Evaluating and Selecting Data Mining Tools

by Mike Ferguson

Over the last few years, many organizations have invested money developing data warehouses and data marts. They have implemented systematic schemes for extracting data from a variety of operational sources, purchased new tools to help with data cleaning and collating, designed new database structures to summarize and store the data, and purchased new hardware for the database servers. The justification for these efforts has varied widely, particularly in the degree of focus on specific business questions to be addressed. While subject-oriented data marts are usually designed to meet a well-defined business need, enterprise-wide data warehousing efforts have often had less specific goals. Some took the “If you build it, they will come” approach.

Generally, regardless of the original motivation for investing in warehousing, that prophecy will come true. In this era of computer information systems, most organizations are drowning in data while starving for real information. Build a warehouse or a data mart offering data in a form that permits easy analysis, and there is likely to be an upsurge of interest in data analysis. Business analysts want to find new ways to increase profitability, budget watchers want to maximize the return on their investment in infrastructure.

Such is the climate that has led to the widespread current interest in data mining, also known as database exploration, information discovery, or knowledge discovery. This article is an overview of data mining, addressing:

- A framework for understanding the elements of data mining
- The mathematical and statistical techniques that data mining tools are based on
- Some applications of these data mining techniques
- A set of evaluation criteria for selecting data mining products
- Planning considerations when introducing today’s (still immature) data mining technology

Knowing What to Look for

No matter how much integrated and summarized data it contains, every data warehouse must include a set of reporting tools that can turn the data into useful information. Data analysis products offer functions ranging from basic query and reporting capabilities to sophisticated development tools for creating online analytical processing, or OLAP, applications. In practice, different tools are aimed at different warehouse users. The most powerful tools are typically used by information object producers, while visual reporting and data delivery tools are designed for information consumers.

Despite their many differences, however, most of these analysis tools share one key characteristic: they depend on their users to guide the investigation. They require a predefined starting point — a hypothesis, a query, a procedure, or a program — that dictates the nature of the data analysis to be carried out by the tool. Naturally, a tool (assuming it is easy to use) does provide its user the facility to more easily explore the contents of a database and test out various hypotheses. However, the tools themselves do not address the need to uncover previously unknown business facts. Yet this is a very real requirement. Thanks to our computers, we have more data than we can handle. As a result, many areas of decision making ‘territory’ that could have potentially enormous value to a business still remain totally unexplored because no one knows of their existence. Today, data mining technology is taking the first step into this unknown territory.

In general, the goal of data mining is to bring the practice of information processing closer to providing the real answers organizations are seeking from their data. Commercial users hope that data mining will supplement human management insights, allowing the business to make more proactive, knowledge-driven decisions in their quest to remain competitive.

A Data Mining Framework

Figure 1 illustrates a simple framework that will help explain what is involved in data mining. Based in part on an analysis by Two Crows Corporation,1 the framework depicts the following five major elements: the business problem, data mining approaches, specific techniques, algorithms or implementations, and tools or products.

The business problem. This, of course, must be the starting point in any consideration of data mining technology. For example, you would like to answer questions like, “Which customers are most likely to default, and why?” Data mining tools seek to address two key business requirements:
• **Description**—discovering patterns, associations and clusters of information (such as customer buying patterns).

• **Prediction**—using those patterns to predict future trends and behaviors (such as the likelihood of a customer defaulting on a loan).

**Data mining approaches.** These are the classes or categories of data mining methods used for *description* and/or *prediction*. There are six major classes:

• **Classification** (developing profiles of groups of items in terms of their attributes)

• **Regression** (establishing a relationship between a series of items, for the purpose of forecasting)

• **Time series** (like regression, but using the additional properties of timed information)

• **Clustering** (segmenting items that exhibit consistent behavior or characteristics into subsets or clusters)

• **Association analysis** (recognizing that the presence of one set of items implies the presence of another set)

• **Sequence discovery** (recognizing that of one set of items is followed by another set)

Classification, regression, and time series are primarily used for prediction, while clustering, association, and sequence discovery are more appropriate for describing relationships that exist in the data.

**Data mining techniques.** These are the basis of all data mining. Over the last thirty years, computers have inspired researchers to supplement the former manual methods of statistical analysis with many new computational techniques for discovering patterns in data. (See the sidebar for a list of popular data mining techniques used in today’s products, together with some examples of typical business applications to which they might be applied.)

**Algorithms.** Although each technique represents a specific type of analysis, products implement many varieties of the techniques. The design of the data mining algorithm(s) can make a significant difference to product performance and scalability. This is crucial when the product is required to mine a very large database; never assume linear scalability of any data mining tool as data volumes grow.

**Data mining products** range from stand-alone desktop-based tools to data mining servers implemented on high-performance client/server or parallel processing computers, exploiting the power of multiprocessors. Some products extract samples of information and mine the samples locally on the desktop while others mine the data directly in the warehouse databases.

**Framework Relationships**

Turning to the *relationships* among the elements of the framework, Figure 1 includes some data modeling notations indicating the cardinality (‘1’ or ‘m’) of each relationship. For those less familiar with data modeling notations, here is what Figure 1 says:

• A business problem may be studied using more than one class of model, and (of course) any modeling approach can be applied to many different business problems.

• More than one technique can be used for any class of model, and any given technique can be used for more than one class of model.

• There is usually more than one way of implementing any given technique.

• Data mining tools may support more than one of the techniques, and (of course) each technique is supported by more than one vendor’s product.
For any given technique (taking slight liberties with the modeling notation), a particular product supports a particular implementation algorithm.

**Evaluation Criteria for Data Mining Products**

To help in selecting data mining tools, we now propose a set of evaluation criteria. These are classified into nine major areas, as follows:

1. Product Architecture
2. Warehouse/OLAP Integration
3. Performance
4. Function
5. Presentation
6. Data Sources
7. Data Preparation
8. Environment
9. Administration

Each area is described briefly below, and represented by a set of evaluation criteria in Table 1, *A Product Evaluation Checklist*. Following this review of the nine areas, we discuss two important product requirements — *rule conversion* and *data visualization* — in a bit more depth.

**1. Product Architecture**

When evaluating any software product, we should understand its operational and connectivity requirements, how its data (and metadata) is stored, and whether run-time performance and scalability requirements have been taken into account in its design. Possible two-tier and three-tier data mining architectures are depicted in Figure 2.

**Sampling vs. Direct Access.** Tools may use sampling techniques to process a representative subset of data or process the data by accessing a warehouse directly. Sampling supports client-based analysis, while direct access is a server-based approach. The more data to be mined, the more we would tend to favor tools that perform their analysis in place. Direct access tools that read the data using the native SQL of the database server are likely to be the most scalable as data volumes rise. On the other hand, a sampling approach will usually offer more opportunity for data cleaning and other data preparation activities (see *Data Preparation* below). But if the data is already stored in a well-designed data warehouse, these activities may have already been carried out. We return to this subject in our discussion of planning considerations, in the last part of this article.

**2. Warehouse/OLAP Integration**

This criterion deals with how well the products integrate with other components of a data warehousing or data analysis architecture. Several forms of integration are possible:
Data Mining Techniques

The most commonly used techniques in data mining, together with examples of potential applications are:

Decision Trees

A decision tree is a tree-shaped structure that visually describes a set of rules (conditions) that caused a decision to be made, such as a decision to purchase a product. From decision trees we can generate rules for the automatic classification of a set of data, for example, to segment a customer database. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID).

Potential applications for decision trees include:

- Medical diagnosis: What are the factors that affect kidney transplant survival rates?
- Retail analysis: What are the categories of results from a customer survey?

Decision trees are often combined with neural networks to explain why a neural network has reached a particular conclusion. They can also be combined with rule induction (see below) to generate logic for repetitive decision making. For example, to help telesales staff confine their sales of mortgages to those customers and prospects who are identified as good risks because their profile matches one automatically mined from the data.

Genetic Algorithms

Genetic algorithms are optimization techniques that can be used to improve other data mining algorithms so that they derive the best model for a given set of data. The resulting model is then applied to the data to uncover hidden patterns or to make predictions. Genetic algorithms are best suited to segmentation/clustering applications but can be applied to most situations involving learning. Note the use of the word optimal in the following questions, which is an indicator that genetic algorithms may be the technique to use to answer these types of questions. Potential applications include:

- Direct mailing: What is the optimal profile of the ideal customer who is likely to invest more than £25,000 in mutual funds this year?
- Risk analysis: What is the optimal profile of a low risk home loan prospect who earns less than $75,000 and wants to live in San Jose?
- Medical diagnosis: What is the optimal treatment plan for a particular diagnosis?
- Retail analysis: What is the optimal store layout for a particular location?

Neural Networks

These are non-linear predictive models that learn how to detect a pattern to match a particular profile through a training process that involves iterative learning, using a set of data that describes what you want them to find. They are so named because, on a superficial level, their operation resembles that of the human brain. Potential applications include:

- Direct mailing: Who will respond to this mailing?
- Risk analysis: Which prospective customers are a good credit risk?
- Medical diagnosis: What disease is this person likely to contract?
- Retail analysis: What product is this customer likely to purchase?

Neural networks are appropriate for clustering, sequencing, and prediction problems. Their main drawback is their opaqueness — they do not explain why they have reached a particular conclusion. But in some applications (like matching handwriting samples) this is not important.

Predictive Modeling

A variety of techniques can be used to identify patterns, which can then be used to predict the future. These include classical linear regression and its derivative, logistic regression analysis, and further extensions such as Generalized Additive Models (GAM) and Multivariate Adaptive Regression Splines (MARS). These techniques are most often used to predict the odds of a particular outcome, based upon the observed data. For example:

- Direct mailing: Which geographic areas are most likely to respond to a new mailing initiative?
- Retail analysis: Which customers are most likely to be interested in a new product?

Rule Induction

Rule induction is the process of extracting useful If...then... rules from data, based on statistical significance. Potential applications include:
Published APIs open up the possibility of interoperation with many other tools.

Next, consider metadata integration. Where and how is a product’s metadata stored, and whether it can be launched from an information directory, for example.

Rule conversion is another useful feature. If a mining tool can convert the rules it has discovered about the business into SQL or 3GL code, this code can then be reused by other decision support tools or applications. See the later section on this topic.

Ideally, an interface with an OLAP tool would permit a two-way exchange of information, supporting an iterative style of data analysis. The OLAP tool could pass selected subsets of a data warehouse to the mining tool to be mined for patterns that are not easily exposed using analytical tools. Then, if the data mining tool can pass its findings as SQL queries directly to the OLAP tool, the OLAP tool can be used to verify the findings in a larger database.

Fuzzy Logic

Fuzzy logic handles imprecise concepts (like small, big, young, old, high, low) and is more flexible than other techniques. It provides the notion of a fuzzy set rather than clear demarcation boundaries; for example, rather than 0 or 1 there are also 0.9, 0.85, 0.93, 0.21, 0.05 etc.

A potential application might be direct mailing: Who might be a likely person to mail in our new campaign?

K-Nearest Neighbor (k-NN)

This technique places an object of interest into a class or group by examining its attributes and grouping it with others whose attributes are closest to it. k-NN is a classic technique for discovering associations and sequences when the data attributes are numeric. With non-numeric attributes or variables, it is a lot harder to apply this technique, because of the difficulty of defining a metric that can be used to quantify the distance between a pair of non-numeric values.

Data mining is inherently computationally expensive and can involve large amounts of data. To support interactive data mining activity, a tool must exploit performance enhancing technology. Does the tool exploit multiprocessors, for example, by running multiple instances of itself, or by handling the processing for a mining technique like a neural network in parallel (see Figure 3).

Next, consider the extent of the data mining function supported by a product — to what degree does it support the full range of potential data mining activities? Users generally need to have a range of techniques available to use in a data mining solution — how many different data mining approaches and techniques are supported in the product? Does it have a set of ‘canned’ rule models and applications that can aid productivity? Application development facilities and agent technology that enables event- or timer-driven mining are also worth looking for.

Clearly the more techniques supported by a product the more types of mining applications can be supported. Furthermore, support for multiple techniques allows users to look at the same problem from many different angles, and so improve the thoroughness of an investigation. If a pattern is not detected using one form of mining, another technique can be used to see if it uncovers anything. Equally, if different techniques automatically come to the same conclusion over pattern discoveries, then user confidence grows in the accuracy of the find.

5. Presentation

This evaluation criterion deals with the user interface. Especially important is the support for data visualization, by providing different ways to view the mined data. We discuss this area further in a later section — see Visualization is Key. Also, a tool’s usability will be influenced by its interfaces with workflow or groupware products, which can allow us to tie together the separate stages of a data mining process and disseminate the findings.

6. Data Sources

Earlier (under Architecture) we addressed the issue of whether the tool supports direct access to data or extracts a sample for mining. Here we
evaluate the level of support for various possible data sources. Of course, for any potential user of a data mining product, the specific data sources of interest will depend on the existing database environment. For some applications, it may also be important to be able to include external data sources in the mining process.

### 7. Data Preparation

A variety of data preparation activities are normally needed before applying the actual data mining techniques, especially when the source data comes from operational databases. A data mining tool may provide support for:

- **Data cleansing**, such as completing missing data fields, identifying inconsistent data in different data sources, or resolving referential integrity violations.

- **Data description**, supplying metadata such as row or value counts or distributions.

- **Data transformation**, adding new derived values, combining continuous attribute values into ranges, or replacing categorical attributes by a series of binary (yes/no) attributes. The need for such transformations will depend upon the tool and the technique(s) to be applied.

- **Data sampling**, as required for training or model building.

- **Data pruning**, identifying dependent, independent, and correlated columns or variables.

Typically, data preparation is the most time-consuming aspect of data mining. Also, the preparation effort increases with the number of data sources to be consolidated. Anything that a tool can do to support this process will speed up the overall process of developing a data mining model.

<table>
<thead>
<tr>
<th>Product evaluation criteria</th>
<th>Features to consider</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Product Architecture</strong></td>
<td>2-tier or 3-tier?</td>
</tr>
<tr>
<td>Client/server architecture</td>
<td>Flat file or relational database?</td>
</tr>
<tr>
<td>Data storage</td>
<td>Extract sampling, direct access to database, or both?</td>
</tr>
<tr>
<td>Data manipulation</td>
<td>ODBC, DBMS gateway, or ORB?</td>
</tr>
<tr>
<td>Connectivity to data</td>
<td>Server-resident mining objects supported?</td>
</tr>
</tbody>
</table>

| **2. Warehouse/OLAP Integration** | |
| Interoperability             | Published APIs exist? |
| Metadata                     | Document mining objects in an information directory? |
| Rule conversion              | Launch mining objects from a directory? |
| Connectivity to other tools  | Convert mined rules to SQL or 3GL code? |
|                             | Pass rules directly to OLAP tools? |
|                             | Receive data for mining from OLAP tools? |

| **3. Performance** |  |
| Scalability         | Support for multiple user access (mining server)? |
| Parallelism         | Multi-threaded mining server? |
|                     | Support for mining very large databases? |
|                     | Architecture exploits multiprocessors, parallel RDBMSs? |
|                     | Algorithms designed for parallel operation? |

| **4. Function** |  |
| Mining approaches | Support for major prediction and description approaches? |
| Mining techniques | How many data mining techniques are supported? |
| Canned applications | For example, pre-built Risk Management applications |
| Canned rule models | For example, Credit Model, Debit Model, Fraud Detection Model, Credit Risk Model, Profitability Model, Attrition Model |
| Application development | Application development capability? |
|                     | Support for timer-based or event-based mining triggers? |

| **5. Presentation** |  |
| Data visualization | Support for graphs, maps, tables, rotation, etc? |
| Workflow interface | Support for a mining process (sample, report, etc.)? |
| Groupware interface | Support for disseminating output (e-mail, publish/subscribe)? |
| GUI                | Toolbar customization etc.? |

| **6. Data Sources** |  |
| Warehouse data      | Direct access to data warehouse? |
| Operational data    | Access to data in operational databases? |
| External data       | Access to external data (demographics, etc.)? |
| Multi-database      | Multiple data sources supported directly, or via a gateway? |

| **7. Data Preparation** |  |
| Data cleansing       | Identify missing, inconsistent or incorrect data? |
| Data description     | Generates metadata from source data? |
| Data transformation  | Performs data conversions, transformations? |
| Data sampling        | Assists with sampling process? |
| Data pruning         | Assists with pruning and selection of independent variables? |

| **8. Environment** |  |
| Platforms           | Client platforms supported? |
|                     | Server platforms supported? |
| Size constraints     | Maximum sample size? |
|                     | Maximum number of rows or records? |
|                     | Maximum number of fields? |
|                     | Maximum number of values for a discrete field? |
|                     | Maximum range for a numeric field? |

| **9. Administration** |  |
| Central administration | Support for versioning of data mining objects? |
| Security              | Usage monitoring and reporting? |
|                       | User access authorization? |
|                       | Private, group, and public mining objects? |

Table 1. A product evaluation checklist
Rule Conversion

Support for rule conversion is an important requirement for the integration of data mining tools with OLAP and other data analysis applications. The ability to convert mined rules into SQL queries and 3GL code supports two requirements of information processing:

- **Data segmentation.** Automatically converting the rules that define mined patterns into SQL or 3GL allows them to be used immediately—for example, to segment a customer base for customized marketing.
- **Rule reuse.** Mined rules, when expressed in the form of SQL queries, become available for reuse by other OLAP applications. Rules converted to 3GL can be stored (for example, on a rules server) as standard procedures or objects, and reused by application development tools.

This is an area in which OLAP and data mining products often work well together, the mining tool generating the SQL from the rules discovered and passing the SQL to an OLAP product for execution. Alternatively, converting mined rules into 3GL code allows these same rules to be included in applications. This capability can then permit some systems to be brought forward to the front office.

For example, suppose that the front-end of a mortgage issuing application—normally simply a data gathering stage—could be extended to check the customer input against previously mined rules that qualify customers as a good risk. Telephone operators and counter clerks could then be empowered to authorize certain safe loans without referring the application forms to back office staff. This example illustrates how data mining can lead to a variety of benefits, including:

- Increased expertise embodied in application software
- Correct reuse of previously established business rules
- More prompt customer service
- Risk management, rather than risk avoidance.

Visualization Is Key

No matter which type of data mining application is planned, or development tools used, a rich set of data visualization capabilities are an important prerequisite. Mining is a combination of two concepts:

- Automatic pattern discovery
- Pattern visualization.

No matter how effective the data mining tool may be discovering patterns, those patterns are viewed, evaluated, and acted upon through the tool's user interface. The tool user has the power to abandon a line of inquiry, to regenerate the inquiry with adjusted parameters, or to make a strategic management decision based on the information presented. Clearly, the influence of the visualization component of the data mining tool is paramount.

The success of the visualization activity relies heavily on the human user's ability to access and comprehend the results of the analysis. Therefore:

- There must be a natural way for users to visualize the results of the data mining activity.
- Visualization must be intuitive and integrate seamlessly into the user's current environment.

Even the best set of rules or the most complete table of data may reveal more information when visualized with color, shape, position, size, orientation, relief, or texture. To illustrate this, an example of segmentation is shown in Figure 4. The figure shows demographic data reporting the levels of response to a marketing promotion, for states and local areas. The height of each state indicates its response rating and embedded columns identify specific localities. [Editor's note: Unfortunately, black and white reproduction reduces the effectiveness of the original colored image.]

Planning the Introduction of Data Mining

Our nine-part checklist proposed a set of evaluation criteria for data mining products. In an ideal world one data mining tool would satisfy all these requirements. However life is not like that — at least, not...
An Implementation Framework

Data mining, like any other technology, should be introduced in an architected manner to ensure true integration. A defined technical architecture is a blueprint for a true end-to-end information delivery solution that includes data mining technology. Figure 5 shows the Data Warehouse Technical Architecture.

The Data Access component of the above technical architecture comprises several related technologies that address data access and analysis:

- Query, reporting and analysis tools
- Multi-dimensional DBMS OLAP clients
- Relational OLAP clients
- DSS application development tools

Data mining technology is the most recent addition to this component, although the overall level of function, and the degree of integration among the current tools in this marketplace is still somewhat limited, as we discuss below.

Early Product Limitations

It is, of course, a fact of business that the maturity and scope of any software product — in both its current and its future releases — depends on market forces, and on the scale of the vendor’s operations and the range of environments the vendor can support.

A Brief Look at the Marketplace

Many of the products in the marketplace are based on a particular main technique. Some support multiple techniques. Below is a list of techniques and some of the products in the marketplace that use them.

Decision Tree

Some tools that use an easy-to-understand graphical categorization of knowledge (decision-making logic) are:

- KnowledgeSeeker from Angoss Software
- XpertRule Profiler from Attar Software
- Business Miner from BusinessObjects
- Clementine from Integral Solutions Ltd.
- Quadstone from DecisionHouse (TreeHouse)

Neural Network

Some tools that support parallel interconnecting computing units learning to do tasks by a training process on available data are:

- Marksman from HNC Software
- IBM’s Intelligent Miner
- RedBrick’s Data Mine
- Clementine from Integral Solutions Ltd.

Unfortunately, many of today’s data mining tools tend to be available only for a limited number of database environments and platforms. In practice, tools offering limited platform support are likely to be used only for specific, focused applications. While data mining is undoubtedly important to the enterprise, top management are unlikely to change their underlying database and operating systems infrastructure to deploy it.

For these reasons, it is likely that many organizations will need to deploy several different mining tools to satisfy the full range of possible data mining applications, and there will
be a strong interest in web-enabled mining servers.

Integration is another weak area. Data mining technology is currently represented by a plethora of data mining tools, many of which are somewhat 'stand-alone' in their first releases:

- Specialist 'freestanding' tools from small-scale independent vendors (such as Clementine from Integral Solutions, Decisionhouse from Quadstone, and XpertRule from Attar Software).
- Specialist 'freestanding' tools from large multi-functional software vendors (such as Intelligent Miner from IBM, and Mineset from Silicon Graphics).
- 'Integrated' tools from decision support vendors (such as Discovery Server from Pilot Software, and SAS System from SAS Institute).

In the long term, as we move into the world of distributed software components, lack of interoperability is a severe limitation. But one short-term advantage of freestanding tools over integrated tool suites is that companies can get started with data mining without having to invest in a complete technology or pay the price of a complete suite. On the other hand, an integrated suite favors developers already using that suite for other development work. They can probably adopt the data mining components quickly, while continuing to use other familiar components for related requirements such as data visualization.

While some tool integration is starting to appear (for example, Red Brick has introduced Red Brick Data Mine, which combines its star schema data warehouse with data mining software from DataMind Corporation), we have yet to see a widespread merger of data mining and decision-support software. Currently, server-based mining tools with APIs (like IBM's Intelligent Miner, for example) hold the most promise in this area. Mergers, acquisitions, and marketing agreements will inevitably lead to further integration.

**Data Quality and Warehouse Access**

One of the biggest problems in data mining is the quality of the data. Unlike ordinary SQL queries, data mining techniques are highly sensitive to missing or inconsistent data. This is particularly true for techniques that use sampling, which is a problem when starting with operational databases; mining samples are likely to contain missing or inaccurate data. One advantage of mining a data warehouse is that the tools are working with data of a more uniform and predictable quality, because the data has already been 'cleansed' as it was taken from operational systems.
As we pointed out in the evaluation framework (under Warehouse/OLAP Integration), to make the best use of data mining techniques, tools need to integrate with a data warehouse (to ensure quality of data within a consolidated database) as well as with other interactive business analysis tools (to ensure visual comprehension of data mining results).

However, the limitations of many of today’s data mining tools noted above — predominantly “stand-alone” tools, supporting a narrow range of platforms — limits their ability to interact with already established data warehouses. This in turn mandates a mining approach involving extraction and importing of data onto a different platform for analysis.

Deciding What Data to Mine

The myth that mining “cannot be done at the atomic level” is not an argument for data sampling. Mining detailed atomic-level transaction data may take longer, but it usually means that the analysis is more exhaustive, leading to more accurate knowledge discovery. When atomic level data is stored in an enterprise data warehouse, with summaries and some detail data being available at the data mart level, it is often the case that data mining is best deployed against the warehouse.

On the other hand, there is a trend towards data mart development rather than enterprise data warehouse development, because of the shorter time scales involved and the faster return on investment. It might seem, therefore, that data mining projects would have to wait for the enterprise level warehouses to be deployed. Actually, this is not the case. Data mining can and does work well on summary data and/or atomic transactional data in a data mart. However, because of the possibility that data summarization has obscured key aspects of a pattern, interoperability with OLAP tools becomes very important. Interoperability allows an analyst to develop an iterative approach to data mining and pattern verification, as we discussed earlier under Warehouse/OLAP Integration.

Future Directions

While data mining technology is still in its infancy, there is no doubt where it is headed. The direction is clearly towards mining servers, mining agents, and mining objects. The industry trend toward thin client architectures will pressure mining vendors to make their products available from web browsers and from groupware front-ends like Lotus Notes.

The actual data mining algorithms, on the other hand, will move as close to the data as possible. Given the trend toward universal (or object-relational) DBMS servers (such as Informix’s Universal Server, and IBM’s DB2 Universal Database), it would be logical to expect that mining algorithms will eventually become an integral part of the DBMS itself, supported by query optimization technology, and exploiting the DBMS’s support for parallelism.

That development will not materialize overnight. In the meantime, there will be fierce competition among the vendors of existing data mining tools, and an inevitable shakeout. Greater tool integration is inevitable, either through acquisition (vendor takeovers), consortia (marketing agreements) or design (integrated design or development agreements). Multiprocessor support, and the evolution of key features including mining servers, multi-threading, scalability, published APIs, and web visualization are all going to become critical over the next two years. So before you dig into your gold mine of data looking to strike it rich, evaluate the data mining and DBMS market carefully.

References