The following sections document the year's Project Prism results, with sections corresponding to the activities section of this report.

1 Digital Object Architectures

1.1. Context-sensitive behaviors

We have introduced a notion called "Structoids" – units of structural metadata representing relationships among content pieces in a Digital Object. Structoids can be viewed as "behavior potential" – the Structoids in a Digital Object allow it to bind to specific behaviors. Structoids can be thought of as recognizable patterns that have to do with the relationships among content pieces. Thus, Digital objects can be defined by a) their content b) their behaviors c) the relationships among the content pieces.

Structoids are represented in XML, which enables us to validate them, both at creation time and at time of use. XML Schemas allow extensible, variable content in structoids while simultaneously promoting scalable processing of diverse structoids via some unifying pattern. The XML Schemas define controlled vocabulary for a set of structoid labels, the tree hierarchy of the labels, and some of the requirements for the content pieces to be assigned to the labels. Schematron schemas are used for rule-based validation; we need rule-based validation to express content requirements such as matching structoid element tags to appropriate MIME types. Schematron is an elegant rule based validation tool built on top of XSLT. The use of XML Schemas and Schematron for validation means we can leverage existing tools, rather than writing our own idiosyncratic validation tools.

Fedora Repositories must expose Structoids associated with Digital Objects to Context Brokers. To do this we have leveraged our work on Open Archives Initiative (OAI), described in Section 4. We implemented the OAI protocol for our Web access version of Fedora. Thus, the Fedora Repository is an OAI data provider, and the Context Broker is an OAI service provider: the Repository provides structural metadata in the form of structoids while the Context Broker provides services in the form of context-sensitive behaviors performed on digital content based on the Structoids.

We created a prototype Context Broker that is an intermediary between users and Fedora repositories; it presents an experience of Fedora Digital Objects it doesn't control, using behavior mechanisms it also doesn't control. There are two requests in our Context Broker's protocol:

- ***ListBehaviors:*** obtain a list of behaviors available for a specified Digital Object. The Context Broker returns a list of behaviors that can be performed by the Context Broker.

- ***PerformBehavior:*** perform a specified behavior on a specified Digital object. The Context Broker obtains and loads the appropriate behavior mechanism, gets any raw content from the Digital Object required as input by the behavior mechanism (indicated by the Structoid), and returns the output of the requested behavior to the user.

The ListBehaviors request is accomplished with a back-end component called the *Behavior Registry*. A Behavior Registry matches structural metadata from a Digital Object against behavior mechanism input requirements – it determines which behavior mechanisms can be applied to a particular Digital Object based on the structural metadata. So a ListBehaviors request gets the Structoids for a Digital Object via an OAI request to the Fedora Repository, and then matches the Structoids to behavior mechanisms via the Behavior Registry. The Context Broker can then present the available behaviors to the user.

A user's experience of a Digital Object depends on which Context Broker is used for mediation, not on which repository contains the Digital Object. The behaviors provided by a Context Broker depend on the mechanisms in the Context Broker's Behavior Registry. Thus Context Broker's can be tailored to provide locally relevant experiences of digital context. Returning to our example, a Context Broker in France may have behavior mechanisms that translate English text and audio into French, while Gallaudet University's Context Broker may have behavior mechanisms that convert audio into text.

In the context of this work we have made the following other improvements to the Fedora architecture:

- Behavior Interfaces (behavior definitions) are now XML documents accessible with URLs. Since Behavior Interfaces are static data, there was no need for them to be disseminated from executable software – it made sense to represent them in XML. We created an XML schema and functional sample instance documents.

- Behavior Mechanisms are now XML documents accessible with URLs. We created an XML schema for XML wrapped executable bytestreams used as mechanisms, and created functional sample instance documents.

- Persistent Storage of Digital Objects in reference implementation is in XML. This allows us to leverage XML pattern matching tools for indexing and doesn't unnecessarily encode data (making the data more accessible from a preservation standpoint and more human readable).

- Simplified and improved the Fedora protocol
    - Reference (remote) DataStreams now support URLs as an access type.
    - Only one type of Disseminator
        - No more bootstrap disseminators: both Behavior Interfaces and Behavior Mechanisms are now obtained via URLs.
    - Removal of interfaces pertaining to Behavior Interfaces obtained via executables.
    - Addition of Structoids.

    - Improved elegance of how SessionContext is bound to Repository and DigitalObject interface.

1.2. University of Virginia Mellon Collaboration

With the collaboration beginning in fourth quarter of 2001, we began initial design work on the joint FEDORA implementation. Some of the features of the production FEDORA area as follows:

- XML and XML Schema for the encoding of Digital Objects
- Management and Access sub-systems, both to be exposed as Web Services, using Standard Object Access Protocol (SOAP) and Web Services Definition Language (WSDL).
- A standardizing of the way Fedora interfaces to external applications. The Fedora repository system will use WSDL to encode interface definition and program binding information for external applications and services used by the repository. The Fedora repository will store special digital objects (Behavior Definition Objects and Behavior Mechanism Objects) that represent these external services. At runtime, Fedora will access these objects to obtain information necessary to invoke a request to an external service (i.e., application or mechanism). We intend to support both basic HTTP bindings and SOAP/HTTP bindings to the services.
- Versioning of digital objects and mechanisms to preserve both the evolution of object content and the look and feel over time. Ultimately, clients can issue date-time stamped access requests to see digital objects in either current or historical states.
- Policy enforcement for any interface or behavior associated with a digital object. We plan to use XML-based policy specification. Policies will be stored inside digital objects. We will also have more general system-wide policies that will be stored within the repository. This builds on our Prism-funded policy work described in Section 5.

In the process of this work, Sandy Payette from the Prism Project has participated in the design of the Metadata Coding and Transmission Standard (METS), coordinated by the Library of Congress. The METS schema is a standard for encoding descriptive, administrative, and structural metadata regarding objects within a digital library. METS will play an important role in the production FEDORA and our work with the METS committee has led to a number of changes in the protocol that make it more applicable for complex digital object architectures.

2 Digital Preservation

2.1. Preservation Risk Management for Web Resources

The preservation research team has defined a basic framework for a preservation risk management program. The program will enable the development of retention plans Web resources and support monitoring of selected resources over which the organization may have varying degrees of control and/or ownership. A fully preservation risk management program will utilize automated and manual tools for monitoring and evaluation to identify and quantify risks; to implement appropriate and effective measures to prevent, mitigate, recover from damage to and loss of Web-based assets; and to support post-event remediation. The program must be scalable, extensible, and cost effective.

2.1.1 Web resource experiments

In stage 1 of developing the preservation risk management program, data gathering and characterization, the team has been carrying out experiments using the Mercator Web crawler, which is described more fully in Section 6. These activities correlate to risk identification and risk classification in classic risk management approaches. The results of increasingly refined crawls are being used to characterize the nature of the content and behavior of representative Web resources, including types of content, types and extent of markup languages use, update cycles, evidence of ownership, extent and nature of HTTP header use, types and features of behavioral elements. Analysis of the results identify combinations of elements in Web pages that are indicators of risk by the presence, absence, correctness, completeness or other status of those elements. The Mercator analyzers have been adapted to respond to specific queries about selected characteristics and features.

Based upon initial stage 1 results, the team is formulating simple risk declaration and detection scenarios in stage 2 development. These will be further developed into contextualized risk declaration and detection agents. Stages 2 and 3 equate to the risk assessment and risk analysis steps of risk management. In the final stage a growing base of risk detection and response scenarios will enable automated preservation policy enforcement. The results of all of these experiments and the tools to address perceived risks are being packaged into a prototype preservation risk management program.

This staged approach is being applied to the real-life need to monitor selected Web sites of political and non-profit organizations in Southeast Asia. This test case is entering the data gathering stage. The Asian Collections unit within CUL plans to create a digital archive for these Web sites. Addressing this need will allow the Prism research team to explore the requirements for the pathways between monitoring/evaluation and custody.

The team has also been formulating experiments using adaptations of a range of Web site creation and maintenance tools to assess the health of servers on which selected Web sites reside and identify appropriate measures to respond to risks at the server level. The experiments address administrative as well as technical aspects of server infrastructure, stability, maintenance and security.

2.1.2 Web resource analysis

The research team has developed an approach and a Web page profile for analyzing Web resources. The approach addresses four levels of context: a Web page as a stand-alone object (ignoring its hyperlinks), a Web page in local context (including the links to and from the page), a Web site as a semantically coherent set of linked Web pages, and a Web site as an entity in a broader technical and organizational context. The data gathering is based on quantifiable elements for levels 1 and 2, which look at pages individually, but analyzing subsets of a URL that are organizationally coherent units requires isolating organizational elements that may be used inconsistently within and between Web sites, may be descriptive information that needs to be interpreted to be identified as indicators of organizational ownership, or may be embedded in Web pages rather than resident in an identifiable locations within the HTTP header or meta tags of the page. The team is developing automated monitoring mechanisms based on the analysis of the data gathered that are informed by organizational as well as technical context. The approach supports building generalizable mechanisms from Web site-specific findings.

The team developed a Web page profile to capture the results of the data gathering Web crawls. The Web page profile is an evolving analytical tool not a proposed metadata schema, but the team will recommend a set of metadata for Web resources that address preservation requirements.

2.2. Digital Information Longevity Study

The longevity study is using quantitative and qualitative methods to evaluate a sample of online electronic journals that were included in the *Directory of Electronic Journals, Newsletters and Academic Discussion Lists*, published by the Association of Research Libraries (ARL) from 1991-97. The longevity study is looking at technological change, as well as organizational, administrative, and economic issues that may have had affected the persistence of the resources. The study is identifying the nature and causes of loss when possible and the characteristics that may have enabled their survival. For example, the impact of the shift from FTP and Gopher to Web sites and the current shift from HTML to various pure and hybrid forms of XML have implications for Internet resources of enduring value. Richard Entlich is the lead researcher for the Longevity Study. His approach to characterization and analysis of loss is described below.

2.2.1 Sample resource set

The universe of resources being analyzed consists of electronic journal titles included in the seven-year set of the *Directory of Electronic Journals, Newsletters and Academic Discussion Lists*, published by the Association of Research Libraries (ARL). From 1991-97, this annual directory chronicled the growth of electronic publishing from the early days of ASCII publications distributed via e-mail through the appearance of mainstream, scholarly journals on the World Wide Web.

The longevity study is using the original machine-readable files, generously provided by ARL, to analyze the fate of this diverse group of publications. We have consolidated seven years worth of files into a single database, eliminated duplicates and other anomalous data, and focused on the subset of titles classified by ARL as 'journals' rather than 'newsletters,' 'magazines' or 'zines.' Though ARL's categorization was somewhat lacking in consistency (for example, titles classified as journals were not necessarily scholarly or peer-reviewed), it still provides a reasonable basis for sampling, leaving a collection of about 1800 unique titles. These titles run the gamut from fairly obscure and iconoclastic self-publishing efforts to widely read and distributed commercial titles.

2.2.2 Analysis

The entire universe of titles is being analyzed to produce a profile of electronic publishing during the pivotal period of change in the size, sophistication, and usability of the Internet. Some profiles will come directly from the data reported in the directories, such as the means of distribution and file formats employed. This analysis will build on extend work already conducted by ARL (see http://dsej.arl.org/dsej/2000/mogge.html) . We are also examining the current status of sites previously used for storage and dissemination of journal content, using a high-speed Web-crawler to test a normalized set of URLs derived from the directory listings.

A smaller subset is being examined in much more depth in order to determine whether the journal content can still be found on the Internet, or in any other form. These titles, taken from a range of years within the published history of the ARL directories, will be profiled in great detail. In addition to documenting various levels of loss (from complete disappearance to minor encoding problems), we are also broadly characterizing each publication and looking for characteristics that seem to correlate with either increased or decreased vulnerability to loss. Though most of this data collection requires interpretation and is not suitable for automated collection, portions of each profile, such as complete enumeration of link status, use of authoring and programming tools, and MIME types is being carried out using a Web crawler.

An even smaller subset of titles will be subject to a further stage of analysis, adding richness to their profiles through interviews with publishers, editors and technology staff originally involved in their creation and dissemination. These interviews will attempt to better understand institutional practices, and the organizational, economic or political factors that may have influenced the preservation status of the title and which are not discernable through passive examination of existing Internet sites.

Additionally, we will attempt to assess not only the degree and causes of loss, but user perception of the value and impact of completely and partially lost content. What is the significance of the material that's been lost? How serious an obstacle to use and interpretation are problems that result in some loss of fidelity to the original presentation?

Finally we will examine the role and fate of some of the efforts that arose during the early to mid-1990s to bring some order to the chaos of early electronic publishing. What role did initiatives such as CICNet (an aggregator of electronic journals) and attempts by individual libraries to collect and catalog electronic journals have on their long-term availability? What lessons can be learned from their success or failure?

2.2.3 Outcomes

The longevity study will add to our understanding of the life-cycle of electronic publications and the factors that influence the survival of their content. Though the focus on electronic journals may at first seem limiting, the titles chosen for analysis cover a wide range of issues and presentation styles, and incorporate a variety of textual, audio and pictorial content. They also come from a particularly volatile period of Internet development, and should shed some light on the impact of rapid technological change on the survival of machine-readable content. Most significantly, however, we expect to begin quantifying both the extent and importance of information loss and move beyond what has primarily been an anecdotal and speculative enterprise to one based on a more formal analysis.

2.3. Project Harvest

As part of the planning year for Project Harvest co-funded by a grant from The Andrew W. Mellon Foundation electronic journal archiving initiative, Anne R. Kenney and Nancy Y. McGovern have developed a Subject-Based Digital Archives (SBDA) model that will be further defined in a report to be published by CLIR in the first half of 2002. The model explores hybrid access and funding models for establishing a sustainable digital archive that maximizes the benefits of aggregation, which is at the heart of the subject-based approach. The report will also contrast the SBDA and Publisher-Based Digital Archives (PBDA) models, discuss multi-tiered alternatives for certification programs for digital archives that have implications for the preservation risk management program being developed in Prism, and considers the requirements for establishing an integrated matrix of digital archives that would provide appropriate redundancy, not mere replication, to insure the longevity of digital archives content. The matrix, especially the mechanisms for integrating certification protocols and the monitoring and evaluation tools that would be needed, are also directly relevant to the formulation of the preservation risk management program in Prism. CUL will be submitting a proposal to The Andrew W. Mellon Foundation for Phase 2 of Project Harvest, which will explore and extend these key components of the SBDA model.

2.4. CUL Central Depository Project

The Prism team led a library-wide effort to develop recommendations for establishing a central depository for preserving Cornell's digital image collections. Their report, *Establishing a Central Depository for Preserving Digital Image Collections - Part I: Responsibilities of Transferee*, calls for the establishment of a central depository within the library's Digital Library and Information Technology (D-LIT) infrastructure for ensuring continuing access to digital image collections over time. Such a central responsibility is seen as essential to ensuring a cost-effective preservation capability. The report outlines the requirements for deposit (including content and usability, legal considerations, conversion, formats, storage, and metadata) and identifies the role and major responsibilities of a central depository. The report defined the starting point for Part II of the Central Depository project, which is defining the roles and responsibilities of the central depository. The Part I report was submitted to the Library Management

Team in March 2001 and is available on the Prism Web site. The Part II report will be completed in Spring 2002.

2.5. Mathematical Models for Preservation

As described in the previous year's report, we began work on developing a digital preservation model that is information centric rather than bit centric. During this year, this work was reported in a paper at the European Digital Library Conference (ECDL 2001).

The ECDL paper describes the initial results of our efforts towards understanding digital (as well as traditional) preservation problems from first principles. Our approach is to use the language of mathematics to formalize the concepts that are relevant to preservation. Our theory of "preservation spaces" draws upon ideas from logic and programming language semantics to describe the relationship between concrete objects and their information contents. We also draw on game theory to show how objects change over time as a result of uncontrollable environment effects and directed preservation actions. We then how to use the mathematics of universal algebra as a language for objects whose information content depends on many components. We use this language to describe both migration and emulation strategies for digital preservation.

3 Human-Centered Research

3.1. Search Behavior

We have completed data collection and we are currently analyzing it for differences between levels of expertise, feedback, and gender. In addition, we have begun to broaden our research to include a younger population, namely school-aged children. This work will be quite applicable to the NSDL testbed, which we desribe in Section 7. The pilot work has begun on this and we intend to develop and test an image-based search engine designed to improve search efficacy in young children.

3.2. Tracking Behavior Online

The results of our tracking behavior experiments were just recently presented and published in the proceedings of the annual CSCL conference in Boulder. In addition, we have developed an algorithm to scrub the URL data to glean only those really meaningful visits to different pages. This allows us to further our investigation of preferred Web sites across semesters and to quantify browsing behavior. We have tied these data in with our investigations of multitasking in the classroom, recently presented and published in the proceeding of HICSS. In addition, we expect to glue Internet and email use with data we have been collecting on student's physical location on campus. Last semester we developed a program that visually represents an individual's location on campus and their transitions between different access points in the network. We are interested in the correlation and interaction between computing behavior and location and how this changes as a function of time. Finally, preferred URL's will be subjected to deeper scrutiny for the specific dimensions that may be responsible for developing that preference. This past year we purchased an eye tracking system that will enable us to investigate what people are looking at, for how long, and to trace the scan path across pages of interest. This will fold into a larger program of research on affective computing.

3.3. Context Aware Computing

CampusAware is a campus-wide tour guide application using Palm Pilots to display information regarding various campus buildings and physical spaces. It allows the users to receive and contribute notes to the database, which subsequent users may then access, making this a truly interactive system. The evaluation of this system proved most successful and the results of this study were presented and published at CHI 2002. MUSE, is a project currently underway which provides museum visitors with information, graphics and audio via an IPAC wireless handheld. The interface has been designed, and the technology that enables the IPAC to receive information via the infrared port from a beacon mounted on the artifact has been designed and tested. The entire system will be tested with patrons during the spring of 2002.

4 Interoperability Architectures

Along with support of the current protocol, we initiated in third quarter of 2001 the work towards version 2 of OAI-PMH. We decided to use the same successful model employed during version 1 development. That is, we are undertaking the process within a closed but representative OAI technical committee (OAI-TC). The OAI-TC includes 15 international technical experts with considerable experience in the deployment of OAI.

From the beginning the mandate of the OAI-TC has been incremental change rather than significant increase in functionality. The OAI remains focused on low-barrier interoperability solutions.

The work of the OAI-TC has proceeded as follows:

- Joint identification of issues by all OAI-tech members, and creation of an abstract describing each issue;
- Categorization/filtering/explanation of issues during a joint conference call;
- Compilation of a white paper per issue by volunteering advocates of OAI-tech;
- Joint on-line discussion of each white paper;
- Proposal for resolution of the issue by the OAI Executive;
- Joint on-line discussion of the proposal from the OAI Executive;
- Sub-committee revision of the v1.x protocol document to include agreed upon changes
- Alpha release of the v2 protocol document on March 1, 2002 and initiation of the alpha testing period.

The goal is public release of the v2 protocol in May 2002.

OAI-PMH v2 addresses the following issues, based on experience with OAI-PMH 1x:

- ***Dates and times*** - Standardize on UTC for all dates and times in protocol requests ("from" and "until" arguments) and responses.
- ***Harvesting Granularity***- Allow all ISO8601 time granularities in dates and times in protocol requests. Allow a data provider to expose its support date/time granularity in the response to an Identity request.
- ***Flow control*** - Improve flow control by allowing the following optional attributes when a resumptionToken is return:
    - *retry-after* - a suggested wait time until the request should be resubmitted
    - *expiration* - the projected expiration of the resumptionToken
    - *count* - number of items in this response batch.
    - *fullsize* - total number of items across entire result set
    - *cursor* - index of first item in this batch within entire result set
- ***set orthogonality*** - It will be possible to specify an identifier as argument to the ListSets verb, permitting a data provider to inquire what to which sets an item belongs. Responses to ListRecords and GetRecord will return the sets to which each item belongs.
- ***base-URL*** - Insulate harvesters from proxy servers by mandating that the visible identity of the "handling server" in responses be that of a persistent "master", that may opaquely reflect requests to slaves.
- ***multiple languages*** - Augment the schema defining the mandatory DC metadata format to allow inclusion of the xml lang attribute (specifying the language of the metadata value).
- ***Dedupping*** - OAI version 2.x will define an optional "provenance" container that can be attached to metadata records that a data provider aggregates from other sources. This will help harvesters detect duplicates harvested from multiple data providers.
- ***Error Handling*** - OAI 1.x tried to use HTTP status codes to report OAI protocol errors and exceptions. The result was not machine understandable and often idiosyncratic. OAI 2.x will include distinct OAI exception and error codes delivered as protocol responses, with defined (via schema) controlled vocabularies.
- ***Collection and Set Descriptions*** - OAI 1.x permits a payload in the Identity request for extensible collection description. OAI 2.x will recommend a basic collection metadata schema that can populate this payload. OAI 1.x includes sets but no way to describe the sets. OAI 2.x will include mechanisms to further describe individual sets. OAI 2.x will also make it possible, perhaps through the use of XML schema and the URLs for those scheme, for multiple data providers to specify that they share identical set structures.

5 Policy Expression and Enforcement

5.1. Policy Enforcement for Digital Objects:

Our previous work in using In-line Reference Monitoring (IRMs) as a strategy for enforcing such policies has continued to yield success in this current year. We have fully integrated Cornell's PoET software into the Fedora reference implementation at Cornell. We have demonstrated the enforcement of policies like the above examples upon our testbed of courseware objects, as well as other policies on a range of digital object types (e.g., image objects, text objects). We have continued support for our hypothesis that policy enforcement should be modular and extensible. First, policies should be completely tailored to meet the needs of specific items, without over-burdening a system-wide mechanism with idiosyncratic policy rules. Second, objects should be extensible in their functionality, and policies must be extensible too to reflect new or changed behaviors in the objects.

5.2. Stateful Policy Enforcement for Multi-Tiered Applications:

This year we have created a multi-tiered web application that prototypes a distance education courseware delivery system. The prototype has two main tiers that pertain to the policy enforcement problem. At tier 1 there is a web server with a Java servlet managing the main functions of the distance education process (registration, payement, course access, assignment submission). The servlet communicates via IIOP with a second tier that is a Fedora repository housing course objects and mechanisms to support access to course information. Using this prototype we have identified the following metrics for evaluating state management strategies for policy enforcement:

- Does the solution keep the Trusted Computing Base small? This means that we don't have to trust many modules and many subsystems to know that our policy enforcement scheme is secure.
- Does the solution provide assurance that application state variables are secure?
- Is the solution easy to implement by application developers (ground-up design)?
- Is the solution easily retrofitted into existing applications?
- It the solution easy to implement by policy writers?
- Does the solution have a low risk in terms modifying target applications in unanticipated ways? This is especially significant when applications are modified *dynamically* (e.g., IRM) and complex state management routines are added to applications.
- Is the solution flexible and extensible in terms of allowing for new application state definitions or changes in applications or policies over time?
- Does it provide a model whereby policies can be easily managed and kept track of? As the number of policy enforcement scenarios increases, will policies get unwieldy? Does it support modularity of policies?

We are currently searching for a post-doctoral associate with sufficient security experience to examine these issues.

5.3. A Logic for Policy Specification

By the beginning of last year, we had collected sample policies from both the digital and the traditional library communities. From these, we had inferred what types of policies are interesting to librarians. We then designed and implemented a basic prototype for expressing and reasoning about them. More specifically, our software allowed users to enter facts about their environment (e.g., Alice is head librarian from Jan 1, 2000 to Jan.

1, 2002), to enter policies (e.g. if you are a library patron, then you are allowed to download any object from the general collection.), and to ask if a particular user is explicitly allowed or forbidden from doing a specific action at a given time. As previously stated, the answer was obtained by translating the given environment and policies into formulas in first-order logic and applying model checking techniques to reason about the formulas.

Over the last year, the prototype's interface was re-done to meet three goals. First, we wanted to improve usability. To do this, we made the interface as standard as possible. This allows new users to rely on intuitions gained from other applications when running ours. We also added a tutorial. Our second goal was to better capture the dynamic, distributed nature of digital libraries. The organization of the first prototype suggested that there was a database corresponding to the environment. The new one implies that parts of the environment may come at different times from various sources such as a database, certificates presented by a user requesting to do an action, or even a third party. Finally, we wanted to mirror Fedora's interface as much as possible. This emphasizes the fact that we are doing a project that has several parts instead of several projects.

In addition to designing and implementing the new interface, we wanted to rigorously prove that we gave the right answer when someone asked if a particular action was permitted or forbidden. Because our reasoning is based on a language with formal semantics (a language based on a fragment of first-order logic), we could analyze our approach formally. We discovered that despite the simplicity and intuitive correctness of our procedures, they were not quite right. Without relying on our formal foundation, it is highly unlikely that this problem would have been noticed in a timely fashion. Having found the trouble during development, we have spent much of last year searching for a solution. More specifically, we have been examining more restrictive languages that can still capture the interesting policies (those inferred from our collection) and for which we can answer the question efficiently and in a provably correct manner. .

During the past year an abstract describing the initial stages of this work was accepted for submission to the Workshop on Information Technology and Systems.

6 Automatic Collection Building

Starting in August, 2001, we had access to Mercator, a fast web crawler located in Palo Alto, CA. It was developed at the Systems Research Center under Digital, and received further development under Compaq. Written in Java, Mercator is extensible and therefore a perfect vehicle for research into bulk collection building.

Putting together online collections of topic-specific educational material is usually done by hand. However, this does not scale at the same rate as the web itself, which is growing exponentially. The goal we set for the Fall 2001 semester was to generate, automatically, a number of collections on topics in science and mathematics. We define a collection as a set of HTML, PDF, and Postscript page on a given topic.

Search engines, while useful for finding a page related to a topic, even an authoritative page and/or a hub, are not useful for building collections because they crawl the entire web on all subjects and therefore cover relatively little of any specific area, e.g. science and mathematics.

We decided to use a new technology called "focused crawling" to find sets of URLs of documents related to specific subjects. To implement a focused crawl, we used both content analysis and link analysis. As HTML pages are downloaded, their words are extracted to build a weighted term vector which is then matched (cosine correlation, Salton 1968) against the term vectors representing each of our topic areas.

The document is tentatively classed with the nearest subject vector, with the correlation (0.0-1.0) being the degree to which the document is considered to be in that collection. If the correlation is sufficiently high, then links from that HTML page are also downloaded. If the link is to a .pdf or .ps file, those are also added to the same collection. All this logic was easily coded into Mercator by extending some of its base classes and adjusting parameters in its configuration file.

We have built 26 collections for topics in mathematics. An inspection of one of these, "plane geometry", was made by visually checking each page in the collection to determine whether the page belonged there or not.With relevance judgements in hand, it is possible to measure the precision of the collection by dividing the number of relevant documents by the size of the collection. Even more helpful, one can plot the number of relevant documents versus document rank, where the rank is in decreasing correlation order. (Collections are built from the most top-ranking documents.)

For a collection of about 50 documents about plane geometry, the precision was about 50%. Making the collection smaller by keeping only the highest ranking 20 documents, the precision rose to slightly over 60%. This was by no means a large crawl either, only 8 minutes for all 26 topics in mathematics. In comparison, the Google Search engine achieved 100% on its first 6 search results (using the plane geometry subject vector as the search query), but fell to about 45% after 20 results, and continued to decline thereafter.

The general success of this implementation of focused crawling depends heavily on the initial subject descriptors. More work must be done in this area. We found that using the index term list from MathForum worked well for developing math subject descriptors, but that a curriculum outline for 1st and 2nd grade science failed because it contained too many broad and general terms, which caused the crawl to have no focus. Here is an example of each:

- *MathForum index*: Basic Algebra Graphing Equations
- *Curriculum outline*: Second Grade Cycles in Nature The seasons change as the earth orbits the sun.

Note that our work is different from projects which go to the web and extract topical collections (e.g. Yahoo's human-based collections). Our goal in building collections for automated digital libraries is to specify the desired collection topics up front. It is true that using our software a collection could be build on-the-fly and thus look like a search engine, but this would be quite inefficient as it would still involve an extensive crawl. By doing several topics at once, we can be efficient AND get better results than a search engine.

Plans for continuing work on collection synthesis include making more runs to determine the relative impact of starting the crawl at various locations on the Web, raising or lowering the threshold for an "on-topic correlation", varying the number of "bad" links we'll follow until finding the next good paper, adding some machine learning to the correlation calculation, and generating Dublin Core metadata to describe the collections.

A paper on this research has been submitted to JCDL 2002 .

7 NSDL Testbed

We submitted a paper to the Joint Digital Library Conference describing the NSDL core architecture.