Running Head: TECHNOLOGIES FOR THE MASSIVE DATA PROBLEM

MetaTech Consulting, Inc.

White Paper

A Characterization of the Massive Data Problem and of Useful Technologies

Jim Thomas

November 26, 2003

Abstract

This paper presents the concepts of the *massive data problem* together with insight to germane technologies. A characterization of the problem is presented prior to describing the most relevant technologies with deference to their maturity. The *data grid*, embodying an appropriate and complete set of features, is revealed as the objective solution to the problem. Additional work to prove or disprove this finding are recommended to organizations with proficiency in the filed of applied technology.

Executive Summary

The purpose of this paper is to provide a brief treatment of the technologies germane to the massive data problem. It is intended to inform the reader of the size of the datasets that the engineering and scientific communities are now beginning to work with the challenges it presents. One source of massive data of note is that which is produced by the space-borne sensors. These, together with other sensors, are discussed to provide a context for the reader.

This paper is founded on research of materials obtained from the public domain (technology forums, periodicals, etc.) as well as materials from professional associations available through membership or via educational sponsorship. References were restricted based on their relevance to the topic, how current the work which they documented, their notability of the works authors, and a subjective assessment of the rigor they presented.

It was found that no single technology is likely going to solve the massive data problem. Rather, a variety of technologies brought together in new and innovative ways is the surest approach. Furthermore, while some of the relevant technologies are mature and stable in today's environment, others are still yet maturing and others are only now emerging. Distributed computing, service oriented architectures, and data grids are featured prominently as key technologies.

It was challenging to adequately address the total of massive data problem in such a concise treatment. For executives and manages, this paper provides an ample overview of the massive data problem. For technicians or engineers preparing to undertake the challenge of addressing the massive data problem, this paper serves only as a primer and as a point of embarkation.

Table of Contents

A Characterization of the Massive Data Problem and of Useful Technologies

The scientific and engineering communities exercise applications that execute on sets of data for the purpose of discerning answers to our most challenging problems. The datasets are frequently formed over extended periods of time (e.g. historical records), from sensors that respond with incredible frequency (e.g. real-time flight instrumentation or integrated sensor networks), from sensors that have tremendously broad bandwidth (e.g. spacecraft), or a combination of all three. As the magnitude of the problem has shifted *megabytes* to *petabytes*, it has been termed *massive data*. Further technological maturity is critical to our efforts at exploiting, to the greatest extent possible, the information, knowledge, and intelligence hidden within these massive data sets.

The purpose of this paper is to provide a brief treatment of the technologies germane to the massive data problem. It begins with an introduction to massive data for the purpose of providing the reader with an understanding of the scope of the problem at hand. The following section provides a discussion of the technologies that are useful in mitigating the massive data problem. These technologies are apportioned into three meaningful categories being a) *mature and stable*, b) *maturing*, and c) *emerging*. Within each category, a brief set of relevant technologies is discussed in turn. The discussion of mature technologies will touch on database management systems, distributed computing architectures, peer-to-peer architectures, and data warehousing solutions. The section on maturing technologies addresses metadata solutions, data mining solutions, and service oriented architectures. The emerging technologies covered focus on computational grids and data grids. Two appendixes provide detailed treatments of technology that were omitted from the principal body of this paper.

The Massive Data Problem

Data, information, and knowledge are often critical to successful business operations. To varying degrees, and at varying stages in their processes, businesses both produce and consume these artifacts. Information technology systems of a business process both constrain and define its capacity to exploit these artifacts over time. Often, even with state of the art technology and expansive budgets, businesses find themselves unable to effectively store, manage, and process all available data – together with information and knowledge – and either actively or passively they must allow the excess to perish.

Each instantiation within each business domain has differing capacity to deal with data which ranges from *gigabytes* (10^9 bytes of data) to *terabytes* (10^{12} bytes of data) to *petabytes* (10^{15} bytes of data). It is this largest scale of data problem that is generally acknowledged as *massive data*.

Massive data is produced from three classes of sources: a) deep archives, b) wideband data collectors such as those employed in space borne sensors, and c) integrated networks of disparate sensors. Each of these is addressed in turn though, with consideration to required brevity, a thorough treatment will be omitted. Interested readers are referred to the cited references for a more complete appreciation of the material.

Deep Archives

The data management challenges of the telecommunications industry are characterized by the vast number of discreet call records that must be maintained for years. Though each individual record is relatively small in size, the number of calls placed is tremendous and their aggregate growth over time approaches the petabyte-scale.

A recent addition to this class of massive data problems is that of network trace data. Auditing and logging of transactions occurring on the network, coupled with traffic and utilization information collected as a function of the network management function, is astounding. Garofalakis and Rastogi (2001), citing work by Fraleigh, Moon, Doit, Lyles, and Tobagi (2000), report that each day the Sprint IP backbone generates 600 gigabytes of packet trace data for traffic management. Though not all of this data must be persisted for extended timeframes, that which is necessary for security accounting must be.

Space Borne Sensors

The capacity (e.g. bandwidth) of space borne collectors is immense. The Earth Observation Satellites are purported to generate 918 gigabyte of data for ingest into the Information Power Grid that is now being developed for the National Aeronautical and Space Administration (Nicholls, 2001). Gad and Calton (1999) assert that satellites account for the greatest quantities of data being produced in support of atmospheric studies. For example, a space borne synthetic aperture radar produces a file as large as 121 megabyte for each image covering a 2500 square-meter area. As it is necessary to capture much larger areas, and images each area must be refreshed repeatedly, the aggregate data requirements grow massive rapidly.

Integrated Sensor Networks

Seldom can a single sensor provide a comprehensive set of data capable of satisfying all needs. It is common to aggregate data produced from multiple sensors to produce a more complete model of the target environment. Such is the case with the Virtual National Air Space Simulation Environment depicted in Figure 1. This project will integrate data from existing performance models, environmental data, and live data feeds from the aircraft (e.g. throttle

settings, elevation data, etc.) to synthesize a complete model of the operational airspace above the Nation (Johnston, 2002). Though the demands of each sensor, taken individually, are not remarkable, aggregating together sensor data from the nearly 22,000 flights a day (each with its own sensors) will result in massive data.

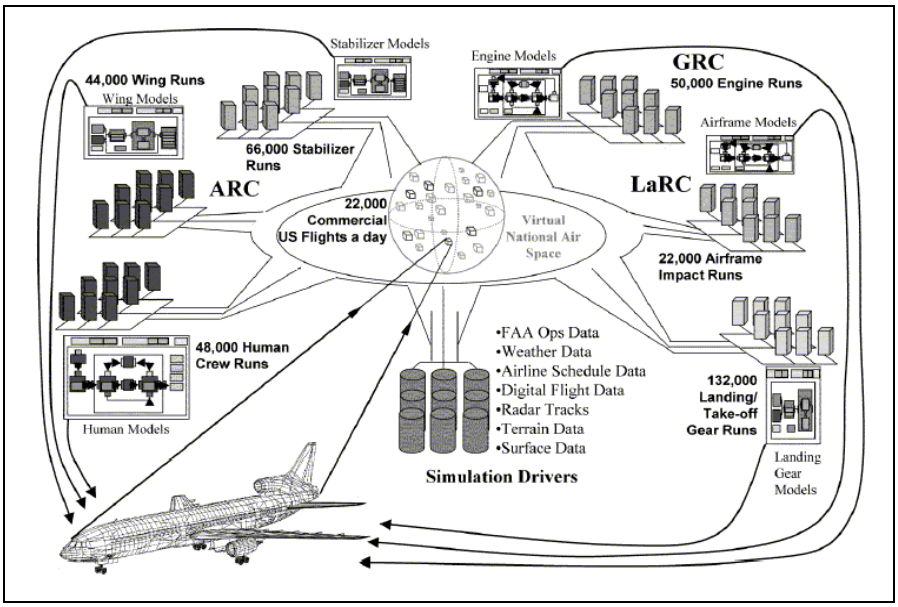

Figure 1. Example of a multi-sensor environment (Johnston, 2002, p. 1088)

Useful Solutions, Capabilities, and Technologies

The scientific, engineering, and business communities of today are hindered in their ability to fully exploit the data captured by the sensors and collection systems (i.e. network status reporting applications) of today. Still yet, there is a drive to deploy even more capable sensors – to generate a far greater quantity of data having significantly higher fidelity (e.g. sample rate, bandwidth, etc.). In the absence of a proportional increase in the capabilities of the information management and processing systems, any increase in the data production capabilities will result in a more diminished ability to exploit that new data.

Researchers in both the public and private sectors have been working to produce viable solutions, capabilities, and technologies that will increase our ability to cultivate data – to

consider data in a more complete context thereby gleaning more information and knowledge from it. The following paragraphs in this section address a subset of the technologies that are believed to have significance to the massive data problem. The technologies addressed are binned to emphasize their respective level of maturity: a) mature and stable, b) maturing, and c) emerging. The listings of technologies within each bin are neither comprehensive nor concrete – the assignment process was purely subjective and represents this authors attempt to partition the products in a meaningful fashion to support the continued discussion on massive data.

Mature and Stable

The technologies identified as mature or stable are have been available commercially for many years. There are formal standards or, at a minimum, accepted practices that are generally accepted by the user communities. This is not to imply that each enjoys a static and singular set of interface specifications or protocols. Vendors continue to strive for market share by providing value-added capabilities that often necessitate deviation from standards. As these technologies are so prevalent in the information management solutions of today, interested readers are encouraged to reference vendor's product literature and the abundance of information available in press only. The technologies identified as mature and stable include a) database management systems, b) distributed computing architectures, c) peer-to-peer architectures, and d) data warehousing solutions.

Maturing

Technologies of this class are less well defined than those previous mentioned. While there are a number of solutions that might be commercially available, there remains only rudimentary progress on resolving on appropriate standards. Those interested in a

comprehensive treatment of these technologies will likely find themselves somewhat frustrated over the lack of agreement on the topic by the most prolific authors on the technologies. The technologies identified as *maturing* are a) metadata solutions, b) data mining solutions, and c) service oriented architectures. A brief discussion of each is provided.

Metadata solutions. *Metadata* – data about data – has existed nearly from the moment that data was first recorded though systemic solutions that employ metadata to solve data management problems are not widely employed (Jeffery, 1998). Quite possibly the widest known successful example of a metadata solution is that of the card catalog systems used by the libraries. Each card within the system contains a reduced set of data about the target artifact.

As automated information systems have taken on the demands of greater quantities of data, there has not been a proportional increase in the sophistication of the systems to manage that data. For even large quantities of data, it has been possible to manage data directly. However, as the quantities of data managed within integrated information systems increase rapidly, the limits of our ability to manage it are beginning to become evident. New strategies with more capability are needed to more completely manage the massive datasets and to more fully harvest the information and knowledge from it.

Jeffery (1998) suggests that a comprehensive metadata solution should address three different types of metadata: a) *schema metadata* that is concerned with the implementation details of the host database or other persistence construct, b) *navigational metadata* that provides information regarding the location and retrieval particulars of data such as the file location or database address, and c) *associative metadata* that provides additional information such as context, dictionaries, security, and content rating. Other authors on the subject have selected other meaningful strategies to partitioning metadata.

Pfister (2002) has described the use of a *metadata clearinghouse* to provide a means of providing user-driven access to the massive data of the Earth Observation Data Distribution System (EODIS) system. The approach allows the user to navigate through the data via metadata based queries similar to those prototyped at the University of Maryland Human-Computer Interaction Laboratory. As standards are more formalized, such approaches are likely to emerge as viable strategies for the management of massive data.

Data mining solutions. The term *data mining* has practical meaning to the generalist and specialist alike. To the generalist, the term refers to any number of techniques by which he is able to examine data – generally located in some form of database(s) – to gain awareness of hidden facts, obscure details, or possibly previously unknown relationships existing in varied data sets. To some, a series of searches on the internet satisfies their definition of the term. To the specialist, data mining connotes a far more discrete set of activities including sophisticated algorithms executed against data sets persisted in specialized data structures prepared for the specific activity. The objectives of the generalist and specialist are quite similar though the processes and technologies employed differ dramatically.

Database management system providers have suggested that data mining can be accomplished effectively on the source data systems without the need to build dedicated systems and without needing conformed copies of data. Figure 2 represents an illustration of the relative maturity of DBMS-enabled data mining as a technology (Strange & Friedman, 2003). That work suggests this approach to data mining is from two to five years until it reaches productivity. This author remains confident that it is unlikely that DBMS-enabled data mining will be effective in complex data environments requiring extensive data transformation and translation. It is possible that approach will be successful in mining cogent datasets that already exist in single

databases or are easily conformed. A more complete treatment of data mining is provided in Appendix A.

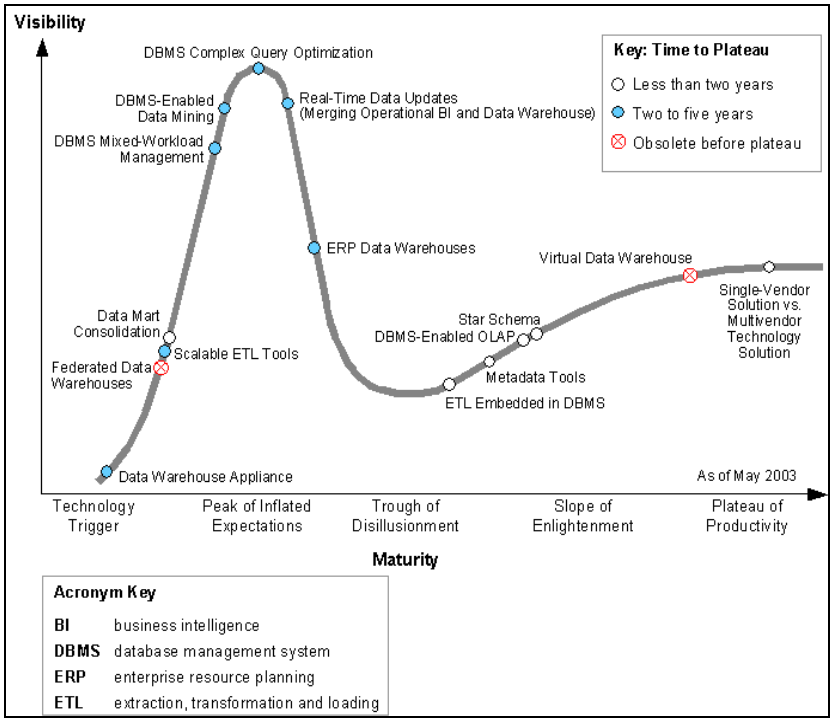

Figure 2. Hype cycle for data warehousing .

Service oriented architectures. A functional definition of a *service oriented architecture* (SOA) is provided by He (2003, p. 2) when he states that a "SOA is an architectural style whose goal is to achieve loose coupling among interacting software agents." The SOA ensures a separation of concerns between the services and allows for classes of services with each member of a class having a different implementation. Different implementations are designed to have desirable characteristics unique to the users needs such as quality of service.

Two identified architectural constraints of this architectural construct are that each service must a) have only a small number of simple, ubiquitous interfaces and b) support the passing of descriptive messages based on a limited vocabulary. He (2003) identifies four characteristics that an architecture must exhibit to be appropriately classified as a SOA. Those are:

1. Messages passed to and from the services must be *descriptive* and not *instructive.* That is to say that the messages must provide details on what the service is to do and not how to do it.
2. Messages to and from a service must adhere to a generally understood format, structure, and vocabulary.
3. The messages definitions (e.g. format, structure, and vocabulary) must be extensible to accommodate releases of later versions of the services.
4. The architecture must provide a mechanism for the services to be discovered within an appropriate context.

In addition to the above enumerated characteristics, services within a SOA may be *stateless* or *stateful*. That is to say that the services may or may not persist state information from one use to the next. Also, services must handle duplicate requests in a defined manner. Services may either treat duplicate messages as unique instructions and execute them accordingly, or it may recognize the duplicates and discard them.

Realizations of the SOA construct are becoming as *web services*. Though there remains disagreement on the formalization of the SOA in general and of web services in particular, it is generally agreed that web services must a) have interfaces based on Internet protocols (e.g. HTTP, FTP, SMTP, etc.) and b) utilize XML to pass all messages. An illustration of a web services model of the SOA construct, as interpreted by Kleijnen and Raju (2003), is provided in figure 3. This model emphasizes conformance to open standards to enable the broadest possible use.

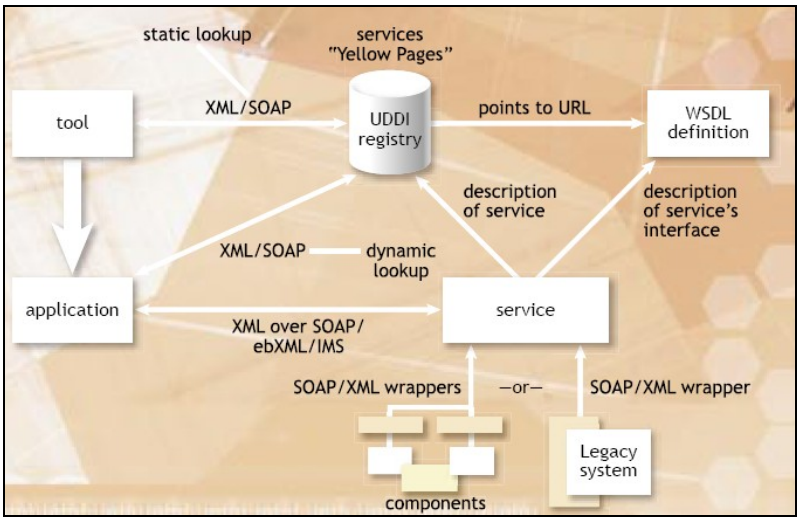

Figure 3. Service Oriented Architecture (Derived from Kleijnen & Raju, 2003, p. 42).

Emerging

Emerging technologies are those that have few implementations outside of the research community. Little information is found on these technologies in the press other those dedicated to the interests of researchers or of academicians. The technologies germane to the massive data that can be identified as *emerging* are a) computational grids and b) data grids. A brief discussion of each is provided.

Computational grids. *Computational grid* was first used as a term by Foster and Kesselman (1998) where they defined it as the "hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities" (p. 3). Their notion of the computational grid as an infrastructure for computing was founded on an analogy of the power grid that had provided ready access to electricity in the early years for the twentieth century. Since publishing that early influential work, the authors, together with others in the research community, have furthered the notions of the computational grid through a number of subsequent papers.

Architectures for computational grids have been envisioned and have been realized through large scale functional research demonstrations. This has led to greater understanding of the viability of the concepts and has establish a new computing paradigm – *grid computing* – as being sufficiently distinct from traditional computing and having promise for practical application in the near future. A treatment of computational grids is provided in Appendix B.

Data grids. Though computational grids address the requirements of complex processing across a distributed environment with loosely managed resources, it does not expressly address the demands of massive data. To say that another way, the principles of grid computing are necessary but not sufficient to satisfy the massive data problems.

Chervenak, Foster, Kesselman, Salisbury, and Tuecke (2001) outlined the need for the data grid in asserting that existing data management infrastructures are inadequate for the demands of certain analytical domains. Specifically cited in that work are efforts in global climate change, high energy physics, and computational genomics which each having data volumes measured in terabytes and are soon expected to increase to the petabyte range.

The data grid, extending the notions of the computational grid, is based on a service oriented architecture. The distinguishing services of the data grid are *storage system services* and the *metadata services* (figure 4). The storage system services provide a layer of abstraction to hide the implementation details of the physical storage constructs used which may include flat files, databases, or any other for of persistence. The abstraction will facilitate persistence of data and access to it through a consistent set of interfaces. The metadata services will provide a robust and fault tolerant infrastructure in a hierarchical and distributed structure similar to that provided by the *lightweight directory access protocol* (LDAP). This will provide data locating and access across the data grid and in support of computational grid functions.

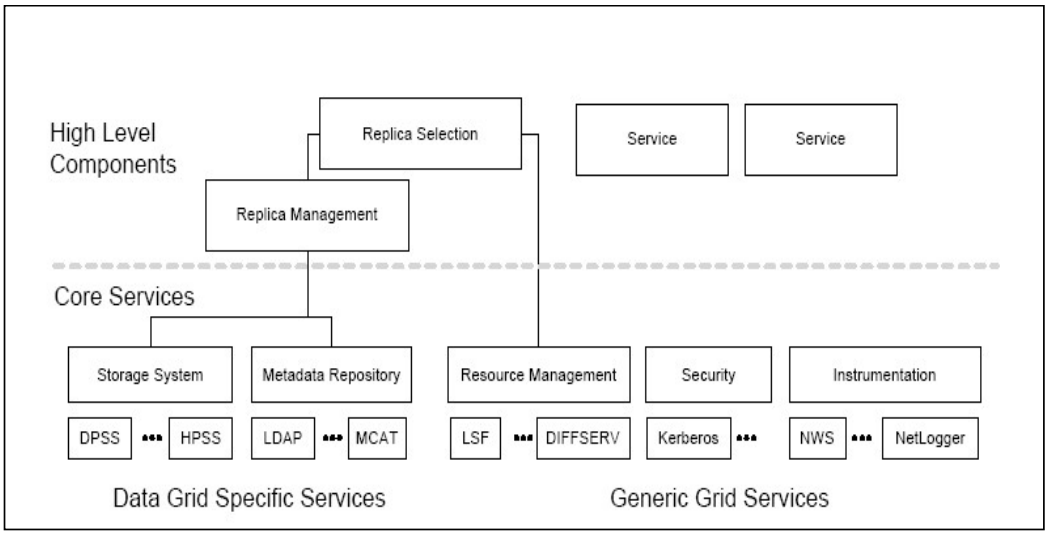

Figure 4. Major components and structure of the data grid architecture.

There are four characteristics that are considered necessary that support the geographically disperse, heterogeneous, multi-institutional environment that distinguishes the data grid.

1. *Mechanism neutrality*. The data grid services must provide sufficient abstraction from the underlying physical resources that persist and access the data.

2. *Policy neutrality*. The underlying resources must support access and other performance policies as necessary by the higher-level services.

3. *Capability with computational grid resources*. The data grid is envisioned to coexist with the computational grid. Physical resources are expected to support computational processes together with data process as appropriate. The parallel constructs must support common interfaces to interoperate.

4. *Uniformity of information infrastructure*. Uniform and convenient access to information about resource structure and state will facilitate adaptation of the system to runtime constraints. A common data model and interface to access the metadata, replica,

and instance catalogs should be used form the data grid and the underlying information infrastructure.

Research activities on the data grid have drawn much attention in the past couple of years. In 2002 the National Science Foundation sponsored the creation of the International Virtual Data Grid (Jay, 2002) and NASA continues to pursue the utility of the model as a solution to the massive data problem (Johnston, 2002).

Conclusions

As technology that supports distributed computing continues to mature, and as computational power increases, the scientific and engineering communities continue to press the envelope of what is possible. Virtual organizations comprised of members from geographically dispersed facilities reach for datasets that have a depth and breadth that seems limitless – the massive data problem. Such is the magnitude of the massive data problem that the storage, management, and access strategies of today are incapable of satisfying the needs of tomorrow.

Through leveraging the best of the capabilities that are already mature and fostering development on less mature technologies, the research community is nearing a breakthrough that will likely satisfy the need for data of the engineering and scientific communities for years to come. Central to overcoming the massive data problem is the further development of computation grids and data grids. Absent these capabilities, few other technologies hold promise to satisfy the seemingly insatiable appetite for data that has become characteristic of the scientific and engineering applications and those who exercise them.

References

Chervenak, A., Foster, I., Kesselman, C., Salisbury, C., & Tuecke, S. (2001). *The Data Grid: Towards an architecture for the distributed management and analysis of large scientific datasets*. Retrieved on November 20, 2003 from http://www.globus.org/documentation/incoming/JNCApaper.pdf.

Foster, I. & Kesselman, C. (Eds.) (1998). *The grid: Blueprint for a new computing infrastructure*. San Francisco, CA: Morgan Kaufmann Publishing.

Gad, L. & Calton, P. (September, 1999). Statistical, physical, and computational aspects of massive data analysis and assimilation in atoms. *Journal of computational and graphical statistics*, 8 (3), 16 – 19.

Fraleigh, C., Moon, S., Doit, C., Lyles, B., & Tobagi, F. (October, 2000). *Architecture of a passive monitoring system for backbone IP networks*. Technical report TR00-ATL-101-801, Sprint Advanced Technologies Laboratorie.

Garofalakis, M. & Rastogi, R. (May, 2001). Data mining meets network management: The NEMESIS project. *ACM SIGMOD International Workshop on Research Issues in Data Mining and Knowledge Discovery*.

He, H. (September, 2003). *What is service-oriented architecture?* Retrieved on November 22, 2003 from http://webservices.xml.com/lpt/a/ws/2003/09/30/soa.html.

Jeffery, K. G. (October, 1998). *Metadata: An overview and some issues.* Retrieved on November 23, 2003 from http://citeseer.nj.nec.com/jeffery98metadata.html.

Johnston, W. E. (October, 2002). Computational and data grids in large-scale science and engineering. *Future generation computer systems*, 18 (8), 1085 – 2001.

Kleinjnen, S. & Raju, S. (March, 2002). An open web services architecture. *ACM Queue*, 1 (1), 38 – 46.

Last, J. (2002). NFS funds global data grid lab. *Research technology management*, 45 (1), 6 – 7.

Nicholls, B. (December, 2001). *Meta clusters in the wild.* Byte.com, December 10 2001.

Pfister, R. (2002). *New paradigm for search and order in EODIS*. Retrieved on November 23, 2003 from http://citeseer.nj.nec.com/416003.html.

Strange, K. H. & Friedman, T. (2003, May). Hype cycle for data warehousing, 2003. *Gartner Strategic Analysis Report*, R-19-9970.

Appendix A. Treatment of Data Mining

Implementation Approach

Mining is performed as a project with specific objectives and requirements. The likelihood of success of a mining project, as with any project, is directly related to the rigor of the process employed. Edelstein (2001) suggests seven steps common to successful data mining projects. Brief summarizations of those steps are presented in the following paragraphs.

Define the business problem. In order to succeed, the project participants must understand their objective as well as the criteria that will be used to measure their accomplishments. Generally, it is an objective of any data mining activity to discover previously unidentified patterns in some set of data. Beyond that understanding, the project must have a clear understanding of what type of patterns are to be searched for as well as the data – in databases, repositories, or other systems – that they will have access to. With this information, the project team can begin to link together available data to form a logical model or web that provides the breadth of understanding to satisfy the objective.

Build data-mining database. The project team, having composed a logical model that links together data fields from possibly many different data sources, will work to define a physical data model that can be built to house the data. Data mining relies on statistical analytic process algorithms that access tremendous quantities of data. While relational data models have proven effective for Boolean queries of simple and moderate complexity, they are largely ineffective for data mining. Other structures, such as *Star Schemas*, *Snowflake Schemas*, and *Dimensional Schemas*, provide a more appropriate structure for the mining problems. It is beyond the scope of this paper to detail these data structures. The author recommends those

interested in greater detail on the matter to published works on star schemas and snowflakes (Kimball et al., 1998). Sweiger (2002) describes the use of dimensional schemas appropriate for more specialized forms of data mining to include *clickstream analysis.* These mining structures have limited application is typical transaction processing environments where relational models are so effective based on their highly de-normalized nature. *Normalization,* the design characteristic of a database to have only a single occurrence of data, is a desirable quality in a relational database that results in efficiencies in storage quantity. Data mining activities are less concerned with minimizing the storage capacity and more concerned with interrelationships of discrete datasets.

Explore data. Through tools and manual manipulation, the mining team and data analysts go though a process of understanding the source data in its native form. Identification of unsatisfied data needs will drive to changes in the project scope or the access limits to available data. In some instances, it may be necessary to collect new data to build bridges between existing data.

Prepare data for modeling. After defining the physical schema that will accommodate data from identified repositories as well as satisfy the business problem at hand, the mining team must then capture the process for ingesting the source data. While some data can be brought in as a simple copy, other data must be translated from one form to another or transformed in some other manner. For instance, data from one source may have names of states captured in the long name (e.g. Maryland), others may contain only the two letter digraph (e.g. MD), and still others may provide a zip code (e.g. 20640). To accomplish effective mining, all data must be represented consistently. This transformation is often the most complex of the mining activity.

Build model. Once the mining database is fully populated with data, data analysts and engineers work together to configure and optimize tools to discern patterns in the data. This is the true mining activity. The algorithms applied use a variety of approaches to discern patterns to include, among others, decision trees, linear regression, neural networks, Bayesian or sequential logic. A treatment of the algorithm strategies is beyond the scope of this paper though detailed works are widely available for consideration.

Building the model is an iterative process that may necessitate changes to the underlying model and changes to the datasets included. The result of the activity is a resulting dataset that represents a model of the source dataset.

Evaluate model. The model resulting from the application of the data mining tools, with optimized algorithms, is frequently visualized with specialized tools that may be integrated with the mining tools. Through visualization, the analysts are able to interpret the effectiveness of the algorithms and can determine if further iterations are necessary.

Act on the results. Mining on data is of value only if the results can benefit the organization. The results may be applied to improve processes, better target customers or business partners, or contour the organization's spending plan. It may not be possible to determine just how the results of an effective mining activity might effect an organization until after it has been completed. It is the assertion of this author that, even if an organization undertakes a mining activity for a specific purpose, they are likely to find the results will reveal other, unintended opportunities for improvement.

Technology Application and Use

Data mining is a technology that can be applied to a variety of business problems in many different domains. This author asserts that data mining can be effectively applied in any data-

intensive environment. This notion is not novel or unique. Follow are summarizations of some applications of data mining. The initial case provides a generalization of the application of data mining within the *e-Business* domain. The second case is a treatment of the *Glass Box Analysis* environment that employs data mining technologies to address the problems facing analysts in the intelligence arena. The final case briefly describes what is arguably the most ambitious data mining activity yet conceived and, therefore, possible the most controversial – *Terrorism Information Awareness*.

e-Business. The evolution of e-Business has provided an ideal environment for data mining to continue to mature. In e-Business, data mining has a plethora of business problems as well as an abundance of closely related data. *Clickstream analysis,* a specialization of *Business intelligence,* is explained by Knox and Buytendijk (2001) as "the tracking and analysis of visits to Web sites." This class of data mining has demonstrated the ability to produce insightful profiles of those browsing an organizations web site by generating accurate models of the patterns of their browsing trends behavior.

The application of clickstream analysis to understand the online customers has been described by Ramachandran, Turankhia, and Sripad (2002) by what they refer to as *e-Customer Analytics.* They assert that the information captured on a single online transaction is of more value than the sale itself. This assertion is based on the implicit value of that information to allow the business to more accurately target customers.

In their article on the subject, Imhoff and Norris-Montanari (2000) identify the wide variety of data available to the clickstream analyst and characterized it as follows:

1. Tracking information. This includes the user's identification consisting of a client IP address or proxy server identification. The use of a proxy generally abstracts

the user's identity and thus provides less value to the data miner. The customer or user identity, authorized user element used when a secure log-on is required is also available. The request in the form of "GET" or "POST" and the date and time that the Web server responded to the request are also considered tracing information.

2. Server request information about the universal resource locator (URL). This category includes the status of request together with the number of bytes sent.
3. Prior site information. Included in this category are: (a) the URL, (b) the host name, (c) the path, (d) the documents requested, and lastly (e) the query string.

Additionally, details of the customer's browser and client operating system are also available. Collectively, this information provides a very insightful profile of the customer.

Glass box analysis. The Advanced Research and Development Activity (ARDA) for Information Technology has initiated an activity to explore ways to build a knowledge base of what both intelligence analysts and consumers of national intelligence know. The project is named *Novel Intelligence from Massive Data (NIMD).* An architectural component of NIMD is the *Glass Box Analysis* (GBA) element. This element will capture and record activities that occur during the course of analysis, including the stream of analytic tasking, queries, documents examined, and reports produced. This encompasses the full range of clickstream data described in the case above and extends it with a rich set of other information to included, if possible, the analyst's hypotheses, assumptions, methods, and biases (Advanced Research and Development Activity [ARDA], 2003). While the intent of the NIMD project in general, and the GBA in

particular, far exceeds even the most grandiose claims of data mining, the technology is certainly a contributing enabler.

Terrorism information awareness. The Terrorism Information Awareness (TIA) program is an effort undertaken by the Defense Advanced Research Projects Agency (DARPA) to develop and integrate computer technologies that will mine public and private databases to find patterns and associations suggesting terrorist activity. TIA consists of several individual activities – each exploring an enabling technology that can be applied to the problem set. The *Evidence Extraction and Lind Discovery (EELD)* activity is needed due to the inability of existing commercial data mining technologies to address the intelligence-oriented problem. A joint report to congress (2003) by the Secretary of Defense, Attorney General, and the Director of Central Intelligence asserted that data mining technologies of today are capable only of finding broadly occurring patterns and they are not effective at "following a narrow trail and building connections from initial reports of suspicious activity."

Security Issues

Many find it alarming just how much information about their time online is exposed for others to exploit. De Lotto (2001) summarized findings of a survey conducted in 2000 by Business Week and Harris Poll that 98% of polled consumers were not comfortable with their browsing habits or shopping patterns linked with their real identities. The American population is even more sensitive to the potential for their personal information to be aggregated by government agencies – even when they perceive the cause to be desirable. In reaction to concern that the program would compile personal information of citizens, Congress imposed a moratorium on the implementation of data mining under the TIA or any other similar program of the Department of Homeland Security. Congress provided a ninety day period for a joint report

by the Secretary of Defense, Attorney General, and the Director of Central Intelligence in which they were to detail the program to determine if it would be effective in the war on terrorism while not violating privacy laws or civil liberties (Coyle, 2003). The required report, dated May 20, 2003, presents three program constrains that are believed to satisfactory address both privacy and civil liberties. Those are:

1. Full compliance with all laws and regulations governing intelligence activities and all laws that protect the privacy and constitutional rights of U.S. persons.
2. Development of, as an integral part of the program, new technologies that will safeguard the privacy of U.S. persons.
3. Use for research and testing of either real intelligence information that the federal government has already legally obtained, or artificial information that does not implicate the privacy interests of U.S. persons.

The Electronic Privacy Information Center (EPIC), a nonprofit public-interest group that believes the TIA program could undermine civil liberties and an individual's freedoms, continues to leverage the Freedom of Information Act as a tool to gain access to documents related to TIA to contribute to awareness of the programs implications.

Application to Homeland Defense

Though the scope of implementation may be constrained due to necessary lamentations based on security, privacy, and civil liberties, data mining most certainly does have a future in the Homeland Defense domain. TIA does represent a view of a capability with the most potential to contribute to a national defense. That view continues to struggle for acceptance and continued support. This author suggests that, in the absence of another 9/11 style attack on the homeland, it is unlikely the American citizens will cede protection of privacy and civil liberties

to allow the operational deployment of TIA. That is not to suggest a belief that core capabilities of TIA will not mature. TIA is both a research and an integration activity. It is not the development of the technologies that EPIC and other rights advocates are resisting; it is the integration of those technologies to achieve a specific end. Individually, those technologies can benefit the Department of Homeland Defense.

Conclusions

Though concerns over security, privacy, and civil liberties are certain to continue, so too will data mining technologies continue to mature. Brick and mortar businesses and e-businesses alike will continue to capture vast quantities of data that provide the fertile environment for continued development of data mining technologies. As the nation reconciles the conflict between safety and security of its citizens and the rights of those same citizens to privacy and civil freedoms, data mining technologies remain ready to be applied to our homeland defense challenges.

References

Attorney General, Director of Central Intelligence, & Secretary of Defense (2003, May). Report to congress regarding the Terrorism Information Awareness program. Retrieved June 16, 2003 from http://www.darpa.mil/body/tia/tia_report_page.htm .

Advanced Research and Development Activity (n.d.). Glass Box Analysis. Retrieved from June 16, 2003 from http://www.ic-arda.org/Novel_Intelligence/ .

Coyle, M. (2003, June). Fretting over U.S. data collection critics see a lack of privacy protection. *National Law Journal*, 25 (82), p1.

De Lotto, R. (2001, April). Clickstream: Fine to track customers; best at losing them. *Gartner Research Note*, SPA-13-2994.

Edelstein, H. A. (2001, March), Pan for gold in the clickstream. Retrieved June 16, 2003 from http://www.informationweek.com/828/prmining.htm .

Imhoff, C. & Norris-Montanari, J. (2000, August). WhoAmI.com. Retrieved June 16, 2003 from http://www.dmreview.com/editorial/dmreview/print_action.cfm?EdID=2541 .

Knox, M. & Buytendijk (2001, April). By definition, web analytics needs explaining. *Gartner Commentary,* COM-13-3114.

Ramachandran, P. M., Turakhia, K., & Sripad, R. (2002, October). E-customer Analytics. Retrieved June 16, 2003 from http://www.dmreview.com/editorial/dmreview/print_action.cfm?EdID=5841 .

Sweiger, M. (2002, April). Is clickstream data warehousing dead? Retrieved June 16, 2003 from http://www.dmreview.com/editorial/dmreview/print_action.cfm?EdID=4949 .

Appendix B. Treatment of Computational Grids

The purpose the following is to provide a concise summary of the state of maturity of grid computing based on a review of available peer-reviewed papers. The fundamentals of the computational grid are provided first through a brief discussion of its structure and function. This is followed by a concise treatment of the grid computing maturation roadmap to include the computational grid as an enabling infrastructure. A discussion of ongoing related research activities and a discussion of issues likely to challenge its continued maturity will be provided. The paper will conclude by summarizing the maturity of grid computing as an emerging paradigm for computing and its interrelationship to existing computing paradigms.

Fundamentals of Grid Computing

The seminal work on grid computing was authored by Foster and Kessleman and published in 1998. They employed an analogy of electrical power grid to illustrate their notion of a computing infrastructure that would bring to bear appropriate computing resources for the computational problem at hand. Their approach, discussed in the paragraphs below, would provide an increase in computational power of five orders of magnitude to users within a decade. This was predicated on innovations being made in the areas of a) technology improvements, b) increase in demand-drive access to computational power, c) increased utilization of idle capacity, d) greater sharing of computational results, and e) new problem solving techniques and tools. Furthermore, they asserted “it is the combination of dependability, consistency and pervasiveness that will cause computational grids to have a transforming effect on how computation is performed and used” (Foster & Kesselman, 1998, p. 3). More recent comparisons (Chetty & Buyya, 2002) largely support the comparison the computational grid to

the electrical power grid while drawing attention to the fact that the computational grid lacks the regulatory oversight imposed on the power grid.

The problem at which grid computing is targeted is “coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations” (Foster et al., 2001, p.2). Central to the problem as stated is the notion of the *Virtual Organization* (VO) which represented a set of entities that are interested in participating to a common set of sharing rules applicable to given resources (e.g. CPUs, sensors, data, etc.). There is no necessary assumption of common geographic presence for any of the entities or resources. Though this problem does overlap to some extent with other technology trends (i.e. peer-to-peer, enterprise computing, distributed computing, and the internet), it is unique in that the other trends do not aspire to address the problem in whole.

Structure. Four views of the grid computing infrastructure that describe its structure are a) protocols, b) standards, c) application programming interfaces, and d) software development kits. The protocols provide the foundation for entities within the VO and the available resources to negotiate, establish, and maintain the sharing relationship (Foster et al., 2001). It is through an standards-based open architecture that it will be possible to achieve extensibility, interoperability, portability, and code sharing. Development and adoption of new protocols and standards are necessary to address necessary attributes of the grid that are absent from other computing trends such as quality of service and dynamic optimization of resources.

Architecturally, the grid infrastructure has been described in a layered model akin to the Internet protocol model (Foster et al., 2001). Figure 1 provides a comparative illustration of the grid protocol architecture to the Internet model. The layers of the grid protocol architecture are named a) fabric, b) connectivity, c) resource, d) collective and e) application. The fabric layer is

the most abstract layer and has the most direct interfaces with the concrete resources on the grid while the application layer is necessarily the most transparent layer to developers and users. The need for brevity precludes even a modest treatment of each of the layers. Interested readers are encouraged to reference the cited works for a complete understanding.

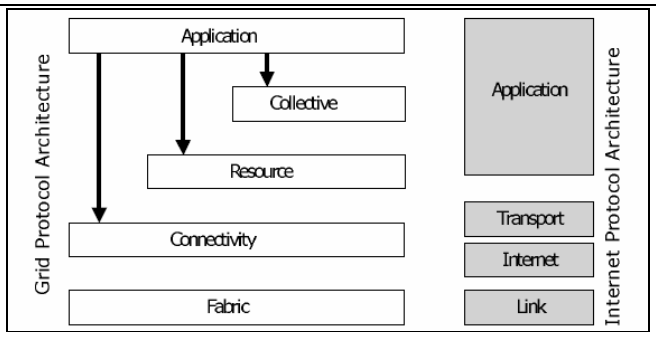

Figure 1. Comparison of the Grid and Internet protocol architectures. (Foster, Kesselman, & Tuecke, 2001, p. 7).

Function. Foster et al. (2002) leveraged the *web services* construct in articulating the function of the grid computing environment. They describe *grid services* functioning together by dynamically aligning themselves in support of needs as they are stated. Grid services, as idealized, will differ from web services in that they will not be tightly coupled with a single (deployed) platform. That is the code that provides a given set of functionality will be mobile. As such, the identification of the software to be applied to a given processing need will be made independent of the selection of suitable hardware. The code will then be transported to the platform and executed in support of the task (e.g. workflow). Central to such large scale distribution of computing is *quality of service (QoS).* There is a need to provision resources (e.g. CPUs, sensors, data, and communications bandwidth) according to some quantifiable schedule. One proposed QoS scheme recognizes distinction between categories of applications users will need and the characteristics of their computing needs (Table 1).

Category	Examples	Characteristics
Distributed supercomputing	DIS Stellar dynamics Ab initio chemistry	Very large problems needing lots of CPU, memory, etc.

High throughput	Chip design Parameter studies Cryptologic problems	Harness many otherwise idle resources to increase aggregate throughput
On demand	Medical instrumentation Network-enabled solvers Cloud detection	Remote resources integrated with local computation, often for bounded amount of time
Data intensive	Sky survey Physics data Data assimilation	Synthesis of new information from many or large data sources
Collaborative	Collaborative design Data exploration Education	Support communication or collaborative work between multiple participants.

Table 1. Classes of grid applications. (Foster & Kesselman, 1998, p. 6).

Related Research

The preponderance of the materials available on the subject suggest that grid computing is an extension of many of the modern distributed computing schemes that have matured over the recent past (e.g. peer to peer, distributed, enterprise, etc). Foster (2001) enumerates four areas of technology with which grid computing is necessarily coupled. Those are: a) world wide web technologies such as the TCP/IP, HTTP, and SOAP protocols and HTML and XML languages, b) applications and storage service providers, c) enterprise computing systems, and d) internet and peer-to-peer computing. It is suggested that grid computing will mature by exploiting these technologies rather than supplanting them. So tightly inter-related are grid computing and peer-to-peer computing that it has been suggested that grid computing is simply a specialization of peer-to-peer (Loo, 2003) though this is not generally supported in the literature. Table 2 illustrates how many of the functions of the inter-related technologies apply to the layered model of grid computing.

	Multidisciplinary Simulation	Ray Tracing

Collective (application-specific)	Solver coupler, distributed data archiver	Checkpointing, job management, failover, staging
Collective (generic)	Resource discover, resource brokering, system monitoring, community authorization, certificate revocation	
Resources	Access to computation, access to data; access to information about system structure, state, performance	
Connectivity	Communication (IP), service discovery (DNS), authentication, authorization, delegation	
Fabric	Storage systems, computers, networks, code repositories, catalogs	

Table 2. Examples of grid services in the layered model. (Foster & Kesselman, &Tuecke, 2001, p. 14).

Challenges

The computational grid is an infrastructure. Distinguishing it as an infrastructure imbues it with expectations of levels of quality of services, availability, reliability, and standardization exceeding what has yet been realized through the related computing paradigms. Extending the notion of infrastructure and utility still further, Yang, Gou, Galis, Yang, & Liu (2003) have added the concept of *resource on demand* which connotes the ability of the utility to automatically provision the available resources to accomplish the aforementioned objectives.

Much has been accomplished in furthering the concepts of grid computing, however many challenges remain. Though not comprehensive, an enumeration of challenges facing the continued maturity of grid computing includes a) standardization of interface and interchange protocols, b) development of robust security architectures that allow for trusted access to shared resources as well as for execution of mobile code, c) development of schemes for provisioning of resources, d) development of schemes for ensuring quality of service, e) installation of dynamically configurable network switching and routing devices (allowing auto-configuration based on mobile code), f) shift to a culture of resource sharing (e.g. allowing others to use otherwise idle CPU cycles), and g) synthesis of data models, taxonomies, and ontologies

(Chervenak, Foster, Kesselman, Salisbury, & Tuecke, 1999) supporting universal access and use of data. Much work remains.

Conclusions

As demonstrated though research activities such as that conducted by the Globus Alliance (http://www.globus.org) are rapidly advancing the concepts of grid computing. Success in deploying computational grids to solve complex, data-intensive problems strengthen the interest in, commitment to, and understanding of this maturing computing paradigm. While it is certain that real, measurable improvements have resulted from the efforts directed toward grid computing, it is too early to determine if the prediction to provide an increase of computing capability of five orders of magnitude (Foster, 1998) within a decade will be realized.

References

Chervenak, A., Foster, I., Kesselman, C., Salisbury,C., & Tuecke, S. (1999). *The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets.* Retrieved October 4, 2003, from http://www.globus.org/documentation/incoming/JNCApaper.pdf.

Chetty, M. & Buyya, R. (2002). Weaving computational grids: How analogous are they with electrical grids? *Computing in Science & Engineering,* 4 (4), 61 – 71.

Foster, I. & Kesselman, C. (Eds.) (1998). *The grid: Blueprint for a new computing infrastructure*. San Francisco, CA: Morgan Kaufmann Publishing.

Foster, I., Kesselman, C., Nick, J. & Tuecke, S. (2002). *The physiology of the grid: An open grid services architecture for distributed systems integration*. Retrieved October 4, 2003, from http://www.globus.org.

Foster, I., Kesselman, C., & Tuecke, S. (2001). The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of High Performance Computing Applications,*15 (3), 1-25.

Loo, A. W. (September, 2003). The future of peer-to-peer computing: An economical method for pumping up computing power by taping into P2P systems using web server technologies. *Communications of the ACM,* 46 (9), 57 – 61.

Yang, K., Guo, X., Galis, A., Yang, B., & Liu, D. (2003). Towards efficient resource on-demand in grid computing. *AMC SIGOPS Operating Systems Review,* 37 (2), 37 – 43.