A vast amount of information is currently stored in digital data repositories, yet it is often difficult to understand what is the important and useful information in those massive data sets. To sift large data sources, computer scientists designed software techniques and tools that can analyze data to find useful patterns—these techniques contribute to the so-called knowledge discovery in databases (KDD) process. In particular, data mining is the basic component of the KDD process for the semiautomatic discovery of patterns, associations, changes, anomalies, events, and semantically significant structures in data. Typical examples of data mining tasks are data classification and clustering, events and values prediction, association rules discovery, and episodes detection [3].

Attempts to automate the process of knowledge extraction began in the early 1980s, with work on statistical expert systems. But today new techniques, mainly in the artificial intelligence field, such as rule induction, neural networks, Bayesian networks, and genetic algorithms, are used. The huge size of data sources means we cannot perform detailed analysis unaided, but must use fast computers applying sophisticated software tools.

Recently, several KDD systems have been implemented on parallel computing platforms to achieve high performance in the analysis of large data sets stored at a single site. However, KDD systems must be able to handle and analyze multisite and multionwer data repositories. The combination of large data set size, geographic distribution of data, users and resources, and computationally intensive analysis demands for new parallel and distributed knowledge discovery (PDKD) platforms. In this setting computational grids are an emerging infrastructure that enables the integrated use of remote high-end computers, databases, scientific instruments, networks, and other resources. Grid applications often involve large amounts of computing and/or data. For these reasons, we believe grids can offer effective support for the implementation and use of PDKD systems.

This article introduces and discusses a reference software architecture for geographically distributed PDKD systems called Knowledge Grid. The architecture is built on top of a computational grid that provides dependable, consistent, and pervasive access to high-end computational resources. The Knowledge Grid uses the basic grid services and defines a set of additional layers to implement the services of distributed knowledge discovery on globally connected computers where each node can be a sequential or a parallel machine. The Knowledge Grid enables the collaboration of scientists that must mine data stored in different research centers as well as analysts that must use a knowledge management system operating on several data warehouses located in the different company establishments.

**Parallel and Distributed Data Mining on Grids**

Parallel and distributed knowledge discovery is based on the use of high-bandwidth communication networks and high-performance parallel computers for the mining of data in a distributed and parallel fashion. This technology is particularly useful for large organizations, environments, and enterprises that manage and analyze data that is geographically distributed in different data repositories or warehouses [6].
The Grid has recently emerged as an integrated infrastructure for coordinate resource sharing and problem solving in distributed environments. Grid applications often involve large amounts of data and/or computing, and are not easily handled by today’s Internet and Web infrastructures. Grid middleware targets technical challenges in such areas as communication, scheduling, security, information, data access, and fault detection [4]. However, mainly because of the recent availability of grid middleware, until recently very few efforts have been devoted to the development of PDKD tools and services for the computational grid. Because of the importance of data mining and grid technologies, it is very useful to develop data mining environments on grid platforms by deploying grid services for the extraction of knowledge from large distributed data repositories.

Motivated by these considerations, we have designed a reference software architecture—the Knowledge Grid—for the implementation of PDKD systems on top of grid systems such as the Globus Toolkit and Legion. We attempt to overcome the difficulties of wide area, multisite operation by exploiting the underlying grid infrastructure that provides basic services such as communication, authentication, resource management, and information. To this end, we organized the Knowledge Grid architecture so that more specialized data mining tools are compatible with lower-level grid mechanisms. The basic principles that motivate the architecture design of a grid-aware PDKD system such as the Knowledge Grid include:

- Data heterogeneity and large data-set-handling;
- Algorithm integration and independence;
- Compatibility with grid infrastructure and grid awareness;
- Openness;
- Scalability; and
- Security and data privacy.

**Basic Grid services.** As mentioned previously, grid infrastructure tools such as Globus Toolkit [4] and Legion, provide basic services that can be effectively used in the development of the Knowledge Grid handling distributed, heterogeneous computing resources as a single virtual parallel computer. To outline the type of services, we list the Globus generic services and Data Grid services [2] in Figure 1. These services address several PDKD requirements discussed here and are helpful for the implementation of the Knowledge Grid architecture.

**The Knowledge Grid Architecture**

The Knowledge Grid architecture is defined on top of grid toolkits and services: it uses basic grid services to build specific knowledge extraction services [1]. Following the Integrated Grid Architecture approach [4], these services can be developed in different ways using the available grid toolkits and services. Here, we discuss an architecture based on the Globus Toolkit 2.0.

**Knowledge Grid services.** The Knowledge Grid services are organized in two hierarchic levels: core K-grid layer and high-level K-grid layer. The former refers to services directly implemented on the top of generic grid services, the latter refers to services used to describe, develop, and execute PDKD computations over the Knowledge Grid. The Knowledge Grid layers are depicted in Figure 2, which shows layers as implemented on the top of Globus services; moreover, the Knowledge Grid data and metadata repositories are also shown. In our architecture, a generic grid node implements the Globus middleware, whereas a K-grid node also implements the Knowledge Grid services.

**Core K-grid layer.** The core K-grid layer has to support the definition, composition, and execution of a PDKD computation over the Grid. Its main goals are the management of all metadata describing characteristics of data sources, third-party data mining, data management, and data visualization tools and algorithms. Moreover, this layer coordinates the PDKD computation execution, attempting to match the application requirements and the available grid resources. This layer comprises the basic services described here.

**Knowledge Directory Service (KDS).** This service extends the Globus Monitoring and Discovery Service (MDS) and is responsible for maintaining a description of all the data and tools used in the Knowledge Grid. The metadata managed by the KDS involves the following kinds of objects:
• Data sources providing the data to be mined, such as databases, text files, XML documents, and other structured or unstructured data. Usually data to be mined is extracted from its sources only when needed;
• Tools and algorithms used to search/find, extract, filter, and manipulate data (data management tools);
• Tools and algorithms used to analyze (mine) data (data analysis tools);
• Tools and algorithms used to visualize, store, and manipulate PDKD computation results (data visualization tools);
• PDKD execution plans, which are graphs describing the interactions and data flows among data sources, data mining tools, visualization tools, and result storing. In fact, an execution plan is an abstract description of a PDKD grid application; and
• PDKD results, that is, the “knowledge” discovered after a PDKD computation.

The metadata information is represented by XML documents and is stored in a Knowledge Metadata Repository (KMR). For example, the metadata describes features of different data sources that can be mined, such as location, format, availability, available views, and level of aggregation of data.

Whereas it could be infeasible to maintain the data to be mined in an ad hoc repository, it could be useful to maintain a repository of the “knowledge” discovered as a result of PDKD computation. This is useful not only for the analysis of results, but also because the output of a computation could be used as the input of another computation. This information is stored in a Knowledge Base Repository (KBR), but the metadata describing it is managed by the KDS. The KDS is used not only to search and access raw data, but also to find prediscovered knowledge that can be used to compare the output of a given PDKD computation when varying data, or to apply data mining tools in an incremental way.

Data management, analysis, and visualization tools usually are preexistent to the Knowledge Grid, so they should not be stored in any specialized repository (they reside over file systems or code libraries). However, to make them available to PDKD computations, relevant metadata must be stored in the KMR. In a similar way, metadata is stored to allow the use of data sources. Finally, the Knowledge Execution Plan Repository (KEPR) is used for storing the execution plans of PDKD computations.

Resource Allocation and Execution Management Services (RAEMS). These services are used to find a mapping between an execution plan and available resources, with the goal of satisfying requirements (performance, response time, and I/O bandwidth) and constraints (available computing power, storage, network bandwidth, and latency). The mapping has to be obtained effectively and by (co-)allocating resources. After the execution plan has been started, this layer has to manage and coordinate the application execution. Other than using the KDS and the MDS services, this layer is directly based on the GRAM services (see Figure 1). Resource requests of single data mining programs are expressed using the Resource Specification Language (RSL). The analysis and processing of the execution plan will generate global resource requests that in turn are translated into local RSL requests for local GRAMs and communication requirements for Nexus or other high-level communication services.

High-level K-grid layer. The high-level K-grid layer comprises the services used to compose, validate, and execute a PDKD computation. Moreover, the layer offers services to store and analyze the discovered knowledge. Main services include those described here.

Data Access Services (DAS). The Data Access Services are responsible for the search, selection (data search services), extraction, transformation, and delivery (data extraction services) of data to be mined. The search and selection services are based on the core KDS services. On the basis of the user requirements and constraints, the DAS automates the searching and finding of data sources to be analyzed by the data mining tools.

The extraction, transformation, and delivery of data to be mined (data extraction) are based on the Globus GASS services and use the KDS. After useful data has been found, the data mining tools can require some transformation, whereas the user requirements or security constraints may need some data filtering before extraction. These operations can usually be done after the data mining tools are chosen.

Tools and Algorithms Access Services (TAAS). These services are responsible for search, selection, and down-

![Figure 2. Knowledge Grid architecture layers.](image-url)
loading of data mining tools and algorithms. As before, metadata regarding their availability, location, configuration, and other information is stored in the KMR and managed by the KDS, whereas the tools and algorithms are stored in the local storage facility of each K-grid node. A node wishing to export data mining tools to other users must publish them using the KDS services, which store the metadata in the local portion of the KMR. Some relevant metadata are parameters, format of input/output data, kind of data mining algorithm implemented, resource requirements and constraints, and so on.

**Execution Plan Management Services (EPMS).** An execution plan is an abstract description of a PDKD grid application. It is a graph describing the interaction and data flows between data sources, extraction tools, data mining tools, visualization tools, and storing of knowledge results in the KBR. In the simplest cases the user directly describes the execution plan, using a visual composition tool where the programs are connected to the data sources. However, due to the variety of results produced by the DAS and TAAS layers, different execution plans can be produced, in terms of data and tool locations, strategies to move or stage intermediate results, and so forth. Thus, the EPMS is a semiautomated tool that takes the data and programs selected by the user and generates a set of different execution plans that satisfy user, data, and algorithm requirements and constraints.

Execution plans are stored in the Knowledge Execution Plan Repository to allow the implementation of iterative knowledge discovery processes, for example, periodical analysis of the same data sources that vary over time. More simply, the same execution plan can be used to analyze different sets of data. Moreover, different execution plans can be used to analyze in parallel the same set of data, and to compare the results using different points of view (performance or accuracy).

**Results Presentation Services (RPS).** This layer specifies how to generate, present, and visualize the PDKD results (rules, associations, models, and classification). Moreover, it offers the functions to store these results in different formats in the Knowledge Base Repository (for example, graphics, animations, and text). The resulting metadata is stored in the KMR to be managed by the KDS. Prediscovered knowledge can be used as input for a new discovery process. Thus the KDS is also used to find available prediscovered knowledge or to apply data mining tools in an incremental way.

**Related Work**

Whereas some PDKD systems supporting high-performance distributed data mining have recently appeared [5] (for a short review see [1]), there are only a few projects attempting to build knowledge grids on the top of computational grids. More specifically, many PDKD systems operate on clusters of computers or over the Internet, but none of those systems, to the best of our knowledge, makes use of the computational grid basic services (authentication, data access, communication, and security services). On the other hand, emerging knowledge grids can be roughly classified as domain-specific (TeraGrid, ADaM), and domain-independent knowledge grids. The Knowledge Grid we designed is one of the first attempts to build a domain-independent knowledge discovery environment on the grid.

Here we briefly mention the most significant grid-based projects/systems, discussing differences and common aspects with respect to our Knowledge Grid system. The TeraGrid project is building a powerful grid infrastructure, connecting four main sites in the U.S. (San Diego Supercomputer Center, National Center for Supercomputing Applications, Caltech, and Argonne National Lab) that will provide access to terabyte-scale amounts of data. The most challenging application on the TeraGrid will be the synthesis of knowledge from very large scientific data sets. The use of the Knowledge Grid services can be potentially effective in those applications.

The ADaM (Algorithm Development and Mining) system is an agent-based data mining framework developed at the University of Alabama in Huntsville used to mine hydrology data in parallel from four sites. This system uses a design approach similar to the Knowledge Grid principles but the system architecture is simpler and the system purpose is limited in the application area for which the system has been designed.

The Discovery Net is a newly announced EPSRC’s project (Engineering and Physical Sciences Research Council), at Imperial College in London. This system aims to develop High Throughput Sensing (HTS) applications by using the Kensington Discovery Platform on the top of the Globus services. In this case the rationale is to port a Java-based distributed data mining system to Grid platforms using the Globus toolkit. The main question is how the preexistent system can adapt to Grid mechanisms and policies. Finally, the National Center for Data Mining at the University of Illinois at Chicago is developing some significant testbeds on knowledge discovery over grids. In summary, these emerging knowledge discovery-oriented grids are almost all facing specific application domains. Our system, other being independent by the application domain, adopts specifically designed tools for the management of knowledge discovery processes that allow a user to
evaluate and compare different knowledge models and for the transparent integration of parallel and sequential data mining tools and algorithms.

**Conclusion**

A prototype of the Knowledge Grid has been implemented on top of Globus. We implemented the main components of the system in a toolset called Visual Environment for Grid Applications (VEGA) that offers the functionalities of the DAS and TAAS services (search and selection of data sources and tools), the EPMS services (design of a PDKD application), the RAEMS services (optimization and translation of the execution plan on the Globus code), and the RPS services (result collection and presentation) through a graphical interface that a user can utilize to compose and execute a knowledge discovery computation in a simple way.

Grid computing is the most promising framework for future implementations of high-performance data-intensive distributed applications. Although today the Grid is mainly used for scientific applications, in the near future it will be used for industrial and commercial applications. In these areas knowledge discovery is very important and critical. Furthermore, the Internet is shifting from an information and communication infrastructure to a knowledge delivery infrastructure. The discovery and extraction of knowledge from geographically distributed sources will be increasingly important in many typical daily activities. The Knowledge Grid is a significant step in the process of studying the unification of knowledge discovery and Grid technologies and defining an integrating architecture for distributed data mining and knowledge discovery based on Grid services. Such an architecture will accelerate progress for very large-scale geographically distributed data mining by enabling the integration of various currently disjointed approaches and revealing technology gaps requiring further research and development.

**REFERENCES**


**MARIO CANNATARO** (cannataro@acm.org) is a senior researcher with ICAR-CNR in Rende, Italy. **DOMENICO TALIA** (taliad@acm.org) is a professor at the University of Calabria in Rende, Italy.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

---

**Faculty Positions, Computer Science**

The Department of Computer Science (www.cs.njit.edu) in the College of Computing Sciences at New Jersey Institute of Technology invites applications for faculty positions beginning August 2003.

NJIT is a public research university. The Department of Computer Science, with 26 faculty members and 1,400 students, is part of the newly formed College of Computing Sciences (www.ccs.njit.edu). Departmental research interests include algorithms, artificial intelligence, computational biology and bioinformatics, computer vision, databases, parallel processing, simulation and modeling, software engineering, storage architectures and computer networking. The Department offers programs at the undergraduate, masters and PhD levels.

The Department especially seeks applicants in software engineering and computational biology, and in computer networking and databases. Strong candidates in other fields will also be considered.

Applicants should have a PhD in computer science or a closely allied field. Senior applicants should have an outstanding record of research, teaching, and service. Junior applicants should have demonstrated potential for original research and a commitment to excellence in teaching. Salaries are competitive, and commensurate with appointment rank and qualifications.

NJIT is located in Newark’s University Heights, close to several other universities, affording ample opportunity for collaborative work. NJIT’s location in the midst of major concentrations of pharmaceutical, telecommunications, and financial companies and research laboratories offer excellent opportunities for consulting and industrially-sponsored collaborative research.

New Jersey enjoys a high standard of living and quality of life. Newark is minutes from New York City and close to the Jersey Shore, providing a wide range of cultural and leisure activities. Inquiries may be directed to this email address: faculty-search@cs.njit.edu or to the following website: http://web.njit.edu/cs/recruiting.htm

Applicants should send a curriculum vitae and a list of at least three references to the following:

**Department of Human Resources, Personnel Box CS-AP**

NJIT is a public research university. The Department of Computer Science, with 26 faculty members and 1,400 students, is part of the newly formed College of Computing Sciences (www.ccs.njit.edu).

**Faculty Positions, Computer Science**

The Department of Computer Science (www.cs.njit.edu) in the College of Computing Sciences at New Jersey Institute of Technology invites applications for faculty positions beginning August 2003.

NJIT is a public research university. The Department of Computer Science, with 26 faculty members and 1,400 students, is part of the newly formed College of Computing Sciences (www.ccs.njit.edu). Departmental research interests include algorithms, artificial intelligence, computational biology and bioinformatics, computer vision, databases, parallel processing, simulation and modeling, software engineering, storage architectures and computer networking. The Department offers programs at the undergraduate, masters and PhD levels.

The Department especially seeks applicants in software engineering and computational biology, and in computer networking and databases. Strong candidates in other fields will also be considered.

Applicants should have a PhD in computer science or a closely allied field. Senior applicants should have an outstanding record of research, teaching, and service. Junior applicants should have demonstrated potential for original research and a commitment to excellence in teaching. Salaries are competitive, and commensurate with appointment rank and qualifications.

NJIT is located in Newark’s University Heights, close to several other universities, affording ample opportunity for collaborative work. NJIT’s location in the midst of major concentrations of pharmaceutical, telecommunications, and financial companies and research laboratories offer excellent opportunities for consulting and industrially-sponsored collaborative research.

New Jersey enjoys a high standard of living and quality of life. Newark is minutes from New York City and close to the Jersey Shore, providing a wide range of cultural and leisure activities. Inquiries may be directed to this email address: faculty-search@cs.njit.edu or to the following website: http://web.njit.edu/cs/recruiting.htm

Applicants should send a curriculum vitae and a list of at least three references to the following:

**Department of Human Resources, Personnel Box CS-AP**

NJIT is a public research university. The Department of Computer Science, with 26 faculty members and 1,400 students, is part of the newly formed College of Computing Sciences (www.ccs.njit.edu).

**Faculty Positions, Computer Science**

The Department of Computer Science (www.cs.njit.edu) in the College of Computing Sciences at New Jersey Institute of Technology invites applications for faculty positions beginning August 2003.

NJIT is a public research university. The Department of Computer Science, with 26 faculty members and 1,400 students, is part of the newly formed College of Computing Sciences (www.ccs.njit.edu). Departmental research interests include algorithms, artificial intelligence, computational biology and bioinformatics, computer vision, databases, parallel processing, simulation and modeling, software engineering, storage architectures and computer networking. The Department offers programs at the undergraduate, masters and PhD levels.

The Department especially seeks applicants in software engineering and computational biology, and in computer networking and databases. Strong candidates in other fields will also be considered.

Applicants should have a PhD in computer science or a closely allied field. Senior applicants should have an outstanding record of research, teaching, and service. Junior applicants should have demonstrated potential for original research and a commitment to excellence in teaching. Salaries are competitive, and commensurate with appointment rank and qualifications.

NJIT is located in Newark’s University Heights, close to several other universities, affording ample opportunity for collaborative work. NJIT’s location in the midst of major concentrations of pharmaceutical, telecommunications, and financial companies and research laboratories offer excellent opportunities for consulting and industrially-sponsored collaborative research.

New Jersey enjoys a high standard of living and quality of life. Newark is minutes from New York City and close to the Jersey Shore, providing a wide range of cultural and leisure activities. Inquiries may be directed to this email address: faculty-search@cs.njit.edu or to the following website: http://web.njit.edu/cs/recruiting.htm

Applicants should send a curriculum vitae and a list of at least three references to the following:

**Department of Human Resources, Personnel Box CS-AP**

NJIT is a public research university. The Department of Computer Science, with 26 faculty members and 1,400 students, is part of the newly formed College of Computing Sciences (www.ccs.njit.edu).