THE INTERNATIONAL MUSIC INFORMATION RETRIEVAL SYSTEMS EVALUATION LABORATORY: GOVERNANCE, ACCESS AND SECURITY

J. Stephen Downie
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign

Joe Futrelle
National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign

David Tcheng
National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign

ABSTRACT

The IMIRSEL (International Music Information Retrieval Systems Evaluation Laboratory) project provides an unprecedented platform for evaluating Music Information Retrieval (MIR) and Music Digital Library (MDL) techniques, by bringing together large corpora and significant computational resources with the necessary rights management and technical infrastructure to support a variety of MIR/MDL research areas. The standardized research collection being deployed represents a large and diverse corpus of musical examples, which we are hosting in our secure environment for use in evaluating MIR/MDL algorithms. Grid services and NCSA's D2K machine learning environment provide a powerful, high-performance, and secure framework for designing, optimising, and executing complex MIR/MDL evaluation applications. IMIRSEL provides a community resource for researchers who would otherwise not be able to afford the content rights and computational resources to carry out large-scale MIR/MDL evaluations.

Keywords: evaluation, system modelling, Grid computing

1. INTRODUCTION

The International Music Information Retrieval Systems Evaluation Laboratory (IMIRSEL) is being constructed at the University of Illinois at Urbana-Champaign. IMIRSEL is an integral sub-component of the "Music Information Retrieval (MIR) / Music Digital Library (MDL) Evaluation Project." The principal goal of the project is the creation and refinement of secure yet robust access mechanisms that will allow the manipulation of a unique, terabyte-scale standard corpus of multimodal music materials (e.g., audio, text, symbolic, and metadata). These materials are being put together for the research and evaluation use of the international MIR/MDL research community. Because

of space limitations, readers are directed to [3] and [4] for background information and detailed explications of project motivations, goals and components. Figure 1 illustrates the basic IMIRSEL framework.

In this paper, we introduce and outline the three key features of IMIRSEL that will directly affect members of the MIR/MDL research community. Section 2 introduces the governance structure of IMIRSEL along with the sets of first prinicples guiding its operation and management. Section 3 presents information on the Virtual Research Lab (VRL) architecture that we are developing to provide seamless access to the test collections and to the supercomputing resources of the National Center for Supercomputing Applications (NCSA). Section 4 discusses the security framework being laid out to protect the test collections from illicit distribution and cyber-vandalism. Section 5 presents the summary and plans for future work.

2. GOVERNANCE

2.1. Introduction

It is important that IMIRSEL outlive its initial four-year funding period. We see the initial four-year time span as only the first phase in the construction of a permanent, evolving and vibrant research resource for the MIR/MDL community. This is particularly true if IMIRSEL is to play its intended role as a continuing primary locus of the proposed annual TREC-like evaluation events. To ensure the long-term sustainability and impact of IMIRSEL, a two-level governance and advisory structure is being put into place. At the University level, a five-member Board of Governors (BOG) is being struck. The BOG is the entity that has official responsibility for all the legal aspects of IMIRSEL including contracts and user-agreements. The International Advisory Board (IAB) forms the second level of governance. The membership of the IAB will be drawn from a wide variety of constituencies and will play a crucial role in ensuring that IMIRSEL fulfils its mandate of service to the MIR/MDL research community.

2.2. Board of Governors

Membership in the Board of Governors consists of the following officers from the University of Illinois at Urbana-Champaign:

1. IMIRSEL Director /Principal Investigator
2. Representative appointed by the Office of the Vice-Chancellor Research
3. Representative appointed by the Office of the University Librarian
4. Representative appointed by the Office of the Director of the National Center for Supercomputing Applications
5. Representative appointed by the Office of the Dean of the Graduate School of Library and Information Science

It is the mandate of the BOG to provide general project oversight. The BOG is charged with the development and vetting of the necessary terms-of-use agreements with the research teams involved in using the test collections. The BOG is also charged with the development and vetting of all contractual matters with present and future contributors of database content.

With regard to the creation of the terms-of-use agreements, and the content-provision agreements, the five basic principles guiding all decisions are:

1. Recognition of the supreme importance of protecting the valuable intellectual property of the content providers.
2. Recognition of the important responsibility each participating team has of protecting the valuable intellectual property of the content providers.
3. Recognition that the materials provided are for the scientific research and evaluation purposes of the international MIR/MDL community.
4. Recognition of the importance of dissemination of development and evaluation results to the broader scientific community via publication of findings.

5. Recognition in publications and presentations of the support being provided by the University, project funders, the content providers, and so on.

2.3. International Advisory Board

An International Advisory Board (IAB) is also being struck. Representatives are being drawn from the following classes of individuals:

1. The content-provider community
2. The user-community (i.e., users of the test-collection database)
3. The International Conferences on Music Information Retrieval steering committee
4. The general MIR and MDL communities
5. The music industry
6. The music library community
7. The traditional IR and TREC communities

It is the mandate of the IAB to report to the BOG on an advisory basis concerning such matters as progress being made, access issues, design and implementation of the TREC-like evaluation experiments, opportunities for future funding and collaborative research, and so on.

Communications with the IAB will reside primarily in the electronic domain with http://music-ir.org playing a central role. We plan on meeting with available members at least one a year, most likely in conjunction with the International Conferences on Music Information Retrieval (ISMIR). At least one on-site meeting (i.e., at UIUC) is also planned. Meeting onsite is intended to deliver a clearer understanding to the IAB of the physical and organizational infrastructure involved in the project and would thus allow the IAB to provide more informed feedback and recommendations. We also see onsite meeting(s) as an opportunity to solicit the support of potential funding agencies and content-providers (i.e., demonstrate to them the project's important features including security, community support, and collaborative research opportunities involving the various parties, etc.).
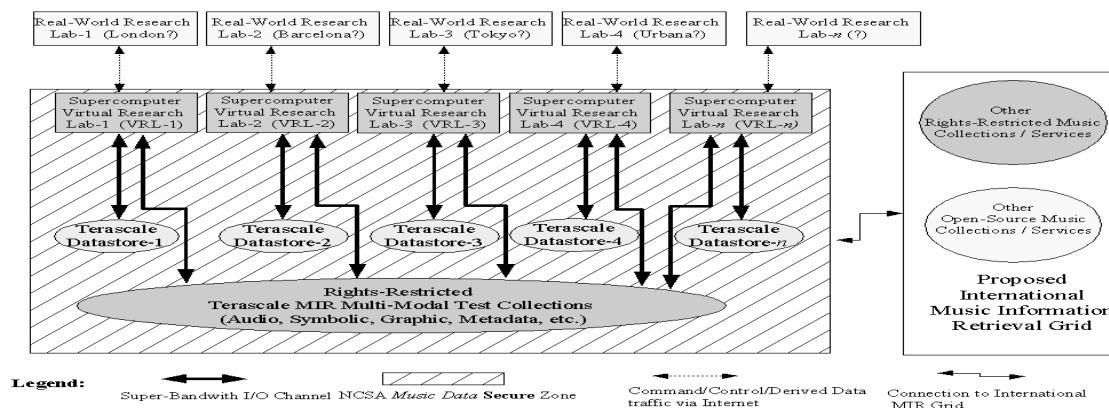

Figure 1 General schematic of IMIRSEL as presented in [3] and [4].

3. VIRTUAL RESEARCH LAB ACHITECTURE

3.1. NCSA and The "Grid"

The National Center for Supercomputing Applications (NCSA) is part of the University of Illinois. Its mandate is to enable academic researchers to get the most out of its vast computing resources. NCSA grid computing resources include five world-class supercomputers with a total of approximately 5000 processors and 40 TB of RAM. NCSA's total compute power is in the 30 Teraflop range [5]. In addition, NCSA maintains high-capacity, high-performance storage resources such as Unitree, a tape-based tertiary storage system with essentially unbounded capacity and 6TB of high-speed disk cache [6].

Within NCSA is the Automated Learning Group (ALG) which specializes in data mining and cyber infrastructure development. Over the years, ALG has embedded all of its computing tools in a software system called "D2K" which stands for "Data to Knowledge" [9]. D2K can be viewed as two things. First, it is a new parallel programming language; second, D2K is a body of reusable software components designed for the quick modeling and deployment of new problem-solving programs.

3.2. D2K

At a fundamental level, D2K is a data flow programming language, along with a development environment called the "D2K Toolkit" that allows users to visualize data flow programs and interactively develop new D2K applications to meet their needs.

3.2.1. Modules and Pipes

D2K's atomic computational unit is called a "module"—a black box with a number of inputs and outputs. Modules can be developed using a number of programming languages (e.g., Java, C, C++, Matlab, Perl, etc.) allowing D2K to integrate diverse codes. The D2KToolkit IDE is written in 100% java for maximum portability. To allow for the incorporation of codes written in languages other than Java, D2K has tools for "wrapping" modules written in other languages—provided they follow standardized API for the language.

When a module executes, it "pulls" data objects from one or more of its inputs, does some computation, and "pushes" the resulting data objects into one or more of its output pipes. A single module execution can result in the pulling or pushing of more than one data object per pipe. Therefore a module is viewed as computing processes rather than mathematical functions.

Connections between modules inputs and outputs are called "pipes". Each pipe has a buffer and can hold more than one data object at a time. As described below, buffers allow for a useful type of parallelism automatically employed when pipes start to fill up.

3.2.2. Itineraries

Modules are composed together to form a directed graph (i.e., a flow chart) linking module inputs to module outputs. The directed graph defines a D2K's data flow program (DFP) also known as an "itinerary".

The execution of an itinerary begins by executing the "head" modules (modules with no inputs). After this initial action, all other module executions are triggered by changes to the state module input pipes or by modules executing themselves.

3.2.3. Parallelism

One key advantage of D2K is that it allows for the easy development of parallel codes. On multi-processor systems or a network of machines, more than one module can be executing at any point in time. Using this form of parallelism, the maximum number of modules that can execute simultaneously is equal to the number of modules in the itinerary. The other form of D2K parallelism is "pipe-based" parallelism and is invoked when a pipe begins to fill with data.
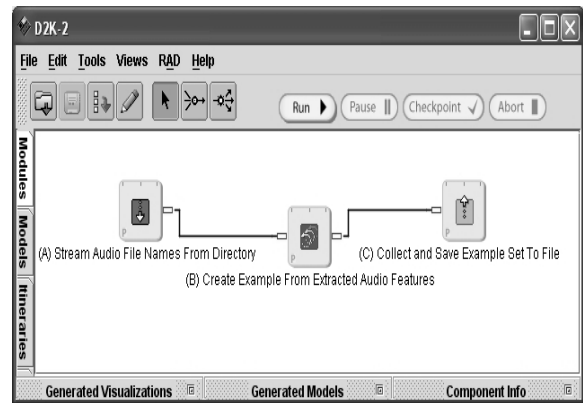

Figure 2: A simple itinerary capable of pipe based parallel processing.

Figure 2 shows a simple itinerary in which module A feeds module B which feeds module C. Module A, pushes a stream of audio file names to module B, one at a time. Module B reads a single file name, extracts features from the audio, and then outputs an example to Module C. Module C, collects all examples created and when the stream ends, writes the set of examples to a file. Modules A, B, and C all fire repeatedly, but only modules A and C need state memory to operate correctly. Module A uses state memory to keep track of which audio file name in the directory to output next, and Module C has state memory to accumulate the complete example set. Module B, however, is stateless, meaning its behavior is independent of previous executions. Since D2K knows that Module B is stateless, it monitors its input pipe and when Module B's input pipe has more than one element, D2K will create multiple copies of the Module B (if there are idle CPUs) routing specific inputs to each copy. All copies of Module B execute in parallel and their results are

collated and pushed into the pipe connecting B and C. On a 256 processor cluster, this itinerary could cause all 256 processors of the cluster to be simultaneously running copies of Module B, reading files, and extracting audio features. D2K users can limit the total number of processors used to any number less than 256.

3.3. D2K and the Virtual Research Labs

NCSA's computing Grid resources are vast, but using them can be cumbersome. Currently the Grid supports only batch processing. Jobs are submitted. They wait in a queue for an unknown amount of time (depending on problem size and system load). They begin execution by loading necessary data from mass storage and, after execution completes, the final results are saved to mass storage. This batch processing paradigm, while efficient from the perspective of overall Grid throughput, does not support interactive design and testing of D2K itineraries which is necessary for VRL users.

To address this problem, we envisage a two stage process for creating new applications: development and production.

In the development phase, the users execute D2K locally on their desktop computers or network of computers. Using local resources allows for interaction with the D2K Toolkit for designing and testing itineraries. Once the user is satisfied that the D2K itinerary runs correctly, the user would proceed to the production phase.

In the production phase, the D2K itinerary and necessary data files will be submitted to the NCSA Grid as a batch process using standard grid protocols. Once the batch process executes the results will be retrieved from the Grid and presented to the user.

Users will be able to "toggle" between local and Grid modes. In local mode, itinerary response will be instantaneous but users will be constrained by their own, probably limited, computational resources. In Grid mode, batch processes will be queued, perhaps for some time, however, once the processing begins, users will benefit from the fact that their full-scale itineraries will be drawing upon the supercomputing resources of NCSA.

3.3. A Virtual Research Lab Setup: An Example

Figure 3 illustrates a D2K itinerary for evaluating a MIR contest solution for genre classification given a predefined set of training and testing audio files and their associate genres.

Audio file names and their genres (as well as other metadata not related to this problem) are stored in an Oracle database. Access to the database is enabled with the modules "Get Genre Classification Training Files From DB" and "Get Genre Classification Test Files From DB". These modules output a stream of audio file names and a stream of associated genres.

These modules are considered *input* modules because they are the source of the data driving the D2K itinerary. Input modules icons are recognized by the downward pointing arrow in their icon.

The module "Create Audio Example" reads a file name, analyzes the audio file extracting a set of numeric audio features, and creates an example by pairing the audio features with the genre the database (the ground truth). The two instances of the "Create Audio Example" module in this itinerary are identical copies of each other. Given the data flow topology, these modules can operate in parallel. Because they are both stateless modules, multiple copies of each can be created, completely saturating all available computing resources.
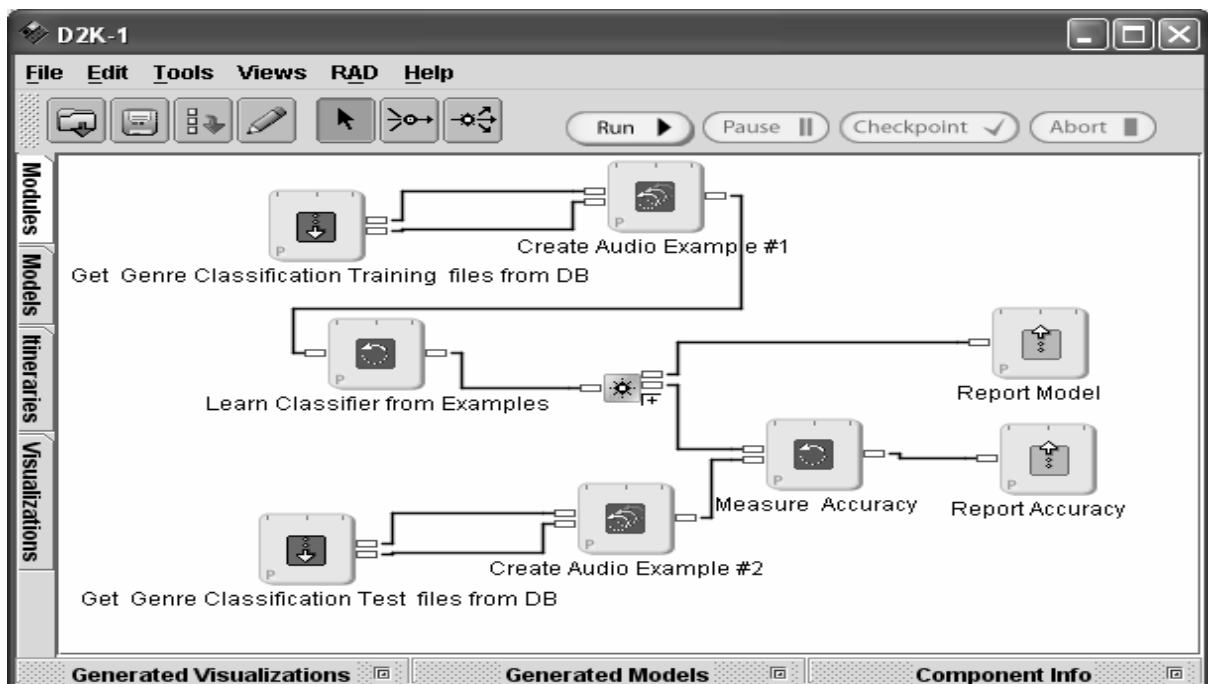

Figure 3: A D2K itinerary for evaluating the performance of a genre classification system.

Stateless modules that can be copied are recognized by the three looping arrows in their icons.

The "Learn Classifier from Examples" module analyzes a set of supervised learning examples and outputs a predictive model that assigns genre classifications to new unseen examples.

The "Report Model" module simply displays the model in text or graphics to the user. This module is considered an output module and output modules can be recognized by the upwards pointing arrow in the icon.

"Measure Accuracy" takes a model, and a set of testing examples, applies the model the testing examples, computes the error of each prediction, and passes accuracy statistics to the "Report Accuracy" module.

D2K comes with many general purpose modules for learning from examples and since they all conform to the same API, users can easily replace one modeling strategy for another by simply swapping this module out for different one. Given this working example, new users can modify it by replacing the example and/or the model creation modules with their own and avoid re-implementing the supporting modules.

4. SECURITY ARCHITECTURE

4.1. Requirements

Protecting the IP rights of content providers (e.g., Naxos) is one of the most critical challenges faced by IMIRSEL. The primary security goal is to prevent the transmission of copyrighted material from the music collection to any third party, who might then knowingly or unwittingly distribute it. The secondary goal is to provide seamless access to the copyrighted material by research codes, so that MIR algorithms can be evaluated against the collection. Achieving these two conflicting goals requires a non-trivial security architecture.

4.2. Architectural Strategy

There are several components to the IMIRSEL security architecture. The first component is a set of legal agreements between IMIRSEL and content providers allowing for the transfer of copyrighted material to IMIRSEL and the use of copyrighted material by MIR research codes. These agreements are designed to specifically allow only these uses, and retain all other rights, and serve to constrain any other agreements or policies that IMIRSEL will make with researchers using the facility.

The second component of the IMIRSEL security architecture is operating system and network hardening. These standard practices involve the use of firewalls and application proxies, disabling non-essential services, auditing, and disallowing external access to data services. Figure 4 shows the proposed network architecture. Data services are protected by a two-level firewall that only allows traffic from the Grid services host, which acts as an application proxy for all requests that might result in access to the data.
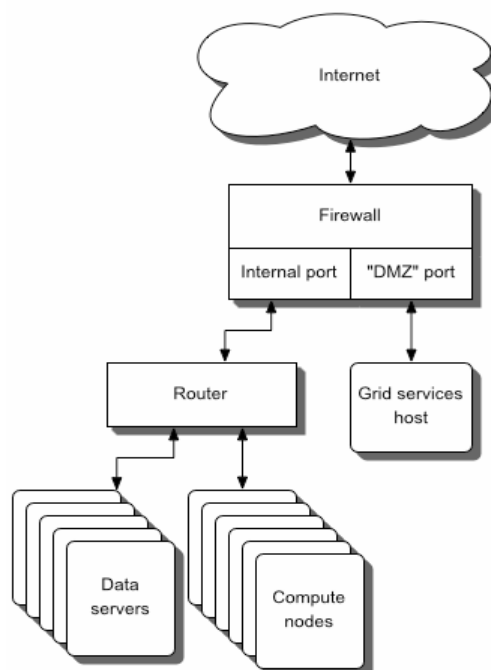

Figure 4: Proposed IMIRSEL network configuration. A two-level firewall with an application proxy (the Grid services host) protects data from external access while providing trusted code on the compute nodes access to the data.

The final component of the IMIRSEL security architecture is the Grid Security Infrastructure (GSI) [2]. GSI provides public-key authentication, message integrity, session encryption, and credential delegation for all services. GSI is a best-of-breed family of security technologies that can be used to protect against unauthorized access, eavesdropping, and code tampering. GSI protects user, host, and service credentials with public-key encryption, and uses a proxy credential strategy to ensure that any credential that is transmitted from one service to another is safe from tampering and cannot be used beyond its limited lifetime. GSI certificates are issued and cryptographically signed by trusted certificate authorities (CA), whose private keys are carefully guarded secrets; IMIRSEL will act as a certificate authority for its users, and will maintain the integrity of the CA's key. No user whose identity has not been verified by IMIRSEL will be given a certificate.

4.3. Trusted Code Model

Since the only entities that require access to copyrighted data are the MIR research codes, IMIRSEL will provide an environment in which only those codes have access to the data, and cannot use that access to transfer the data outside of the IMIRSEL system. This is extends the idea of trusting users to trusting users' code.

The trustworthiness of code will be assured using several mechanisms:

1. *Digital signature*. Researchers will need to cryptographically sign their codes, so that

IMIRSEL will be able to verify at any time that a researcher's code has not been modified by a third party. By signing their code, researchers will be asserting that it is not designed to attempt to circumvent any of IMIRSEL's security protocols.

2. *Auditing*. Researchers' codes will be monitored as they run and any attempts at unauthorized access will be brought to the attention of both system administrators and code authors.

3. *Proxy authentication*. Codes will access data from data server(s) using Grid services. Each code author will delegate his/her credentials to his/her codes, which will act on the author's behalf. In effect, IMIRSEL will trust researchers' code as much as it trusts them. It will be able to manage this trust relationship dynamically [8].

Data will be protected from unauthorized access by Grid access control mechanisms [7] and the firewall architecture described above. Codes will be able to access copyrighted material, but will be prevented from transmitting it outside of IMIRSEL's private network. Researchers will be able to access results of codes (e.g., MIR query results, statistical measures of MIR algorithm performance) using several secure mechanisms, such as Grid-authenticated D2K API calls and GridFTP [1] access to a limited, scratch file area behind the firewall which will be closely monitored so to make sure no copyrighted material is placed there by codes.

5. SUMMARY AND FUTURE WORK

IMIRSEL provides tools, data, computational resources, software support, and security for MIR researchers. It also represents a new, collaborative, Grid-computing based way of doing MIR/MDL research. By supporting a variety of MIR/MDL research in the IMIRSEL environment, IMIRSEL will facilitate the development of shared, reusable technologies and methodologies.

To realize this goal, we plan two major kinds of activities. First, we will provide user support to assist MIR researchers in using the IMIRSEL facility. This involves both training users in how to adapt their codes to run in the IMIRSEL environment (e.g., how to develop D2K applications and interface existing codes such as Matlab codes into D2K), as well as providing system administration and account management. Initially, we will work closely with a small set of "early adopters", then develop FAQ's, tutorials, sample applications, and other training materials. Users will be encouraged to share their collective work using IMIRSEL-provided tools such as discussion groups and shared repositories.

Second, we will extend the D2K framework to accommodate both batch and interactive modes of execution, so that researchers can develop an application using local computational resources and data and then seamlessly deploy it in IMIRSEL's secure environment, or deploy applications in distributed, hybrid environments were computation is shared between local resources and Grid resources. As the project matures, the deployment of hybrid environments will be the grounding upon which we hope to facilitate interactive evaluation modalities (i.e., real-time system and interface evaluations).

6. ACKNOWLEDGEMENTS

7. REFERENCES

[1] Allcock, B., et al. "Data Management and Transfer in High Performance Computational Grid Environments", *Parallel Computing Journal*, 28(5): pp. 749-771, 2002.

[2] Butler, R., et al. "A national-scale authentication infrastructure", *IEEE Computer* 33(12): pp. 60-66, 2000.

[3] Downie, J. S. "The scientific evaluation of music information retrieval systems: Foundations and future", *Computer Music Journal* 28 (2): pp. 12-23, 2004.

[4] Downie, J. S. "Toward the scientific evaluation of music information retrieval systems", *Proceedings of the Fourth International Conference on Music Information Retrieval*, Baltimore, MD, pp. 25-32, 2003.

[5] "NCSA: Hardware", http://www.ncsa.uiuc.edu/AboutUs/Hardware/, 2004.

[6] "NCSA's Mass Storage System", http://www.ncsa.uiuc.edu/Divisions/CC/HPDM/unitree/, 2004.

[7] Pearlman, L., et al. "A Community Authorization Service for Group Collaboration", *IEEE 3rd International Workshop on Policies for Distributed Systems and Networks*, 2002.

[8] Welch, V., et al. "X.509 Proxy Certificates for Dynamic Delegation", *3rd Annual PKI R&D Workshop*, 2004.

[9] Welge, M., et al. "Data to Knowledge (D2K) An Automated Learning Group Report", National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign, 2003.