Logfile analysis: A review of the literature

Tina Leard & Allyson Fiona Hadwin
Concordia University, Montreal, QC, Canada

DRAFT. Do not cite without permission of the authors.

Paper presented as part of the symposium, *Logfile Navigation Profiles and Analyses: Methods for Tracking and Examining Hypermedia Navigation*. A structured poster session at the American Educational Research Association Annual Meeting, April 2001, Seattle, WA.

Support for this research was provided by the Centre for the Study of Learning and Performance. *Authors can be contacted at allysonh@education.concordia.ca or tina_leard@education.concordia.ca.*

Hypermedia learning environments are popular for delivering instruction at all educational levels. These electronic environments allow information to be presented in non–linear formats and to be augmented with audio and visual components (Ambrose, 1991; Marchioni, 1988). Hypertext formats present content through a series of informational units called nodes. People can navigate non–linearly through content nodes by choosing and following links. Navigating through content pages on the worldwide web is the most salient example of a hypertext environment. The structural complexity of these environments varies from simple links forward and backward (much like a book), to complex branching networks that provide many choices as to which links are accessed and in what order nodes are traversed (Misanchuck & Schwier, 1991). While hypermedia designers determine the content boundaries and possible navigation paths, the learner/user controls the pace and navigation paths followed (Gall & Hannafin, 1994).

Hypermedia environments have great instructional appeal because they have potential to provide access to content and instruction without being bound by geographical location or time. Hypermedia environments are being developed for classroom instruction, industrial training, entertainment, news delivery, and accessing library resources to name only a few. As these environments become increasingly prevalent and sophisticated in design, it becomes important to investigate how students navigate and use hypermedia tools to enhance learning (MacGregor, 1999). Answering this question is central to usability testing and improving system design, as well as to understanding and evaluating the effectiveness of learners' studying patterns and strategies.

In addition to enhancing usability testing, tracking students' on–line navigation is a non–intrusive way to collect information for research about learners as they learn in hypermedia environments. Examination of learning as it evolves in the context of hypermedia instruction can be enhanced by collecting precise traces of student engagement with on–line materials (Barab, Bowdish, & Lawless, 1997; Rouet & Passerault, 1999; Winne, Gupta, & Nesbit, 1994). Other methods, such as students' self reports and interviews, require thinking about actions and thought processes in retrospect and, therefore, may be less accurate than directly tracing human computer interaction. Computer traces are different from observational data, which are constrained by time and human ability to attend to and record detail. They are powerful traces of activity because they are so comprehensive; however, it is their comprehensive nature that makes them very difficult to analyze, synthesize, and interpret. In contrast to traditional means of assessment, hypermedia–based logfile data documents the dynamic, situated nature of learning, as well as individual differences in activity (MacGregor, 1999, Marchionini, 1990; Winne, et al., 1994).

The analysis of logfile data is central to research on learning in hypermedia environments. Logfile data have potential to enhance the understanding of instructional design principles and of learners' cognitive processes. However, there are a range of challenges associated with the diversity in tools and language used for collecting and describing logfile data.

We began this review with the intent of uncovering a range of analytical techniques used for analyzing logfile data that had been collected in a previous study (Hadwin, Jamieson–Noel, McTavish, McNamara, & Winne, 2000). Within no time, it became apparent that a number of challenges confront researchers interested in collecting and analyzing this type of data. This review attempts to identify and clarify some of these challenges. It also uncovers salient issues that require coordination between programmers, designers, instructors, and researchers involved in developing, using, and evaluating hypermedia environments.

Our review unfolds in three parts. First, we describe our methods for accessing literature describing the analysis of logfile data. This section demonstrates that logfile data are widely used across an array of disciplines. It also identifies the need for developing a consistent language with which to describe logfile data that are currently labeled as audit trails, dribble files, logfiles, navigation trails, event recordings, and event traces. Second, we describe a range of tools used for collecting logfile data across hypermedia contexts. Although recent advances in technology have provided means with which to collect data about hypermedia navigation, many of these tools are not easily accessible and there is no industry norm for recording and saving detailed traces of hypertext navigation. For the most part, the tools we describe required additional programming and were custom developed for a given application. Essentially this means that new tools are developed each time navigation trace data are required. Therefore, the collection of logfile data is reserved for researchers and developers who have the foresight and funding to build these tools. Finally, our review overviews a range of analytical techniques used for organizing, classifying and interpreting logfile data. We discuss these techniques according to the complexity of the hypermedia systems being used. This section of the paper provides direction for researchers who are collecting logfile data and finding themselves limited to simple summary statistics that do not reflect the complexity of human computer interactions, usability, or cognitive strategies.

Method

Computer searches of the ERIC database were conducted for articles published between the years 1982 – 2000 using the following search terms: tracking (1777 hits), transaction log analysis (70 hits), audit trail (27 hits), log file (10 hits), and navigation patterns (4 hits). Search terms constituted words in an article's title and abstract. Combining these terms with the ERIC descriptor “computer–assisted instruction” and excluding hits with the search term “library” yielded 4 articles. The search term “transaction log analysis” yielded studies involving the evaluation of library catalogues through logfile analysis; these studies were eliminated from the literature search because they were not conducted in an educational context and they focused on general statistics of patrons’ interactions. Less than 1% of the hits overlapped, demonstrating the inconsistency of the terms used in the field for logfile analysis. As a result of this inconsistency, reference lists of articles uncovered through this literature search were examined; this method was more promising and produced 64 articles relevant to hypermedia learning environments and logfile analysis.

Inconsistency of Terminology

While reading the articles, we discovered a variety of terms to denote “logfile” including dribble file, event record, event traces, interaction history, history transcripts, and navigation trails. This accounted for some of our initial difficulties in locating articles using ERIC searches. The abundance of terms is reflective of the vast array of disciplines and contexts wherein logfiles are examined; however, it also points to the need for developing some cross–disciplinary discussion about the collection, analysis, and interpretation of logfile data.

Diversity of Logfile Use

While conducting the literature search, it became evident that logfile analysis was not unique to educational research. Besides the prevalence of logfile analysis in library research, several articles in the field of computer science that highlighted usability testing and cybernetic modeling of user behavior were located. Although the articles that discussed modeling user behavior (Fjeld, Schluep, & Rauterberg, 1998; Rauterberg, 1993) provided valuable ways of viewing human–computer interaction through state transition vectors and Petri net descriptions, studies in this area were eliminated from this review because they concentrated on systems other than hypermedia environments. Within the field of education, research applying logfile analysis focussed on either hypermedia environments or computer–supported collaborative learning (CSCL) environments. CSCL related articles (Meistad & Wasson, 2000; Nurmela, Lehtinen, & Palonen, 1999; Wasson, 1999) were also eliminated because this research focussed on learner–learner interaction rather than the learner–system interaction present in hypermedia related studies.

Lack of Illustrative Logfile Analysis

Our literature search unveiled more articles that discussed rather than applied logfile analysis. We collected 23 articles containing descriptions of how to analyze logfile data, but only 13 of these articles focused on the analysis of logfiles for educational investigations. Further, only 6 of the 23 logfile analysis reports illustrated the described analyses with actual data; the remaining articles only focused on the explanation of analyses. Analysis descriptions were also touched upon in the 18 articles that described tools; 8 of these articles highlighted tools for logfile analysis, 4 presented tools for logfile collection, and 6 proposed tools that both analyze and collect logfile data. Table 1 lists all articles besides empirical studies that report findings and whether the paper (a) reported descriptions on potential ways to analyze logfiles (usually a conference paper or technical report), (b) described a logfile analysis tool, or (c) described a logging tool.

Table 1. Classification of the contents of the 41 articles on logfile analysis that did not report findings from empirical studies.

Articles	Analysis Report	Analysis Tool	Logging Tool
1. Badre, Guzdial, Hudson, & Santos (1995)	•		
2. Barab, Fajen, Kulikowich, & Young (1996)		•	
3. Beasley (1992)			•
4. Beasley (1997)	•		
5. Becker, McLauglin, & Rebelsky (1999)	•		
6. Berger & Jones (1995)	•		
7. Canter, Rivers, & Storrs (1985)	•		

8. Chaver, Carrasco, Rossell, & Vega (1998)		•	
9. Chen, Horney, & Anderson–Inman (1995)	•		
10. de Vries, Netto, & Hanegraaf (1993)		•	
11. Drott (1998)			•
12. Fjeld, Schluep, & Rauterberg (1998)	•		
13. Gay & Mazur (1993)		•	•
14. Guzdial (1993)		•	
15. Guzdial (1995)	•		
16. Guzdial, Berger, Jones, Horney, Anderson–Inman, Winne, & Nesbit (1995)	•		
17. Guzdial, Santos, Badre, Hudson, & Gray (1994)	•		
18. Guzdial, Walton, Konemann, & Soloway (1993)	•		
19. Jones & Jones (1997)		•	
20. Lamain, Mulder, & Mulder (1994)	•		
21. MacLeod (1992)		•	•
22. Marchionini (1990)	•		
23. Marzo–Lázaro, Verdú–Carbó, & Fabregat–Gesa (1998)			•
24. Meistad (2000)		•	•
25. Meistad & Wasson (2000)		•	•
26. Misanchuk & Schwier (1991)	•		
27. Misanchuk & Schwier (1992)	•		
28. Niegemann (2000)		•	
29. Nurmela, Lehtinen, & Palonen (1999)	•		
30. Olson, Herbsleb, Rueter (1994)	•		
31. Okada & Asahi (1999)		•	•
32. Pitkow & Bharat (1994)		•	
33. Rauterberg (1993)	•		
34. Recker (1994)	•		
35. Rouet & Passerault (1999)	•		
36. Schwier & Misanchuk (1990)	•		
37. Wasson (1999)		•	•
38. Westerman, Hambly, Alder, Shryane, Crawshaw, & Hockey (1996)			•
39. Williams & Dodge (1993)	•		
40. Winne, Gupta, & Nesbit (1994)	•		
41. Xia, Liebscher, & Marchionini (1991)		•	

Out of the 64 articles collected, only 20 articles reported empirical studies that were conducted in an educational context and involved hypermedia–based logfile data; 13 of these articles were published in journals, 4 were published in conference proceedings, 2 were unpublished conference papers, and 1 was a published chapter. The remaining 3 empirical studies containing logfile data (Barab, Bowdish, & Lawless, 1997; Barab, Bowdish, Young, & Owen, 1996; Lai & Waugh, 1995) did not focus on the use of hypermedia as an educational tool, but rather as an informational tool for general reference. Table 2 lists articles that report findings from an empirical study and indicates whether the paper was (a) a journal article, (b) a conference paper, or (c) a chapter in a published book, and d) if the study was conducted in an educational context.

Table 2. Classification of the contents of the 23 articles on logfile analysis that reported findings from empirical studies. The 12 articles reviewed in the Analytical Techniques section are highlighted in gray.

Article	Journal Article	Conference Paper	Chapter	Educational Context
1. Andris (1996)	•			•
2. Barab, Bowdish, & Lawless (1997)	•			
3. Barab, Bowdish, Young, & Owen (1996)	•			
4. Beasley & Vila (1992)	•			•
5. Beasley & Waugh (1997)	•			•
6. Britt, Rouet, & Perfetti (1996)			•	•
7. Fitzgerald & Semrau (1997)		•		•
8. Fitzgerald & Semrau (1998)	•			•
9. Gomes (1997)		•		•
10. Hall, Balestra, & Davis (2000)		•		•
11. Horney (1993)	•			•
12. Horney & Anderson–Inman (1994)	•			•
13. Jih (1996)		•		•
14. Jones & Berger (1995)		•		•
15. Kelly & O'Donnell (1994)	•			•
16. Lai & Waugh (1995)	•			
17. Larsen, Kinzie, Boker, & Burch (1996)		•		•
18. Lawless & Kulikowich (1996)	•			•
19. Lawless & Kulikowich (1998)	•			•
20. Lickorish & Wright (1994)	•			•
21. Marchionini (1989)	•			•
22. Reed & Oughton (1997)	•			•
23. Schroeder & Grabowski (1995)	•			•

Challenges

The lack of illustrative logfile analysis, the diversity of logfile use, and the inconsistency in terminology pose challenges for researchers. These challenges are further complicated by confusing and often inadequate descriptions of the hypermedia environments, tracking tools, and analytical techniques used in these studies. Each of these issues needs to be addressed in order to improve interpretation and comparison of studies as well as to further research on logfile analysis. We explore some solutions to these problems in the Discussion and the Conclusion.

Logging Tools

Although technological advances have provided new means for observing learners/users as they interact with a hypermedia learning environment, many of these logging tools are not readily available to the research community. We uncovered three categories of logging tools in the literature: 1) customized logging tools, 2) standard web server logging tools, and 3) keylogging tools. As the use of hypermedia learning environments increases, standardized logging tools need to be developed, so researchers can collect traces of students' interaction with any application or website.

Customized Logging Tools

Most of the logging tools that we uncovered in the literature were tailored for specific hypermedia environments and required additional programming. Customized logging tools were used in 17 of the 23 studies located in our literature search. Extended programming of HyperCard's environment enabled customization of logfile data collection in 8 (e.g., Lai & Waugh, 1995; Reed & Oughton, 1997) of the 17 studies that used customized logging tools; additional programming to other systems besides HyperCard was applied in 9 studies (e.g., Fitzgerald & Semrau,1997; Jih, 1996; Schroeder & Grabowski, 1995).

Despite the scarcity of standardized logging tools, commercial hypermedia authoring tools, such as HyperCard and Toolbook, do offer the possibility of creating a logging function through their programming languages. During the early 1990s, several efforts were made to extend HyperCard's programming language in order to log user–computer interactions. Horney (1993) developed EntryWay, a hypertext authoring extension for HyperCard that records users' keyboard entries. McLeod (1992) created AutoMonitor, and Berger and Jones (1995) used Event Recorder; both of these tools capture events in HyperCard's environment and are capable of analyzing these events. More recently, Winne, Hadwin, McNamara, Chu, and Field (1998) developed a prototype electronic notebook CoNoteS2 using the authoring language STUDY, which is based on HyperCard's programming language; STUDY also makes possible the collection of logfile data. Other efforts to extend authoring tools besides HyperCard that do not contain a prepackaged function for capturing this type of data were made. Beasely (1992) extended IBM's hypermedia authoring language LinkWay while Gomes (1997) used Toolbook to capture events. Despite these advancements, the transferability of these logging tools is limited.

Web Server Logging Tools

Web environments provide a means to collect user–computer interactions that is more accessible to researchers than customized logging tools. Most standard web servers can collect users' interactions with a web environment. However, the type of data that is collected in web server logs rarely represents individual learners' interactions, which is fundamental to educational research (Becker, McLaughlin, & Rebelsky, 1999; Guzdial, Berger, Jones, Horney, Anderson–Inman, Winne, & Nesbit, 1995). Web servers usually only record general statistical information, such as site use distributed over time or counts of the number of times that each page was served. For this reason, it is difficult to determine what student is associated with which event.

Figure 1 illustrates the nature of a standard server log. Each line represents a request made by *a* user from a particular computer workstation. The information shown is related to the workstation (pcl.udg.es): date, time, the type of request (in this case, “GET”), and the URL requested (for instance, /~usd/Prog/Users/guest.tmp.html). The information presented in this standard logfile does not reveal whether requests made from the workstation were initiated by the same user.

Figure 1. Standard logfile contents.
(Marzo–Lázaro, Verdú–Carbó, & Fabregat–Gesa, 1998)

```
pc1.udg.es – – [22/Oct/1997:18:00:38] "GET /~usd/CS.CGI?2WuJ1uJ1uJdxwqaZSKaCaoM
HTTP/1.0" 200 511

pc1.udg.es – – [22/Oct/1997:18:00:38] "GET /~usd/Prog/Users/guest.tmp.html HTTP/1.0" 200
1743

pc1.udg.es – – [22/Oct/1997:18:00:39] "GET /~usd/CS.CGI?504J1uScfwnuA11Z3o1ZWZ7
HTTP/1.0" 200 1834
```

Marzo–Lázaro, Verdú–Carbó, Fabregat–Gesa (1998) propose a server program called CustomServer that aims at identifying individual users. Becker et al. (1999) propose a similar logging system called Project Clio. Both of these logging systems attach user identification information to each event as it leaves the user’s computer workstation and travels towards the web server, where events are logged in a database. This database can contain time stamped entries of the documents retrieved by the user as well as the user’s response to tests or exercises. Figure 2 illustrates custom logfile contents that are associated with the user through the user’s login. The process of delivering user identification information is achieved through Common Gateway Interface (CGI) scripting, which as its name suggests acts as a gateway of communication, or intermediary, between the user’s computer and the server. Although collecting user–computer interactions in web learning environments remains somewhat of a challenge, owing to the additional programming that is required so that communication between users’ computers and web servers can be furthered, there is definitely movement towards creating standard logging programs that track individual usage for standard web servers.

Figure 2. Custom logfile contents after login.
(Marzo–Lázaro, Verdú–Carbó, & Fabregat–Gesa, 1998).

Requested HTML–document	time in seconds
*Unidad: LES XARXES DE COMUNICACIONS I LA INFORMÀTICA	
teleinfo/index.html	9 (00:00:09)
teleinfo/repercusions/comunicacio.html	227 (00:03:47)
teleinfo/index.html	330 (00:05:30)

Keylogging Tools

An alternative to customizing logging systems programmed within an established authoring tool is to use a keylogging system that is able to capture events independent of any one application. Keylogging systems run in the background of other applications and act as an intermediary between the application being used and the operating system. Keylogging systems are able to capture keyboard entries and mouse events from the operating system and record these events. An example of this type of logging tool is Datalogger (Westerman, Hambly, Alder, Shryane, Crawshaw, & Hockey, 1996). Datalogger runs on Microsoft Windows and is able to capture every user event and replay these events within the application that they were generated. Okada and Asahi's (1999) GUITESTER is another example of a keylogging system that also runs on Windows. This particular system was designed to capture events that could be analyzed from a usability perspective. The tool generates usability analysis data from the logfiles and represents these analyses visually. As Meistad (2000) points out, Datalogger is able to generate data specific to particular users while GUITESTER's logfiles reveal information about a group of users. Further development and marketing of keylogging systems is fundamental in making logfile data collection more accessible to the general public and, in particular, enabling studies of electronic learning environments that do not contain a customized logging tool.

Coding the Complexity of the System

Early in our review, it became apparent that the hypermedia environments from which logfiles were drawn differed a great deal. Some were very simple systems allowing forward and backward linear movement, while others were very complex interactive environments. This became a concern because the complexity of the system seemed to have some bearing on the complexity of the logfiles that needed to be recorded. For example, when movement can only occur in a forward and backward linear progression, very little information about context is required in the logfile. It would be sufficient to provide a time stamp and indicate which event occurred. In contrast, complex hypermedia environments involving a great deal of interactivity require more sophisticated logging techniques. For instance, one may need to record the time, event, source of the event, and function used to create that event. In an attempt to better understand how logfile data have been analyzed, we decided to organize the analyses according to the kinds of systems wherein logfiles were generated.

Drawing from theories in systems and cybernetics, cognition and memory, and computer assisted instruction, we have compiled a means for coding the overall complexity of hypermedia learning systems. Since logfile analysis involves examining the interaction between the learner and the learning environment, we conceptualize a hypermedia learning "system" in terms of the learner and the hypermedia environment. We propose that each system differs in complexity depending upon: (a) the type of task students are assigned (Barab, Bowdish, & Lawless, 1997; Canter, Rivers, & Storrs, 1985; Marchionini, 1989), (b) the hypermedia structure (Beasley & Vila 1992; Britt, Rouet, & Perfetti, 1996; Misanchuk & Schwier, 1992), and (c) the interactivity afforded by the system's architecture (Nelson & Palumbo, 1992; Wright & Lickorish, 1994). These aspects interact to produce the overall complexity of a system and the "cognitive load" associated with that system.

Task Complexity

Task complexity refers to the concrete nature of the activity as defined within the learning environment. It ranges from well–defined tasks, such as “searching and retrieving information,” to undefined tasks where students explore the environment without any externally defined goals. We identified three levels of task complexity in studies that logged and examined student interactions with the software (see Table 3).

Level 1 tasks were simple and well defined. These were tasks in which students were asked to complete a search or to find the answer to a question. The learning episode had a definite completion point and a correct response. For example, in Marchionini’s (1989) study, one group of participants was asked to perform what Marchionini labeled a closed task: Participants were asked to find a fact – the first year speed skating was introduced into the Olympic games.

Level 2 tasks were ill defined. These tasks contained abstractly defined goals and no pre–defined response. Often students were asked to study or learn something using the software and tools provided. An example is Beasley and Waugh’s (1997) study where participants were required to “learn the material in the lesson to the best of their abilities.”

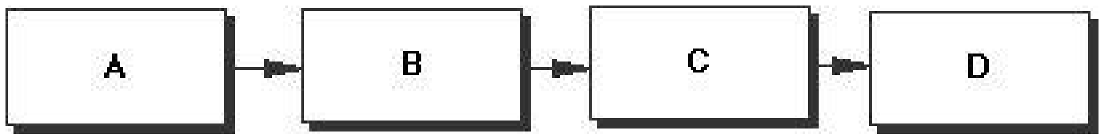
Level 3 tasks were undefined and abstract. We viewed these as the most complex tasks because students were presented with the software and tools and had to decide how they would interact with the software as well as to what end. Usually this meant exploring or navigating free of external goals or standards. For example, Horney (1993) identified general tasks for which a hypertext editing system called EntryWay might be useful, and participants decided how they would use the system. Three participants used EntryWay as a part of their data analysis procedures during qualitative research, another three participants created bibliographies, and one participant categorized a set of images for a hypermedia–based instructional unit.

Hypermedia Structure

The notion that hypermedia structures differ in complexity is not new. Misanchuck and Schwier (1991) suggest that the structure of the system becomes more complex when it moves from more linear structures to hierarchical or branching structures. Cognitive flexibility theory is based on the notion that hypermedia environments can be built to more suitably correspond to the natural complexity of content areas and subject matter (Spiro and Jehng, 1990). Hypermedia structures also define the complexity of logfiles with more complex structures revealing more elaborate logfile traces (Misanchuk & Schwier, 1991). An examination of logfile analyses revealed three levels of hypermedia structure consistent with those identified by others (Alessi & Trollip, 2001; Misanchuck & Schwier, 1992).

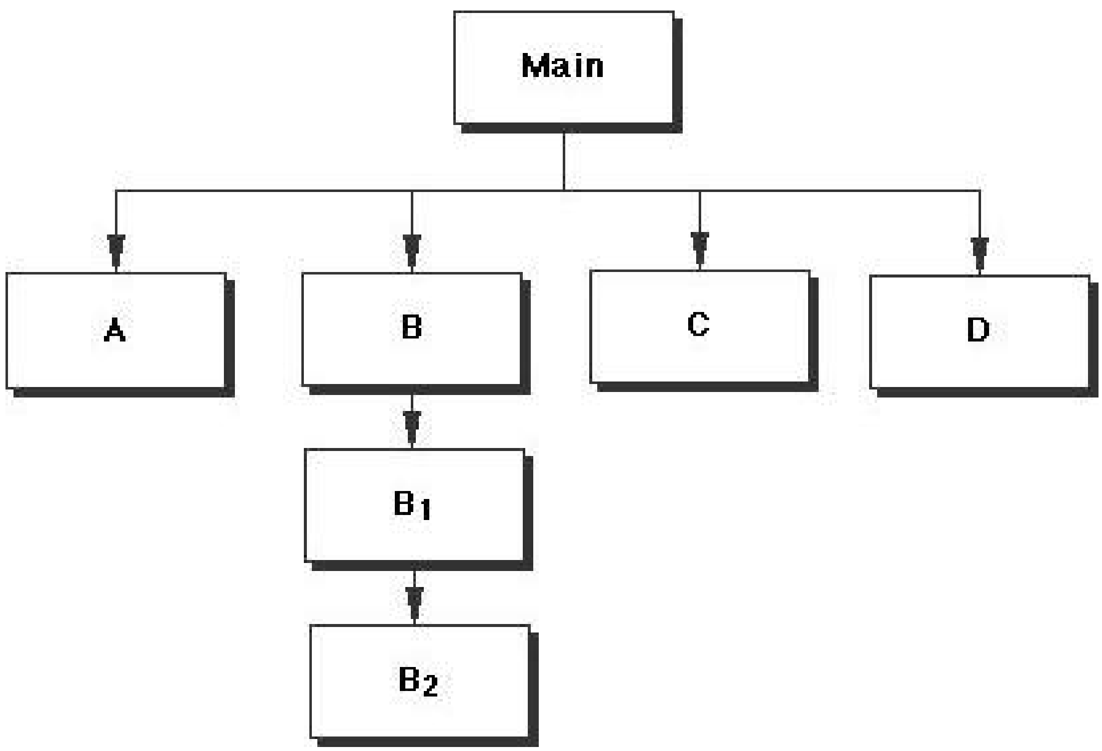
Simple structures (Level 1) include both linear and hierarchical structures. Linear structures allow for navigation both forward and backward along a string of connected pages (see Figure 3). They are very similar in structure to a book. Students can progress forward or backward, but there are no variations from that structure. A hypermedia program does not typically have the traditional organization of sequential pages.

Figure 3. Simple structure (Linear)



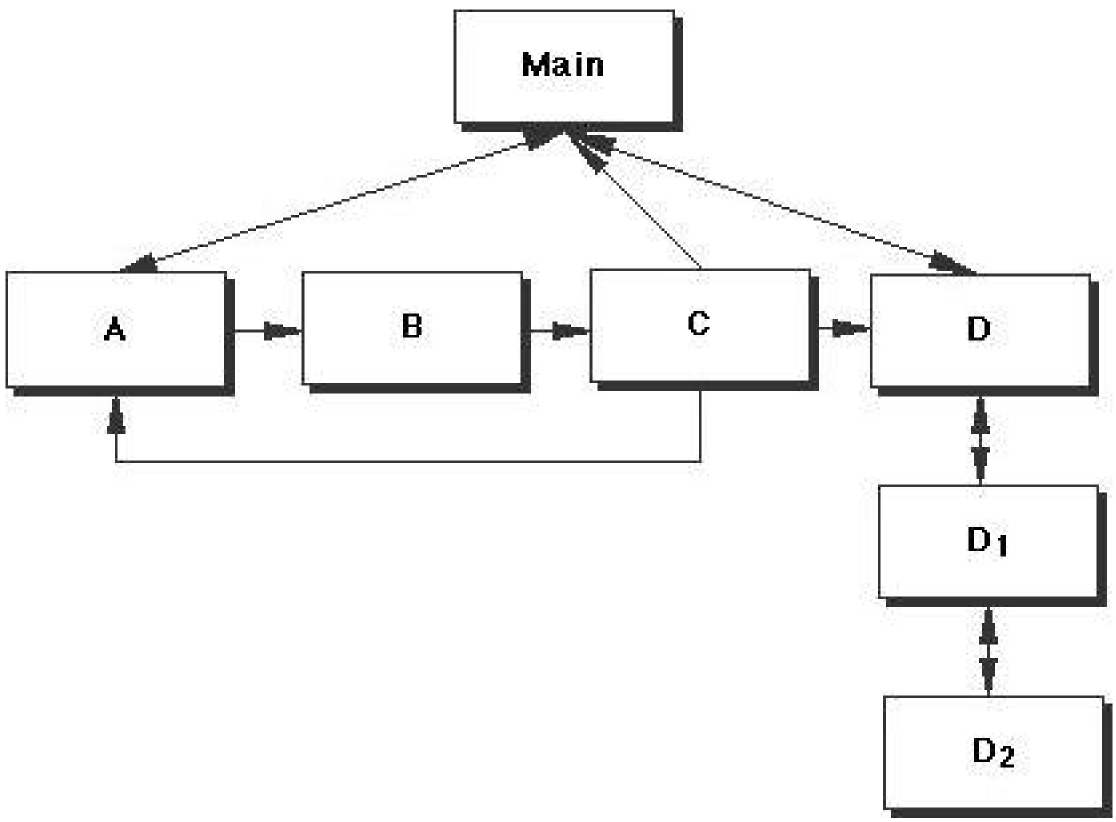
Hierarchical structures (Level 1) introduce a little more complexity because the learner is provided with some choice points (see Figure 4). Pages may provide two or more alternatives, but there is still a general path to follow through the content. A hierarchical structure is similar to a book with a table of contents in that it directs the user to content areas that are more or less structured sequentially. A characteristic of hierarchical structures is that each layer represents a different level of depth. For example, the learning environment used in Beasley and Waugh's (1997) study was structured in a hierarchical fashion with the most general concepts appearing highest in the hierarchy, and the most detailed concepts appearing lowest in the hierarchy; no overlapping subconcepts occurred between higher level concepts.

Figure 4. Simple structure (Hierarchical)



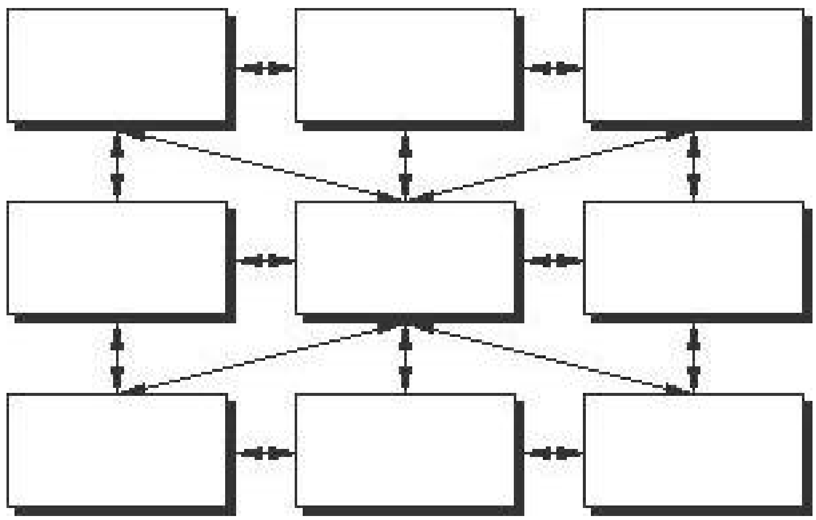
Mixed structures (Level 2) have a non–linear component (see Figure 5). They may contain some linear or hierarchical components but they must incorporate a non–linear feature. Users may navigate freely but they are occasionally constrained to linear presentations of content or to information that is most logically organized in a hierarchy, such as an index or table of contents that imposes structure on an otherwise open system. An example of this type of structure is present in the HyperCard environment that Lawless and Kulikowich's (1996) use in their study. This particular environment contains a main menu screen in which users can choose to explore information on one of three learning theories. Each theory has ten base cards, and each card presents the user with three types of navigational choices: 1) to move to the next base card in the series, 2) to move to one of the other two learning theories, or stacks of base cards, and 3) to return to the main menu. Although the main menu imposes a somewhat hierarchical structure, learners are able to move in between stacks and access hotwords that appear in dialogue boxes.

Figure 5. Mixed structure (Level 2)



Complex structures (Level 3) are non–linear structures that afford opportunities for users to navigate freely within or between paths and structures (see Figure 6). Navigating in a complex structure can be compared to having a collection of articles to research for a topic. The researcher moves back and forth and between those articles freely, similar to a hypermedia user navigating in a non–linear environment. There are hyperlinks between items, but the starting and finishing points are not predetermined. For example, one of the navigation systems in Wright and Lickorish's (1994) study entailed a main screen with a matrix of shops as the column headings and products as row headings. Clicking the mouse on any cell in this matrix displayed the associated price list. Further, users were also able to make a plan, take notes, and access questions to be answered from both the main screen and the price list screen. All of these screens were interconnected, so the user could access them from any other screen.

Figure 6. Complex structure (Level 3).



Interactivity Afforded by the Architecture

Hypermedia learning environments also differ according to the degree of interaction afforded by the system. Hypermedia systems are interactive by nature because users control pacing and movement by choosing and clicking on various hyperlinks. However, the degree of interactivity differs depending upon the system. We identified three levels of interactivity in systems described throughout the logfile literature.

Low interactivity (Level 1) refers to hypermedia environments that provide a predetermined number of states and transitions. Hyperlinks are provided for students to transition between screens or pages, but students cannot add to or change existing content. The most common example of low interactivity is a standard website. Students can transition between pages by clicking on each hotlink.

Moderate interactivity (Level 2) refers to hypermedia environments that are augmented with features that allow users to do more than just click a hyperlink. These features provide means for students to add to or alter existing content. The placement of these features is predetermined; therefore, inputs associated with these features are structured for the user. An example of moderate interactivity is a website containing forms. The forms provide a location for students to input data. The placement of the input field is defined and the amount of information that can be entered in a field is also limited.

High interactivity (Level 3) refers to hypermedia environments that provide flexible tools, such as notetaking and indexing tools. In these environments, students can add to content and change features of the environment with a great deal of flexibility. One example of a high interactivity system is CoNoteS2 (Winne, Hadwin, McNamara, Chu, & Field, 1998). CoNoteS2 is a notetaking system that enables users to take notes, enter definitions into a personal glossary, create hyperlinks between note and glossary entries, index these entries, and highlight text at any time while reading.

We hypothesized that the complexity of analytical techniques would match the complexity of the system; however, this was not necessarily the case. This hypothesis was based on the notion that the more complex the learning environment's interactivity architecture and hypermedia structure, the finer the granularity of the recorded events in a logfile could be. Therefore, the analyses conducted on these detailed logfiles would need to be intricate in order to represent the learner's range of activity while completing complex tasks within this rich environment. Detailed logfiles have the potential to reveal more contextually situated activities, and analytical techniques should capture this complexity.

Analytical Techniques

The diverse nature of the use of logfiles renders them a challenge to collect and analyze. We reviewed 12 empirical studies in the field of education that used logfile data in order to identify how event recordings have been represented. All six published and unpublished conference papers, the one chapter, and one journal article classified as educational studies were eliminated from our review because they did not contain a detailed description of the necessary constructs for complexity coding. The one journal article eliminated presented a study conducted by Horney (1993). Although this study applied a complex hypermedia learning environment, the hypermedia structure was undetermined; learners used the hypermedia environment as an authoring tool, and the study focussed on navigational patterns of these learners rather than the hypermedia structures that they created within the environment. The 12 empirical studies that were coded for complexity were published journal articles and contained information about the task performed within the hypermedia learning environment, the environment's hypermedia structure, and the environment's interactivity architecture. Each of these three constructs was necessary to code for the system's complexity and to ultimately investigate how logfile data describing learner/user behavior was being analyzed in accordance to this complexity (see Table 3). This section describes four ways the reviewed studies examined logfile data: 1) frequency counts, 2) patterns of activity, 3) time–based analysis, and 4) content analysis.

Table 3. Analytical techniques applied for each of the 12 reviewed studies with complexity scores[1].

Studies	Analytical Techniques	Scores
Marchionini (1989)	Frequency Counts, t–test, Patterns through Transition Matrix, Total Time, Mean Time	1 2
Beasley & Waugh (1997)	Patterns through Transition Matrix	2
Beasley & Vila (1992)	Frequency Counts, Analysis of Variance	2
Reed & Oughton (1997)	Frequency Counts, t–test, Total Time	2
Kelly & O'Donnell (1994)	Frequency Counts, Analysis of Variance	2
Lawless & Kulikowich (1998)	Frequency Counts, Cluster Analysis, Total Time	4
Schroeder & Grabowski (1995)	Frequency Counts, Pearson Correlation, Time On Node	4
Lawless & Kulikowich (1996)	Frequency Counts, Cluster Analysis, Total Time	4
Lickorish & Wright (1994)	Frequency Counts, Analysis Of Variance, Total Time	6
Andris (1996)	Frequency Counts, Total Time, Time on Node	8
Horney & Anderson–Inman (1994)	Frequency Counts, Graphical Representations to Code Patterns, Time On Node	8
Fitzgerald & Semrau (1998)	Frequency Counts, Total Time, Time On Node, Counts, Content Analysis	12

Frequency Counts

Eleven out of the 12 studies used frequency counts to represent logfile data. What was counted and how these data were exploited differed. Ten studies used frequency counts as a means of quantifying logfile data in order to perform parametric tests. Reed and Oughton (1997), for example, performed t–tests to examine whether there was a relationship between the total frequency of linear or non–linear steps and total time. Fitzgerald and Semrau (1998) used t–tests to compare the number of times each program choice was accessed for 2 groups with different learning styles. While these two studies used a similar analytical technique, the complexity of their systems varied. The complexity of Reed and Oughton's system was rated at 2 whereas Fitzgerald and Semrau's system scored the highest out of the 12 studies (complexity =12). It is questionable whether Fitzgerald and Semrau captured the complexity of learners' interactions with the hypermedia learning environment, especially since frequency counts were calculated for only 6 of the 23 participants. Other parametric tests performed on frequency counts included Analysis of Variance (e.g., Beasley & Vila, 1992; Kelly & O'Donnell, 1994), Pearson's Correlation (Schroeder & Grabowski, 1995), and Ward's Hierarchical Cluster Analysis (Lawless & Kulikowich, 1996; Lawless & Kulikowich, 1998).

Although parametric tests are a means to compare frequency counts and relate these counts to other variables, analyses based solely on frequency counts of events may not provide an accurate representation of learner engagement (Guzdial, Berger, et al., 1995). Relationships across events are difficult to interpret from results hinged on frequency counts (Misanchuk & Schwier, 1992). For example, although cluster analysis may be a useful technique in classifying a large number of individuals into a smaller number of groups, Lawless and Kulikowich's (1998) application of cluster analysis may

[1] Complexity scores were derived by multiplying task complexity by hypermedia structure complexity by interactivity architecture complexity; each of these three constructs was rated from 1 to 3 as outlined in the Coding the Complexity of the System section.

not have fully captured the dynamic process of navigating within a hypermedia environment. Lawless and Kulikowich sought to define navigation profiles based on frequency counts of base cards accessed for each of the three topics presented, the number of deviations from base cards, resources accessed (i.e., hotwords, vocabulary list), and special features visited (e.g., movies and sound effects). By performing a cluster analysis on these variables and including total time as another variable, Lawless and Kulikowich identified three clusters of participants that exhibited similar navigational profiles: 1) knowledge seekers, 2) feature seekers, and 3) apathetic hypertext users. However, these groupings ignore the possibility that, although two knowledge seekers may choose to view the vocabulary list several times, for instance, these participants may have different strategies for accessing vocabulary words. The mean for the number of resources, including vocabulary words, that were accessed by feature seekers (n=13) was 13.54 (s=8.92). Not only did the frequency of resources accessed vary, but it is unclear when vocabulary words were accessed as well as whether the events surrounding accessing a vocabulary word reflected the students' efforts to understand the text. Although frequency counts provide some evidence to support navigation behavior, triangulating conclusions based on frequency counts with conclusions drawn from other analytical techniques or methods may provide a more accurate representation of user interaction with the hypermedia learning environment.

Patterns of Activity

Patterns of activity may be more revealing than frequency counts because they focus on strategies and they do not isolate events. The main method to establish patterns of activity was to build transition matrices. In a transition matrix, each column–row intersection represents a move from one action to another. Both Marchionini (1989) and Beasley and Waugh (1997) used transition matrices in their studies. Marchionini mapped event transitions in a 9 X 9 matrix, which was collapsed into a 2 X 2 matrix to compare navigation strategies across two groups and for two tasks. Beasley and Waugh, on the other hand, developed a 40 X 40 matrix that included every information node in the learning environment. Predominant navigation patterns were based on the predetermined structure of the environment. For instance, if five links were accessible from one page, most frequent patterns across all learners were sought within that structure. While transition matrices have the potential to provide a rich representation of learners' actions in a complex hypermedia learning environment (Winne, Gupta, et al., 1994) both Marchionini's as well as Beasley and Waugh's systems were rated 2, among the lowest in terms of complexity. In other words, these transitions were examined in very simple hypermedia environments rather than in complex environments where numbers of user choices and logged events increase exponentially.

A limitation of transition matrices is that only two–event transitions can be addressed. The hierarchical structure of the hypermedia environment used in Beasley and Waugh's (1997) study compensated for the investigation of two–event patterns because, once all possible paths from one node to another node were mapped out, the predominant transition for each node–to–node movement was calculated. All node–to–node transitions were then pieced together based on the structure of the hypermedia environment and represented visually in a flowchart fashion. Through this technique, Beasley and Waugh reached the conclusion that participants tend to employ a systematic, top–down, left–to–

right navigation strategy to ensure full coverage of the lesson material. Unlike Beasley and Waugh, Marchionini (1989) concentrated solely on user–initiated actions: He reduced the granularity of the logfile data from nine to two actions representative of information seeking strategies. By doing this, Marchionini simplified possible information seeking strategies within the hypermedia environment, and key information about how this environment may have influenced learners may have been missed. Marchionini transformed his matrix data into mean ratios of lookup and examining search strategies across two tasks for the two groups in his study. Neither Beasley and Waugh nor Marchionini presented matrices of individual user's actions, which may have surfaced more of a variety of patterns.

Another technique to investigate patterns of activity was employed by Horney and Anderson–Inman (1994), who used graphical representations. Visualizations have potential to provide imprecise but intuitive insight into patterns of activity (Jones & Jones, 1997). Horney and Anderson–Inman's Action Code Charts were designed to give a graphic overview of a learner's entire interaction with the environment. In this graph, each event is represented by a vertical bar and is graphed chronologically. Patterns within this sequence are interpreted in order to compare cases and reach qualitative conclusions about the data. For example, the authors deduced from a chart illustrating one particular participant's sequence of events that this participant moved steadily through the content from page 1 to page 43; without an indication of how long she spent on those pages, the participant's level of engagement is unclear. The problem with this type of graph and with transition matrices is that the duration of each event is not indicated; coupling patterns with durations has the potential to reveal richer information about participants' interactions with the hypermedia environment (Guzdial, Berger, et al., 1995).

Time–based Analysis

Representing time collected in logfile data is a challenging analytical technique because without other methods of data collection, it is difficult to analyze what the learner is doing into between events (Rouet & Passerault, 1999). It is perhaps for this reason that elaborate time–based analysis was not present in the 12 studies that we reviewed. Mainly, total time or mean time within the learning environment was described (e.g., Lawless & Kulikowich, 1998; Lickorish & Wright, 1994). Time spent on a particular node was also considered by Andris (1996), Schroeder and Grabowski (1995), and Fitzgerald and Semrau (1998). Schroeder and Grabowski's system was coded at 4, and Andris' system was coded 8 for complexity. As mentioned earlier, Fitzgerald and Semrau's system was rated 12. According to this categorization, the logfiles collected in Fitzgerald and Semrau's study had the potential to be more detailed than the logfiles collected in Andris' and Schroeder and Grabowski's studies because of the system's high level of complexity; this means that a high level of interactivity could have been recorded and that the duration between two events may have been more meaningful. However, in their analysis, Fitzgerald and Semrau focused on the total time of a problem–solving activity within a section of content rather than the time between each learner–initiated action. In answering their question of whether engagement time differed between the two groups of participants with different learning styles, Fitzgerald and Semrau found that one group spent about half the time than the other group on on–line problem–solving activities; their argument would have been augmented had they triangulated this finding or increased the granularity of time measurements. Horney and Anderson–Inman (1994), on the other hand, furthered the measurement of time spent on a certain node by categorizing durations into levels of engagement and then counting the frequencies of each category. None of the 12 studies reviewed attempted to graph time, as suggested by Niegemann (2000).

Content Analysis

Only Fitzgerald and Semrau (1998) coded on–line text entries collected in logfile data. It is important to note that the hypermedia learning environment used in Fitzgerald and Semrau's study was the only one out of the 12 studies that provided this level of interactivity: Students were able to create reports, which were written as a part of the problem–solving activities in the program. The reports were scored, and the scores were used in t–tests to find significant differences between two groups of learners with different learning styles. One limitation of content analysis is that it is performed out of context of the other events recorded in the logfile. As interactivity of hypermedia systems increases, content analysis may become more prevalent.

Discussion

The kinds of logfile analysis conducted in the 12 empirical studies that we reviewed are not reflective of the range of possible analytical techniques proposed in the literature for the representation of logfile data. Analytical techniques focusing on time and patterns of activity were limited in the 12 studies that we reviewed. Our literature search uncovered a series of articles (see Table 1) that further examined the possibilities of logfile analysis and provided references for future directions. However, these articles also neglected to address time–based analysis, short of suggesting graphs of sequences with some indication of duration between events. Suggestions for establishing patterns of activity were made on the basis of graph theoretical methods (Nigemann, 2000; Winne, Gupta et al., 1994) and multidimensional scaling analysis (Berger & Jones, 1995). Electronic logfile analysis tools played a significant role in realizing the suggested analytical techniques for establishing patterns of activity by taking advantage of complex algorithms to search through large logfiles.

Nigemann (2000) asserts that sequential statistics (such as log–linear modeling), Markoff chain analysis through transition matrices, and grammatical methods (such as lag sequential analysis) are not sufficient for analyzing complex hypermedia environments containing a large number of nodes. While sequential statistics do examine whether patterns of activity occur more than by chance, the explosion of the amount of possible combinations of sequences makes it difficult to deal with this type of data. Although Markoff chains provide the probability of two–event transitions, they are memoryless and, therefore, do not establish patterns based on more than two events; further, a coarser granularity of events is needed to make senses of the data and to compare groups (Guzdial, Walton, Konemann, & Soloway, 1994). The lag sequential procedure has potential to detect patterns of activity by comparing pairs of events; however, this procedure would assume a binomial distribution of the probability of accessing a node within a hypermedia environment, and this is uncharacteristic of complex hypermedia structures.

Graph theoretical methods seem to be the most feasible method of investigating logfile data derived from complex hypermedia learning environments (Nigemann, 2000; Winne, Gupta, et al., 1994). Graph theoretical methods entail performing analyses on data from transition matrices. Winne, Gupta, et al. (1994) developed three graph theoretic measures for analyzing event sequences through their software logfile analysis tool LogMill (Nesbit & Winne, 1994): 1) density, a measure of regularity and structure in the sequence; 2) S*, a measure of similarity between two sequences; and 3) structrual equivalence, a measure that gauges whether these events are used interchangeabley or equivalently.

Multidimensional analysis techniques may also prove to be useful in determining patterns of activity (Guzdial, Berger, et al., 1995). Similar to performing graph theoretical methods, this technique also begins with a transition matrix; however, computations on the matrix data differ. Berger and Jones' (1996) software logfile analysis tool Event Recorder supports these computations, which include building a matrix of C correlations to input to a multidimensional scaling algorithm for cluster analysis.

A more recent method of establishing patterns of activity is being made possible through electronic logfile analysis tools (e.g., Jones & Jones, 1997; Okada & Asahi, 1998). These tools support pattern–based sequential analysis and enable searching for specific patterns defined by the researcher. Queries for patterns that are longer than two events can be made, which gives this technique an advantage over analyses based on transition matrices. An example of this type of tool is Jones and Jones' (1997) MacSQEAL. MacSQEAL allows the researcher to transform logfile data by converting events into more abstract representations, recoding events for selective distinctions, and segmenting the data into sequences. MacSQEAL queries generate a list of items specified to display matching lines and the frequency count for the queried pattern. Similar to frequencies of two–event transitions derived from transition matrices, these frequencies have potential to reveal dominant patterns for one particular participant and groups of participants by applying multi–dimensional scaling or graph theoretical methods.

Conclusion

As hypermedia systems become more common learning environments, it is imperative that researchers are able to capture and analyze the complexity of learner–computer interactions. To accomplish this, a common language for logfile collection and analysis must be established in order to make research on the use of logfiles more accessible. Further, this research needs to detail task complexity, hypermedia structure complexity, and interactivity architecture complexity as well as relate these complexities to comprehensive descriptions of analytical techniques and results in order to improve interpretation and comparison of studies. With the continuous evolution of hypermedia learning environments, employing analytical techniques that address the complexity of the learner's cognitive functions in these rich environments will no doubt be challenging. Establishing standards for logfile collection may facilitate the analysis of these data; keylogging tools and standard web server logging tools have already moved toward this goal.

References

Alessi, S. M., & Trollip, S. R. (2001). *Multimedia for learning: Methods and development*. 3rd ed. Needham Heights, MA: Allyn & Bacon.

Ambrose, D. W. (1991). The effects of hypermedia on learning: A literature review. *Educational Technology, 31* (12), 51–55.

Andris, J. F. (1996). The relationship of indices of student navigational patterns in a hypermedia geology lab simulation to two measures of learning style. *Journal of Educational Multimedia and Hypermedia*, *5*, 303–315.

Badre, A. N., Guzdial, M., Hudson, S. E., & Santos, P. J. (1995). A user interface evaluation environment using synchronized video, visualizations and event trace data. *Software Quality Journal*, *4*, 101–113.

Bakeman, R., & Gottman, J. M. (1997). *Observing interaction: An introduction to sequential analysis*. Cambridge: Cambridge University Press.

Barab, S. A., Bowdish, B. E., & Lawless, K. A. (1997). Hypermedia navigation: Profiles of hypermedia users. *Educational Technology Research and Development, 45* (3), 23–41.

Barab, S. A., Bowdish, B. E., Young, B. F., & Owen, S. V. (1996). Understanding kiosk navigation: Using log files to capture hypermedia searches. *Instructional Science, 24*, 377–395.

Barab, S. A., Fajen, B. R., Kulikowich, J. M., & Young, M. F. (1996). Assessing hypermedia navigation through Pathfinder: Prospects and limitations. *Journal of Educational Computing Research, 15* (3), 185–205.

Beasley, R. E. (1992). A methodology for collecting navigation data using IBM's LinkWay: A hypermedia authoring language. *Journal of Education Multimedia and hypermedia, 1*, 465–470.

Beasley, R. E. (1997). Using proximity matrices to identify predominant patterns of navigation through hypermedia knowledge spaces. In T. Müldner & T. C. Reeves, *Proceedings of Ed–Media 1997* (pp. 1342–1349). Charlottesville, VA: AACE.

Beasley, R. E., & Vila, J. A. (1992). The identification of navigation patterns in a multimedia environment: A case study. *Journal of Educational Multimedia and Hypermedia, 1*, 209–222.

Beasley, R. E., & Waugh, M. L. (1997). Predominant initial and review patterns of navigation in a fully constrained hypermedia hierarchy: An empirical study. *Journal of Educational Multimedia and Hypermedia, 6* (2), 155–172.

Becker, R., McLauglin, K., & Rebelsky, S. A. (1999). Project Clio: Tools for tracking student use of course webs. In B. Collis & R. Oliver, *Proceedings of Ed–Media 1999* (pp. 981–986). Charlottesville, VA: AACE.

Berger, C., & Jones, T. (1995). Analyzing sequence files of instructional events using multiple representations. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA.

Britt, M. A., Rouet, J.–F., & Perfetti, C. A. (1996). Using hypertext to study and reason about historical evidence. In J.–F. Rouet, J. J. Levonen, A. Dillon, & R. J. Spiro (Eds.), *Hypertext and cognition*. Mahwah, NJ: Lawrence Erlbaum.

Canter, D., Rivers, R., & Storrs, G. (1985). Characterizing user navigation through complex data structures. *Behaviour and Information Technology*, *4* (2), 93–102.

Chaver, J. C., Carrasco, J., Rossell, M. A., & Vega, J. M. (1998). A graphical tool for analyzing navigation through educational hypermedia. *Journal of Educational Multimedia and Hypermedia*, *7*, 33–49.

Chen, D.–T., Horney, M. A., & Anderson–Inman, L. (1995). The standardized states technique: A utilization of cluster analysis in analyzing students' hypertext processing patterns. *Journal of Educational Multimedia and Hypermedia*, 4, 53–73.

de Vries, H., Netto, W. J., & Hanegraaf, P. L. H. (1993). MatMan: A program for the analysis of sociometric matrices and behavioural transition matrices. *Behaviour, 125 (3–4),* 157–175.

Drott, M. C. (1998). Using web server logs to improve site design. Paper presented at ACM SIGDOC 1998 Conference, Quebec City, QC.

Fitzgerald, G. E., & Semrau, L. P. (1997). Hypermediated learning: Learning styles, path analysis, and knowledge outcomes. In T. Müldner & T. C. Reeves, *Proceedings of Ed–Media 2000* (pp. 494–500). Charlottesville, VA: AACE.

Fitzgerald, G. E., & Semrau, L. P. (1998). The effects of learner differences on usage patterns and learning outcomes with hypermedia case studies. *Journal of Educational Multimedia and Hypermedia, 7,* 309–331.

Fjeld, M., Schluep, S., & Rauterberg, M. (1998). Automatic, action driven classification of user problem solving strategies by statistical and analytical methods: A comparative study. In F. E. Ritter & R. M. Young, *Proceedings of the Second European Conference on Cognitive Modelling* (pp. 98–103). Nottingham, England: Nottingham University Press.

Fujihara, H., Snell, J. R., & Boyle, C. D. (1992). Intelligent search in an educational hypertext environment. *Journal of Educational Multimedia and Hypermedia, 1*, 401–415.

Gall, J., & Hannafin, M. (1994). A framework for the study of hypertext. *Instructional Science , 22*, 207–232.

Gay, G., & Mazur, J. (1993). The utility of computer tracking tools for user–centered design. *Educational Technology, 33* (4), 45–59.

Gomes, M. J. (1997). A quantitative approach to navigation paths and behaviour of inexperienced users of hypermedia documents. In T. Müldner & T. C. Reeves, *Proceedings of Ed–Media 1997* (pp. 570–576). Charlottesville, VA: AACE.

Guzdial, M. (1993). Deriving software usage patterns from log files. (Tech. Rep. No. GIT–GVU–93–41). Atlanta: Georgia Institute of Technology, College of Computing, Graphics, Visualization and Usability Center.

Guzdial, M. (1995). Student articulation and reflection in Project LITERACY–HI's hypermedia system. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA.

Guzdial, M., Berger, C., Jones, T., Horney, M., Anderson–Inman, L., Winne, P. H., & Nesbit, H. (1995). *Analyzing student use of educational software with event recordings*. Unpublished manuscript, Georgia Institute of Technology, Atlanta.

Guzdial, M., Santos, P., Badre, A., Hudson, S., & Gray, M. (1994). Analyzing and visualizing log files: A computational science of usability. (Tech. Rep. No. GIT–GVU–94–08). Atlanta: Georgia Institute of Technology, College of Computing, Graphics, Visualization and Usability Center.

Guzdial, M., Walton, C., Konemann, M., & Soloway, E. (1993). Characterizing process change using log file data. (Tech. Rep. No. GIT–GVU–93–44). Atlanta: Georgia Institute of Technology, College of Computing, Graphics, Visualization and Usability Center.

Hadwin, A. F., Jamieson–Noel, D. L., McTavish, R., McNamara, J. K., & Winne, P. (March, 2000). *Designing Courses to Support Self–Regulated Learning: Teacher as Researcher*. Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans, LA.

Hadwin, A. F., & Winne, P. H. (in press). CoNoteS2: A software tool for promoting self–regulation and collaboration. *Educational Research and Evaluation*.

Hall, R. H., Balestra, J., & Davis, M. (2000). A navigational analysis of linear and non–linear hypermedia interfaces. Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans, LA.

Horney, M. A. (1993). Case studies of navigational patterns in constructive hypertext. *Computers in Education, 20,* 257–270.

Horney, M. A., & Anderson–Inman, L. (1994). The ElectroText project: hypertext reading patterns of middle school students. *Journal of Educational Multimedia and Hypermedia, 3,* 71–91.

Jih, H. J. (1996). The impact of learning pathways on performance in an interactive multimedia learning environment. In P. Carlson & F. Makedon, *Proceedings of Ed–Media 1996* (pp. 78–83). Charlottesville, VA: AACE.

Jones, T., Berger, C. (1995). Students' use of multimedia science instruction: Designing for the MTV generation? Paper presented at the meeting of The National Association for Research in Science Teaching, San Francisco.

Jones, T., & Jones, M. (1997). MacSQEAL: A tool for exploration of hypermedia log file sequences. In T. Müldner & T. C. Reeves, *Proceedings of Ed–Media 1997* (pp. 709–716). Charlottesville, VA: AACE.

Kelly, A. E., & O'Donnell, A. (1994). Hypertext and the study strategies of preservice teachers: Issues in instructional hypertext design. *Journal of Educational Computing Research, 10,* 373–387.

Kurth, M. (1993). The limits and limitations of transaction log analysis. *Library Hi Tech, 42 (11:2),* 98–104.

Lai, Y.–R., & Waugh M. L. (1995). Effects of three different hypertextual menu designs on various information searching activities. *Journal of Educational Multimedia and Hypermedia, 4,* 25–52.

Lamain, W., Mulder, G., & Mulder, L. J. M. (1994). Measuring task performance in human–computer interaction. In F. J. Maarse, A. E. Akkerman, A. N. Brand, L. J. M. Mulder, & M. J. van der Stelt (Eds.), *Computers in psychology: Applications, methods, and instrumentation.* Berwyn, PA: Swets & Zeitlinger.

Larsen, V. A., Kinzie, M. B., Boker, S. M., & Burch, J. B. (1996). Net–Frog: Analyzing monthly user access patterns on the WWW. In P. Carlson & F. Makedon, *Proceedings of Ed–Media 1996* (pp. 78–83). Charlottesville, VA: AACE.

Lawless, K. A., & Kulikowich, J. M. (1996). Understanding hypertext navigation through cluster analysis. *Journal of Educational Computing Research, 14,* 385–399.

Lawless, K. A., & Kulikowich, J. M. (1998). Domain knowledge, interest, and hypertext navigation: A study of individual differences. *Journal of Educational Multimedia and Hypermedia, 7* , 51–69.

Lickorish, A., & Wright, P. (1994). Menus and memory load: Navigation strategies in interactive search tasks. *International Journal of Human–Computer Studies, 40*, 965–1008.

MacGregor, S. K. (1999). Hypermedia navigation profiles: Cognitive characteristics and information processing strategies. *Journal of Educational Computing Research, 20* , 189–206.

MacLeod, M. (1992). Tools for monitoring and analysing the use of hypermedia courseware. In A. Oliveira (Ed.), *Vol. F 92: Hypermedia courseware: Structures of communication and intelligent help* (pp. 19–33). Berlin: Springer–Verlag.

Marchionini, G. (1988). Hypermedia and learning: Freedom and chaos. *Educational Technology, 28* (11) , 8–12.

Marchionini, G. (1989). Information–seeking strategies of novices using a full–text electronic encyclopedia. *Journal of the American Society for Information Science, 40*, 54–66.

Marchionini, G. (1990). Evaluating hypermedia–based learning. In D. H. Jonassen & H. Mandl (Eds.), Vol. F 67 *Designing hypermedia for learning* (pp. 355–373). Berlin: Springer–Verlag.

Marchionini, G., & Shneiderman, B. (1988). Finding facts vs. browsing knowledge in hypertext systems. *IEEE Computer, 21* (1), 70–80.

Marzo–Lázaro, J. L, Verdú–Carbó, T., & Fabregat–Gesa, R. (1998). User identification and tracking into an educational web environment. In T. Ottmann & I. Tomek, *Proceedings of Ed–Media 1998* (pp. 956–961). Charlottesville, VA: AACE.

Mayes, T., Kibby, M., & Anderson, T.. (1990). Learning about learning from hypertext. In D. H. Jonassen & H. Mandl (Eds.), *Vol. F 67 Designing hypermedia for learning* (pp. 227–250). Berlin: Springer–Verlag.

McAleese, R. (1990). Concepts as hypertext nodes: The ability to learn while navigating through hypertext nets. In D. H. Jonassen & H. Mandl (Eds.), *Vol. F 67 Designing hypermedia for learning* (pp. 97–115). Berlin: Springer–Verlag.

Meistad, Ø. (2000). *Collaborative telelearning: Using log–files to identify collaboration patterns*. Unpublished master’s thesis, University of Bergen, Norway.

Meistad, Ø., & Wasson, B. (2000). Supporting collaborative telelearning research using server logs. In J. Bourdeau & R. Heller, *Proceedings of Ed–Media 2000* (pp. 679–684). Charlottesville, VA: AACE.

Misanchuk, E. R., & Schwier, R. A. (1991). Interactive media audit trails: Approaches and issues. In M. R. Simonson & C. Hargrave (Eds.), *13th Annual Proceedings of Selected Research Presentations at the 1991 Annual Convention of the Association for Educational Communications and Technology* (pp. 499–520). Ames, IA: Association for Educational Communications and Technology. (ERIC Document Reproduction Service No. 334 996).

Misanchuk, E. R., & Schwier, R. A. (1992). Representing interactive multimedia and hypermedia audit trails. *Journal of Educational Multimedia and Hypermedia, 1*, 355–372.

Nelson, W. A., & Palumbo, D. B. (1992). Learning, instruction, and hypermedia. *Journal of Educational Multimedia and Hypermedia, 1*, 287–299.

Nesbit, J. C., & Winne, P. H. (1994). *LogMill* [computer program]. Burnaby, BC: Simon Fraser University.

Niegemann, H. M. (2000). Analyzing processes of self–regulated hypermedia–supported learning: On the development of a log–file analysis procedure. Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans, LA.

Nurmela, K., Lehtinen, E., & Palonen, T. (1999). Evaluating CSCL log files by social network analysis. Paper presented at the Computer Supported Collaborative Learning Conference, Standford, CA.

Okada, H., & Asahi, T. (1999). GUITESTER: A log–based usability testing tool for graphical user interfaces. IEICE Transactions Vol. E82, 6, 1030–1041.

Olson, G. M., Herbsleb, J. D., Rueter, H. H. (1994). Characterizing the sequential structure of interactive behaviors through statistical and grammatical techniques. *Human–Computer Interaction, 9*, 427–472.

Pitkow, J. E., Bharat, K. A. (1994). WebViz: A tool for World–Wide Web access log analysis (Tech. Rep. No. GIT–GVU–94–20). Atlanta: Georgia Institute of Technology, College of Computing, Graphics, Visualization and Usability Center.

Rauterberg, M. (1993). AMME: An automatic mental model evaluation to analyse user behaviour traced in a finite, discrete state space. *Ergonomic, 36* (11), 1369–1380.

Recker, M. M. (1994). A methodology for analyzing students' interactions within educational hypertext. In K. Andrews, *Proceedings of Ed–Media 1994*. Charlottesville, VA: AACE.

Reed, W. M., & Oughton, J. M. (1997). Computer experience and interval–based hypermedia navigation. *Journal of Research on Computing in Education, 30* (1), 38–52.

Rouet, J.–F., & Passerault, J.–M. (1999). Analyzing learner–hypermedia interaction: An overview of online methods. *Instructional Science, 27*, 201–219.

Schroeder, E. E., & Grabowski, B. L. (1995). Patterns of exploration and learning with hypermedia. *Journal of Educational Computing Research, 13*, 313–336.

Schwier, R., & Misanchuk, E. R. (1990). Analytical tools and research issues for interactive media. Paper presented at the annual meeting of the Association for Media and Technology in Education in Canada, St. John's, NF. (ERIC Document Reproduction Service No. 321 751).

Seymour, S. (1991). Online public access catalog user studies: A review of research methodologies, March 1986–November 1989. *Library and Information Science Research, 13*, 89–102.

Spiro, R. J., & Jehng, J.–C. (1990). Cognitive flexibility and hypertext: Theory and technology for the nonlinear and multidimensional traversal of complex subject matter. In D. Nix & R. Spiro (Eds.), *Cognition, education and multi media: Exploring ideas in high technology* (pp. 163–205). Hillsdale, NJ: Lawrence Erlbaum.

Unz, D. C., & Hesse, F. W. (1999). The use of hypertext for learning. *Journal of Educational Computing Research, 20*, 279–295.

van Oostendorp, H. (1996). Studying and annotating electronic text. In J.–F. Rouet, J. J. Levonen, A. Dillon, & R. J. Spiro (Eds.), *Hypertext and cognition*. Mahwah, NJ: Lawrence Erlbaum.

Wasson, B. (1999). Design and evaluation of collaborative telelearning activity aimed at teacher training. Paper presented at the Computer Supported Collaborative Learning Conference, Standford, CA.

Westerman, S. J., Hambly, S., Alder, C., Shryane, N. M., Crawshaw, C. M., & Hockey, G. R. J. (1996). Investigating the human–computer interface using the Datalogger. *Behavior Research Methods, Instruments, & Computers, 28* (4), 603–606.

Williams, M. D., & Dodge, B. J. (1993). Tracking and analyzing learner–computer interaction. In M. R. Simonson & K. Abu–Omar (Eds.), *15th Annual Proceedings of Selected Research and Development Presentation at the National Convention of the Association for Educational Communications and Technology*. Ames, IA: Association for Educational Communications and Technology. (ERIC Document Reproduction Service No. 362 212).

Winne, P. H. (1987). Why process–product research cannot explain process–product findings and a proposed remedy: the cognitive mediational paradigm. *Teaching & Teacher Education, 3*, 333–356.

Winne, P. H. (1993). A landscape of issues in evaluating adaptive learning systems. *Journal of Artificial Intelligence in Education*, *4*, 309–332.

Winne, P. H., Gupta, L., & Nesbit, J. C. (1994). Exploring individual differences in studying strategies using graph theoretic statistics. *The Alberta Journal of Educational Research, XL (2)*, 177–193.

Winne, P. H., Hadwin, A. F., McNamara, J. K., Chu, S. T. L., & Field, D. (1998). *CoNoteS2: An Electronic Notebook with Support for Self–regulation and Learning Tactics* [computer program]. Burnaby, BC: Simon Fraser University.

Xia, L., Liebscher, P., & Marchionini, G. (1991). Graphical representations of electronic search patterns. *Journal of the American Society for Information Science, 42*, 469–478.