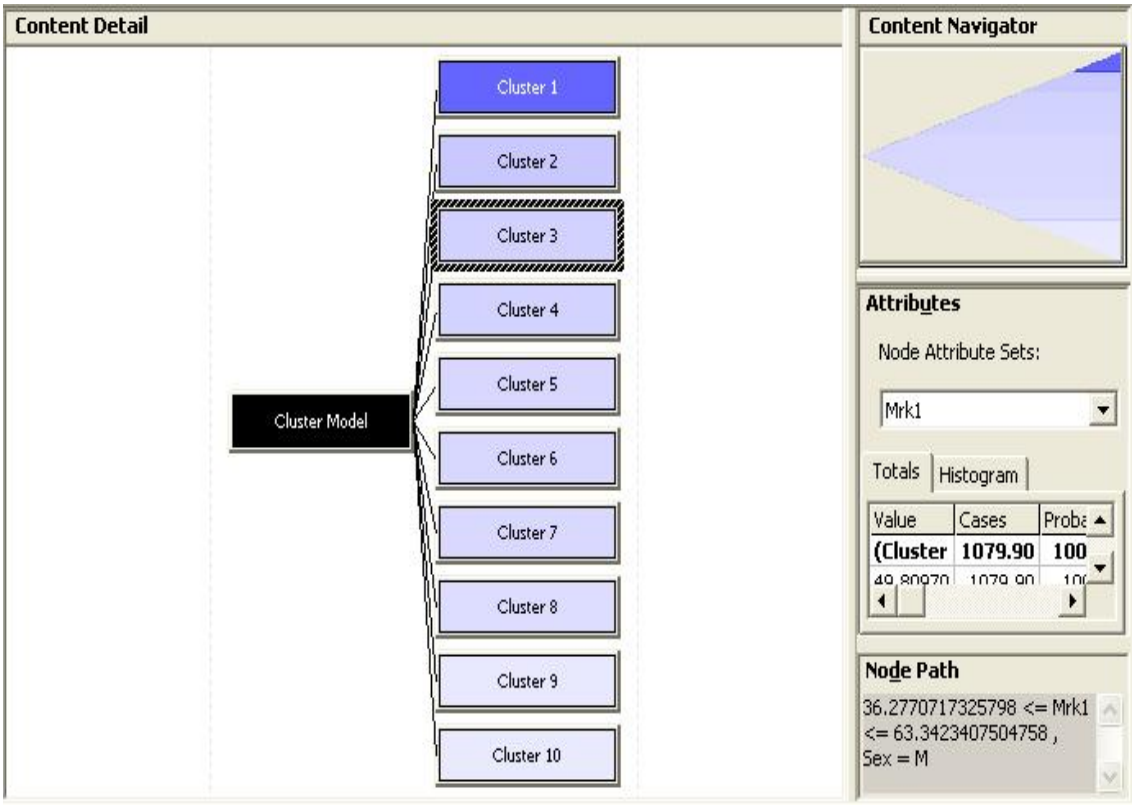

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
(Cluster
1079.90
100
Node Path
36.2770717325798 <= Mrk1 <= 63.3423407504758 , Sex = M

Fig. 4.16

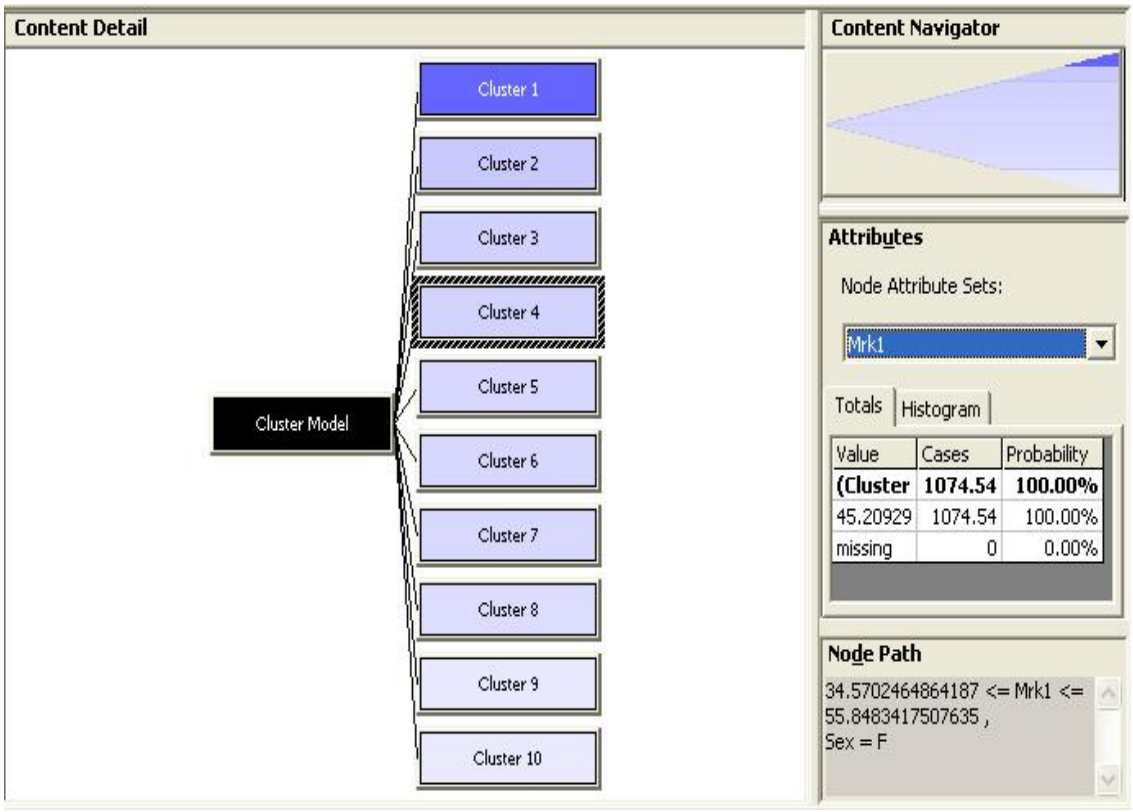

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
Probability
(Cluster
1074.54
100.00%
45.20929
1074.54
100.00%
missing
0
0.00%
Node Path
34.5702464864187 <= Mrk1 <= 55.8483417507635 ,
Sex = F

Fig. 4.17

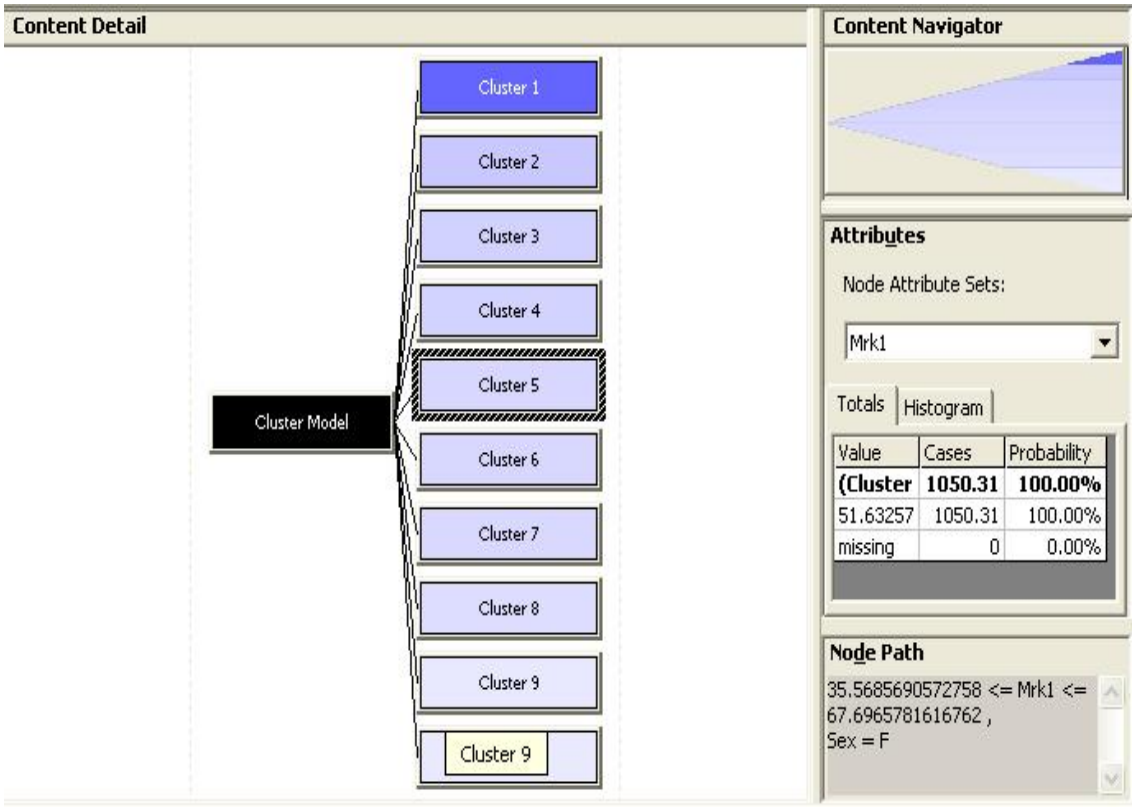

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 9
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
Probability
(Cluster
1050.31
100.00%
51.63257
1050.31
100.00%
missing
0
0.00%
Node Path
35.5685690572758 <= Mrk1 <= 67.6965781616762 ,
Sex = F

Fig. 4.18

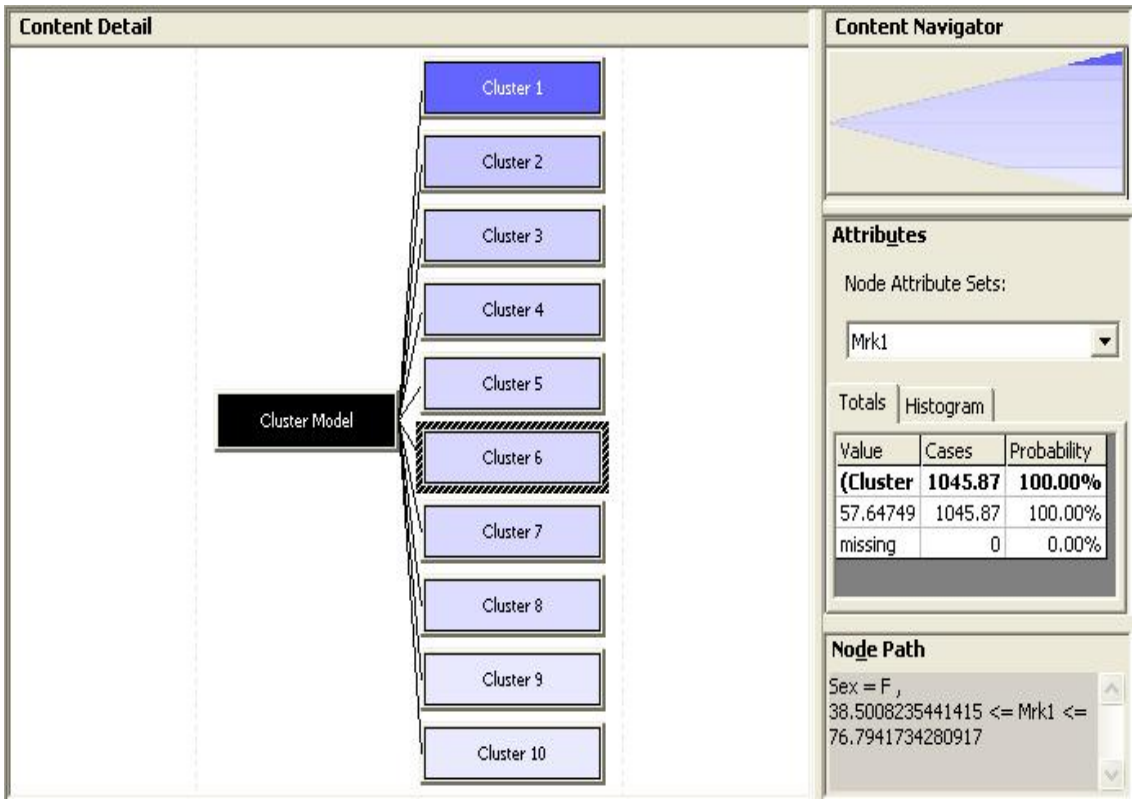

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
Probability
(Cluster
1045.87
100.00%
57.64749
1045.87
100.00%
missing
0
0.00%
Node Path
Sex = F ,
38.5008235441415 <= Mrk1 <=
76.7941734280917

Fig. 4.19

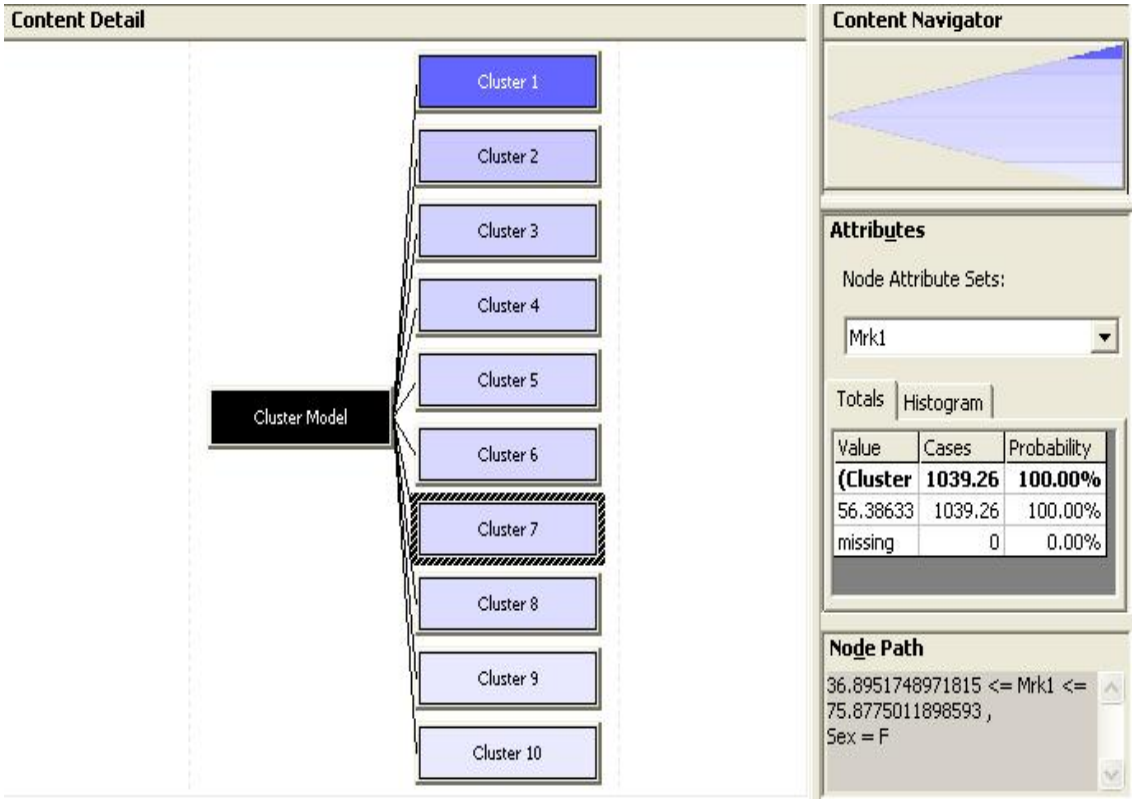

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
Probability
(Cluster
1039.26
100.00%
56.38633
1039.26
100.00%
missing
0
0.00%
Node Path
36.8951748971815 <= Mrk1 <= 75.8775011898593 ,
Sex = F

Fig. 4.20

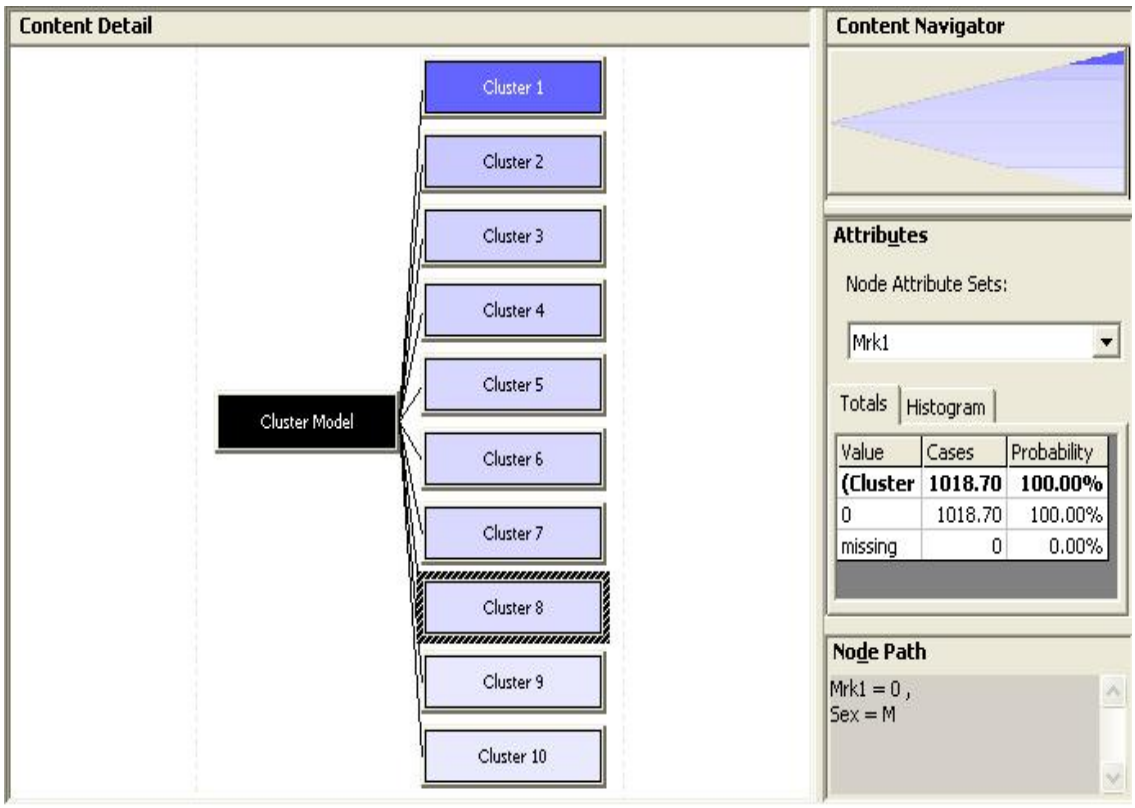

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
Probability
(Cluster
1018.70
100.00%
0
1018.70
100.00%
missing
0
0.00%
Node Path
Mrk1 = 0 ,
Sex = M

Fig. 4.21

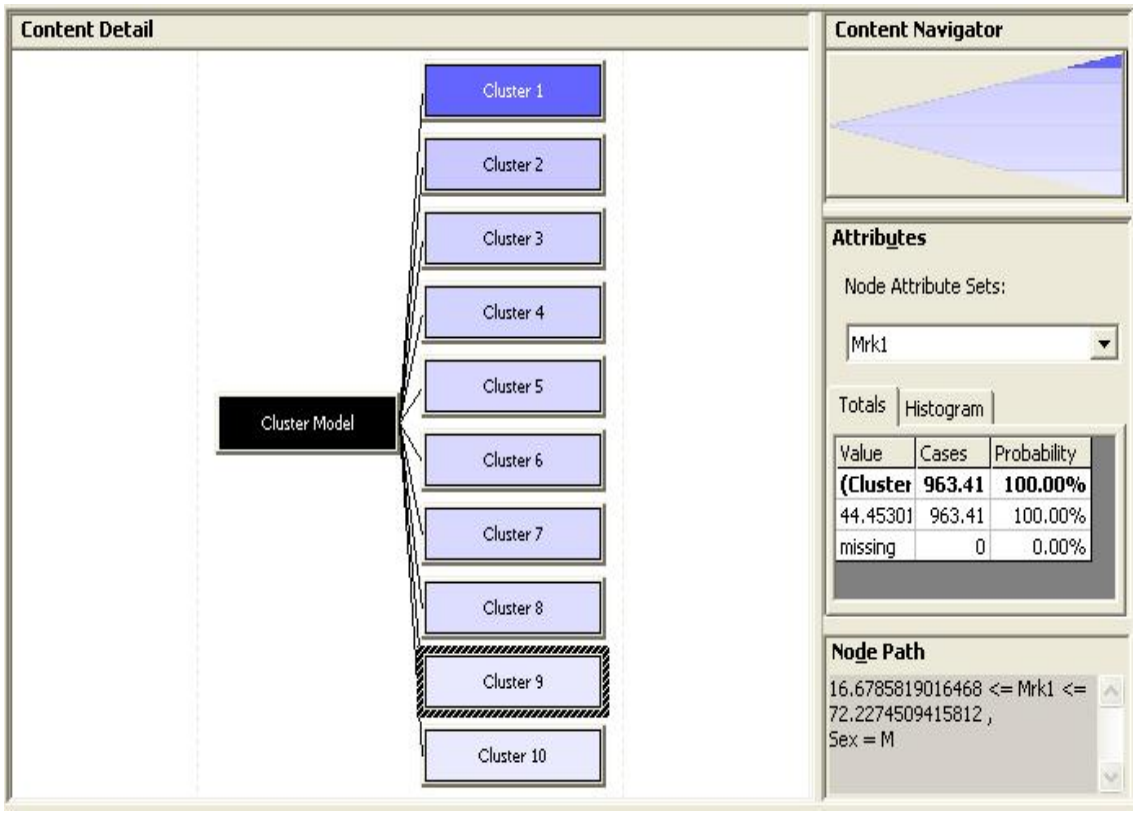

Content Detail
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster Model
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Content Navigator
Attributes
Node Attribute Sets:
Mrk1
Totals
Histogram
Value
Cases
Probability
(Cluster
963.41
100.00%
44.45301
963.41
100.00%
missing
0
0.00%
Node Path
16.6785819016468 <= Mrk1 <= 72.2274509415812 ,
Sex = M

Fig. 4.22

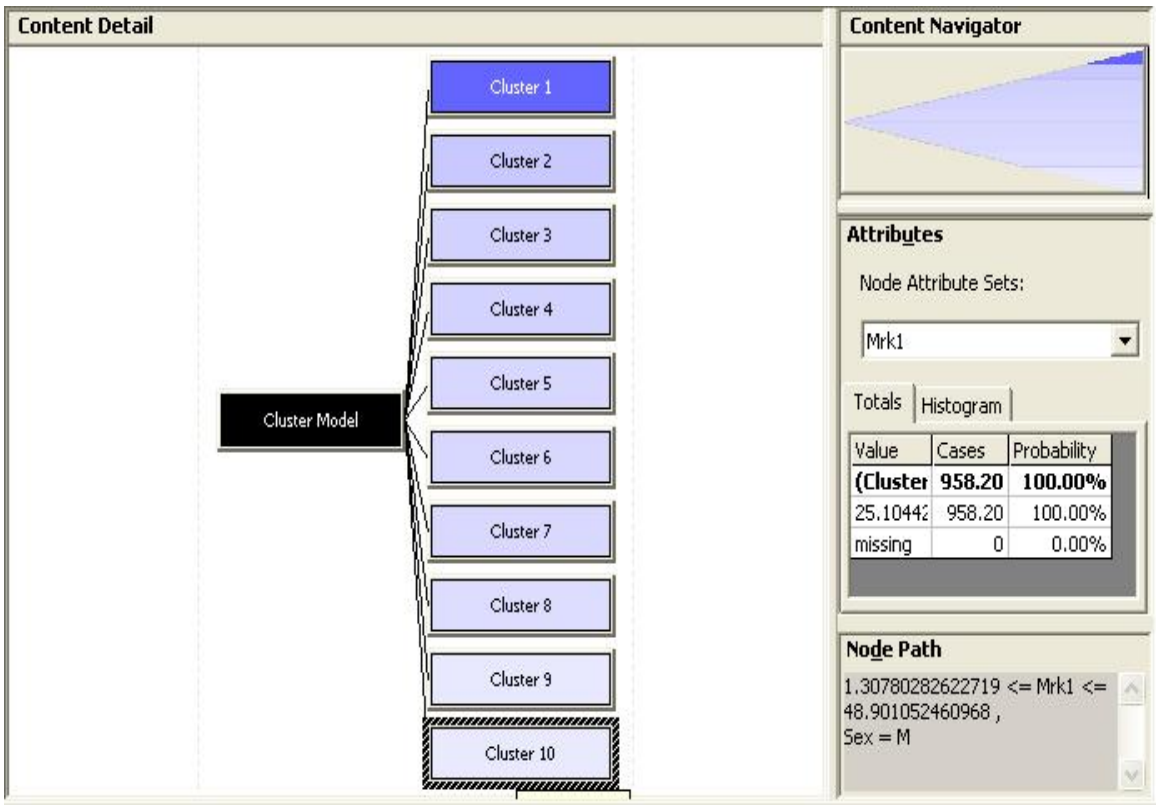


Fig. 4.23

Fig. 4.15 to Fig. 4.23 shows the different cluster generated for the student data for the subject mark Mrk1.

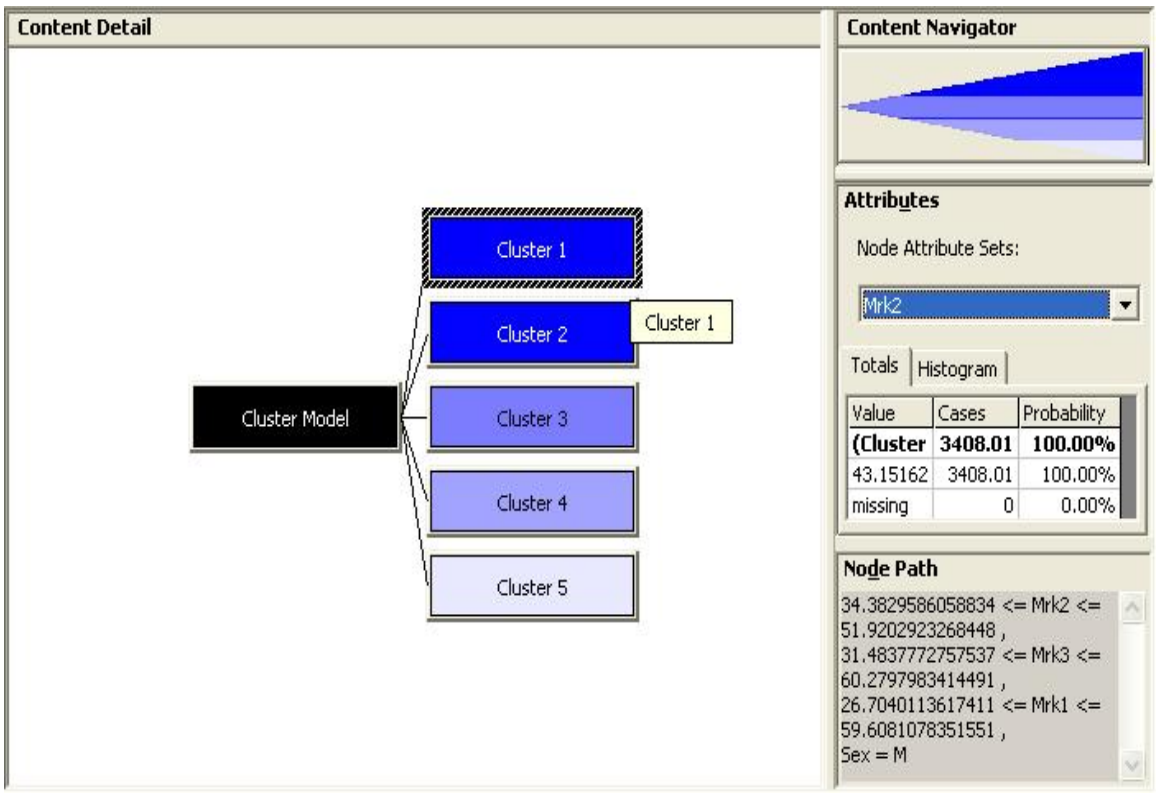

Content Detail
Content Navigator
Attributes
Node Attribute Sets:
Mrk2
Totals
Histogram
Value
Cases
Probability
(Cluster
3408.01
100.00%
43.15162
3408.01
100.00%
missing
0
0.00%
Node Path
34.3829586058834 <= Mrk2 <= 51.9202923268448 , 31.4837772757537 <= Mrk3 <= 60.2797983414491 , 26.7040113617411 <= Mrk1 <= 59.6081078351551 , Sex = M
Cluster Model
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 1

Fig. 4.24

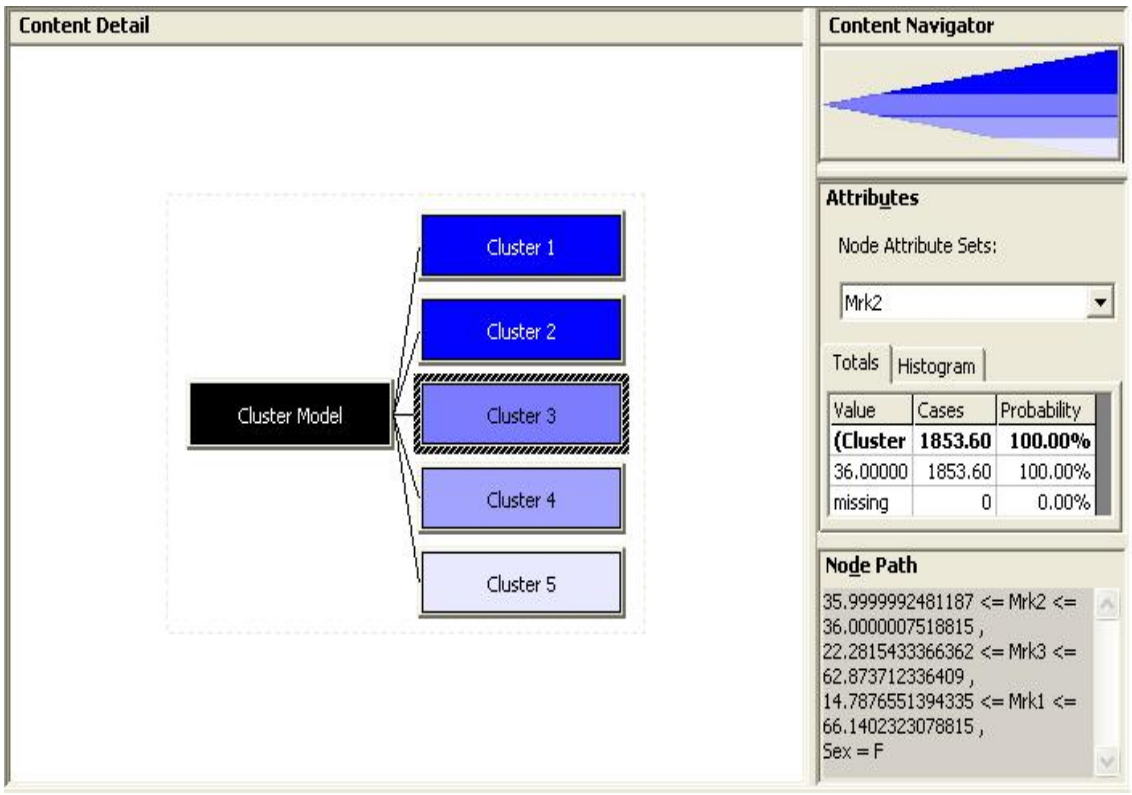

Content Detail
Content Navigator
Cluster 1
Cluster 2
Cluster Model
Cluster 3
Cluster 4
Cluster 5
Attributes
Node Attribute Sets:
Mrk2
Totals
Histogram
Value
Cases
Probability
(Cluster
1853.60
100.00%
36.00000
1853.60
100.00%
missing
0
0.00%
Node Path
35.9999992481187 <= Mrk2 <= 36.0000007518815 ,
22.2815433366362 <= Mrk3 <= 62.873712336409 ,
14.7876551394335 <= Mrk1 <= 66.1402323078815 ,
Sex = F

Fig. 4.25

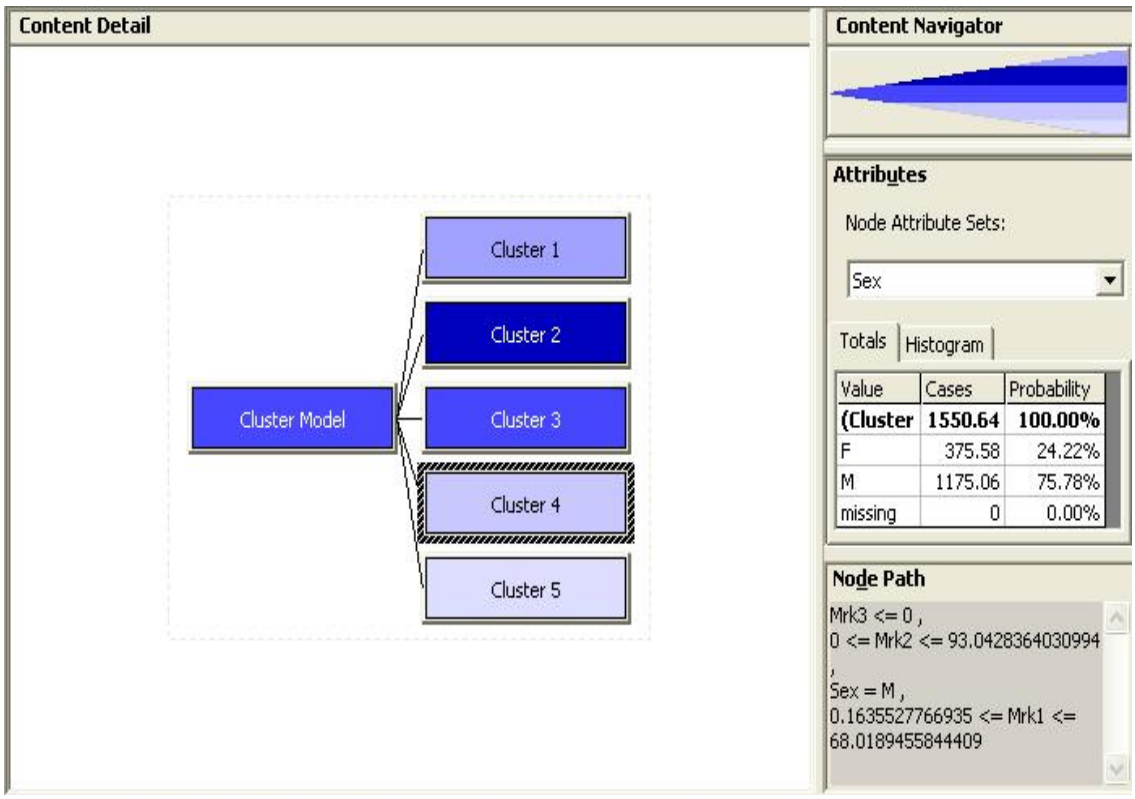

Content Detail
Cluster 1
Cluster 2
Cluster Model
Cluster 3
Cluster 4
Cluster 5
Content Navigator
Attributes
Node Attribute Sets:
Sex
Totals
Histogram
Value
Cases
Probability
(Cluster
1550.64
100.00%
F
375.58
24.22%
M
1175.06
75.78%
missing
0
0.00%
Node Path
Mrk3 <= 0 ,
0 <= Mrk2 <= 93.0428364030994
,
Sex = M ,
0.1635527766935 <= Mrk1 <=
68.0189455844409

Fig. 4.26

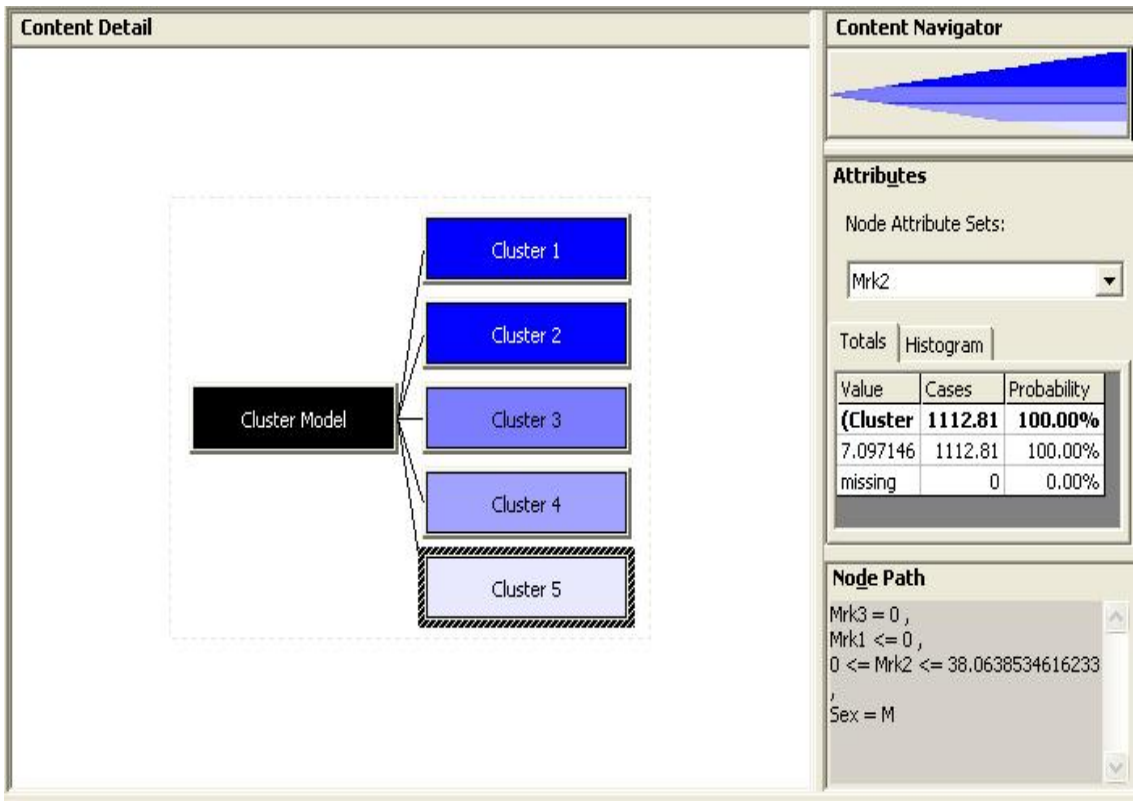


Fig. 4.27

Fig. 4.24 to Fig. 4.27 shows the five clusters for the same data for which we have generated the ten clusters. So it is possible to generate the cluster according to our need.

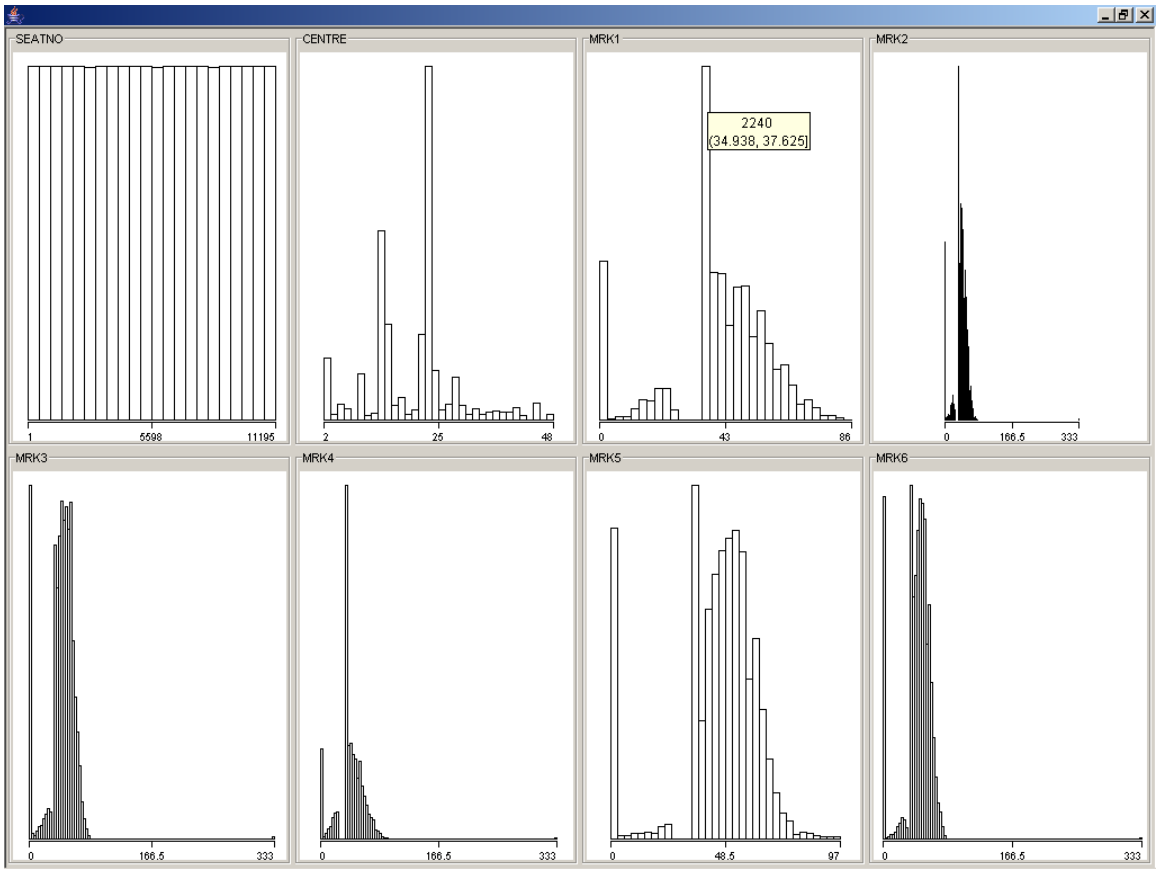


Fig. 4.28

Fig 4.28 shows the result of Data Visualization using the WEKA (an open source data mining tool developed using JAVA). For the above example the data is taken again from First Year of B.Com. of the year 2000. There are total 11195 records in it. To make the data useful to WEKA, converted in the ARFF format using the Notepad and Excel.

By just Visualizing the result's one can easily see the marks range where the number of students are more. For example in subject Mrk1 the 2240 student out of 11195 having marks in between 34.93 to 37.625

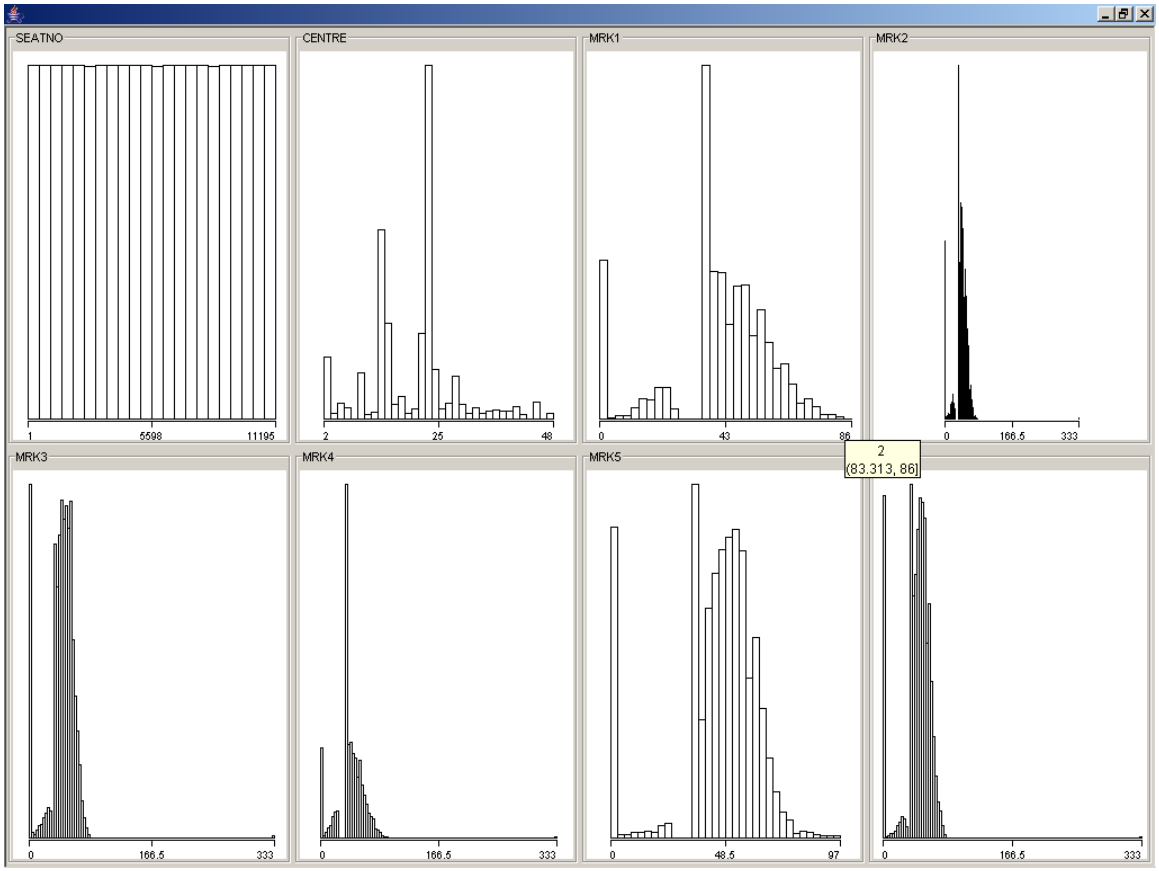


Fig. 4.29

Fig. 4.29 shows that only 2 student having the marks more than 84 in subject Mrk1.

You can also observe that the data is not 100% correct in the subjects Mrk2, Mrk3, Mrk4 and Mrk6. Because the marks are more than 100 which in not possible because the exam is of 100 marks maximum. So one can use the visualization for data validity.

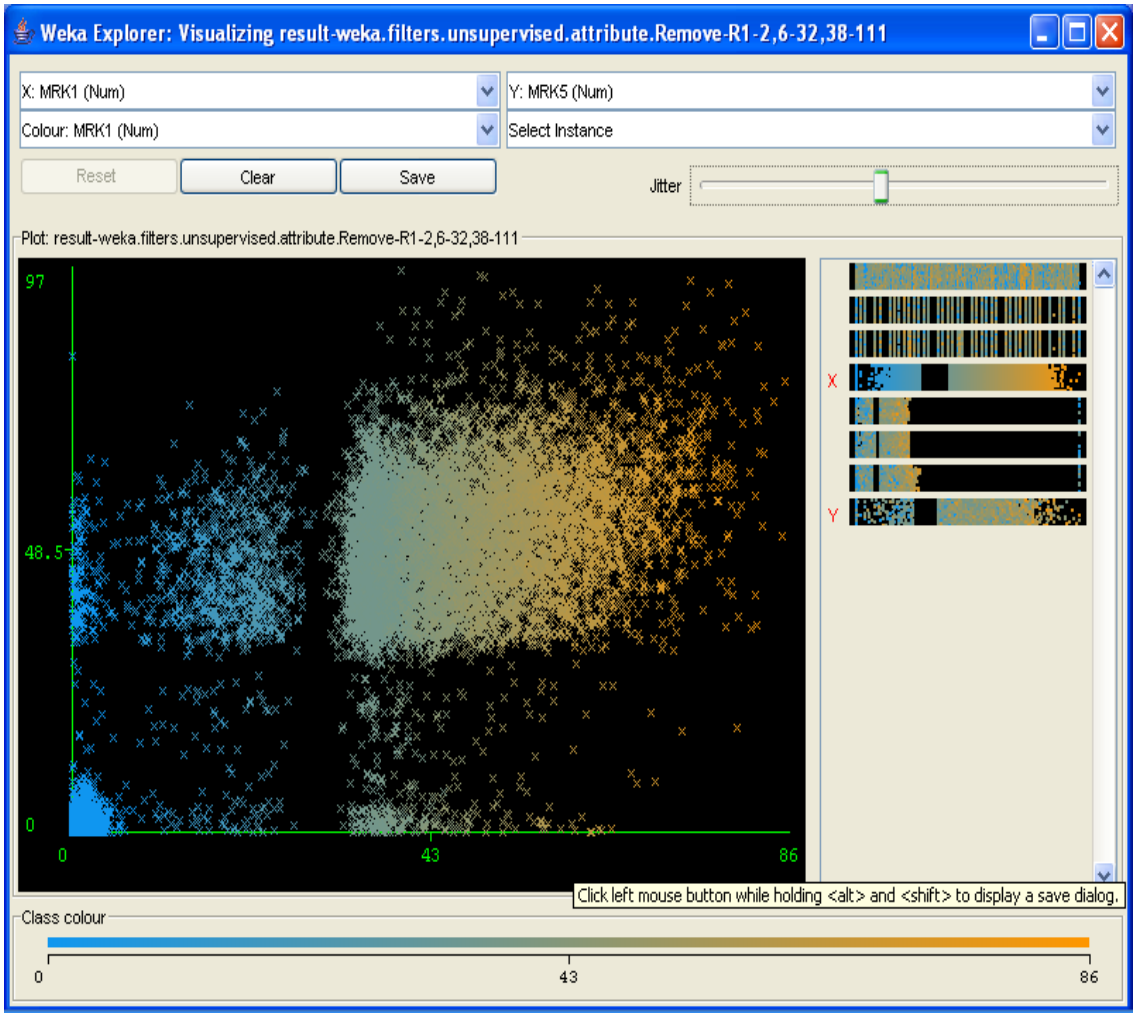


Fig. 4.30

Fig. 4.30 shows the chart for subject Mrk1 and subject Mrk5. On X-axis the data for Mrk1 is plotted and on Y-axis the data of Mrk2 is plotted. Below is the data which is indicated one of the point of the chart. Using this instance information one can generate the rule too.

Plot : Master Plot
Instance: 2212
SEATNO : 2213.0
CENTRE : 13.0
COLLEGE : 1305.0
MRK1 : 51.0
MRK2 : 46.0
MRK3 : 46.0
MRK4 : 22.0
MRK5 : 52.0

Plot : Master Plot
Instance: 3353
SEATNO : 3354.0
CENTRE : 15.0
COLLEGE : 1503.0
MRK1 : 51.0
MRK2 : 36.0
MRK3 : 37.0
MRK4 : 36.0
MRK5 : 50.0

Plot : Master Plot
Instance: 3397
SEATNO : 3398.0
CENTRE : 15.0
COLLEGE : 1503.0

MRK1 : 50.0
MRK2 : 41.0
MRK3 : 45.0
MRK4 : 40.0
MRK5 : 46.0

Plot : Master Plot
Instance: 6499
SEATNO : 6500.0
CENTRE : 23.0
COLLEGE : 2325.0
MRK1 : 48.0
MRK2 : 22.0
MRK3 : 41.0
MRK4 : 54.0
MRK5 : 41.0

Plot : Master Plot
Instance: 7700
SEATNO : 7701.0
CENTRE : 23.0
COLLEGE : 2330.0
MRK1 : 51.0
MRK2 : 47.0
MRK3 : 66.0
MRK4 : 36.0
MRK5 : 50.0

Plot : Master Plot

Instance: 9003
SEATNO : 9004.0
CENTRE : 23.0
COLLEGE : 2349.0
MRK1 : 53.0
MRK2 : 52.0
MRK3 : 57.0
MRK4 : 36.0
MRK5 : 46.0

Plot : Master Plot
Instance: 10485
SEATNO : 10486.0
CENTRE : 34.0
COLLEGE : 3401.0
MRK1 : 51.0
MRK2 : 36.0
MRK3 : 37.0
MRK4 : 44.0
MRK5 : 50.0

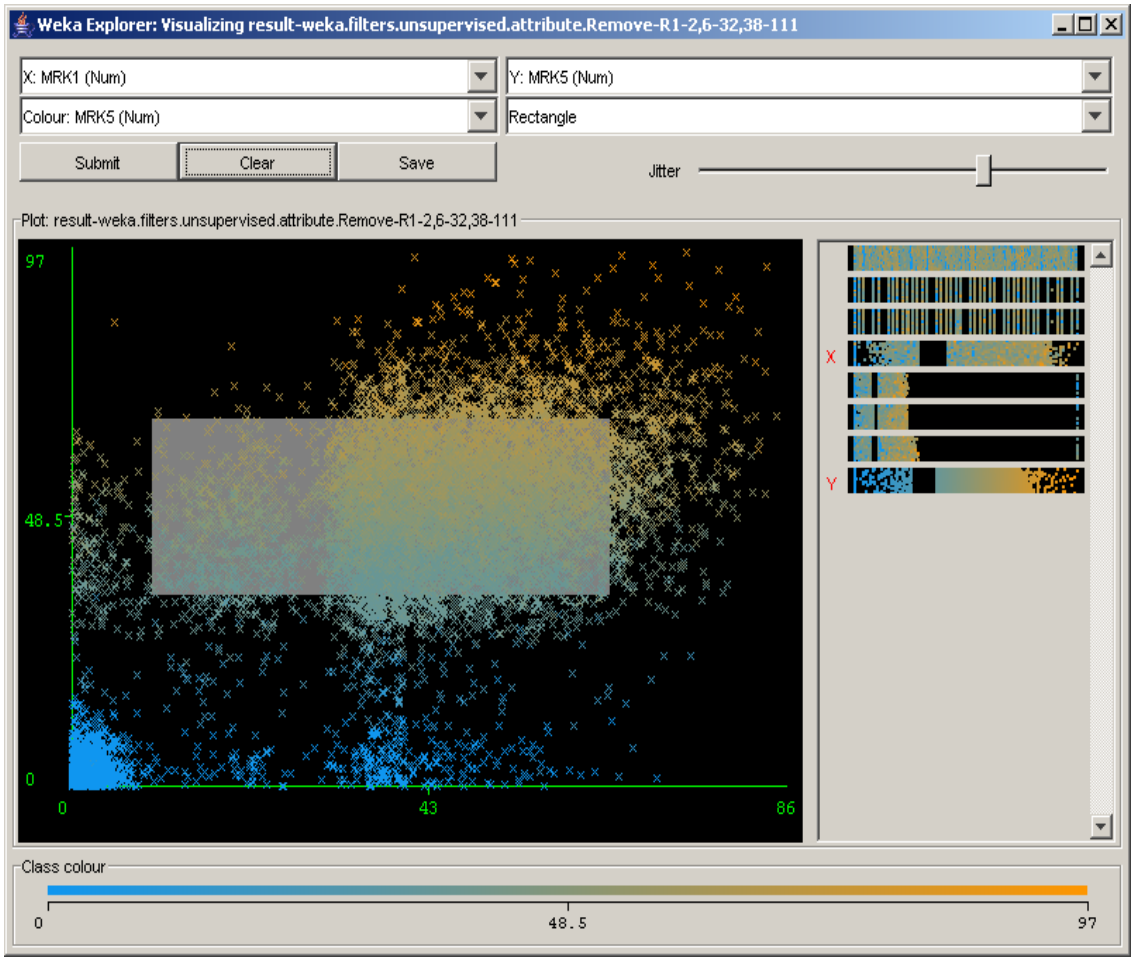


Fig. 4.31

Fig. 4.31 shows the chart for subject Mrk1 and subject Mrk5. On X-axis the data for Mrk1 is plotted and on Y-axis the data of Mrk2 is plotted. Below is the data which is indicated the selected rectangle of the chart.

Plot : Master Plot

Instance: 1197

SEATNO : 1198.0

CENTRE : 10.0

COLLEGE : 1001.0

MRK1 : 51.0

MRK2 : 59.0
MRK3 : 47.0
MRK4 : 73.0
MRK5 : 47.0

Plot : Master Plot
Instance: 1738
SEATNO : 1739.0
CENTRE : 13.0
COLLEGE : 1301.0
MRK1 : 44.0
MRK2 : 38.0
MRK3 : 36.0
MRK4 : 40.0
MRK5 : 40.0

Plot : Master Plot
Instance: 2804
SEATNO : 2805.0
CENTRE : 13.0
COLLEGE : 1307.0
MRK1 : 45.0
MRK2 : 53.0
MRK3 : 57.0
MRK4 : 49.0
MRK5 : 45.0

Plot : Master Plot
Instance: 3119

SEATNO : 3120.0
CENTRE : 14.0
COLLEGE : 1401.0
MRK1 : 43.0
MRK2 : 36.0
MRK3 : 40.0
MRK4 : 42.0
MRK5 : 50.0

Plot : Master Plot
Instance: 5223
SEATNO : 5224.0
CENTRE : 22.0
COLLEGE : 2202.0
MRK1 : 40.0
MRK2 : 44.0
MRK3 : 46.0
MRK4 : 36.0
MRK5 : 48.0

Plot : Master Plot
Instance: 5448
SEATNO : 5449.0
CENTRE : 22.0
COLLEGE : 2205.0
MRK1 : 40.0
MRK2 : 45.0
MRK3 : 40.0
MRK4 : 42.0

MRK5 : 51.0

Plot : Master Plot
Instance: 6379
SEATNO : 6380.0
CENTRE : 23.0
COLLEGE : 2325.0
MRK1 : 43.0
MRK2 : 43.0
MRK3 : 58.0
MRK4 : 36.0
MRK5 : 44.0

Plot : Master Plot
Instance: 10551
SEATNO : 10552.0
CENTRE : 35.0
COLLEGE : 3501.0
MRK1 : 44.0
MRK2 : 38.0
MRK3 : 36.0
MRK4 : 6.0
MRK5 : 41.0

Results for weka.classifiers.rules.ZeroR

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

ZeroR predicts class value: 45.009289861545334

Time taken to build model: 0.03 seconds

=== Cross-validation ===

=== Summary ===

Correlation coefficient	-0.0282
Mean absolute error	12.8975
Root mean squared error	18.0772
Relative absolute error	100 %
Root relative squared error	100 %
Total Number of Instances	11195

Results for weka.classifiers.rules.ZeroR

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: split 66% train, remainder test

=== Classifier model (full training set) ===

ZeroR predicts class value: 45.009289861545334

Time taken to build model: 0 seconds

=== Evaluation on test split ===

=== Summary ===

Correlation coefficient	0
Mean absolute error	12.9393
Root mean squared error	18.1137
Relative absolute error	100 %
Root relative squared error	100 %
Total Number of Instances	3807

Results for weka.attributeSelection.CfsSubsetEval

Evaluator: weka.attributeSelection.CfsSubsetEval

Search: weka.attributeSelection.BestFirst -D 1 -N 5

Relation: result-weka.filters.unsupervised.attribute.Remove-R1-2,6-32,38-111

Instances: 11195

Attributes: 8

SEATNO
CENTRE
COLLEGE
MRK1
MRK2
MRK3
MRK4
MRK5

Evaluation mode: 10-fold cross-validation

=== Attribute selection 10 fold cross-validation seed: 1 ===

number of folds (%) attribute

0(0 %) 1 SEATNO
0(0 %) 2 CENTRE
0(0 %) 3 COLLEGE
10(100 %) 5 MRK2
10(100 %) 6 MRK3
10(100 %) 7 MRK4
10(100 %) 8 MRK5

Results for weka.attributeSelection.CfsSubsetEval

Evaluator: weka.attributeSelection.CfsSubsetEval

Search: weka.attributeSelection.GeneticSearch -Z 20 -G 20 -C 0.6 -M 0.033 -R 20 -S 1

Relation: result-weka.filters.unsupervised.attribute.Remove-R1-2,6-32,38-111

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Evaluation mode: 10-fold cross-validation

=== Attribute selection 10 fold cross-validation seed: 1 ===

number of folds (%) attribute

0(0 %) 1 SEATNO

0(0 %) 2 CENTRE

0(0 %) 3 COLLEGE

10(100 %) 4 MRK1

10(100 %) 6 MRK3

10(100 %) 7 MRK4

10(100 %) 8 MRK5

Results for weka.classifiers.rules.DecisionTable -X 1 -S 5

Instances: 11195

Attributes: 9

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

MRK6

Test mode: split 50% train, remainder test

=== Classifier model (full training set) ===

Decision Table:

Number of training instances: 11195

Number of Rules : 302

Non matches covered by Majority class.

Best first search for feature set,

terminated after 5 non improving subsets.

Evaluation (for feature selection): CV (leave one out)

Feature set: 4,5,6,8,9

Time taken to build model: 5.94 seconds

=== Predictions on test split ===

inst#,	actual,	predicted,	error
1	61	41.563	-19.438
2	36	48.916	12.916
3	47	56.706	9.706
4	36	41.288	5.288
5	57	49.71	-7.29
6	53	47.556	-5.444
7	36	51.093	15.093
8	0	2.355	2.355
9	38	35.833	-2.167
10	42	45.295	3.295
11	45	53.302	8.302
12	43	44.25	1.25
13	60	48.745	-11.255
14	41	45.706	4.706
15	0	0	0
16	0	0	0
17	43	54.6	11.6
18	57	53.833	-3.167
19	46	46.889	0.889
20	60	48.745	-11.255
21	22	18	-4
22	50	53.886	3.886
23	45	54.955	9.955
24	54	51.162	-2.838
25	43	51.162	8.162
26	24	45.295	21.295

27	0	2.852	2.852
28	0	2.355	2.355
29	0	0	0
30	39	1.688	-37.313

…….Many more

=== Evaluation on test split ===

=== Summary ===

Correlation coefficient	0.7616
Mean absolute error	7.4246
Root mean squared error	11.9364
Relative absolute error	58.3284 %
Root relative squared error	65.681 %
Total Number of Instances	5598

Results for weka.classifiers.meta.Vote -B
weka.classifiers.rules.ZeroR

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: 5-fold cross-validation

=== Classifier model (full training set) ===

Vote combines the probability distributions of these base learners:

weka.classifiers.rules.ZeroR

Time taken to build model: 0.16 seconds

=== Cross-validation ===

=== Summary ===

Correlation coefficient	-0.0123
Mean absolute error	12.8967
Root mean squared error	18.0763
Relative absolute error	100 %
Root relative squared error	100 %
Total Number of Instances	11195

Results for weka.classifiers.rules.DecisionTable -X 1 -S 5

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: split 66% train, remainder test

=== Classifier model (full training set) ===

Decision Table:

Number of training instances: 11195

Number of Rules : 148

Non matches covered by Majority class.

Best first search for feature set,

terminated after 5 non improving subsets.

Evaluation (for feature selection): CV (leave one out)

Feature set: 4,5,6,7,8

Time taken to build model: 4.2 seconds

=== Evaluation on test split === === Summary ===

Correlation coefficient	0.8021
Mean absolute error	8.0574
Root mean squared error	10.8186
Relative absolute error	62.2703 %
Root relative squared error	59.7259 %
Total Number of Instances	3807

Results for weka.classifiers.trees.DecisionStump

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Decision Stump

Classifications

MRK3 <= 1.5: 4.208522212148686

MRK3 > 1.5: 49.46858898137138

MRK3 is missing: 45.009289861545334

Time taken to build model: 0.98 seconds

=== Cross-validation ===

=== Summary ===

Correlation coefficient	0.7449
Mean absolute error	8.8853
Root mean squared error	12.0586
Relative absolute error	68.8913 %
Root relative squared error	66.7062 %
Total Number of Instances	11195

Results for weka.classifiers.trees.DecisionStump

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: split 50% train, remainder test

=== Classifier model (full training set) ===

Decision Stump

Classifications

MRK3 <= 1.5: 4.208522212148686

MRK3 > 1.5: 49.46858898137138

MRK3 is missing: 45.009289861545334

Time taken to build model: 0.38 seconds

=== Evaluation on test split ===

=== Summary ===

Correlation coefficient	0.7464
Mean absolute error	8.8881
Root mean squared error	12.0702
Relative absolute error	68.7037 %
Root relative squared error	66.5509 %
Total Number of Instances	5598

Results for weka.classifiers.trees.DecisionStump

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: evaluate on training data

=== Classifier model (full training set) ===

Decision Stump

Classifications

MRK3 <= 1.5 : 4.208522212148686

MRK3 > 1.5 : 49.46858898137138

MRK3 is missing : 45.009289861545334

Time taken to build model: 0.39 seconds

=== Evaluation on training set ===

=== Summary ===

Correlation coefficient	0.7462
Mean absolute error	8.8714
Root mean squared error	12.0325
Relative absolute error	68.7992 %
Root relative squared error	66.5679 %
Total Number of Instances	11195

Results for weka.classifiers.functions.PaceRegression -E eb

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: evaluate on training data

=== Classifier model (full training set) ===

Pace Regression Model

MRK5 = 5.6505 + -0.0003 * SEATNO + 0.934 * CENTRE +
-0.0082 * COLLEGE + 0.2477 * MRK1 + 0.2106 * MRK2 +
0.3662 * MRK3 + 0.097 * MRK4

Time taken to build model: 0.89 seconds

=== Evaluation on training set ===

=== Summary ===

Correlation coefficient	0.7867
Mean absolute error	8.0399
Root mean squared error	11.1588
Relative absolute error	62.351 %
Root relative squared error	61.7344 %
Total Number of Instances	11195

Results for weka.classifiers.functions.PaceRegression -E eb

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Pace Regression Model

MRK5 = 5.6505 + -0.0003 * SEATNO + 0.934 * CENTRE +
-0.0082 * COLLEGE + 0.2477 * MRK1 + 0.2106 * MRK2 +
0.3662 * MRK3 + 0.097 * MRK4

Time taken to build model: 1 seconds

=== Cross-validation ===

=== Summary ===

Correlation coefficient	0.781
Mean absolute error	8.06
Root mean squared error	11.2901
Relative absolute error	62.4928 %
Root relative squared error	62.4551 %
Total Number of Instances	11195

Results for weka.classifiers.functions.PaceRegression -E eb

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: split 50% train, remainder test

=== Classifier model (full training set) ===

Pace Regression Model

MRK5 = 5.6505 + -0.0003 * SEATNO + 0.934 * CENTRE +

-0.0082 * COLLEGE + 0.2477 * MRK1 + 0.2106 * MRK2 +

0.3662 * MRK3 + 0.097 * MRK4

Time taken to build model: 0.31 seconds

=== Evaluation on test split ===

=== Summary ===

Correlation coefficient	0.7931
Mean absolute error	8.2308
Root mean squared error	11.0622
Relative absolute error	63.6231 %
Root relative squared error	60.9929 %
Total Number of Instances	5598

Results for weka.classifiers.lazy.LWL -U 0 -K -1 -W weka.classifiers.trees.DecisionStump

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: split 50% train, remainder test

=== Classifier model (full training set) ===

Locally weighted learning

==========

Using classifier: weka.classifiers.trees.DecisionStump

Using linear weighting kernels

Using all neighbours

Time taken to build model: 0.03 seconds

=== Evaluation on test split ===

=== Summary ===

Correlation coefficient	0.751
Mean absolute error	8.794
Root mean squared error	11.9777
Relative absolute error	67.9761 %
Root relative squared error	66.0405 %
Total Number of Instances	5598

Results for weka.classifiers.rules.M5Rules -M 4.0

Instances: 11195

Attributes: 9

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

MRK6

Test mode: split 50% train, remainder test

=== Classifier model (full training set) ===

M5 pruned model rules

(using smoothed linear models) :

Number of Rules : 8

Rule: 1

IF

MRK5 > 36.5

MRK3 <= 49.5

MRK2 > 36.5

MRK5 > 42.5

MRK2 <= 49.5

THEN

MRK6 =

0 * SEATNO

+ 0.0009 * MRK1

+ 0.2497 * MRK2
+ 0.2075 * MRK3
+ 0.055 * MRK4
+ 0.1328 * MRK5
+ 18.3028 [1774/65.196%]

Rule: 2
IF
MRK3 > 36.5
MRK5 > 47.5
MRK3 <= 58.5
MRK2 > 43.5
THEN

MRK6 =
0.0002 * SEATNO
+ 0.0599 * MRK1
+ 0.1621 * MRK2
+ 0.2659 * MRK3
+ 0.033 * MRK4
+ 0.1948 * MRK5
+ 14.7016 [1767/51.97%]

Rule: 3
IF
MRK3 > 36.5
THEN

MRK6 =

0.0003 * SEATNO
+ 0.0755 * MRK1
+ 0.2267 * MRK2
+ 0.1492 * MRK3
+ 0.0583 * MRK4
+ 0.3037 * MRK5
+ 9.107 [5422/73.841%]

Rule: 4
IF
MRK3 > 0.5
THEN

MRK6 =
0.0013 * MRK1
+ 0.0454 * MRK2
+ 0.1383 * MRK3
+ 0.03 * MRK4
+ 0.3463 * MRK5
+ 18.3759 [1135/68.96%]

Rule: 5
IF
SEATNO <= 2810.5
SEATNO <= 1807.5
THEN

MRK6 =
-0.0001 * SEATNO

+ 0.002 * MRK1
+ 0.2251 * MRK2
+ 0.0062 * MRK5
+ 1.7738 [205/193.153%]

Rule: 6
IF
SEATNO > 9229
THEN

MRK6 =
0.0024 * SEATNO
+ 0.0104 * MRK5
- 20.1582 [179/238.06%]

Rule: 7
IF
SEATNO > 5059.5
MRK4 <= 12.5
SEATNO <= 8096.5
SEATNO <= 7609.5
THEN

MRK6 =
0.0001 * SEATNO
+ 0.0018 * MRK1
+ 0.0011 * MRK2
+ 0.0103 * MRK5
+ 1.475 [142/196.327%]

Rule: 8

MRK6 =

0.064 * MRK1

+ 0.1852 * MRK5

+ 1.8001 [571/206.953%]

Time taken to build model: 107.63 seconds

=== Predictions on test split ===

inst#,	actual,	predicted,	error
1	61	42.755	-18.245
2	36	44.34	8.34
3	47	54.411	7.411
4	36	40.534	4.534
5	57	49.758	-7.242
6	53	51.861	-1.139
7	36	46.31	10.31
8	0	2.018	2.018
9	38	36.135	-1.865
10	42	45.011	3.011
11	45	50.146	5.146
12	43	50.3	7.3
13	60	60.018	0.018
14	41	44.52	3.52

15	0	2.128	2.128
16	0	0.397	0.397
17	43	45.684	2.684
18	57	58.834	1.834
19	46	52.262	6.262
20	60	56.731	-3.269
21	22	37.587	15.587
22	50	55.992	5.992
23	45	52.228	7.228
24	54	53.197	-0.803
25	43	45.238	2.238
26	24	42.686	18.686
27	0	1.584	1.584
28	0	6.732	6.732
29	0	0.791	0.791
30	39	2.045	-36.955
31	36	37.161	1.161
32	52	51.692	-0.308
33	45	45.386	0.386
34	9	20.237	11.237
35	52	49.566	-2.434
36	45	48.584	3.584
37	36	39.962	3.962
38	46	44.68	-1.32
39	29	39.92	10.92
40	48	18.642	-29.358
41	56	51.855	-4.145
42	57	47.385	-9.615
43	0	2.035	2.035

44	42	48.738	6.738
45	65	52.459	-12.541
46	46	45.792	-0.208
47	41	48.775	7.775
48	0	16.654	16.654
49	50	46.841	-3.159
50	49	42.773	-6.227

……And Many More

=== Evaluation on test split ===

=== Summary ===

Correlation coefficient	0.8325
Mean absolute error	6.5833
Root mean squared error	10.0794
Relative absolute error	51.7191 %
Root relative squared error	55.4631 %
Total Number of Instances	5598

Results for weka.classifiers.rules.DecisionTable -X 1 -S 5

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: split 50% train, remainder test

=== Classifier model (full training set) ===

Decision Table:

Number of training instances: 11195

Number of Rules : 148

Non matches covered by Majority class.

Best first search for feature set, terminated after 5 non improving subsets.

Evaluation (for feature selection): CV (leave one out)

Feature set: 4,5,6,7,8

Time taken to build model: 3.53 seconds

=== Evaluation on test split ====== Summary ===

Correlation coefficient	0.7953
Mean absolute error	8.0904
Root mean squared error	11.0131
Relative absolute error	62.5379 %
Root relative squared error	60.7224 %
Total Number of Instances	5598

Results for weka.classifiers.rules.M5Rules -M 4.0

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

M5 pruned model rules

(using smoothed linear models) :

Number of Rules : 28

Rule: 1

IF

MRK1 > 36.5

MRK3 > 48.5

MRK3 <= 59.5

MRK4 > 36.5

MRK2 <= 45.5

THEN

MRK5 =

-0.0002 * SEATNO

+ 0.1518 * MRK1

+ 0.1666 * MRK2
+ 0.361 * MRK3
+ 0.0792 * MRK4
+ 16.2836 [1029/66.517%]

Rule: 2
IF
MRK3 > 36.5
MRK3 > 48.5
MRK4 > 36.5
MRK4 <= 65.5
THEN

MRK5 =
-0.0001 * SEATNO
+ 0.0588 * MRK1
+ 0.2315 * MRK2
+ 0.2997 * MRK3
+ 0.0753 * MRK4
+ 20.1696 [2260/64.062%]

Rule: 3
IF
MRK3 > 36.5
MRK3 <= 50.5
MRK2 > 40.5
SEATNO > 2404
MRK4 <= 54.5

MRK3 > 42.5

THEN

MRK5 =

0 * SEATNO

+ 0.049 * MRK1

+ 0.2381 * MRK2

+ 0.0164 * MRK3

+ 0.1138 * MRK4

+ 29.8132 [775/57.79%]

Rule: 4

IF

MRK3 > 26.5

MRK3 <= 48.5

MRK4 > 38.5

THEN

MRK5 =

0 * SEATNO

+ 0.0787 * MRK1

+ 0.0704 * MRK2

+ 0.4266 * MRK3

+ 0.001 * MRK4

+ 22.7605 [2029/65.511%]

Rule: 5

IF

MRK3 > 1.5

MRK3 <= 48.5

MRK4 > 21.5

MRK3 > 37.5

THEN

MRK5 =

0.0001 * SEATNO

+ 0.0893 * MRK1

+ 0.1829 * MRK2

+ 0.391 * MRK3

+ 0.002 * MRK4

+ 17.1887 [1051/50.506%]

Rule: 6

IF

MRK3 > 1.5

MRK3 <= 48.5

MRK4 > 15.5

MRK3 > 32.5

THEN

MRK5 =

0 * SEATNO

+ 0.112 * MRK1

+ 0.0035 * MRK2

+ 0.468 * MRK3

+ 0.0029 * MRK4

+ 20.1515 [770/46.88%]

Rule: 7

IF

MRK3 > 1.5

MRK3 > 48.5

MRK4 > 51

SEATNO > 2951.5

SEATNO <= 8306.5

THEN

MRK5 =

0 * SEATNO

+ 0.0963 * MRK1

+ 0.1619 * MRK2

+ 0.2355 * MRK3

+ 0.0043 * MRK4

+ 30.3818 [272/30.672%]

Rule: 8

IF

MRK3 > 1.5

MRK3 <= 48.5

MRK4 > 9.5

SEATNO > 1579

MRK2 <= 36.5

THEN

MRK5 =

0 * SEATNO

+ 0.179 * MRK1

+ 0.1237 * MRK2

+ 0.1459 * MRK3

+ 0.0047 * MRK4

+ 23.5027 [258/41.103%]

Rule: 9

IF

MRK3 <= 1.5

MRK4 > 15.5

SEATNO > 1229.5

SEATNO <= 8090

THEN

MRK5 =

0 * SEATNO

+ 0.3145 * MRK1

+ 0.2756 * MRK2

+ 0.0037 * MRK3

- 0.0013 * MRK4

+ 0.5335 [232/40.793%]

Rule: 10

IF

MRK3 <= 1.5

MRK1 > 13.5

THEN

MRK5 =

0 * SEATNO

- 0.0001 * MRK1

+ 0.0008 * MRK2

+ 0.0045 * MRK3

- 0.0065 * MRK4

+ 2.4061 [184/37.132%]

Rule: 11

IF

MRK1 <= 10.5

SEATNO <= 7926.5

SEATNO <= 5643.5

SEATNO > 1043.5

THEN

MRK5 =

0 * SEATNO

+ 0.257 * MRK1

+ 0.0024 * MRK2

+ 0.145 * MRK3

+ 0.0019 * MRK4

+ 4.467 [268/62.226%]

Rule: 12

IF

MRK1 > 10.5

MRK3 > 48.5

MRK4 <= 51

MRK2 > 38.5

MRK3 <= 60.5

SEATNO > 6174.5

THEN

MRK5 =

0.0014 * SEATNO

+ 0.013 * MRK1

+ 0.183 * MRK2

+ 0.378 * MRK3
+ 0.0259 * MRK4
+ 8.7874 [178/36.406%]

Rule: 13

IF

MRK1 > 10.5
MRK3 > 48.5
MRK4 <= 51
MRK4 > 30.5
MRK3 <= 59.5

THEN

MRK5 =

-0.0004 * SEATNO
+ 0.0159 * MRK1
+ 0.021 * MRK2
+ 0.0028 * MRK3
+ 0.0106 * MRK4
+ 50.3467 [375/48.594%]

Rule: 14

IF

MRK1 > 17.5
MRK3 > 48.5
MRK4 > 30.5
MRK4 > 51
SEATNO <= 9116
SEATNO > 1845

THEN

MRK5 =

0.0001 * SEATNO

+ 0.2649 * MRK1

+ 0.0169 * MRK2

+ 0.043 * MRK3

+ 0.4579 * MRK4

+ 12.6258 [132/58.379%]

Rule: 15

IF

MRK1 > 10.5

MRK3 > 48.5

MRK4 > 30.5

MRK2 > 51.5

THEN

MRK5 =

0.0002 * SEATNO

+ 0.0291 * MRK1

+ 0.0236 * MRK2

+ 0.4721 * MRK3

+ 0.0567 * MRK4

+ 22.3854 [214/42.676%]

Rule: 16

IF

MRK1 > 5.5

MRK3 > 47.5

MRK4 > 22.5

THEN

MRK5 =

0 * SEATNO

+ 0.112 * MRK1

+ 0.0207 * MRK2

+ 0.0018 * MRK3

+ 0.0092 * MRK4

+ 47.8593 [254/38.116%]

Rule: 17

IF

MRK3 > 1

MRK2 > 22.5

MRK3 <= 47.5

SEATNO > 961.5

MRK4 <= 30.5

THEN

MRK5 =

0 * SEATNO

+ 0.0431 * MRK1

- 0.0539 * MRK2

+ 0.2747 * MRK3

+ 29.7429 [115/51.486%]

Rule: 18

IF

MRK1 > 10.5

MRK2 > 20.5

SEATNO > 956

THEN

MRK5 =

0.0001 * SEATNO
+ 0.0404 * MRK1
+ 0.0086 * MRK2
+ 0.0057 * MRK3
- 0.105 * MRK4
+ 45.766 [180/51.601%]

Rule: 19

IF

MRK3 > 10.5
MRK2 > 4.5

THEN

MRK5 =

0.0007 * SEATNO
+ 0.2289 * MRK1
+ 0.0592 * MRK2
+ 0.0033 * MRK3
+ 0.0051 * MRK4
+ 21.7413 [122/95.883%]

Rule: 20

IF

MRK4 > 17.5
SEATNO <= 9213
SEATNO > 438.5

THEN

MRK5 =
0 * SEATNO
- 0.0106 * MRK2
- 0.0233 * MRK4
+ 1.5846 [70/31.232%]

Rule: 21
IF
SEATNO > 9970.5
THEN

MRK5 =
-0.0001 * SEATNO
- 0.0136 * MRK2
- 0.0244 * MRK4
+ 3.4584 [93/77.322%]

Rule: 22
IF
SEATNO > 649.5
MRK2 <= 27
SEATNO <= 9436.5
SEATNO <= 7926.5
SEATNO <= 7526
THEN

MRK5 =

0.0001 * SEATNO
+ 1.3197 * MRK2
- 0.0256 * MRK4
+ 5.5112 [87/101.555%]

Rule: 23
IF
MRK2 <= 1
SEATNO > 7553.5
SEATNO <= 9625.5
THEN
MRK5 =
0.0003 * SEATNO
- 0.0141 * MRK2
- 0.0375 * MRK3
- 0.0616 * MRK4
+ 2.9138 [86/89.521%]

Rule: 24
IF
MRK2 > 1
THEN
MRK5 =
-0.0007 * SEATNO
- 0.0684 * MRK2
- 0.0591 * MRK4
+ 10.6846 [65/63.783%]

Rule: 25
IF
MRK4 <= 2.5

SEATNO > 404.5

THEN

MRK5 =

-0.001 * SEATNO

- 0.0726 * MRK4

+ 29.7591 [57/104.214%]

Rule: 26

IF

SEATNO > 422

THEN

MRK5 =

+ 0.0427 [21/0%]

Rule: 27

IF

SEATNO <= 401.5

THEN

MRK5 =

+ 0.1149 [14/0%]

Rule: 28

MRK5 =

0.2204 * SEATNO

- 88.8163 [4/10.433%]

Time taken to build model: 251.69 seconds

Results for weka.classifiers.rules.M5Rules -M 4.0

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: evaluate on training data

=== Classifier model (full training set) ===

M5 pruned model rules

(using smoothed linear models) :

Number of Rules : 40

Rule: 1

IF

MRK3 > 36.5

MRK4 > 39.5

MRK2 <= 47.5

SEATNO <= 8276.5

SEATNO <= 6554

THEN

MRK1 =

0.0003 * SEATNO
+ 0.2788 * MRK2
+ 0.0984 * MRK3
+ 0.0021 * MRK4
+ 0.1041 * MRK5
+ 23.2161 [1567/77.016%]

Rule: 2

IF

MRK5 > 36.5
MRK4 <= 40.5
MRK3 > 38.5
MRK2 > 42.5

THEN

MRK1 =

-0.0001 * SEATNO
+ 0.2212 * MRK2
+ 0.0905 * MRK3
+ 0.1242 * MRK4
+ 0.1409 * MRK5
+ 17.659 [1304/81.45%]

Rule: 3

IF

MRK5 <= 36.5
MRK3 > 0.5
MRK4 > 19.5

MRK4 <= 43.5

THEN

MRK1 =

-0.0002 * SEATNO

+ 0.0861 * MRK2

+ 0.1412 * MRK3

+ 0.0027 * MRK4

+ 0.1082 * MRK5

+ 26.4041 [702/85.278%]

Rule: 4

IF

MRK5 > 37.5

MRK4 > 40.5

MRK3 > 48.5

MRK4 > 64.5

THEN

MRK1 =

0.0006 * SEATNO

+ 0.1261 * MRK2

+ 0.1936 * MRK3

+ 0.0049 * MRK4

+ 0.2211 * MRK5

+ 19.8301 [632/70.121%]

Rule: 5

IF

MRK3 > 28.5

MRK4 <= 40.5

MRK4 > 21.5

MRK3 <= 47.5

SEATNO <= 7020

THEN

MRK1 =

0 * SEATNO

+ 0.1357 * MRK2

+ 0.0026 * MRK3

+ 0.0089 * MRK4

+ 0.1111 * MRK5

+ 29.7686 [690/74.983%]

Rule: 6

IF

MRK3 > 22.5

MRK4 > 40.5

MRK3 <= 51.5

MRK2 > 46.5

SEATNO <= 8279

THEN

MRK1 =

-0.0003 * SEATNO

- 0.0895 * MRK2

+ 0.1323 * MRK3

+ 0.0591 * MRK4

+ 0.084 * MRK5
+ 39.4278 [653/69.099%]

Rule: 7
IF
MRK3 > 10.5
MRK4 > 40.5
MRK2 <= 47.5
SEATNO > 8276.5
SEATNO > 9005.5
THEN

MRK1 =
0.0031 * SEATNO
+ 0.3591 * MRK2
+ 0.1577 * MRK3
+ 0.2238 * MRK4
+ 0.0138 * MRK5
- 20.2488 [562/59.586%]

Rule: 8
IF
MRK3 > 1
MRK4 > 40.5
SEATNO <= 8279
MRK2 > 47.5
THEN

MRK1 =

0 * SEATNO
+ 0.0036 * MRK2
+ 0.2984 * MRK3
+ 0.001 * MRK4
+ 0.0764 * MRK5
+ 27.6164 [892/64.786%]

Rule: 9
IF
MRK3 > 1
MRK4 > 40.5
SEATNO > 8276.5
SEATNO > 8602
SEATNO > 8862
SEATNO > 9087
THEN

MRK1 =
0.0058 * SEATNO
- 0.0076 * MRK2
+ 0.14 * MRK3
+ 0.2861 * MRK4
+ 0.0116 * MRK5
- 29.903 [372/50.986%]

Rule: 10
IF
MRK3 > 1
MRK4 > 24.5

SEATNO <= 8281.5

SEATNO <= 7107.5

SEATNO <= 6648.5

THEN

MRK1 =

0 * SEATNO

+ 0.1982 * MRK2

+ 0.0551 * MRK3

+ 0.0027 * MRK4

+ 0.0079 * MRK5

+ 31.0087 [572/59.62%]

Rule: 11

IF

MRK3 > 1

MRK4 <= 40.5

MRK4 > 13.5

MRK3 <= 47.5

THEN

MRK1 =

0 * SEATNO

+ 0.0018 * MRK2

+ 0.0043 * MRK3

+ 0.0883 * MRK4

+ 0.311 * MRK5

+ 19.0781 [759/61.547%]

Rule: 12

IF

MRK3 > 1.5

MRK4 > 20.5

SEATNO <= 8276.5

SEATNO > 7025

THEN

MRK1 =

0.0044 * SEATNO

+ 0.5168 * MRK2

+ 0.0096 * MRK3

+ 0.0066 * MRK4

+ 0.0191 * MRK5

- 15.4801 [345/47.519%]

Rule: 13

IF

MRK3 > 1.5

MRK4 <= 24.5

MRK4 > 0.5

THEN

MRK1 =

0 * SEATNO

+ 0.005 * MRK2

- 0.0326 * MRK3

+ 0.347 * MRK4

+ 0.2427 * MRK5

+ 20.3462 [292/54.116%]

Rule: 14

IF

MRK3 <= 2

MRK4 <= 12.5

MRK2 <= 1

SEATNO > 2910

SEATNO > 5595

THEN

MRK1 =

0 * SEATNO

+ 0.0004 * MRK2

+ 0.0036 * MRK3

+ 0.0005 * MRK4

+ 0.0049 * MRK5

+ 8.8665 [256/63.957%]

Rule: 15

IF

MRK3 > 2

MRK4 > 36.5

SEATNO <= 8602

SEATNO > 8298.5

THEN

MRK1 =

0.0004 * SEATNO

+ 0.1276 * MRK2
+ 0.209 * MRK3
+ 0.0131 * MRK4
+ 0.0392 * MRK5
+ 41.1412 [195/33.26%]

Rule: 16
IF
MRK3 <= 2
MRK4 > 12.5
SEATNO > 1231.5
SEATNO <= 5835.5
THEN

MRK1 =
0 * SEATNO
+ 0.0896 * MRK2
+ 0.0038 * MRK3
- 0 * MRK4
+ 0.2168 * MRK5
+ 1.7801 [164/39.645%]

Rule: 17
IF
MRK3 <= 2
SEATNO <= 6705.5
SEATNO > 479.5
SEATNO > 1242.5
SEATNO > 2913

SEATNO > 3760.5

THEN

MRK1 =

0.0074 * SEATNO

+ 0.0018 * MRK2

+ 0.0042 * MRK3

+ 0.0036 * MRK4

+ 0.0082 * MRK5

- 26.1141 [123/69.241%]

Rule: 18

IF

MRK3 <= 2

SEATNO <= 5258

SEATNO > 478.5

SEATNO > 805

SEATNO <= 2913

THEN

MRK1 =

-0.0023 * SEATNO

+ 0.0029 * MRK2

+ 0.0047 * MRK3

+ 0.0042 * MRK4

+ 0.0892 * MRK5

+ 9.831 [136/50.377%]

Rule: 19

IF

MRK5 > 14.5

MRK4 > 36.5

SEATNO <= 8862

SEATNO <= 8757

THEN

MRK1 =

-0.0002 * SEATNO

+ 0.0127 * MRK2

+ 0.0307 * MRK3

+ 0.016 * MRK4

+ 0.2573 * MRK5

+ 32.6602 [180/52.263%]

Rule: 20

IF

MRK3 > 2

MRK4 > 12.5

SEATNO > 9007

THEN

MRK1 =

0.0021 * SEATNO

+ 0.397 * MRK2

+ 0.0061 * MRK3

+ 0.0198 * MRK4

+ 0.2239 * MRK5

- 6.9822 [127/42.58%]

Rule: 21

IF

MRK3 > 2

MRK4 <= 18

THEN

MRK1 =

0.0001 * SEATNO

+ 0.0289 * MRK2

+ 0.0091 * MRK3

+ 0.0308 * MRK4

+ 0.0833 * MRK5

+ 10.4311 [78/84.7%]

Rule: 22

IF

MRK3 <= 11.5

SEATNO <= 478.5

SEATNO > 402.5

THEN

MRK1 =

-0.0021 * SEATNO

+ 0.0806 * MRK2

+ 0.0297 * MRK3

- 0.0096 * MRK4

+ 2.8422 [74/0%]

Rule: 23

IF

MRK3 > 11.5

SEATNO > 8866.5

THEN

MRK1 =

-0.0001 * SEATNO

+ 0.0041 * MRK2

+ 0.1711 * MRK3

+ 0.0087 * MRK4

+ 0.0109 * MRK5

+ 50.9971 [73/30.224%]

Rule: 24

IF

MRK3 <= 17

SEATNO > 5258

SEATNO > 7589.5

SEATNO > 8017.5

SEATNO <= 9462

THEN

MRK1 =

-0.0002 * SEATNO

+ 0.0299 * MRK3

+ 4.613 [62/36.064%]

Rule: 25

IF

MRK3 > 14.5

SEATNO > 8591.5

THEN

MRK1 =

-0.0015 * SEATNO

- 0.0963 * MRK2

+ 0.0909 * MRK3

- 0.0425 * MRK4

+ 0.1207 * MRK5

+ 43.9315 [57/65.117%]

Rule: 26

IF

MRK4 <= 30

SEATNO <= 5258

SEATNO <= 3693.5

SEATNO > 2971

THEN

MRK1 =

-0.0003 * SEATNO

+ 0.061 * MRK3

+ 0.0058 * MRK4

+ 9.2458 [34/40.436%]

Rule: 27

IF

MRK4 > 30

MRK3 > 24

MRK5 <= 57.5

THEN

MRK1 =

-0.0008 * SEATNO

+ 0.0796 * MRK3

+ 0.2665 * MRK5

+ 41.0597 [33/42.753%]

Rule: 28

IF

SEATNO > 5258

SEATNO > 7589.5

SEATNO > 9522.5

SEATNO <= 10923

THEN

MRK1 =

-0.0006 * SEATNO

+ 0.0324 * MRK2

+ 0.4013 * MRK3

+ 9.4208 [39/44.093%]

Rule: 29

IF

SEATNO <= 7601

SEATNO > 5258

THEN

MRK1 =

-0.0005 * SEATNO

- 0.0088 * MRK2

+ 0.0823 * MRK3

+ 6.5827 [74/37.884%]

Rule: 30

IF

MRK4 <= 28.5

SEATNO <= 1859.5

SEATNO > 507

THEN

MRK1 =

-0.0001 * SEATNO

+ 0.1052 * MRK3

+ 0.0266 * MRK4

+ 9.7975 [20/0%]

Rule: 31

IF

SEATNO <= 7939.5

SEATNO > 480.5

SEATNO <= 513

THEN

MRK1 =

-0.0012 * SEATNO
- 0.0246 * MRK2
+ 0.1175 * MRK3
+ 36.9417 [25/17.979%]

Rule: 32
IF
SEATNO <= 8207.5
SEATNO <= 7610.5
SEATNO <= 3728.5
SEATNO > 1756
THEN

MRK1 =
-0.0126 * SEATNO
+ 0.1298 * MRK3
+ 64.441 [21/55.047%]

Rule: 33
IF
SEATNO <= 8207.5
SEATNO <= 7768.5
SEATNO > 2163
THEN

MRK1 =
0.0013 * SEATNO
+ 0.1719 * MRK3
+ 22.9525 [19/44.056%]

Rule: 34

IF

SEATNO <= 8207.5

SEATNO > 500.5

THEN

MRK1 =

-0.0004 * SEATNO

+ 0.2715 * MRK3

+ 5.2943 [19/0%]

Rule: 35

IF

MRK3 > 24

THEN

MRK1 =

-0.0006 * SEATNO

+ 0.4295 * MRK3

+ 0.3991 * MRK4

+ 31.4461 [11/28.473%]

Rule: 36

IF

SEATNO > 383

MRK4 <= 18

SEATNO <= 440.5

THEN

MRK1 =

-0.0004 * SEATNO

+ 0.4178 * MRK4

+ 5.9756 [8/0%]

Rule: 37

IF

SEATNO <= 383

THEN

MRK1 =

+ 11.3182 [7/0%]

Rule: 38

IF

SEATNO <= 10974.5

SEATNO <= 9468.5

THEN

MRK1 =

-0.0018 * SEATNO

- 0.1906 * MRK2

+ 44.84 [7/11.326%]

Rule: 39

IF

SEATNO <= 10974.5

THEN

MRK1 =

-0.3247 * MRK2

+ 23.1427 [6/68.151%]

Rule: 40

MRK1 =

+ 0 [5]

Time taken to build model: 312.97 seconds

=== Evaluation on training set ===

=== Summary ===

Correlation coefficient	0.7771
Mean absolute error	8.2597
Root mean squared error	11.0031
Relative absolute error	64.6061 %
Root relative squared error	62.9606 %
Total Number of Instances	11195

Results for weka.classifiers.rules.M5Rules -M 4.0

Instances: 11195

Attributes: 6

SEX

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: evaluate on training data

=== Classifier model (full training set) ===

M5 pruned model rules

(using smoothed linear models) :

Number of Rules : 12

Rule: 1

IF

MRK3 > 36.5

MRK4 <= 39.5

THEN

MRK1 =

2.1265 * SEX=F

+ 0.2101 * MRK2

+ 0.0005 * MRK3

+ 0.257 * MRK4

+ 0.202 * MRK5

+ 13.15 [3206/93.817%]

Rule: 2

IF

MRK3 > 36.5

MRK2 > 47.5

MRK4 <= 58.5

THEN

MRK1 =

2.8324 * SEX=F

+ 0.1136 * MRK2

+ 0.1776 * MRK3

+ 0.1695 * MRK4

+ 0.0017 * MRK5

+ 23.6853 [1609/77.035%]

Rule: 3

IF

MRK3 > 20.5

MRK3 > 39.5

MRK2 <= 48.5

SEX=F <= 0.5

THEN

MRK1 =

0.0669 * SEX=F

+ 0.362 * MRK2

+ 0.0036 * MRK3

+ 0.0618 * MRK4

+ 0.1017 * MRK5
+ 21.79 [1599/67.435%]

Rule: 4
IF
MRK3 > 0.5
MRK3 > 39.5
MRK2 > 47.5
THEN

MRK1 =
2.5872 * SEX=F
+ 0.0012 * MRK2
+ 0.1845 * MRK3
+ 0.1807 * MRK4
+ 0.1038 * MRK5
+ 23.9427 [1028/61.443%]

Rule: 5
IF
MRK3 > 0.5
MRK4 > 39.5
SEX=F > 0.5
MRK5 <= 60.5
THEN

MRK1 =
0.0634 * SEX=F
- 0.0015 * MRK2

+ 0.172 * MRK3
+ 0.2064 * MRK4
+ 0.007 * MRK5
+ 27.7099 [920/64.349%]

Rule: 6
IF
MRK3 > 0.5
MRK4 <= 40.5
MRK4 > 19.5
THEN

MRK1 =
3.1927 * SEX=F
+ 0.0003 * MRK2
+ 0.1362 * MRK3
+ 0.0052 * MRK4
+ 0.0937 * MRK5
+ 28.234 [802/58.642%]

Rule: 7
IF
MRK3 <= 1.5
MRK4 <= 12.5
MRK2 <= 1
THEN

MRK1 =
-4.1797 * SEX=F

- 0.0009 * MRK2
+ 0.0085 * MRK3
- 0.0015 * MRK4
- 0.0005 * MRK5
+ 10.1159 [558/76.116%]

Rule: 8
IF
MRK3 <= 1.5
THEN

MRK1 =
-0.0964 * SEX=F
+ 0.0585 * MRK2
+ 0.0112 * MRK3
+ 0.0645 * MRK4
+ 0.3122 * MRK5
+ 1.3371 [544/59.486%]

Rule: 9
IF
MRK4 > 30
SEX=F <= 0.5
MRK2 > 37.5
THEN

MRK1 =
0.1534 * SEX=F
- 0.0678 * MRK2

+ 0.2626 * MRK3
- 0.0018 * MRK4
+ 0.0115 * MRK5
+ 37.0672 [259/99.615%]

Rule: 10
IF
MRK5 > 60.5
THEN

MRK1 =
0.2028 * MRK3
+ 0.0015 * MRK4
+ 0.5388 * MRK5
+ 6.616 [246/90.833%]

Rule: 11
IF
MRK4 > 30
THEN

MRK1 =
-0.3024 * SEX=F
+ 0.2868 * MRK2
- 0.025 * MRK4
+ 0.0204 * MRK5
+ 30.3767 [216/109.023%]

Rule: 12

MRK1 =

-0.1497 * MRK3

+ 0.3215 * MRK5

+ 22.1561 [208/109.374%]

Time taken to build model: 154.42 seconds

=== Evaluation on training set ===

=== Summary ===

Correlation coefficient	0.7427
Mean absolute error	8.8643
Root mean squared error	11.7022
Relative absolute error	69.3346 %
Root relative squared error	66.9605 %
Total Number of Instances	11195

Results for weka.classifiers.rules.M5Rules -M 4.0

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: evaluate on training data

=== Classifier model (full training set) ===

M5 pruned model rules

(using smoothed linear models) :

Number of Rules : 15

Rule: 1

IF

MRK5 <= 36.5

MRK5 > 0.5

THEN

MRK3 =

0 * SEATNO

+ 0.2442 * MRK1

+ 0.108 * MRK2

+ 0.0003 * MRK4

+ 0.0111 * MRK5
+ 24.3058 [1187/159.519%]

Rule: 2
IF
MRK5 > 40.5
MRK5 <= 54.5
MRK1 > 36.5
MRK2 > 38.5
THEN

MRK3 =
0.0001 * SEATNO
+ 0.0949 * MRK1
+ 0.1071 * MRK2
+ 0.0435 * MRK4
+ 0.5004 * MRK5
+ 12.6915 [2362/64.766%]

Rule: 3
IF
MRK5 <= 38.5
THEN

MRK3 =
0 * SEATNO
+ 0.1566 * MRK1
+ 0.0897 * MRK2
+ 0.0006 * MRK4

+ 0.7785 * MRK5
+ 2.1163 [1565/118.989%]

Rule: 4
IF
MRK5 <= 54.5
MRK4 > 36.5
THEN

MRK3 =
0.0002 * SEATNO
+ 0.1312 * MRK1
+ 0.2796 * MRK2
+ 0.03 * MRK4
+ 0.4841 * MRK5
+ 4.0571 [1477/94.613%]

Rule: 5
IF
MRK5 > 54.5
MRK2 <= 50.5
MRK2 > 37.5
SEATNO > 4045.5
SEATNO > 7017.5
THEN

MRK3 =
0 * SEATNO
+ 0.1759 * MRK1

+ 0.2116 * MRK2
+ 0.0022 * MRK4
+ 0.1754 * MRK5
+ 23.8246 [531/95.533%]

Rule: 6
IF
MRK5 > 54.5
MRK2 <= 45.5
MRK2 <= 37.5
THEN

MRK3 =
0 * SEATNO
+ 0.1747 * MRK1
+ 0.5674 * MRK2
+ 0.0723 * MRK4
+ 0.0053 * MRK5
+ 17.9783 [462/97.673%]

Rule: 7
IF
MRK5 > 54.5
SEATNO <= 4017.5
MRK2 <= 55.5
SEATNO > 1852
THEN

MRK3 =

-0.0001 * SEATNO
+ 0.1357 * MRK1
+ 0.3924 * MRK2
+ 0.0487 * MRK4
+ 0.0034 * MRK5
+ 22.9117 [396/77.955%]

Rule: 8
IF
MRK5 > 54.5
MRK1 <= 54.5
SEATNO <= 5694.5
THEN

MRK3 =
0.0004 * SEATNO
+ 0.0999 * MRK1
+ 0.1397 * MRK2
+ 0.0952 * MRK4
+ 0.0069 * MRK5
+ 37.6842 [709/69.568%]

Rule: 9
IF
MRK5 > 54.5
MRK5 <= 63.5
SEATNO <= 7562
THEN

MRK3 =

0.0008 * SEATNO

+ 0.0035 * MRK1

+ 0.2239 * MRK2

+ 0.0538 * MRK4

+ 0.3161 * MRK5

+ 22.1153 [542/71.947%]

Rule: 10

IF

MRK5 > 54.5

MRK5 <= 63.5

THEN

MRK3 =

0 * SEATNO

+ 0.0961 * MRK1

+ 0.295 * MRK2

+ 0.141 * MRK4

+ 0.0158 * MRK5

+ 26.0931 [333/71.487%]

Rule: 11

IF

MRK5 <= 59

MRK1 <= 36.5

SEATNO <= 9353.5

MRK4 > 20.5

MRK2 > 36.5

THEN

MRK3 =

0 * SEATNO
+ 0.0249 * MRK1
+ 0.02 * MRK2
+ 0.0287 * MRK4
+ 0.316 * MRK5
+ 26.1817 [277/78.271%]

Rule: 12

IF

MRK5 <= 59

THEN

MRK3 =

0.0004 * SEATNO
+ 0.1424 * MRK1
+ 0.4954 * MRK2
+ 0.228 * MRK4
+ 0.3671 * MRK5
- 7.3769 [933/110.433%]

Rule: 13

IF

SEATNO <= 8757.5
SEATNO > 5704.5

THEN

MRK3 =

-0.0027 * SEATNO

+ 0.104 * MRK1

+ 0.1136 * MRK2

+ 0.0033 * MRK4

+ 0.172 * MRK5

+ 60.4102 [184/115.759%]

Rule: 14

IF

MRK2 <= 64.5

THEN

MRK3 =

0.1243 * MRK1

- 0.1354 * MRK2

+ 59.575 [172/121.317%]

Rule: 15

MRK3 =

0.0721 * MRK4

+ 60.4318 [65/121.887%]

Time taken to build model: 255.38 seconds

=== Evaluation on training set ===

=== Summary ===

Correlation coefficient	0.7707
Mean absolute error	7.0526
Root mean squared error	12.4607
Relative absolute error	53.8001 %
Root relative squared error	63.7171 %
Total Number of Instances	11195

Results for weka.classifiers.rules.M5Rules -M 4.0

Instances: 11195

Attributes: 8

SEATNO

CENTRE

COLLEGE

MRK1

MRK2

MRK3

MRK4

MRK5

Test mode: evaluate on training data

=== Classifier model (full training set) ===

M5 pruned model rules

(using smoothed linear models) :

Number of Rules : 26

Rule: 1

IF

MRK3 > 38.5

MRK2 <= 46.5

THEN

MRK4 =

0.0001 * SEATNO

+ 0.2543 * MRK1

+ 0.4119 * MRK2
+ 0.0001 * MRK3
+ 0.1912 * MRK5
+ 6.0222 [4514/95.827%]

Rule: 2
IF
MRK3 > 38.5
THEN

MRK4 =
-0.0005 * SEATNO
+ 0.294 * MRK1
+ 0.2769 * MRK2
+ 0.1214 * MRK3
+ 0.2238 * MRK5
+ 5.9593 [3650/79.373%]

Rule: 3
IF
MRK3 > 1.5
MRK5 <= 37.5
SEATNO > 4399.5
THEN

MRK4 =
0 * SEATNO
+ 0.3736 * MRK1
+ 0.0214 * MRK2

+ 0.2783 * MRK3
+ 0.4063 * MRK5
- 2.3794 [470/100.317%]

Rule: 4
IF
MRK3 > 1.5
MRK5 > 38.5
MRK2 > 39.5
THEN

MRK4 =
0 * SEATNO
+ 0.2439 * MRK1
+ 0.0116 * MRK2
+ 0.0043 * MRK3
+ 0.1898 * MRK5
+ 22.2313 [413/60.869%]

Rule: 5
IF
MRK3 > 5.5
MRK5 > 37.5
MRK5 <= 42.5
THEN

MRK4 =
0 * SEATNO
+ 0.2431 * MRK1

+ 0.2892 * MRK2
+ 0.0019 * MRK3
+ 0.0167 * MRK5
+ 15.5033 [300/64.631%]

Rule: 6
IF
MRK3 > 5.5
THEN

MRK4 =
0 * SEATNO
+ 0.8445 * MRK2
- 0.349 * MRK3
+ 0.1968 * MRK5
+ 14.4154 [738/164.547%]

Rule: 7
IF
SEATNO > 7935
SEATNO > 8986.5
THEN

MRK4 =
0 * SEATNO
- 0.0121 * MRK1
- 0.0077 * MRK2
+ 8.1355 [187/83.329%]

Rule: 8

IF

SEATNO > 1242.5

SEATNO > 7546.5

SEATNO > 7597

THEN

MRK4 =

0.0054 * SEATNO

- 0.0393 * MRK1

- 0.0083 * MRK2

- 31.0206 [136/92.847%]

Rule: 9

IF

SEATNO > 1238

SEATNO > 5805.5

SEATNO <= 7546.5

THEN

MRK4 =

-0.0104 * SEATNO

- 0.0181 * MRK1

- 0.009 * MRK2

+ 83.7532 [112/87.628%]

Rule: 10

IF

SEATNO > 650.5

SEATNO > 2857.5
MRK2 <= 22
MRK1 <= 5.5
SEATNO > 3717
SEATNO <= 5804.5
SEATNO <= 5059.5
THEN

MRK4 =
0.0008 * SEATNO
- 0.0427 * MRK1
- 0.0101 * MRK2
- 0.1797 * MRK5
+ 9.0922 [83/80.937%]

Rule: 11
IF
SEATNO > 597.5
SEATNO > 2857.5
MRK1 > 17.5
THEN

MRK4 =
0.0021 * SEATNO
- 0.0877 * MRK1
- 0.0413 * MRK2
- 0.005 * MRK5
+ 0.62 [61/67.753%]

Rule: 12

IF

SEATNO <= 597.5

SEATNO > 420.5

SEATNO <= 470.5

THEN

MRK4 =

-0.0121 * SEATNO

- 0.1063 * MRK1

+ 0.0363 * MRK2

- 0.0216 * MRK5

+ 43.9637 [50/43.496%]

Rule: 13

IF

SEATNO > 2857.5

MRK2 > 22

SEATNO > 2912

SEATNO <= 7582.5

THEN

MRK4 =

-0.0005 * SEATNO

+ 0.0124 * MRK1

- 0.041 * MRK2

- 0.0061 * MRK5

+ 14.3033 [40/72.984%]

Rule: 14

IF

SEATNO > 2857.5

SEATNO > 6676

THEN

MRK4 =

0.0007 * SEATNO

+ 0.0142 * MRK1

- 0.0738 * MRK2

+ 28.6168 [43/93.797%]

Rule: 15

IF

SEATNO > 2857.5

SEATNO > 2898.5

SEATNO > 3296.5

SEATNO > 5076

THEN

MRK4 =

-0.0017 * SEATNO

+ 0.0161 * MRK1

- 0.1206 * MRK2

+ 23.1887 [36/74.548%]

Rule: 16

IF

SEATNO > 2857.5

SEATNO > 2888
SEATNO > 2911.5
SEATNO <= 3696.5
THEN

MRK4 =
0.0072 * SEATNO
+ 0.0252 * MRK1
- 8.6854 [39/98.422%]

Rule: 17
IF
SEATNO <= 2857.5
SEATNO <= 1825.5
SEATNO > 491.5
SEATNO > 646.5
SEATNO <= 1043.5
THEN

MRK4 =
0.0066 * SEATNO
+ 0.0786 * MRK1
+ 0.0628 * MRK2
- 0.7783 [34/40.202%]

Rule: 18
IF
SEATNO <= 2857.5
SEATNO <= 1825.5

SEATNO > 491.5
SEATNO <= 1807.5
THEN

MRK4 =
-0.0125 * SEATNO
+ 0.0507 * MRK1
+ 0.0379 * MRK2
+ 29.1792 [68/93.934%]

Rule: 19
IF
SEATNO <= 2857.5
SEATNO <= 1825.5
SEATNO <= 1809.5
SEATNO > 388.5
SEATNO <= 445.5
THEN

MRK4 =
0.0074 * SEATNO
+ 0.2427 * MRK2
- 0.6212 [26/0%]

Rule: 20
IF
SEATNO <= 2857.5
SEATNO <= 1825.5
THEN

MRK4 =

0.0149 * SEATNO

+ 0.4802 * MRK2

- 0.3851 [49/63.616%]

Rule: 21

IF

SEATNO <= 2857.5

SEATNO <= 2291.5

THEN

MRK4 =

0.0023 * SEATNO

+ 0.0684 * MRK1

+ 0.1034 * MRK2

- 2.2362 [26/33.726%]

Rule: 22

IF

SEATNO > 2857.5

MRK2 <= 18

SEATNO <= 2893.5

THEN

MRK4 =

0.1052 * SEATNO

- 0.2571 * MRK2

- 261.6112 [25/28.561%]

Rule: 23
IF
SEATNO <= 3304
SEATNO > 2841
THEN

MRK4 =
-0.0003 * SEATNO
+ 0.1631 * MRK1
+ 3.3644 [32/0%]

Rule: 24
IF
SEATNO > 2832.5
SEATNO <= 4050.5
THEN

MRK4 =
0.011 * SEATNO
- 0.9109 [18/47.061%]

Rule: 25
IF
SEATNO <= 4905
SEATNO > 2299.5
THEN

MRK4 =

0.0035 * SEATNO

+ 0.1255 * MRK1

- 2.6188 [26/59.17%]

Rule: 26

MRK4 =

0.0078 * SEATNO

+ 6.0678 [19/85.58%]

Time taken to build model: 115.25 seconds

=== Evaluation on training set ===

=== Summary ===

Correlation coefficient	0.6549
Mean absolute error	10.1177
Root mean squared error	15.3842
Relative absolute error	75.05 %
Root relative squared error	75.5845 %
Total Number of Instances	11195

4.4 Web Mining and Text Mining Model

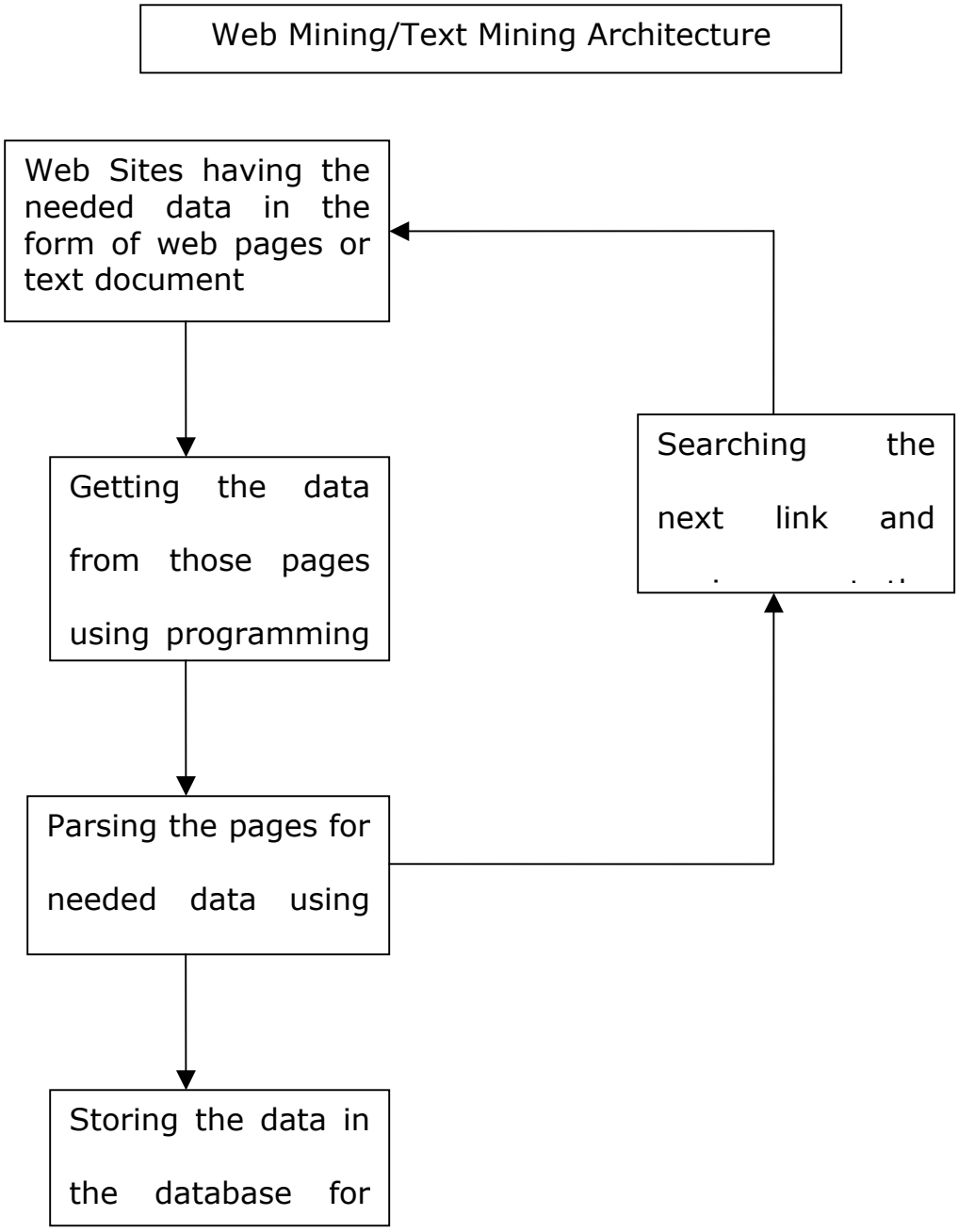


Fig. 4.32

WEB Mining

For Example Consider the following fig. 4.33, HTML page having the information about stock prize of TATA Consultancy Services listed on NSE India

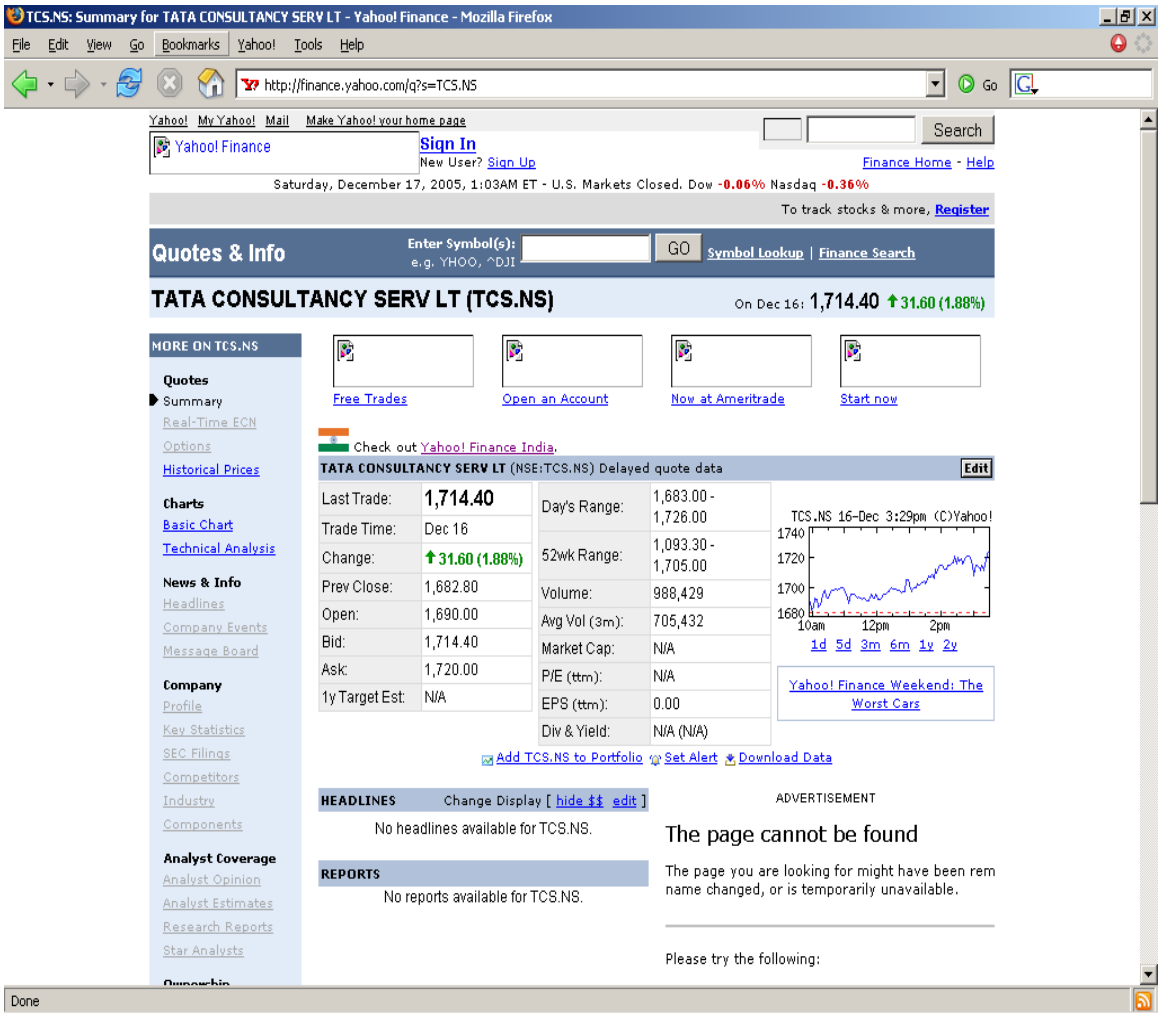


Fig. 4.33

TATA CONSULTANCY SERV LT (NSE:TCS.NS) Delayed quote data

Last Trade:	1,714.40	Day's Range:	1,683.00 - 1,726.00
Trade Time:	Dec 16	52wk Range:	1,093.30 - 1,705.00
Change:	↑ 31.60 (1.88%)	Volume:	988,429
Prev Close:	1,682.80	Avg Vol (3m):	705,432
Open:	1,690.00	Market Cap:	N/A
Bid:	1,714.40	P/E (ttm):	N/A
Ask:	1,720.00	EPS (ttm):	0.00
1y Target Est:	N/A	Div & Yield:	N/A (N/A)

Fig. 4.34

From the Fig. 4.33 we are only interested in the small part and that is shown in Fig. 4.34. So to extract only this part form whole data

Like wise we have more than 750 NSE Stocks and we need this data after every one minutes. Similarly of nearly 6000 BSE Stocks and more than 9000 USA Market Stock.

Below is the code part for achieving the above

```
use HTTP::Headers;
use HTML::TableContentParser;
use WWW::Mechanize;
```

```
my $url = "http://finance.yahoo.com/q?s=TCS.NS";
my $p = HTML::TableContentParser->new();
my $mech = WWW::Mechanize->new( autocheck => 0 );


$mech->timeout(1200);
# Get the URL.
$mech->get ( $url );

if($mech->success())
{
     my $html = $mech->content;
     #print "\n$html\n";
          my $tables = $p->parse($html);
          TABLE : for my $t (@$tables[11,13])
          {
          for my $r (@{$t->{rows}}) {
                    my $col=0;
               for my $c (@{$r->{cells}}) {
                    my $a= $c->{data};
                    if(!$a eq '')
                    {
                         $a =~ s/<[^<>]*//g;
                         $a =~ s/>//g;
                         $a =~ s/\ \;/ /g;
                    }
                     print "$a\t";
               }
               print "\n";
```

```
            }
        }
    }
```

Above code will get the data for only one ticker, you can generalize the things according to the need. You can see the html code and can see the structure of the page and according to your need you can fetch the needed part. Here we are interested in the content of html table number 11 and 13, which has the needed information we are interested in.

After fetching these data we can store it in our own format in database or file. These data can be used directly in the application or we can do data mining on these data.

Text Mining

Fig. 4.35

As shown in the Fig. 4.35, you can see that you are having the chunk of data and you are only interested in small part of it. These data is not in a structured or semi structured form, it is in the un-structured form and you need data from that. Again it is not a single un-structured document, there are 100 thousands of different type of documents and having small common information in all and you need that information.

For text mining research I have taken a data from EDGAR SEC (An U.S. Securities and Exchange Commission), where all public domestic companies were required to make their filings. Out of these filings I have used the 10-K and 10-Q form for my work. And from this form I have tried to extract the balance sheet data. You can see the source code of that at the end of this thesis.

4.5. Analyzing the Developed System

Data Transfer using the application

Researcher has used the Data Transformation Services (DTS) of SQL Server 2000 because it has the application-programming interface (API), which has a set of objects encapsulating services that assist with building a data warehouse. DTS can be used in applications written in languages that support Automation or COM:

- DTS transfers data between heterogeneous OLE DB data sources.
- DTS performs customized transformations that can convert detailed online transaction processing (OLTP) data to a summarized form for easy analysis of trend information.

Researcher observed that the DTS is very fast comparing to other coding style.

Decision Tree using Analysis services

A decision tree is a form of classification shown in a tree structure, in which a node in the tree structure represents each question used to further classify data. The various methods used to create decision trees have been used widely for decades, and there is a large body of work describing these statistical techniques.

The algorithm builds a tree that will predict the value of a column based upon the remaining columns in the training set. Therefore, each node in the tree represents a particular case for a column. The decision on where to place this node is made by the algorithm, and a node at a different depth than its siblings may represent different cases of each column.

You can see the result of decision tree in the Fig. 4.9 to Fig. 4.12

Data Cluster Using the Analysis services

Like decision trees, clustering is a well-documented data mining technique. Clustering is the classification of data into groups based on specific criteria. The topic discussing the Microsoft Clustering algorithm goes into greater detail regarding the details of clustering as a data mining technique. The Clustering algorithm is an expectation method that uses iterative refinement techniques to group records into neighborhoods (clusters) that exhibit similar, predictable characteristics. Often, these characteristics may be hidden or non-intuitive.

You can see the result from Fig. 4.13 to fig.4.27 for Data Cluster

Analyzing the Visual Result

As shown in Fig. 4.28 to Fig.4.31, by seeing you can understand the patterns in the data and also you can compare one subject marks

with other subject marks in a graphical way and find the particular point (instance) information.

Weka Classifier Result

Here researcher has applied the different data mining algorithm for classification using the different test mode.
The test modes are

- **Using training set.** The classifier is evaluated on how well it predicts the class of the instances it was trained on.
- **Supplied test set**. The classifier is evaluated on how well it predicts the class of a set of instances loaded from a file.
- **Cross-validation.** The classifier is evaluated by cross-validation, using the number of folds.
- **Percentage split.** The classifier is evaluated on how well it predicts a certain percentage of the data which is held out for testing.

The result table shows the summary, a list of statistics summarizing how accurately the classifier was able to predict the true class of the instances under the chosen test mode.

Correlation coefficient shows the strength of relationship between variables. So if it is high then it will give a good result.

Absolute error shows difference between a measurement and its true value. So if this value is low, that indicate good algorithm. The low value of root mean square error also indicates the good result.

From the following table you can select the best algorithm based on the given parameter.

Algorithm	**Test mode**	**Correlation coefficient**	**Mean absolute error**	**Root mean squared error**	**Relative absolute error**	**Root relative squared error**
ZeroR	10-fold cross-validation	-0.0282	12.8975	18.0772	100 %	100 %
ZeroR	split 66% train, remainder test	0	12.9393	18.1137	100 %	100 %
DecisionTable -X 1 -S 5	split 50% train, remainder test	0.7616	7.4246	11.9364	58.3284 %	65.681 %
Vote -B	5-fold cross-validation	-0.0123	12.8967	18.0763	100 %	100 %
DecisionTable -X 1 -S 5	split 66% train, remainder test	0.8021	8.0574	10.8186	62.2703 %	59.7259 %
DecisionStump	10-fold cross-validation	0.7449	8.8853	12.0586	68.8913 %	66.7062 %
DecisionStump	split 50% train, remainder	0.7464	8.8881	12.0702	68.7037 %	66.5509 %

Algorithm	Test mode	Correlation coefficient	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error
	test					
DecisionStump	evaluate on training data	0.7462	8.8714	12.0325	68.7992 %	66.5679 %
PaceRegression -E eb	evaluate on training data	0.7867	8.0399	11.1588	62.351 %	61.7344 %
PaceRegression -E eb	10-fold cross-validation	0.781	8.06	11.2901	62.4928 %	62.4551 %
PaceRegression -E eb	split 50% train, remainder test	0.7931	8.2308	11.0622	63.6231 %	60.9929 %
LWL -U 0 -K -1 –W	split 50% train, remainder test	0.751	8.794	11.9777	67.9761 %	66.0405 %
M5Rules -M 4.0	split 50% train, remainder test	0.8325	6.5833	10.0794	51.7191 %	55.4631 %
DecisionTable -X 1 -S 5	split 50% train, remainder test	0.7953	8.0904	11.0131	62.5379 %	60.7224 %
M5Rules -M 4.0	evaluate on training data	0.7771	8.2597	11.0031	64.6061 %	62.9606 %
M5Rules -M 4.0	evaluate on training data	0.7427	8.8643	11.7022	69.3346 %	66.9605 %

Algorithm	Test mode	Correlation coefficient	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error
M5Rules -M 4.0	evaluate on training data	0.7707	7.0526	12.4607	53.8001 %	63.7171 %
M5Rules -M 4.0	evaluate on training data	0.6549	10.1177	15.3842	75.05 %	75.5845 %

Web Mining

As shown in Fig. 4.33 if you open the url and see the source code of that page and if you copy that source code to a word file then it needs nearly 35 pages. And if you see the source code of the Fig. 4.34, it just needs nearly 3 pages and also if you extract the data only and remove the html tags then it is needs only few lines. So if you have large number of html pages and you want the data after some duration then you can not store the whole pages into the disk space, and again if you do that then also the data is not that much useful. So by web mining you are making the data useful.

Text Mining

In web mining the data is in the semi-structured form, but in text mining the data is not in structured or semi-structured form, and to find the needed data from this is a very difficult task. Here I try to mine the text data, but I have not got the 100% success for the problem I mention in text mining part, I just got the success of nearly 81%.

5.4. Summary of work

The research has a focused attention on the study of Data Mining Techniques supported by available tools. The enhance need of data mining particularly web mining and text mining was encountered which needed support by way of research findings. The outcome of the research is the development of algorithm not from scratch but as an extension to available algorithms to meet with enhanced requirements in web mining. The algorithms are coded in language of PERL to implement the prototype.

In this research we have transfer the data from legacy system to the data warehouse database. For this we have used the DTS and Visual Basic programming. Then these data is used for the pure data mining using the open source tool as well as by a professional vendor. For the open source system we have used the WEKA (Data Mining Software in Java). We have used many algorithms for classification and visualization of data. For the professional tool, we have used the SQL Server 2000 Analysis services. In this we have used the both available data mining algorithm i.e. Decision Tree and Data Cluster.

We have also done some work in the area of web mining and text mining. For web mining, we have used the web data and extracted the needed information from that. For text mining, we have used the EDGAR SEC data, in these data we have used the Form 10-K and 10-Q for our research and we have written the algorithm and Perl programs for separating the balance sheet data to the separate file.

5.5. Conclusion

The research work has taken a shape of data warehousing and data mining linked system that can effectively and an efficiently be implemented in diversify segment of data handling through data bases and file systems. The research work also provides a way to automate the process of data conversation from distributed database mode to data warehouse mode. The research work has also put a focus on the knowledge of the characteristics of data to be and stored in the data warehouse for the purpose of data mining leading to unseen the hidden patterns in the data. The research has contributed to ascertain situation and context specific need of technologies and tools, which makes suitability of their uses. The comparison of the tools and technology are not limited to technical specific based but it extends to analyzed the outcomes of various stages of data warehousing and data mining. The ultimate aim of research was to add value to the data using the integrated approach of data warehousing and data mining.

The case study included in the research can lead other users to take required depth of the strategies, which are easy effective and an efficient to be implemented for them making data a valuable assets.

5.6. Future work

Knowledge discovery can be broadly defined as the automated discovery of novel and useful information from commercial databases. Data mining is one step at the core of the knowledge discovery process, dealing with the extraction of patterns and relationships from large amounts of data.

Today, most enterprises are actively collecting and storing large databases. Many of them have recognized the potential value of these data as an information source for making business decisions.

The dramatically increasing demand for better decision support is answered by an extending availability of knowledge discovery and data mining products, in the form of research prototypes developed at various universities as well as software products from commercial vendors.

However, despite its rapid growth, KDD is still an emerging field. The development of successful data mining applications still remains a tedious process. The following is a (naturally incomplete) list of issues that are unexplored or at least not satisfactorily solved yet:

Integration of different techniques:

Currently available tools deploy either a single technique or a limited set of techniques to carry out data analysis. Our research shows that there is no best technique for data mining. The issue is therefore not which technique is better than another, but rather which technique is suitable for the problem at hand. A truly useful tool has to provide a wide range of different techniques for the solution of different problems.

Extensibility:

This is another consequence from the fact that different techniques outperform each other for different problems. With the increasing number of proposed techniques as well as reported applications, it becomes clearer and clearer that any fixed arsenal of algorithms will never be able to cover all arising problems and tasks. It is therefore important to provide an architecture that allows for easy synthesis of new methods, and adaptation of existing methods with as little effort as possible.

Seamless integration with databases:

Currently available data analysis products generally fall into two categories. The first is drill-down analysis and reporting, provided by vendors of RDBMS's, sometimes in association with on-line analytical processing (OLAP) vendors. These systems provide a tight connection with the underlying database and usually deploy the processing power and scalability of the DBMS. They are also restricted to testing user provided hypotheses, rather than automatically extracting patterns and models. The second category consists of (usually stand-alone) pattern discovery tools, which are able to autonomously detect patterns in the data. These tools tend to access the database offline; that is, data is extracted from the database and fed into the discovery engine. Many tools even rely on keeping all their data in main memory, thus lacking scalability and, therefore, the ability to handle real world problems. Additionally, these tools are often insufficiently equipped with data processing capabilities, leaving the data preprocessing solely to the user. This can result in time consuming repeated export-import processes.

Managing changing data:

In many applications, including the vast variety of nearly all business problems, the data is not stationary, but rather changing and evolving. This changing data may make previously discovered patterns invalid and hold new ones instead. Currently, the only solution to this problem is to repeat the same analysis process (which is also work-intensive) in periodic time intervals. There is clearly a need for incremental methods that are able to update changing models, and for strategies to identify and manage patterns of temporal change in knowledge bases.

Non-standard data types:

Today's databases do not contain only standard data such as numbers and strings but also large amounts of nonstandard and multimedia data, such as free-form text, audio, image and video data, temporal, spatial and other data types. Those data types contain special patterns, which cannot be handled well by the standard analysis methods. Therefore, these applications require special, often domain-specific, methods and algorithms.

While there are fundamental problems that remain to be solved, there have also been numerous significant success stories reported, and the results and benefits are impressive. Although the current methods still rely on fairly simple approaches with limited capabilities, reassuring results have been achieved, and the benefits of KDD technology have been convincingly demonstrated in the broad range of application domains. The combination of urgent practical needs

and the strong research interests lets us also expect a future healthy grow of the field, drawing KDD tools into the mainstream of business applications.

The current research has a possibility of extension in the areas specified here under

Future possibility of work in

Data Transfer:

One can develop the application, which can transfer the data from any type of database to any one database using a single system, and the system will be intelligent too.

Analysis services of SQL Server Environment:

I have studied the Analysis services of SQL Server 2000, and at this time just Microsoft has release the new version of SQL Server having the good Business Intelligent support, so one can study those algorithm and can compare it with other algorithms provided by other vendors.

WEKA Environment:

One can enhance the result of WEKA using the following in his/her research.

- Use more data sets:

  The use of a large number of data sets would allow an increase in size of the data sets generated for classifying analysis.

- Use more data set sources:

  The data sets used in this thesis came mainly from the one course data collection. The use of a larger variety of real data sets from different courses may allow the formation of decision tree, which reveal patterns between different courses and data mining algorithms.

- Use small to very big data sets:
  In the data mining industry the size of the data to be analyzed can be very large. The maximum size of the data sets used in this thesis was 11195. It would be useful to see what kind of performance is obtained and what types of decision tree are formed when large data sets are used.
- Use optimal parameter values by fine-tuning the settings of each algorithm
- Use of more characteristics of data sets:
  Only 'number of instances' and 'number of attributes' were used in this thesis. Characteristics of the data sets such as whether or not the data sets contain numeric, symbolic or mixed values and missing values could be useful.
- Use visualization tools to analyze the generated data set:
  Visualization of the generated data set may provide important information and may allow better analysis of the decision tree formed.

Web Mining:

We anticipate the seamless adoption of XML in the future, so one can work on support for XML-style well-formed structures and links. In the future, as the web evolves from a structural web to a semantic web, we envision such duality mining over data and metadata to identify higher level metadata as a key area of research.

Future Work in Text Mining:

Text mining concerns looking for patterns in unstructured text. The related task of Information Extraction (IE) is about locating specific items in natural-language documents. One can design a framework for text mining, using a learned information extraction system to transform text into more structured data which is then mined for interesting relationships. In addition, rules mined from a database extracted from a corpus of texts are used to predict additional information to extract from future documents, thereby improving the current developed extraction system.

1. Data Mining and Data Warehouse Tools

A

SQL Server 2005 Data Mining Features

Business IntelligenceTechnologyCapabilities

Analysis Services

Analysis Services delivers extensions to the scalability, manageability, reliability, availability, and programmability of data warehousing, business intelligence, and line-of-business solutions.

Data Transformation Services (DTS)

A complete redesign of the DTS architecture and tools provides developers and database administrators with increased flexibility and manageability.

Reporting Services

Reporting Services is a new report server and tool set for building, managing, and deploying enterprise reports.

Data Mining

Data mining is enhanced with four new algorithms as well as improved data modeling and manipulation tools.

Create an easy–to–use, extensible, accessible, and flexible business intelligence platform and take the next step in business intelligence with SQL Server data-mining capabilities. Explore your data, discover patterns, and uncover business data to reveal the hidden trends about your products, customer, market, and employees, and better

analyze those components that are critical to your organization's success.

Integration

SQL Server Data Mining is part of a family of business intelligence technologies that can be used together to enhance and develop a new breed of intelligent applications. These technologies include the following:

- **SQL Server 2005 Integration Services.** Create a more powerful data pipeline by working with SQL Server 2005 Integration Services, allowing your organization to flag outliers, separate data, and fill in missing values based on the predictive analytics of the data-mining algorithms.

- **SQL Server 2005 Analysis Services.** Create a richer Unified Dimensional Model by adding data-mining dimensions that slice your data by the hidden patterns within.

- **SQL Server Reporting Services.** Create smarter, insightful reports based on data-mining queries that present the right information to the right audiences.

Architecture

Providing data mining to organizations of any size introduces new challenges. Deployment, scalability, manageability, and security all become important factors. SQL Server Data Mining is part of the SQL Server Analysis Services server that provides all the enterprise-class server features you would expect:

- **Deployment.** SQL Server Data Mining is based on a client-server architecture, allowing you to access models from your local area network (LAN), wide area network (WAN), or the Internet. Standard application programming interfaces (APIs) provide access to your models regardless of location or client platform.

- **Scalability.** SQL Server Data Mining is designed from the ground up with a parallel architecture to scale to enterprise-class data sets and thousands of concurrent users, and can respond to millions of queries per day.

- **Manageability.** SQL Server Data Mining is integrated into the new SQL Server Management Studio, providing a one-stop tool for managing all your SQL Server family properties.

- **Security.** SQL Server Data Mining provides fine-grained, role-based security to ensure that your intellectual property will be further protected.

Data Mining Wizard

Using the built-in Data Mining Wizard and Designer, you can build sophisticated models with only a few mouse clicks.

Integrated directly into Microsoft Visual Studio, the SQL Server Data Mining toolset lets you explore and manipulate data, as well as design and edit your models. SQL Server Data Mining provides more than a dozen interactive visualizations to help you understand the patterns that data mining can discover. Additionally, lift and profit charts are provided so you can compare and contrast the quality of your models before you commit to deployment.

Simple, Rich API

When it comes to applying models, SQL Server opens a new chapter in data mining. Data Mining eXtensions (DMX) for SQL make it easy for developers and database administrators to create data mining–aware applications. For the first time, those responsible for creating applications and handling data are empowered to use data-mining technology using tools they already understand.

Extensibility

SQL Server Data Mining is fully extensible through Microsoft .NET–stored procedures and plug-in algorithms and viewers that embed seamlessly to take advantage of all the platform abilities and integration. Adopting SQL Server Data Mining as your platform means that you will never be limited by the inherent functionality of your data-mining system because it can always be extended to meet your needs.

B

DB2 Intelligent Miner

IBM's data mining capabilities help you detect fraud, segment your customers, and simplify market basket analysis. IBM's in-database mining capabilities integrate with your existing systems to provide scalable, high performing predictive analysis without moving your data into proprietary data mining platforms. Use SQL, Web Services, or Java to access DB2's data mining capabilities from your own applications or business intelligence tools from IBM's business partners.

Tools

DB2 Intelligent Miner Modeling

Delivers DB2 Extenders for modeling operations

DB2 Intelligent Miner for Scoring

Provides scoring technology as database extensions: DB2 Extenders and Oracle cartridges

DB2 Intelligent Miner Visualization

Provides Java visualizers to interact and graphically present the results of associations

DB2 Intelligent Miner for Data

Provides new business insights and harvests valuable business intelligence from your enterprise data

Tool Benefits

Using PMML

Share data mining data using the industry standard PMML

Scalable Mining

Intelligent Miner tools integrate seamlessly with DB2 UDB

C

Informatica PowerCenter Advanced Edition

One of the biggest challenges facing organizations today is the fragmentation of data across disparate IT systems. Unlocking and

deriving business value from these strategic data assets — no matter where they reside — has become a top priority.

Companies are realizing that in order to support their business objectives--such as providing a single view of the customer, migrating away from legacy systems to new technology, or consolidating multiple instances of an ERP system—they must be able to effectively integrate, move and access their data, or its business value will be lost.

Informatica PowerCenter Advanced Edition, the leading independent data integration platform, addresses this need--delivering the industry's most comprehensive set of capabilities for enterprise-wide data integration. PowerCenter Advanced Edition supports the entire data integration lifecycle—allowing companies to access, discover, and integrate data from the widest variety of enterprise systems and deliver that data to other operational systems, applications, databases and to the business user for decision making.

With Informatica PowerCenter Advanced Edition you can:

- **Deliver universal access to enterprise data** - broadest access to data provides your organization with a holistic view of enterprise data
- **Cost-effectively scale to respond to business needs for information** - deliver data in a secure, scalable environment that provides immediate data access to all disparate sources
- **Increase productivity** - expedite time to results with enhanced design and collaboration features, and reuse of existing development assets across projects

D

BusinessObject

BusinessObjects XI Release 2 provides performance management, reporting, query and analysis, and data integration in one solution.

- Performance management - Match actions with strategy.
- Reporting - Access, format, and deliver data.
- Query and analysis - Self-serve analysis for users.
- BI platform - Manage BI tools, reports, and applications.
- Data integration - Access, transform, and integrate data.

E

PeopleSoft

Oracle's PeopleSoft Enterprise applications are designed for the most complex business requirements. They provide web services integration with multivendor and homegrown applications and can be easily configured and adapted to meet the most unique customer requirements. In addition, PeopleSoft Enteprise supports a very broad choice of technology infrastructure.

PeopleSoft Enterprise belongs to the Oracle Applications product line, which also includes JD Edwards EnterpriseOne, JD Edwards World, and the Oracle E-Business Suite.

Campus Solutions

Manage the student lifecycle end to end-driving operational efficiency, reducing costs, and freeing resources to support the goals of higher education institutions.

Customer Relationship Management

Increase revenues and drive customer satisfaction and loyalty through Sales, Marketing, and Service effectiveness.

Enterprise Performance Management

Achieve world-class performance by aligning the right information and resources to strategic objectives.

Financial Management

Build a foundation for sustainable compliance and growth using our comprehensive suite of financial applications.

Human Capital Management

Manage and mobilize a unified, global workforce, and align workforce contribution with business objectives.

Service Automation (Project Management)

Optimize your project investments, reduce project delivery costs and maximize resources to increase utilization and value to your organization.

Supplier Relationship Management (Procurement)

Manage all aspects of your supplier relationships including indirect and direct goods, as well as services procurement.

Supply Chain Management

Take advantage of solutions that promote business-to-business interaction throughout the supply chain, from customer to supplier.

F

Cognos 8 Business Intelligence

Cognos 8 Business Intelligence is the only BI product to deliver the complete range of BI capabilities: reporting, analysis, scorecarding, dashboards, business event management as well as data integration, on a single, proven architecture.

Easy to integrate, deploy and use, Cognos 8 BI delivers a simplified BI environment that improves user adoption, enables better decision-making, and serves as an enterprise-scale foundation for performance management.

Reporting

Reporting is a key capability within Cognos 8 Business Intelligence, a single product that provides complete BI capabilities on a proven architecture.

Reporting gives you access to a complete list of self-serve report types, is adaptable to any data source, and operates from a single metadata layer for a variety of benefits such as multilingual reporting.

Analysis

Analysis is a key capability within Cognos 8 Business Intelligence, a single product that provides complete BI capabilities on a proven architecture.

Analysis enables the guided exploration and analysis of information that pertains to all dimensions of your business-regardless of where the data is stored. Analyze and report against online analytical processing (OLAP) and dimensionally aware relational sources.

Scorecarding

Scorecarding is a key capability within Cognos 8 Business Intelligence, a single product that provides complete BI capabilities on a proven architecture. Scorecarding helps you align your teams and tactics with strategy; communicate goals consistently, and monitor performance against targets.

Dashboards

Business dashboards communicate complex information quickly. They translate information from your various corporate systems and data into visually rich presentations using gauges, maps, charts, and other graphical elements to show multiple results together.

Business Event Management

Cognos 8 BI business event management tracks significant events that need attention. It monitors these events and uses decision-process and business-process automation to compress the time to action and resolution.

Data Integration

Data integration is a key component within Cognos 8 Business Intelligence, a single product that provides complete BI capabilities on a proven architecture.

Cognos data integration is an enterprise-wide ETL solution designed for high performance business intelligence. It optimizes data merging, extraction, transformation, and dimensional management to deliver data warehouses ready for business reporting and analysis.

G

MicroStrategy

MicroStrategy Data Mining Services is a fully integrated component of the MicroStrategy BI platform that delivers the results of predictive models to all users in familiar, highly formatted and interactive reports and documents. It empowers business users to leverage sophisticated predictive models that describe the likelihood of a future event. Data Mining Services allows organizations to use predictive functions that are natively available with MicroStrategy, or import models from third-party data mining vendors. With Data Mining Services, users can access highly formatted, data-rich, predictive reports through all the user interfaces supported by the MicroStrategy BI platform, as well as perform standard report manipulations and create new reports on the fly. Data Mining Services extends the reach of the traditional data mining insight by incorporating the scoring process on-demand in a single BI platform.

MicroStrategy is the first to deliver data mining and predictive analysis to all users through a fully integrated, enterprise-caliber BI platform. Using Data Mining Services, business users, report designers and analysts alike can view and build predictive reports using MicroStrategy and distribute these reports to all relevant decision makers and stakeholders.

Features and Benefits

- Apply data mining models on demand against terabytes of data, rather than waiting for the data miner or DBA to score the database tables and columns
- Slice and dice predictive information within a limited analytical domain making it simple and safe for casual users
- Create flexibly organized and highly formatted predictive reports for easiest possible user consumption and professional presentation
- Import sophisticated predictive algorithms in PMML format using a few simple clicks of the mouse
- Leverage the full analytical power of the MicroStrategy BI platform through a fully integrated solution
- Out-of-the-box predictive calculation using over 250 analytical functions including multi-variable linear regression
- Deliver individualized messages and predictive reports to very large populations through a single "service definition" based on event triggers or schedules
- Provide large user populations with ad-hoc query and ad-hoc analysis on predictive data of an entire database without

requiring knowledge of SQL, table structures, or predictive models

- Implement even the strictest security scheme to users within and outside the organization
- Apply data mining models against terabytes of data

H

Hyperion Intelligence

Hyperion Intelligence is the most advanced and easy-to-use set of tools available for sophisticated query and analysis. Leveraging data from existing enterprise transactional systems, Hyperion Intelligence provides information developers, analysts and consumers with the ability to make raw data useful for decision making. An intuitive, Web-enabled dashboard interface enables every employee across the enterprise to benefit from the insightful analyses and reports. By delivering business-critical information - not just data - Hyperion Intelligence drives an interactive understanding of business opportunities and trends, empowering employees to make optimal decisions.

Hyperion Intelligence Release 8.3 Highlights:

Integration

- Single Sign-On
- Hyperion Hub
- RelatedContent.properties File
- Related_Content_Viewer Role

- List and Launch Hyperion Analyzer and Hyperion Reports from the Foundation
- Browse Module
- Intelligence Dashboard Access of Hyperion Analyzer and Hyperion Reports
- Passing Point of View (POV) Information to Hyperion Performance Suite

Installer

- CSS
- Hyperion Hub
- Swing conversion
- ISMP 5.0.3
- JRE 1.4.2
- IONA 6.1
- Apache Tomcat 4.1.30 standalone Web server
- Apache Tomcat with Microsoft IIS 6
- IBM WebSphere 5.0.1

Job Manager

A new Consolidated Job Status list in the Job Manager module shows the status of jobs in the system in one place, including the names of the job's owners, schedules, and events, and the job's last run status and next/last run times, with links to modify these items and add new schedules.

Security Enhancements

- SQL statements are encrypted between the client and the Data Access Service (for the plug-in or Web clients).
- Stronger encryption of the database password.
- Passwords are only sent when required (and not in cases where pass-through is enabled).
- Many parameters, notably database username and database password, are sent as POST parameters instead of as options in the URL.

Supported Platforms

Introduces support for Oracle 10g both as a repository and as a data source.

I

Weka : Data Mining Software in Java

An exciting and potentially far-reaching development in computer science is the invention and application of methods of machine learning. These enable a computer program to automatically analyse a large body of data and decide what information is most relevant. This crystallised information can then be used to automatically make predictions or to help people make decisions faster and more accurately.

The overall goal of our project is to build a state-of-the-art facility for developing machine learning (ML) techniques and to apply them to real-world data mining problems. Our team has incorporated several standard ML techniques into a software "workbench" called WEKA, for Waikato Environment for Knowledge Analysis. With it, a specialist in a

particular field is able to use ML to derive useful knowledge from databases that are far too large to be analysed by hand. WEKA's users are ML researchers and industrial scientists, but it is also widely used for teaching.

Our objectives are to

- make ML techniques generally available;
- apply them to practical problems that matter to New Zealand industry;
- develop new machine learning algorithms and give them to the world;
- contribute to a theoretical framework for the field.

Our machine learning package is publically available and presents a collection of algorithms for solving real-world data mining problems. The software is written entirely in Java and includes a uniform interface to a number of standard ML techniques.

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes. Weka is open source software issued under the GNU General Public License.

J

Java Data Mining API

The Java Data Mining API (JDM) is the first attempt to create a standard Java API to access data-mining tools from Java applications. JDM promises to bring to data mining what JDBC brought to databases, and to make data mining a new and useful part of an enterprise Java developer's tool chest. This article introduces basic data-mining concepts, and illustrates sample JDM code to model customer behavior.

JDM addresses the need for a pure JavaTM API that supports data mining operations and activities. JDM 2.0 extends JDM with requested functionality for new mining functions, mining algorithms, and corresponding web services specification. Features that should be considered in JDM 2.0 include, but are not limited to, the following:

- Sequential Patterns / Time Series - mining functions to address forecasting and modeling seasonal or periodic fluctuations in data.
- Transformations interface - data preparation is a key aspect of any data mining solution. A separate JSR for transformations is likely warranted. Having a close integration with such a JSR and addressing transformations in the next version has high priority.
- Ensemble models - define composite models structured with logic, e.g., boosting and bagging approaches.
- Apply for Association - augment specification to enable prediction based on association rules.
- Text Mining - enable mining of unstructured text data both by explicit feature extraction and the accepting of text attributes as model predictors

- Model Comparison - introduce ability to compare multiple models according to various quality metrics, e.g., accuracy and lift for classification.
- Multi-record real-time scoring - enable scoring of multiple records in the record apply task as a performance optimization for applications.
- Multi-target models - enable the specification of multiple targets for supervised models as a model performance and representation optimization.

The goal of the JDM 2.0 Expert Group will be to investigate these features and to identify and pursue others necessary for the data mining and Java community.

K

Oracle

Oracle Data Mining (ODM), an option to Oracle Database 10g Enterprise Edition, enables companies to extract information efficiently from the very largest databases and build integrated business intelligence applications. Data analysts can find patterns and insights hidden in their data. Application developers can quickly automate the extraction and distribution of new business intelligence—predictions, patterns and discoveries—throughout the organization.

ODM supports functionality in Oracle Database 10g for the following data mining problems: classification, prediction, regression, clustering, associations, attribute importance, feature extraction and

sequence similarity searches and analysis (BLAST). All model-building, scoring, and metadata management operations are accessed via the Oracle Data Mining Client and either a PL/SQL or Java-based API and occur entirely within the relational database.

Oracle Data Miner is a graphical user interface for Oracle Data Mining that helps data analysts mine their Oracle data to find valuable hidden information, patterns, and new insights. Oracle Data Miner Release 10gR2 adds many new features that makes data mining easier and produces even more actionable information. New Mining Activity Guides streamline the data mining process by providing step-by-step guidance and automate data mining and model scoring. In just a few clicks, you are mining your data and discovering new insights.

Oracle Data Miner Release 10gR2 introduces the popular Decision Tree algorithm for classification problems that can provide human readable "IF...THEN..." rules that communicate the patterns discovered by ODM. The new Anomaly Detection algorithm flags rare events and supports fraud and compliance monitoring. Oracle Data Miner now supports mining multiple tables at once (e.g., star schema) and supports mining unstructured "text" data. Oracle Data Miner also supports PREDICT and EXPLAIN "one-click data mining" predictive analytics. Oracle Data Miner Release 2 adds Receiver Operating Characteristics support for model evaluation and tuning. Oracle Data Miner can automatically generate the Java and SQL components needed to transform the data mining steps into an integrated data mining/BI enterprise application. Lastly, a new

Gateway to Oracle Discoverer enables data analysts to publish their results for viewing through Oracle Discoverer.

With Oracle Data Miner and Oracle Data Mining, the data never leaves the database: all data movement is eliminated. In addition, Oracle Data Miner and Oracle Data Mining provide the security of the Oracle database.

L

SAS

Call data, mail-order addresses, sales histories, POS data, Web transactions, even free-form text notes ... if your organization could fully harness and exploit this wealth of information, the potential would be enormous. With data mining, the possibilities are endless.

Achieving industry-leading status almost upon its introduction, our data mining technology continues to receive rave reviews from industry experts and users alike. Our latest enhancements include the addition of text mining capabilities that enable you to quickly determine key information contained in large document collections, as well as integrate the text-based information with structured data for an enriched data mining process.

Forward-thinking companies today are using data mining to reduce fraud, anticipate resource demand, increase acquisition and curb customer attrition. SAS award-winning data mining solutions enable you to:

- **Seek and retain your most profitable customers**. Use demographic data and customer buying patterns to develop lifelong relationships with your customers, anticipating and fulfilling their needs.
- **Segment markets for a targeted approach**. Target marketing campaigns to dramatically increase response rates, analyze clickstream data and sharpen your e-commerce strategy.
- **Predict the future and identify factors to secure a desired effect**. Improve production process quality by anticipating problems before they occur, forecast resource demands, increase acquisitions and assess the risk of customer credit applications.

Now you can unleash the hidden potential in your data with the SAS Data Mining Solution, tackle business challenges with new savvy and have greater confidence in the expected outcomes. By applying data mining techniques, you can fully exploit data about customers' buying patterns and behavior to gain a greater understanding of consumer motivations.

Data mining defined

Data mining is the process of data selection, exploration and building models using vast data stores to uncover previously unknown patterns. What does this mean to you?

You can produce new knowledge to better inform decision makers before they act. Build a model of the real world based on data collected from a variety of sources, including corporate transactions, customer histories and demographics, even external sources such as credit bureaus. Then use this model to produce patterns in the information that can support decision making and predict new business opportunities. Text mining capabilities enable you to apply such analyses to text-based documents. With SAS's rich suite of text processing and analysis tools, you can uncover underlying themes or concepts contained in large document collections, group documents into topical clusters, classify documents into predefined categories and integrate text data with structured data for enriched predictive modeling endeavors.

Data mining reaches across industries and business functions

- Telecommunications, stock exchanges and credit card and insurance companies use data mining to detect fraud, optimize marketing campaigns and identify the most profitable strategies.
- The medical industry uses data mining to predict the effectiveness of surgical procedures, medical tests and medications.
- Retailers use data mining to assess the effectiveness of coupons and special events, and predict which offers are most appropriate for different consumers.

The complete data mining solution

Our comprehensive data mining solution includes services and training that allow you to explore large quantities of data and discover relationships and patterns that lead to proactive decision making. Along with SAS consultants, our Quality Partner Program and strategic partnerships with leading consulting organizations are available as resources to meet your organization's specific needs.

Enterprise Miner, our enhanced data mining software, combines a rich suite of integrated data mining tools with unprecedented ease of use, empowering users to explore and exploit corporate data for strategic business advantage all in a single environment.

SAS Text Miner expands our data mining capabilities to include vast deposits of textual data. Integrating text-based information with structured data enriches your predictive modeling capabilities and provides new stores of insightful and valuable information for driving your business and research initiatives forward.

M

SPSS

SPSS data mining solutions and services have enabled hundreds of organizations to achieve remarkable results in many areas. For example, organizations have used data mining to:

- Boost sales by 50 percent and reduce marketing costs by 30 percent by uncovering cross-selling and "rollover" sales opportunities. Read the complete data mining customer story.
- Triple online profits by improving personalization features. Read the complete data mining customer story.

- Secure an additional $50 million in revenue by using an accurate propensity model to target offers. Read the complete data mining customer story.
- Improve the response rate of direct mail campaigns by 100 percent. Read the complete data mining customer story.

Using data mining tools

Most analysts separate data mining software into two groups: data mining tools and data mining applications. Data mining tools provide a number of techniques that can be applied to any business problem. Data mining applications, on the other hand, embed techniques inside an application customized to address a specific business problem. Regardless of whether we are aware of it, our daily lives are influenced by data mining applications. For example, almost every financial transaction is processed by a data mining application to detect fraud. Both data mining tools and data mining applications are valuable, however. Increasingly, organizations are using data mining tools and data mining applications together in an integrated environment for predictive analytics.

So what do data mining tools add? Data mining tools are used to ensure flexibility and the greatest accuracy possible. Essentially, data mining tools increase the effectiveness of data mining applications. Since no two organizations or data sets are alike, no single technique delivers the best results for everyone. Not only do data mining tools deliver in-depth techniques, but data mining tools also deliver flexibility to use combinations of techniques to improve predictive accuracy.

Because data mining tools are so flexible, a set of data mining guidelines and a data mining methodology have been developed to help guide the process. The Cross-Industry Standard Process for Data Mining (CRISP-DM) ensures your organization's results with data mining tools are timely and reliable. This methodology was created in conjunction with practitioners and vendors to supply data mining practitioners with checklists, guidelines, tasks, and objectives for every stage of the data mining process.

Clementine data mining transforms data into actionable results

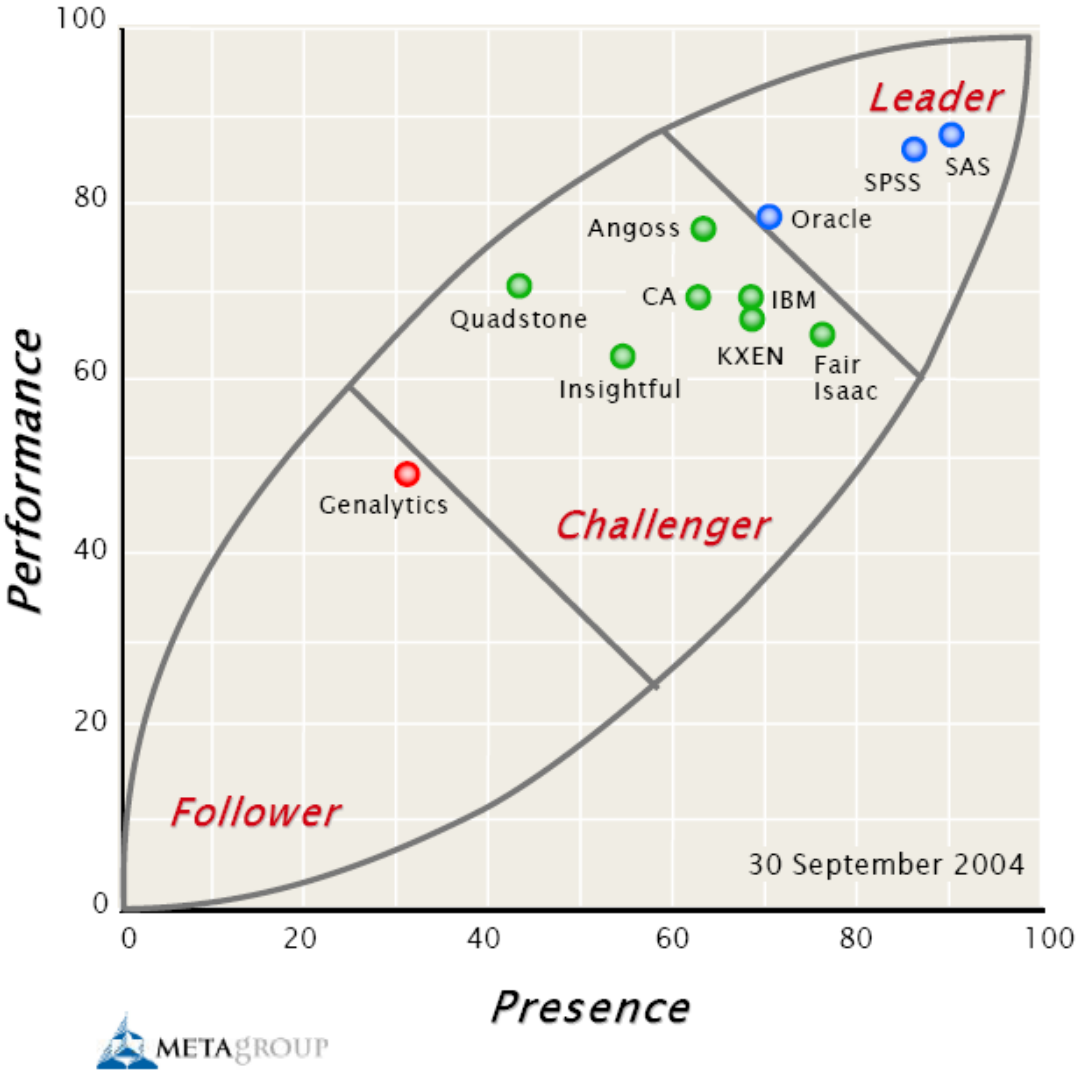
Clementine, the SPSS data mining workbench, enables your organization to quickly develop predictive data mining models and deploy those data mining models into your organization's operations - improving decision making. Using Clementine's powerful, visual data mining interface and your business expertise, you can quickly interact with your data and begin discovering patterns you can use to change your organization for the better. To learn more about the Clementine data mining workbench, visit the comprehensive Clementine page.

The latest data mining advances—text mining and Web mining

Recent advances have led to the newest and hottest trends in data mining—text mining and Web mining. These two data mining technologies open a rich vein of customer data in the form of textual comments from survey research and log files from Web servers, which were previously unusable. Applying data mining to these data adds a richness and depth to the patterns already uncovered through your data mining efforts.

2. Comparison on different Data Mining Enabled Tools

Comparison by META Group



Comparison Elder Research

Tools Evaluated

Product	Company	URL	Version Tested	Our Experience
Clementine	Integral Solutions, Ltd.	http://www.isl.co.uk/clem.html	4	Moderate
Darwin	Thinking Machines, Corp.	http://www.think.com/html/products/products.htm	3.0.1	Moderate
DataCruncher	DataMind	http://www.datamindcorp.com	2.1.1	High
Enterprise Miner	SAS Institute	http://www.sas.com/software/components/miner.html	Beta	Moderate
GainSmarts	Urban Science	http://www.urbanscience.com/main/gainpage.htm	4.0.3	Low
Intelligent Miner	IBM	http://www.software.ibm.com/data/iminer/	2	Low
MineSet	Silicon Graphics, Inc.	http://www.sgi.com/Products/software/MineSet/	2.5	Low
Model 1	Group 1/Unica Technologies	http://www.unica-usa.com/model1.htm	3.1	Moderate
ModelQuest	AbTech Corp.	http://www.abtech.com	1	Moderate
PRW	Unica Technologies, Inc.	http://www.unica-usa.com/prodinfo.htm	2.1	High
CART	Salford Systems	http://www.salford-systems.com	3.5	Moderate
NeuroShell	Ward Systems Group, Inc.	http://www.wardsystems.com/neuroshe.htm	3	Moderate
OLPARS	PAR Government Systems	mailto://olpars@partech.com	8.1	High
Scenario	Cognos	http://www.cognos.com/busintell/products/index.html	2	Moderate
See5	RuleQuest Research	http://www.rulequest.com/see5-info.html	1.07	Moderate
S-Plus	MathSoft	http://www.mathsoft.com/splus/	4	High
WizWhy	WizSoft	http://www.wizsoft.com/why.html	1.1	Moderate

3. Data Transformation Code

```
Public pathtocopy As String
Public yinfo As String
Public cinfo As String
Private Sub Command1_Click()
    pathtocopy = Dir2.Path

    MsgBox Mid(Dir1.Path, InStrRev(Dir1.Path, "\") + 1)
    yinfo = Mid(Dir1.Path, InStrRev(Dir1.Path, "\") + 1)
    MsgBox Mid(Dir2.Path, InStrRev(Dir2.Path, "\") + 1)
    cinfo = Mid(Dir2.Path, InStrRev(Dir2.Path, "\") + 1)
    Main cinfo, yinfo, pathtocopy
    MsgBox "Process Completed"

End Sub

Private Sub Dir1_Change()
    Dir2.Path = Dir1.Path
End Sub
Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

Private Sub Form_Load()
    Drive1.Drive = "e:\"
    Dir1.Path = ""
    Dir2.Path = ""
```

```
End Sub
Option Explicit
Public goPackageOld As New DTS.Package
Public goPackage As DTS.Package2
Private Sub Main()
        Set goPackage = goPackageOld
        goPackage.Name = "Final Package"
        goPackage.WriteCompletionStatusToNTEventLog = False
        goPackage.FailOnError = False
        goPackage.PackagePriorityClass = 2
        goPackage.MaxConcurrentSteps = 4
        goPackage.LineageOptions = 0
        goPackage.UseTransaction = True
        goPackage.TransactionIsolationLevel = 4096
        goPackage.AutoCommitTransaction = True
        goPackage.RepositoryMetadataOptions = 0
        goPackage.UseOLEDBServiceComponents = True
        goPackage.LogToSQLServer = False
        goPackage.LogServerName = "(local)"
        goPackage.LogServerFlags = 256
        goPackage.FailPackageOnLogFailure = False
        goPackage.ExplicitGlobalVariables = True
        goPackage.PackageType = 0

'-------------------------------------------------------------------------
' begin to write package global variables information
'-------------------------------------------------------------------------

        Dim oGlobal As DTS.GlobalVariable
```

```
        Set oGlobal = goPackage.GlobalVariables.New("cdetail")
        oGlobal = """the"""
        goPackage.GlobalVariables.Add oGlobal
        Set oGlobal = Nothing

        Set oGlobal = goPackage.GlobalVariables.New("ydetail")
        oGlobal = """man"""
        goPackage.GlobalVariables.Add oGlobal
        Set oGlobal = Nothing



'-----------------------------------------------------------------------
' create package connection information
'-----------------------------------------------------------------------

Dim oConnection As DTS.Connection2

'------------ a new connection defined below.
'For security purposes, the password is never scripted

Set oConnection =
goPackage.Connections.New("Microsoft.Jet.OLEDB.4.0")

        oConnection.ConnectionProperties("User ID") = "Admin"
        oConnection.ConnectionProperties("Data Source") =
"E:\examdata\yearmar2000\regexam\FYBCOM"
        oConnection.ConnectionProperties("Extended Properties") =
"dBase 5.0"
```

```
        oConnection.Name = "dBase 5"
        oConnection.ID = 1
        oConnection.Reusable = True
        oConnection.ConnectImmediate = False
        oConnection.DataSource =
"E:\examdata\yearmar2000\regexam\FYBCOM"
        oConnection.UserID = "Admin"
        oConnection.ConnectionTimeout = 60
        oConnection.UseTrustedConnection = False
        oConnection.UseDSL = False

        'If you have a password for this connection, please uncomment
and add your password below.
        'oConnection.Password = "<put the password here>"

goPackage.Connections.Add oConnection
Set oConnection = Nothing

'------------- a new connection defined below.
'For security purposes, the password is never scripted

Set oConnection = goPackage.Connections.New("SQLOLEDB")

        oConnection.ConnectionProperties("Integrated Security") =
"SSPI"
        oConnection.ConnectionProperties("Persist Security Info") =
True
```

```
        oConnection.ConnectionProperties("Initial Catalog") =
"dataexam"
        oConnection.ConnectionProperties("Data Source") = "(local)"
        oConnection.ConnectionProperties("Application Name") = "DTS
Designer"

        oConnection.Name = "Microsoft OLE DB Provider for SQL
Server"
        oConnection.ID = 2
        oConnection.Reusable = True
        oConnection.ConnectImmediate = False
        oConnection.DataSource = "(local)"
        oConnection.ConnectionTimeout = 60
        oConnection.Catalog = "dataexam"
        oConnection.UseTrustedConnection = True
        oConnection.UseDSL = False

        'If you have a password for this connection, please uncomment
and add your password below.
        'oConnection.Password = "<put the password here>"

goPackage.Connections.Add oConnection
Set oConnection = Nothing

'-------------------------------------------------------------------------
' create package steps information
'-------------------------------------------------------------------------

Dim oStep As DTS.Step2
```

```
Dim oPrecConstraint As DTS.PrecedenceConstraint

'------------- a new step defined below

Set oStep = goPackage.Steps.New

        oStep.Name = "DTSStep_DTSDataPumpTask_1"
        oStep.Description = "Transform Data Task: undefined"
        oStep.ExecutionStatus = 4
        oStep.TaskName = "DTSTask_DTSDataPumpTask_1"
        oStep.CommitSuccess = False
        oStep.RollbackFailure = False
        oStep.ScriptLanguage = "VBScript"
        oStep.AddGlobalVariables = True
        oStep.RelativePriority = 3
        oStep.CloseConnection = False
        oStep.ExecuteInMainThread = False
        oStep.IsPackageDSORowset = False
        oStep.JoinTransactionIfPresent = False
        oStep.DisableStep = False
        oStep.FailPackageOnError = False

goPackage.Steps.Add oStep
Set oStep = Nothing

'------------- a new step defined below

Set oStep = goPackage.Steps.New
```

```
        oStep.Name = "DTSStep_DTSActiveScriptTask_1"
        oStep.Description = "ActiveX Script Task: undefined"
        oStep.ExecutionStatus = 4
        oStep.TaskName = "DTSTask_DTSActiveScriptTask_1"
        oStep.CommitSuccess = False
        oStep.RollbackFailure = False
        oStep.ScriptLanguage = "VBScript"
        oStep.AddGlobalVariables = True
        oStep.RelativePriority = 3
        oStep.CloseConnection = False
        oStep.ExecuteInMainThread = False
        oStep.IsPackageDSORowset = False
        oStep.JoinTransactionIfPresent = False
        oStep.DisableStep = False
        oStep.FailPackageOnError = False

goPackage.Steps.Add oStep
Set oStep = Nothing

'------------ a precedence constraint for steps defined below

Set oStep = goPackage.Steps("DTSStep_DTSActiveScriptTask_1")
Set oPrecConstraint =
oStep.precedenceConstraints.New("DTSStep_DTSDataPumpTask_1")
        oPrecConstraint.StepName = "DTSStep_DTSDataPumpTask_1"
        oPrecConstraint.PrecedenceBasis = 0
        oPrecConstraint.Value = 4

oStep.precedenceConstraints.Add oPrecConstraint
```

```
Set oPrecConstraint = Nothing

'-------------------------------------------------------------------------
' create package tasks information
'-------------------------------------------------------------------------

'------------- call Task_Sub1 for task DTSTask_DTSDataPumpTask_1
(Transform Data Task: undefined)
Call Task_Sub1(goPackage)

'------------- call Task_Sub2 for task DTSTask_DTSActiveScriptTask_1
(ActiveX Script Task: undefined)
Call Task_Sub2(goPackage)

'-------------------------------------------------------------------------
' Save or execute package
'-------------------------------------------------------------------------

'goPackage.SaveToSQLServer "(local)", "sa", ""
goPackage.Execute
goPackage.Uninitialize
'to save a package instead of executing it, comment out the
executing package line above and uncomment the saving package
line
Set goPackage = Nothing

Set goPackageOld = Nothing

End Sub
```

```
'------------- define Task_Sub1 for task
DTSTask_DTSDataPumpTask_1 (Transform Data Task: undefined)
Public Sub Task_Sub1(ByVal goPackage As Object)

Dim oTask As DTS.Task
Dim oLookup As DTS.Lookup

Dim oCustomTask1 As DTS.DataPumpTask2
Set oTask = goPackage.Tasks.New("DTSDataPumpTask")
Set oCustomTask1 = oTask.CustomTask

        oCustomTask1.Name = "DTSTask_DTSDataPumpTask_1"
        oCustomTask1.Description = "Transform Data Task: undefined"
        oCustomTask1.SourceConnectionID = 1
        oCustomTask1.SourceSQLStatement = "UPDATE    Finaltable
SET  Yeardetail = DTSGlobalVariables('ydetail').Value, courcedetail
=DTSGlobalVariables('cdetail').Value WHERE     (Yeardetail = NULL)"
        oCustomTask1.DestinationConnectionID = 2
        oCustomTask1.DestinationObjectName =
"[dataexam].[dbo].[Finaltable]"
        oCustomTask1.ProgressRowCount = 1000
        oCustomTask1.MaximumErrorCount = 0
        oCustomTask1.FetchBufferSize = 1
        oCustomTask1.UseFastLoad = True
        oCustomTask1.InsertCommitSize = 0
        oCustomTask1.ExceptionFileColumnDelimiter = "|"
        oCustomTask1.ExceptionFileRowDelimiter = vbCrLf
```

```
        oCustomTask1.AllowIdentityInserts = False
        oCustomTask1.FirstRow = "0"
        oCustomTask1.LastRow = "0"
        oCustomTask1.FastLoadOptions = 2
        oCustomTask1.ExceptionFileOptions = 1
        oCustomTask1.DataPumpOptions = 0

Call oCustomTask1_Trans_Sub1(oCustomTask1)
Call oCustomTask1_Trans_Sub2(oCustomTask1)
Call oCustomTask1_Trans_Sub3(oCustomTask1)
Call oCustomTask1_Trans_Sub4(oCustomTask1)
Call oCustomTask1_Trans_Sub5(oCustomTask1)



goPackage.Tasks.Add oTask
Set oCustomTask1 = Nothing
Set oTask = Nothing

End Sub

Public Sub oCustomTask1_Trans_Sub1(ByVal oCustomTask1 As
Object)

        Dim oTransformation As DTS.Transformation2
        Dim oTransProps As DTS.Properties
        Dim oColumn As DTS.Column
        Set oTransformation =
oCustomTask1.Transformations.New("DTS.DataPumpTransformCopy"
)
```

```
        oTransformation.Name = "DTSTransformation__1"
        oTransformation.TransformFlags = 63
        oTransformation.ForceSourceBlobsBuffered = 0
        oTransformation.ForceBlobsInMemory = False
        oTransformation.InMemoryBlobSize = 1048576
        oTransformation.TransformPhases = 4

        Set oColumn =
oTransformation.SourceColumns.New("colgname", 1)
            oColumn.Name = "colgname"
            oColumn.Ordinal = 1
            oColumn.Flags = 106
            oColumn.Size = 50
            oColumn.DataType = 130
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True

        oTransformation.SourceColumns.Add oColumn
        Set oColumn = Nothing

        Set oColumn =
oTransformation.DestinationColumns.New("colgname", 1)
            oColumn.Name = "colgname"
            oColumn.Ordinal = 1
            oColumn.Flags = 104
            oColumn.Size = 50
            oColumn.DataType = 130
            oColumn.Precision = 0
```

```
                oColumn.NumericScale = 0
                oColumn.Nullable = True

            oTransformation.DestinationColumns.Add oColumn
            Set oColumn = Nothing

        Set oTransProps = oTransformation.TransformServerProperties



        Set oTransProps = Nothing

        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing

End Sub

Public Sub oCustomTask1_Trans_Sub2(ByVal oCustomTask1 As
Object)

        Dim oTransformation As DTS.Transformation2
        Dim oTransProps As DTS.Properties
        Dim oColumn As DTS.Column
        Set oTransformation =
oCustomTask1.Transformations.New("DTS.DataPumpTransformCopy"
)
            oTransformation.Name = "DTSTransformation__2"
            oTransformation.TransformFlags = 63
            oTransformation.ForceSourceBlobsBuffered = 0
            oTransformation.ForceBlobsInMemory = False
```

```
        oTransformation.InMemoryBlobSize = 1048576
        oTransformation.TransformPhases = 4

        Set oColumn =
oTransformation.SourceColumns.New("centername", 1)
            oColumn.Name = "centername"
            oColumn.Ordinal = 1
            oColumn.Flags = 106
            oColumn.Size = 15
            oColumn.DataType = 130
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True

        oTransformation.SourceColumns.Add oColumn
        Set oColumn = Nothing

        Set oColumn =
oTransformation.DestinationColumns.New("centername", 1)
            oColumn.Name = "centername"
            oColumn.Ordinal = 1
            oColumn.Flags = 104
            oColumn.Size = 15
            oColumn.DataType = 130
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True

        oTransformation.DestinationColumns.Add oColumn
```

```
            Set oColumn = Nothing

        Set oTransProps = oTransformation.TransformServerProperties


        Set oTransProps = Nothing

        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing

End Sub

Public Sub oCustomTask1_Trans_Sub3(ByVal oCustomTask1 As
Object)

        Dim oTransformation As DTS.Transformation2
        Dim oTransProps As DTS.Properties
        Dim oColumn As DTS.Column
        Set oTransformation =
oCustomTask1.Transformations.New("DTS.DataPumpTransformCopy"
)
            oTransformation.Name = "DTSTransformation__3"
            oTransformation.TransformFlags = 63
            oTransformation.ForceSourceBlobsBuffered = 0
            oTransformation.ForceBlobsInMemory = False
            oTransformation.InMemoryBlobSize = 1048576
            oTransformation.TransformPhases = 4
```

```
        Set oColumn =
oTransformation.SourceColumns.New("SEATNO", 1)
            oColumn.Name = "SEATNO"
            oColumn.Ordinal = 1
            oColumn.Flags = 122
            oColumn.Size = 0
            oColumn.DataType = 5
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True

        oTransformation.SourceColumns.Add oColumn
        Set oColumn = Nothing

        Set oColumn =
oTransformation.DestinationColumns.New("SEATNO", 1)
            oColumn.Name = "SEATNO"
            oColumn.Ordinal = 1
            oColumn.Flags = 120
            oColumn.Size = 0
            oColumn.DataType = 5
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True

        oTransformation.DestinationColumns.Add oColumn
        Set oColumn = Nothing

    Set oTransProps = oTransformation.TransformServerProperties
```

```
        Set oTransProps = Nothing

        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing

End Sub

Public Sub oCustomTask1_Trans_Sub4(ByVal oCustomTask1 As
Object)

        Dim oTransformation As DTS.Transformation2
        Dim oTransProps As DTS.Properties
        Dim oColumn As DTS.Column
        Set oTransformation =
oCustomTask1.Transformations.New("DTS.DataPumpTransformCopy"
)
                oTransformation.Name = "DTSTransformation__4"
                oTransformation.TransformFlags = 63
                oTransformation.ForceSourceBlobsBuffered = 0
                oTransformation.ForceBlobsInMemory = False
                oTransformation.InMemoryBlobSize = 1048576
                oTransformation.TransformPhases = 4

                Set oColumn =
oTransformation.SourceColumns.New("CENTRE", 1)
                        oColumn.Name = "CENTRE"
                        oColumn.Ordinal = 1
```

```
                oColumn.Flags = 122
                oColumn.Size = 0
                oColumn.DataType = 5
                oColumn.Precision = 0
                oColumn.NumericScale = 0
                oColumn.Nullable = True

            oTransformation.SourceColumns.Add oColumn
            Set oColumn = Nothing

            Set oColumn =
oTransformation.DestinationColumns.New("CENTRE", 1)
                oColumn.Name = "CENTRE"
                oColumn.Ordinal = 1
                oColumn.Flags = 120
                oColumn.Size = 0
                oColumn.DataType = 5
                oColumn.Precision = 0
                oColumn.NumericScale = 0
                oColumn.Nullable = True

            oTransformation.DestinationColumns.Add oColumn
            Set oColumn = Nothing

        Set oTransProps = oTransformation.TransformServerProperties


        Set oTransProps = Nothing
```

```
        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing

End Sub

Public Sub oCustomTask1_Trans_Sub5(ByVal oCustomTask1 As
Object)

        Dim oTransformation As DTS.Transformation2
        Dim oTransProps As DTS.Properties
        Dim oColumn As DTS.Column
        Set oTransformation =
oCustomTask1.Transformations.New("DTS.DataPumpTransformCopy"
)
                oTransformation.Name = "DTSTransformation__5"
                oTransformation.TransformFlags = 63
                oTransformation.ForceSourceBlobsBuffered = 0
                oTransformation.ForceBlobsInMemory = False
                oTransformation.InMemoryBlobSize = 1048576
                oTransformation.TransformPhases = 4

                Set oColumn =
oTransformation.SourceColumns.New("COLLEGE", 1)
                        oColumn.Name = "COLLEGE"
                        oColumn.Ordinal = 1
                        oColumn.Flags = 122
                        oColumn.Size = 0
                        oColumn.DataType = 5
                        oColumn.Precision = 0
```

```
                oColumn.NumericScale = 0
                oColumn.Nullable = True

            oTransformation.SourceColumns.Add oColumn
            Set oColumn = Nothing

            Set oColumn =
oTransformation.DestinationColumns.New("COLLEGE", 1)
                oColumn.Name = "COLLEGE"
                oColumn.Ordinal = 1
                oColumn.Flags = 120
                oColumn.Size = 0
                oColumn.DataType = 5
                oColumn.Precision = 0
                oColumn.NumericScale = 0
                oColumn.Nullable = True

            oTransformation.DestinationColumns.Add oColumn
            Set oColumn = Nothing

        Set oTransProps = oTransformation.TransformServerProperties



        Set oTransProps = Nothing

        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing

End Sub
```

```
Public Sub oCustomTask1_Trans_Sub110(ByVal oCustomTask1 As
Object)

    Dim oTransformation As DTS.Transformation2
    Dim oTransProps As DTS.Properties
    Dim oColumn As DTS.Column
    Set oTransformation =
oCustomTask1.Transformations.New("DTSPump.DataPumpTransform
Script")
        oTransformation.Name = "DTSTransformation__110"
        oTransformation.TransformFlags = 63
        oTransformation.ForceSourceBlobsBuffered = 0
        oTransformation.ForceBlobsInMemory = False
        oTransformation.InMemoryBlobSize = 1048576
        oTransformation.TransformPhases = 4

        Set oColumn =
oTransformation.DestinationColumns.New("Yeardetail", 1)
            oColumn.Name = "Yeardetail"
            oColumn.Ordinal = 1
            oColumn.Flags = 104
            oColumn.Size = 50
            oColumn.DataType = 129
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True
```

```
            oTransformation.DestinationColumns.Add oColumn
            Set oColumn = Nothing

        Set oTransProps = oTransformation.TransformServerProperties

            oTransProps("Text") =
"'**************************************************
**********" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "'  Visual
Basic Transformation Script" & vbCrLf
            oTransProps("Text") = oTransProps("Text") &
"'**************************************************
**********" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "'  Copy
each source column to the destination column" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "Function
Main()" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "   Main =
DTSTransformStat_OK" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "End
Function"
            oTransProps("Language") = "VBScript"
            oTransProps("FunctionEntry") = "Main"

        Set oTransProps = Nothing

        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing
```

```
End Sub

Public Sub oCustomTask1_Trans_Sub111(ByVal oCustomTask1 As
Object)

    Dim oTransformation As DTS.Transformation2
    Dim oTransProps As DTS.Properties
    Dim oColumn As DTS.Column
    Set oTransformation =
oCustomTask1.Transformations.New("DTSPump.DataPumpTransform
Script")
        oTransformation.Name = "DTSTransformation__111"
        oTransformation.TransformFlags = 63
        oTransformation.ForceSourceBlobsBuffered = 0
        oTransformation.ForceBlobsInMemory = False
        oTransformation.InMemoryBlobSize = 1048576
        oTransformation.TransformPhases = 4

        Set oColumn =
oTransformation.DestinationColumns.New("courcedetail", 1)
            oColumn.Name = "courcedetail"
            oColumn.Ordinal = 1
            oColumn.Flags = 104
            oColumn.Size = 50
            oColumn.DataType = 129
            oColumn.Precision = 0
            oColumn.NumericScale = 0
            oColumn.Nullable = True
```

```
            oTransformation.DestinationColumns.Add oColumn
            Set oColumn = Nothing

        Set oTransProps = oTransformation.TransformServerProperties

            oTransProps("Text") =
"'**************************************************
**********" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "'  Visual
Basic Transformation Script" & vbCrLf
            oTransProps("Text") = oTransProps("Text") &
"'**************************************************
**********" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "'  Copy
each source column to the destination column" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "Function
Main()" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "   Main =
DTSTransformStat_OK" & vbCrLf
            oTransProps("Text") = oTransProps("Text") & "End
Function"
            oTransProps("Language") = "VBScript"
            oTransProps("FunctionEntry") = "Main"

        Set oTransProps = Nothing

        oCustomTask1.Transformations.Add oTransformation
        Set oTransformation = Nothing
```

```
End Sub

'------------ define Task_Sub2 for task
DTSTask_DTSActiveScriptTask_1 (ActiveX Script Task: undefined)
Public Sub Task_Sub2(ByVal goPackage As Object)

Dim oTask As DTS.Task
Dim oLookup As DTS.Lookup

Dim oCustomTask2 As DTS.ActiveScriptTask
Set oTask = goPackage.Tasks.New("DTSActiveScriptTask")
Set oCustomTask2 = oTask.CustomTask

        oCustomTask2.Name = "DTSTask_DTSActiveScriptTask_1"
        oCustomTask2.Description = "ActiveX Script Task: undefined"
        oCustomTask2.ActiveXScript =
"'**********************************************************
**********" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "'
Visual Basic ActiveX Script" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript &
"'**********************************************************
**********" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript &
"Function Main()" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
'msgbox DTSGlobalVariables(""ydetail"").Value" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Dim oPkg, oExecSQL, sSQLStatement" & vbCrLf
```

```
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
' Build new SQL Statement" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
sSQLStatement = ""UPDATE    Finaltable  SET  Yeardetail =
DTSGlobalVariables('ydetail').Value, courcedetail
=DTSGlobalVariables('cdetail').Value WHERE     (Yeardetail =
NULL)""" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
dim conn" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
set conn = DTSGlobalVariables(""MyConn"").value" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
conn.provider=""sqloledb""" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
conn.open  ""(local)"", ""sa"", """""" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
conn.DefaultDatabase = ""pubs""" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
conn.execute(""Create Table YTDSales (Totals int)"")" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Main = DTSTaskExecResult_Success" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
' Clean Up" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Set oExecSQL = Nothing" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Set oPkg = Nothing" & vbCrLf
	oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
' Clean Up" & vbCrLf
```

```
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Set oDataPump = Nothing" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Set oPkg = Nothing" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript & "
Main = DTSTaskExecResult_Success" & vbCrLf
        oCustomTask2.ActiveXScript = oCustomTask2.ActiveXScript &
"End Function"
        oCustomTask2.FunctionName = "Main"
        oCustomTask2.ScriptLanguage = "VBScript"
        oCustomTask2.AddGlobalVariables = True

goPackage.Tasks.Add oTask
Set oCustomTask2 = Nothing
Set oTask = Nothing

End Sub
```

4. Web Mining Extracted HTML Code

<table width="580" id="yfncsumtab" cellpadding="0" cellspacing="0" border="0"><tr><td colspan="2"></td></tr><tr valign="top"><td width="100%" colspan="3"><table width="100%"
class="yfnc_modtitlew1" cellpadding="2" cellspacing="0" border="0"><tr
class="yfnc_modtitle1"><td><small>TATA CONSULTANCY SERV LT (NSE:TCS.NS) Delayed quote
data</small></td><td align="right"></td></tr></table><table border=0 cellspacing=0 cellpadding=0
width="1"><tr><td height=1><spacer type=block height=1 width=1></td></tr></table><table cellpadding="0"
cellspacing="0" border="0" width="580" class="yfnc_modtitlew1"><tr valign="top"><td><table width="185"
class="yfncsumdatagrid" cellpadding="0" cellspacing="0" border="0"><tr valign="top"><td
class="yfnc_datamodoutline1"><table width="100%" cellpadding="2" cellspacing="1" border="0"><tr><td
class="yfnc_tablehead1" width="48%">Last Trade:</td><td
class="yfnc_tabledata1"><big>1,714.40</big></td></tr><tr><td
class="yfnc_tablehead1"
width="48%">Trade Time:</td><td class="yfnc_tabledata1">Dec 16</td></tr><tr><td
class="yfnc_tablehead1"
width="48%">Change:</td><td class="yfnc_tabledata1"><img width="10"

height="14" border="0"

src="http://us.i1.yimg.com/us.yimg.com/i/us/fi/03rd/up_g.gif" alt="Up"> <b

style="color:#008800;">31.60

(1.88%)</td></tr><tr><td class="yfnc_tablehead1"
width="48%">Prev Close:</td><td
class="yfnc_tabledata1">1,682.80</td></tr><tr><td
class="yfnc_tablehead1" width="48%">Open:</td><td
class="yfnc_tabledata1">1,690.00</td></tr><tr><td
class="yfnc_tablehead1" width="48%">Bid:</td><td
class="yfnc_tabledata1">1,714.40</td></tr><tr><td
class="yfnc_tablehead1" width="48%">Ask:</td><td
class="yfnc_tabledata1">1,720.00</td></tr><tr><td
class="yfnc_tablehead1" width="48%">1y Target
Est:</td><td
class="yfnc_tabledata1">N/A</td></tr></table></td></tr></table
></td><td width="5"
nowrap><spacer type="block" width="5"
height="1"></spacer></td><td><table width="185"
class="yfncsumdatagrid" cellpadding="0" cellspacing="0"
border="0"><tr valign="top"><td
class="yfnc_datamodoutline1"><table width="100%"
cellpadding="2" cellspacing="1" border="0"><tr><td
class="yfnc_tablehead1" width="48%">Day's Range:</td><td
class="yfnc_tabledata1">1,683.00 -
1,726.00</td></tr><tr><td class="yfnc_tablehead1"
width="48%">52wk Range:</td><td
class="yfnc_tabledata1">1,093.30 - 1,705.00</td></tr><tr><td

class="yfnc_tablehead1"
width="48%">Volume:</td><td
class="yfnc_tabledata1">988,429</td></tr><tr><td
class="yfnc_tablehead1"

width="48%">Avg Vol <small>(3m)</small>:</td><td
class="yfnc_tabledata1">705,432</td></tr><tr><td
class="yfnc_tablehead1" width="48%">Market Cap:</td><td
class="yfnc_tabledata1">N/A</td></tr><tr><td
class="yfnc_tablehead1" width="48%">P/E
<small>(ttm)</small>:</td><td
class="yfnc_tabledata1">N/A</td></tr><tr><td
class="yfnc_tablehead1" width="48%">EPS
<small>(ttm)</small>:</td><td
class="yfnc_tabledata1">0.00</td></tr><tr><td
class="yfnc_tablehead1"
width="48%">Div & Yield:</td><td
class="yfnc_tabledata1">N/A
(N/A)</td></tr></table></td></tr></table>

5. Perl Code for Text Mining

```
#!/usr/bin/perl
use DBI;
use Archive::Zip;
use File::Basename;
use HTML::TreeBuilder;
use HTML::FormatText;
use WWW::Mechanize;
use Cwd;
my @filename;
my $j;
&Main();
sub Main(){
    $count=0;
    $total=0;
    $working=0;
    $i=0;
    chdir("data");
    $root = "$0";
    $root=dirname($root)."/";
    #print "\n $root";
    opendir(DIR,$root);
    print "\n   ".  scalar localtime;
    while (defined($file = readdir(DIR)))
    {
        if(-d $file){
            $filename[$i] = $file;
            $i++;
```

```
        }
    }
    for($j=2;$j<$i;$j++){
        chdir("$filename[$j]");
        chdir("10-K");
        @a = <*.zip>;
        foreach $file(@a){
          $bb=$file;
          $bb =~ s/\.zip/\.txt/g;
              $cc = $bb;
              $cc =~ s/\.txt/\.zip/g;
              $dd = $bb;
            if(-f $cc){
              if(-s $cc  > 22){
                my $zip = Archive::Zip->new();
                print 'read error' unless $zip->read( $cc ) ==
AZ_OK;
                    my ( $member, $status, $bufferRef );
                    $member = $zip->memberNamed( "$bb" );
                    $member->desiredCompressionMethod(
                                COMPRESSION_STORED );
                    $status = $member->rewindData();
                    open (OUTFILE, ">$bb");
                    close (OUTFILE);
                    open (OUTFILE, ">>$bb");
                    print "error $status" unless $status == AZ_OK;
                    while (! $member->readIsDone() )
                    {
                        ( $bufferRef, $status ) = $member->
```

```
                                        readChunk();
                print (OUTFILE $$bufferRef);
        }
        close (OUTFILE);
        $total++;

        my $InputText = &ExtractBalance($bb);
        if(length($InputText) > 10){
                        $working++;
                        $dd =~ s/\.txt/\.bal/g;
                        open(OFILE, ">$dd");
                        print(OFILE $InputText);
                        close(OFILE);
                }
                else
                {
                        my $InputText =
&ExtractSpecialBalance($bb);
    $dd =~ s/\.txt/\.bal/g;
    open(OFILE, ">$dd");
    print(OFILE $InputText);
    close(OFILE);
   }


        }
        else
        {
              print "\n Less size $aa==$cc ";
```

```
                        $count++;
                    }
                }
                else
                {
                    print "\nFile Not Available $aa==$cc";
                    $count++;
                }
            }
            chdir("..");
            chdir("..");
    }

    chdir("..");
    print "\n Total Working files are $working outof $total";
    print "\n   ".  scalar localtime;
}



sub ExtractBalance                                          #
{
   my $filename1 = shift(@_);
   my $Output    = "";
   my $count     = 0;
   my $row =0;
   my $stat=0;
   my $filetype=0;
   open(INFILE, $filename1);
```

```
while(<INFILE>){
     if($_ =~ m/<HTML>/i)
     {
             $filetype=1;
             last;
       }
       elsif($_ =~ m/<TABLE>/i)
       {
             $filetype=2;
             last;
       }

}

#print "\nInside Balance";
close(INFILE);

if($filetype==2){
  open(INFILE, $filename1);
  BEGIN : while(<INFILE>)
   {

     $count=0;
     $row =0;
     $Output    = "";
       $stat=0;
```

```
                if($_ =~
m/<TABLE>|balance\s*?sheet[s]?|FINANCIAL\s*?POSITION/i)
{
                 if($_ =~ m/<TABLE>/i){
                            $stat=1;
                       }
                       else
                       {
                            $stat=0;
                       }
                      while(my $innerline = <INFILE>)
                        {
                            $row++;
                                    if($innerline =~
m/balance\s*?sheet[s]?|FINANCIAL\s*?POSITION/i && $stat==1)
                            {
                                $count = 1;
                            }
                            elsif($innerline =~ m/<TABLE>/i &&
$stat==0)
                            {
                                $count = 1;
                            }
                            elsif($count == 0 && $innerline =~
m/<\/TABLE>/i){
                                                    next BEGIN;
                                    }
                                    elsif($count == 1 && $innerline =~
m/<\/TABLE>/i){
```

```
                                        goto  WriteFile;
                                }
                                elsif($count == 1){
                                        $Output =
$Output.$innerline;
                                }

                                if($row > 5 && $count == 0){
                                        next BEGIN;
                                }

                        }
                WriteFile:   return($Output);
        }


    }
    return($Output);
  }
  elsif($filetype==1){


 my $output  = "";

 open(INFILE, $filename1);

 my @arrBalSheet = ("Balance\\s*?Sheet",
```

```
"Condensed\\s*?Consolidated\\s*?Statement.?\\s*?of\\s*?Financial\\
s*?Position");
   my @patterns = map { qq/$_/ } @arrBalSheet;

   my $lastLine  = "";
   my $count     = 0;

BEGIN1:

    while(<INFILE>)
    {
        $_ = $lastLine . $_;
        for my $pat (@patterns)
        {
            if($_ =~ m/$pat/i)
            {
                my $htmlTable = $_;

                $lastLine = substr($_, length($lastLine));

                while(my $outerline = <INFILE>)
                {
                    $outerline = $lastLine . $outerline;


                    if($outerline !~ m/<\/table>/i)
                    {
```

```
                    $htmlTable .= substr($outerline,
length($lastLine));
                }
                elsif($htmlTable =~ m/<table/i)
                {

                        $htmlTable = $htmlTable .
substr($outerline, length($lastLine));
                    if($htmlTable =~ m/Total Assets|Total Liabilities/i
&& $htmlTable !~ m/month.?\s*?ended/i)
                    {
                        $output .= "<table>";
                                $output .=  $htmlTable;

                        $count = 1;

                        $lastLine = substr($outerline,
length($lastLine));
                        next BEGIN1;
                    }
                    next BEGIN1;
                }
                else
                {
                    $htmlTable .= substr($outerline,
length($lastLine));
                }
                $lastLine = substr($outerline, length($lastLine));
            }
```

```
                next BEGIN1;
            }
            elsif( ($_ =~
m/Notes\s*?to\s*?.*?\s*?.*?\s*?financial\s*?statement/i) && ($_ !~
m/[See|accompanying]\s*?Notes\s*?to.*?financial\s*?statement/i)
&& ($count == 1))
            {
                last BEGIN1;
            }
            elsif( ($_ =~
m/Management.?s\s*?Discussion\s*?and\s*?Analysis\s*?of\s*?Finan
cial/i) && ($count == 1))
            {
                last BEGIN1;
            }
            elsif( ($_ =~ m/<\/table>/i) && ($count == 1))
            {
                last BEGIN1;
            }
        }
        $lastLine = substr($_, length($lastLine));
    }
    close(INFILE);
    return($output);




    }
    else{
```

```
        return();
    }
}
sub ExtractSpecialBalance
#
{
   my $filename1 = shift(@_);
   my $Output    = "";
   my $count     = 0;
   my $row =0;
   my $stat=0;
   my $filetype=0;

   open(INFILE, $filename1);
   BEGIN : while(<INFILE>)
     {

        $count=0;
        $row =0;
        $Output    = "";
          $stat=0;
            if($_ =~
m/balance\s*?sheet[s]?|FINANCIAL\s*?POSITION|Financial\s*?Condi
tion/i)
           {
                while(my $innerline = <INFILE>)
                {
                        $row++;
```

```
                                    if($innerline =~ m/in
thousand[s]?\)|In Million[s]?\)/i)
                            {
                                $count = 1;
                            }
                            elsif($count == 1 && $innerline =~
m/accompanying/i){
                                                goto  WriteFile;
                                    }
                                    elsif($count == 1){
                                                $Output =
$Output.$innerline;
                                    }

                                    if($row > 5 && $count == 0){
                                            next BEGIN;
                                    }

                        }
                    WriteFile:   return($Output);
            }



      }
      return($Output);

}
```

References

Books

- Data mining Explained
- A manager's guide to customer-centric business intelligence
  Rhonda Delmater
  Monte Hancock
  Digital Press
- Data Mining
  Pieter Adriaans
  Dolf Zantinge
- Data Warehousing in the real world
- A Pratical guide for Business DSS
  Sam Anahory
  Dennis Murray
- **Data Munging *with Perl***
  DAVID CROSS
- Building the data warehouse
  W.H. Inmon
- Data Warehousing, Data Mining, & OLAP
  Alex Berson, Stephen J. Smith
- Data Mining with Microsoft Sql Server 2000
  Technical Reference
  Claude Seidman
- Data Warehousing
  BPB Publications
- Data Mining
  BPB Publications

Internet Sites

- http://www.kenorrinst.com/dwpaper.html
- http://www.inmoncif.com
- http://www.thearling.com/text/dmwhite/dmwhite.htm
- http://intelligent-web.org/wsm/overview/
- http://en.wikipedia.org/wiki/Text_mining
- http://www.microsoft.com/sql/prodinfo/features/features-at-a-glance.mspx
- http://www.microsoft.com/sql/technologies/dm/default.mspx
- http://www-306.ibm.com/software/data/iminer/
- https://informatica-news.com/
- http://www.businessobjects.com/products/businessobjectsxi/default.asp
- http://www.oracle.com/applications/peoplesoft-enterprise.html
- http://www.cognos.com/
- http://www.microstrategy.com/Software/Products/Service_Modules/DataMining_Services/
- http://dev.hyperion.com/products/intelligence/
- http://www.cs.waikato.ac.nz/~ml/index.html
- http://www.jcp.org/en/jsr/detail?id=247
- http://www.oracle.com/technology/products/bi/odm/index.html
- http://www.spss.com/data_mining/index.htm
- http://www.sas.com/technologies/analytics/datamining/
- http://www.datamininglab.com