# The Concept of Pattern Warehouse and Contemplate an Application in Integrated Network Data Ware

**Jaesoon Park, Youngwok Kim, Youngmin Cha, Seongbum Kim**
**Telecommunication Network Lab., Korea Telecom**
**463-1 Junmin-dong, Yusung-gu, Taejeon, Korea**
**{jaesoon, kimyw, ymcha, sbkimm}@kt.co.kr**

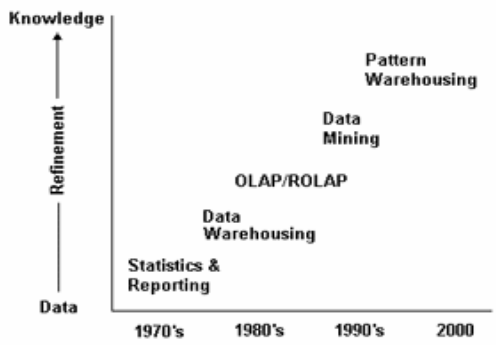
## Abstract

Data management has two facets, management of operational data by systems that process transaction and the management and analysis of historical data by decision support systems that provide business users with insight. As decision support has matured, it has became clear that business users do not want massive data, but are interested in the patterns and trends buried within data. In this article we introduced the concept of pattern management and pattern warehouse and discuss how it is distinct from data management.

## 1. Introduction

The concept of a data warehouse was championed in the 1980's as a repository for corporate data elements. The idea was to create a central storage facility where everyone in the corporation could go and get "data" on demand, whenever they needed it. And, the central repository would help increase corporate data quality and consistency because everyone obtained for data from a single source.

In the 1990's it became clear that data in warehouses are often too coarse and unmanageable for detailed decision making, business users needed much more refined knowledge. Moreover, most organizations realized that what they really wanted was the knowledge, trends and patterns within the data, not the data itself. The concept of "data mining" hence gained momentum and the need for knowledge extraction from data became widely accepted. Business users expected to get refined knowledge, not data.




**(Figure 1) Progress of Knowledge Discovery**

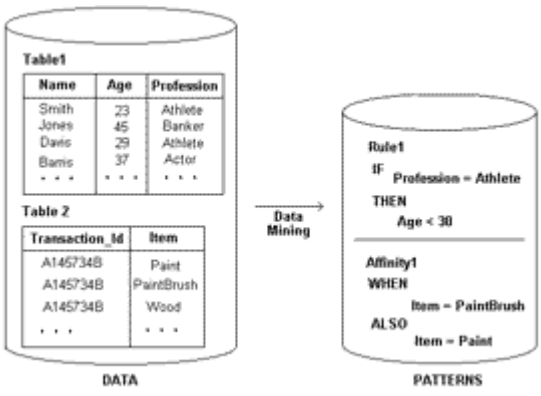
We can view the progress of the field over the last 30 years in terms of a series of steps, each providing better and more refined information.

Once data warehouses were subjected to data mining, business users encountered three immediate issues. First, most business users found the technical details of the data mining task more than they had bargained for. Secondly, piece-meal and fragmented analysis on the central warehouse began to give inconsistent results -- 10 business users could get 8 different answers from the same data, depending on how they approached it. And, the response time for follow up analysis from a large warehouse and the need for analyst intermediaries would often slow down the process of knowledge extraction.

Today, we take the concept of "data management" for granted an entire industry revolves around it. Data management has two facets: management of operational data by systems that process transactions (often in real-time) and the management and analysis of historical data by decision support systems that provide business users with insight. As decision support has matured, it has become clear that business users do not want massive volumes of data, but are interested in the patterns and trends buried within data. These patterns need to be accessed, manipulated and managed, just as data elements are managed.

## 2. Data vs. Patterns

The data is rough but patterns are refined. A pattern expresses relationships between data items, but not the data. There are several classes of patterns, including influence patterns (often reflecting probabilities or likelihood) as well as affinity patterns that deal with associations (e.g. market basket patterns) or comparative patterns that point out differences among data sets. Each pattern class has specific rules of inference for the manipulation of patterns. As a simple analogy, consider data as grapes and patterns of knowledge as wine -- data mining is then like the wine making process. While a data repository is a storage facility for grapes, a pattern repository is like a wine cellar. Data mining tools are then like wine making equipment. Although users can make their own wine by getting grapes from the warehouse, this takes both time and know-how and naturally most users business prefer bottled wine. Note that with pattern management data mining still takes place behind the scenes, but the business user is unaware of it.




**(Figure 2) Data vs. Pattern**

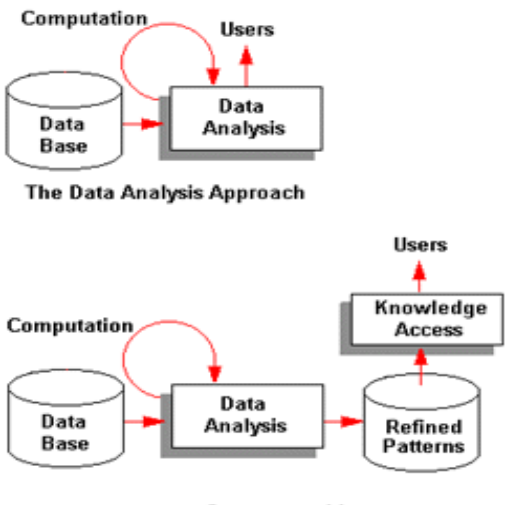
The above figure 2. Illustrate how patterns are distinct from data, consider the sample demographic data in Table 1. This table implicitly includes rule-based patterns that are stored in Pattern-Set1, after they have been discovered by data mining. Similarly, the sales transaction data in Table 2 includes affinity patterns that are stored in Pattern-Set2.

Let us note three facts from this example. First, those patterns are much more condensed than data. The database may contain 1 million or 100 million records, but the pattern-set just uses a few bytes per pattern. Although large databases include more and more patterns, the condensation ratio is immense terabytes of data include megabytes of patterns. Secondly, let us note that different data sets include different pattern types. While Table1 includes rule-based patterns, Table 2 is structured for affinity patterns. Third, let us note that various pattern types may be related. For instance, a class of customers who is highly profitable based on an influence pattern may also have specific affinities that are then used for cross selling. [3,6]

## 3. Data Analysis vs. Pattern Management

To distill information from a database we obviously need to perform analysis at some time. The key question is: "When." In other words, does the analysis takes place at the time the user needs the knowledge or is it done beforehand, with the knowledge ready to access?" Traditionally, data mining analysis were performed upon user request. The knowledge access paradigm rescues users from delayed analysis by pre-mining refined knowledge. Hence there are two distinct paradigms for empowering users with knowledge:

- The Data Analysis Paradigm: in which users operate on data to discover information. This paradigm relies on the "analysis on demand" approach, i.e. when a user wants knowledge, analysis is performed.
- The Knowledge Access Paradigm: in which the analysis is automatically done beforehand, refined patterns are pre-generated and users just get knowledge when needed, i.e. the "knowledge on demand" approach.




**(Figure 3) The Knowledge Access Approach**

The knowledge access paradigm provides a multitude of benefits to the business user:

- Easy to Use, yet powerful: Business

users without technical know-how can access knowledge without training they just click a graphic user interface from within a web-browser. And, the knowledge access approach is more powerful because multiple types of powerful patterns are automatically merged to answer serious questions. With the analysis paradigm, business users inevitably rely on simple models and can not deal with complex situations on their own.

- Fast Response and Overall Efficiency: When a user requests knowledge, no analysis is needed and follow-up questions are answered quickly, without delay. Data mining on a very large database may take time, but pattern look-up is fast. And, because patterns are not re-computed each time for each user, the overall system efficiency is much higher. Computations take place once, and users access the refined knowledge again and again with ease.
- Accuracy and Quality: Because sampling and extract files are avoided, the discovered patterns correspond to the entire database and have high accuracy, resulting in better decisions. And, because patterns are stored in a single repository, all users get similar answers, rather than relying on fragmented analyses. This is in contrast to the data analysis paradigm where different users may draw different conclusions from the same data.
- Condensed Information: Because of disk space limitations, many organizations only store 12 or 24 months worth of historical data. However, because knowledge is so much more compact than data, the Pattern Warehouse is only a fraction of the size of the database, allowing many years' worth of patterns to be stored with ease, even when the data is no longer available.
- Up-to-Date Knowledge: Because the Pattern Warehouse is incrementally updated, recent patterns are always available. With the data analysis paradigm, there is usually not enough time to continuously analyze new data and often users are forced to rely on out of date analyses.

The knowledge access paradigm is a truly revolutionary idea with a multitude of business and technical benefits that reinforce each other. It avoids the probability of 100 users getting 100 different answers from the same data, because now corporate knowledge is centralized.[3,4]

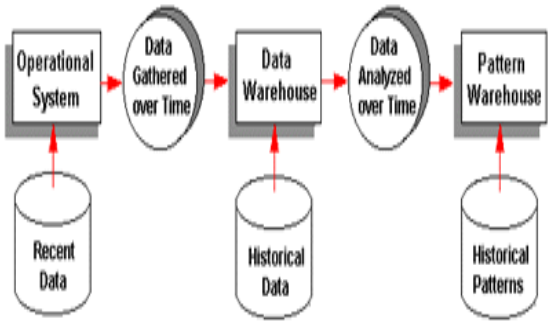
## 4. Data Warehouse vs. Pattern Warehouse

The patterns a data mining system discovers are stored in the Pattern Warehouse. Just as a data warehouse stores data, the Pattern Warehouse stores patterns it is an information repository that stores relationships between data items, but not the data. While data items are stored in data warehouse, we use the

Pattern Warehouse to store the patterns and relationships among them.

A Pattern Warehouse is not a knowledge base. A knowledge base includes information that is usually known to humans, is often hand-coded and is somewhat static changing it will require care and effort. A Pattern Warehouse holds far more dynamic information (which is automatically re-generated once a month with new data) is often surprising to users, and detects trends and patterns of change as they happen.

A Pattern Warehouse is a repository that holds historical patterns rather than historical data. With a pattern warehouse, almost all the relevant patterns in the data are found beforehand, and stored for use by business users such as marketing analysts, bank branch managers, store managers, etc. Business users get the interesting patterns of change every week or month or can query the Pattern Warehouse at will.




**(Figure 4) Data vs. Pattern Warehouse**

Because of disk space limitations, many organizations only store 12 to 18 months worth of historical data and in some cases there are so many transactions that data for only a few months is actually available. However, because knowledge is so much more compact than data, the Pattern Warehouse is only a fraction of the size of the data warehouse, allowing the patterns many years to be stored with ease, even when the data is no longer available.

To get a perspective on the time and space scales, consider an example where the recent operational data refers to one month of a bank's customer information, while the historical data in the data warehouse goes back 1 year. However, the historical patterns in the pattern warehouse may go as far as 5 or 10 years and still be a small fraction of the size of the data warehouse. This provides a huge amount of knowledge over time at a low cost for disk space and response time is far better than the data warehouse because the patterns have already been extracted, ready for look-up. This provides an environment for long-term corporate knowledge management.

## 5. Components of Pattern Management

To deal with patterns, we need to collect, store, manipulate, access and visualize them, we need repositories, query languages and systems to deal with refined patterns rather than raw data. Each of these has an equivalent in the data management world, as shown below.

**(Table-1) Comparative table**

| Data Management | Pattern Management |
|---|---|
| Data /Table | Pattern |
| Data Base | Pattern Base |
| Data Collection | Data Mining |
| SQL | Pattern Oriented Query |
| Relational Algebra | Pattern Algebra |
| Data Visualization | Pattern Visualization |
| Report | Explainable Document |

Patterns can in fact be represented as a set of “pattern-tables” within a traditional relational database. This solves several potential issues regarding user access rights, security control, multi-user access, etc. Obviously, we need a language to access and query the contents of pattern repository. SQL may be considered an obvious first candidate for this, but when SQL was designed over 30 years ago, data mining was not a major issue. SQL was designed to access data stored in databases. We need pattern-oriented languages to access pattern repository storing various types of exact and inexact patterns. Often, it is very hard to access these patterns with SQL. Patterns cannot be conveniently queried in a direct way using a relational query language. Not only are some patterns not easily stored in a simple tabular format, but by just looking up influence factors in pattern-tables we may get incorrect results. We need a “pattern-kernel” that consistently manages and merges patterns. While SQL relies on the relational algebra, pattern query uses the “pattern algebra”. Pattern query process should use SQL as part of its operation, i.e. pattern queries are decomposed into a set of related SQL queries, and then the results are re-combined. However, business users just click on a graphic user interface to retrieve patterns on the intranet. They can begin to access knowledge immediately without lengthy training sessions or analytical know-how.

With pattern visualization the user still performs analysis (e.g. visualizes affinity patterns) the results delivered for the same level of computational effort are orders of magnitude better because the user now analyzes refined knowledge, not data. And now 100 different analysts will no longer get 100 different answers from the same data because there is a central knowledge repository.

A natural way of delivering pattern-based information to users on the web is a document organized as a collection of information of different types, e.g. text, data, graphs, etc. An Explainable Document looks like any other web page at first, but does an incredible amount more by allowing users to dynamically obtain explanations that clarify, justify and substantiate the patterns presented within the document. Explainable documents are in fact a key element of Machine-Man Systems allowing for the intelligent exchange of refined information between users and systems.[4,6]

## 6. Fast Accessing to the Pattern Warehouse

The Pattern Warehouse is represented as a set of "pattern-tables" within a traditional

relational database. This solves several potential issues regarding user access rights, security control, multi-user access, etc.

But obviously, we need a language to access and query the contents of Pattern Warehouses. SQL may be considered an obvious first candidate for this, but when SQL was designed over 30 years ago, data mining was not a major issue. SQL was designed to access data stored in databases. We need pattern-oriented languages to access Pattern Warehouses storing various types of exact and inexact patterns. Often, it is very hard to access these patterns with SQL.

Hence a Pattern Warehouse can't be conveniently queried in a direct way using a relational query language. Not only are some patterns not easily stored in a simple tabular format, but also by just looking up influence factors in pattern-tables we may get incorrect results. We need a "pattern-kernel" that consistently manages and merges patterns. The pattern-kernel forms the heart of PQL(Pattern Query Language), which does for decision support spaces, what SQL does for the data space. While SQL relies on the relational algebra, PQL uses the "pattern algebra". PQL was designed to access Pattern Warehouses just as SQL was designed to access databases. PQL was designed to be very similar to SQL. It allows knowledge-based queries just as SQL allows data based queries. And, PQL uses SQL as part of its operation, i.e. PQL queries are decomposed into a set of related SQL queries, then the results are re-combined. However, business users do not usually see PQL. They just click on a graphic user interface to retrieve patterns on the intranet. They can begin to access knowledge immediately just by clicking on a browser-based graphic user interface without lengthy training sessions or analytical know-how. Using PQL has a multitude of technical and business benefits that reinforce each other. Not only does it provide faster response with less computing, but delivers more accurate, consistent and higher quality knowledge. Responses to knowledge queries are more efficient because patterns have already been pre-computed. Avoiding the repeated discovery sessions that are unknowingly performed by multiple analysts reduces the overall computational burden. In many cases, avoiding repeat discovery sessions performed by the same analyst is itself a significant benefit. With the PQL the user still performs analysis (e.g. visualizes affinity patterns) the results delivered for the same level of computational effort are orders of magnitude better because the user now analyzes refined knowledge, not data. And now 100 different analysts will no longer get 100 different answers from the same data because there is a central knowledge repository.

## 7. Web-Delivery of Knowledge

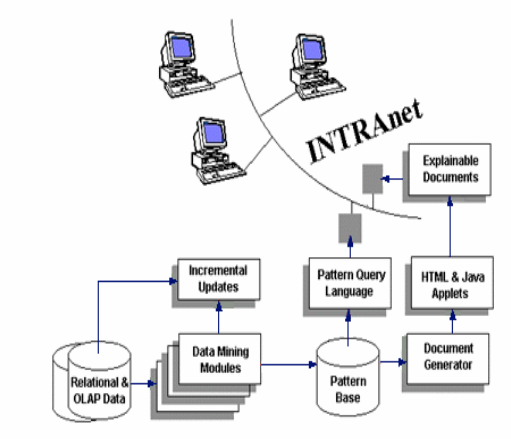
The web is a natural medium for delivering knowledge. As more and more corporations deploy intranets, it has become an essential vehicle for information sharing. The Pattern Warehouse should naturally be supported on

the corporate intranet and can easily reach out to the Internet.

Inter/Intranets and Pattern Warehouses work well together because the net easily delivers what the Pattern Warehouse stores. And, because knowledge is not measured by volume, but by its content and impact, what moves across the intranet does not have high volume, but has high impact. The result is the ability to access and distribute knowledge with unprecedented ease. And other Inter/Intranet resources are just a mouse-click away. The knowledge delivery method used in conjunction with PQL: The Pattern Query Language is browser-based and the graphic user interface uses JavaScript so you can easily tailor it to your specific needs. And, web-based delivery eases the bottleneck of corporate software distribution, making it possible to support many users.

A natural way of delivering information to users on the web is a document organized as a collection of information of different types, e.g. text, data, graphs, etc. An Explainable Document looks like any other web page at first, but does an incredible amount more by allowing users to dynamically obtain explanations that clarify, justify and substantiate the information presented within the document.

Explainable documents are automatically composed and contain English text and graphs that the system generates all by itself. Each piece of information, be it a sentence or a graph, is generated for a specific purpose and makes a specific point. When a user requests an explanation for a piece of information, the system composes a concise and to the point explanation and presents it as yet another explainable document.




**(Figure 5) Web Delivery Architecture**

## 8. Conclusion

In this article we are introduce the concept of Pattern Management and Pattern Warehouse. Also we are compared the data warehouse with the pattern warehouse and we knows that the pattern warehouses can be used both for category management and other specific analysis which take advantage of patterns.

For the next few years in KT, we encountered with the new integrated network management related project like that Service Assurance, Network Management, Quality Management, Integrated Facility, Work Force Management and Network Data Ware.

Especially up to now a practical used method in Network Data Ware related project is a Data Warehouse construction technique.

In the process of Data Warehousing, Everybody knows the fact is that business

users do not want massive volumes of data and integration, but are interested in the patterns and trends buried within data. Therefore we are making a wider and deeper study of pattern warehouse and make a plan to application in various fields of next generation integrated network management.

## [ References ]


[1] R. Baeza-Yates and B. Riberio-Neto. Modern Information Retrieval. Reading , MA. Addison-Wesley, 1999.

[2] M.S. Chen, J.Han, and P.S.Yu. Data mining: An overview from a database perspective. IEEE Trans. Knowledge and Data Engineering, 1996.

[3] Information Discovery Inc, http://www.datamining.com

[4] Data Miners, (Michael J A Berry), http://www.data-miners.com

[5] J.Peiand J. Han. Can we push more Constraints into frequent pattern Mining? In Proc.2000Int. Conf.Knowledge Discovery and Data Mining (KDD'00), Boston, MA, Aug 2000.

[6] W.Shen,K.Ong, B. Mitbander, and C. Zaniolo. Metaquries for data mining. In U.M. Fayyad, G.Piatestky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in knowledge Discivery and Data Mining, Cambridge, MA: AAA/MIT Press,1996.



박 재 순
1989 충남대학교 계산통계학과 석사
1996 충남대학교 계산통계학과 박사
1990 - 2000 한국통신 연수원 교수
2001 - 현재 한국통신 통신망연구소 선임보 연구원
관심분야 : TMN, 데이터 마이닝

김 영 옥
1985 서울대학교 계산통학과 학사
1989 성균관대학교 정보처리학과 석사
1985 -현재 한국통신 통신망연구소 선임연구원
관심분야 : TMN, 데이타웨어하우스

차 영 민
1995 전북대학교전자계산학과 학사
1995 - 현재 한국통신 통신망연구소 전임 연구원
관심분야 : TMN, XML, IP/QoS

김 성 범
1980 한양대학교 전자공학과 학사
1987 한양대학교 전자공학과 석사
1982 - 1983 한국전자통신연구소 (ETRI) 연구원
1982 - 1983 독일 HHI(Heinrich Hertz Institute) 연구소 교환 연구원
1984 -현재 한국통신 통신망연구소 책임연구원
관심분야 : 네트워크 및 분산처리 시스템관리, 분산데이타베이스