Convergence of Knowledge Management and E-Learning: the GetSmart Experience

Byron Marshall, Yiwen Zhang,
Hsinchun Chen, Ann Lally
*Department of Management
Information Systems.
The University of Arizona*
byronm@eller.arizona.edu

Rao Shen, Edward Fox
*Department of Computer
Science
Virginia Polytechnic and State
University*
rshen@vt.edu

Lillian N. Cassel
*Department of Computing
Sciences
Villanova University*
Lillian.cassel@villanova.edu

Abstract

The National Science Digital Library (NSDL), launched in December 2002, is emerging as a center of innovation in digital libraries as applied to education. As a part of this extensive project, the GetSmart system was created to apply knowledge management techniques in a learning environment. The design of the system is based on an analysis of learning theory and the information search process. Its key notion is the integration of search tools and curriculum support with concept mapping. More than 100 students at the University of Arizona and Virginia Tech used the system in the fall of 2002. A database of more than one thousand student-prepared concept maps has been collected with more than forty thousand relationships expressed in semantic, graphical, node-link representations. Preliminary analysis of the collected data is revealing interesting knowledge representation patterns.

1. Introduction

The National Science Foundation (NSF) through the National Science Digital Library (NSDL -- www.nsdl.org) intends to create a place where students and educators will find exceptional science education resources. Exemplary resources and services have been and continue to be developed and interconnected to meet this goal. The NSDL is built on a solid base of digital library technology research. While accessing quality resources and employing good technology are important, a learning tool should be also be crafted to support knowledge construction.

According to one definition, "digital library" is not merely equivalent to a digitized collection with information management tools, but also involves a series of activities that brings together collections, services, and people in support of the full life cycle of creation, dissemination, use, and preservation of data, information, and knowledge [20]. GetSmart was developed as an integration of knowledge management (KM) and digital library techniques shaped by constructivist learning theory.

2. Literature review

In this section, we present a few learning models which describe a learner-centered information search process and identify concept mapping as a helpful learning technique. Several types of E-Learning tools are characterized in light of that information search process. We finish with a review of knowledge management ideas which are applicable to the learning process as implemented in the GetSmart system.

2.1. Learning theory

Several learning models were considered as GetSmart was designed and developed. In contrast with logical positivism, which emphasizes observed truths and quantitative data, the constructivist model of learning centers on the learner's knowledge construction. Based on the constructivist model of learning, concept mapping has been identified as an effective learning tool. The GetSmart system encourages meaningful learning by supporting the information search process based on a constructivist outlook.

Ausubel created a learning theory model and evaluated the elements of the teaching-learning environment [1]. He used the term "meaningful learning" and provided significant theoretical support for constructivism. Rote learning is contrasted with meaningful learning and reception learning is contrasted with autonomous discovery learning. In rote learning, facts such as multiplication tables are presented to the learner in a final

form and internalized. In autonomous learning, information must be discovered before it can be meaningfully incorporated into the student's cognitive structure. Ausubel outlined and analyzed various learning factors. His work pointed to the need for new approaches to learning. Many of these ideas are embodied in the constructivist model which emphasizes three main ideas [6]. Firstly, there is no single "correct" representation of knowledge. Secondly, people learn through active exploration when exploration uncovers inconsistency between experience and current understanding. Finally, learning occurs within a social context.

The constructivist model has been used to support concept mapping as a learning technique and is used to describe a six-stage information search process. Novak and Gowin synthesized key ideas from Ausubel's work and applied them to concept maps [17]. Having learners construct concept maps moves away from a fact-focused teaching event towards a meaningful learning experience focused on the relationships between knowledge elements as understood by the learner. Novak points out that concept mapping supports meaningful learning, which is different from the rote learning found in most school situations. Kuhlthau applies a constructivist approach to information-age learning environments, identifying six stages in the information search process: initiation, selection, exploration, formulation, collection and presentation [14,13].

2.2. E-Learning tools

In this section we review three types of e-learning tools: curriculum tools, digital library tools and knowledge representation tools. Recalling Kuhlthau's six-stage information search process, we could generally say that each type of tool emphasizes different parts of the process. Curriculum tools provide a systematic and standard environment to support classroom learning; their functions are particularly helpful in the initiation and selection stages. Digital library tools facilitate effective and efficient access to resources to support exploration and collection while knowledge representation tools focus on formulation and representation.

2.2.1. Curriculum tools. Curriculum tools are widely used in high school and college education. Materials are selected and organized to facilitate class activities. Additional tools, such as discussion forums and online quizzes, are integrated to support collaboration and evaluation. A typical commercial curriculum tool includes 3 integrated parts: instructional tools, administration tools, and student tools. Instructional tools include curriculum design and online quizzes with automated grading. Administration tools include file management,

authentication, and authorization. Student tool functions include:

- Browsing class material: readings, assignments, projects, other resources
- Collaboration and sharing: asynchronous and synchronous bulletin boards and discussion forums
- Learning progress scheduling and tracking: assignment reminders and submission, personal calendars, and activity logs
- Self-testing and evaluation: tests designed by instructors to evaluate student performance

WebCT and Blackboard are the most popular commercial curriculum tools. A review comparing these two tools suggests that BlackBoard's flexible content management and group work support [2] make it more suitable for independent and collaborative learning. WebCT's tighter structure and fully embedded support tools make it more appropriate for guided, less independent learning. In general, these tools are tailored more to support class activities than independent research or self-study.

2.2.2. Digital library. While curriculum tools support class functions, digital library tools focus on locating resources. These functions support the exploration and collection phases of information search. Digital library tools help users find the right information amidst a huge amount of digital material. Digital library features usually include search, browsing, and discovering special collections or exhibits. Search and browsing are used to locate resources and explore related topics. Special collections or exhibits contain organized materials representing a unique treasure for interested users.

The NSDL is a digital library of exemplary resource collections and services, organized in support of science education at all levels. Starting with a partnership of NSDL-funded projects, NSDL is emerging as a center of innovation in digital libraries as applied to education, and a community center for groups focused on digital-library-enabled science education. The NSDL includes more than one hundred projects, each with a different focus. Project activities include exemplary collection creation, service development, and core integration. NSDL collections cover a wide range of topics including astronomy, biology, economics, mathematics, and technology. In December 2002 the NSDL core-integration team connected many of these collections in a combined search interface.

In addition to the NSDL, other digital library research continues to address learner needs. For example, the University of Michigan Digital Library (UMDL) project is a learner-oriented initiative. One recent UMDL study recognized that even well-intentioned, kid-oriented search engines have difficulties related to learning support

because they: (1) return too many hits; (2) provide no place to store results; and (3) provide no thesaurus [21]. Many well-designed digital libraries address these issues. One example is CITIDEL (www.citidel.org) which records user search activity.

2.2.3. Knowledge representation – concept maps. Another type of learning tool helps learners visually review, capture, or develop knowledge. Curriculum tools rely primarily on a text-based, syllabus approach to describing course content. This approach often fails to delineate the relationship of concepts and skills covered in one course to those covered in another. It also fails to show the knowledge base that a learner will have acquired at the end of his/her course of study. A visualization tool can engage both learners and instructors in an active learning process when they construct spatial-semantic displays of the knowledge, concepts, and skills that the learner possesses and acquires [19]. Concept mapping is one such knowledge visualization tool. Concept maps represent concepts and relationships as node-link diagrams.

Concept maps and other forms of spatial-semantic displays have "evolved as an alternative to traditional linear presentations of information, and as the basis of effective learning strategies" [5]. Chmielewski and Dansereau found that training participants in knowledge mapping helps those people process text even when a mapping strategy is not explicitly used. Literature evaluating the usefulness of spatial-semantic displays shows them to be effective in cooperative interactions, as study aids, as substitutes for traditional text, and for updating and editing knowledge.

CMap and WebMap are two popular concept-mapping tools which have been documented in a number of publications. CMap was developed by the IHMC (Institute of Human and Machine Cognition) at the University of West Florida. Users can share concept maps through the Internet using CMap's synchronous communication component [11]. IHMC researchers also combine CMap with case-based reasoning to support knowledge access, reuse, and capture [3]. WebMap was developed at the Knowledge Science Institute at the University of Calgary [9]. Gaines & Shaw proposed that concept maps be regarded as basic components of any hypermedia system, complementing text and images with formal and semi-formal active diagrams. They have investigated a number of concept mapping applications including education, artificial intelligence, active documents, hypermedia indexing and concurrent engineering [10].

Vocabulary overlap and link naming impact the construction and use of concept maps to represent knowledge. Research suggests that different people choose the same word to represent the same object less than twenty percent of the time [8]. Some concept mapping research uses closed lists of concepts [16]. Most of the current concept mapping tools allow users to choose their own concept and link names [18]. However, the benefit of using a closed link system, in which users select from a list of link names, has been discussed in the literature on knowledge representation, learning evaluation, and critical thinking [12, 15]. There is a growing consensus among researchers that links should be named, modifiable, directional, and represented by canonical sets.

2.3. The knowledge management process

In some ways, educational systems are like knowledge management (KM) systems; both involve the creation of useful knowledge from information or data found in available resources. Knowledge management is a "system and managerial approach to collecting, processing and organizing enterprise-specific knowledge assets" [4]. Accenture views knowledge management functions as a six-step process: (1) acquire, (2) create, (3) synthesize, (4) share, (5) use to achieve organizational goals, and (6) establish an environment conducive to knowledge sharing. Ernst and Young promotes a 4-phase KM approach: (1) knowledge generation, (2) knowledge representation, (3) knowledge codification, and (4) knowledge application. Not surprisingly, there are some similarities between Kuhlthau's information search process and these KM models.

Both top-down and bottom-up approaches are used in KM systems. For example, search engines are often characterized as either directories or indexes. Directory-based portals such as Yahoo are developed top-down by people who manually categorize information into a pre-defined structure. A user navigates the structure to identify relevant resources. In contrast, indexed-based portals like Google fetch and index the words found in documents on the web. This bottom-up approach tends to let the documents speak for themselves.

3. The GetSmart system

The GetSmart system was built based on a model of how communities and individuals create and share knowledge. Several design goals were identified and a system was developed; digital library tools are integrated with concept mapping and classroom support tools to produce an environment where students can construct representations of the knowledge they acquire. From a KM perspective, GetSmart is a system for the generation, codification and representation of knowledge or, in Accenture's nomenclature, a system to acquire, create,

synthesize, and share knowledge. The system design includes both top-down and bottom–up elements.

3.1. Knowledge development in GetSmart

GetSmart is organized to help individuals, groups, and communities develop knowledge. Curriculum tools provide a context for individual and group learning. As users construct concept maps they explore available information and then synthesize selected ideas into personal knowledge representations. This implements the "learn by exploring" constructivist idea. When group maps are created, several users collaborate, clarifying concepts and relationships and fitting them together. The search and curriculum functions access repositories of community knowledge. These repositories tend to be more formal and to use established vocabulary. The search tools help knowledge travel as information to the user/learners. As information is transferred to the individual it becomes enriched, expanded, and synthesized into new or unique contexts. These processes are viewed as information flowing from experts and repositories to individuals and groups. When a body of maps has been created, the information flow could be reversed. Map searching and merging tools would help capture and share individual knowledge representations in a community context. Much of the future work envisioned for GetSmart would support this process. This two-way flow embodies the top-down / bottom-up

character of the knowledge management process.

3.2. Architecture and System Description

In this section, we present the architectural design of GetSmart discussing the major components and technical issues. GetSmart was designed to meet several goals:
- Educational Goals
  - support effective learning strategies
  - ease system deployment and training
  - support student evaluation
- Digital Library Design Goals
  - provide content from a flexible platform
  - integrate search and map drawing
  - support access to external resources
  - support generation of new knowledge from external resources

3.2.1. Platform requirements. The GetSmart system is browser-based so that learners can access it from a typical university computer lab. JSP and Java Servlet technologies are used in the middleware and run on a web server. Messages are passed between client and server in an XML format to enhance modularity and support future interfaces for concept map sharing. Much of the program logic for message parsing and authentication is written into stored procedures in the database. GetSmart uses Microsoft SQL Server 8.0. The map drawing tool is a java applet developed using Java 1.4.
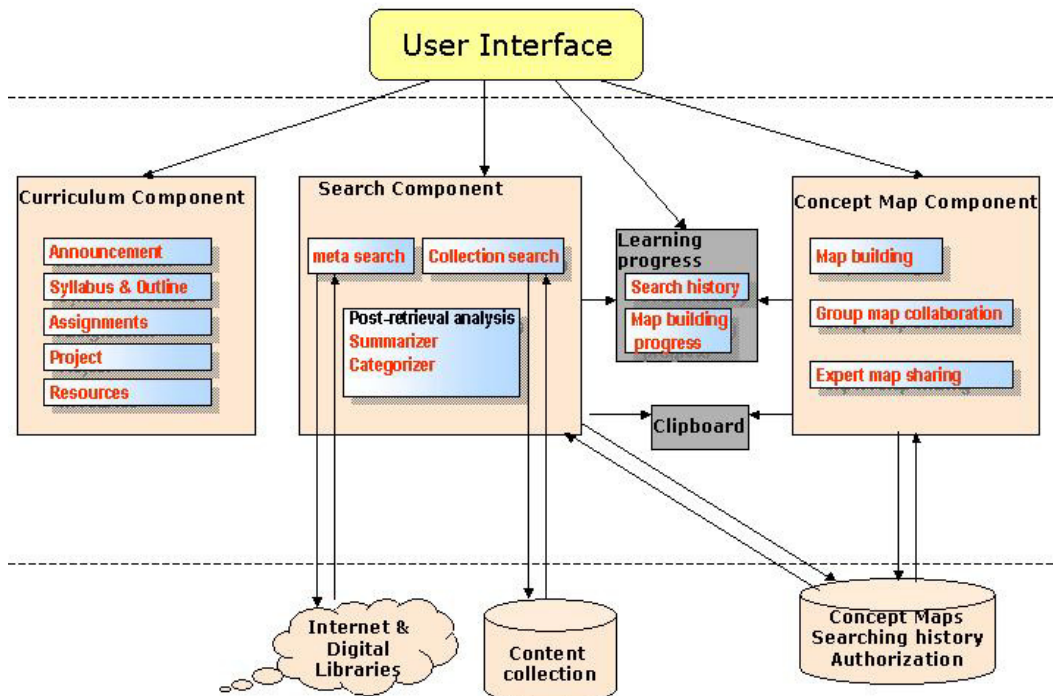

Figure 1. GetSmart architecture

3.2.2 Major components. The GetSmart user interface consists of four major components (1) curriculum, (2) search, (3) concept map, and (4) learning progress. These components are represented in Figure 1. The concept map data, user authorizations, and search histories are stored in a database. Another database contains the course resources collection index. Figure 2 shows the user interface. The navigation bar at the top of the interface allows users to switch among the system components -- "Class Info", "Concept Maps", "Search", and "Learning Progress". In addition a row of index-tab buttons highlight available functions for the main window.

The Curriculum Component: A curriculum component provides access to official class information including announcements, the syllabus, a course outline, assignment information, project information, and class resources. These categories are similar to categories used in WebCT and BlackBoard.

The Search Component: The searching component is based on a metasearch framework and supports queries and post-retrieval analysis. Metasearch modules handle query requests and results for several existing search portals.

The first searchable resource listed is labeled "Class Resources". An index was built in advance including instructor-selected material. The selected resources were fetched and indexed. A metasearch module for this collection works with GetSmart's metasearch interface allowing a single search to select results from both the instructor-identified, highly-relevant sources and from outside sources.

Metasearch modules have been created to access a short list of other search engines and digital libraries. Users can select the search engines they want to use for each of their queries. Currently, Altavista, eBizPort, CITIDEL and ACM Digital Library have been implemented as targets for metasearch. Altavista was selected as a general search engine to provide GetSmart a global searching scope. ebizPort is a business and IT information portal. Two digital library portals related to the fall courses were selected. CITIDEL is a part of the NSDL and aims to serve the computing education community. CITIDEL accesses more than one hundred thousand unique resources from member collections. The ACM Digital Library references documents published by the ACM, including journals, conference proceedings, and forums. These resources are not hosted by the GetSmart system; instead, GetSmart accesses the individual search interfaces of the selected portals and compiles the results.

Users can select the resources they want to access and choose the number of results to be listed on each result page. Figure 3 presents several screen shots of query and analysis results. Query results are displayed in a results
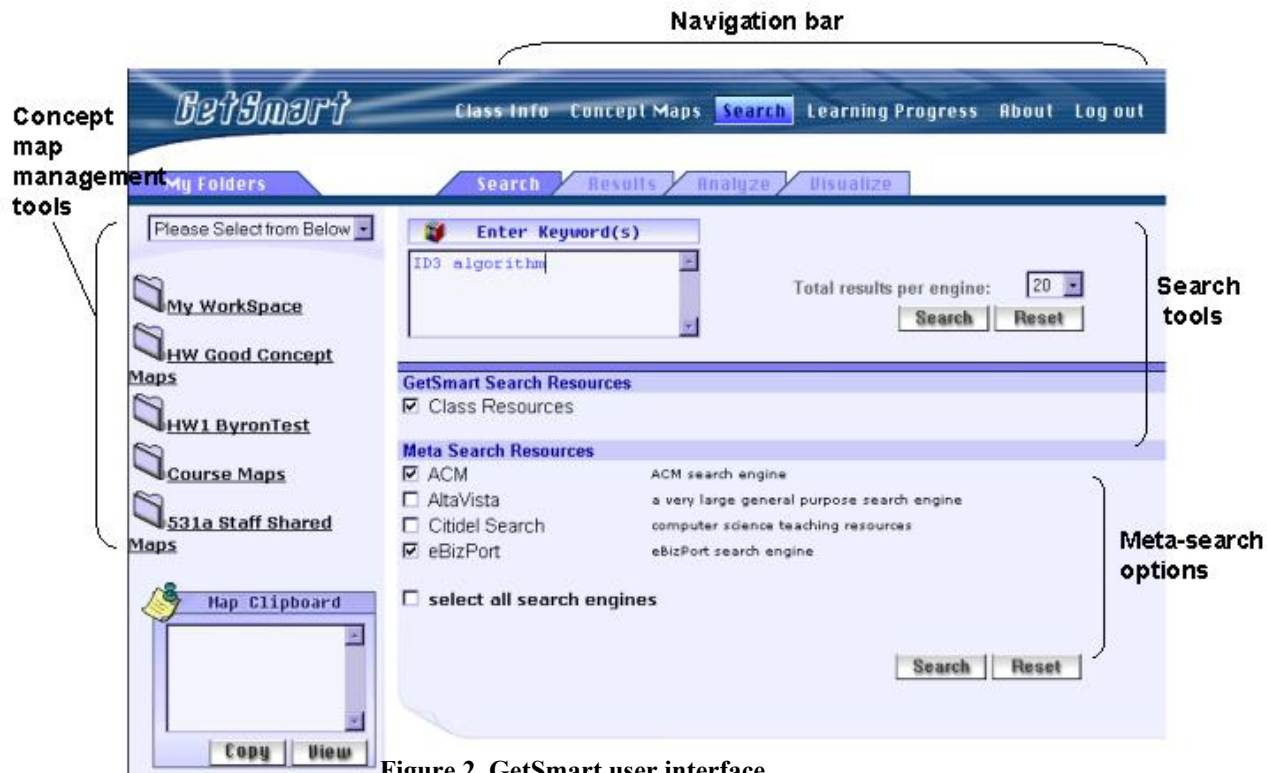

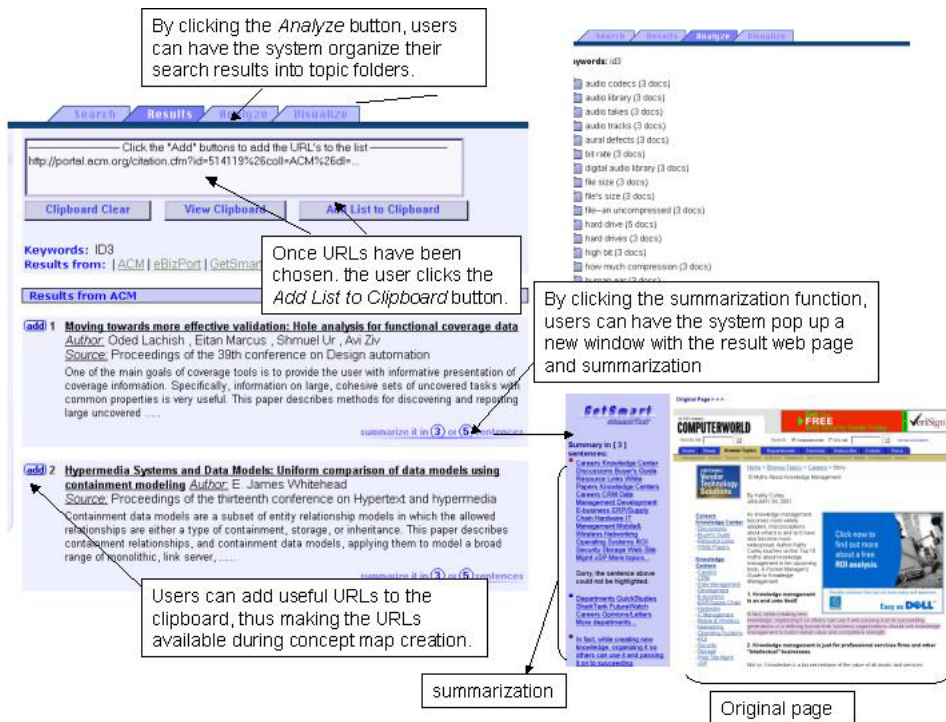**Figure 2. GetSmart user interface**

Figure 3. Results page with analysis functions

page with controls to initiate post-retrieval analysis functions. The query results are grouped by source. Users can select and add URLs they consider important to the clipboard, making the URLs available to the concept mapping applet. Each returned page can be summarized and a set of results can be categorized using the post-retrieval processing functions.

The Concept Map Component: The concept mapping component consists of a set of concept map management functions and a concept map building applet.

The concept map management panel is displayed on the left hand side of the interface. Concept maps are organized into folders; users can create different folders to organize the concept maps. Expert and group concept maps are authorized by the system administrator, are displayed in the same folder structure, and are identified with special icons. Three types of concept map operations can be performed using the commands in the pull-down menu.

(1) folder operations: create, delete, and rename a folder
(2) map operations: create, delete, and rename a map
(3) advanced map operations:

- "Turn in a map" submits maps to the instructor.
- "Print a map" requests generation of a web page showing a map image, and a listing of nodes, links and attached resources.

- "Import/export a map" allows XML representations of a concept map to be imported or exported.

After a map has been created and assigned to a folder using the management tools, it is updated using the concept map building applet. Clicking on a folder in the concept map management panel expands the display to list the titles of concept maps stored in the folder. Clicking on the title of a concept map activates a new Java applet window like the one shown in Figure 4.

The applet has three windows. The bigger window on the right is the main window used for graphical manipulation of the concept map elements. Small icons on the nodes indicate resources have been associated with the node. These resources include URLs, notes, or other maps. Users can click pull-down menu options to "Save", "Print", or "Turn In" a map. The procedure for adding links and nodes to a map is depicted in Figure 5. It also shows how resources can be associated with nodes.

The upper left window is the map resource window. It displays all the concept map elements hierarchically. The lower left window is the clipboard window. The map clipboard allows users to paste notes from other sources for inclusion in the concept map. The client side security restrictions may require the use of this feature if direct copy and paste is restricted for applets.

When several users are authorized to update a concept map, additional functions are needed. All authorized users
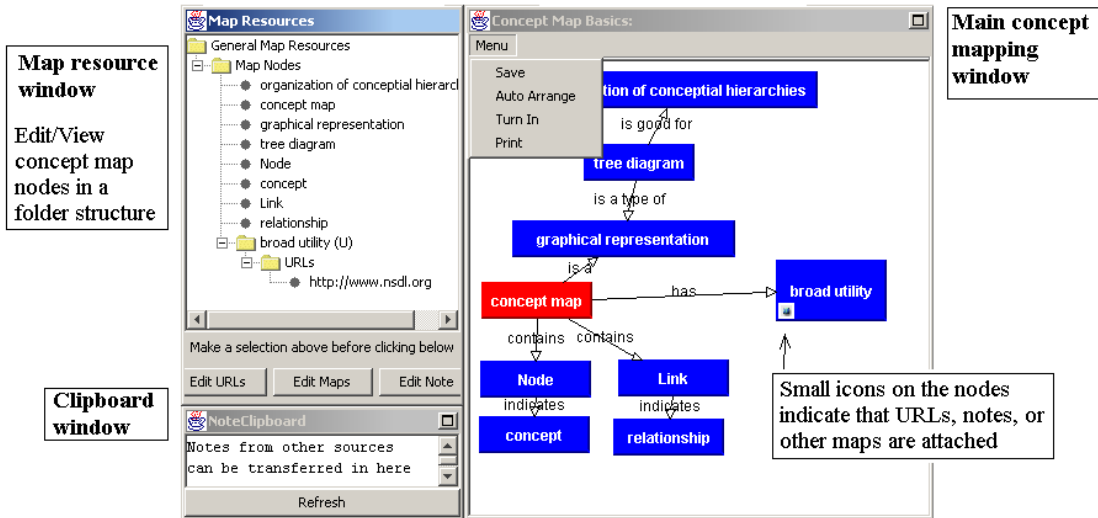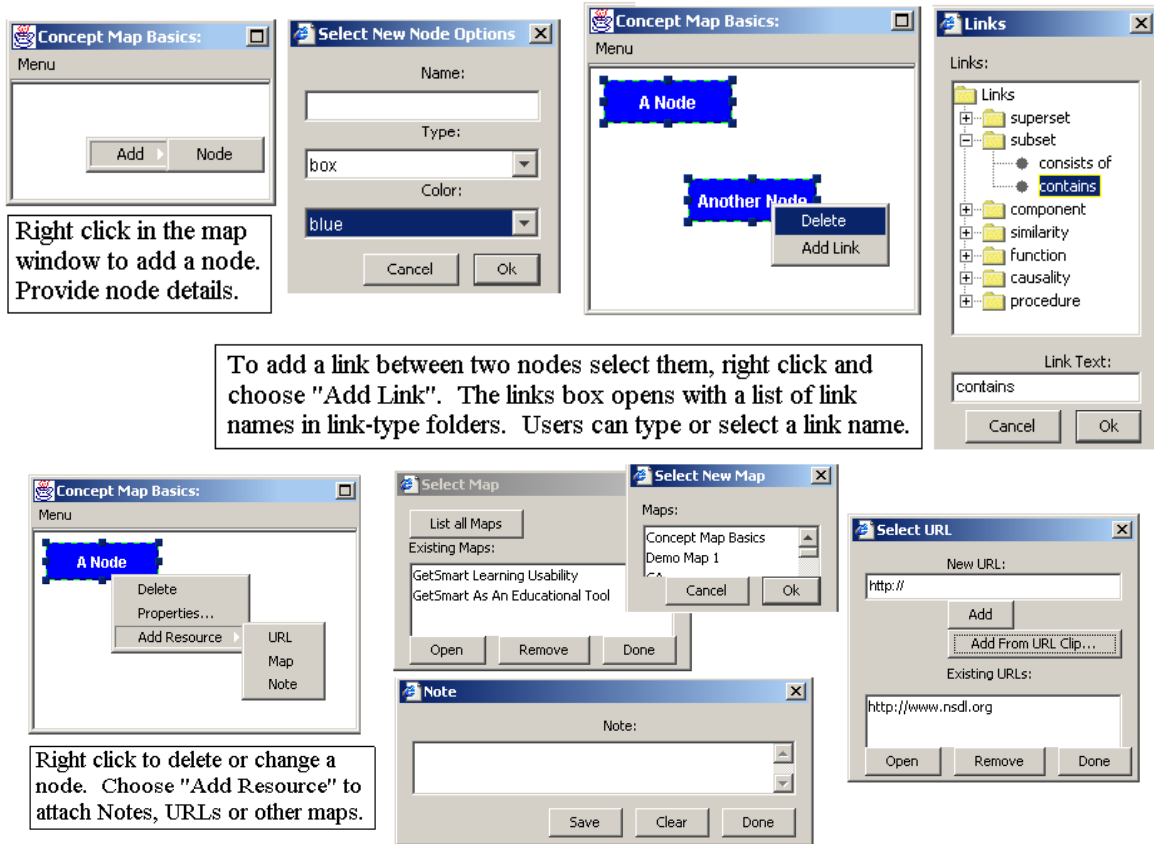
Figure 4. Concept map building applet

Map resource window

Edit/View concept map nodes in a folder structure

Clipboard window

Map Resources
General Map Resources
Map Nodes
organization of conceptial hierarch
concept map
graphical representation
tree diagram
Node
concept
Link
relationship
broad utility (U)
URLs
http://www.nsdl.org

Make a selection above before clicking below

Edit URLs | Edit Maps | Edit Note

NoteClipboard
Notes from other sources can be transferred in here
Refresh

Main concept mapping window

Concept Map Basics:
Menu
Save
Auto Arrange
Turn In
Print

tion of conceptial hierarchies
is good for
tree diagram
is a type of
graphical representation
is a
concept map   has   broad utility
contains contains
Node   Link
indicates   indicates
concept   relationship

Small icons on the nodes indicate that URLs, notes, or other maps are attached

Figure 5. Node and link functions

Concept Map Basics:
Menu
Add   Node

Right click in the map window to add a node. Provide node details.

Select New Node Options
Name:
Type:
box
Color:
blue
Cancel   Ok

Concept Map Basics:
Menu
A Node
Another Node
Delete
Add Link

Links
Links:
Links
superset
subset
consists of
contains
component
similarity
function
causality
procedure
Link Text:
contains
Cancel   Ok

To add a link between two nodes select them, right click and choose "Add Link". The links box opens with a list of link names in link-type folders. Users can type or select a link name.

Concept Map Basics:
Menu
A Node
Delete
Properties...
Add Resource   URL
Map
Note

Right click to delete or change a node. Choose "Add Resource" to attach Notes, URLs or other maps.

Select Map
List all Maps
Existing Maps:
GetSmart Learning Usability
GetSmart As An Educational Tool
Open   Remove   Done

Select New Map
Maps:
Concept Map Basics
Demo Map 1
Cancel   Ok

Note
Note:
Save   Clear   Done

Select URL
New URL:
http://
Add
Add From URL Clip...
Existing URLs:
http://www.nsdl.org
Open   Remove   Done

can view a map at any time. Users with ownership authority for a map can lock the map for update, but only one user can lock the map at any given time.

The Learning Progress Component: The learning progress component allows users to review their concept map and search history. Users can view mapping activity such as the last action performed, when a map was turned

in, the number of maps created, and the number of nodes, links, and resources. Search history shows the last 10 searches performed and the results returned.

4. Preliminary results and lessons learned

Although a rich data set has been collected, only preliminary evaluation of the data has been accomplished. Some usage statistics are available. Early evaluation suggests interesting possibilities for a study of how different students represent the same information. Section 4.1 briefly describes overall system usage. Section 4.2 reviews experiences using GetSmart at the U of A. Section 4.3 documents a study done at Virginia Tech using the GetSmart system.

4.1. System usage

The GetSmart system was used at the University of Arizona and at Virginia Tech in graduate-level courses in the fall of 2002. Dr. Edward Fox taught "Information Storage and Retrieval" (CS5604) at Virginia Tech with 60 students creating and accessing concept maps. Dr. Hsinchun Chen taught "Data Structures and Algorithms" (MIS531A) to 54 management information students at the University of Arizona. Students at both universities prepared a series of concept maps. The MIS531A students each created 3 sets of maps as homework assignments and worked in groups to map the main points of their group presentations. The CS5604 students, singly and in groups, created concept maps of the material in each chapter covered in the course. These maps were presented and reviewed during class meetings.

When homework assignments were due the system was accessed twenty four hours a day by students as they completed their assignments. Table 1 lists some overall usage statistics.

Table 1. Overall usage: MIS531A and CS5604

114	Student Users
4,000 +	User Sessions
1,400 +	Homework and Presentation Maps
600 +	Searches Performed
50 +	Group Maps
40,000 +	Relationships Mapped

4.2. U of A concept mapping experiences

Students were assigned to create concept maps covering material presented over several weeks of classes. Maps were scored for "coverage", "correctness", and "creative connections". Approximately 15 key concepts were identified for each assignment. Each map was checked for the inclusion of those concepts. Students were encouraged to specify relationships specifically. The number of unique link names was used as a proxy for correctness. The maps were reviewed for inclusion of cross-links between hierarchical sections of each map, and for the number of attached resources. Only a moderate level of accomplishment was required to achieve an "A" grade, for example, any recognized expression of approximately three quarters of the concepts was given full credit. Students were given remediation for any difficulties they encountered including extra time and/or compensatory credit if they experienced any difficulties with the system.

Five groups of students in MIS531A created shared maps for their class presentations. We found that on the average 2, and as many as 5, of the 6 students in each group saved changes to each group map. Almost all of the group members opened each of the maps multiple times.

All 54 students in Arizona completed all three of the homework mapping assignments even though they accounted for a total of only five percent of the final class grade. It seems that the system was reasonably usable and the group maps were prepared with some collaboration. Table 2 shows selected map characteristics for the three U of A homework assignments (HW1, HW2, and HW3). While the experiences depicted in Table 2 represent light analysis rather than experimental results, the figures suggest that, over time, students created more complex maps and added information to the maps in less time.

Table 2. MIS531A homework map characteristics

	Maps	Nodes Per Map	Edges Per Map	Elements Per Minute
HW1	2.9	16.9	18.6	0.35
HW2	5.4	22.3	23.8	0.51
HW3	5.2	27.0	27.9	0.75

4.2.1. Map characteristics. A preliminary analysis of the characteristics of the concept maps created is leading to some interesting observations on how users represent their knowledge.

Visual elements: Color and arrangement of nodes in student maps demonstrated a variety of representational schemes. Most of the users used 3-5 colors in each map. Some students used color to indicate the clusters of concepts while others used it to highlight hierarchical relationships. Closely related concepts were usually drawn close to each other. More general concepts tended to be in upper or central sections of the map, while more specific concepts were usually found in the lower or peripheral positions.

Concept vocabulary: For key concepts in one set of maps, we found that sixty percent of the students used the

same word or phrase to represent the same concept. This calculation was made after allowing for spelling variances, abbreviations, and compound names. For example, a time complexity of O(n2) was represented 13 different ways and the concept "bubble sort" was found in maps with an abbreviation (bs), several formats (like "bubblesort", "bubble sorting" and "bubble algorithm"), and 4 different misspellings. On the average each concept was represented by eight different words or phrases. This is a much higher level of vocabulary overlap than was found in [8]. However, the maps charted a focused topic, and the material was taken from lecture notes.

Link names: Link names are one important way concept map creators express their knowledge of conceptual relationships. The GetSmart interface suggests a set of categorized link names but the user can override them and enter any words they choose. Our users chose the system provided names less than one-third of the time. A preliminary set of heuristics was used to group the student-entered names into semantically-similar categories based on cue words or phrases in the link names. Some preliminary work has been done to categorize these links further into a short list of abstract link types. Examples of these link types include hierarchical, componential, comparative and procedural.

Our initial analysis of the links in the map collection found the following patterns:
(1) 39,000 analyzed links contained more than 5,300 distinct link names. These links could be clustered into 120 semantically similar categories.
(2) More than half of the links indicate hierarchical or componential relationships although the link-type distribution varies between topics. For example, a higher percentage of hierarchical and componential links were used in topics covered in class lectures based on well-defined concepts. A higher percentage of other link types was found in maps covering material learned by reading course materials or maps generated from open-ended material.

4.3. Results from Virginia Tech

This section reports on a study of GetSmart use at Virginia Tech. In January 2003 we began running GetSmart servers at Virginia Tech for two sections of our spring class "Multimedia, Hyertext, and Information Access" (CS4624), but earlier, in 2002 all use of GetSmart was hosted at the University of Arizona.

4.3.1. GetSmart use and study design. In the graduate class "Information Storage and Retrieval" (CS5604) in the fall of 2002, each student prepared a number of concept maps. Some were to help with documenting their group term projects, but most were related to the textbook

"Modern Information Retrieval". Each student completed a concept map, using GetSmart from home, for each chapter assigned. Instead of listening to a lecture on each of these chapters, students spent a full period discussing and presenting group-produced concept maps. To make this logistically feasible, the class met in two sections, each with about 30 students. Each section had two meetings per week, one in a classroom with an overhead projector and screen, and one in a computer lab where any student display could be shared across the room. This allowed comparison of concept maps drawn by hand using color pens and transparencies, versus maps created in the lab using GetSmart.

When students met to discuss a textbook chapter, each had previously prepared and submitted for grading their individual map. This ensured that they stayed on schedule and kept current with the reading schedule. Then, in class, they joined with 3-5 others to discuss their maps and to prepare a group map. During the first half of the course all of the group maps covered the entire chapter. During the second half of the semester, the chapter was divided into 3-4 parts, and two groups focused on each of the parts, so that a more detailed map could be constructed.

Each group presented its map to the class as a whole to receive comments and suggestions from classmates and the instructor. This helped the group members identify misconceptions and reinforce understanding.

Another class requirement was for each student, sometime during the semester, to pass (with a score of 80% or higher) an automatically graded online quiz on each chapter in the textbook. If their first attempt was unsuccessful, they were encouraged to study more, and then to take a second version of the quiz. Since mastery was the objective, a third attempt was allowed. More than 90% of the students preferred to prepare a concept map using GetSmart before taking a quiz. The average number of times that students took to pass a quiz in fall 2002 was 1.25. In the same class a year earlier, before GetSmart was developed, the average number of times that students took to pass a quiz was 1.55. That is, without the GetSmart assignments, students took more quizzes to master a chapter.

4.3.2. Analysis. Our first analysis was a survey, at mid-term. The original data file (www.csc.villanova.edu /~cassel/vtsurveyresults/midtermoriginal.data) is available online. Most of the students thought GetSmart was well designed and helpful, though there were complaints about access to and use of GetSmart (which had its first test in August 2002).

Our second analysis was a comparison of quiz grades, contrasting students in the fall 2002 with those who took the same course in fall 2001. Results are summarized in Figure 6. Our hypothesis, that the mean quiz score in fall

2002 is significantly greater than that in fall 2001, is accepted (using a t test with $\alpha=0.25$).
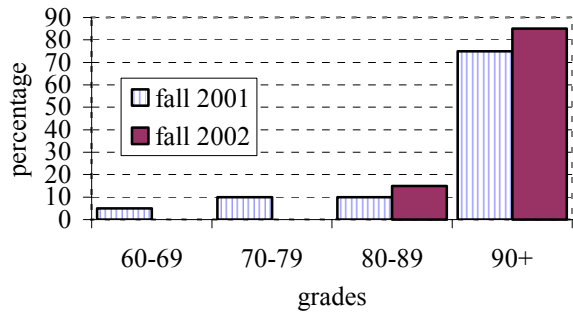

Figure 6. Quiz grades in course CS5604

Our third analysis compared the group maps created by GetSmart with the ones created manually. We found that maps created with GetSmart contain more cross-links. Since cross-links may reflect deeper thought regarding relationships among key concepts, we found this result encouraging. We think GetSmart enables much easier production and modification of concept maps while the pen and transparency approach required more time and effort when drawing and revising maps.

Our fourth analysis concerned concept maps created to support term project work. Project group members shared a folder in GetSmart, which allowed collaboration. To facilitate the project evaluation, students were required to present project maps in the final project demo. Study of the students' project maps suggests that some groups that performed better in collaborative concept mapping produced some remarkably good concept maps and were highly successful in their term projects.

Our fifth analysis compared concept maps developed for project documentation with those developed to aid learning about the chapters. We found that the latter shared more common characteristics regarding structure, while the former had different types of hierarchical structures. Some project maps documented the division of responsibility among group members. Some highlighted the succession of tasks carried out over the course of the project. Further, more URLs and notes were attached to the concept nodes in project maps than to those in chapter maps.

Our final analysis dealt with characteristics of maps related to the textbook chapters. We compared maps created by individuals with maps created by groups (see Table 3). We also compared maps created early in the semester regarding the whole chapter, with maps created later in the semester regarding only a part of the chapter. Our hypothesis that the average number of nodes and links in a group map are significantly greater than those in an individual map is accepted by statistical analysis (t

test with $\alpha=0.01$). We also noted some qualitative differences, suggesting that maps improved over the course of the semester. Early in the term, key concepts in the introduction and in most of the section titles of a chapter appeared in almost every student's map. Later in the term, students expressed concepts more in their own terms, apparently thinking more deeply about ideas and relationships.

Table 3. Concept map characteristics

	Nodes per Map	Links per Map
Individual	22-23	23-24
Group	29-30	31-32

5. Discussion

Most of the features of the CMap, WebMap, and GetSmart mapping tools are comparable. GetSmart has some features which distinguish it from the others: a familiar right-click interface to access many of its functions, integrated search tools, a browser-based architecture, and a folder-view listing of each node's properties. CMap uses undirected links; GetSmart uses one-directional links; and WebMap allows single-directional and bi-directional links. WebMap implements a scripting function to support the transfer of information from WebMap concept maps to KRS, a knowledge representation system. GetSmart uses an XML format for import, export, and internal communication. CMap and GetSmart export maps as images. Unlike the others, CMap has an embedded synchronous chat function. GetSmart allows several users to share a map while WebMap allows users to make alternative versions of other users' maps. WebMap records a node type for each node, e.g., "Report", "Picture", "Movie", and "Document." The visual representation (color, font, and shape) of each node is controlled by user-editable node type definitions. GetSmart and CMap allow users to change the color of the individual nodes and associate resources with those nodes. GetSmart supports notes, maps, and URLs as resources; CMap allows maps, URLs, images, movies, sounds, and text resources. In WebMap each resource is represented by its own node. Overall, the CMap interface is more mature than the GetSmart interface, but the GetSmart system implements features which are promising as part of an integrated learning environment.

The initial implementation of the GetSmart system meets many of our educational design goals. Our first goal was to support effective learning strategies. Effective learning strategies are supported by the integration of curriculum tools, search tools, and concept mapping.

Combining concept mapping and search technologies is intended to leverage the cognitive benefits of concept mapping in a digital library context. The curriculum component supports the initiation and selection stages of the information search process. The search functions support exploration and collection of information, and the concept mapping tool aims to help students formulate and present the information.

Our second goal was to ease system deployment and training. Because GetSmart is browser-based no specialized software is needed on a client machine. Messages passed in XML over HTTP connections allow for the development of additional interfaces to the backend processes.

 The vocabulary and link analysis we are undertaking might lead to automated map scoring techniques that will help the system meet our third goal of supporting student evaluation.

Our digital library design goals also have been largely addressed. The modular, web-based implementation is reasonably flexible. The system ties together searching and mapping functions; however, the search tool was not extensively used (4,000 user sessions with only 600 searches run). Perhaps this is because class activities emphasized map drawing. User input suggests that students are familiar with their favorite search tool and did not see a need to use our search interface. The extensible metasearch framework nicely supports our goal to support access to external resources.

Our last goal was to support the generation of new knowledge from external resources. The GetSmart system supports information flow from outside sources to users in the provision of expert concept maps and in the connection to various search portals. The concept mapping interface can facilitate a precise and compact representation of acquired knowledge. We expect that analysis of the student-created maps will lead to procedures and algorithms for the evaluation, merging, and searching of information captured in concept maps.

Well designed digital libraries, with special services tailored to support teaching and learning, especially supporting visual learning, may help students to clarify their thinking, and to process, organize, and prioritize new information. Experience in MIS531a and a study of the use of GetSmart in the CS5604 course showed that concept maps may be of benefit in both individual and collaborative settings.

6. Future directions

Based on our success, we have extended GetSmart to make it XTM compliant as well as OAI-PMH compliant. We have also developed a standalone version. We hope to make GetSmart more portable by configuring it into an ODL [7] component.

The data collected includes at least 10 individual topic areas. Fifty or more students mapped each topic. This data should provide interesting insights into how students choose to represent acquired knowledge in concept maps. Vocabulary overlap will be analyzed and compared between groups of maps to help in the development of map merging algorithms. Several link alternatives will be explored including closed lists of links, hybrid interfaces where the user can specify a name but must also specify a type, and algorithmic categorization of the link names entered by students. Algorithms to compare maps to expert maps will be proposed to help in the scoring of maps and as a way of developing our knowledge development model.

7. Acknowledgements

8. References

[1] D. P. Ausubel, Educational Psychology: A cognitive view. New York: Rinehart and Winston, 1968.

[2] S. Bayne and J. Cook, "WebCT vs BlackBoard? An Evaluation of Two Virtual Learning Environments", http://www.ltss.bris.ac.uk/interact21/in21p04.htm,

[3] A. J. Canas, D. B. Leake, and D. C. Wilson, "Managing, Mapping and Manipulating Conceptual Knowledge: Exploring the Synergies of Knowledge Management & Case-Based Reasoning," presented at AAAI Workshop on Exploring Synergies of Knowledge Management and Case-based Reasoning, Menlo Park, CA, 1999.

[4] H. Chen, "Knowledge Management Systems: A Text Mining Approach", http://ai.bpa.arizona.edu/go/download/chenKMSi.pdf, 2001

[5] T. L. Chmielewski and D. F. Dansereau, "Enhancing the recall of text: Knowledge mapping training promotes implicit transfer," Journal of Educational Psychology, vol. 90, pp. 407-413, 1998.

[6] B. Dalgarno, "Interpretations of constructivism and consequences for computer assisted learning," British Journal of Educational Technology, vol. 32, pp. 183-94, 2001.

[7] Edward A. Fox, Hussein Suleman, Ming Luo: Building Digital Libraries Made Easy: Toward Open Digital Libraries. ICADL 2002: 14-24

[8] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais, "The Vocabulary Problem in Human-System Communication," Communications of the ACM, vol. 30, pp. 964-971, 1987.

[9] B. R. Gaines, "Class Library Implementation of an Open Architecture Knowledge Support System," International Journal of Human-Computer Studies, vol. 41, pp. 59-107, 1995.

[10] B. R. Gaines and M. L. G. Shaw, "Concept Maps as Hypermedia Components," International Journal of Human-Computer Studies, vol. 43, pp. 323-361, 1995.

[11] S. Hamilton, "Thinking Outside the Box at the IHMC," IEEE Computer Magazine, vol 34, pp. 61-71, 2001.

[12] H. E. Herl and et. al., "Reliability and validity of a computer-based knowledge mapping system to measure content understanding," Computers in Human Behavior, 1999.

[13] C. Kuhlthau, "Inside the Search Process: Information Seeking from the User's Perspective," Journal of the American Society for Information Science, vol. 42, pp. 361-371, 1991.

[14] C. Kuhlthau, "Learning in Digital Libraries: An Information Search Process Approach," Library Trends, vol. 45, pp. 708, 17p, 1997.

[15] J. G. Lambiotte, D. F. Dansereau, D. R. Cross, and S. B. Reynolds, "Multi-relational semantic maps," Educational Psychology Review, vol. 1, pp. 331-367, 1989.

[16] J. R. McClure, B. Sonak, and H. K. Suen, "Concept Map Assessment of Classroom Learning: Reliability, Validity and Logistical Practicality," Journal of Research in Science Teaching, vol. 36, pp. 475-492, 1999.

[17] J. Novak and D. B. Gowin, Learning How To Learn. Cambridge, UK: Cambridge University Press, 1984.

[18] J. M. Oughton and W. M. Reed, "The influence of learner differences on the construction of hypermedia concepts: a case study," Computers in Human Behavior, vol. 15, pp. 11-15, 1999.

[19] A. Saad and A.-R. M. Zaghloul, "A Knowledge Visualization Tool for Teaching and Learning Computer Engineering Knowledge, Concepts, and Skills," presented at 32nd ASEE/IEEE Frontiers in Education Conference, Boston, MA, 2002.

[20] N. Seamans and G. McMillan, "Digital Library Definition for DLI2", scholar.lib.vt.edu/DLI12/defineDL.html June 5, 1998

[21] E. Soloway, C. Norris, P. Blumfield, B. Fishman, J. Krajcik, and R. Marx, "K-12 and the Internet," Communications of the ACM, vol. 43, pp. 19-23, 2000.