Web Usage Mining: Application To An Online Educational Digital Library Service

Bart C. Palmer
Utah State University

Follow this and additional works at: http://digitalcommons.usu.edu/etd
Part of the Education Commons, and the Philosophy Commons

Recommended Citation
WEB USAGE MINING: APPLICATION TO AN ONLINE EDUCATIONAL DIGITAL LIBRARY SERVICE

by

Bart C. Palmer

A dissertation submitted in partial fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY in Instructional Technology and Learning Sciences

Approved:

Dr. Mimi Recker
Major Professor

Dr. James Dorward
Committee Member

Dr. Anne R. Diekema
Committee Member

Dr. Jamison Fargo
Committee Member

Dr. Andrew Walker
Committee Member

Dr. Mark R. McLellan
Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2012
Copyright © Bart C. Palmer 2012

All Rights Reserved
ABSTRACT

Web Usage Mining: Application to an Online
Educational Digital Library Service

by

Bart C. Palmer, Doctor of Philosophy
Utah State University, 2012

Major Professor: Dr. Mimi Recker
Department: Instructional Technology & Learning Sciences

This dissertation was situated in the crossroads of educational data mining (EDM),
educational digital libraries (such as the National Science Digital Library;
http://nsdl.org), and examination of teacher behaviors while creating online learning
resources in an end-user authoring system, the Instructional Architect (IA;
http://ia.usu.edu). The knowledge from data/database (KDD) framework for
preparing data and finding patterns in large amounts of data served as the process
framework in which a latent class analysis (LCA) was applied to IA user data. Details of
preprocessing challenges for web usage data are included. A meaningful IA activity
framework provided four general areas of user behavior features that assisted in the
interpretation of the LCA results: registration and usage, resource collection, project
authoring, and project usage. Four clusters were produced on two samples (users with
5–90 logins and those with 10–90 logins) from 22 months of data collection. The analyses
produced nearly identical models with both samples. The clusters were named according to their usage behaviors: *one-hit wonders* who came, did, and left and we are left to wonder where they went; *focused functionaries* who appeared to produce some content, but in only small numbers and they did not share many of their projects; *popular producers* who produced small but very public projects that received a lot of visitors; and *prolific producers* who were very verbose, created many projects, and published a lot to their students with many hits, but they did not publish much for the public. Information about EDM within the context of digital libraries is discussed and implications for the IA, its professional development workshop, and the larger context of educational digital libraries are presented.
This dissertation examined how users of the Instructional Architect (IA; http://ia.usu.edu) utilized the system in order to find online learning resources, place them in a new online instructional activities, and share and use them with students. The online learning resources can be found in educational digital libraries such as the National Science Digital Library (NSLD; http://nsdl.org) or the wider Web. Usage data from 22 months of IA use were processed to form usage features that were analyzed in order to find clusters of user behavior using latent class analysis (LCA).

The users were segmented into two samples with 5–90 and 10–90 logins each. Four clusters were found in each sample to be nearly identical in meaning and were named according to their usage characteristics: one-hit wonders who came, did, and left and we are left to wonder where they went; focused functionaries who appeared to produce some content, but in only a small numbers and they did not share many of their projects; popular producers who produced small but very public projects that received a lot of visitors; and prolific producers who were very verbose, created many projects, and published a lot to their students with many hits, but they did not publish much for the public.

Because this work was in the field of educational digital libraries and educational data mining, the process and results are discussed in that context with an eye to assist novice data miners become familiar with the process and caveats. Implications of user behavior clusters for the IA developers and professional development planners are discussed. The importance of using clustering for the users of educational digital library and end-user authoring tools lies in at least two different areas. First, the clusters can be a very useful framework in which to conduct future studies about how users use the system, from which the tool may be redesigned or simply enhanced to better support different kinds of use. Second, the ability to develop different instructional or promotional outreach for each class can help users accomplish their purposes more effectively, share ideas with other users, and help those struggling with the use of technology.
ACKNOWLEDGMENTS

First, thank you to a loving God in Heaven who has watched over my family while on this long and demanding journey. To my loving and capable wife, Jennifer, and my seven amazing children, thank you for tolerating long hours and short times together. To my parents, in-laws, and friends, thank you for not ever giving up on me through constant questions of how much longer this would take.

Thank you to my advisor, Mimi Recker, for her tireless patience and gentle, nagging encouragement—this would not be here without you. To my committee members, Andy Walker, Anne Diekema, Jim Dorward, and Jamison Fargo, thank you for your willingness to take this project on and your help in making this a great report. I owe all of the requirement tracking help to the Instructional Technology and Learning Sciences staff, Melany Bodily and Launa Julander.

Thank you to my colleagues at USU and particularly the Instructional Architect team over the years for good times and many discussions: Ye Liu, Jayang Park, Deonne Johnson, Xin Mao, Beijie Xu, M. Brooke Robertshaw, and many others who came and went. Thank you again to Jamison Fargo and the Center for Methodological & Data Sciences (OMDS) at USU for advice and direction with statistical and technical matters. Thank you Mary Ellen Heiner for reviewing this document.

My appreciation goes to Ric Ott and the rest of the folks in the Research and Evaluation Department at the Missionary Training Center in Provo, Utah for their patience and concern that this project finish.
Finally, I express appreciation to many who will never know of this work. To those who develop R, PostgreSQL, PHP, and the NSDL teams. Thanks to developers of LaTeX, particularly for the APA6 package—your warnings not use this style for a thesis or dissertation went unheeded. Thank you, BYU Physics Department, for the BYUPhys class that provided a basis for this document.

This material is based upon work supported by the National Science Foundation under Grants No. 840745 & 0434892, and Utah State University. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

Bart C. Palmer
CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Statement of the Problem</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Purpose of the Study</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Definition of Terms</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Significance of the Study</td>
<td>7</td>
</tr>
<tr>
<td>II.</td>
<td>LITERATURE REVIEW</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Knowledge Discovery from Data/Databases</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Machine Learning</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Web Usage Mining</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>Educational Data Mining</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>Educational Digital Libraries and Data Mining</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>Progressive EDM and the Instructional Architect</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>Synthesis of Literature</td>
<td>63</td>
</tr>
<tr>
<td>III.</td>
<td>METHODS</td>
<td>66</td>
</tr>
<tr>
<td></td>
<td>Purpose</td>
<td>66</td>
</tr>
<tr>
<td></td>
<td>Research Questions</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>Pattern Mining Methods</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>Characterization Methods</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>Methods for Reporting Data, Methods, and Tools</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>Methods for KDD Documentation and Generalization</td>
<td>88</td>
</tr>
<tr>
<td>IV.</td>
<td>RESULTS</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>Pattern Mining (KDD) Results</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>Presentation Revisited: Characterization of Results</td>
<td>151</td>
</tr>
<tr>
<td></td>
<td>Data, Methods, and Tool Report</td>
<td>172</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>-------</td>
<td>------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>Example Goals, Questions, and Data for Educational Web Usage Mining</td>
<td>27</td>
</tr>
<tr>
<td>2</td>
<td>A Comparison of Metrics Available Within Each Data Source</td>
<td>49</td>
</tr>
<tr>
<td>3</td>
<td>Study Overview</td>
<td>68</td>
</tr>
<tr>
<td>4</td>
<td>Initial LCA Feature Space</td>
<td>111</td>
</tr>
<tr>
<td>5</td>
<td>Included Direct Effects</td>
<td>130</td>
</tr>
<tr>
<td>6</td>
<td>Summary of Final Models for Two Samples</td>
<td>131</td>
</tr>
<tr>
<td>7</td>
<td>Cluster Size Summary</td>
<td>133</td>
</tr>
<tr>
<td>8</td>
<td>Cluster Change Summary Between Models</td>
<td>135</td>
</tr>
<tr>
<td>9</td>
<td>Cluster Means for Users with 5–90 Logins</td>
<td>137</td>
</tr>
<tr>
<td>10</td>
<td>Cluster Means for Users with 10–90 Logins</td>
<td>138</td>
</tr>
<tr>
<td>11</td>
<td>Binned and Shaded Normalized Cluster Means for 5–90 and 10–90 Logins</td>
<td>152</td>
</tr>
<tr>
<td>12</td>
<td>Cluster Characterizations</td>
<td>173</td>
</tr>
<tr>
<td>13</td>
<td>Comparison with a Similar Study</td>
<td>187</td>
</tr>
<tr>
<td>14</td>
<td>Comparison of Cluster Description and Sizes with a Similar Study</td>
<td>203</td>
</tr>
<tr>
<td>B1</td>
<td>Initial Data Transformations from the IARD Tracking Entries</td>
<td>230</td>
</tr>
<tr>
<td>B2</td>
<td>Session-Level User Activity Feature Space</td>
<td>231</td>
</tr>
<tr>
<td>B3</td>
<td>Resource-Related User-Aggregated Activity</td>
<td>234</td>
</tr>
<tr>
<td>B4</td>
<td>Project-Related User-Aggregated Activity Feature Space</td>
<td>235</td>
</tr>
<tr>
<td>B5</td>
<td>Project Edit-Session Activity Feature Space</td>
<td>236</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The Instructional Architect (IA) home page</td>
<td>42</td>
</tr>
<tr>
<td>2</td>
<td>A sample IA project designed to teach children about the brain</td>
<td>43</td>
</tr>
<tr>
<td>3</td>
<td>An example web server log from the Instructional Architect</td>
<td>51</td>
</tr>
<tr>
<td>4</td>
<td>An example Google Analytics visit plot from May 2007 to January 2008</td>
<td>53</td>
</tr>
<tr>
<td>5</td>
<td>Three example Google Analytics visit plots from January 2008 to November 2009 with smoothing</td>
<td>54</td>
</tr>
<tr>
<td>6</td>
<td>An example Google Analytics geo-location report from January 2008 to November 2009</td>
<td>55</td>
</tr>
<tr>
<td>7</td>
<td>Data relationship map</td>
<td>73</td>
</tr>
<tr>
<td>8</td>
<td>Clickstream example</td>
<td>76</td>
</tr>
<tr>
<td>9</td>
<td>Feature histograms for the presegmented data</td>
<td>116</td>
</tr>
<tr>
<td>10</td>
<td>Feature histograms for the 5–90 login sample</td>
<td>119</td>
</tr>
<tr>
<td>11</td>
<td>Feature histograms for the 10–90 login sample</td>
<td>120</td>
</tr>
<tr>
<td>12</td>
<td>Box &amp; whisker plots of all features for the 5–90 login and 10–90 login samples without cluster breakouts</td>
<td>122</td>
</tr>
<tr>
<td>13</td>
<td>BIC comparison plot</td>
<td>126</td>
</tr>
<tr>
<td>14</td>
<td>Cluster size panel plots: raw, log, and normalized</td>
<td>134</td>
</tr>
<tr>
<td>15</td>
<td>A plot of the raw means for Clusters 1–4 on the 5–90 and 10–90 login models</td>
<td>140</td>
</tr>
<tr>
<td>16</td>
<td>A log-scale plot of the means for Clusters 1–4 on the 5–90 and 10–90 login models</td>
<td>142</td>
</tr>
<tr>
<td>17</td>
<td>A plot of normalized means for the 5–90 and 10–90 login models for Clusters 1–4</td>
<td>144</td>
</tr>
<tr>
<td>18</td>
<td>A plot of normalized means for Clusters 1–4 for the 5–90 and 10–90 login models</td>
<td>145</td>
</tr>
<tr>
<td>19</td>
<td>Box &amp; whisker plots of Clusters 1–4 for the 5–90 login group</td>
<td>149</td>
</tr>
</tbody>
</table>
CHAPTER I
INTRODUCTION

Statement of the Problem

Making sense of data to gain knowledge about the world is a common occurrence. Newborn babies, scientists, students, and teachers all use data in order to better understand, adapt to, or change their surroundings. While astute observation and reflection suffice to extract knowledge from certain data, there are many kinds of data which are not easily subjected to knowledge discovery, even with the aid of analysis tools (Ye, 2003).

As education becomes more technologically advanced the volume and complexity of available data about learning also increases. For example, in computer-based learning environments, student and instructor behavior can be captured, combined, and analyzed to investigate and improve the educational experience (Hwang, Chang, & Chen, 2004; Romero & Ventura, 2007). Specifically, a Web-based educational site may track pages visited and in what order (i.e., random access, sequential, or both) to determine if user behaviors indicate or support potential learning as intended by the instructional design and user interface. These data are regularly collected in web server log files as web usage data (e.g., Apache Software Foundation, The, 2005).

Unfortunately, however, much of these fine-grained data are underutilized by educational researchers, due in part to several causes: (a) the nature of these data (e.g., volume, distributions, and type) are unfamiliar as are their sources (e.g., “obtaining observation data from a computer log?”); (b) the exploratory approach of data analysis is
not as comfortable as the more ubiquitous null-hypothesis significance testing and safety of statistical significance in confirmatory types of analyses (Tukey, 1962; Zhao & Luan, 2006); (c) difficulty feeling comfortable in "finding the right question [which] is more important than finding the answer" (Tukey, 1980, p. 24); and finally, (d) educational researchers do not yet have much of the needed knowledge to use developed tools to analyze these data and interpret the results (Romero & Ventura, 2007). Indeed, expanded thinking and robust methods are needed to deal with this data deluge.

Fortunately, education is not alone in the struggle to make meaning of copious information as with the rise of the information and computing age vast amounts of data are now available in many fields. Since the 1980s, the knowledge discovery from data/databases framework (KDD; also knowledge discovery), and its well-known step of data mining (DM; or automated and convenient analysis), have become relatively well defined (Benoit, 2002; Trybula, 1997; Vickery, 1997) and are responsible for enormous advances in many fields such as biology, chemistry, math, medicine, computer science, and business (Bath, 2004; Han & Kamber, 2006; Vickery, 1997; Witten & Frank, 2005; Ye, 2003).

One application of KDD uses data from the World Wide Web (WWW or Web) and is called web usage mining (WUM). In web usage mining, data generated by or made available from web-based communications can be analyzed for user activity for web metrics. Web-server logs that document web browser requests are a common source of WUM data.
E-commerce has made extensive use of web metric data to gain insights into traffic patterns and how to improve the probability of goals (for example, the goal of improving completed sale *conversion*, or the accomplishment of a business goal—closing the sale, e.g., Cooley, 2003). Many analytic tools have been developed that help website owners analyze visit patterns and track the effects of changes to the website (e.g., Omniture and Google Analytics).

Notwithstanding the availability of KDD, data mining, and WUM techniques, the quest to conduct meaningful analyses of available educational data using modern tools is only just beginning. *Educational data mining* (EDM) for research has been gaining interest and credibility in special topics, special interest groups, and workshops at other conferences and has led to the creation of the International Educational Data Mining Society (http://www.educationaldatamining.org/proceedings.html). Conferences (see http://educationaldatamining.org/IEDMS/events), a journal (see http://www.educationaldatamining.org/JEDM/), and handbook (Romero, Ventura, Pechenizkiy, & Baker, 2010) have marked the emergence of EDM as a field of study. As a new field of study, data mining in education has ample need for additional research into the application of the KDD process.

Meanwhile, the potential for positive impact on the classroom from the educational use of online learning resources has been promising—particularly those cataloged in educational digital libraries (EDL, Marchionini & Maurer, 1995). The digital library movement, which began in the early 1990s, continues to grow and expand in both storage/retrieval and services (Lagoze, Krafft, Payette, & Jesuroga, 2005). In general,
goals of digital libraries are like the goals of brick-and-mortar libraries, to facilitate preservation, discovery, and use of the items in the repository (e.g., Kastens et al., 2005).

There is an unexplored space with respect to the ultimate usage of educational digital library resources by teachers and students. This void springs from, among other things, the difficult nature of observing conditions in which the online learning resources are employed (Bartolo, Diekema, Khoo, & McMartin, 2006). Additionally, longitudinal- or observation-intensive questions have often not been in the scope of educational digital library project goals, thereby eluding exploration. For example, exploring “how do teachers’ instructional design activities with online learning resources change over time?” would be intractable unless the library has implemented a user identification and tracking system—which has not always been the case. Finally, since educational digital libraries are accessed from nearly everywhere in the world via the Internet and Web, it is also very difficult to gain access to the wide variety of instructional contexts in which the digital holdings used or even how they are presented to students.

For example, if a user searches in a large metadata (akin to library card information) digital library and finds an online learning resource that is physically located within a special-topic digital library then the large library will know of the search techniques employed to find the resource and the resource repository will know of the resource use, but neither will know what the other knows or in what instructional context the resource was placed.

Digital library end-user authoring tools (e.g., the Instructional Architect [IA] at http://ia.usu.edu) are situated where teachers can create and utilize online learning
contexts that embed links to online learning resources. Traditional research studies into the behaviors of teachers in these design activities as well as use of these new online learning contexts has informed both the metadata libraries and the resource repositories of learning resource use (Recker, Dorward, Dawson, Halioris, et al., 2005; Mao, 2007; Walker, Recker, Leary, & Robertshaw, 2010). While some web metrics have been reported in these studies, a full data mining approach has only just begun to be explored (Maull, Saldivar, & Sumner, 2010; Recker, Xu, Hsi, & Garrard, 2010; Xu & Recker, 2010; Xu, Recker, & Hsi, 2010).

At the intersection of recent advances in KDD and data mining and lack of educational application (especially in the realm of educational digital libraries), there has been a gap in the knowledge-base of how teachers behave when given an opportunity to find, annotate, and use online learning resources in their teaching. A data mining approach to the exploration of user behaviors with an end-user authoring system that provides functionality where the knowledge gap exists was a next step to better understanding.

**Purpose of the Study**

A particular educational digital library, the National Science Digital Library (NSDL; [http://nsdl.org](http://nsdl.org)), has several end-user targeted partners who have created different services in order to enable more effective usage of the library.

One of these NSDL-funded services is the IA ([http://ia.usu.edu](http://ia.usu.edu)), a simple end-user online authoring service that was created with the dual intent of (a) facilitating K–12 teachers to design their own online educational resources by annotating and
sequencing existing online resources, and (b) conducting research on teacher and student use of existing and newly created resources.

The IA has been operating since 2002 and has been introduced to many teachers in tens of professional development workshops (particularly K–12), as well as virally since late 2003. Due to the purposes, longevity, and data amassed, the IA was an educational digital library service ripe for investigation into teacher behavior while they engage in designing lessons and teaching with online resources.

One study has been performed by Xu (2011) on IA user behavior data that examined a certain analysis technique called latent class analysis (LCA) as a suitable approach to categorization. However, there was still a lack of knowledge of the overall KDD process in examining such data. This work sought to fill in knowledge of the preprocessing for the novice educational digital library data miner as well as provide perspective of an additional approach to examination of web user data.

Therefore, the purpose of this study was to perform a research project in the field of EDM (Romero & Ventura, 2007, specifically WUM) through application of the knowledge discovery from data framework (KDD; e.g., Han & Kamber, 2006; Vickery, 1997) upon the IA user data.

The specific goals of this study were as follows:

1. Mine for patterns emerging from meaningful IA user activity data (pattern mining).
2. Characterize user behavioral patterns in plain terms (characterize).
3. Report on how data, methods, and tools were utilized for this study (report).
4. Document the EDM process for the study of educational digital libraries

(*document*).

**Definition of Terms**

Within the various communities that work with KDD and data mining in general, different definitions exist for the same term as well as different terms for the same definition. Appendix A contains a glossary of technical terms (mainly denoted by *italic* font) as applied in this dissertation in order to remove confusion.

**Significance of the Study**

There are multiple facets of this study’s significance.

1. Expanding the body of work in EDM. This work was produced during the early days of EDM, and expands the body of literature in that field. While there are ample sources for data mining in business and other areas, few related journal articles exist and even fewer dissertations to date to assist the novice educational data miner. When the vast amount of education data that could be mined is considered, this work represents another needed stake in the ground of EDM.

2. Expand WUM in educational digital libraries to include patterns of online learning resource use in the context of teachers designing instructional activities. In the area of educational digital libraries, it is not often that one can trace how teachers reuse the learning resources available to them. While the context for this dissertation was somewhat constrained to the opportunities
afforded by the IA, it was significant that it peeked into the design behaviors of teachers with regard to online learning resources.

3. Explore and describe the utility of LCA as a clustering technique for user behavior. In the educational digital library world, there have been a lot of traditional methodologies used along with some newer methods of visualization and web metric analysis. However, there had not yet been any classification using LCA with the exception of Xu (2011) who used a top-down approach. The current study was significant in that it explored and described the use of LCA with a bottom-up approach.

4. Inform other educational researchers of the process of knowledge discovery. There are many educational digital library researchers who have not yet utilized data mining techniques to understand user behaviors. This work also filled a need in the educational digital library community as outlining the process of KDD. The pitfalls and caveats that befell the researcher have been generalized and can serve to help other researchers in their use of KDD with educational data.

5. Inform the managers of educational digital libraries (both metadata and resource repositories) of usage patterns as feedback for further development of these systems. This research also informed the IA developers about the online usage behavior that can be translated into changes of the IA and associated teacher professional development workshops. As well, important information was learned that can impact how digital libraries index and offer their resources.
CHAPTER II
LITERATURE REVIEW

Traditional educational research data collection methods (e.g., survey, observation, interview, assessment) are often enhanced with computer technology. Additionally computer-aided learning environments (e.g., online courses, intelligent tutors, course management systems) often have traditional assessments and participation measures built in. Not only does computer enhancement cause these traditional learning and research mediums to have a wider reach and simplify the collection and archiving of data, they also have made available fine-grained, nontraditional data—data not initially collected for educational purposes.

These new data are easily collected when client and server software are configured to log user activity tirelessly, consistently, and efficiently. Originally, these logs were intended to inform the environment developers and administrators of any production-limiting problems; however, keystroke and mouse-click capture (Laubscher, Olivier, Venter, Eloff, & Rabe, 2005), biofeedback data (Laufer & Németh, 2008), audio and video recordings (Petrushin, 2007), and client/server logs (Srivastava, Cooley, Deshpande, & Tan, 2000) have now been utilized as a methods of tracking user behavior. Together, the combination of new sources of data with computer-enhanced methods of traditional data sources provide today’s educational researchers with unprecedented access to learner data (Merceron & Yacef, 2005; Witten & Frank, 2005; Ye, 2003).
In the construction of this research project in which large amounts of web data were to be used to classify and characterize user groups, a review of the literature was performed covering the following areas: (a) knowledge discovery from data/databases (KDD), (b) machine learning, (c) web usage mining (WUM), and (d) educational data mining (EDM). To map the terrain of these very broad and intertwined topics, a set of general KDD, data mining, EDM, and methodology reviews were read (e.g., Bath, 2004; Benoit, 2002; Romero & Ventura, 2007; Vickery, 1997) and data mining textbooks consulted (Han & Kamber, 2006; Hastie, Tibshirani, & Friedman, 2001; Witten & Frank, 2005; Ye, 2003). Discussions with the Office of Methodological and Data Services (OMDS) at Utah State University (http://cehs.usu.edu/psychology/omds) provided additional understanding and direction in this search.

Keyword searches for “knowledge discovery from data,” “educational data mining,” “data mining,” and “web metrics” (among other variations, e.g., “web-o-metrics”) were performed in various databases to evaluate their result relevance. Google Scholar, ERIC, Education Full Text, ACM, Utah State University’s Merrill-Cazier Library catalog, and Digital Dissertations were all searched with varying degrees of success. In general, noneducational KDD-related information was readily available and growing. For example, searching on the quoted string “data mining” in Google Scholar (scholar.google.com) returned over 400,000 results and nearly 700,000 by 2011; but searching for “educational data mining” showed a relatively small number of references—a little over 200 in 2008 and just under 1,400 by 2011.
A summary of each review area will now be presented. Additional background information is also presented about educational digital libraries and to-date research about their users. Finally, a synthesis of findings is given in relation to this dissertation.

**Knowledge Discovery from Data/Databases**

With the rise of the information and computing age vast amounts of data are now available in many fields. Algorithms and methods for discovering patterns in large and complex data stores have been developed that make the procurement of knowledge from data/databases (KDD or knowledge discovery, see Benoit, 2002; Trybula, 1997; Vickery, 1997) more explicit and procedural. Since the 1980s, the KDD process, and its well-known step of data mining (DM; or automated and convenient analysis), have become popular and are responsible for enormous advances in many noneducational fields such as biology, chemistry, math, medicine, computer science, and business (Bath, 2004; Han & Kamber, 2006; Vickery, 1997; Witten & Frank, 2005; Ye, 2003).

Many variations of KDD exist, are based on the questions asked and the kinds of data available, and are often performed by repurposing data from other inquiry or activity logs. In general, knowledge discovery from collected data has four steps (Han & Kamber, 2006; Liao, 2003; Vickery, 1997):

1. cleaning and integrating,
2. selecting and transforming,
3. data mining (or actual analysis), and
4. evaluating and presenting.
The term *preprocessing* has been applied to the first two steps—the preprocessing before data mining and interpretation.

The following descriptions are intentionally general with specific discussion of the application of these ideas in the context of this study in Chapter III. Most of the information is taken from Chen and Chau (2004), Han and Kamber (2006), Vickery (1997), Witten and Frank (2005), and Wong, Leung, and Cheng (2000).

**Cleaning and Integrating**

Quite often KDD is applied to multiple and distributed data sources that were independently developed and maintained. Because of this independence missing, incomplete, incompatible, or otherwise *noisy* data are to be expected. Noise in data can pose a significant problem for conducting any meaningful analysis and therefore data must be cleaned and distributed sources integrated before further work may proceed. Several authors warn that dealing with noise is one of the most important and time-consuming problems in KDD. The cleaning (and part of transforming) process, therefore, may include activities such as: filtering noise, filling gaps, correcting erroneous data, and matching date and other key identifier fields.

The process of integration is bringing all the data sources into one large, searchable, filterable, and consistent location. Data cubes, relational databases, and tables are popular ways to organize data with multidimensional lookup and convenient summarization abilities in a *data warehouse* or smaller *data mart*. 
Because the preprocessing steps are very individualized for each data mining instance, only general data structures are mentioned in the literature. Even case studies do not always give the specifics of their cleaning and integration. The lack of helpful information or even listing possible caveats in preprocessing does not serve newcomers to the data mining community and has constituted a gap in the literature that needs filling.

**Selecting and Transforming**

With so much data, it is important to remember that patterns that are very difficult to comprehend are very hard to interpret and act on in a meaningful way. In other words, it is difficult to find patterns that are simple enough to be useful. Furthermore, the presence of irrelevant variables or examples of variables (i.e., data points that are artifacts of repurposed data) can confuse or prolong the analysis (Blum & Langley, 1997). Therefore, it is desirable to have a parsimonious model created from just a few select variables. With a priori knowledge about the data, it is possible to select the most appropriate variables quickly. If there is no reason to select some variables over others, different variable combinations may need to be explored.

Various kinds of transformations exist and are used to: (a) aggregate data (aggregation), (b) simplify the data (smoothing), (c) convert primitive or low-level data to higher-level constructs (generalization), (d) create new variables which may be “hidden” in existing data (feature construction), and (e) reshape the data in order to fit the assumptions of analysis (normalization; see Han & Kamber, 2006). Each of these
transformations takes time and effort. However, if care is taken in the cleaning and integration then some transformations will not be necessary.

An example will illustrate this KDD step with web data generated from a browser. Suppose a learning management system enables course material to be posted online, assignments submitted, and communication between class members via a threaded discussion. User data web-generated data may show a sequence of browser requests as a user logs in, looks at an assignment page, posts to the class’ threaded discussion, follows some links to external information, and then submits an assignment.

Preprocessing this example user data may include combining a web server log (that contains what pages were visited, when, and other information) with a backend database (that stores course content, discussion posts, and submitted assignments) into a single data source that contains all information about the sequence of events.

The sequence described above can be transformed into a session, visit, or episode that has one or more purposes. Additional variables may also be extracted for analysis such as the length of the visit in time, the length of the discussion post, what key words were used in the post, and an indication as to the relationship between the assignment page and external information viewed related to the assignment submitted. Thus, the raw data are transformed into meaningful indicators or features of user behavior.

After transformation the data miner will then select certain variables for analysis based on their ability to answer the question being asked. Finally, depending on the analysis technique (that may assume a particular distribution like normal or Poisson) the data may be further mathematically transformed prior to analysis.
Data Mining—as Analysis

At the heart of KDD is exploring data for patterns, coverage, interconnectedness, similarities and differences, or outliers. Because of its central place in KDD the term data mining is often used to refer to the entire KDD process.

Two major divisions among analyses are traditional statistics and machine learning (see Zhao & Luan, 2006, this is elaborated upon in the section on machine learning below). Depending on the kind of knowledge desired (i.e., the questions asked) and available data (e.g., text, clickstream, count), different analyses are appropriate. Vickery (1997) provided an early review of methods and applications. Cooley (2003) constructed a helpful figure of pattern discovery layers that connects different data mining applications, methods, and appropriate algorithms that may be used based upon the questions being asked (2003, Figure 25.7, p. 608). The chart is not exhaustive, but it can be very helpful as a starting point for researching alternatives and recent analytical developments.

Evaluating and Presenting

The number of models that result from analyses vary depending on the methods used, therefore, evaluation is essential before culling results that are not helpful or otherwise do not inform the purpose of the research. Many algorithms have the ability to reduce the number of results through a number of different criteria that aid in the evaluation process (Blum & Langley, 1997, some evaluation methods are mentioned in the pertinent analyses section below).
After evaluation and selection of the model from data mining, it is necessary to communicate this new knowledge to others. Graphs, illustrations, 3D plots, use of color, and other visualization techniques are commonly used to present the findings in an easy to understand manner. Additionally, helpful textual descriptions, summaries, and even metaphors can be used to communicate the results of data mining.

**Machine Learning**

Analyses of data have long been done through traditional statistical methods. Recker and Palmer (2006) utilized simple statistical methods to show that there are different usage groups of an end-user authoring system. While this kind of analysis can be helpful in understanding large stores of data, there are problems as well (Zhao & Luan, 2006). For instance, with a large N in a typical ANOVA or linear regression, it is quite easy to obtain statistical significance but the resultant power of the test may be compromised (Howell, 2002). In addition, hypotheses are required that may simply be unknown at the outset of analysis therefore other methods must be applied (Zhao & Luan, 2006).

Machine learning is a popular method to use for data mining because of the ability to handle large quantities of data and find patterns which were previously unknown (Chen & Chau, 2004). While machine learning may, at first hearing, sound like a science fiction marvel, it is simply algorithms (or procedures) taking advantage of the computational abilities of computers. Several books are available on machine learning algorithms (e.g., Hastie et al., 2001) and KDD or data mining tools (e.g., Witten & Frank, 2005) that not
only provide methods but dissect algorithms so that the data mining researcher can understand how the computer is deciding upon the outcome. Truly there are many options for the KDD researcher with more sophisticated and scalable analyses than traditional statistics (Zhao & Luan, 2006).

Due to the large space of machine learning algorithms, an exhaustive review is outside the scope of this work. A scheme for differentiating machine learning algorithms is now presented, followed by a more specific review of analyses that fit the application in this study as per the application, method, and algorithm chart mentioned above from Cooley (2003).

**Supervision in machine learning.** Two groupings of machine learning analyses that are commonly used are: supervised and unsupervised (Chen & Chau, 2004; Hastie et al., 2001). The selection of algorithm will depend on the reason for data mining.

**Supervised.** Algorithms, which require the outcome classes to be known a priori from which the machine will learn or train are known as *supervised*. This class of algorithms is helpful for (among other things) prediction and classification of future examples, hence the need to have a specified outcome variable (or class). Examples of this type of analysis are: neural networks, association rules, classification trees, and certain regressions.

These algorithms are helpful in situations similar to the traditional use of confirmatory analyses (Tukey, 1962, 1980). Once the machine has learned its rules (which can be time-consuming), it can efficiently and effectively perform its function as long as the rules remain valid (i.e., there is no change in the expected frequencies, distribution,
interpretation, etc., of the input data). Most applications of data mining require the use of supervised algorithms, but not in this dissertation study.

**Unsupervised.** Algorithms, which derive segments or classes from the data rather than having the classes explicitly defined as with supervised algorithms are known as *unsupervised*. Examples of unsupervised algorithms are: association rule mining, traditional clustering ($k$-means), self-organizing maps, and latent variable models.

When there is no conceptual model known to exist within the data, these methods serve to find patterns that might suggest some kind of relationship between or among variables—similar to the traditional exploratory kinds of analysis (Tukey, 1962, 1980). A priori knowledge of data relationships can improve performance by ruling out some relationships or suggesting that other relationships exist, thus reducing the volume and complexity of the search space. Once found, these classes can be compared empirically or tested against expert knowledge for accuracy, depending on the specific algorithm, and confirmed through subsequent use or triangulation (Zhao & Luan, 2006).

The purpose of this work was to apply the KDD process and in so doing discover and understand similarities and differences between behavioral patterns. This kind of analysis requires unsupervised machine learning techniques. The next section will expand on this area specific to this study.

**Pertinent analyses.** Cooley (2003) mapped the data mining terrain in a figure that connects appropriate analyses to business questions through various applications of data mining and methodologies. According to this chart, when the business question deals with customer description as this study does, then the application is segmentation, which
is accomplished via any of the following three methods: clustering (e.g., k-means), association rules (e.g., apriori), or collaborative filtering (e.g., Pearson correlation) (2003, see Figure 25.7, p. 608).

While many algorithms exist for these three segmentation methods, not all were applicable to the purposes of this study. Collaborative filtering is a methodology that looks at commonalities between users and can also be used in recommender applications (Adomavicius & Tuzhilin, 2005, e.g., Amazon’s recommendation system). Because the focus of the current study’s data mining was description (and not recommendation), collaborative filtering was not used and will not be discussed in any further detail.

Association rule mining is an unsupervised algorithm that essentially counts the occurrences of relationships in the data (Webb, 2003, p. 26). These rules may be helpful in further confirming and clarifying the characteristics of latent clusters or in discovering additional insights the cluster analyses. An example rule may look like this (a fictional example based on section 4.4.1 of Khoo et al., 2008):

\[ \text{When timeOnPage} > 2.5 \text{min and weekday then userIsStudent (90\% coverage)} \quad (II.1) \]

Meaning that 90\% of the time that the timeOnPage is more than 2.5 minutes and the visit occurred on a weekday then the visitor was a student. Because association rules do not provide a whole clustering mechanism, but would rather be more helpful exploring the feature relationships within a clustering model, this data mining analysis was not pursued and will, like collaborative filtering, not be discussed further.
Clustering is the process of finding the closest similarities within groups while finding the greatest differences between groups (Jung & Wickrama, 2007; Vermunt & Magidson, 2003). There were several techniques available for clustering that are classified as *black-box* or *transparent* by the level of information provided concerning the inner workings of group-membership decisions. Two relevant algorithms will be introduced here: *k*-means and latent class analysis (LCA).

**k-means clustering.** For the identification and characterization of unknown groups in data a researcher would commonly aggregate vectors (variables) to a single level and apply traditional cluster analysis (e.g., counts, sums, and averages of a user; for an example, see Luan, 2004).

The *k*-means clustering algorithm operates on a defined notion of distance. Therefore, if a researcher wanted to cluster students according to their homework and test scores, she could average the homework scores (aggregate) and list them with test scores before running the algorithm. She would provide the machine with the number of clusters to find (based upon her knowledge) and perhaps even the initial within-group averages. Then, the machine would iteratively look at the distance of each student’s scores from the group means and other students along the two dimensions and determine which are close and which are far, readjusting the cluster means and group membership as needed. In the end, the algorithm will produce *k* sphere-shaped groups, no matter how large or well they fit the data. Decisions on the fitness, utility, and comparison between these groups are left up to the researcher’s discretion.
Because traditional cluster techniques are not transparent (i.e., black-box analyses), they must be followed by a subsequent analysis (e.g., logistical regression) in order to characterize the groups (Magidson & Vermunt, 2002). Other drawbacks of traditional cluster analysis (as well as factor and principal component analyses) are that they require continuous variables, are not as stable with missing data, and their results are unable to be statistically compared (for additional description of these techniques and their limitations, see Apley, 2003; Magidson & Vermunt, 2002, 2004; Stevens, 2002).

**Latent class analysis.** Originating in psychometrics, LCA has begun to see some widespread use and can be used to probabilistically classify and characterize input variables into groups in one transparent analysis step (unlike $k$-means clustering, see Ip, Cadez, & Smyth, 2003; Jung & Wickrama, 2007; Vermunt & Magidson, 2003). Variants of LCA are known as: latent variable analysis, latent cluster analysis, finite mixture models, and growth mixture models. In this paper, LCA will be used to mean the whole family of latent class modeling unless a specific analysis is mentioned.

The strength of LCA lays in its ability to classify based on probabilities of class membership with many kinds of data (e.g., continuous, categorical) and with several underlying distributions (e.g., normal, Poisson). Traditional cluster algorithms depend on a continuous distance measure, which does not accurately model some aspects of the real world (Magidson & Vermunt, 2002; Vermunt & Magidson, 2003).

The class membership decision mechanism during LCA analysis is commonly based upon the expected maximization (EM) algorithm (Magidson & Vermunt, 2004) which seeks for a best fit even with missing data and with a mixture of underlying
distributions. The expected maximization process begins with random starting model parameter values and then looks at the marginal probabilities of actually being part of that class (i.e., class membership) for each observation. Class membership is adjusted and probabilities are again calculated with each iteration until the model becomes stable. A likelihood statistic is calculated for class memberships as a model fit measure.

The number of clusters to be found in LCA are still determined by the researcher, like traditional $k$-means, but decisions regarding model selection are made with the assistance of various statistics that can support expert knowledge of the data. For example, the Baysian Information Criteria (BIC) is a measure of relative model fit when comparing two models (Ip et al., 2003; Muthén, 2004) and is calculated from the overall likelihood of class membership. A small BIC value indicates a better model with respect to its complexity.

When two models are compared, a conditional bootstrap analysis may also be performed to verify that two models are significantly different, assuming that the more restrictive model (i.e., the model with a smaller $k$) is a true model (Vermunt & Magidson, 2005, pp. 98–99). Statistical significance on a bootstrap analysis indicates that the new model is significantly better.

Criticism of LCA has been targeted at the assumption of local independence between the observed variables within each class (see Bauer, 2007). However, extensions to the original algorithms allow for relaxing this assumption by allowing direct effects between features. This relaxation is done through manipulation of the off-diagonal variance/covariance matrix, forcing them to 0 or near 0 entries (Vermunt & Magidson,
2005, pp. 75–76). Other extensions to LCA have expanded its power as well as expanded the power of LCA to additional situations (Vermunt & Magidson, 2003, e.g., longitudinal and additional data types).

Following the descriptions above it is clear that different cluster algorithms could work well in different situations. For this study, LCA was the clear choice because of the mixture of input data (e.g., categorical and continuous), a desire to allow for nonspherical clusters, to have a transparent view of the cluster makeup, and for the ability to statistically compare different models.

**Web Usage Mining**

Knowledge discovery research in the arena of the Web has generally taken place in one of three areas (Chen & Chau, 2004): (a) web structure mining, also called webometrics or cybermetrics (e.g., looking at interconnectivity based upon hyperlinks, Björneborn & Ingwersen, 2001); (b) web content mining (e.g., for search engine or digital library development, D. Kim, Jung, & Lee, 2003); and (c) web usage mining (WUM) that is sometimes synonymous with web metrics, webmetrics, or web analytics (e.g., human-computer interaction and user behaviors, Eirinaki & Vazirgiannis, 2003; Y. S. Kim, 2007).

As user behavior is the purpose of this study, WUM was the focus of this review. The focus of WUM is on user actions in an online environment as recorded in tracking and transaction data, web metrics, and other sources. While more granular data are available for page-level and -design analyses (Atterer, Wnuk, & Schmidt, 2006), the
data are generally collected from server logs, relational databases, or some other long-term storage (Cooley, Mobasher, & Srivastava, 1999; Kosala & Blockeel, 2000). Server logs generally consist of time and date stamps, a session identifier, client Internet address (IP), the requested uniform resource indicator (URI—used synonymously in this dissertation with URL), and at times, the URL of the referring page (each web server has its own method of configuration, e.g., Apache Software Foundation, The, 2005).

In conjunction with server logs, backend application relational databases are used to collect returning user information such as name, address, user preferences, and user visit history. Analyses of these data can answer a variety of questions about user patterns, interface changes, and even user intentions (when combined with triangulating data) within the context of particular parts, or sites, of the Web (e.g., Björneborn & Ingwersen, 2001; Cooley, 2003; Han & Kamber, 2006).

Each entity that implements web-based tracking and analysis methods generally progresses iteratively in asking questions and finding answers before settling on what data, questions, and analyses are most meaningful for that organization’s goals (for a how-to and example, see Sparks, 2010). As answers to questions are realized, additional questions that can be asked of the data are envisioned and the data mining process can start again. Many sites start with questions that can be answered with simple web metric analyses (such as hit count, traffic sources, and frequency of visit, e.g., Recker & Palmer, 2006), but over time, more in-depth questions require increased sophistication in analyses and methods (sequence of pageviews, click heatmaps, and segmentation of site according to purpose, measuring learning, e.g., Hwang et al., 2004).
Some conditions make utilizing web metrics interesting and challenging, including the following (specific terms are more fully described in Appendix A).

- Multiple meanings, misuse, or confusion of terms (e.g., page views vs. requests or session vs. visit vs. unique visit).
- Incomplete data (e.g., the length of view for the last page visited because an end-time is generally not available).
- Network address translation (NAT, which confounds computer addressing).
- Browser configuration (disabled JavaScript or cookies, screen resolution, etc.).
- Nonnormal (or non-Gaussian) distributions.

Furthermore, key metrics are still being discovered, and what is a key for understanding one site’s goal-oriented patterns may not work well for another site, thus underscoring the uniqueness of each implementation (Khoo et al., 2008). For example, a high bounce rate—the number of single-page visits divided by the total number of page-visits—for a blog is normal, but is undesirable on a product page. Notwithstanding these difficulties, web metrics have proven very valuable in understanding user behaviors (Cooley, 2003; Eirinaki & Vazirgiannis, 2003; Y. S. Kim, 2007; Ye, 2003).

The end result of WUM is an improved probability of a goal. To illustrate this, consider the implications of understanding user activity in relation to business goals (purchases), which are quite important to e-commerce websites (Cooley, 2003). A situation where customers place a product in their virtual shopping cart and proceed to checkout but do not complete the process is interpreted as lost revenue. Investigation of the user experience as recorded in the server logs can indicate at which point the customer
is leaving, which in turn leads to additional investigation and experimentation upon the process to determine the likely causes of abandonment.

Analogous goals, questions, and iterative experimentation can be found in an educational website. Translation of business goals for educational purposes could include replacing purchase paths with a learning path or perhaps changing marketing campaign to course offering, thus making WUM applicable for an educational site.

End goals for a website could be any useful measure of site success. Specific questions should be formulated whose answers will assess or investigate such things as: achievement of each goal, the appropriateness of the data and its ability to measure the goal, and whether additional data should be collected. To illustrate, Table 1 displays four hypothetical goals and associated questions to answer and relevant data that may be essential to finding answers to questions and attaining goals.

As with traditional research methods, WUM data collection should span a sufficient time span to ensure capture of relevant behaviors, or enough to answer the question at hand. Some data may already be waiting in a data warehouse (or be converted into a warehouse) which can answer questions about the current state of the system while other data collection and analyses will be planned for evaluation of any system changes.

**Educational Data Mining**

The availability and use of KDD, data mining, and WUM techniques has ignited a quest to conduct meaningful analyses of available educational data using these modern tools (Baker & Yacef, 2009; Romero & Ventura, 2007).
### Table 1

**Example Goals, Questions, and Data for Educational Web Usage Mining**

<table>
<thead>
<tr>
<th>Goal</th>
<th>Questions</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce assignment attrition.</td>
<td>At what point(s) do students stop responding?</td>
<td><em>transaction-funnel-clickstream</em>&lt;sup&gt;a&lt;/sup&gt; and <em>exit-pages</em>.</td>
</tr>
<tr>
<td>Evaluate and improve accessibility of the user interface.</td>
<td>Do navigation paths show overly complicated menus? What does increasing the visibility of navigation shortcuts do to path complexity?</td>
<td><em>clickstream</em> and <em>link-tracking</em>.</td>
</tr>
<tr>
<td>Plan and evaluate supplemental instructional activities.</td>
<td>What segments exist in the visitor population? What activities were most successful given the characteristics of each segment?</td>
<td><em>user-profile</em> and <em>link-tracking</em>.</td>
</tr>
<tr>
<td>Evaluate lesson content.</td>
<td>From where did visitors link to a content page? How long do students stay on a page? Where do they go after viewing the content? Can we deduce that students were able to find the content they wanted or needed?</td>
<td><em>traffic-source</em>, <em>time-on-page</em>, <em>exit-page</em>, <em>clickstream</em>, and <em>bounce-rate</em>.</td>
</tr>
</tbody>
</table>

<sup>a</sup>Definitions for *italicized* words can be found in Appendix A.

**Educational data mining** (EDM) for has been gaining interest and credibility in special topics, special interest groups, and workshops at other conferences (e.g., [http://www.educationaldatamining.org/IEDMS/events](http://www.educationaldatamining.org/IEDMS/events)) lists the following pre-2008 events: International Conference on Intelligent Tutoring Systems [ITS], International Conference on Artificial Intelligence in Education [AIED], International Conference on User Modeling [UM], and the European Conference on Technology Enhanced Learning).
The emergence of EDM as a field of study was marked by new conferences, publications, and an organization. The First International Conference on Educational Data Mining was held in 2008 and the Fourth in 2011 (see the URL above to events); the Journal of Educational Data Mining made its debut in the fall of 2009 and the second volume was published in December of 2010 (http://www.educationaldatamining.org/JEDM/); the first edition of the Handbook of Educational Data Mining has also been published (Romero et al., 2010); and the International Educational Data Mining Society was officially founded in July 2011 (http://educationaldatamining.org/).

As a new field of study, EDM has ample need for additional research into how KDD can be applied in education since most data mining in education, as early as 1995, has focused in the areas of intelligent tutors and institutional research. According to Romero and Ventura (2007), the field needs more specific techniques that can be applied by educators in order to mature.

With today’s computer-based learning environments, student and instructor behavior are being captured but not always combined and analyzed to improve the educational experience (Hwang et al., 2004; Romero & Ventura, 2007). The following will present general information and knowledge in the literature regarding EDM that informed this study.

Finding EDM Literature

At the commencement of this research in 2008 searches for EDM-relevant literature were conducted with Google Scholar (http://scholar.google.com) and the
Education Resources Information Center (http://ERIC.ed.gov/) database with little success. An interesting phenomenon was noted: while “data mining” brought too many results for a focused review, most databases returned very few relevant results for “educational data mining.” For example, searching with ProQuest’s Digital Dissertations returned 55 results for unquoted “educational data mining,” only two of which were relevant to this study. The disparity of results was a bit surprising even with the recency of KDD or data mining application in education, and further underscores the need for much more research to explore the capabilities of these methods in furthering understanding of education and learning.

The ERIC was determined to contain a greater proportion of reliable educational research (with 116 results for the unquoted string “data mining” in the spring of 2008), therefore it was used for the foundational review. Google Scholar had the next best relevant hit count (with 171 results for the quoted “educational data mining” query) and was used to fill gaps or find relevant articles with a large number of citations. Using ERIC is also justified by the likelihood of other educational researchers using ERIC as a source for other educational literature and even further illustrates the need for wider publication and dissemination of the KDD efforts in education.

Some EDM fields of inquiry were relatively well represented in the literature (e.g., intelligent tutors, student profile analysis). This search in the ERIC database on “data mining” (unquoted) revealed that of the 116 results mentioned above, 30 were on general data mining or KDD descriptions, 31 were empirical studies, and 35 reported on case
studies that utilized data mining. The remaining 20 contained only casual mentions of data mining.

Romero and Ventura (2007) have surveyed EDM in various settings from 1995 to 2005. In general terms, the survey compares EDM constructs for online education with business purposes and definitions—noting that similar methods are used, but with some explicit differences in domain, data, objective, and techniques (Romero & Ventura, 2007, p. 136). It is worthwhile to note that even within EDM differences abound. In the following, Romero’s comparison of business and EDM are summarized for later comparison with the literature.


- **Data**—Business tends to utilize server logs exclusively while educational sites have additional user information.

- **Objective**—Business has explicit, tangible metrics (e.g., number of sales) where education objectives are often subjective and difficult to measure.

- **Technique**—Foundational characteristics between business and education make some techniques relevant, otherwise, new techniques are created.

The foundational review was conducted by focusing on references with similarity to this study in terms of domain, data, objectives, and techniques. Twelve somewhat relevant references remained with varied objectives and domains. While the objectives and domains varied, they were able to inform this study because of the kinds of data (web-usage) or methods used (within KDD).
The following sections will briefly highlight the domains of profile analysis and user behavior of students, intelligent tutoring, and institutional research, while educational digital library impact and evaluation will be covered in more depth in a following section as the domain of this research. The domain and data of this study will be covered while introducing the case in a later section.

**Profile Analysis and User Behavior**

Romero and Ventura (2007) and Merceron and Yacef (2005) both reported that EDM has been applied to user profiles of an educational system’s users—as learners. One example is Hwang et al. (2004) where Newton’s laws of motion were used to describe different learning rates discerned through data analysis of user profiles.

The domain of profile analysis would also include course management system data analysis and is largely concerned with the objective of learner progress through a system. The data can be with web metrics where client/server communications can be tracked, but also can be very fine grained and collected in highly controlled learning environments (e.g., eye-tracking or physiological state data collected in a laboratory). Analysis techniques can often be used that are supervised or predictive in nature with this domain because the course or assignment outcome can be known at the end of the course before mining.

**Intelligent Tutor Research**

Adaptive programs use EDM to customize tutoring sessions through artificial intelligence driven by rule learning algorithms. Carnegie Mellon University’s Human
Computer Interaction Institute and the Pittsburgh Science in Learning Center group applied EDM to ascertain if students are, among other things, *gaming* the system by examining user interaction with the system (Baker, Corbett, & Koedinger, 2004; Lee, 2007; Merceron & Yacef, 2005). New studies are ensuing to detect the moment a student has gained mastery or insight in order to encourage stable and deep learning (Baker, Goldstein, & Heffernan, 2011).

The domain of intelligent tutors is understanding the learning process with objectives to enable more efficient and effective learning. The techniques are able to be supervised since the outcome of learning can be measured (in quizzes and tests). Often these intelligent tutors are client programs that communicate with a server, thus producing some of the same transactional data as a web-based applications.

Lessons learned from intelligent tutor research indicate that user behavior can be discovered and that interventions can be designed and applied to help users be more effective in their use of the application.

**Institutional Research**

Traditional brick-and-mortar institutions are exploring aspects of student retention through EDM (e.g., Herzog, 2006; Luan, 2002), analogous to the retention of customers (or, reducing *churn*, e.g., Wei & Chiu, 2002). Luan and Zhao (2006) have been working with traditional institutions and EDM to better understand student retention. A series of articles were produced which show a progression of thought, methods, and techniques in EDM: from data gathering, warehousing, and application theory (Luan, 2002; Luan &
through decision tree analysis of learning outcomes (Luan, 2003); to discovering new patterns of students by observation of behaviors that affect student records (in lieu of traditional demographics) via traditional clustering (Luan, 2004). In the end, teaching others how to utilize EDM to understand student retention (Luan & Zhao, 2006; Zhao & Luan, 2006).

The research objectives above have been focused on characterizing a population in the domain of student retention (analogous to customer loyalty in the business domain). The data has generally had some specific outcome to which input variables could be matched (i.e., he used supervised learning).

In a single report Luan (2004), a traditional unsupervised learning algorithm (k-means) was employed. The use of the traditional clustering was an attempt to find previously unknown meaningful segments in the population that could be better understood and perhaps aided more effectively through cluster-specific interventions.

**Recent Literature Additions**

Between the commencement of this research in 2008 and its culmination in 2011, several relevant works have been published that expand and compliment the work detailed in this dissertation. Baker and Yacef (2009) conducted an updated review of the state of the field of EDM and found that four main purposes compose the majority of EDM applications.

1. Finding individual differences among students.

2. Developing and enhancing domain structure knowledge.
3. Studying pedagogical support effects in the learning process.

4. Investigation and measurement of instructional models.

Additional changes reveal that from 2005–2009 there were more studies with EDM than before in the same amount of time, and an added methodology of discovery through models (where the output of an initial model are mined for additional understanding and development of predictive models). Student modeling is still the focus and teachers are still rarely modeled.

From 2008–2009, Baker and Yacef (2009) also found that more prediction purposed papers were published and new group of psychometric researchers were using EDM and publishing within the field. Another development in the field was that often very large public domain datasets were being mined for relevant patterns. This last development has implications for increased application of EDM where local data may be tied to national-level data to determine impact of discoveries.

One example of the discovery through models methodology is where Amershi and Conati (2009) combined supervised and unsupervised models to cluster student interaction and domain exploratory behaviors. They utilized logged interface interaction and eye-tracking data as students explored new knowledge in an environment free from imposed instructional structure (Amershi & Conati, 2009). This research was still in its beginning phases, but explored a framework for modeling user behaviors within an educational environment.

The user modeling framework Amershi and Conati (2009) employed unsupervised clustering to find user behavior groups and then supervised methods to classify and
predict group membership of the users. The end goal was to influence the development of
the learning environment by detecting user behavior classes in realtime and adapting the
environment to encourage or redirect user behavior based on expert knowledge of
productive or unproductive exploratory approaches. This is not unlike the work in Luan
(2004) mentioned above but with learning environment data in lieu of student retention
data.

Additional studies utilized unsupervised methodologies in building user models.
Hidden Markov modeling was used by Jeong and Biswas (2008) in a learning-by-teaching
environment and by Shih, Koedinger, and Scheines (2010) tying results to student
outcomes while finding patterns of user behavior. Hardof-Jaffe, Hershkovitz, Abu-Kishk,
Bergman, and Nachmias (2009) used traditional $k$–means clustering to explore how
students organize their network storage spaces to determine organizational behaviors.
Other cluster studies explored other methods of analysis or presentation of results (e.g.,
Shanabrook, Cooper, Woolf, & Arroyo, 2010).

Several papers and posters presented at the Third International Conference of
Educational Data Mining in 2010 focused on teacher curriculum planning behaviors and
are directly related to the research in this study (e.g., Maull et al., 2010; Xu, 2011; Xu &
Recker, 2010). These papers will be covered in more depth while setting the context of
this study below.

A final note toward the increasing interest and availability of direction for EDM
researchers is the compiling of lists that assist with data collection (International
Educational Data Mining Society, n.d.-b) and the process of cleaning and analyzing data
(International Educational Data Mining Society, n.d.-a). Other organizations are related to the work in EDM and are beginning to contribute to the expansion of the field (for a list, see http://educationaldatamining.org/relatedorgs).

**Criticism of Data Mining**

One of the most frequent criticisms of data mining is the sole reliance on behavior analysis in making strong statements about intent, often without ever confirming these statements through other data sources and analyses (Zhao & Luan, 2006). For example, data indicate that frequent business travelers and terrorists have several common air-travel preferences. Simply acting alone on a preference for isle seats near the front of the aircraft and traveling with no checked baggage, legitimate business travelers would be likely subjects of investigation (Gilden, 2006).

An educational example highlights the need for triangulating data. Specifically, when end-of-year tests are used exclusively for the next year's decisions, it “is like driving a school bus looking out the rearview mirror. I can see where my students have been but I cannot see where we are going” (Salpeter, 2004, p. 1). While all data collection peers into the past, it is important to also take into account the current situation of students in determining approaches to instruction.

In spite of these challenges, EDM is flourishing and gaining wider acceptance as another wide-ranging tool with which to observe, measure, and aid the learning process. Specifically when coupled with multiple data sources from multiple time periods, sensible knowledge may be gained and applied to the benefit of the student.
Educational Digital Libraries and Data Mining

The idea of *educational digital libraries* was hailed as a breakthrough for broader and more substantive distributed learning opportunities (Marchionini & Maurer, 1995). Efforts to create educational digital libraries have been wide and varied but with one overarching goal: seek to create or find high quality learning resources (generally online) and catalog them for convenient discovery by teachers and students alike (e.g., Kastens et al., 2005). A last, in lieu of a one-size-fits-all textbook, teachers would have the option of finding online learning resources that could be annotated and sequenced as regular curricular activities or referenced as supplemental information (Recker, Dorward, Dawson, Mao, et al., 2005).

Because educational digital libraries are essentially large databases of resources and their associated *metadata*, the library collections are generally only available via the Internet. Some libraries (e.g., the National Library of Virtual Manipulatives, NLVM.usu.edu) actually house their own resources for users to directly access and are referred to as resource repositories. Other libraries harvest the metadata of many libraries and serve as clearinghouses (or metadata repositories) from which searches may be more efficient (e.g., NSDL). This second type of library is like a web search engine that gathers data about the data in different websites but makes searching much more convenient for the end user.

Often, the only end-user-friendly way to access educational digital library holdings is through the Web making it a natural fit for web mining.
Two of the three common types of web mining were heavily emphasized during the construction and maturation of educational digital libraries: structure and content mining (e.g., Diekema & Chen, 2005; Lagoze, Krafft, Jesuroga, et al., 2005; Lagoze, Krafft, Cornwell, Dushay, et al., 2006; Lagoze, Krafft, Cornwell, Eckstrom, et al., 2006). Mining the structure of resources and cataloging and how each resource links to other resources helped to create a better understanding of how educational resources interoperate. Content mining was used to find the attributes and information within resources that would cause the library to not only retrieve what was relevant, but also not retrieve what was not.

As late as 2006, most of the educational digital library end-user resource usage studies had been accomplished via traditional methods (e.g., survey and direct observation). Even though WUM is a natural fit for end-user studies of educational digital libraries, it has taken time for the field to understand something of user behaviors in order to incorporate more web metric and WUM techniques later (Bartolo et al., 2006; Bishop, 1998; Choudhury, Hobbs, Lorie, & Flores, 2002).

Evaluation and impact studies of educational digital library utilizing web metrics and EDM began with usage and usability questions (e.g., Bishop, 1998) and have progressed with increased sophistication toward understanding and personalization for segments of users (e.g., Khoo et al., 2008). While there is still relatively little EDM done with educational digital libraries, interested is growing as highlighted by a recent call for more adoption of e-business marketing and patron retention activities (Gider, 2007).
This pattern of traditional to data mining progression in research is not without precedence in the digital library arena as demonstrated by the following two examples.

A prominent example of digital library improvement using end-user experience data with traditional methods is the well-known research of Druin (2005) where children were intensely involved in the planning and development of the International Children’s Digital Library. Focus groups and development iterations were used to fine-tune the digital library for the specific needs and capabilities of the users. Recent evaluations have utilized web-based data through iPhone and other mobile phone use to study and improve access and user interaction (International Children’s Digital Library Foundation, 2008).

Another example of using KDD techniques following traditional methodology is the improvement of a research digital library end-user experience as reported by Bollen and Luce (2002). Originally, library subjects were organized to mirror subjects of job descriptions within the organizational structure—assuming that each job had a set of related documents that were rarely shared across organizations. In the newer research where document retrieval pattern clusters were examined, documents that were requested in proximity to each other were interpreted to be part of implicit classes of knowledge needs within the organization. In the end, the digital library administrators were able to improve library structure and holdings based upon implicit communities of users and knowledge requirements rather than the explicit organizational structure.

Typical difficulties in applying WUM also appear in educational digital library use of web data, such as slippery definitions, standards, and connecting or integrating data sources (i.e., log file information and site surveys Harley & Henke, 2007). The dispersed
nature of data sources is particularly difficult while researching large clearinghouse-type
digital libraries such as the NSDL because the search and linking from one Internet
domain results in usage from another without knowing how that link was used.

As an example of this difficulty, suppose the example of a user searching for “fish”
would find many resources listed in the NSDL, but the actual content is spread across
many other sites. While the NSDL was able to capture the search process and even detect
which links are clicked on, once a link has been copied or followed to the other site, the
NSDL will have no knowledge about what the user does at that site. Perhaps a teacher
found the link and pasted it into a learning activity document that was given to students in
a lab or as homework. The NSDL will not have known if or how the link was used by the
teacher or students.

The destination library that holds the actual resource will not have access to the
search history, but will have known if a link was taken directly from the NSDL to their
resource by observing the referrer in the page request information. If the resource visit has
come from a document or bookmark, there would be no referrer and no knowledge that
the visit came from the NSDL. The destination library, however, would be able to have
captured much information about how that resource was used by the end user—but there
may still be some question about whether the visit was a teacher or student.

Neither the metadata-harvesting library nor destination library (resource
repository) can really ascertain the learning context for which the resource was used. Was
this a teacher who introduced, presented, and followed up on the resource with their
students? Or was it a student doing independent learning? These questions have been
pursued by several NSDL-funded or partner projects where users have been afforded the ability to create new learning resources for their students. In such environments it was possible to fill the gap between the clearinghouse digital library and the learning resource repository. Such digital library end-user tools have been recognized as repositories in their own right and have been tapped by the NSDL as additional sources of metadata.

Examples of educational digital library end-user tools are many, only two are mentioned here. The Multimedia Educational Resource for Learning and Online Teaching (www.MERLOT.org) has been very active and has been focused on higher education audiences and has had a mission to encourage peer review of the generated learning resources. An example that has targeted the K-12 arena is the IA where teachers can situate online learning resources in a context (or project) and then share the new resource with their students via a special passphrase or share them openly on the IA website.

As noted above in the discussion on EDM, most research has focused on students with little done to assist and improve teachers’ experiences and capabilities through mining behavioral data. The following section introduces the IA as an EDM research context in which teacher use of digital library resources in learning activities can be observed while accessing and preparing educational digital library content for student use. Additionally, there are indicators in this context that suggest student use of resources.

Mining teacher behavior while developing online learning resources that utilize other online learning resources begin to answer the questions above that lie at the intersection of aggregation digital libraries and learning resource repositories. Knowledge of these behaviors can not only help improve the end-user authoring tools, but can also
inform both metadata repositories and resource repositories regarding their collections and contexts in which their holdings are used.

**Progressive EDM and the Instructional Architect**

The IA ([http://ia.usu.edu](http://ia.usu.edu), see Figure 1) has been described as a simple, K-12 targeted, educational digital library end-user authoring service where teachers can collect, sequence, annotate, and share (or publish) online learning resources with their students and other teachers in the form of *projects* (Recker, 2006). An IA project consists of teacher-generated prose (annotation) intermingled with links to online learning resources and is the instructional centerpiece of the IA (see Figure 2).

![Image of the Instructional Architect home page](image)

*Figure 1. The Instructional Architect (IA) home page.*
Figure 2. A sample IA project designed to teach children about the brain. There is reference to a (presumably) paper-based worksheet that the students must complete, so this project may be intended to augment and not supplant other instruction.

The IA was built using funding from the National Science Foundation (http://NSF.gov) and in connection with the NSDL to provide context for researching teacher use of online learning resources, hence, it was important to analyze the various data collected in conjunction with user activity. Traditional modes of inquiry (e.g., observation, survey, reflective journals) have thus far constituted the bulk of analysis with respect the IA usage. Recently, however, web metrics analyses have been used in increasing proportion because they contain valuable information about user activity (Khoo et al., 2008; Recker & Palmer, 2006). With data amassing in various forms the time had come to explore the usage of the IA through EDM.
This section briefly describes the Instructional Architect website and a meaningful activity framework from which user information may be interpreted. Descriptions of traditional IA research, initial web metric work, and more recent work that parallels this study follow.

System Description

From the home page of the IA (see Figure 1), users can (a) browse shared teacher-created content (or projects), (b) register as a new user, or (c) login as a registered user or as a guest. Guests can do everything a registered user can do but do not have exclusive control over projects and resource collection, cannot publish projects to students or publicly to anyone, and guest has no student login as well.

Browse. Users can access teacher-created projects that are shared with everyone (when published publicly) by performing keyword searches or by browsing projects by subject area, grade level, author’s last name, or project title. As noted by Recker (2006), it is apparent from a cursory examination of IA projects that “…teacher-created projects are fairly simple. Teachers are not web designers, nor should we expect them to be. Instead, they are professionals attempting to efficiently and effectively address classroom and learning issues” (Recker, 2006).

As teachers often share and modify their teaching resources in the tangible world, the ability to copy publicly browsable projects was added to the IA browse tool in the electronic world. Each publicly published project is released under a Creative Commons
Attribution license (http://creativecommons.org/about/licenses/) that is maintained as copied projects contain a link to the parent project.

**Register.** Users can create a free account, which provides them exclusive access to, and control over, their saved online resources and IA projects. They also can provide their target grade levels, subjects, how they were introduced to the IA (by workshops, a class, conference presentation, or word of mouth). Later improvements also ask for information regarding technology ability and zip code (Xu, 2011). It is on their registration/profile page where users can set up a passphrase which students can use to access a specific set of projects as setup by the user (see description of the My Projects Tool below).

**Login.** After the user logs in, the IA offers two major usage modes to work with online resources and the user’s projects.

**My Resources tool.** First, with the My Resources tool, users can collect resources from the NSDL, IA, and the Web. Within the integrated NSDL search interface (Lagoze, Krafft, Cornwell, Eckstrom, et al., 2006), queries are sent to the NSDL REST search interface and results (a set of metadata records) are displayed in an abbreviated form for perusal and selection. Additionally, publicly available IA projects can be browsed (as described above) and saved as resources. Finally, users can also add any Web resource by entering its URL. (Of course, these Web resources do not have associated metadata records.) Users can also organize their saved resources into folders.

**My Projects tool.** Second, with the My Projects tool, users can create webpages in which they sequence and annotate their selected resources in order to create
instructional projects. Users can also add basic metadata about their projects, including subject area, grade level, and core curriculum standard. These metadata values are used to support the IA’s project browse and search.

Finally, users can share their projects by Publishing them and setting permissions on who can view them: (a) author only; (b) student view, via a user-created passphrase as entered during registration; or (c) public view, anyone browsing the IA site and also grants permission for other users to copy and modify the newly copied project.

**Meaningful User Activity**

There are several activities that IA users perform that are more significant than others. These meaningful activities have been explored in several of the IA research activities to-date and are presented here so the past research described below is better contextualized. Each activity is manifested in multiple ways: first, with traditional education research methods through teacher self-report, interview, or direct observation; and second, as logged user activity (or digital footprints) that provide an accurate account of each user action which is then available as a feature for traditional analyses or application of EDM. Specifically, the meaningful IA user activities are as follows.

- Registration and subsequent visits.
- Collection of online resources.
- Annotation and sequencing of collected resources in a new online educational resource (i.e., project authoring).
- Use of the authored resources.
**Traditional IA Research**

Past research of the IA and its users began with user-centered design—using focus groups to gain feedback on the user interface and functionality of the tool. In 2003 when professional development workshops were introduced as a way of teaching users about digital libraries and the use of the IA, observations of the workshop experience as well as focus groups guided further workshop and IA development (Recker et al., 2004).

Research efforts since then have focused upon measuring and improving the impact of the workshop. Data sources have included direct observation of the use of IA projects in classrooms and labs, group discussions, individual interviews, and pre- and exit-surveys. Analyses of these data found that attitudes surrounding the IA and online resource use in the classroom are generally positive and increase with exposure and use. Behaviors (specifically recent teacher use of online learning resources with students) have also changed in conjunction with the professional development workshop—at least in the short term of the workshop (Recker et al., 2006; Recker, Dorward, Dawson, Mao, et al., 2005).

Other findings include that not all teachers adopt this new technology (Recker, Dorward, Dawson, Halioris, et al., 2005), technological aptitude may differentiate patterns of use and satisfaction with the IA (Mao, 2007), and that digital library resource grain size may affect the mode of resource use (Recker et al., 2007). The most recent thrust of research has explored the effect of a problem-based pedagogical approach has upon workshop participants and their use of the IA, online resources, and design decisions (Walker et al., 2010).
Many of the IA Team’s papers and publications are available at http://digitalcommons.usu.edu/iagroup/.

Web Metric IA Research

In order to more fully understand how teachers are designing with and annotating original resources, IA user behavior data has been collected in progressively greater detail since beginning operation in 2002. Initially, marketing questions drove investigation and analysis of IA user data (e.g., “Do we have users?”) and subsequently motivations included questions to better understand our users (e.g., “What are they doing?”).

Results of these web metric analyses have been regularly published (see Khoo et al., 2008; Mao, 2007; Recker, 2006; Recker, Dorward, Dawson, Halioris, et al., 2005; Recker, Dorward, Dawson, Mao, et al., 2005; Recker et al., 2006, 2007; Recker & Palmer, 2006). Nearly all of these investigations have reported focused on findings from data collected in traditional forms just described enhanced somewhat with some aspect of user web data analysis. While each analysis has become more sophisticated and meaningful, all have still been on small groups of users in a cross-sectional manner (i.e., snapshots of users at the end of the short time period of a workshop) rather than a broader, user-lifetime view of user behavior.

The rest of this section will describe the different sources of IA data that have been collected and review several studies on this information that have been performed during the term of the current study.
IA data sources. Three methods of usage data collection were implemented over time in order to gain greater intelligence of user activity: web server logs, Google Analytics, and a custom user-tracking system. Each data collection method was implemented for different reasons but in the end could be used for some kind of user behavior analyses. Each method has strengths and weaknesses as described below. A summary comparing the different kinds of data collected within each source is contained in Table 2.

Web-server logs. Web server logs of the IA were kept as a record of electronic transactions with the IA server. Originally these logs were used for streamlining development (e.g., reducing the size of each page), monitoring the system health (e.g., unexpected errors), and security (e.g., exploitation attempts). When it was realized that

### Table 2
A Comparison of Metrics Available Within Each Data Source

<table>
<thead>
<tr>
<th>Metric</th>
<th>IA relational database</th>
<th>Web server log</th>
<th>Google Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTTP Request</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Student Login ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Request Date/Time</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTTP Status Code</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Size of Response</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Geo-Location</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note. Metrics are listed in descending order of significance to this study. A darker background indicates those data collected over the life of the source. A lighter background indicates that the data are available for a shorter time and that the information cannot be reconstructed in detail. Cells with no color indicate the absence of data from the source.
user behavior observation was possible through processing and analysis of these logs, an
effort was made to both save the logs and configure them to be as complete as possible.

In the default setup, server logs are overwritten so they do not become too large
and begin to consume too many system resources. Rotating logs is one technique for
limiting the size of the active log file while allowing a history to be kept. At the same
time, if certain information is excluded from logging activity, such as requests for images,
style sheets, and other browser requests that are not essential to determining user behavior,
then the size of the log can be significantly reduced. Both of these techniques were
employed with the IA’s server log.

In order to gather as much information as possible, the logs were configured to
record the IP address of the requesting machine, the time of the request, the requested
script, the referring URL (if any), the time of the request, the server status upon
attempting to fill the request, and the client browser identification string (for configuration
options, see Apache Software Foundation, The, 2005, see also Figure 3).

Server log analyses are limited in two respects: granularity and session
disambiguation. The granularity of the information is often at the page level. For instance,
if the HTML form submission method is GET, then the form information is embedded in
the URL and granularity increases. Using the HTML POST method does not show form
information in the URL and, therefore, it is not logged; and if there is opportunity for the
user to perform multiple actions on a single page, then it is quite difficult to know for
certain which activity was actually performed.
Figure 3. An example web server log from the Instructional Architect (IA). Note: IP addresses and project identifiers have been obscured for privacy purposes.
Additionally, the resolution of a user can be limited by the use of IP addresses as identifiers. For security and other reasons (e.g., availability of IPv4 addresses), computer networks often use the same IP address on the Internet to represent multiple computers on the inside of that network. This type of setup is called network address translation (NAT). Furthermore, schools and libraries often use a proxy to filter their Internet traffic—sending all requests through one server that then represents the actual user under the guise of a single IP address—further hindering the accuracy of IP address use for user identification.

While there were instances where a successful login is required to visit certain URLs (e.g., an authoring page where the project identifier is associated with at most one user, such as http://ia.usu.edu/authorproject2.php?project=ia:10524); but the rest of the visit based upon IP address alone was confounded by the last two issues mentioned above.

**Google Analytics.** The metrics collected by Google Analytics since May 21, 2007 were from pageviews and link clicks via JavaScript (see http://google.com/analytics). Therefore all user, project, and resource information that was embedded in pageview and link URLs may be tracked—again excluded are data contained in the POST information, such as project content changes. Google Analytics graphs and reports have been very convenient tools for analyzing overall usage patterns—several of which were detected during related studies (e.g., Xu et al., 2010).

The strengths of Google Analytics are in its convenient and powerful reports. Figures 4 through 6 show trending of IA visits on three different levels of granularity. Other reports afford a visual analysis of trends for different web metrics such as unique
Figure 4. An example Google Analytics visit plot from May 2007 to January 2008. Depicted are the number of daily visits from the implementation of GA with the IA to the start of data collection for this study. Note the mixed cycles of weekly and US public school calendar ebbs and flows.
Figure 5. Three example Google Analytics visit plots from January 2008 to November 2009 with smoothing. Depicted are the number of daily (top), weekly (middle), and monthly (bottom) visits during the data collection period for this study. Note the mixed cycles of weekly and US public school calendar ebbs and flows, but as smoothing occurs (from top to bottom) those cycles are less evident.
Figure 6. An example Google Analytics geo-location report from January 2008 to November 2009. Depicted are visits from estimated US state location during data collection for this study. Note that the concentration of hits occur from states where the IA professional development workshops have been presented.
visitors, pageviews, and goal-tracking funnel information as users click through the progression of links in performing meaningful user activities. Each of these reports has been useful for determining how changes to the IA or linking from another site (such as the NSDL) affect usage of the site as well as the accomplishment of goals.

Geo-location analysis is another strength of Google Analytics (as displayed in Table 2). In short, Google Analytics calculates and saves a latitude and longitude for the IP address before discarding the address—making several different reports available, one of which informs the researcher as to which countries, states, or cities has provided the most activity. Figure 6 demonstrates this capability.

For all the strengths of Google Analytics, there is one outstanding weakness with respect to integration with other application data. Out of privacy concerns the Google Analytics data are generally available only as summaries and aggregated to visitors as a group, but not available for specific visitors. Even though user information was added to data sent to Google, integration with local project and demographic information would have eliminated the advantage to Google’s analytic tools—thus rendering it merely another data gathering tool.

**Custom user tracking: The IA relational database.** The Instructional Architect, as a web application, is implemented in PHP (http://PHP.net) with a PostgreSQL database (http://www.postgresql.org) where user registration, resource collection, and project authoring information are stored. This IA relational database (IARD) has increased in complexity and utility over time.
Initial user tracking consisted of the registration date, last login date, the creation and last modification dates for resources and projects. In other words, the only possible analyses could investigate little more than the number of users and most recent activities.

In the Fall of 2006, additional tracking was added for both saving the project history as well as project and resource views. This allowed deeper views into user activity as investigated for users participating in the workshops—as they received the majority of research focus. As analyses progressed it became apparent that user sessions were important for understanding the differences between user visits and investigating user behavior.

With added motivation for improving the IARD user tracking, on January 21, 2008, the IARD was modified to collect the page views associated with each PHP session identifier. These page-tracking variables include:

- user id,
- group id,
- PHP session id,
- IP address,
- page referrer,
- page request, and
- a comment as to the specific action taken (e.g., moved resource from folderId1 to folderId2).

Again, note that prior to that time, project hits and resource visits were collected along with previous project versions, but without any tie to the current user’s status. This means
that the IARD contained a mixture of user specific and near anonymous entries. With this additional data, users (or groups) could be analyzed within and across sessions and visits. Additionally, with the referrer and request information, it is possible to reconstruct clickstream data—useful for examining navigation behavior and improving site design.

Few refinements have come since early 2008, and data have been collecting ever since. It is important to note that the IARD contains an approximate superset of the information contained in the server logs alone. To provide a sense of the magnitude of use since January 2008, the page-tracking entries had 1,524,315 entries as of November 7, 2009. By contrast, the project history contained 47,792 project revisions from August 2006 until November 2009.

The IARD has persistent limitations. For example, in the case of hits on projects, it is not always known who is hitting the page. For public projects, no user or student login information is required to be associated with the project view and so each view is counted as a project view for the user’s account.

With student login information we know from the clickstream that somehow that visitor has obtained and entered the student login code chosen by the project creator; but whether it was the creator, their students, another teacher, or another teacher’s students with whom the student login was shared, we cannot determine. In cases such as these, the use of projects will be associated with the creator-user’s profile—in other words, project use will be reported in the project creator’s profile, even though it may have been another user actually viewing the project.
**IA data summary.** The data required to conduct a user behavior analysis consists of page requests and clickstream data that can be integrated with saved user input in the IA’s backend database (IARD). From what was shown earlier in Table 2, we see that the web server logs can provide the clickstream data based on IP address, but there is little to no information that would allow us to link that to the IARD. The table also shows that Google Analytics also could provide Geo-location where that would have to be calculated from the logs or IARD but integration with the IARD was also difficult. While location could have been a nice addition to the data, it would have significantly increased the complexity of data integration.

A parallel work noted that the application database with its enhancements was the superior data source for this study because it contained the same data but was marked with user identifiers, cleaner, better formatted, and easier to extract (Xu, 2011). While the server logs and Google Analytics contain valuable information for other purposes, in the end it was determined that for the purposes and questions pursued in this study that the IARD contained all of the necessary information and was the only data source utilized.

**Related teacher behavior pattern studies.** Prior to the application of web metric analysis and EDM to the IA user data, user studies over longer periods of time have been unrealistic due to time and cost constraints. However, through the use of automatic data collected on user behavior, IA use was tracked and ready for an application of KDD (the results of which can subsequently be verified through past and future inquiry) with little expense beyond researcher and server time.
In addition to the current study, there was an additional researcher utilizing the IA data for data mining as well as work on the Curriculum Customization Service project that is similar in nature to the IA. Several papers and posters were presented at the Third International Conference of Educational Data Mining in 2010 by these researchers that focused on teacher curriculum planning behaviors and are directly related to the research in this study (Maull et al., 2010; Recker et al., 2010; Xu & Recker, 2010). These other works are briefly reviewed and this study situated in that context.

Maull et al. (2010) work with the Curriculum Customization Service (CCS) that is integrated with the Digital Library for Earth Systems Education (DLESE.org) has been focused on US middle and high school earth science teachers. Like the IA, there is a professional development workshop where teachers are introduced to the CCS and how they can integrate it into their curriculum.

One of the assumptions in CCS research is that teachers feel overwhelmed while trying to select and align online resources to standards in creating new instructional material. Therefore, the CCS itself is structured around preexisting curriculum and goals but offers the teachers the ability to customize and supplement the online curriculum with personally chosen online learning activities.

The EDM focus was to identify patterns of use that could be explored and confirmed through further traditional methodologies. As appropriate, several usage features were chosen that allowed for preliminary analyses and experimentation based upon researcher expert knowledge of the system. The data collection window was approximately one year with almost 120 teachers from grades 6–9. Both $k$–means and
expectation-maximization (EM) algorithms were used (EM is closely related to the LCA used in this research) and produced similar results. This was the beginning of this research and subsequent publications will demonstrate and verify or contradict these findings.

Several papers have been published that investigated several ways of slicing and linking IA data with other data. Xu et al. (2010) examined how to deal with the volume of data and what different data points can mean in the study of digital libraries. Several views of the data in the form of visits and interpretation of meaning, watching visits change when dissemination strategies are employed, and also comparing the use of a digital library with other national demographic data to better understand an audience. This paper also explored different features and the methods of constructing additional features.

Recker et al. (2010) published a short paper that further looked at overlaying the demographic data mentioned above with the geo-location of IA users in order to more accurately examine one year of IA visits based on location and median family income. Population and per capita income were significant predictors in IA visits.

In Xu and Recker (2010), social network analyses were performed utilizing the project visit records and tracking use of the copy project feature of the IA. As described above, an IA project that is published to the public view may be viewed by other users and also copied. The copy can be modified by the copier—but an attribution link is maintained to the original project. The results revealed that project views do not yet seem to yield much knowledge of social network structures, but that the copying of projects identified several user groups with similar interests. Indeed, this was verified by their chosen subject matter in the registration information.
The remainder of this section on similar work is dedicated to dissertation work by Xu (2011). She was working on clustering IA user data simultaneously with the current study and while her work included documenting the KDD process and investigating the use of LCA as a suitable classification algorithm, she also looked at combining additional profile data with the results of the LCA. Xu focused on exploration and refinement of the data analysis and exploring the results.

In three parts, Xu’s work used LCA twice—the first as preliminary use and then a refinement of the clustering techniques—and then using frequent item set mining as a filter and smoothing of the LCA classes. The first LCA application resulted in seven classes of users: isolated islanders, lukewarm teachers, goal-oriented brokers, window shoppers, key brokers, beneficiaries, classroom practitioners, and dedicated sticky users.

In the second study Xu refined the features and model to find three clusters: key brokers (with 31% of users), insular classroom practitioners (with 32.8% of users), and ineffective islanders (with 36.2% of users). The third study used these three clusters combined with teacher demographics where teaching experience and technology knowledge showed some impact on the usage patterns of registered IA users—an instance of what Baker and Yacef (2009) called the methodology of discovery through models.

Xu’s conclusions are that LCA is a good tool for classifying users, but with advantages and disadvantages. Advantages are that a quick LCA analysis is very easy to do but the results can be difficult to interpret. Another approach was also attempted by pruning the results through subsequent analyses and proved time consuming However,
with the second approach the clusters are clear but many users may be eliminated from the study because of data sparseness.

**Synthesis of Literature**

From the above review it is plain to see that data mining is an interdisciplinary endeavor, no matter the subject matter domain to which it is applied. Every application of data mining will differ based upon domain, data, objective, and technique (Romero & Ventura, 2007).

While there are many descriptions, the general knowledge discovery (KDD) framework of (a) clean and integrate data sources, (b) select and transform data, (c) data mine for patterns, and (d) evaluate and present the results, is useful for organizing KDD efforts (Han & Kamber, 2006). While the data mining step often receives the most attention in books and articles, care must be taken to not underestimate how much time it takes for the first two preprocessing steps (Witten & Frank, 2005).

Techniques used in KDD are the result of combined effort from many fields. When traditional statistics break down due to the volume and complexity of modern computer-generated data, alternative approaches include machine learning (Hastie et al., 2001; Zhao & Luan, 2006, e.g., association rule mining) and newer statistical procedures such as recent extensions to LCA (see Muthén, 2004).

Data mining is a very broad field of study with many areas of application (Ye, 2003). Mining data available from usage of the Web is just one possible application, and there are yet three general divisions contained in web mining. An area of obvious
applicability to the objective of discerning user characteristics is WUM, which involves tracking user activity on the web for concurrent or subsequent analysis.

Another domain of relatively new application of data mining is the domain of educational data (or EDM). EDM has been used with both traditional as well as Internet-age data with the understanding users of educational websites as just one of many broad applications Romero and Ventura (2007). Most studies with EDM, however, have focused on the ability of students to learn on the web or with intelligent tutors, while few have focused on teacher use of technology.

Educational digital libraries have been developed utilizing some aspects of data mining but investigation into user behaviors has been difficult because of the dispersed nature of resources. Large repositories of metadata make searching for online learning resources more convenient, but while the large repository gathers data about the finding of resources, it is limited in information about how that resource was ultimately used. Likewise, while the online learning resource repository can know intimate details concerning resource use, they have little knowledge of how that resource was found by the user. At this intersection of repositories are end-user authoring tools which capture the context in which online learning resources are used with students.

Educational digital library user research has largely been traditional in nature (surveys, observation, etc.) but more recently they have begun to use web metrics to better understand their visitors. However, no work had been done in the scope of EDM.

This work, along with two other related projects, sought to begin filling the lack of educational digital library knowledge of contextual resource use by teachers and students.
as well as to further the use of EDM in the field of educational digital libraries. The remainder of this work will outline the Methods and report on the Results of applying KDD to the IA user data. The Conclusion revisits the research questions and contrasts are made with similar works, then finally, limitations and suggestions for future work are described.
CHAPTER III

METHODS

Purpose

The purpose of this study was to perform a research project in applying the knowledge discovery from data framework (KDD, see Han & Kamber, 2006; Vickery, 1997) to the field of educational data mining (EDM; Romero & Ventura, 2007). More specifically, this research extends previous EDM work by applying web usage mining (WUM) to categorize and characterize teacher use of a digital library end-user authoring service in order to better understand user behavior patterns that can have implications for tool development, professional development workshop planners, and educational digital library managers.

Educational digital library end-user authoring systems connect two kinds of libraries—the metadata repository and resource repository—by providing an observable learning context in which students are introduced to and work with online learning resources. The context for the study is the Instructional Architect (IA; http://ia.usu.edu), which was introduced in Chapter II as an end-user service where teachers can design new online learning activities using existing online resources (particularly those found at the National Science Digital Library; NSDL; http://NSDL.org).

The contextual framework for observing and interpreting user behaviors pertinent to this study focus on the following meaningful IA activities that are performed by
teachers as they design and use IA projects: (a) user registration and subsequent visits, (b) collection of online resources, (c) annotation and sequencing of collected resources in a new online educational resource (i.e., project authoring), and (d) use of authored resources by students and other users.

Three different data collection methods were described in Chapter II. For the current study only the IA Relational Database (IARD) was employed. Additionally, a reflexive journal was kept by the researcher in order to document the process, decisions, and lessons learned. Procedural decisions within the KDD process about data generation, integration, cleaning, as well as the analysis and interpretation were all based in the ideas from the contextual framework of meaningful IA activities as just described and emerged as the study proceeded.

The specific goals of this study were to:

1. Mine for patterns emerging from meaningful IA user activity data (pattern mining).
2. Characterize user behavioral patterns in plain terms (characterize).
3. Report on how data, methods, and tools were utilized for this study (report).

**Research Questions**

Each of the four goals for this study have parallel research questions and are displayed in Table 3 aligned with the data sources and planned analyses at each step.
### Study Overview

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Research question</th>
<th>Data source</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern mining</td>
<td>What patterns of user behavior emerge in terms of key features and metrics?</td>
<td>IARD</td>
<td>KDD with LCA, &amp; Visualization</td>
</tr>
<tr>
<td></td>
<td>Specifically, what patterns arise within meaningful IA activity?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Characterize</td>
<td>How can these patterns be described and characterized in ways that are germane to</td>
<td>KDD results</td>
<td>Qualitative interpretation and</td>
</tr>
<tr>
<td></td>
<td>the purposes of the IA?</td>
<td></td>
<td>description of patterns.</td>
</tr>
<tr>
<td>Report</td>
<td>How were data, methods, and tools used, and what are possible implications for</td>
<td>Journal &amp;</td>
<td>Qualitative analysis and</td>
</tr>
<tr>
<td></td>
<td>other online user behavior studies?</td>
<td>KDD results</td>
<td>generalization for implications.</td>
</tr>
<tr>
<td>Document</td>
<td>What general procedures emerge that are applicable to the study of educational</td>
<td>Journal</td>
<td>Qualitative analysis and</td>
</tr>
<tr>
<td></td>
<td>digital libraries?</td>
<td></td>
<td>generalization into procedures</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and considerations.</td>
</tr>
</tbody>
</table>

Each purpose and question are discussed below in more detail with their associated data and methodology. Note that each question seeks to clarify and utilize the results of preceding EDM processes. For example, one cannot characterize patterns until they have been identified in the data.

**Pattern Mining Methods**

The approach of the pattern mining portion of this study was a combination and extension of the KDD, WUM, and EDM concepts applied in other contexts, as described
in Chapter II. Specifically, the domain, data, objectives, and techniques of this study relate to the literature in the following ways (see Romero & Ventura, 2007, p. 136):

- **Domain:** The domain under consideration was teacher use and authoring of online learning resources in the context of educational digital libraries—much like past digital library web metrics, but at a level beyond merely reporting activity (e.g., Khoo et al., 2008) and with a more explicit framework (KDD and meaningful IA activity). Romero and Ventura (2007) and Merceron and Yacef (2005) report that EDM has been applied to user profiles of an educational system (as learners, e.g., Hwang et al., 2004) but not in the context of teacher professional development or teacher creation of curricular activities.

- **Data:** The data source for this study was similar to other WUM studies where a user database was employed with some demographic information and transaction records of meaningful activities. As mentioned above the IARD contains essential information also used in processing web server logs and Google Analytics information. The data are similar to recent educational digital library research (e.g., Khoo et al., 2008), but adding user profiles.

- **Objective:** The objective of this study was characterizing user behavior at different levels of activity, which is not unlike retention studies. Institutional research has explored aspects of student retention through EDM (e.g., Luan, 2002), but not online users. Customer characterization and retention research (a common topic in service industries, e.g., Wei & Chiu, 2002), has only recently
been applied to educational digital libraries and their services (Khoo et al., 2008).

- **Technique:** Applicable techniques to this study resulted in user characterization through the use of unsupervised machine learning algorithms and psychometric analysis. Most applications of EDM have been performed with supervised algorithms with known outcomes or classes (Romero & Ventura, 2007). A singular ERIC document was found that used traditional unsupervised classification (Luan, 2004), and none were found that utilized the more recent techniques of LCA (Muthén, 2004).

The specifics of pattern mining in the context of IA user data are now discussed within the framework of the KDD process that was introduced in Chapter II.

**Data Collection and Preprocessing**

After data collection occurred it was subjected to preprocessing (i.e., integration, cleaning, selection, and transformation) in order to be usable.

**Collection.** The IA relational database (IARD) was introduced in Chapter II as the backend database for the IA. The data from this source was incrementally improved to become the most granular data source about each user and their visit activity. This source collection has been relatively stable since January 2008. It should be noted that while the majority of web server logs and Google analytics data are also in the IARD, the IARD contains the most relevant and unique information for this study. The collection window was from January 21, 2008 to November 7, 2009.
**Integration.** Integration occurred by combining information spread across relational data tables with relevant keys using advanced structured query language (SQL), stored procedures, and some PHP scripts (http://php.net). First, a snapshot of the production database was created. Second, project publish and user group information were combined. Third the project history information for each edit for text-processing and comparison between versions were processed. Fourth, session clickstream information was compiled for each session. Fifth, and finally, overall session views were created by summarizing by counts and averages the session clicks, events, actions, project edits, etc. into a single record per session.

**Cleaning.** Inconsistencies and noise were known to exist in the IARD that required cleaning prior to continuation of the preprocessing. The a priori noise consisted of: (a) corrupt data (e.g., registration dates lost during an upgrade) and (b) noisy traffic (e.g., spam, web crawlers, and IA team-member accounts).

Scripts were written to look for and remove this noise where possible. Both the details and model algorithms are treated as results of the KDD process in the following chapter.

**Selection and transformation.** Selection and transformation are iterative steps and they are presented in the context of two main types or kinds. First, preprocessing methods for feature construction are presented; second, methods are outlined for selecting and further transformation (smoothing and generalizing, described in Chapter II) of data for actual analysis.
The desired outcome of preprocessing was the creation of a datamart or data warehouse from which data could be pulled as needed for subsequent analysis.

In order to better understand selection and transformation, it is important to first understand the terrain, depth, and interconnectedness of the data. The data terrain is presented here and followed by the methods of selection and transformation.

**Mapping the data terrain.** While only four meaningful activities are identified for this study, the number of features that indicated an occurrence of activity expanded to be quite numerous, hierarchical, and were moderately interconnected as illustrated in Figure 7. At the top level, the variables mirror the meaningful activities: (a) registration and visits, (b) resource collection, (c) project authoring, and (d) project use, which are then expanded into subcategories that are related to other IA data.

In Figure 7 the lines connecting leaf-nodes illustrate the interconnectedness of the data. For example, the number of resource visits is connected to all three sources of resources. The same connections are present with the number of resources in the projects node cluster. Many links are not drawn here to keep the graph somewhat understandable (e.g., another link could have been drawn between the number of public visits to the number of resources visited). Because of these connections, the number of variables necessary to examine each of these in detail multiplied quickly.

In the end, over 200 features were produced at the user-aggregated level (see Tables B2 through B5 in Appendix B. However, a parsimonious model was desired—hence one purpose of this study was discovering what subset of these features differentiate user behaviors into classes.
Figure 7. Data relationship map illustrating the interconnectedness of the IA user data. Nearly all meaningful activity areas were related to another in some way. The lighter shading for Subject and Audience indicates that these features were not used in the analyses.
**Preprocessing transformation and selection.** The selection of data for analysis began with data collection—meaning that certain data points were selected for collection and others were discarded as described below. In this case, Table 2 illustrates the basic information collected along with additional description in Chapter II and previously in this chapter.

Most of the transformation activity was in conjunction with selection of what specific features to extract (or transform) from clickstream or historical data that would indicate an enactment or manifestation of meaningful user activity as defined for this study (i.e., feature construction, smoothing, and generalizing), developing algorithms to accomplish the task, and then doing the actual transformations.

In general, much of the features would come from clickstream data, but there was a good amount collected in the user profile table about registration and visit information as well as in the project and resource tables associated with authoring, use, and collection activities. Methods for transforming these three areas of data are now covered.

**Registration and visit transformations.** Registration and visit data consisted of information such as registration date, grades and subjects taught, and where they heard of the IA. Registration data were largely gathered in the user profile table and did not require much work. Only the information about where they heard of the IA was transformed into one of two bins based on the amount of instruction and followup time included in the process: formal and casual. Workshops and college classes were considered formal—as multiple instructional situations with assignments and followup—and conferences, word
of mouth, etc. were binned as casual—or one-time instructional situations without assignments or followup.

Visit data were transformed into features such as how long has the user been registered, how long since the last visit, how long between registration and the last visit, and so on. Date calculations were used in the majority of these transformations.

**User activity transformations.** Transforming registered user activity data as well as project and resource use data into features of user behavior (feature construction) required SQL queries and PL/SQL scripting that could identify and interpret clickstreams (e.g., “was the saving of a resource a new resource or a modification of an old resource?”) and thereby provide information as to the meaningful IA activities enacted by the user. In other words, transforming the raw data into a record of meaningful activities occurred through decoding URLs, clickstream, dynamically created comments, and other collected data.

As an example, consider a clickstream sequence that was decoded into a usage variable that indicated a resource view was generated by a student-viewed project. The clickstream involved with this action is illustrated in Figure 8. While these clickstreams could provide valuable information it also had some limitations such as not knowing if the student login was performed by an actual student or the registered user/teacher using the student login in front of a class with a projector to display the IA project.

Additional features were created in association with the clickstream interpretation by combining IA application data associated with users, projects, and resources. Through this process it was possible to determine the following for every resource view.
Figure 8. A clickstream example illustrating the sequence of requests used to determine that a resource view was generated from a student viewing a project.

- From which source the viewed resource was collected (NSDL, IA, or Web).
- The login status of the viewer (public, a registered user, or a student).
- How the resource was accessed by the viewer (e.g., as an author from the resource collection or project construction pages, or as a project user from a project view).
- Construction of additional features gleaned from the metadata (e.g., search history, digital library source, technologies used in the resource, etc.) associated with each resource—these features were determined to be beyond the scope of this work and not calculated.

Project history and resource collection transformations. Another group of features that were constructed from the IA application data was associated with user projects and the resources they collected. These data were constructed for each project save, for each session, and finally aggregated to the user level.

For example, in addition to simple counts and data calculations, there were many features generated in conjunction with project histories using PHP to transform the raw project content with HTML code into data (e.g., content without HTML, content as a Soundex string [to remove spelling and phonetic differences], a word count, and a difference from the previous version of the content). The difference features and other
comparative features had to be calculated between saves as well as between the first and last states of a project for each session before being aggregated for the user.

An additional example was with resource collection. Each resource was added to the user’s collection through one of the following sources: the integrated NSDL search, the IA public projects, or as a separate web resource. Every resource was also used in the user’s projects zero or more times. The following features highlight each of the different area of resource collection and use:

- the total number of resources gathered,
- the percentage of gathered resources from each source,
- a count of resources used in projects of different publication status (public, private, and student),
- the percentage of used resources for each source, and
- a percentage of resource views that came from each source for each project view type.

Results from preprocessing integration, cleaning, selection, and transformation were finally available for use in many different kinds of analyses at a session-level in a data mart. Next was the final transformation and selection of features and participants to be used in an analysis.

**Transformation and selection for analysis.** This first attempt at EDM was simple and focused on an aggregation of longitudinal user behavior into sessions. However, because this project was looking at user-level behaviors as a beginning, additional transformations were necessary to aggregate information to the user level for
The multitude of features produced in preprocessing were counted, summed, converted into ratios, and/or averaged with some standard deviations kept all on the basis of sessions for a user.

A further transformation that has often been required would have transformed the data to fit the underlying assumptions of the subsequent analysis. Typically this would necessitate a linear transformation with a mathematical function (e.g., a log-transformation can make skewed data have a more normal shape). While such transformations may make analysis possible, it increases the complexity of interpretation that must be applied to the results. Fortunately, due to recent extensions developed for LCA, this final transformation was neither anticipated nor needed.

Finally, examination of the data identified outliers and decisions were made as to whether or not they should be dropped or transformed to be within certain parameters (e.g., the second standard deviation).

Features selected for analysis represented the four areas of the meaningful user activity framework and also contained enough data to be analyzed. For example, not all users had resources from all three sources (IA, Web, and NSDL) and features that were based on resource source (e.g., \( \text{PctProjWithNRes} \) or the percent of projects with NSDL resources) were discarded and the overall percentages, counts, and averages were used instead (e.g., \( \text{PctProjWithRes} \), or the percent of projects with resources).

As this was a study focused on users and not nonusers meaning that by removing all registrants who did not return or perform significant actions then they would have enough data to analyze. It was expected that the number of visits would be highly
correlated with many of the features—as additional visits would logically produce additional usage records—and could be used as a means of finding the most users with as many features as possible. The planned analysis (LCA) also had mechanisms that assisted with further feature selection and deselection and will be described in more detail below.

**Selection of participants.** The following criteria served in the final selection of participants as preliminary analyses indicated the need for segmentation. These criteria are in alignment with previous IA user studies (including self-selection into workshop participation and IA usage both as an authoring and instructional tool; e.g., Recker & Palmer, 2006; Recker et al., 2006; Walker et al., 2010).

- **Registered users**—only registered visitors were included.
- **Registration date**—only those users who registered between January 21, 2008 and November 7, 2009 were included (given the tracking improvement in January 2008, it follows to only include users for whom we have complete data and the end was when data cleaning algorithms were ready).
- **Number of visits**—only those with at least 5 or 10 visits were included (typical to online sites, the majority of IA registrants try it out for a visit or two, and then never return), and those with 90 or fewer visits as there were three users with well over 100 visits with the maximum at 211 who were dropped as outliers.
- **Project creation**—only users with three or more projects were included (just as the number of visits indicates usage and typical to online sites, the number of users producing content has been much smaller than those who merely consume). (Note: users in workshops have typically created at least two
 projects, so three was a number indicating that they continued use past the workshop and has been used in other IA studies as an inclusion criterion.)

Data Analysis Methods and Tools

Commonly, user-aggregated data would be subjected to a traditional $k$-means cluster analysis (e.g., Luan, 2004) coupled with a subsequent logistical regression to characterize the groups (Magidson & Vermunt, 2002). However, recent advances in statistics and data mining make other analyses attractive options.

Analysis and tool selection. The following six different criteria were considered in selecting the analysis for this study.

- Unsupervised methods are preferred in order to generate user groups from the data since the IA user classes were unknown.
- Ability to adjust the models based on the specific distribution shapes of the input variables because the data consists mainly of count types, not of the type often assumed by traditional methods (e.g., normally distributed continuous).
- Transparent and not of a black-box (or closed) internal structure so that a second analysis is not required in order to explore the results.
- Probabilistic (as opposed to deterministic) outcomes that more realistically model group composition by allowing partial membership in clusters.
- Stable models that are repeatable, not dependent on such techniques as rotation, and insensitivity to outliers, are preferred above those that are not.
Empirical comparison between different clustering is desirable in order to assist and justify model selection.

Like other commonly employed analysis techniques, \( k \)-means cluster analysis does not meet the analysis criteria—neither does factor analysis, nor principal component analyses (for a description of these techniques and their limitations, see Apley, 2003; Magidson & Vermunt, 2002, 2004; Stevens, 2002).

The selected analyses for KDD was LCA, which was introduced in Chapter II. Even with inherent shortcomings of LCA, it fits the above criterion more closely than any other analysis known to the researcher. Some limitations already known with LCA are that: (a) While LCA has extensions which allow for many shapes assumed in the data, the true data may still not match those assumptions very well—this limitation is somewhat overcome by the ability to decompose odd observation data into mixtures of multiple distributions; (b) LCA has quite stable models, however, local minimums may be mistakenly found—and may be overcome by using random starts over multiple iterations; and finally (c) the empirical comparisons may not always yield the most understandable model, and therefore a combination of analyst knowledge of the data and interpretation of results of statistical comparison are considered best practice.

Since the data is in a PostgreSQL database, it was desirable to have the ability to pull the data directly into the statistical environment for exploration. Additionally, with the purpose of reporting on tools for digital libraries, and with ubiquitous budget constraints, a set of tools that are inexpensive or free for use were desirable. For these reasons the R statistical package (http://www.r-project.org) and LatentGold
(http://www.statisticalinnovations.com/, the only software purchased for this study) were used. Notepad++ and VIM were utilized as the R-language editors and NppToR was used to send the commands to R. Finally, many of the plots and tables were constructed using the TikZ library in R so this document could be produced in LATEX.

Other packages that offer LCA are SPSS with Amos or Clementine (http://SPSS.com) and SAS (http://SAS.com) by way of E-M analysis. Both of which are both popular choices for both statistical analysis and data mining. Another tool recommended for many different kinds of analysis (including LCA) is mPlus (http://www.statmodel.com/). There are two packages in R that do LCA: poLCA and MCLUST. Both are generally considered fine for functionally, if not a little more difficult to learn and use but were too limited for this study (Haughton, Legrand, & Woolford, 2009).

**Latent cluster analysis methods.** The logical progress of analyses with so much new and unexplored data was to first look at those who were actual users of the IA with a few of the less complex features allowing for more users to be included in the analysis as well as providing a first look at latent clusters and their features. Future analyses will then have a baseline upon which to operate and delve into the more esoteric features (including longitudinal analyses).

While there is a little overlap here with the next portion of the KDD framework, namely evaluating and presenting the results, the part of evaluation relating to the quality of analysis and results will be described here with evaluation of the interpretation and final determination of the utility of the results for the next section.
The methodology of this analysis includes the following iterative steps.

- Choose initial features and their data types.
- Estimate the expected number of classes.
- Iterative selection or removal of classes, features, covariates, and appropriate interactions based upon model impact.
- Evaluate the results for a balance of statistically optimal model with comprehensibility of the model.

It was advantageous to select the features and range of the number of classes reasonably needed by way of expert knowledge to describe the data. Then, inputs that statistically contribute to the analysis are kept, and finally testing the model to see if the classes have meaning in the context of the IA meaningful use framework. In other words, one must balance what the statistics indicate is the best fit and the ability to comprehend and make sense in terms of the study context.

Each of the analysis steps will now be described in more detail.

**Choosing initial features and their types.** In selecting the features for initial inclusion into LCA, the larger elements in Figure 7 were considered first. This excluded fewer users because of a missing data. Additionally, this approach allowed for a simpler, and likely, more understandable primary model.

The initially included features are listed in Table 4 in Chapter IV. Again, because of recent extensions to LCA, each feature can have a different data type which then can be decomposed into mixture models. In order for LatentGold 4.5 to work well with the data,
each variable must be given a data type (i.e., ordinal, nominal, numeric, count or Poisson, or bivariate count). Each type is also included in Table 4.

**Expected number of classes.** The number of classes expected from LCA for this study ranged from a single, monolithic class \((k = 1)\) to a class for every user \((k = N)\). For understandability, a parsimonious model was desired, so while the minimum number of classes asked for was 1, the maximum number tested was limited to 10—any more would be a complex model, indeed. It was expected that the number of classes would fall between four and seven in light of the work by Xu (2011).

It was probable that the fit of the model would increase as more classes are added to the model, which is a normal occurrence with cluster analysis. While more complex models may fit better statistically, the utility of the model commonly wanes quickly as the class sizes approach one. Preliminary analyses of IA data (prior to the final participant and feature selection described above) showed that even with \(k = 20\) the fit statistics (described below) continued to indicate better models but the models were far too complex. This indicated that while a great many latent classes could be found, many of them differed only slightly across all features from other class definitions and they were quite small in proportion to the number of cases in the data. This unexpected result motivated the removal of much of the data through a tightened final user selection.

**Inclusion and removal of model inputs.** Just as in other latent analysis methods (e.g., factor analysis) the standard procedure with LCA is to combine inputs and then remove those that have no statistical significance according to the impact they have in the analysis. These inputs can include features (or outcome variables), covariates (or
preexisting classifications that may be used to further explore and understand latent classes), and allow dependencies between variables within each latent class.

Features. Iterative inclusion or removal of features (or outcome variables) based upon statistically evaluated model impact. After the each analysis was run on the data, each feature was assigned an $R^2$ value indicating the level of its effect on the model fit. It is common practice to remove features with $R^2 < 0.1$ (indicating that less than 10% of the feature’s variance is accounted for in the model; Vermunt & Magidson, 2005).

However, the higher the number of classes, the more opportunity there is for the inputs to significantly contribute to the model—since more information will be utilized to differentiate additional (and somewhat smaller) classes—but the model quickly becomes very complex and more difficult to interpret. After removing ineffective variables, the model was again computed and the process repeated until a stable set of features was obtained.

Covariates. Iterative inclusion or removal of covariates based upon statistically evaluated model impact much like the features. Covariates may also be left in the model in an inactive position so one can observe what does happen to them, but they do not impact the outcome of the analysis.

Interactions. In the original LCA methodology, local independence within classes was among the assumptions, meaning that all variance was accounted for in the class membership. However, with recent extensions, off-diagonal entries in the variance/covariance matrix could be allowed, thus relaxing that assumption.
Examination of the bivariate residuals between two features (calculated as chi-square divided by the degrees of freedom) in a model made it possible to determine how similarly they worked in the model. If there was a strong connection (when the residual was greater than the reference value of 1), it was flagged in LatentGold as a place to relax the local independence assumption by forcing a zero or near-zero entry into the variance/covariance matrix (Vermunt & Magidson, 2005, pp. 75–76).

**Balancing the statistical model with the comprehensible model.** After a series of LCA models were calculated on the feature set, a Bayesian information criterion (or BIC, see Ip et al., 2003; Muthén, 2004) for each model was used as a statistical basis for comparing relative model fit. A general pattern observed with LCA analyses indicating an optimal \( k \) is that the BIC decreases as \( k \) increases to a point and then the BIC begins to rise again as \( k \) increases indicating a likely optimal model at the lowest BIC (or best fit).

At times, the model with the lowest BIC is not the optimal model and a different model is selected for its comprehensibility (Vermunt & Magidson, 2003, 2005). A conditional bootstrap analysis was also performed to verify that two models are significantly different, assuming that the more restrictive model is a true model (Vermunt & Magidson, 2005, pp. 98–99).

Because of the opportunity for a local minimum model to be found (Vermunt & Magidson, 2003), it was important to run each model several times to see if the BIC is stable over many random starting points. If the BIC was not stable, then features, covariates, and interactions needed examination for adjustment as described above.
Once the right balance had been struck between statistically optimal and comprehensible models, the analysis was complete and it was time for interpretation, sense making, and impact planning on the resulting clusters.

**Evaluation and presentation**

Once a stable model was selected, the clusters needed to be examined and the results of this study presented in an understandable and intuitive manner. Multiple modes of communication were utilized to clearly communicate the LCA results, including: visualizations, descriptions, transformation of the results, and presentation techniques (e.g., tables, graphs, multiple views of line and box plots, color-coding).

Cluster sizes, cluster feature means, and distributions (depicted in box plots) were the primary information reported.

**Characterization Methods**

The KDD process with LCA as the analysis provided results for evaluation and presentation, but not for the interpretation of these results (revisiting the presentation). Therefore it was necessary that the latent classes (or clusters) are described and understood in their original context: that of IA users and meaningful IA activity.

The output of the KDD LCA analysis consisted of (among other information) the size of each latent cluster, the means for each class for each feature, and a classification table (raw inputs with class information). Combining these raw outputs with transformed versions of them and visualization in graphical format, clusters were compared and contrasted in a more technical manner—all this was done as part of KDD.
To fill the purpose of this study, the results were also analyzed in the context of meaningful IA activity in order to provide names and descriptions of class characteristics in a more intuitive and helpful manner. The clusters were named in plain terms according to behavioral patterns, the classes were described from the perspective provided by expert knowledge of IA users. These expert descriptions have been confirmed against the expert knowledge of other IA team members for accuracy, meaning, and clarity.

**Methods for Reporting Data, Methods, and Tools**

Another purpose of this research project was to report on how data, methods, and tools were utilized for this study. Qualitative methods were used for the analysis of the reflective journal (Glesne, 2006). The journal was scoured for information relating to data sources and their utility to this study (in terms of value added) of the data sources. Journal information regarding methodological and technical decisions was also gleaned. These findings were described and generalized to the larger idea of WUM and EDM when focused on user behavior as implications for future research efforts.

**Methods for KDD Documentation and Generalization**

The KDD process documentation occurred throughout the study. Activity journals were analyzed for the sequence and kinds of decisions that must be made (e.g., how to clean web server log data, what kinds of dirty data can be expected in web-usage data) and the outcomes of these decisions. Categorization into general procedures and considerations of these decisions and activities was the bulk of this analysis. The presentation of process documentation and generalization to the larger world of
educational digital libraries was in autoethnographic format (Glesne, 2006) focusing not only on the processes but reactions of the researcher as well.

This purpose was broader than the tool report section above, as this will generalize to the process of KDD in an educational library setting and not any particular data source, method, or tool.
CHAPTER IV
RESULTS

This chapter describes the implementation and results of the research plans outlined out in Chapter III. The order of presentation follows the research questions in Table 3: (a) progression through the application of knowledge discovery from data/databases (KDD) in pattern mining; (b) a characterization of emergent user groups; (c) a reflexive look at data, methods, and tools used; and, finally, (d) a look at the general procedures that have emerged from this case for the general area of educational digital libraries.

Nowhere, in all the literature reviewed, did there appear a detailed description of the preprocessing steps of KDD. Therefore, this study was not only an application of educational data mining (EDM) following the KDD framework to the end-user activities of a digital library service, but also an opportunity to describe and lay out some of the difficulties that can be encountered while engaging in this kind of research. Indeed, this chapter contains a very thorough description of the actual processes applied, speed-bumps encountered, decisions made, and results obtained from the methods described in Chapter III.

Pattern Mining (KDD) Results

As mentioned in Chapter II, and again in Chapter III, KDD was the central framework for the pattern mining portion of the study. The first step after data collection is preprocessing—consisting of cleaning, integrating, selecting and integrating the data
(early preprocessing results were also reported in Palmer & Recker, 2009). Next, the pivotal step of data analysis was performed—in this study, latent class analysis (LCA) was used—and finally evaluation and presentation of the results for the final step.

There was much to accomplish in pursuit of the first research question: “What patterns of user behavior emerge in terms of key features and metrics?” This section describes the results of this KDD application.

**Data Collection and Preprocessing Results**

The end goal for the preprocessing portion of methods was the creation of a user behavior *data mart* that would be suitable for many different kinds of analyses, and not just for the analysis of this study. Han and Kamber (2006) and other authors have warned that preprocessing is the most time-consuming part of data mining. Notwithstanding, little was mentioned about the actual process and experience of preprocessing. This section fills that gap by providing the detail that identifies specifics of the difficulties encountered. In every way, this was the most time-consuming of all tasks for this study, taking hundreds of hours alone.

The data mart was not an explicit purpose of this study, but as the data processing progressed, it became obvious that without too much more effort, cleaning all the data would have a desirable outcome and an advantage for continued studies of IA user behavior, therefore pains were taken to make the data as complete and of general form as possible.
**Data collection and integration.** As described in Chapter III, three data collection schemes had been implemented and were available for use, but the actual study was done solely on the Instructional Architect Relational Database (IARD). This was done because while the web server logs were nearly identical to the IARD for clickstream data, the IARD had the user identifiers associated with the records. Likewise with Google Analytics, while there were some user-level identifiers, the data were much more accessible in the IARD. For a user study it is important to have user-centric data and as a result, only the IARD was employed.

**IARD collection.** Data were gathered in the IARD starting in 2002 with simple application-specific dates. Later, additional data were collected with major enhancements in the detail of collection in January of 2008. The cleaned and integrated data included all data collected from the beginning until November 7, 2009.

**IARD internal integration.** Because the data were all collected in one database (i.e., the IARD), integration with other data sources was not needed. However, the data were distributed across several relational tables: users, projects, project_history, resources, folders, and user_tracking. Utilizing PHP (http://PHP.net) scripts initially and ultimately a combination of PHP and PL/pgSQL (see Chapter 38 of PostgreSQL Global Development Group, The, 2008) the researcher created a series of PL/pgSQL functions and SQL queries that performed the integration across tables into a format where cleaning, selection, and transformation (especially the creation of new variables) were able to be performed.
Only PHP scripts were originally slated for the preprocessing. However, as the steps progressed and the number of rows to process increased, it became evident that the relational database system was much more efficient at performing calculations and much of the string processing than transferring the data from PostgreSQL to PHP for processing and back into the database.

Fortunately most major relational database systems have some kind of scripting available for producing stored procedures or functions that operate on specified data. With PostgreSQL, PL/pgSQL was the language of choice and was utilized for several tasks on the data. For example, linking each user request with the one before and the one after it required looping over all visit data looking up the previous and next link for that clickstream—a task impossible in simple SQL.

The integration scripts followed this sequence.

1. Copy the production database to a new database and ensure proper indexing.
2. Combine current project data publish information and student group id.
3. Create a project history table with columns for the results of text-processing and comparison.
4. Combine saved project information with the current project state into project history table.
5. Combine click-tracking data into sessions.
6. Link session information with each project history utilizing date information.

Integral to the integration across tables was a normalized database with referential integrity (abundant use of primary keys and foreign keys), which enabled linking the
tables in the original database. For example, the unique user identifier (UserId) key linked the user to projects, resource folders, and tracking tables.

The simplicity of the above list belies the difficulty in accomplishing some of those tasks. As the data were integrated, new cleaning challenges manifested and were dealt with before attempting the integration again.

Cleaning. Missing, erroneous, or otherwise noisy data interfere with the quality of the analyses; therefore, it was necessary to fix, or remove these data while preserving as much valid data as possible. In the IARD the following problems with the data were found and dealt with:

- corrupt data in the forms of lost registration dates and mismatched Daylight Saving Time (DST) adjustments,
- session id reuse, and
- noisy traffic (spam, IA team, web crawlers).

Of these issues, only lost registration dates and noisy traffic were known beforehand. Problems with time and session id reuse were discovered while attempting to transform the data and were the most difficult and time-consuming part of cleaning. These unanticipated issues underscore the iterative nature of this work. The details and cleaning results of each problem set are now treated.

Corrupt data. Two kinds of corrupt data were found: (a) dates lost during system migration and (b) mismatched date information.

Lost or missing data. When computing how long each user had been registered, it was necessary to have valid registration data. However, in 2004 the IARD was migrated
from one database structure to another and in the process all existent (N=364) user registration dates were reset to ’1990-01-01 00:00:00’.

Up to that point in the IA’s history users were nearly all participants in professional development workshops. Therefore, lost registration data could be filled by taking the first valid time stamp known to the system and subtracting five minutes—roughly equivalent to the time required for workshop participants to register and begin collecting resources or creating projects.

The value of this registration date was only in creating the data mart since in the end these early registrants were dropped from the study. Furthermore, only eight of these early users had logged in since January 2007 and none had visited in the last year of data collection.

Other missing data were the effect of progressive improvements in data collection (i.e., data collection start times were not all the same). This had no remedy, but was helpful in understanding the constraints on the depth of the data and contributed to the final user inclusion criteria described in a later section.

Mismatched dates. Beginning in 2007, Daylight Saving Time (DST) in the United States was advanced from beginning on the first Sunday of April to the second Sunday of March. The end of DST was delayed from the last Sunday of October to the first Sunday of November. This was the first major change in 20 years (WebExhibits, 2008) and provided approximately one extra month of DST.

Unfortunately software is unaware of congressional acts unless patched. Some software for the IA was updated after the legislation went into effect and others were not
due to requirements for a collaborative project with the NSDL. The result was that during
two times of the year, any date generated in the unpatched PHP and saved into the
database was off by one hour from any dates automatically generated by the updated
PostgreSQL. These dates required correcting so that session and project edit times could
be aligned. If the time was off by an hour then the session information would likely not
match and the data could not be integrated.

The solution consisted of finding all dates generated by PHP which were then sent
to the IARD and then adjusting them one hour ahead or back as appropriate for the
affected dates. This corrupted data only affected the times associated with saving and
editing projects. To avoid this problem, all dates should be generated in the database and it
may be helpful to store time zone and DST information separately.

**Session ID reuse and disambiguation.** Several kinds of PHP session reuse
occurred: multiple users visiting from the same computer in close temporal proximity,
limitations in PHP’s session-handling process, and a change in the purpose of the user. In
the last case, it is helpful to think of a session as a set of activities that have a related
purpose and is more inferential than the others.

**Multiple users.** In two situations, session identifiers alone could not be counted
on to identify only one user’s activities: when (a) a person beginning a session as
anonymous or logging in with the guest account and then subsequently registering and the
need to unite these sessions, and (b) cases where multiple users logged into one computer
without the PHP identifier changing.
For the former, if there was at most one valid (i.e., nonguest and nonmissing), then the valid identifier was applied to the entire session. In the latter case, the number was very small (less than 380) and they were dropped.

Limitations in PHP. The IA was written in PHP where randomly generated session identifiers were assigned to each requesting agent (i.e., browser) and kept track of in a client-side cookie in order to save and protect the user information associated with that session.

Because of the random session id generation algorithm, duplicate session ids are eventually issued. This is generally not a problem if the id is large enough to ensure no two clients obtain the same id simultaneously (i.e., session id collision, see nielsene, 2005). Id collision would result in real-time corrupted session information; and from that, corrupted database information. Fortunately it is unlikely this collision has occurred in the IA because of the low number of simultaneous users.

While attempting to reconstruct session information even noncolliding, duplicate session ids were necessarily made unique—otherwise multiple users’ information would appear to be a single clickstream. Disambiguation of user sessions was based upon the time between requests. As a result, an appropriate gap between session information was needed. While determining the time between requests another phenomenon was discovered which will now be described before the solution to both problems is presented.

User distraction or inactivity. As session id reuse was studied to determine the appropriate length of inactive time that would be considered a different session it became
apparent that some nonreused session ids indicated unusually long durations some being as long as days.

In commerce site data mining, the traditional session-idle timeout has generally been considered to be around 30 minutes—meaning that after 30 minutes of inactivity, subsequent requests are assumed to have a new purpose or direction. These meaningful segments of activity have been called episodes (Cooley et al., 1999; Srivastava et al., 2000). The application of this concept meant that any large gap in user activity was likely the beginning of a new set of activities and so even long sessions that had a relatively large delay in activity should also divide unique sessions.

Given the reused session identifiers and differences between commerce and education domain—and even more, with the gaps caused by resource visits by linking from IA projects—the maximum allowed IA episode idle time had to emerge from the data.

The following steps were applied in order to discover the appropriate length of session timeout to use:

1. An initial sessionLength was selected.
2. All sessions that were longer than sessionLength were selected as part of longSessions.
3. The average time between clicks of longSessions was calculated as avgLongSessionDelay.
4. Queries were executed that split longSessions at sessionLength into candidateSessions.
5. The average time between clicks of candidateSessions was calculated as 
   avgCandidateSessionDelay.

6. avgLongSessionDelay was Compared to avgCandidateSessionDelay to see if 
   the average time between candidateSession clicks were relatively shorter than 
   when grouped by the original sessionLength.

7. If avgCandidateSessionDelay was relatively shorter and an analysis of 
   borderline activity indicated a continuing episode (i.e., by editing the same 
   project, searching for similar concepts, etc.), then sessionLength was increased 
   and steps 2–6 were repeated.

8. Otherwise, an appropriate sessionLength has emerged.

The application of this algorithm was initialized with a sessionLength of 30 minutes and 
ended with 60 minutes.

Other tables (e.g., project_history) had to be updated after manipulating the 
session identifiers to reflect the new session identifier, start time, and stop time. Of all the 
cleaning activities, this was by far the most time consuming and difficult.

**Noisy traffic.** In studying phenomena associated with specific traffic sources, 
today’s noise is tomorrow’s data. Therefore, different traffic sources were identified 
(where possible) and flagged in such a way that they could be selected or excluded for 
analysis. Three traffic sources were considered noise with respect to the study of real-user 
behavior: IA team visits, spam, and web crawler activity.

**IA team visits.** As the IA was developed, the IA developers and other team 
members visited projects for workshop presentations, research purposes, and system
testing. These enumerated users were removed as outliers due to their inordinately high number of logins, projects, resources, and visit frequencies.

*Spam.* As the IA is a fairly open authoring system (e.g., allowing HTML tags to be inserted into projects) the occasional spammer has registered and created projects which either forwarded to a new site or contained noneducational content (e.g., advertisements or other generally offensive material) that was incongruent for a PreK-16 educational site.

Upon discovery, each spamming account was disabled. On one occasion a spammer created a script whereby more than 400 user accounts were created with published projects before being stopped. The spam-discouragement solution chosen was to implement reCAPTCHA (http://recaptcha.net/) on the registration, project email, and comment pages. All 498 spam-related accounts were also flagged as invalid so they could be quickly added or removed from analyses.

*Web crawler activity.* Finally, as spiders, crawlers, and other search bots (both with good- and mal-intent) cover the Internet, they sometimes attempt to hide their identity. Some were filtered in the data collection process by finding associated strings in the user-agent data using PHP’s PERL-compatible regular expression matching (e.g., `preg_match('/spider|crawl|nutch|bot|Yahoo!|Slurp/i', $_SERVER[’HTTP_USER_AGENT’]))), but others perhaps exist and may only be found as outliers during the analyses. All web crawler activity is still contained in the web server logs.

**Preprocessing transformation and selection results.** Once the data had been collected, integrated, and cleaned, further work could proceed on transformation of the
data that would be studied and subsequently selected for analysis. Preprocessing transformation and selection activities did not happen in a silo or at one time. Nevertheless, they are reported here in a group and brief statements will help in placing them in correct temporal order.

Most of the transformation activity was in conjunction with selection of what specific features to extract (or transform through feature construction, smoothing, and generalizing) from clickstream or historical data that would indicate an enactment or manifestation of meaningful user activity as defined for this study, developing algorithms to accomplish the task, and then doing the actual transformations. There were three main areas of transformation: (a) registration and visit data, (b) user activity data, (c) project history and resource collection data. Each area is now covered, followed by a discussion of database management techniques that significantly reduce processing time.

**Registration and visit transformations.** User profile data (e.g., grades and subjects taught) were available in the IARD user table and required only the following transformations.

The source of their IA introduction (or IA information source) was smoothed assigning user-entered data into one of two nominal categories: formal or casual. Formal introduction into the IA was any users who marked workshop or online, or marked the *other* category with a known classroom feature (e.g., “class USU”) and would have been exposed to multiple instructional episodes with assignments and follow-up contact. Casual introductions were from word of mouth, conferences, and the balance of the *other* category where one-time introductions with no formal assignments or follow-up contact.
This binary informational variable (IAInfoBin) was the only one that could have been considered a preexisting class and used as a covariate to the clustering analysis.

The number of times they had logged in (NumVisits), dates for registration (RegDate), last login (LastLoginDate), and the number of dates activity occurred (discoverable after the activity transformation below) were utilized to produce such variables as the number of IA registrants who registered the same day (RegDayCnt), the number of days from registration to the end of data collection in November 2009 (DaysReg2Data), the percentage of days since registration that they visited (ActivePct), and the number of days visited (NumDaysVisited. It was assumed that the number of registrants would be related to the IA introduction variable above, and perhaps serve as an predictive measure of later activity.

On the other hand, the visit click-level data were encoded in tracking tables in the form of URLs, sequences of URLs, and comments and required a great deal of processing to be usable. Additionally, project history information also required some effort to generate associated measures. See Table B2 in Appendix B for additional features and descriptions related to registration and visit activities.

**User activity transformations.** These transformations were the first accomplished and provided experience for making the project transformation a faster process. This activity can be characterized as generalization and feature construction from the variables listed in Table B1 in Appendix B on a per-click basis. Counts, averages, and other calculations were performed to accomplish the goal of a session-level aggregation. The resulting variables are the subject of Tables B2 through B5 in Appendix B.
Note that in the three feature space tables (Tables B3–B4) the notation including square brackets ([and]) and pipes (|) is a shorthand notation for several variables that are collapsed into one table entry. For example, in Table B2 under Resource Gathering, # [Total|NSDL|IA|Web] resources saved, there were actually four variables created; one each for Total, NSDL, IA, and Web resources saved during that session.

The following steps were utilized in applying the methods described in Chapter III to produce the various levels of data (from single click to user aggregation). The technologies used are in parentheses.

1. Created a brand-smoothing variable in the resources table—collapsing all specific digital library brands into one of three values indicating their source as NSDL, IA, or Web (SQL).

2. Created and filled meaningful activity table through generalization and feature construction—at most one activity per user-tracking entry—the result was a very long and wide table with a 1 if the activity occurred and a 0 if not (SQL).

3. Filled the rest of the activity table that required looping, clickstream linking lookup (i.e., finding and linking the last, current, and next click information), and other, more complex operations—it was this phase in which the need for session id cleaning was discovered and solved (PL/pgSQL).

4. Aggregate session-level activity information and generate session statistics (SQL).

The first three transformation steps involved feature construction from page-level tracking table data into 110 different activity variables based upon:
• URL script name,
• HTTP GET variables,
• script name,
• clickstream parent and child, and
• the presence/absence of the user identifier (UserId) and student login setup (GroupId).

Built-in functionality in PostgreSQL made most of the above relatively simple implementations of regular expression pattern matching and case statements (PostgreSQL Global Development Group, The, n.d.-a). It was during this process that the need for session id cleaning was discovered, as well as the importance of indexing tables (addressed below); as such, this was the most time consuming process of all totaling hundreds of hours.

Table B2 in Appendix B demonstrates the number of variables created during this process (and some of the project variables as well) for session-level information. The session was considered the first level of aggregation since it was assumed that there was a specific purpose behind the visit, and the patterns of purposes performed by the user was the original object of inquiry for the researcher.

**Project history and resource collection transformations.** While resource collection counts were a simple matter of SQL that counted up the number of resources, their sources, and calculating percentages for each source, the project history was much more involved. However, because of lessons learned while transforming activity data the time required to accomplish this process was significantly reduced.
The following five transformations produced Table B5 in Appendix B (examined in more detail below). The first two deal with each particular change in the project history and the balance deal with the goal of a session-level aggregation. The technologies used are in parentheses, and more detail for each step is given after the list.

1. Compare project histories to construct distance data between consecutive project saves. (PL/pgSQL).

2. For each project save, calculate word counts, HTML use, and text differences between project saves (PHP).

3. For each session, calculate comparative date, resource usage, and other variables between the initial and final state of in the project history (SQL).

4. For the session, calculate word counts, HTML use, and text differences between initial and final project state (PHP).

5. Combine project session information with the session activity information table (SQL).

**Step 1.** This was accomplished through feature construction that was similar to processing clickstream data described in the last section—how long since the last, how soon to the next, etc.

**Steps 2 and 4.** For comparison of textual content from one version of a project to another, PHP soundex (PHP Documentation Team, The, n.d.-d) and similar_text (PHP Documentation Team, The, n.d.-c) string functions were used. Together these provided a phonetic representation of each word (looping over each word and using soundex to build
a new string) and how different each version of the project was from the last/next version (running similar_text over the entire new string).

By using soundex, spelling changes were considered a trivial or nonexistent and would result in a higher similarity index. Other PHP string-comparison options available were metaphone (PHP Documentation Team, The, n.d.-b) and Levenshtein (PHP Documentation Team, The, n.d.-a) distance. In practice, however, the former did not provide the granularity desired and the later was too resource-intensive to be of use.

Text characteristics (such as “was HTML used?”, word counts, and resource counts) and text changes were captured in 39 variables. Combining these for each project and session gave 108 different variables.

**Steps 3 and 5.** Another kind of transformation required was generation of new session-level variables from existing data (steps 1 & 2). Some examples were mentioned above, such as the following user metrics from Table B2 in Appendix B: duration (in time) and depth of visit; number of projects created or modified this session; number of words added, removed, and net change; number of resources added to and removed from projects this session.

**Data transformation for analysis.** At this point the decision was made to change the analysis from a session-based longitudinal cluster analysis to that of a user-based clustering due to the lack of data (see section Changes to the Proposed Study in Chapter III). Therefore, additional transformation was necessary to aggregate information to the user level for analysis.
The multitude of features produced during preprocessing in Tables B2, B3, and B4 in Appendix B were once again transformed into a user-aggregate form by being counted, summed, converted into ratios, and/or averaged (in addition to some standard deviations calculated). The final count of user-level aggregated features was 219 and they are listed in Tables B2 through B5 of Appendix B by meaningful activity.

Aggregating session information was fairly quickly done since the now-unique session identifiers allowed for grouping and counting with SQL.

Attributing session information to a user, on the other hand, was a bit more difficult because (as described in Cleaning, above) each session could have more than one user. Sessions with two users and one of them being guest were attributed wholly to the other registered user.

There were approximately 380 mixed-user sessions remaining where not only could two registered users be represented, but different student logins could also exist in each. In the visit_user table, the unique user identifier (UserId) for each session user was recorded along with information about the percentage of pages were attributed to that user for that visit to see if there was a way of dividing up the sessions. In the end, however, the mixed-user sessions were dropped from analysis.

**General database management concerns.** At each step, applicable indexes were created to expedite database lookup and updating. This was a lesson learned the hard way more than once as algorithms were the focus of the development and not initially speed. With tables of 1.3 million rows, it was essential to have indexes to reduce the time to find a particular row for reading, updating, or calculation. For example, when initially
working on Step 3 in User Activity Transformation above, there were approximately 50,000 rows for one month’s usage and processing was taking about 750 milliseconds for each row—a total of 37,000 seconds or 10 hours and 25 minutes for only one month’s data. Once an index was created on the lookup key, the same operation—in entirety—took only 68 seconds to complete! After a second experience like this, great pains were taken to ensure proper indexing before an operation.

Another essential practice with PostgreSQL is that of vacuuming (VACUUM) and analyzing (ANALYZE) the table (PostgreSQL Global Development Group, The, n.d.-b). Because PostgreSQL supports transactions—including rollback—it is built to not overwrite a modified row (updated or deleted) and will simply mark the outdated row as dirty and add a new row at the end of the table and new entry in the index. Thus, if every row is modified, the size of the table in memory and on the hard drive will double. The indexes also retain the dirty row information, slowing them down as well.

The VACUUM command marks dirty rows for reuse. A deeper version of VACCUM includes the FULL modifier. Once FULL is invoked, VACUUM locks the entire table and writes the information into a clean location and deletes the old location—eliminating dirty rows and shrinking the size of the database. The ANALYZE command will recalculate the table statistics so that SQL queries can be planned with much more accuracy—as the statistics change with many inserts and modifications. An additional REINDEX command also assists the query planner by rebuilding the indexes, but without doing a VACUUM FULL the dirty rows remain.
At this point the data mart was ready for almost any kind of analysis of IA user meaningful activities.

**Final Data Selection and Initial Analysis Results**

Initially, simply running an LCA on the whole mass of 200 features with all users crashed the system. Once several features were removed, it performed better, but there were so few users (e.g., $n = 30$) that could be included because of the lack of data that the results were highly unstable and unusable. It was evident that additional paring down of the features was necessary.

In selecting what features to keep, Tukey (1962) stated, “We consider it appropriate to combine subject-matter wisdom with statistical knowledge in planning what factors shall enter a complex experiment, at how many versions each shall appear, and which theses versions shall be” (1962, p. 46).

Accordingly, features created in preprocessing were placed under scrutiny for probable impact on the analysis based heavily on their meaning (what meaningful IA activity they addressed), with some concern for density (avoiding missing information or too many 0 entries), and to a lesser degree the shape and distribution of the data. Specific feature and participant selection for analysis was cyclic with and is described below with the results of preliminary analyses.

Selecting which data to keep or discard, as well as what to include in each phase of analysis was a balance between selecting the features that could inform meaningful usage patterns and retain as many users as possible for a more comprehensive model. The more
granular and specific the variable, the fewer users had data for that variable. Fortunately, most users performed at least one action in each meaningful activity category so the density was less of a concern than meaning.

Initial feature selection. Just as the changes to planned analysis were to look first at a simpler model, the initial data selection criteria was moderately restrictive, concentrating on only the highest-level information for the meaningful IA activities. Later, when the data are fuller or wherever interesting characteristics appeared, further analyses can be performed to shed light on the subtleties of the more granular data in future studies.

By highest-level information it is meant that when there were subdivisions available, only the top level was selected. For example, with the percent of projects with resources (PctPrjWithRes) there are also three sub-percentages calculated for the percent of projects with: (a) NSDL (PctPrjWithNRes), (b) IA (PctPrjWithIRes), and (c) Web (PctPrjWithWRes) resources, respectively. Likewise, the lower level of actually employing a resource from any given source were dropped in favor of the overall percentage of resources used (PctResUsed).

Table 4 shows the subset of all generated and aggregated variables (see Appendix B for more details) that were originally analyzed to see the layout of emergent user classes. Of the 214 features from the user aggregate data that did not crash LatentGold, only 44 were selected and initially fed into the LatentGold analysis software. Table 4 describes the initial feature space in terms of their category of meaningful IA activity, their data type and whether or not they were used in the final model.
### Table 4

**Initial LCA Feature Space**

<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
<th>Description</th>
<th>Data Type</th>
<th>Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration and Usage</td>
<td>IAInfoGrp</td>
<td>User profile input indicating how they were introduced to the IA.</td>
<td>Nominal</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>IAInfoBin</td>
<td>IAInfoGrp binned into Formal (workshop, class) or Casual (conference, word of mouth)</td>
<td>Nominal</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>RegDayCnt</td>
<td># users registered the same day</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>DaysReg2Data</td>
<td># days from registration to data capture</td>
<td>Count</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>ActivePct</td>
<td>DaysReg2Data/DaysReg2LastVisit</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumVisits</td>
<td># logins</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgDaysLastVis</td>
<td>Average # days since prior visit</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>AvgSessLenMin</td>
<td>Average session length in minutes</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>AvgSessDepth</td>
<td>Average # pages requested per visit</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumUserSess</td>
<td># unique PHP session identifiers</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>UserIPCnt</td>
<td># unique IP addresses used by the User</td>
<td>Count</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumDaysVisited</td>
<td># unique days logged in</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>NumUserResView</td>
<td># resources viewed by user</td>
<td>Count</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumUserPrjView</td>
<td># IA projects viewed by user</td>
<td>Count</td>
<td>No</td>
</tr>
<tr>
<td>Resource Collection</td>
<td>NumResources</td>
<td># of user-gathered resources</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>PctResUsed</td>
<td># unique resources used / NumResources</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumFolders</td>
<td># folders used to organize user-gathered resources</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgResPerFolder</td>
<td>NumResources / NumFolders</td>
<td>Continuous</td>
<td>No</td>
</tr>
</tbody>
</table>

(table continues)
<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
<th>Description</th>
<th>Data Type</th>
<th>Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>NumPrjWords</td>
<td># non-HTML words in all projects</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td>Creation and Edit</td>
<td>AvgPrjWords</td>
<td>Average # non-HTML words per project</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>WordsPerRes</td>
<td>Ratio of # non-HTML words in all projects to the # resources used in all projects</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgWordsPerRes</td>
<td>AvgWordsPerRes / NumPrj</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgEditSessLen</td>
<td>Average duration of sessions in which at least one project was edited</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumPrjEditSess</td>
<td># sessions in which a project was edited</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgEditResDiff</td>
<td>Average net resource added/removed from projects in each session</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>AvgNumWordDiff</td>
<td>Average non-HTML word count difference between edit sessions</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgSoundexDiff</td>
<td>Average non-HML project content changes as described in the Project history transformations section above as a comparison between phonetic representations of pre/post edit session content using PHP’s soundex and similar_text functions</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgResPerPrj</td>
<td>Average resources per project</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>PctPrjWithRes</td>
<td># projects with resources / NumPrj</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumPrj</td>
<td># projects created by the user (excluding deleted projects)</td>
<td>Count</td>
<td>Yes</td>
</tr>
</tbody>
</table>

$table continues$
<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
<th>Description</th>
<th>Data Type</th>
<th>Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>NumStudentPrj</td>
<td># user’s projects published to students</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td>Usage</td>
<td>NumPublicPrj</td>
<td># user’s projects published to the public</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>NumPrivatePrj</td>
<td># of user’s projects only accessible by the user</td>
<td>Count</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>PctPublishedPrj</td>
<td>(NumPrj - NumPrivatePrj) / NumPrj</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>PctPrivatePrj</td>
<td>NumPrivatePrj / NumPrj</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>PctStudentPrj</td>
<td>NumStudentPrj / NumPrj</td>
<td>Continuous</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>PctPublicPrj</td>
<td>NumPublicPrj / NumPrj</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>StuLoginExists</td>
<td>1 if a student login has been created for this user, otherwise 0</td>
<td>Boolean</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumPrjVisits</td>
<td># visits to user’s projects</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessPrjView</td>
<td>Average project-views between user sessions</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessResView</td>
<td>Average resource-views between user sessions</td>
<td>Continuous</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>NumStuLogin</td>
<td># times user’s student login has authenticated</td>
<td>Count</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>NumStuPrjViews</td>
<td>Total # of student-published projects viewed by someone using the user’s student group credentials</td>
<td>Count</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>NumStuResViews</td>
<td>Total # of resource views by someone using the user’s student group credentials</td>
<td>Count</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Subsequent feature selection was driven by iterative application of LCA by scrutiny of the impact on the analysis results and is described below. Only 23 features remained in the final model.

**Initial participant selection.** Some participants were preselected for elimination. For example, user information belonging to team members and spammers were excluded from analysis. Outliers were also investigated and dropped or segmented for separate analyses due to their potential for biasing the analyses.

As with integration, cleaning, and transformation, the process of participant selection happened in cyclic episodes with preliminary analyses.

Of the 5,378 existing accounts when data were collected in November 2008, 498 were secured (or spam) accounts and approximately 50 were guest, test, or IA staff member accounts. Removing these users left approximately 4,830 actual users.

Data collection had progressed in sophistication from the humble beginnings in 2002 when only simple dates were kept (e.g., user registration date, RegDate). User login counts (LoginCount) were added in 2004, project history since mid 2007 (e.g., NumberNewResourcesUsed), project visit tracking since August 2006 (e.g., NumVisitsToProject), and detailed click tracking since January 2008 (e.g., NumResourcesVisitedFromProject). Because of the incremental nature of data collection it was impossible to make user aggregations for early-users that would be consistent with those for whom complete data were available.

The impact of dropping the early-users was explored and found that of those registering before January 2008, only 239 have logged in since January 2008 and just 90
visited between January and November 2009. These visits account for only 8% of the 1,123 users visiting in the same 11 months. Since they contributed such a small proportion of the visits, it was decided that in order to maintain greater consistency in the data, only those registered after the January 21, 2008 tracking enhancements were implemented would be included in the analyses.

This last restriction left 2,091 users (43.3% of all actual IA users) registered in the 22 months between January 2008 and November 2009. Those not used in this study can be examined through other studies for understanding the relatively small, but committed user base.

The histograms for the remaining user data are displayed in a panel format for visual analysis of distribution shape in Figure 9.

**Initial results.** After initial feature and participant selection activities, an LCA was performed on the data as described in the latent cluster analysis methods section of Chapter III for 1 to 20 clusters (k). As the number of clusters requested increased from 1 to 20, the BIC (see Chapter III, balancing the statistical model with the comprehensible model; see also Ip et al., 2003) continued to decrease with no known limit (perhaps until the worst-case k = n situation). While a lower BIC is desirable, the complexity of a model with 20 classes was likely beyond the reasonable expectation for this particular data set.

Indeed, with k = 20 the cluster feature means were all parallel (i.e., only varying by degree of activity, not kinds of activity) from 1 to about 10 clusters. For example, Cluster 1 had the lowest cluster mean on all features, Cluster 2 means were just higher than Cluster 1 on all feature means, Cluster 3 was just above Cluster 2 but the difference
Figure 9. Feature histograms for the presegmented data. Each shows the general shape of selected feature distributions with all users (0–211 visits). The vertical axes are counts, percentages, and averages as per the title of each feature.
was less; and so it continued with each successive cluster just above the last but the gap was ever decreasing as was the size of the clusters.

Above 10 clusters, the relatively small groups that had actually used various IA features in differentiable ways began to surface in a plot comparing normalized cluster means for each feature (see Figure 18, discussed and shown later in this chapter, and accompanying explanation for the normalization process). The new groups had relatively higher means in some areas and lower in others—thus finally differentiating user behavior that subject-matter knowledge and experience had observed and knew existed. However, the cluster sizes eventually reached less than one person for the later classes to be calculated. This result was unexpected and rather puzzling at first.

Further examination of the data revealed that users did not return very much—in fact, approximately 50% had never returned after registering (their zeroth visit). Indeed, this zero-or-one-visit group weighed so heavily in the probabilities that initial LCA analyses required more than 15 classes before any differentiating behavior (besides simple parallel levels of activity) were teased out of the mass. It was apparent that many so-called users were actually nonusers and needed to be segmented so that meaningful activity could be studied.

This result did expose a limitation to using LCA with an overwhelmingly large group that dominates the probabilities and increases model complexity with little information gained. Once it was understood in the context of the study—one of users and not nonusers—it was clear that the data required further segmentation than simply the January 2008 cutoff.
**Segmentation of data.** With the data first aggregated at the session level, it seemed appropriate to segment the users based on the number of visits or logins they had made to the IA (NumVisits). As mentioned in the last chapter, other IA research has used such criteria for *active* user status (e.g., at least three projects).

The number of visits (NumVisits) ranged from 0 to 211 for those meeting the initial criteria. Only three users had logins greater than 90, and they were well into the hundreds, therefore 90 visits appeared to be a reasonable upper limit to the accepted logins. The remaining data were broken into two overlapping groups based on their number of logins (or visits, NumVisits): 5–90 logins ($n=366$) and 10–90 logins ($n=197$, a proper subset of the 5–90 group).

**Histogram analysis.** Lattice histograms (aligned plots) for 20 selected features are shown in Figure 9 (users with 0–211 logins) that gives perspective to how the features of those with zero to four logins dominated the distributions when compared to the distributions in Figures 10 (users with 5–90 visits) and 11 (users with 10–90 visits).

Visual analysis of the figures reveals that diversity in the data becomes more distinguishable as the center of mass is no longer dominated by one group. Note how, in Figure 9 the number of user sessions (NumUserSess) are nearly undetectable past 30 sessions with one large group near zero; however, with the following two matrices a more Poisson-like distribution becomes evident.

While this visual analysis is dependent on the scale and binning of the histogram, it does provide a view into what the probabilistic LCA observed with a single dominating
Figure 10. Feature histograms for the 5–90 login sample. Each shows the general shape of selected feature distributions for the 5–90 login group. The vertical axes are counts, percents, and averages as per the title of each feature.
Figure 11. Feature histograms for the 10–90 login sample. Each shows the general shape of selected feature distributions for the 10–90 login group. The vertical axes are counts, percents, and averages as per the title of each feature.
group and underscores the necessity of segmentation. Segmentation of this user sample brought encouraging results.

**Box plot analysis.** Another useful visual analysis for understanding the data was by employing box plots (see the references section in R Foundation, The, n.d.). Because means are influenced by outliers, it was also important to examine median and inter-quartile range values for each feature.

While Figures 10 and 11 are histograms of the data fed into the LCA analysis, some of the data were unusable in the analyses and additional observations were dropped. Figure 12 contains box plots of the data actually used by the analyses and provides a different view of the analyzed data.

Box plots are useful graphical analysis tools and are featured here to emphasize some of the similarities and differences in the data. Using the R statistical package (R Development Core Team, The, 2011) lattice library to create the box plots with default settings used for boxplot.stats (the first reference is the Lattice package manual, the second is specifically the documentation for the box plot stats setup: Sarkar, 2008; R Foundation, The, n.d.), the box plots are interpreted as explained here.

- Median values are indicated by the horizontal line across the box.
- First quartile is represented by the lower boundary of the box (lower hinge).
- Third quartile is represented by the upper boundary of the box (upper hinge).
- The minimum first and maximum fourth quartile values that are not more than 1.5 times the length of the box away from the box are represented by the whisker length (marked by the umbrella, the horizontal line at the end of the whisker).
Figure 12. Box & whisker plots of all features for the 5–90 login and 10–90 login samples without cluster breakouts.
• Outliers appear as dots beyond the extent of the whiskers.

• Notches are the tapering of the box and correspond to the distance proportional to the inter-quartile range (IQR; $+/ - 1.58 \frac{IQR}{\sqrt{n}}$) or “roughly that of a 95% confidence interval for the difference in two medians” (R Foundation, The, n.d.)—giving a visual estimate of statistically significant median differences when one median is outside the notch of a parallel box plot.

A note about notches. When the notch extends past the hinge of the box, it will fully extend and then return back to the actual hinge position. This was observed in Figure 12 for the number and percentage of public projects of the 5–90 visit group (NumPublicProj and PctPublicPrj) feature where there was an upper limit of 1.0, and indeed the median was at that point, but the so-called upper confidence interval actually extended beyond the limit (a ceiling effect).

Comparing the sample distribution differences displayed in Figure 12 it was possible to realize that several differences existed between the two samples in the way of outliers, medians, and inter-quartile ranges (IQR).

It was not difficult to observe that several of the outliers are common between the two samples. Checking back to the data it was discovered that the outliers were not always the same users. Even though some users are sometimes an outlier on one or more features, they were not always the outlier on others and so their data were included in the study.

Some shifts in the median values are apparent between these figures. For example, the number of same-day registrants (RegDayCnt) dropped quite a bit when the five to nine login group is removed. The percentage of public projects (PctPublicPrj) also dropped,
indicating that in moving to the smaller sample a lot of publicly producing users were dropped from the study. Other differences between the two samples are notable where the inter-quartile range generally grows when those with five to nine logins are removed.

A last observation to note about these plots of input data: they nearly all have a skewed distribution—a justification for utilizing the Poisson (or count) distribution as the assumed underlying distributions mixed in the population.

**Final participant selection criteria.** As mentioned in Chapter III and to summarize what was described in detail above, the final user selection criteria are condensed here. These criteria are in alignment with participant selection methods from previous IA user studies (including self-selection into workshop participation and IA usage both as an authoring and instructional tool; e.g., Recker et al., 2006; Recker & Palmer, 2006; Walker et al., 2010; Xu, 2011):

- *Registered users*—only registered visitors were included.
- *Registration date*—only those users who registered on or after January 21, 2008 and before or on November 7, 2009 were included (RegDate, given the tracking improvement in January 2008, it follows to only include users for whom we have complete data, so for this study).
- *Number of visits*—only those with at least 5 or 10 visits and less than or equal to 90 visits were included (NumVisits, typical to online sites, the majority of IA registrants try it out for a visit or two, and then never return).
- *Project creation*—only users with three or more projects were included (NumPrj, just as the number of visits indicates usage and typical to online sites,
the number of users producing content has been much smaller than those who merely consume. *Note:* users in workshops have typically created at least two projects, so three was a number indicating that they continued use past the workshop.)

**Final LCA Analysis**

With the data created, integrated, transformed, selected, and explored with initial analyses, the next step was to tune the analysis and settle on a model that fits the data without being so specific that it cannot have some reliability when generalized.

After segmentation and culling the nonuser group, analyses proceeded first on the 5–90 login group, and then the 10–90 login group was analyzed. The LCA methodology described in the Inclusion and Removal of Model Inputs section of Chapter III were again followed for both samples adjusting the following settings.

- Clusters—the number of clusters to create (identified by $k$).
- Features—based on $R^2$ values.
- Covariates—based on model impact.
- Interactions—through relaxing the assumption of local independence.

**Number of clusters.** After looking at the results from one to seven clusters, it became apparent that a model with two clusters was too low (i.e., not descriptive enough), and that one with six had too many. The criteria included:

- BIC (desirably low);
• cluster sizes (moderate sizes—the first cluster calculated will be the largest, the last the smallest, and worst-case is $k = n$);

• the complexity of the model as evidenced by the number of parameters in the model, calculated as: $k \times (1 + N_{\text{Features}} + N_{\text{RelaxedInteractions}})$;

• and the normalized cluster feature means compared across clusters (as described above, variation is desirable for characterizing distinct behavioral actions).

Even though the conditional bootstrap analysis (Vermunt & Magidson, 2005, pp. 98–99) indicated that the models were all statistically significantly different ($p < 0.0001$) when moving from 3 to 4 and from 4 to 5 clusters, several model selection criteria began to change with $k > 4$. The difference in Bayesian information criterion (or BIC; Akaike information criterion, AIC; see Ip et al., 2003) from one $k$ to another continued to diminish indicating that each additional class contributed less to the knowledge of user behaviors. Figure 13 shows how the BIC changed with $k$ for both samples.

![Figure 13](image)

*Figure 13.* A plot of BIC statistics for $k = \{1 : 10\}$ for 23 features on two samples (users with 5–90 logins and 10–90 logins).
With more clusters the higher clusters’ sizes became impractically small (e.g., a cluster size of 1.6 people). The trade-off between comprehensible and practicality vs. the complexity of a better fit (lower BIC on many small clusters with an increasing number of parameters) reached a tipping point. At last, the normalized cluster plot showed that more clusters did, in fact, discriminate on additional features; however, balancing with cluster size, they were impractical. The opposite was true with \( k < 4 \).

Therefore \( k = 4 \) was settled on as an appropriate number of clusters for its balance of fit and to keep the models meaningful.

**Features.** A determination to drop or keep features was based on the \( R^2 \) value for each. Wherever \( R^2 < 0.1 \) was true, the feature was dropped (indicating that less than 10% of the feature’s variance is accounted for in the model, see Vermunt & Magidson, 2005). Only 23 of the initial 44 remained in the models as indicated in Table 4.

**Covariates.** The single preexisting feature that could be used as a covariate was the user-entered source of their IA exposure (IAInfoGrp) or its binned derivative (IAInfoBin). The binned version was added as both a feature and a covariate in many different scenarios. However, the effect was never significant or impacted the models in any detectible way. Therefore it was decided that planned covariates would be pulled from the analyses until such time as additional data are available.

**Interactions.** Relaxing the local independence constraint was a major advancement in LCA capabilities by allowing interactions between features within the clusters (i.e., including direct effects) as described in Chapter III. This is done by forcing the variance/covariance matrix entry of two features to a 0 or near-0 value. By comparing
the bivariate residual to the reference value of 1 it was iteratively determined which direct effects to include. These were set in an ad hoc approach, with each iteration of the model the residuals were examined and starting with the largest value direct effects were included for values greater than 1 (Vermunt & Magidson, 2005, pp. 75–76).

As will be mentioned below, allowing interactions between features within a single cluster proved to be the difference between several unstable models and one confidently solid model for the 5–90 login sample.

**Arriving at stable models.** As stated above, the 5–90 group was first to be analyzed but it would not come to a stable model over approximately 60 analysis runs. The BIC varied as follows: 34% of results with $\text{BIC} = 156,230.7869$; 27% with $\text{BIC} = 156,253.4293$; 18% with $\text{BIC} = 156,238.6526$, with the remaining 20% split between 4 other values of nominally different BICs. To see if the instability was caused by a yet undetected dominating group, this larger set was put to the side to see how the smaller, but assumed more active, user group (10–90 logins) data would turn out.

The 10–90 visit data quickly converged on a solid model that consistently provided a single model. This set required 25 features to explain, had a BIC of 98,006.01, and required 171 parameters.

After the 10–90 group’s model stabilized, attention was given again to the 5–90 group. It was then discovered that one additional direct effect needed to be included to obtain a solid model. This stable model had a lower BIC than all previous unstable models for the 5–90 group. The 5–90 group’s model also required two fewer parameters (for a total of 23 features) with only 12 additional parameters than the 10–90 group’s initial
stable model and yet it almost doubled the number of users—a fair tradeoff in complexity for a much more inclusive model.

Attention was again turned to the 10–90 group to see if a reduction to 23 features (removing the percent of days registered that they visited the IA [ActivePct] and average resources per folder [AvgResPerFolder]) produced a better model for the group with a smaller \( n \). After including two additional direct effects, a new model was found with a lower BIC and required fewer parameters—indicating a better and less complex fit.

Table 5 shows the final direct effects included in the final models. Examining the kinds of features that had direct effects included, it was easy to see that most are related in concept and it was no surprise that a correlation was found in the data. It is interesting to note that the majority of features were from the project creation and edit activity category (i.e., ratio of words to resources used [WordsPerRes] and the average per project [AvgWordsPerRes], the average of words per project [AvgPrjWords], the average size of word count changes per edit [AvgNumWordDiff] with the soundex difference [AvgSoundexDiff], and finally the average number of resources in each project [AvgResPerPrj]).

Perhaps it was possible that the number of resources compared to the number of words may have indicated the importance, quality, or complexity of the resource relevant to the user’s own words, ideas, or even confidence in the resource. For example, a resource considered high quality may have needed no introduction, and therefore would have required few words to support it. On the other hand, a resource considered highly complex may have required a lot of explanation and therefore would have had many
Table 5

*Included Direct Effects (“X” indicates inclusion)*

<table>
<thead>
<tr>
<th>Features Involved</th>
<th>5–90 Logins</th>
<th>10–90 Logins</th>
</tr>
</thead>
<tbody>
<tr>
<td>WordsPerRes</td>
<td>AvgPrjWords</td>
<td>X</td>
</tr>
<tr>
<td>AvgWordsPerRes</td>
<td>AvgPrjWords</td>
<td>X</td>
</tr>
<tr>
<td>AvgWordsPerRes</td>
<td>WordsPerRes</td>
<td>X</td>
</tr>
<tr>
<td>AvgNumWordDiff</td>
<td>AvgPrjWords</td>
<td>X</td>
</tr>
<tr>
<td>AvgNumWordDiff</td>
<td>WordsPerRes</td>
<td>X</td>
</tr>
<tr>
<td>AvgNumWordDiff</td>
<td>AvgWordsPerRes</td>
<td>X</td>
</tr>
<tr>
<td>AvgSoundexDiff</td>
<td>AvgPrjWords</td>
<td>X</td>
</tr>
<tr>
<td>AvgSoundexDiff</td>
<td>WordsPerRes</td>
<td>X</td>
</tr>
<tr>
<td>AvgSoundexDiff</td>
<td>AvgWordsPerRes</td>
<td>X</td>
</tr>
<tr>
<td>AvgInterSessResView</td>
<td>AvgInterSessPrjView</td>
<td>X</td>
</tr>
<tr>
<td>AvgResPerPrj</td>
<td>AvgPrjWords</td>
<td>X</td>
</tr>
<tr>
<td>AvgResPerPrj</td>
<td>AvgSoundexDiff</td>
<td>X</td>
</tr>
<tr>
<td>AvgResPerPrj</td>
<td>AvgInterSessResView</td>
<td>X</td>
</tr>
</tbody>
</table>

words to explain it. Another plausible cause of the correlation was that users who put more effort, in terms of words or great care when selecting resources, into the creation of their projects also would have ensured that the projects saw a lot of usage and emphasized the importance of the resources in their projects.

Only two features included in direct effects were from the Project Usage category: the average number of visits to the user’s projects between user visits.
(AvgInterSessPrjView) and the number of resources visited from the user’s projects between user visits (AvgInterSessResView). Using the same kind of logic as above, these correlated features could be understood by usage observed with some library media staff who had long lists of links with very little user-added text (a low words-to-resource ratio). These lists of links were the browser home page in the computer lab and had many visits and a lot of resource views as a result. Those projects may have contributed to a higher correlation between these two features.

Without the LCA extension of including direct effects, this kind of analysis would have been much more difficult and less descriptive of the user characteristics.

Table 6 holds the overall model information for the two models. Note that a little less than half of the total records available after culling nonusers were only in the 5–90 group (366 – 197 = 169 or 46%) and that after dropping cases with missing data in a case-wise strategy, there were about 74% (270/366 for 5–90 logins and 146/197 for 10–90) of both groups that had enough valid data for analysis. The larger group (5–90 logins) required more direct effects to be included thus increasing the number of parameters in the model (Nparams).

Table 6

<table>
<thead>
<tr>
<th>Model</th>
<th>Nrecords</th>
<th>N</th>
<th>Nparams</th>
<th>LL</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>5–90 Logins</td>
<td>366</td>
<td>270</td>
<td>183</td>
<td>-77521.78</td>
<td>156068.07</td>
</tr>
<tr>
<td>10–90 Logins</td>
<td>197</td>
<td>146</td>
<td>155</td>
<td>-48204.82</td>
<td>97182.10</td>
</tr>
</tbody>
</table>
With two working models it was necessary to evaluate them both to see which would provide the most meaningful characterization.

**Evaluation and Presentation of LCA Results**

With stable models, the KDD process proceeded to evaluation of the analysis results before they were presented in a more understandable way. The results were yet another data set to be explored and understood before the presentation could proceed. As stated above, the two models differed: one modeled a larger, but proportionally less active (in terms of visits) sample of users (with 5–90 logins) and the other modeled a smaller, and proportionally more active subset of users (with 10–90 logins). The understanding gained in this section’s work was used heavily in the next section where the clusters were characterized and given names.

**Surface-level model comparisons.** Evaluation of the results began with simple comparisons of the two models. From Table 6, we see that the models differ in complexity (Nparams) and the number of users they described (N). As noted above, both models appeared statistically to be the best choice out of all the possibilities for the respective samples. While the complexity differences are not too great, the number of users they describe was very different. Further descriptions of result evaluations below provide valuable insight into how different and similar these models are.

**Cluster size comparison.** In an effort to understand how the two models differed relative to one another—and from there determine how similar the two really are in discerning user behaviors—cluster size (and later cluster means) was examined in four
ways: (a) raw values plotted linearly, (b) raw values plotted logarithmically, (c) normalized values plotted linearly, and (d) class changes between models. The first three used Figure 14 and Table 7 while the final scrutiny looked at Table 8. Surprising results came from these several analysis that challenged previous assumptions about what behaviors would be common and different between the models.

Figure 14 demonstrates the difference between the raw-value plot and the proportional or normal plot as a method for comparing models in a relative manner—something that the log-scale could not do (see the middle panel of Figure 14). When the raw cluster size panel of Figure 14 was examined, it became clear that there were many more users in Cluster 1 than 2, 2 than 3, and 3 than 4, albeit the differences between cluster size diminished between models as we move up the clusters (as represented by decreasing negative slopes).

Table 7

*Cluster Size Summary*

<table>
<thead>
<tr>
<th>Solution</th>
<th>Cluster</th>
<th>N</th>
<th>Normalized N</th>
</tr>
</thead>
<tbody>
<tr>
<td>5–90 Logins</td>
<td>1</td>
<td>167</td>
<td>0.62</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>62</td>
<td>0.23</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>33</td>
<td>0.12</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>8</td>
<td>0.03</td>
</tr>
<tr>
<td>10–90 Logins</td>
<td>1</td>
<td>81</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>38</td>
<td>0.26</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>21</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>6</td>
<td>0.04</td>
</tr>
</tbody>
</table>
Figure 14. Cluster size panel plot: raw, log, and normalized. A plot of raw (top), $\log_{10}$ (middle), and normalized (bottom) cluster sizes for Clusters 1–4 on 2 models.
Table 8

*Cluster Change Summary Between Models*

<table>
<thead>
<tr>
<th>Model cluster number</th>
<th>5–90 logins</th>
<th>10–90 logins</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td><strong>Total between-model movement</strong></td>
<td><strong>15</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In the sequence of analyses, the raw cluster size panel of Figure 14 gave rise to concerns that the two models were, in fact, addressing two different models because the values were so very different. However, in comparing N sizes, it was helpful to normalize values and look at the proportion in lieu of raw values. Therefore, the cluster size was normalized by the number of users included in the model and the results are in Table 7. The near equivalence of proportional cluster size was somewhat surprising. If the cluster profiles were also similar from one sample to another, then this would mean that the same kinds of behavior occurred proportionally in each sample—regardless of the number of logins. Perhaps the number of logins (NumVisits) was not a good differentiator between active and not-as-active users.

The lower panel of Figure 14 is a true proportion of the total number of participants for each cluster size for the two models. Even though there were many more users in the 5–90 model, the proportion of users in each cluster were nearly identical. This discovery strengthened the notion that the two models were very close and, indeed, the
features were useful in classifying not only the moderately active users (10–90 logins), but also those who were a bit less active (five to nine logins).

Table 8 summarizes the changes of users who were common between models (the 10–90 login group) who changed class membership from one model to another. Only 15 users (5.5% of the 5–90 group and 10.3% of the 10–90 group) changed cluster membership. Two thirds of the movement (10) was to a lower cluster with eight moving to Cluster 1 and seven moving to Cluster 3. Clusters 2 and 4 did not receive any of the users that changed cluster between models. Otherwise, cluster membership did not change for the 10–90 login group, therefore most of the differences between the two models is due to the addition of the five to nine login portion of the 5–90 login group.

**Cluster means comparison.** Part of the output from LatentGold was a table of means for each feature for each cluster and Tables 9 and 10 contain those means. It was observed that means differed between the two models and were generally higher in the 10–90 group (Table 10). These differences might have been small and unhelpful, but needed to be looked at more closely.

Comparison of tabular data was not a simple matter when looking for a big picture, however, the similarities and differences between the two models could be more easily discerned through visual analysis following a transformation to a more visual display method. Several figures were produced to aid in examination of the cluster size and means, and will now be introduced with a brief discussion of what knowledge was learned or gained from different types of display.
Table 9

*Cluster Means for Users with 5–90 Logins*

<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
<th>Cluster 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration and Usage</td>
<td>RegDayCnt</td>
<td>43.60</td>
<td>20.95</td>
<td>43.82</td>
<td>22.39</td>
</tr>
<tr>
<td></td>
<td>NumVisits</td>
<td>11.49</td>
<td>18.71</td>
<td>14.70</td>
<td>37.83</td>
</tr>
<tr>
<td></td>
<td>NumUserSess</td>
<td>9.16</td>
<td>13.97</td>
<td>11.21</td>
<td>29.82</td>
</tr>
<tr>
<td></td>
<td>NumDaysVisited</td>
<td>5.51</td>
<td>8.72</td>
<td>7.94</td>
<td>17.58</td>
</tr>
<tr>
<td>Resource Collection</td>
<td>NumResources</td>
<td>14.25</td>
<td>28.87</td>
<td>24.97</td>
<td>94.90</td>
</tr>
<tr>
<td></td>
<td>NumFolders</td>
<td>4.74</td>
<td>5.79</td>
<td>5.21</td>
<td>11.09</td>
</tr>
<tr>
<td>Project Creation and</td>
<td>NumPrjWords</td>
<td>159.25</td>
<td>901.46</td>
<td>398.58</td>
<td>3486.43</td>
</tr>
<tr>
<td>Edit</td>
<td>AvgPrjWords</td>
<td>30.51</td>
<td>154.79</td>
<td>60.84</td>
<td>364.21</td>
</tr>
<tr>
<td></td>
<td>WordsPerRes</td>
<td>14.82</td>
<td>53.63</td>
<td>20.66</td>
<td>69.00</td>
</tr>
<tr>
<td></td>
<td>AvgWordsPerRes</td>
<td>13.17</td>
<td>43.53</td>
<td>18.22</td>
<td>56.81</td>
</tr>
<tr>
<td></td>
<td>NumPrjEditSess</td>
<td>4.78</td>
<td>8.13</td>
<td>7.18</td>
<td>18.19</td>
</tr>
<tr>
<td></td>
<td>AvgNumWordDiff</td>
<td>11.32</td>
<td>33.87</td>
<td>17.90</td>
<td>56.81</td>
</tr>
<tr>
<td></td>
<td>AvgSoundexDiff</td>
<td>85.08</td>
<td>557.75</td>
<td>203.30</td>
<td>1377.84</td>
</tr>
<tr>
<td></td>
<td>AvgResPerPrj</td>
<td>2.55</td>
<td>4.39</td>
<td>3.97</td>
<td>9.86</td>
</tr>
<tr>
<td></td>
<td>NumPrj</td>
<td>4.97</td>
<td>5.63</td>
<td>6.06</td>
<td>13.58</td>
</tr>
<tr>
<td>Project Usage</td>
<td>NumStudentPrj</td>
<td>4.53</td>
<td>3.97</td>
<td>5.39</td>
<td>12.33</td>
</tr>
<tr>
<td></td>
<td>NumPublicPrj</td>
<td>4.14</td>
<td>2.26</td>
<td>5.24</td>
<td>3.63</td>
</tr>
<tr>
<td></td>
<td>PctPublicPrj</td>
<td>0.81</td>
<td>0.41</td>
<td>0.87</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>NumPrjVisits</td>
<td>259.66</td>
<td>282.44</td>
<td>1526.01</td>
<td>1450.19</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessPrjView</td>
<td>9.15</td>
<td>21.03</td>
<td>40.63</td>
<td>49.80</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessResView</td>
<td>14.00</td>
<td>49.28</td>
<td>55.03</td>
<td>97.35</td>
</tr>
<tr>
<td></td>
<td>NumStuLogin</td>
<td>18.78</td>
<td>98.29</td>
<td>99.54</td>
<td>293.62</td>
</tr>
<tr>
<td></td>
<td>NumStuResViews</td>
<td>4.20</td>
<td>3.84</td>
<td>5.00</td>
<td>8.97</td>
</tr>
</tbody>
</table>
Table 10

Cluster Means for Users with 10-90 Logins

<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
<th>Cluster 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration and</td>
<td>RegDayCnt</td>
<td>37.18</td>
<td>19.92</td>
<td>22.53</td>
<td>21.01</td>
</tr>
<tr>
<td>Usage</td>
<td>NumVisits</td>
<td>17.47</td>
<td>22.97</td>
<td>23.00</td>
<td>39.80</td>
</tr>
<tr>
<td></td>
<td>NumUserSess</td>
<td>13.46</td>
<td>16.71</td>
<td>17.86</td>
<td>30.96</td>
</tr>
<tr>
<td></td>
<td>NumDaysVisited</td>
<td>7.30</td>
<td>10.21</td>
<td>11.85</td>
<td>18.63</td>
</tr>
<tr>
<td>Resource Collection</td>
<td>NumResources</td>
<td>18.30</td>
<td>30.79</td>
<td>41.52</td>
<td>106.22</td>
</tr>
<tr>
<td></td>
<td>NumFolders</td>
<td>4.83</td>
<td>5.82</td>
<td>6.24</td>
<td>12.62</td>
</tr>
<tr>
<td>Project Creation</td>
<td>NumPrjWords</td>
<td>178.09</td>
<td>939.39</td>
<td>812.14</td>
<td>4133.45</td>
</tr>
<tr>
<td>and Edit</td>
<td>AvgPrjWords</td>
<td>35.13</td>
<td>160.87</td>
<td>104.44</td>
<td>434.10</td>
</tr>
<tr>
<td></td>
<td>WordsPerRes</td>
<td>14.82</td>
<td>54.71</td>
<td>28.82</td>
<td>80.60</td>
</tr>
<tr>
<td></td>
<td>AvgWordsPerRes</td>
<td>12.66</td>
<td>41.33</td>
<td>24.37</td>
<td>66.92</td>
</tr>
<tr>
<td></td>
<td>NumPrjEditSess</td>
<td>5.91</td>
<td>9.08</td>
<td>11.61</td>
<td>17.78</td>
</tr>
<tr>
<td></td>
<td>AvgNumWordDiff</td>
<td>10.84</td>
<td>31.85</td>
<td>28.15</td>
<td>68.82</td>
</tr>
<tr>
<td></td>
<td>AvgSoundexDiff</td>
<td>108.34</td>
<td>596.30</td>
<td>334.99</td>
<td>1669.61</td>
</tr>
<tr>
<td></td>
<td>AvgResPerPrj</td>
<td>3.41</td>
<td>4.22</td>
<td>5.82</td>
<td>10.67</td>
</tr>
<tr>
<td></td>
<td>NumPrj</td>
<td>4.93</td>
<td>5.61</td>
<td>7.47</td>
<td>15.10</td>
</tr>
<tr>
<td>Project Usage</td>
<td>NumStudentPrj</td>
<td>4.14</td>
<td>3.97</td>
<td>5.71</td>
<td>14.42</td>
</tr>
<tr>
<td></td>
<td>NumPublicPrj</td>
<td>3.62</td>
<td>1.71</td>
<td>5.37</td>
<td>2.68</td>
</tr>
<tr>
<td></td>
<td>PctPublicPrj</td>
<td>0.70</td>
<td>0.31</td>
<td>0.71</td>
<td>0.54</td>
</tr>
<tr>
<td></td>
<td>NumPrjVisits</td>
<td>271.29</td>
<td>195.37</td>
<td>1532.02</td>
<td>1196.94</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessPrjView</td>
<td>12.12</td>
<td>18.47</td>
<td>62.73</td>
<td>43.61</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessResView</td>
<td>22.31</td>
<td>51.50</td>
<td>123.34</td>
<td>33.54</td>
</tr>
<tr>
<td></td>
<td>NumStuLogin</td>
<td>34.20</td>
<td>92.50</td>
<td>280.36</td>
<td>173.13</td>
</tr>
<tr>
<td></td>
<td>NumStuResViews</td>
<td>3.85</td>
<td>3.74</td>
<td>5.85</td>
<td>9.95</td>
</tr>
</tbody>
</table>
**Raw cluster means comparison plot.** The same sequence was followed in the examination of cluster means as in the cluster size: raw, then log, then proportional.

Figure 15 is a simple look at the cluster means for the 5–90 model. With the scale being so large on the vertical axis, only the greatest absolute differences are discernible.

Just as with the tabular display of the means, it was difficult to observe trends except that for the most part, Cluster 1’s feature means were lower than all the others, Cluster 2’s feature means were generally just above the means of Cluster 1. Likewise, the means of Cluster 3 were generally just above Cluster 2’s means and below Cluster 4’s means, and Cluster 4’s were generally the highest.

Notice how the feature means of Clusters 1 and 2 did not differ much between models, this indicated that the behaviors of the five to nine login users (the real difference between the two samples) had little impact on Clusters 1 and 2, the largest clusters.

Cluster 3’s feature means had more differences between models with the 5–90 login model slightly lower on the number of project words (NumPrjWords), average nontrivial project changes (AvgSoundexDiff), and number of student logins (NumStuLogin). Cluster 4’s between model feature means differences were similar to those of Cluster 3, but with the 5–90 login model was slightly higher on project count (NumPrj) and number of student logins (NumStuLogin).

Because we can only visually see differences between raw cluster means when the differences are very large due to the scale differences between features, another method of visual analysis was needed.
Figure 15. A plot of the raw means for Clusters 1–4 on the 5–90 and 10–90 login models. The vertical axis is the raw feature value of count (cnt/num), average (avg/per), and percents (pct) as indicated in the feature names.
**Log-scaled cluster means comparison plot.** The same data in Figure 15 was plotted against a log-scale to reduce the disparity of vertical range between features (see Figure 16). While interpretation of log-scales is not always intuitive, this display method had the benefit of seeing each difference between models as they were proportional to the scale of the variable. In other words, proportionally large differences on a feature with a small scale looked the same as proportionately large differences on a feature with a larger scale.

Notwithstanding the spacing with the log-scale, most of the differences between clusters were parallel—Cluster 1 the lowest and Cluster 4 the highest—and that the exceptions (and most interesting parts of this figure) were only in the number of same-day registrants (RegDayCnt), the number of public projects NumPrj, and Project Use feature group of Clusters 3 and 4 (e.g., PctPublicPrj, NumStuLogin).

Upon viewing this plot, the perception surfaced that the differences between the two models (and, hence, the underlying user behaviors) were minimal even though the number of users were so different. In other words, this highlighted again that users with five to nine logins behaviors were not so different from those found in users with 10–90 logins; and that using the model to the larger group (5–90 logins) would provide a more generalizable model.

**Normalized means comparison plots.** The panel plots in Figures 17 and 18 were created by normalization within each model’s features by dividing each cluster feature mean by the sum of the 4 cluster means as follows:
Figure 16. A log-scale plot of the means for Clusters 1–4 on the 5–90 and 10–90 login models. The vertical axis is the log-transformed feature value of count (cnt/num), average (avg/per), and percents (pct) as indicated in the feature names.
For each feature $m$, and each cluster $j = \{1 \text{ to } k\}$:

$$
\frac{\mu_{m,j}}{\sum_{i=1}^{k} \mu_{m,i}}.
$$

The result was a normalized look at each feature directly compared to the other clusters within the model. In Figure 17 the results are presented by model and in Figure 18 they are presented by cluster.

Had Figures 17 and 18 been rendered as stacked or cumulative chart, the top of each vertical axis would be 1.0. A scale from 0 to 0.7 helped increase the vertical differentiation—but needed to be remembered in interpretation since the proportion from the bottom of the plot to each line was proportional, the space from the lines to the highest possible value (1.0) was not.

The two models were compared within each cluster, and as with the cluster size, it was observed that all four clusters had nearly identical normalized means between the two models—in other words, the relative value of each mean to the other means was nearly identical. This was quite a surprising and unexpected result after viewing initial results and the difficulty getting a stable model for the 5–90 visit sample. In addition, the relative sameness between the models indicates that the same behaviors existed in users with five to nine login group (the true difference between the two samples) as in the 10–90 group—although most of the users unique to the 5–90 login sample were clustered in the first three clusters as indicated by the difference between models in the top panel of Figure 14.
Figure 17. A plot of normalized means for the 5–90 and 10–90 login models for Clusters 1–4. Means normalized to the proportion of the sum of means—the relative size of the mean compared to the other means in the model. The vertical axis is the normalized feature value of count (cnt/num), average (avg/per), and percents (pct) as indicated in the feature names.
Figure 18. A plot of normalized means for Clusters 1–4 for the 5–90 and 10–90 login models. Means normalized to the proportion of the sum of means—the relative size of the mean compared to the other means in the model. The vertical axis is the normalized feature value of count (cnt/num), average (avg/per), and percents (pct) as indicated in the feature names.
The major differences between models occurred in Cluster 3 with the number of same-day registrants (RegDayCnt, 5–90 being higher), and Clusters 3 and 4 in the number of project visits between user sessions (AvgInterSessPrjView), the number of resource visits between user sessions (AvgInterSessResView), number of student logins (NumStuLogin), and the number of student resource views from the user’s projects (NumStuResViews). These project and resource usage differences are nearly opposite—where Cluster 3 is higher for the 10–90 group, it is low for the 5–90 group and vice-versa on Cluster 4.

The normalized means were transformed further by binning the values into High, Medium, and Low. Determining the boundaries for these bins followed the following logic.

If the cluster means of a feature were all the same for four clusters, then the normalized mean would be 0.25 (1.0/4). Therefore, 0.25 was considered an equal (or Medium) state.

When the normalized value dropped below 0.20, then the mean accounted for less than one fifth of the sum leaving 80% to the other three, and at this point, a value considered relatively low. Likewise, if the value was larger than 40%, then it began to be weightier and left less than 60% of the sum for the other three to share. For this reason, values greater than 0.40 were considered high.

The binned labels of each model’s clusters were then tabularly displayed as combined when the same and jointly when different (e.g., when both were High then display High, when one was High and the other Med, then display High/Med with Cluster
3 on the left). This display was helpful in discerning the patterns of differences between clusters of different models.

Finally, to aid with the visual discernment, color coding was also added to this table as kind of a heat map. Color coding was not completely unlike the idea behind output from self-organizing maps where both position and color have significance. Position in the table was also important as it grouped related features. The final results of this process were placed in Table 11 and figured heavily in the characterization of behavior clusters.

**Box plot analysis.** Again, box plots served to visually inspect the LCA clusters with more robust visual depiction of the data than means alone can give. Figures 19 and 20 were helpful in seeing more details of the data than Tables 9 and 10.

Figures 19 and 20 are the same data as used in Figure 12, but now distributed across clusters.

Differences that manifested at a cluster level between the figures were mainly that in Figure 20 the inter-quartile ranges are generally larger for the users in Cluster 4 with somewhat expanded with Cluster 3’s IQRs, and less with Clusters 1 and 2. Good examples of these changes can be seen in:

- several dramatic shifts in the percent of projects made public (PctPublicPrj) across all clusters;
- registration day count (RegDayCnt), the number of projects (NumPrj), the number of resources viewed between each visit (AvgInterSessResView), and the number of student logins (NumStuLogin) with Clusters 3 and 4; and
• Cluster 4 where large differences in the number of folders (NumFolders),
  average of project words (AvgPrjWords), the average words per resource
  (AvgWordsPerRes), and average size of project changes (AvgNumWordDiff).

Clusters 1 and 2 did not change too dramatically except as noted in the list above.

Several of the aforementioned differences could be attributed to changes in cluster
membership of those users common between the two samples. In Table 8 it was noted that
two users moved from Cluster 4 in the 5–90 login sample’s model to Cluster 3 in the
10–90 login sample’s model. Table 7 showed that the fourth cluster’s size is quite small,
therefore any data added or removed will have a large effect on the descriptive statistics
displayed in the box plots. This phenomenon is easy to see with features such as the
number of student logins (NumStuLogin) where the inter-quartile range varied
dramatically between Clusters 3 and 4 when a user with a large number of visits moved
between them. In other circumstances, Cluster 4 changed but the user that moved was
more in line with those already in Cluster 3 so it did not change so dramatically (see the
number of folders box plot [NumFolders]).

In addition, dramatic instances of elongated and returning notches were found in
Figures 19 and 20, most often with the data from Cluster 4. The long notches were an
artifact of the small n in those groups.

Other visual analyses. Other ways of presenting this data (other than tabular,
line-chart, and box plot) could have been in surface-area plots where hills and valleys
would have indicated the differences between groups. However, this method was not
utilized in this study.
Figure 19. A matrix of box & whisker plots of all features for Clusters 1–4 for the 5–90 login group’s model. Larger versions of these plots are in Appendix C
Figure 20. A matrix of box & whisker plots of all features for Clusters 1–4 for the 10–90 login group’s model. Larger versions of these plots are in Appendix C
Accepted solution. From the knowledge gained from the plots produced it was clear (and surprising) that the two models were nearly identical in their composition of discernible characteristics. Therefore, they could largely be treated as one for the purposes of characterizing these emergent clusters.

Had the focus of this work been predictive or proscriptive in nature, the more general model (for the 5–90 login group) would have been preferred because the number of users successfully modeled was nearly doubled. However, since this was a descriptive analysis and not a predictive one, the two models were characterized together and differences described where applicable.

Presentation Revisited: Characterization of Results

With the technical exploration and description of the model complete, it was time to proceed with the research purpose of characterizing user behavioral patterns in plain terms (see Table 3). The first characterization accomplished was in the context of each meaningful IA activity in answer to the second question of the pattern mining purpose, “Specifically, what patterns arise within meaningful IA activity?” This was done so that the range of activity in each category could be understood.

Next, the clusters were characterized and named utilizing the knowledge gained in the category activity descriptions in order to answer the characterization purpose, “How can these patterns be described and characterized in ways that are germane to the purposes of the IA?” Both characterizations primarily used Figures 14, 18, 19, and 20 with Table 11 to interpret the results.
Table 11

_Binned and Shaded Normalized Cluster Means for 5–90 and 10–90 Logins_

<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
<th>Cluster 4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Registration and Usage</strong></td>
<td>RegDayCnt</td>
<td>Med</td>
<td>Low</td>
<td>Med</td>
<td>Low/Med</td>
</tr>
<tr>
<td></td>
<td>NumVisits</td>
<td>Low</td>
<td>Med</td>
<td>Low/Med</td>
<td>High/Med</td>
</tr>
<tr>
<td></td>
<td>NumUserSess</td>
<td>Low</td>
<td>Med</td>
<td>Low/Med</td>
<td>High/Med</td>
</tr>
<tr>
<td></td>
<td>NumDaysVisited</td>
<td>Low</td>
<td>Med</td>
<td>Low/Med</td>
<td>High/Med</td>
</tr>
<tr>
<td><strong>Resource Collection</strong></td>
<td>NumResources</td>
<td>Low</td>
<td>Low</td>
<td>Low/Med</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>NumFolders</td>
<td>Low</td>
<td>Med/Low</td>
<td>Low/Med</td>
<td>High</td>
</tr>
<tr>
<td><strong>Project Creation and Edit</strong></td>
<td>NumPrjWords</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>AvgPrjWords</td>
<td>Low</td>
<td>Med</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>WordsPerRes</td>
<td>Low</td>
<td>Med</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>AvgWordsPerRes</td>
<td>Low</td>
<td>Med</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>NumPrjEditSess</td>
<td>Low</td>
<td>Med</td>
<td>Low/Med</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>AvgNumWordDiff</td>
<td>Low</td>
<td>Med</td>
<td>Low/Med</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>AvgSoundexDiff</td>
<td>Low</td>
<td>Med</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>AvgResPerPrj</td>
<td>Low</td>
<td>Med/Low</td>
<td>Low/Med</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>NumPrj</td>
<td>Low</td>
<td>Low</td>
<td>Med</td>
<td>High</td>
</tr>
<tr>
<td><strong>Project Usage</strong></td>
<td>NumStudentPrj</td>
<td>Low</td>
<td>Low</td>
<td>Med</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>NumPublicPrj</td>
<td>Med</td>
<td>Low</td>
<td>Med/High</td>
<td>Med/Low</td>
</tr>
<tr>
<td></td>
<td>PctPublicPrj</td>
<td>Med</td>
<td>Low</td>
<td>Med</td>
<td>Med</td>
</tr>
<tr>
<td></td>
<td>NumPrjVisits</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td>High/Med</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessPrjView</td>
<td>Low</td>
<td>Low</td>
<td>Med/High</td>
<td>High/Med</td>
</tr>
<tr>
<td></td>
<td>AvgInterSessResView</td>
<td>Low</td>
<td>Med</td>
<td>Med/High</td>
<td>High/Med</td>
</tr>
<tr>
<td></td>
<td>NumStuLogin</td>
<td>Low</td>
<td>Low</td>
<td>Low/High</td>
<td>High/Med</td>
</tr>
<tr>
<td></td>
<td>NumStuResViews</td>
<td>Low</td>
<td>Low</td>
<td>Med</td>
<td>High</td>
</tr>
</tbody>
</table>

*Note.* Means were binned into High ($\mu > 0.40$), Med ($0.40 \geq \mu \geq 0.20$), and Low ($\mu < 0.20$).
In this section, the two competing models will be largely treated as one, except for a few notable differences in registration and project usage features that will be further described as the characterization proceeds.

Table 11 was constructed utilizing the color-coded and binned normalized cluster feature means as a tabular heat map with the meaningful IA activities roughly aligned with the features. Note that the shading indicated the relative magnitude of each mean compared across clusters within each model. The areas of light and darker shading facilitated discernment of patterns that were not easily seen with text alone.

The alignment of meaningful IA activity to feature was rough in the sense that the features may have belonged to two activities. For example, project count (NumPrj) aligned well with the project creation and edit feature group, but also had something in common with the project usage activity group. The same can be said for the average number of resources per project (AvgResPerPrj) and the resource collection activity features. Even with this shortcoming, the alignment worked well for the purpose of this study.

**Characterization by Meaningful Activity**

By examination of Figure 18 (the normalized plot) and Figure 16 (the log plot), with Tables 9 and 10 (the means tables) and Table 11 (the heat map) it was apparent that there were different shapes to each cluster’s plotted means in the figures and different patterns in the table that indicated the relative level of activity. Alignment of features to meaningful IA activity categories further assisted by grouping the features into
larger-picture concepts (e.g., see the vertical lines in Figure 18). Inside each meaningful activity, therefore, there are subgroups of features.

The normalized plot and heat map were most heavily used in the interpretation of the results. However, it was important to refer back to the means tables and box plots (Figures 19 and 20 in order to gain a sense of the magnitude of actual activity represented and the differences in the numbers of users in each cluster. The understanding gained in the construction of Table 8 that 90% of users in the 10–90 login group were also in the same cluster number in the 5–90 model, so the models clustered in a very similar and stable pattern.

As an example of integrating the knowledge from the tables and figures, consider what a Medium level of visits activities (e.g., NumVisits) really means for the users in Cluster 2. Tables 9 and 10 indicated an average of about 19 to 23 logins for the 5–90 and 10–90 groups, respectively. The box plots further contribute to the understanding of how the feature was distributed across the clusters and that Cluster 1 had a tight distribution with many outliers and Cluster 4 had a wide distribution. In general, Clusters 1 and 3 have lower medians than 2 and 4.

After this examination, it was obvious that compared to popular online applications the IA usage was very low, but still sufficient to better understand teacher behavior when using online authoring services in the context of online digital libraries.

The shapes on the plots and patterns in the table are now described for each meaningful IA activity.
Registration and usage. As denoted in the title of this activity area, there are two sub-activities: registration and usage. These two areas cover actions known to be performed by the user when visiting the IA and logging in as opposed to regular projects visits (see below) that could be the user or someone else.

Registration. From the figures and tables, it is apparent that the users in Cluster 3 had a relatively large difference between models for the number of same-day registrants (RegDayCnt) than those in the other three clusters. The users of Cluster 1 and the 5–90 group of Cluster 3 had the highest number of users registering on the same day.

When the trends in the relative feature means and box plots for Clusters 3 and 4 were compared, it was interesting to notice that when the five to nine login users were added, more users registered in a single day (i.e., the number of same-day registrants [RegDayCnt]) for Cluster 3 and the other usage activity relative feature means were slightly lower. This trend continued across other activity groups.

The relative means for Cluster 4, on the other hand, were slightly lower for the number of same-day registrants, but the other usage features were slightly higher relative to the other clusters. Table 7 informed that there were more of the five to nine users clustered in Cluster 3 ($n = 33$) than the 10–90 users ($n = 21$) and the addition of these less-active users seems to have changed the activity level of Cluster 3 between the two models—readily apparent in the box plots.

Having a large number of other registrations on the same day can indicate participation in a workshop, conference presentation, or classroom instruction on the IA; of these groups of registrants, few had adopted the IA as possibly indicated by the overall
low relative feature means of Cluster 1—the largest cluster. Following this line of thought, the users in Cluster 3 in the five to nine sub-group may be those who somewhat adopted the IA after a large group introduction to the IA as opposed to those who had 10–90 logins, registered with fewer people, and had slightly higher activity levels.

However, since the binned IA introduction source (IAInfoBin) had no statistical impact on the model, this assumption may need to be examined further in future study. For example, it is known that some classroom or online tutorial exposure to the IA did not require participants to register on a single day, perhaps the bins were too few and broad, or the addition of a new feature counting the number of users registering in the seven days centered on the user’s registration date would combine with the number of same-day registrants (RegDayCnt) and assist in differentiating registration patterns.

**Usage.** The visit behavior features of number of logins (NumVisits), number of PHP sessions (NumUserSess), and the number of days in which they logged in (NumDaysVisited) count different statistics about visits behavior. For example if a user would log in and out repeatedly before their PHP Session would expire, then they would have had many more logins than PHP sessions. The log means plot, Figure 16, shows these three follow each other in a nearly linear fashion and the box plots, Figures 19 and 20 show that distribution differences were small between models.

Otherwise, normalized activity levels were all closer to the Low/Medium boundary with the exception of the users in Cluster 4, which hovered around the Medium/High boundary. This indicated that the vast majority of users did not visit much but they did login a couple times per session or on the few days they visited.
Tables 9 and 10 show that while there was usage, the cluster means for the number of logins (NumVisits) only ranged from 11.49 to 39.80 for any cluster across the two models. This was partly inherent in data from more recent registrants, but it indicated that the IA has not been a site daily visited by many (or any) registered users.

**Resource collection.** There were two subtle groups in this category, with one feature each: gathering (NumResources) and organization (NumFolders). Constraints in the system include: (a) each resource was placed in at least one folder; (b) each resource could be saved to multiple folders; (c) a resource could only exist in each folder at most once; and (d) if a user utilized the project copy feature, a new folder was created with all the resources for the new project appropriately copied and placed in a new folder. So, a high number of folders could have indicated many copied projects, or that the user collected resources for many different topics (the granularity of which is determined by the user).

Highlights of this category were with the users of Cluster 1 and their lack of resource gathering altogether and yet have several folders on average, and Cluster 4’s users’ relatively large personal repository building activity with relatively more resources than folders on average than the other user groups (by at least double; see Tables 9 and 10). The range and concentration of these distributions are not extremely different between models as represented on the box plots (Figures 19 and 20).

The direction of the resource to folder ratio was an interesting result: for users in Cluster 4 there were more resources than folders and vice-versa in the other three clusters.
Project authoring activity. This was a complex category with several subgroups: project size, project changes, and number of projects. In the normalized plot, it was evident that the shape of normalized feature means for the users in clusters 1 and 3 are remarkably similar while those representing users in clusters 2 and 4 are unique.

The cluster means in Tables 9 and 10 helped to understand the magnitude of these features. The means for Cluster 4’s users were an entire order of magnitude higher than the other clusters in several instances, often followed by Cluster 2, then 3 and finally 1 as the lowest. Again the box plots confirm these trends across most of the feature quartile numbers and ranges.

Size. This feature subgroup described the aggregated sizes of user projects and consisting of features number of words in all the user’s projects (NumPrjWords), average of words per project (AvgPrjWords), the overall number of words per resource used in projects (AvgWordsPerRes), the average number of resources per project (AvgResPerPrj), and the number of words per resources over all projects (WordsPerRes).

The relative feature means for the users of clusters 1 and 3 were quite parallel with the exception of the average resources per project (AvgResPerPrj), with activity in Cluster 3 being slightly higher. Overall, the size of projects for users of Clusters 1 and 3 was relatively lower than those of users in clusters 2 and 4. These results are confirmed in the box plots.

Note that the users of Cluster 4 were much more verbose in terms of the total words in their projects (NumPrjWords) with respect to the averages, but the median (see
Figures 19 and 20), while still higher, demonstrates the sensitivity of means to outliers and that the real difference may not be as dramatic as the means portray.

**Changes.** Features such as the number of PHP sessions in which a project edit occurred (NumPrjEditSess), the average word count difference between edits (AvgNumWordDiff), and average nontrivial project changes (AvgSoundexDiff) made up this subcategory that exposed how many times and how much the users changed their projects. The relative feature means for Clusters 1 and 3 peaked on the number of PHP sessions in which a project edit occurred, while 2 and 4 generally saw a dip. Again, Cluster 4 dominated the landscape with many changes in the number of words per edit session—when they made changes, they were large ones—which may have been due to the large number of words in each project, so any change to resources or intended direction of the project would require more modification, but is more likely the pull of the outlier as evidenced in Figures 19 and 20.

**Number.** The project count (NumPrj) moved fairly consistently from low to high following the cluster number. Otherwise, there was not much of differentiation in this subcategory.

**Authoring summary.** As mentioned earlier, the consistently low activity of the users in Cluster 1 for this category was indicative of those users who minimally participate, and then stop using the tool. They have edited projects a medium number of times and even placed some resources in projects, but their projects were short on words per project, textual changes per edit session, and did not get many inter-edit-session visits of projects or resources. Again, this has been common among those attending IA
workshops without adoption. Cluster 3 was very similar to Cluster 1 in shape and only a little proportionally higher indicating very similar authoring behaviors, but just a little more active.

Cluster 2’s users authoring behaviors tended to be more verbose per resource and in the size of substantive changes but in relatively fewer sessions—in essence, just the opposite shape from Clusters 1 and 3 for those two subgroups of features. On the other hand, users in Clusters 1 through 3 were much the same in the average number of resources in each project (AvgResPerPrj) and project count (NumPrj) showing that none of them are highly productive in terms of project creation.

The prolific powerhouses of project production were grouped in Cluster 4. They were relatively high in every feature compared to the other users, with respect to the normalized average. However, on inspection of the box plots it was evident that an outlier (or a few) were affecting the mean, indicating a more mellow difference. The number of project words (NumPrjWords) and the average number of words per project (AvgPrjWords) were a huge proportion of everything done in the IA—including the average size of substantial change of their projects—again, though, by just a few of the users.

**Project usage.** This was another complex category and perhaps the most interesting because of differences between the two models. It has also been subdivided into subgroups: publishing status and use. Additionally, while project count (NumPrj) and the average number of resources in each project (AvgResPerPrj) were aligned with Project Authoring, they can impact this category as well.
**Publish status.** Three features indicated how many projects are published and to whom: students (NumStudentPrj), the public (NumPublicPrj), and the percent of all the user’s projects publicly published (PctPublicPrj). When combined with project count (NumPrj) a picture emerged of what a user tended to do with their projects (i.e., with whom did they share?).

Student-published projects appeared to be relatively more common among users in Clusters 3 and 4, while users in Clusters 1 and 3 tended to publish to the public more. Users in Cluster 2 seem not to have shared much at all.

**Usage.** Combining the total number of project visits (NumPrjVisits), the number of project visits between user sessions (AvgInterSessPrjView), the number of resource visits between user sessions (AvgInterSessResView), number of student logins (NumStuLogin), and the number of student resource views from the user’s projects (NumStuResViews) into an overall view of project usage, this subcategory was the most interesting of all because of the large disparity among Clusters 3 and 4 between the two models.

In general, use by students followed an emphasis on publishing to students. This may also indicate that users who produced projects with their students in mind were reluctant or did not think to share with the public. An anecdotal example situation where public sharing may be limited is when teachers, who used the IA in teams, shared common user and student logins in a school-, subject-, or grade-wise fashion in order to facilitate coordinated instruction. Such efforts had a school-centric feel and may not have been of interest to users outside that group.
It did not always occur that the overall project views follow publishing to the
public. Users of Cluster 1 had a greater proportion of their projects publicly published
than to student, but they did not receive many visits on those projects (perhaps a result of
low interest in the IA and subsequent low quality). Users grouped in Cluster 3 (with the
10–90 group) and Cluster 4 (with the 5–90 group) have many total visits, but they also had
frequently published to and had many visits from students.

For this subgroup, it appeared that when the five to nine login group is added,
usage drops for Cluster 3 and is enhanced for Cluster 4 in a dramatic way. Initially, it was
assumed that the level of activity demonstrated by the Cluster 4 users in the 10–90 model
was unlikely to be found among the five to nine login users (i.e., the 5–90 login group).
Therefore, the changes were assumed to likely be caused by the transition of some users
between clusters 3 and 4 between models, even though the number of users that moved
from one cluster to the other was small (see Table 8). Additional investigation was
undertaken.

From the box plots in Figures 19 and 20, it was evident that with the number of
student logins (NumStuLogin) varied dramatically between models with the two clusters
of interest (3 and 4).

A median was also calculated for number of student logins (NumStuLogin) in
order to better understand this phenomenon. Indeed, the median number of student logins
(NumStuLogin) assisted in understanding what differences existed between the two
models.
For Cluster 3 in the 5–90 model the median and mean number of student logins (NumStuLogin) were 26 and 100, respectively, while Cluster 4 had 165 and 294 showing that the mean is 2–4 times greater than the median indicating a heavily skewed distribution (common in Poisson distributions). Yet in the 10–90 model the Cluster 3 median and mean were 234 and 280 and Cluster 4 was 85 and 173, which are much closer and have a more narrow distribution (yet still Poisson in nature).

The differences between the medians and means demonstrate the count-nature of the feature, but that the 10–90 model Cluster 3 had less spread than the others, so the differences were not necessarily caused by people moving clusters between models, but rather was because of the broader behaviors needing to be clustered into only four clusters. This can be noted by the difference in the range for Cluster 2 on the number of student login feature, in Figure 19 there are several outliers that do not appear in Figure 20. Therefore, while other features covered above were relatively similar between models, the additional diversity in Clusters 3 and 4 with the five to nine login group showed much less project usage in Cluster 3 and much more in Cluster 4.

**Project usage summary.** In terms of the published state of projects (to students [NumStudentPrj], the public [NumPublicPrj], and the percent of all projects marked public [PctPublicPrj]) the users of Cluster 2 were the least sharing of all but the resources they included were visited. Users in Cluster 1 had a higher rate of projects publicly published but little else different from those in Cluster 2 (again, that could fit with minimally participating workshop groups). Users in Cluster 3 may not have produced many projects, but what they did produce, they also opened up to the public and had a lot
of visits. Users in Cluster 4 produced a lot of projects, but kept them mostly for their own students and opened only a few to the public.

Having a relatively low number of total project visits (NumPrjVisits) also indicated that Cluster 1 users either did not have projects that draw people in and spontaneously share or that they did not advertise their projects (i.e., not used them with their students much)—this corroborates with the student group activity being low on number of student logins (NumStuLogin)).

Cluster 3 users published at a similar magnitude and rate to Cluster 1 (see NumPublicPrj and PctPublicPrj), but also had a lot of student activity (NumStuLogin and NumStuResViews) which drives total project visits (NumPrjVisits) up—again, this corresponded to the trend seen in Student Group Activity.

Cluster 4 seemed to be just the opposite of Cluster 1 in both the number of projects created and the number of visits—where Cluster 4 users created few projects and did not publish them publicly, they had a lot of student visits.

The most notable difference between the two models occurred in this category of use. With the 5–90 model, users with relatively fewer project visits and student logins were probabilistically grouped in Cluster 3 while those in Cluster 4 had much higher (relatively speaking) visits. The reverse was true in the 10–90 model. In the case of Cluster 4, we know that two users in that cluster for the 5–90 model were in Cluster 3 for the 10–90 model and that no one moved to Cluster 4 at the same time.

By examination of the box plots, one could see that the range of Cluster 4 drops dramatically and that the inter-quartile range of Cluster 3 increases dramatically when the
two (and other users from the 5–90 login group are added). Once again, this appears to be the effect of two outliers who really caused the means to be quite volatile—in deed, both distributions were heavily affected.

**Characterization by Cluster**

Examining the results, it was surprising to find the normalized cluster sizes and feature means so nearly identical (see Figures 14 and 18), and even the medians and inter-quartile ranges were relatively stable (see Figures 19 and 20). This indicates that the behaviors seen in more active users (with 10–90 logins) were also seen in the less active sample (those with five to nine logins included in the 5–90 login analysis).

It is also important to note that the raw means (Tables 9 and 10) were lower than anticipated by the researcher and that the differences between Cluster means were quite subtle for most features. In the end the normalized means as depicted in Figure 18 and Table 11, along with the box plots, were mainly used to analyze the relative behavior levels across clusters and features. For example, the number of student projects (NumStudentPrj) are all relatively low (ranging from 3.97 for Cluster 2 to 12.33 for Cluster 4 for the 5–90 login sample) and the first three clusters being nearly indistinguishable in the raw form. So caution is urged that these are relative comparisons within the IA.

This section covers the research question of the Characterize purpose: “How can [the] patterns be described and characterized in ways that are germane to the purposes of the IA?” Each of the four clusters is covered.
Cluster 1: One-hit wonders. In the results from both analyses, this cluster was by far the largest group and about the same size proportionally to both sample sizes (roughly 60%, see Table 7 and Figure 14). The normalized feature means plot showed this cluster’s users behavior was relatively the same in both samples (Figure 18). The heat map (Table 11) shows this cluster as being low on everything except medium on registration (RegDayCnt) and public publishing (NumPublicPrj and PctPublicPrj). The box plots confirm this as Cluster 1 is generally a tight distribution toward the lower-end of the scale.

Given that it was not likely that there would be that many organic registrations on a specific day it had been assumed these users largely consisted of workshop participants or those introduced to the IA in a classroom setting. Surprisingly, however, the self-reported IA information group (IAInfoGrp, e.g., workshop, word of mouth, etc.) and the binned version (IAInfoBin) did not figure statistically into the final models. While they may or may not have attended a workshop, their experience did not make them loyal users of the IA and their behaviors were similar to workshop participants (Recker, Dorward, Dawson, Halioris, et al., 2005; Recker, Dorward, Dawson, Mao, et al., 2005; Recker et al., 2006; Recker & Palmer, 2006) who have only gone through the motions of creating a few projects, collected only a few resources, and more of their projects are public than most others.

Their almost-medium proportion of the number of student resource views from the user’s projects (NumStuResViews) is also indicative of later workshop use where participants were required to utilize at least one of their projects and report how the experience went and discuss the integration of online resources with other participants.
While it is likely that the low usage behaviors by the users in Cluster 1 was because of disinterest, it also may well include users identified by other IA research as those on the extreme ends of the technology ability spectrum (Johnson, 2007). Users who have little technology experience don’t feel comfortable using the IA or do not use it with their students. On the other hand, they may be those who know so much that they can look up resources on their own and build their own webpages and who find the IA authoring system too restrictive.

To sum up Cluster 1, they came in droves, created a few projects and did not publish many, but when they did it was public and not many published specifically to students. No matter the case, neither their students nor the public visited these projects.

A phrase that described these single-experience users was one-hit wonder—meaning they visit once (or for a single purpose) and then we were left to wonder where they went and why they did not return.

**Cluster 2: Focused functionaries.** As expected, fewer users were grouped in Cluster 2 (approximately 25% of each sample, see Table 7 and Figure 14). The normalized means and box plot dimensions were nearly identical between the two samples for every feature (see Figures 18, 19, and 20). When they did vary it was surprising to see that the 5–90 sample had a slightly higher relative activity means on some features but their activity levels were not different enough to cause the mean to change dramatically. The 5–90 group also registered on the same day with slightly fewer people than the 10–90 group which may have contributed a different experience with the IA and have some influence on activity levels.
Examination of the heat map showed that users of Cluster 2 had relative medium levels of visits, folder organization, words per project and per resource, significant project changes, and resource visits (even with few published projects). This cluster did not distinguish itself with any relatively high levels on any feature.

These behavioral patterns reveal users who may have found the IA to be useful, but only in narrow ways (as evidenced by few projects and resources). They used what they gathered and their resources were visited at a relatively high rate. Perhaps they are good at finding the perfect resource for their intended IA use.

There have been anecdotal instances where teachers would have the students rotate between stations in the classroom, one of which was the computers where the project was already loaded and all they needed to do was read the text and click on the links but not reload the project. Hence, there were many resources viewed, but fewer project views.

Summarizing the user behaviors combined in Cluster 2: A medium number of words to resources but low overall word count and projects; a moderate amount of changes when they do change; they are medium in one important area to the IA: their resources get visited.

This cluster seemed to have had a focused purpose to their resource gathering, authoring, and use. But they did return and seem to have been competent in their use. For these reasons, the phrase to describe them was: focused functionaries.

Cluster 3: Popular producers. The third cluster was again a bit smaller than the Focused Functionaries of Cluster 2 (with approximately 13–15%, see Table 7 and Figure 14).
The feature means (raw and normalized) of both models for the users in this cluster largely paralleled those of the One Hit Wonders of Cluster 1, but at a relatively higher level (see Figures 18, 19, and 20, and note the parallel nature of Clusters 1 and 3). Exceptions to the similarities to Cluster 1 are in Registration and project visit features.

An interesting phenomenon of the Cluster 3 features is that the addition of the five to nine login users generally lowered the normalized mean, but the shape of the plot was retained for all but the number of same-day registrants (RegDayCnt) and the project use features. Notice that the addition of these five to nine login users also relatively increased the mean number of same-day registrants.

Other features were affected by the less-active group, only much less dramatically. As with the Focused Functionaries above, the normalized means of the 5–90 sample’s model were slightly different, but this time a little lower than those for the 10–90 group with the exception of registration behavior. The 5–90 group’s normalized number of same-day registrants (RegDayCnt) was nearly as large as the One Hit Wonders of the first cluster.

As mentioned in the means and medians discussion in the Project Usage section above, the larger sample’s feature values were weighty enough to dramatically influence the project usage feature means.

Project usage subcategory features of the number of project visits between user sessions (AvgInterSessPrjView), the number of resource visits between user sessions (AvgInterSessResView), and number of student logins (NumStuLogin), are probably the most interesting part of Cluster 3 user behaviors (especially when viewed next to
behaviors in Cluster 4, as the two are nearly opposite between the two analyses). This clusters’ 10–90 login sample had much higher means for the three aforementioned features but the 5–90 sample was lower. This would indicate that the five to nine login part of the 5–90 sample utilize their projects much less than the 10–90 part and that the 10–90 part is quite student oriented in addition to having many public project visits.

An additional study into the composition of the projects could reveal some of the design choices and subject matter that may be more engaging to the public or specifically designed for particular students or to the public in general. Whatever the case, it is clear that very similar general user behaviors are present in the five to nine and the 10–90 visit subsamples.

To summarize this cluster: When they come they tend to edit in small bits. They have small, but stable, projects that are both public- and student-published and attract a lot of visitors—especially students with the 10–90 login users.

A phrase that fit this cluster was: popular producers.

Cluster 4: Prolific producers. As the smallest cluster (approximately 3–4% of each sample, see Table 7 and Figure 14), the users of Cluster 4 were a view into a specialty type IA user. The raw cluster sizes were nearly identical between models as Table 8 indicated. Two of the eight users that were in Cluster 4 in the 5–90 sample’s model were in Cluster 3 in the 10–90 sample’s model, leaving six users still in Cluster 4. The small cluster size caused the means to be more volatile with the loss or addition of those two users, and accounted for the entire change in the feature means for Cluster 4—yet overall the relative plot showed mostly stable behavior patterns with exceptions noted below.
This cluster’s users had the largest normalized means over almost all features with the exception of the number of same-day registrants (RegDayCnt) and the number and percent of publicly published projects. Notable spikes occurred at the number of resources, words in projects, significant project changes (on the latter two, they were an entire order of magnitude larger than any other cluster in raw score, see Tables 9 and 10). Examination of the box plots of Figures 19 and 20 reveal the skewed distribution and the presence of several outliers that have affected the cluster mean when compared to the median and inter-quartile ranges.

Differences between the models for the two samples indicated the impact of the two users mentioned above. In IA visits features and number of PHP sessions in which a project edit occurred (NumPrjEditSess), the two users’ data pulled the means slightly upward in the 5–90 sample’s model. Whereas the smaller sample (10–90 logins) was strikingly lower in the number of project and resource views and with student logins, while the larger 5–90 sample had a much greater proportion. These project usage difference were the reverse of differences in Cluster 3, where the smaller sample had the higher normalized means. Interestingly, though, both models showed approximately the same normalized mean for the number of student resource views.

It was interesting to note the different publishing, project views, and student login results between the popular producers of Cluster 3 and Cluster 4 and between the two models. Some users’ efforts were very focused on their students and they did not publish much to the public—does this mean that those teachers were unwilling to share, or were
they unsure that their contributions would benefit a wider population? Such questions can be the focus of future research.

To sum up the behaviors of Cluster 4: They did not register on the same day as many other people. They performed every activity at a higher level than every other cluster with the exception of the number of projects they chose to share with the public. When more active users were considered (10–90 sample), they tended to not have as many student visits, but this does not reduce the number of resources the students view.

A phrase to describe this group is *prolific producers*. They produced large personal collections of resources, they produced a lot of words in each project, produced large changes in their projects, and produced large numbers of student visits.

Table 12 contains the summaries for all four clusters. Again, caution is advised as with any generalization, in that these groups are very general and that other behaviors are likely embedded in these clusters because the analysis sought to produce four clusters. However, these groups appear to be congruent with past research and anecdotal use of the IA and can be used as the basis of future research.

**Data, Methods, and Tool Report**

The process journal was reviewed and key information was extracted to answer to the third research question: “How were data, methods, and tools used, and what are possible implications for other online user behavior studies?” Much of this information was covered in minute detail in the KDD pattern mining results section above, this section
Table 12

*Cluster Characterizations*

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Identifying Phrase</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>One-Hit Wonders</td>
<td>They came in droves, created a few projects and did not publish many, but when they did it was to the public and not students. Neither students nor the public visit their projects. The users visited a few times and then we were left to wonder where they went and why they did not return.</td>
</tr>
<tr>
<td>2</td>
<td>Focused Functionaries</td>
<td>With medium numbers of words to resources but low overall word count and projects, a few moderately sized project changes, these users are only distinguished in one important area to the IA: their resources get visited. But they did return and seem to have been focused in their use.</td>
</tr>
<tr>
<td>3</td>
<td>Popular Producers</td>
<td>They have small, but stable, projects that are both public- and student-published and attract a lot of visitors—especially students with the 10–90 login users.</td>
</tr>
<tr>
<td>4</td>
<td>Prolific Producers</td>
<td>They did not register on the same day as many other people. They performed every activity at a higher level than every other cluster except in public projects. Those with more visits tended to not have as many student visits, but number of resources the students view was stable. They produced large personal collections of resources, they produced a lot of words in each project, produced large changes in their projects, and produced large numbers of student visits.</td>
</tr>
</tbody>
</table>
will condense this information in a single place with suggestions that may assist future WUM and EDM processes.

When reviewing this information it was important to keep in mind that the goals of each web application may differ from the traditional bottom-line of money. Indeed, the IA has a bottom line (or goal) of assisting teachers to create engaging learning environments. Whereas engaged learning is difficult to assess in an Internet-based environment, proxy indicators (such as activity features) inform behaviors that may lead to or indicate an engaged learner (e.g., project or resource views). Any increase in the bottom-line of an educational site may be investigated and empirically understood to at least some degree using purposeful proxy information.

The following sections condense the detailed description of KDD from above and describe the data, methods, and tool choices that can assist other sites who desire to investigate user behaviors with WUM.

**Data Choices**

When data collection began in earnest, little was known about the level of activity for the average IA user or even if the term *average* could be applied to any user. The decision was made to collect a very detailed level of usage and behavioral actions (e.g., clicks, resource saves, project differences, etc.) into three repositories: the IARD, web server logs, and Google Analytics.
In the end, only the IARD was utilized because it was largely a duplicate of the server logs and Google Analytics data, but with the addition of user identification data that made it possible to link to the project and resource data.

The data collection for WUM necessarily depends on the kind of knowledge desired from the data. For example, if general site statistics and page views are desired, the default information collected by server logs and Google Analytics should suffice. However, if knowledge of individual users (i.e., how much they changed some text each save) is required in conjunction with their page views, then the page view data must contain user identifiers.

Server log files are almost always collected by default by web servers and are very easy data to gather. While server logs contain page request information, they do not generally capture user-identifiable information by default (making connections between web-usage and user information impossible). There is an ability to inject user data into server logs (e.g., using Apache Module mod_log_config, see http://httpd.apache.org/docs/current/mod/mod_log_config.html, specifically the %{Foobar}C format string when used with a cookie containing session or user information). The server log might only be written before other page-generation code completes and results can be injected—such as when an error occurred while attempting some action—that may inform the goals of the site.

Furthermore, unless log settings are configured correctly more cleaning is necessary because they tend to record many more requests from nonuser entities such as search crawlers—automated scripts that follow links in webpages that return information
to search engines like Google and Bing. There are several log analysis applications available (e.g., AWStats, http://awstats.sourceforge.net; Webalizer, http://www.webalizer.org, and many others) that provide graphs and trends, but do not generally have the features of larger applications like Google Analytics and Omniture (http://www.omniture.com).

Google Analytics (http://www.google.com/analytics/) is tailored for general analytics and viewing trends and could be the basis of many studies. But because data sent to Google Analytics is stripped of all user-identifiable information, it is useless for a study of user behavior linking application data with web usage data. One can embed user-specific (e.g., userID) into the information sent to Google Analytics, but this could also make other Google Analytics statistics unstable unless done with care.

Both the web server logs and Google Analytics were easy to use and require little programming knowledge to implement. But they both were limited when it came to tracking user behavior in web usage and application activity.

An application-specific logging solution can address most or all of the shortcomings of web server logs and Google Analytics data collection methods. Crawlers can be filtered, comments about user activity can be captured, web usage and application usage can more easily be tied together, and so forth. However, application-specific data collection requires programming and can be expensive to implement.

Additionally, there is the temptation to gather much more data than will be used. For example, even most of the IARD data was thrown out simply because there were so few users who had utilized every possible IA feature. For example, it had been assumed
that user visits were long with sustained work within a session on one or more projects. However, the inter-session time lapse, duration, and depth of most visits were discovered during KDD to show that users worked in short visits, generally not doing much each time. For example, the median visit duration for all registered users was zero. Dropping all zero-length visits, then median rose to 1 min 14 sec. While the median for all users fed into the analysis ($N=366$) was 19 min 30 sec (rising to 19 min 46 sec when zero-length sessions were dropped) showing that there were many nonusers who had registered.

This nonuse caused a more-sparse-than-expected analysis space and required most features to be dropped. In retrospect, it would have been more fruitful for the IA had the general data been examined first to determine levels of activity and from there decide what detail was needed rather than attempting to generate the most minute and comprehensive dataset possible.

**Methodological Choices**

As with the overcollection of data, an overabundance of features were produced from the IARD and an unrealistic anticipation of longitudinal analysis was developed when, in reality, had a more conservative methodology been followed the same level of knowledge would have been produced (because so much data was eliminated), and much time would have been saved in the process.

Therefore, wisdom dictates that if you know little about the data, begin with only a little of the data. From analysis results each piece of information (e.g., features) can be examined to expose any underlying complexity that requires further investigation. In
short, gather so called low-hanging fruit first and then reach for the more complex and
difficult to attain after the basics are understood.

**Tool Choices**

For many design, business, and marketing purposes, Google Analytics covers the
needs of many websites. Google and other free and commercial analytics applications are
very good at this tuned behavior. Whenever possible, utilize these prebuilt, automated
services and their reporting mechanisms. Additionally, if there is a feature or report that
you would like to have that the tool does not include, it is very appropriate to contact the
developers and make requests that would help the product be more appealing to you.

However, there are times when a particular level of data is needed and that is the
time when custom programming must come into play. Having a custom tool is very
expensive in terms of time (if the researcher is the programmer) or money (if hiring
outside resources), but if the information is useful, large dividends can recuperate the
costs.

In the case of the IA, the budget was built around the use of no-cost to use
software: PostgreSQL for the database, VIM as the editor, PHP as a scripting language, R
as the statistical and plotting engine, and \LaTeX{} for document preparation. The IA is
currently hosted on RedHat Linux by Cornell University—an NSDL partner—for a
minimal fee. For analysis, the only software that was purchased was LatentGold 4.5 for
about $1000. Other clustering options include the free R packages (such as clust,
MCLUST, and poLCA) and, for about the same cost as LatentGold, MPlus
There are several courses available (online and face-to-face) for each of these applications.

These applications performed very well for the purposes of this study and are recommended for those on a stricter budget. If additional funds are available, commercial software equivalents are available for each layer (from operating system to analysis package).

**KDD Process Documentation**

While generalities about the data-gathering decision process were covered in the preceding section, the individual pieces of the knowledge from data/databases (KDD) framework are evaluated and generalized for those studying EDM. This section specifically addresses the research question: “What general procedures emerged that are applicable to the study of educational digital libraries?”

The motivation for this research purpose was to provide insight to the uninitiated who have aspirations of performing EDM on web-usage data. As the study proceeded, it became clear how important it was to document the KDD process in finer detail than existed in any of the literature reviewed and provide cautionary insights. For instance, a naïve expectation from this study illustrates the need for these insights: notwithstanding several one-line warnings about the time consuming nature of preprocessing the data in several of the literature reviewed in Chapter II, there was an overconfidence in the ability to accomplish this study in a short amount of time.
Clean and Integrate

As mentioned earlier, this was the most laborious and painful (in terms of lessons learned the hard way) part of the entire process. Lessons learned brought deep respect for the one-line warnings in the literature about the difficulty of preprocessing, but also a sense of frustration in that those previous works did not share their preprocessing caveats that might have made this study more efficient. The following lessons learned are presented in order to avoid a similar situation for others in the future.

Intimate knowledge of the data is prerequisite to effective work, but perhaps even more essential are the end goals, which must be defined beforehand or much time will be considered wasted when irrelevant metrics or results are discarded.

While data are accruing, it is important to check the data for integrity and consistency so that there will be fewer surprises when preprocessing. In fact, it would behoove the developer to plan and implement the data cleaning and integration as the data are planned for and being collected, then initial assumptions can quickly be checked as data are tested long before the final cleaning.

With the IA, the data collection was implemented several months before the cleaning scripts were created. With so much data collected first, situations arose that required more complex scripts to clean and generate features than would have been had the features and scripts been planned first—especially, if the features could have been generated as part of the actual user tracking scripts. With the IA, simplicity on the collection of data was offset by the complexity of the preprocessing scripts.
Another lesson learned is that much of what may initially be thought as important may not be in the end. For example, the finer-grained data on the source of collected resources turned out to be useless in the end. Had the data collection followed the wise saying to gather low-hanging fruit first, much time would have been saved in trying to cover all possible analyses in one collection effort.

Finally, lessons learned about preprocessing include proper utilization of database tools. Stored procedure and functions are able to quickly perform looping and other kinds of functionality not available in structured query language (SQL). Utilizing indexes is imperative for quick lookup of data; otherwise the database is forced to loop over all rows in a table every time a value is looked up. For the processing of text strings, it is important to have regular expression functionality available so that patterns can be found (e.g., substrings of URLs). Lastly, writing the scripts in external files ensures that the process is repeatable, documented, and questions about how features are constructed can be answered.

**Select and Transform**

When selecting the data to use in a study, it is important to have a framework of how that data informs the subject being studied—especially in an emergent study. In utilizing the meaningful IA activities framework, it was possible to ensure coverage of the most important usage features when making decisions of what features to discard or keep.

Additionally, when it was discovered that the finer-grained features were unusable, transforming them into larger aggregate parts was simple to do (one small advantage of
early fine-grained focus). On the other hand, had data collection started at a higher granularity, many of the features would have been inspected prior to proceeding with finer-grained collection and the most appropriate divisions or segmentation of the data would have occurred.

Transforming the data into a usable form can include standardization of features, log-transformations, or other statistical techniques to prepare data for analysis. In the case of the IA, LatentGold binned the features prior to analysis and no transformations were required. A different analysis choice for a different purpose may have required additional transformations. Just be prepared to interpret the analysis of transformed data.

Data Mining (LCA)

The actual data mining step was perhaps the most exciting part of the whole KDD process.

With such powerful tools so easily available it was important that the researcher keep detailed notes on which analyses were run and with what parameters or options in order to reproduce results and ensure coverage of possible model settings that might impact the final model.

With the current study, each sample was subjected to analysis of 1 to 20 clusters. Following that, various models were selected for tuning in order to find the appropriate number of clusters, the right mixture of features, and the correct direct-effects to include. A spreadsheet of these settings and their outcomes may be the most important tool a researcher can use to track all that activity.
As mentioned before, it was imperative to filter features through the framework in order to ensure proper coverage of concepts at the start of analyses. It was possible that all features related to a particular behavior would have been unimportant to the final results, but it was important to know this empirically.

Finally, it was important to check that all options in the model are understood and have been adjusted to proper values. For example, with LCA, the ability to include direct-effects in the models is one of the reasons the modern LCA is so powerful. With this study, the relaxation of one analysis constraint by allowing a direct effect was the difference between an unstable set of models and one final and consistent model.

**Evaluate and Present**

A fun, yet challenging, part of KDD was evaluating and presenting the results of the data mining in an understandable way for those not intimate to the fine details of the analysis. Evaluation of the results was best done while checking for consistency between the results and expert knowledge. In the case of the IA, expert knowledge of past research was important in understanding the clusters and that they did indeed describe user behavior patterns observed in workshop studies in the past.

As mentioned above in the last Selection and Transformation section above, the data was not transformed as inputs to the LCA modeling, but to make the results comprehensible, it was necessary to transform the results in order to summarize a complex analysis. The transformation of cluster means by a log plot, normalization, binning the normalized values, and applying color or shading to the bins were a series of helpful
transformations and presentation techniques that enabled a more understandable picture of
the results.

In all, the KDD process was a helpful analysis framework to guide the efforts in
this study. While the process was many times cyclic, it also helped keep track of where the
cycles should begin and end. Combining KDD with the IA meaningful activity framework
for grouping the features and looking at users gave a model for understanding the results
and portraying them in a meaningful and understandable way. The final characterizations
of the clusters was made much more easy by these techniques.
CHAPTER V

CONCLUSIONS

This chapter presents a summary of the research performed in this study for each research question, reviews implications for the developers of the Instructional Architect (IA; http://ia.usu.edu) and associated professional development workshops, summarizes the lessons learned that are generalizable to other studies of educational data mining (EDM, and more specifically those involving web usage mining, or WUM). Limitations to this study and future work are also covered.

Research Summary

Data mining, in general, has become possible through the use of modern computing technologies. A recently evolved branch of data mining is EDM that has become a field in its own right. However, most EDM applications have been investigating student behaviors and outcomes while very little has examined teacher use of technology.

At the same time and through the same means, digital libraries and subsequently educational digital libraries have become their own respective field and branch. The natural course was to use EDM with educational digital libraries—and it has been, but mostly in the construction and development of the library with web content and web structure mining. Little has been done in the way of WUM.

Educational digital libraries occur in all shapes and sizes. Some contain educational learning resources (i.e., resource repositories, like the National Library of Virtual Manipulatives at NLVM.usu.edu) along with their associated metadata and other
libraries contain only the metadata harvested from other libraries (i.e., metadata repositories, like the National Science Digital Library at http://NSDL.org).

Each kind of library is limited to observing user behavior in their own context. For example, a metadata repository can only directly log searching techniques and perhaps a click that indicates a visit to a resource. On the other hand, a resource repository that receives that click has little to no idea of the search that led the end user to the resource, but can log intimate details about the usage of the resource. Thus integrating end-user information of both metadata searching and resource usage has been difficult.

End-user authoring tools, like the IA, provide a view into both searching and instructional context in which teachers can use online learning resources while providing a traceable link for the resource repository wherein a more complete end-user profile may be investigated. These investigations have been accomplished mainly with traditional means. Now, with the application of EDM techniques such as latent class analysis (LCA) in the knowledge discovery from data/databases (KDD) framework, the profiling of users of the IA with detailed usage information has begun on a larger scale.

The current work was done in parallel with another researcher who investigated the IA, but had a different set of questions, data, and users (Xu, 2011). The similarities and differences between the two studies have been woven into the following presentation and are summarized in Table 13.

Both of these research studies were guided by two frameworks. The process framework was KDD, which outlined the following steps in working with collected data:

- cleaning and integration,
Table 13

Comparison with a Similar Study

<table>
<thead>
<tr>
<th>Study Characteristic</th>
<th>Palmer</th>
<th>Xu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Framework</td>
<td>KDD</td>
<td>KDD</td>
</tr>
<tr>
<td>Focus</td>
<td>Overall Process</td>
<td>Suitability of LCA</td>
</tr>
<tr>
<td>Duration</td>
<td>January 2008 - November 2009</td>
<td>February - December 2009</td>
</tr>
<tr>
<td>Number of Users (N)</td>
<td>270 then 146</td>
<td>757 then 348</td>
</tr>
<tr>
<td>Features</td>
<td>44 reduced to 23</td>
<td>19 reduced to 9</td>
</tr>
<tr>
<td>Demographic Features</td>
<td>IA Introduction source</td>
<td>Postal Code, Teaching Tenure, Subject</td>
</tr>
<tr>
<td>Number of Clusters (k)</td>
<td>4 in two models</td>
<td>7 then 3</td>
</tr>
</tbody>
</table>

- selection and transformation,
- pattern mining, and
- evaluation and presentation of results.

The first two steps are often grouped under the term preprocessing.

The framework that structured the choices of data analyzed and interpretation of results was a set of meaningful IA activities that teachers perform while using the IA: (a) user registration and subsequent visits, (b) collection of online resources, (c) annotation and sequencing of collected resources in a new online educational resource (i.e., project authoring), and (d) use of authored resources.

It is important to note that the common goal between this work and Xu (2011) was to characterize IA users through EDM. Xu’s work documented an analysis refinement
process in order to better understand the suitability of LCA (also known as latent cluster analysis) as a clustering tool from which characterizations may be drawn. This work was more focused on the preprocessing and characterization using LCA as one method of analysis in the larger KDD process as a case study from which educational digital libraries may gain insight and implications for conducting their own EDM experiments.

Additionally, overall differences between Xu (2011) and this work are that this work used the same usage features on two sets of users and employed several different presentation modes of the results in order to characterize user clusters. Xu, while investigating the suitability of LCA as a clustering approach, modified her usage features in order to refine the data between iterative LCA analyses with subsequent pruning and analysis of the resultant clusters.

**Research Questions—Revisited**

The specific goals of this study were as follows.

1. Mine for patterns emerging from meaningful IA user activity data (pattern mining).
2. Characterize user behavioral patterns in plain terms (characterize).
3. Report on how data, methods, and tools were utilized for this study (data, methods, and tools report).
4. Document the EMD process for the study of educational digital libraries (EDM process documentation).
Upon reflection of the entire process that was covered in Chapter IV, the conclusions for each research question are better presented in a different order. First, conclusions about the EDM process will be overviewed to set the stage for the presentation of the pattern mining and characterization conclusions that will be presented together. Lastly, conclusions on data, methods, and tools will be presented.

**EDM Process Documentation**

*What general procedures emerge that are applicable to the study of educational digital libraries?*

The entire KDD process was represented in investigating this questions and will again serve as the organizing framework for this summary.

**Cleaning and integration.** This was by far the most time consuming part of the research. In the literature reviewed, there was mention as to the immensity of this task, but very little guidance on particular problems. While the absence of these discussions are likely because every project’s data are relatively unique and will not likely have the same challenges to arriving at useable data and one solution will not fit all circumstances, it does not follow that examples from one case could not inform another.

Two pages of generic kinds of information to be gathered (International Educational Data Mining Society, n.d.-b) and gleaned (International Educational Data Mining Society, n.d.-a) were produced from discussions among many researchers. The description of how to convert from one to another, though, is largely missing from the EDM literature. These descriptions may well exist in the broader data mining literature,
but that body is largely inaccessible to educational researchers due to its highly technical
nature. This work sought to remedy a portion of this problem.

Several issues were discovered in what was originally thought to be clean data
after standard *noisy* data were removed (e.g., spiders, web crawlers, spammers, site staff
information).

First, due to the fact that data gathering became more comprehensive over time,
not all users of the IA were well represented in the final data set. This issue precluded
inclusion of more users when certain usage features were simply missing with no reliable
way to fill in the data. The lesson here is not to expect perfection from historical data. If it
is desired to examine particular usage behaviors, then be patient enough to get the right
data and practical enough to discard incomplete data.

Second, there were changes in daylight saving time legislation that caused
problems when trying to integrate data that were produced by two different programs.
These discrepancies were not difficult to remedy, but finding them was a surprise and was
time consuming. A lesson to be learned is to ensure the proper setup and maintenance of
servers. Know where the data is produced (e.g., a time stamp) and be sure these methods
are consistent throughout your data collection. If it is unrealistic to change server settings,
at least be aware that these kinds of problems are remedied in the cleaning.

Finally, another kind of dirty data resulted from the way PHP (the language in
which the IA was written, *PHP.net*) creates random session identifiers. Computers only
approximate true randomness and can be expected, over time, to produce the same random
sequence again, thus causing reuse of user session identifiers. This made some sessions
appear days, weeks, or months long when in reality there were several separate sessions that were closer to 30 or 60 minutes long. These long sessions can be somewhat empirically divided into plausible user sessions (i.e., disambiguated) based on user identifiers and by looking for long periods of inactivity.

Additionally, another kind of session reuse is when the same user may have multiple intentions (or episodes) within one PHP session. While the former kind of session reuse can be empirically disambiguated, this former kind is more qualitatively determined and will require the researcher to identify specific patterns that indicate each episode. The lesson to be learned is that expert knowledge of the system producing the data is required in order to determine plausible meaning to certain behaviors.

All of these instances of unexpected problems in the data led to the understanding that data are messy and one can nearly always expect some kind of time-consuming cleaning. However, with due diligence and forethought, many of these problems can be avoided, or at least known in advance.

Integration of the data was eased because it consisted mainly of linking tables together within the IARD using appropriate key information. For this study the user registration, clickstream logs, resource collection, and project authoring and visit histories were integrated into as complete picture of a user as possible in a data mart. Nearly any kind of question about IA usage could be examined by selection and transformation of this information.

The lesson here is to plan wherever possible the integration of data from the start. It is also good to do the integration with a small set of examples with real code and not
just a paper description of the process so that the process can be periodically checked for completeness and surprises dealt with along the way. Otherwise, the expectation of quickly cleaning and integrating data will be met with confusion, delay and frustration.

**Selection and transformation.** This study followed a bottom-up philosophy of fine granularity in data collection that would subsequently be weeded out as needed or justified by the analysis. The assumption was that more data could more fully describe user behaviors. In contrast, Xu followed a top-down approach by selecting only a few usage features assuming that a few specific and purposefully chosen features would speak to user behaviors. Each approach has its advantages and disadvantages.

There were well over 200 features produced in the data mart. Preliminary analyses on the data mart indicated that the data were very sparse. In fact, the IA usage features were so much more sparse than originally anticipated and forced a vast reduction in features used in the analysis. In the end, only 44 were applied to the analysis and only 23 contributed to the final models.

The lesson to be learned is to know the data so that the selection of features is based on experimentation and not just assumptions.

While it is generally good to keep as much information as possible, it is better to keep the right information. As it is common to grow into data collection for each project, identify the meaningful data that indicates the kind of usage that is of interest and collect that data. Periodic testing and exploration of the data as it is collecting will provide opportunity to tweak the collection and produce better data in the end—and even
eliminate some data collection that is determined to not mean what it was originally assumed to indicate.

The main transformation of data was in construction of features. For example, transforming the project history into a series of features including the total number of projects (NumPrj), the number of words in all projects after the removal of HTML code (NumPrjWords), the number of resources used in all user projects divided by the number of words in all projects (WordsPerRes), and the number of non-HTML code words from one project save subtracted from another project save (AvgNumWordDiff).

In general these kinds of transformation are what bring interpretation and meaning to raw clickstream data. It is important to examine all assumptions of clickstream meaning to see that the data supports particular interpretations. It is also important to express these assumptions as data are presented.

Another transformation that for this study was in the construction of user-aggregate data. This meant a lot of averages and counts for each user and it was important to correctly interpret the results. It is important to understand the limitations of such statistics (distributions and subjectivity to outliers) in order to interpret the results.

Lessons to be learned about selection and transformation are that each approach—top-down or bottom-up—has its limitations. Top-down approaches can answer specific questions, but perhaps miss nuances or associations that are simply not present in a very narrow dataset. Bottom-up can find unanticipated associations and insights, but at the same time can be very difficult to ensure there are enough data to analyze and interpret. The difference in N between Xu and this study are primarily because of the
difference in approach. Many more users had all the features she initially selected, while many did not have all of the features of this study.

**Pattern mining (LCA).** Once the data were ready for analysis, 44 features were fed into the LCA. Through the iterative processes of LCA described in Chapter III, only 23 features had enough impact on the models to keep them in the analysis. In the same iterative processes, the number of clusters was settled at \( k = 4 \) based upon the Bayesian information criterion (BIC) statistic.

During initial LCA analyses the LatentGold program crashed when all user data was included. Therefore, it was decided to segment the data by the number of logins or visits from the users. It was discovered that approximately 50% of registrants had never returned and dominated the first analyses. A further segmentation removed users with 0–5 return visits who were considered nonusers.

Two samples were analyzed: the 5–90 login group and its proper subset of users with 10–90 logins. The true difference between these samples was those who had five to nine logins—assumed to be less active users. Arriving at a stable model for each sample took time, patience, and purposeful record keeping as clusters and features were added or discarded. One very important part of the analysis was allowing for dependencies to exist within the features of each cluster for certain pairs of features—indeed, this proved to be pivotal to obtaining consistent and solid models.

Xu’s clustering results were originally in seven clusters, but a refinement of features reduced that to three clusters that were more understandable, but this was done at the cost of users included (\( N \) reduced from 757 to 348). Her subsequent frequent item-sets
analysis, Xu included additional demographic information from teachers to see how the classes were associated. She discovered that teaching experience and technology knowledge showed some impact on the usage patterns of registered IA users. The process of using the outcomes of one model in a subsequent analysis is an instance of what Baker and Yacef (2009) called the methodology of discovery through models.

The first lesson to be learned about the analysis is that it is one of the more enjoyable parts of the whole KDD process. Another lesson to be learned is that there are trade-offs in all analyses. This is where expert knowledge of algorithms and the data can combine in a powerful way in order to better understand users. A final lesson is that the purpose of the analysis should drive which analysis (or analyses) is utilized—as there are multiple analyses that can answer a single inquiry. Viewing data from multiple, complimentary directions is good for validating and exploring results.

**Evaluation and presentation of results.** Surprisingly, in the end, the models for both samples were largely identical with the exception of some large differences between project usage feature means of Clusters 3 and 4. These differences resulted from the combination of users being classified into different clusters (when common between the two samples) and the noncommon subset of five to nine users causing a wider distribution of the data in the 5–90 login sample.

The results were found to have discernible meaning when evaluated within the framework of meaningful IA activity. The cluster evaluation process lessons and conclusions are presented first with details about the clusters and their names presented in the Characterize section below.
The results of the analyses were presented in several tabular and graphical layouts. Transformations of the data through log functions and normalization processes further assisted in understanding the data. A very clear way to present the data was with box and whisker plots since they display the shape of each underlying distribution for each cluster. Finally, a sort of heat map was produced that further aided in discerning usage patterns in each cluster that led to being able to characterize each cluster in plain terms.

With the means for each feature displayed in table format (see Tables 9 and 10), it was difficult to interpret the raw means with so many values. Because visual analyses assist in detecting patterns, means were then plotted in raw, log-transformed, and normalized panel plots (see Figures 15 through 18). Each successive plot was produced to find a better way in which to view the results.

Box plots also assisted in evaluating and interpreting the results because they provided insight into the distribution of each cluster’s feature values (see Figures 19 and 20 for within sample comparisons and Figures C1 through C7 in Appendix C). Box plots were of great assistance when trying to understand the few real differences between the models for each sample.

Finally, the normalized means plotted in Figure 18 were smoothed by binning them into relative High, Medium, and Low values with respect to the other clusters in each model. These binned values were then presented as a table with shading to provide a sort of heat map view of the clusters in Table 11.

The most important lesson learned during this process was that exploration of data can occur on many levels and with many different displays of the data. Each different
display can provide more or less interpretive power. For this study the plotting began simply and then progressed in sophistication until differences and similarities in the data were plain to see. If the purpose had been prediction and not simply characterization of the clusters, the results would have been more quantitatively reviewed and examined; but since the interpretation was purposed to be clear and meaningful to the nonexpert, visual analysis techniques were preferred.

Another lesson that was learned (and is covered in the Pattern Mining and Characterization sections below) is that researcher assumptions are at risk of being proven wrong when asking the data questions. One surprising result was that the cluster means were nearly identical in pattern from one sample to the other when it had been assumed that users with more visits (i.e., 10–90 logins) would have very different patterns than would exist with those with fewer visits (i.e., 5–90 logins).

A final lesson here is the proper use of your data manipulation, analytical, and presentation tools. It is essential to have good database management principles in order to speed up any database operations. Anywhere that scripts can be written to automate the cleaning and integration, it should be done to ensure repeatable results.

**EDM process wrap-up.** The most important piece of the process is to have the goals and frameworks of the study beforehand—adjustments can always be made later, but have a reasonable end goal to begin with. Purposefully preselect data as much as possible to reduce any wasted effort producing too many features.

As data are accruing, test and transform them as early as possible to avoid unnecessary difficulty while cleaning and integrating data sources. In fact, having these
process predetermined is as essential to a speedy study as having reasonable and well-thought out goals.

Particularly for the IA, technical improvements could include the cleaning and integration of data on a daily basis. For this study, data accrued for the entire time of the study (22 months) and then analyzed. For future studies, it would be more efficient to modify the existing scripts and set them to run on a daily basis so that data are available at any moment for analysis. An automated analysis could inform some kind of intelligent tutor that would help teachers know what else they can do with the tool and integration into their classroom teaching.

**Pattern Mining**

*What patterns of user behavior emerge in terms of key features and metrics?*

*Specifically, what patterns arise within meaningful IA activity?*

This and the next research question on characterization of user patterns are simply two different ways of looking at the clustering results. This section will view the clusters within the contextual framework of the meaningful IA activities and, consequently, there are fewer lessons learned since this information is more specific to the IA.

One of the assumptions held by the researcher was that more visits would produce more intense usage of the IA. In other words, it was expected that the clusters that emerged from the 5–90 sample group would be very different from those of the 10–90 sample; and simply put, “How could someone visit so often and yet do so little?” Surprisingly, that assumption was shown to be false with only a couple of exceptions.
Of all the displays of data, Figure 18 (the normalized plot) and Table 11 (the heat map) were the most helpful to see how each cluster in each model was similar to or different from the others. The box plots in Figures 19 and 20 were very helpful in seeing how the distributions differed and overlapped as well.

Overall trends observed indicate that subgroups of features tended to follow each other—this is not surprising because many of them are very similar. For example, the Registration and Usage features had two subgroups that appeared to function independently: Registration day counts (RegDayCnt) and three different ways of counting visits (logins [NumVisits], PHP Sessions [NumUserSess], and days visited [NumDaysVisited]). It appears that when users register on the same day with many other users their other features are generally lower. This plays out in Cluster 3 where the 5–90 sample has an increased registration day count and all the other features were slightly lower and in project view information, dramatically lower.

Of the three ways of counting visits, they all acted in parallel with each other and only slightly higher normalized means in each cluster. This indicates that users only visit once on each day and with only one session on that day. Therefore, the visit counting methods could probably be reduced in future studies because they acted as one. However, if at a future date the IA professional development workshop expects multiple visits to the site daily, then it would be prudent to retain these three features so that changing relationships can be detected.

Using the above-mentioned displays of the results also demonstrates the parallel nature of the normalized means and distribution of features across clusters. An example of
this is with the Resource Collection features (resource count [NumResources] and the number of folders a user created [NumFolders]) were quite parallel between models but appeared to reverse their relative numbers as cluster number increased. Relative to other clusters, Cluster 1 had fewer resources than folders, Clusters 2 and 3 had about the same, and Cluster 4 had a lot more resources than folders. However, the feature that varied was the number of resources while folder counts remained nearly consistent across all but Cluster 4. So the amount of resource collection building activity varies with cluster number.

Another example of these similarities existed with the project authoring activity features’ relative cluster means, which had three different shapes. Clusters 1 and 3 were quite parallel in relative feature means with rises in project edit sessions and number of projects, Cluster 3 being generally higher. Cluster 2 had almost an opposite shape with higher relative means in words per project and the amount of change per session. Cluster 4 was simply the greatest relative means in every aspect of authoring activity with notable spikes in overall number of words and the greatest changes in each session.

Another interesting discovery was how certain features seemed to vary together across meaningful usage groups. An interesting example is with project authoring features where relative means were higher when the number of same-day registrants were lower and vice versa (Clusters 2 and 4 vs. Clusters 1 and 3).

Finally, trends across models and groups were very visible with the Project Usage features where Clusters 3 and 4 differed between models on two project view features and on the number of student logins—Cluster 3 was higher with the 10–90 login group and
Cluster 4 with the 5–90 login group. Cluster 2 had the lowest overall project usage, Clusters 1 and 3 published relatively more to the public than others, and Cluster 4 published many more to student relative to the other clusters. With differences like this, the Box Plots were very helpful for understanding these similarities and differences.

Upon examination of the box plots, it was discovered that the 8 users in Cluster 4 of the 10–90 login sample were in both samples but that 2 of them moved clusters between models (into cluster 3 of the 5–90 login sample’s model). All of the change in Cluster 4 between models can be accounted for by the loss of these two users. With Cluster 3, 7 of the 12 users in the 5–90 sample were in other clusters in the 10–90 login sample’s model and one user moved to Cluster 1. So, while user movements between models have somewhat to do with the changing of cluster assignment, but also was heavily influenced by the addition of new users to the sample.

In the end, it was instructive to have an interpretive framework within which to view the results in their constituent parts. Viewing the range, scope, and interplay of the features within and across meaningful activities highlighted the differences between samples and models. Couching of the results into this meaningful IA usage was not used by Xu, but her features came from these groups of activities.

Characterize

How can these patterns be described and characterized in ways that are germane to the purposes of the IA?
The characterization of the LCA clusters is really part of the presentation of results in KDD. However, because this was its own research purpose it was treated separately. A brief description of the four clusters found in this study are followed with contrasts with the work of Xu and finally implications for the IA and other educational digital libraries and their services based on these.

The four clusters were examined and given names based on their relative usage behaviors of the IA: one-hit wonders who came, did, and left and we are left to wonder where they went; focused functionaries who appeared to produce some content, but in only a small numbers and they did not share their projects very much; popular producers who produced small but very public projects that received a lot of visitors; and prolific producers who were very verbose, created many projects, and published a lot to their students with many hits, they just did not publish for the public much.

**Contrasts with other work.** The same behavior trends were found to exist in other IA research and it was concluded that LCA was successful in empirically finding observed user behaviors in the web-usage data. A reconciliation between the final cluster characterization of Xu (2011) and this work is in Table 14. Seeing so much agreement in the final clusters from both studies gives credence that these user types truly exist.

However, because the data are mostly continuous, the divisions are more of an arbitrary level of activity that distinguishes the behavioral group (a concept discussed in the limitations below). Notwithstanding this limitation, the combinations of relative levels of activity that produce the clusters are instructive and have implications for the IA. Perhaps these user clusters are even farther generalizable in the sense that most sites will
Table 14

Comparison of Cluster Description and Sizes with a Similar Study

<table>
<thead>
<tr>
<th>Palmer</th>
<th>Xu</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>One-Hit Wonders</strong></td>
<td>Ineffective</td>
<td>Neither group really was active in any particular activity in the IA.</td>
</tr>
<tr>
<td>(≈ 58%)</td>
<td>Islanders (36%)</td>
<td></td>
</tr>
<tr>
<td><strong>Focused Functionaries</strong></td>
<td>Insular Classroom Teachers (33%)</td>
<td>While these groups produced, they did not share with the public very often and they saw only a small amount of project visits.</td>
</tr>
<tr>
<td>(≈ 25%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Popular Producers</strong></td>
<td>Key Brokers (31%)</td>
<td>These are the most active users in all senses of the word. Both of Palmer’s groups are listed here because the N-size is so small on Prolific Producers that they could have been included in with the other group. However, because they were not, we can see some usage differences between two kinds of stellar users: some publish smaller but more projects to the public than students and vice versa.</td>
</tr>
<tr>
<td>(≈ 14%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>and</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Prolific Producers</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(≈ 3%)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

have different levels of participation, some users choosing to use one feature or another more than others—all with different views about how the site can best be used.

As Xu noted that even her Ineffective Islanders and Insular Classroom Practitioners found something that could assist them with some kind of instructional need—even if just once. Therefore the IA appears to have fulfilled at least one objective (Xu, 2011).

**Implications for the IA.** The preceding newfound knowledge of IA users has several possible implications for the IA developers and those who plan and carry out instruction to use the IA (i.e., online courses and professional development workshops). Just as Maull et al. (2010) was attempting to find plausible groups that could be further
studied and targeted with differentiated treatments, this work along with that of Xu hope to inform future IA work.

The professional development series was significantly modified to a more problem-based approach shortly after these data were gathered and therefore, the results may not reflect the current participation model. Notwithstanding that limitation, the results of this study can inform the workshop developers of the various user behaviors from which they can follow-up on and give help to teachers to further integrate online learning resources into their curriculum.

Additionally, the workshop participants could receive differentiated help, follow-up emails, and contacts about supplemental workshops or other opportunities based on cluster membership. Additional contacts may help overcome some of the reasons teachers have given for not using the IA (e.g., lack of time to learn or understand the system, technological knowledge [too much or too little], etc.).

A final knowledge nugget that is not new, but now confirmed, is that over half of all IA registrants are considered nonusers. In light of the different clusters and examination of those groups, another fruitful field of inquiry for the IA team would be to contact these nonusers and see what their experience was and if or how they are using online learning resources in their teaching.

**Implications for educational digital libraries.** Finally, in the current study, there was no resource search data used in the integrated NSDL search. Therefore, little has been studied about user searching and selection of resources. Research along these
lines would provide insight into how learning resources are found and subsequently used in one location.

Additional studies into the IA can help both resource designers or repositories in their quest to provide engaging and enlightening learning objects. For example, patterns of prose used in IA projects surrounding particular resources from different classes of users would perhaps inform resource designers how their resources are being used.

Another interesting study would also look at how these resources are being introduced and followed up by the teacher. For example, if a teacher has to produce a lot of explanation surrounding a resource, then perhaps there could be changes to the resource for future release—whether it is difficult to navigate, caveats to watch for, incorrect information, not as easy to use as another resource, and so forth. Perhaps they would learn that different teaching styles like to use the resources differently and would be able to accommodate different usage contexts. In such a position, the IA could be an informal and candid feedback system for resource developers and metadata repositories alike.

**Data, Methods, and Tools Report**

*How were data, methods, and tools used, and what are possible implications for other online user behavior studies?*

In the case of the IA, most available data was duplicated across three sources (the application relational database, web server logs, and Google Analytics). In the end, only the application relational database (IARD) was employed since it included the relevant data of the other two in addition to providing necessary user-identifiable information for
linking page views and click stream data with IA user information (such as projects, registration data, and resource gathering). Log files can be configured to contain user or session information and Google Analytics can be made to embed user information, but neither solution was as convenient as using the IARD.

The methodology of collecting very detailed information was helpful when the time came for user-level aggregation; on the other hand, much time was required for producing that data. As a lesson learned, it may often be the case where larger-grained data should be explored first and then finer grained data collected and examined based on the overall analyses (i.e., a *top-down* approach). In other words, look at resource use overall before delving into or expecting resources from every library to be used.

Tools used in this study largely consisted of Open Source and free-use software (PHP, PostgreSQL, VIM, R, and \LaTeX). The sole piece of purchased software was LatentGold which was utilized for performing the actual LCA. These tools showed that with little software costs, educational sites can perform user behavior studies utilizing WUM techniques.

Several methods were employed in viewing and analyzing the data. Standard tabular output of the cluster means (Tables 9 and 10) helped to demonstrate the scale of each feature. However, because there were so many features it was difficult to get a picture of all of the relationships with numbers alone.

Several different visual displays of the data assisted in understanding the distributions of so many features in an intuitive way. The input data sets were examined with histograms (Figures 9 through 11) as well as box plots (Figure 12). Cluster means
were more helpful during characterization of user behavior when visually displayed in raw, log-scaled, normalized, and color-coded bins (see Figures 15 through 18 and Table 11). Moreover, box plots were again very helpful in understanding the variation between clusters as well as between models (Figures 19, 20, and C1 through C7 in Appendix C).

There are many other tools for gathering data, processing and analyzing data, and displaying the results of analyses from which researchers may choose. Each has benefits and drawbacks. In each case it is almost always best to determine what resources are currently available and use those first before delving into new processes and tools. However, over time new tools will naturally be explored as questions and directions of data mining change within a particular site.

**Limitations**

This study was an initial foray into EDM and as such could not be as sophisticated and a later study will eventually become. Several limitations contributed to the small scope and some uncertainty about the results.

First, the assumption that the number of visits to the IA would differentiate user behavior appears to be flawed and perhaps another feature would be better for exploring each different feature for better prediction of use.

Second, there were not enough regular users to completely rely upon the results of this study. Had all 5,000 registered IA users been able to be used in the analyses, it is likely that more clusters would have been necessary and additional patterns would have
emerged. Similarly, with the changing workshop model, it is possible that the
terpretation needs to be updated to reflect the current participation model.

Third, additional demographic information about the users would have provided
additional covariates that could be used in predicting a likely class from the outset.
Several variables have been added to the data collection and combined with very similar
data in other studies (e.g., zip code and technological comfort level, see Xu, 2011).
Combining the approaches of each study, perhaps a better IA user model can be achieved.

Fourth, the results are quite specific to the IA because of the functionality
constrained set of features utilized in analysis. However, even though that is the case, the
kinds of users observed in the IA can likely be found in many educational sites.

Finally, at this point it is important to realize that while the variables were
discretized by binning continuous data into groups before being analyzed. While this
action allows for analysis it also introduces possibly artificial divisions in the data thus
introducing a topic of concern. Are the features such that discretization is preexistent or
inherent in the data or is it an artificial construct? While it is sometimes difficult to know
the answer to this question, when interpreting the results it is important to remember that
clusters based on false categories would, in turn, be false. For this reason, it is vitally
important that any analysis either be verified or taken with the proverbial grain of salt.

In this study, most of the features are likely more continuous and therefore the
results should be carefully scrutinized. However, these results are of utility as one
approach to better understand users and while not yet verified, may serve as an
enlightening view into some aspects of user behavior that can be of some use to future research.

**Future Work**

Several opportunities for future work have been identified above, in each case it would be built upon the knowledge and experience gained from this study. One opportunity for future work is a longitudinal analysis of users to see if or how they move between clusters over their IA lifetime. Understanding these lifetime can help educational sites envision how users move between different behavioral patterns as they are introduced to and learn about a site, become comfortable with it, begin using it with their students, teach others how to use it, and continue into or pass through the online community of users. Such an analysis would provide the basis for some sort of program that would adapt the user experience based on the perceived cluster membership and user trajectory into or out of the community.

Additionally, with ever-accruing data, study of the particulars of each major feature would provide insight to just how users found the resources they used in their projects—whether through the NSDL, Web, or other IA projects—among other things. Studies into how teachers introduced and directed their students to use the resources would help designers understand the mental models of teachers and their understanding of the content as well as their teaching styles. Providing search suggestions could assist users to become more familiar with under-used online resource sources. For example, if a user
only saved resources from the NSDL, then an email notice could be sent to them introducing them to the process for adding a resource from the wider Web.

Another interesting study would look closer at the demographic information, especially the IA Introduction source (IAInfoGrp and IAInfoBin). Understanding where a user heard of the IA would also inform educational outreach efforts so that the most users could hear of the tool and educational digital library offerings with good expectations about adoption of the tool.

Yet another subsequent investigation could delve into search techniques and viewed vs. saved resources. Coupling data mining with interviews and observations can give rich meaning to sequences of behaviors that can, in turn, inform the way search engine optimization and display of results.

Alternatively, the results from this study’s work with clickstream and project content analysis could be enhanced by utilizing Maul’s *bag of clicks* and *bag of words* constructs for identifying textual features (Mauill et al., 2010). However, since in that study each teacher’s work was based on a common curriculum, that may have had some advantages when looking at text. The IA has no a priori context from which the projects are built and therefore the contexts can be vastly different.

Further research into the reasons users chose to share their projects publicly, to only their students, both, or none at all, would help in understanding the dispositions of teachers and their willingness to share and borrow each other’s work. In conjunction with this study, a look at the content of each project would assist in understanding the common
threads between project content and project popularity. Employing textual data mining
techniques could find some common threads of themes in resources and projects alike.

Indeed, there are many opportunities for future research with the IA and the results
of this cluster analysis and EDM effort. In the end, this work can serve as a warning about
the difficulties and caveats of EDM and a starting point for future studies.
REFERENCES


Magidson, J., & Vermunt, J. K. (2004). Comparing latent class factor analysis with the traditional approach in datamining. In H. Bozdogan (Ed.), *Statistical data mining and knowledge*


Appendix A

Glossary

b

black-box analysis  Referring to clustering methods where only cluster membership is known, these analyses do not allow for examination of the reason for membership assignment.

bounce  A phenomenon where the entrance page is the only page visited in that visit (a single-page visit).

bounce rate  The number of single-page visits divided by the total number of page-visits.

browser configuration  These are variables collected by some data sources indicating the state of the client browser. Such information such as the enabled state of JavaScript or cookies, screen resolution, browser name and version, operating system, etc.

C

cleaning  Refers to a preprocessing step in the KDD process where missing or malformed data are corrected.

clickstream  A series of clicked links that a user activates while navigating a site.

client-side script  A short program that is run in an interpreter within the user’s browser (e.g., JavaScript, Flash)

cookie  A file of information stored on the client machine by the server. Cookies enable much greater tracking capabilities by overcoming NAT issues as well as visit, unique visitor, etc. by enabling the tracking of time since last activity.

d

data mart  A store of data that has been subjected to preprocessing and can be used in various analyses. Generally considered a subset of the information that is stored in a data warehouse.

data mining  The act of analyzing large stores of data for patterns which were hitherto unknown. Often described as automated or semi-automated analysis. This is the third step in KDD.
**data warehouse** The permanent storage location of preprocessed data. Warehouses can take many forms depending on the amount of data stored, in other words, in the case of a smaller operation, the data mart and data warehouse might be the same thing; whereas a larger operation will create data marts out of the warehouse that are for specific kinds of analyses.

**educational data mining** Utilizing data mining in the field of Education—this can be as broad as educational digital library service analysis, or as focused as a single user’s behavior in a learning module.

**entrance page** The first page a user sees on a website—this is not always the “front door” or home page of the site.

**evaluation** The KDD step after analyses and before interpretation, where resultant models are scrutinized for coherence, meaningfulness, and utility.

**exit (page)** The last page requested of a particular website during a session or visit. Due to constraints of client-side scripting, tracking the length of time on the exit page, as well as the nature of the exit (e.g., through a link, closed the browser) is generally unknown.

**hit** The request of a web page from a web server. In early web metric use, hit was a popular indicator of site popularity, but because of several limitations it has fallen from importance. For instance, each image, style-sheet, javascript, or flash object requested for a single web page will also create an entry in the web-server’s log. In addition, single visitors reloading the page will see several hits, but this is from a single visitor.

**integration** The combining of data from different sources into a central repository (e.g., data warehouse, data cube) with appropriate links between data records and summarization.

**interpretation** The KDD step after evaluation and before presentation, where new knowledge (i.e., rules, statistical results, etc) are necessarily explicated for understanding in the context of the research questions.
javascript  A programming language used by site developers to manipulate a web page while it is in a user’s browser. Javascript has been used for such things as HTML form validation, dynamic manipulation of HTML elements (DHTML), asynchronous communication with the web server that includes only essential information (AJAX). Certain vulnerabilities exist from the exploitation of scripts in browsers and, therefore, some users and administrators choose to turn off some or all scripting, preferring safety over convenience and effect.

knowledge discovery from data/databases (KDD)  Often described in four steps: (a) cleaning and integrating, (b) selecting and transforming, (c) data mining, and finally, (d) evaluating and presenting. The term preprocessing has been applied to the first two steps.

landing page  See entrance page.

leverage  Difference between how often an association appears and to the product of the supports of the individual association elements (Webb, 2003, p. 32).

lift  A ratio of how often an association occurs to the support of the consequent—or outcome side of an association (Webb, 2003, p. 31–32).

link tracking  A mode of tracking where each clicked link is recorded. This can be done in client-side scripts or on the server with each request.

manifest variable  In latent variable (or class) analysis, these are the observable variables which are analyzed for patterns which then are grouped to form latent classes.

“meaningful” activity  User activity on a website that is in-line with site goals. In the case of the IA, known meaningful activities are IA project creation and use indicators with regard to the following meaningful activities: (a) user registration and visit patterns, (b) collection of online resources, (c) annotation and sequencing of a new online educational resource, and (d) use of the new resources.
**metadata**  In short, Data about data. In digital libraries, where the resources are considered data, metadata is often the “library card” for a particular resource but can also be annotations, comments, links to similar or opposing content, etc. Several standards exist for library metadata, e.g., Dublin Core (dublincore.org), METS (www.loc.gov/standards/mets/), OAI-PMH (www.openarchives.org/OAI/openarchivesprotocol.html), and many others.

---

**network address translation (NAT)**  The effect which a gateway may have upon the visibility of an IP address from one side of the gateway to the other—in other words, many computers with individual IP addresses on one network (e.g., in a school or lab) may appear to all have the same IP address to the outside network (i.e., the Internet). Therefore, several requests from the same IP could have been multiple computers from behind the same gateway.

**noise (noisy data)**  Data with spikes, inconsistent collection, errors in collection, mismatched fields (e.g., date formats), and other anomalies in data that create a difficulty in analysis. These have several sources and remedies.

**non-normal distribution**  Any non-Gaussian distribution.

**normal distribution**  A Gaussian distribution.

---

**page view**  The modern example of a hit, where the page of interest is counted as being loaded only once, no matter how many images or other requests come as a result of this view.

**preprocessing**  The first two steps as outlined in KDD. The preprocessing of data can be one of the most time consuming and troublesome parts of knowledge discovery (EDM considerations Romero & Ventura, 2007)

**presentation**  The final step in KDD, right after interpretation, where results and visualizations are prepared for communicating and sharing with others.

**project-hits-since-last-visit - variable**  The number of times an IA project was visited since the last meaningful activity session of its owner.

**proxy variable**  A manifest variable that stands in the place of another variable. These can be used to approximate a multiple type time series with varying time by
defining other representations of the time series so that observations can be taken and used in analysis.

**purchase path**  The clickstream from the “shopping cart” to the “Thank you for your purchase” page. In education, this could be the registration process.

**purchase history**  The record of past purchases. In education a transcript would be analogous.

**referrer**  The page upon which a link was clicked which led to the current page.

**request**  The http header information sent from a browser to a web server which is interpreted and the desired page or an error is returned.

**selection**  Part of the second *KDD* step in which variables of interest are selected from a data warehouse or other storage collection.

**session** (*-time, -path, etc.*)  A session is similar to a visit where the session closes after a certain period of inactivity (around 30 minutes for most websites). The session-time or -path (and others) are manifest variables associated with a single session or visit. It is non-trivial to determine when a session truly ends because of the generally undetectable end of a visit.

**supervised machine learning**  Algorithms for pattern discovery which utilize training data with instances of outcome variables with which input variables are associated (see *unsupervised machine learning*).

**timeout**  Used to tell a cookie when to expire, or for a page to refresh itself or forward on to another page.

**time-since-last-visit - variable**  The difference from the current IA visit to the last known visit of a particular user or IP address.

**traffic source**  The referrer to the landing page from an outside source (e.g., search engine, link from another site, direct—where the URL is typed directly into the browser or accessed via a bookmark) as opposed to an internal site source.
**transaction funnel clickstream** The sequence of clicks that takes a user from the beginning of a desired path to the end—such as from a “proceed to checkout” link that would eventually lead to the final “thank you for your purchase” page. In education, a transaction funnel could be a sequence of instructional content.

**transformation** Adjusting or condensing information into a more manageable size or format in the second step of KDD. Reduction to summary data may be desirable (frequency count for different variable combinations will be important for LC analysis).

**U**

**unique visit(or)** In Google Analytics, as detected by a cookie with a midnight timeout. If the cookie exists, then the visitor is not unique, otherwise the visitor is unique, and the cookie is set. However, within a lab setting, the number of unique visitors will likely be under-reported.

**unsupervised machine learning** Algorithms for pattern discovery which utilize manifest variables and covariates to detect groups or clusters of patterns in the input data (see supervised machine learning).

**user profile** Information collected from a user registration or preference information, or constructed from their browsing or purchasing patterns.

**V**

**visit** Generally this is similar to a session, where a single person (or computer) requests additional information from a website within a certain period of time. Business industry standard sets the timeout on visit to about 30 minutes of inactivity.

**visitor** A user or computer which enacts a visit.

**W**

**web structure mining** Data mining with respect to the association of pages, sites, and the links contained therein.

**web content mining** Data mining for text or semantic properties of pages, sites, and the links contained.

**web usage mining** Data mining server logs and other sources of actual user behavior on pages and sites.
**words-added-this-session - variable**  The number of words added to all IA projects edited during a visit or session.
Appendix B

Data Sources and Variables

The following tables display the various data that were available for this study. First, in Table B1, the general type transformations (Han & Kamber, 2006) used with the raw data are displayed. Most of the information in this appendix came from the three raw variables contained in Table B1, because of this, similar variables exist between the different views and constructs.

Table 4 in the proceeding prose displays the final features that are an aggregate of the information in this Appendix. That cross-sectional view of the user and were the features fed into the latent class analyses and the Used column indicates whether the feature was in the final models.

Tables B2 shows the session-aggregated variables produced that were then aggregated into Tables B3–B5 from which the features in Table 4 were selected. For future studies on session-level behaviors, Tables B2 contains a data set that could be used.

Table B3 contains the activities and associated variables of resource finding, reuse, and visiting. Table B4 has user-aggregated project information. Table B5 focuses on the project creation, editing, and visiting activities on a project edit-session basis.

Note that in the three feature space tables (Tables B3–B5) the notation including square brackets and pipes is a shorthand for several variables that are collapsed into one table entry. For example, in Table B2 under resource gathering, # [Total|NSDL|IA|Web] resources saved, there were actually four variables created; one each for Total, NSDL, IA, and Web resources saved during that session.

Table B1

Initial Data Transformations from the IA Relational Database Tracking Entries

<table>
<thead>
<tr>
<th>Raw</th>
<th>Transformed</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timestamps</td>
<td>Session length / time between sessions</td>
<td>Smoothing, Generalization, Feature Construction</td>
</tr>
<tr>
<td>URL</td>
<td>Meaningful activities</td>
<td>Generalization, Feature Construction</td>
</tr>
<tr>
<td>Comments</td>
<td>Meaningful activities</td>
<td>Feature Construction</td>
</tr>
</tbody>
</table>
**Table B2**

*Session-Level User Activity Feature Space*

<table>
<thead>
<tr>
<th>Category</th>
<th>Features</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Visit Statistics</strong></td>
<td><strong>Time since last visit</strong></td>
<td>Interval</td>
<td>First time of this visit minus last time on last visit</td>
</tr>
<tr>
<td></td>
<td><strong>Time to next visit</strong></td>
<td>Interval</td>
<td>First time of next visit minus last time of this visit</td>
</tr>
<tr>
<td></td>
<td><strong>Duration</strong></td>
<td>Interval</td>
<td>Last time of this visit minus first time of this visit</td>
</tr>
<tr>
<td></td>
<td><strong>Depth of visit</strong></td>
<td>Count</td>
<td># of pages visited</td>
</tr>
<tr>
<td></td>
<td>**% visit as project [creation</td>
<td>view]**</td>
<td>Continuous</td>
</tr>
<tr>
<td></td>
<td>**% visit as resource [gathering</td>
<td>view]**</td>
<td>Continuous</td>
</tr>
<tr>
<td></td>
<td><strong>% visit as profile activity</strong></td>
<td>Continuous</td>
<td># profile/registration pages divided by depth of visit</td>
</tr>
<tr>
<td></td>
<td><strong>Bounce</strong></td>
<td>Boolean</td>
<td>Was there no meaningful activity this visit? (1 = no/bounce, 0 = yes)</td>
</tr>
<tr>
<td></td>
<td><strong>Was student too</strong></td>
<td>Boolean</td>
<td>Logged in as user’s student login simultaneously</td>
</tr>
<tr>
<td><strong>Resource Gathering</strong></td>
<td>**# [NSDL</td>
<td>IA Browse] searches**</td>
<td>Count</td>
</tr>
<tr>
<td></td>
<td>**# [NSDL</td>
<td>IA</td>
<td>Web] resources saved**</td>
</tr>
<tr>
<td></td>
<td>**% [NSDL</td>
<td>IA] saved**</td>
<td>Continuous</td>
</tr>
<tr>
<td></td>
<td>**[NSDL</td>
<td>IA] resource save ratio**</td>
<td>Continuous</td>
</tr>
<tr>
<td></td>
<td><strong># folders used</strong></td>
<td>Count</td>
<td># folders receiving saved resources this visit</td>
</tr>
<tr>
<td></td>
<td>**# [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>**# [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>**# DNS domains [visited</td>
<td>saved]**</td>
<td>Count</td>
</tr>
</tbody>
</table>

*(table continues)*
<table>
<thead>
<tr>
<th>Category</th>
<th>Features</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td># new projects</td>
<td>Count</td>
<td># of projects created this session</td>
</tr>
<tr>
<td>Authoring</td>
<td># projects copied</td>
<td>Count</td>
<td># projects copied from other projects</td>
</tr>
<tr>
<td></td>
<td>% browsed projects copied</td>
<td>Continuous</td>
<td># public projects copied this visit divided by all projects visited this visit</td>
</tr>
<tr>
<td></td>
<td># projects edited</td>
<td>Count</td>
<td># of user projects modified</td>
</tr>
<tr>
<td></td>
<td>Net word change(^a)</td>
<td>Count</td>
<td>With edited projects, # words at the end of the session minus the # of words at the beginning of the session</td>
</tr>
<tr>
<td>Change</td>
<td>Change similartext</td>
<td>Continuous</td>
<td>similartext index(^b) of old vs. new project information</td>
</tr>
<tr>
<td>similartext</td>
<td>Average change similartext</td>
<td>Continuous</td>
<td>similartext index(^b) of start vs. end project information (averaged by # edited projects)</td>
</tr>
<tr>
<td></td>
<td># new [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td># [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>Net [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>Word change to resource change ratio</td>
<td>Continuous</td>
<td>Net word change(^a) divided by net resource change</td>
</tr>
<tr>
<td></td>
<td>Overall text to resource ratio</td>
<td>Continuous</td>
<td>For all projects, # words(^a) divided by # used resources</td>
</tr>
<tr>
<td></td>
<td># words deleted</td>
<td>Continuous</td>
<td>Word count of all projects divided by # used resources</td>
</tr>
<tr>
<td></td>
<td>% of newly-used resources are [NSDL</td>
<td>IA</td>
<td>Web]</td>
</tr>
<tr>
<td></td>
<td>Average edits per project</td>
<td>Continuous</td>
<td># project saves divided by # undeleted projects</td>
</tr>
<tr>
<td></td>
<td>Average visits between edits</td>
<td>Continuous</td>
<td># project visits between edits divided by # undeleted projects</td>
</tr>
</tbody>
</table>

(table continues)
<table>
<thead>
<tr>
<th>Category</th>
<th>Features</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Visits</td>
<td># [owner</td>
<td>public</td>
<td>student] project visits since last session</td>
</tr>
<tr>
<td></td>
<td># [owner</td>
<td>public</td>
<td>student] visits to [NSDL</td>
</tr>
<tr>
<td></td>
<td>Average # of student visits to student-viewable projects</td>
<td>Continuous</td>
<td># visits to NSDL resources divided by # used NSDL resources</td>
</tr>
<tr>
<td></td>
<td>Average # of user visits to user-only projects</td>
<td>Continuous</td>
<td># visits to IA resources divided by # used IA resources</td>
</tr>
<tr>
<td></td>
<td>Average visits for used Web resources</td>
<td>Continuous</td>
<td># visits to Web resources divided by # used Web resources</td>
</tr>
<tr>
<td></td>
<td>Average visits for all resources</td>
<td>Continuous</td>
<td># resource visits divided by # all gathered resources</td>
</tr>
<tr>
<td></td>
<td># [own</td>
<td>public</td>
<td>student] projects browsed</td>
</tr>
<tr>
<td></td>
<td># [NSDL</td>
<td>IA</td>
<td>Web] resources visited from [own</td>
</tr>
</tbody>
</table>

*a* after removal of all non-printing html.

*b* calculated on html-stripped text with PHP’s similartext string function.
Table B3

*Resource-Related User-Aggregated Activity Feature Space*

<table>
<thead>
<tr>
<th>Category</th>
<th>Features</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Collection</td>
<td># [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td>Resource Reuse</td>
<td>% of used resources are [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>% of [NSDL</td>
<td>IA</td>
<td>Web] resources used</td>
</tr>
<tr>
<td></td>
<td>Average Text to Resource[^a^]</td>
<td>Continuous</td>
<td># of words in projects divided by the # of resources used</td>
</tr>
<tr>
<td>Resource Visits</td>
<td>Average visits of used [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
</tbody>
</table>

[^a^]after removal of all non-printing html.
Table B4

*Project-Related User-Aggregated Activity Feature Space*

<table>
<thead>
<tr>
<th>Category</th>
<th>Features</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project</strong></td>
<td>% projects copied</td>
<td>Continuous</td>
<td># projects copied from other users divided by # undeleted projects</td>
</tr>
<tr>
<td>Authoring</td>
<td># projects created</td>
<td>Count</td>
<td># of undeleted projects created by this user</td>
</tr>
<tr>
<td></td>
<td>Total length in words(^a)</td>
<td>Count</td>
<td># words of all projects</td>
</tr>
<tr>
<td></td>
<td>Average length in words(^a)</td>
<td>Continuous</td>
<td># words of all projects divided by the # of undeleted projects</td>
</tr>
<tr>
<td></td>
<td># of used resources from</td>
<td>Continuous</td>
<td># gathered [Total</td>
</tr>
<tr>
<td></td>
<td>[Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>Average length(^a) to resource ratio</td>
<td>Continuous</td>
<td># words of all projects divided by the # of used resources</td>
</tr>
<tr>
<td></td>
<td># project edits</td>
<td>Count</td>
<td># project saves</td>
</tr>
<tr>
<td></td>
<td>Average edits per project</td>
<td>Continuous</td>
<td># project saves divided by # undeleted projects</td>
</tr>
<tr>
<td></td>
<td>% projects with HTML</td>
<td>Continuous</td>
<td># projects with HTML divided by # undeleted projects</td>
</tr>
<tr>
<td><strong>Project</strong></td>
<td># [own</td>
<td>public</td>
<td>student] projects(^b)</td>
</tr>
<tr>
<td>Sharing</td>
<td>% project that are [own</td>
<td>public</td>
<td>student](^c)</td>
</tr>
<tr>
<td><strong>Project</strong></td>
<td># visits to [Total</td>
<td>own</td>
<td>public</td>
</tr>
<tr>
<td>Visits</td>
<td>Average # visits to [Total</td>
<td>own</td>
<td>public</td>
</tr>
<tr>
<td></td>
<td>Average inter-session visits to</td>
<td>Continuous</td>
<td># of between-session [public</td>
</tr>
<tr>
<td></td>
<td>[public</td>
<td>student</td>
<td>owner] projects</td>
</tr>
</tbody>
</table>

\(^a\)after removal of all non-printing html.

\(^b\)these counts may sum greater than the number of project the user owns because student and public are non-exclusive with each other.

\(^c\)these percents may sum greater 100% because student and public are non-exclusive with each other.
Table B5

*Project Edit-Session Activity Feature Space*

<table>
<thead>
<tr>
<th>Category</th>
<th>Features</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Authoring</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td># modifications</td>
<td>Count</td>
<td># of times a user saved the project on steps 2 and 4 of authoring</td>
</tr>
<tr>
<td></td>
<td># words</td>
<td>Count</td>
<td>Current # of words at the end of the edit session</td>
</tr>
<tr>
<td></td>
<td># [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>Words to resources ratio</td>
<td>Continuous</td>
<td># words divided by # resources</td>
</tr>
<tr>
<td></td>
<td>[public</td>
<td>student</td>
<td>owner/private] view Html</td>
</tr>
<tr>
<td></td>
<td>% project length is html</td>
<td>Continuous</td>
<td>One minus total # words after stripping html then divided by # words after stripping html</td>
</tr>
<tr>
<td></td>
<td>Net word change&lt;sup&gt;a&lt;/sup&gt;</td>
<td>Count</td>
<td># words at end of the session minus # words at beginning of the session</td>
</tr>
<tr>
<td></td>
<td>Change similartext</td>
<td>Continuous</td>
<td>similartext index&lt;sup&gt;b&lt;/sup&gt; of old vs. new project information</td>
</tr>
<tr>
<td></td>
<td># newly used [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td># [Total</td>
<td>NSDL</td>
<td>IA</td>
</tr>
<tr>
<td></td>
<td>Word change to resource change ratio</td>
<td>Continuous</td>
<td>Net word&lt;sup&gt;a&lt;/sup&gt; change divided by net resource change</td>
</tr>
<tr>
<td></td>
<td># [total</td>
<td>public</td>
<td>student</td>
</tr>
<tr>
<td></td>
<td>% visits are [public</td>
<td>student</td>
<td>owner]</td>
</tr>
<tr>
<td></td>
<td>% [NSDL</td>
<td>IA</td>
<td>Web] resources visited</td>
</tr>
</tbody>
</table>

<sup>a</sup>after removal of all non-printing html.

<sup>b</sup>calculated on html-stripped text with PHP’s similartext string function.
Appendix C

Detailed Feature Box Plots

This appendix contains detailed box plots for the 5–9 login and 10–90 login model features that are in the text as Figures 19 and 20. Each feature is plotted side-by-side with the same feature from the other model. This view gives opportunity for scrutiny of the way the features clustered across the two models.

Be aware that the scale on each pair of features may change. The scale is based upon the meaning of the feature. If “cnt” or “num” are in the title, then they are simply counts, “avg” for averages, “per” for normalized on the second variable name, and so forth.

To emphasize the utility of this level of detail, the following figures are noted. Figure C1 note how RegDayCnt shows nearly identical patterns between the two models with the exception of Cluster 3, the popular producers, where the 5–9 login model has a greater variation. But on the variable that was selected for segmentation, NumVisits, the two graphs are nearly identical—a very surprising finding. Perhaps NumVisits is necessary but not sufficient segmentation power.

Figure C6 displays the two publish states of interest, NumStudentPrj and NumPublicPrj, with the percentage of public projects PctPublicPrj. Notice how the median helps us understand the large mean for student publishing in Cluster 4 in both models—the means were subject to some extreme values, but the median held its place. Cluster 3 with the number of public projects also displays an common trend as the sample is smaller with the 10–90 login group, the box plots get much taller, indicating less influence of a mass around the mean or median. Finally, the percent of public projects has some dramatic changes in Cluster 2 and somewhat Cluster 3 between the two models. The changes here emphasize the effect of sample size just visited and also the reluctance of Cluster 2’s focused functionaries to publish publicly.

Figure C7 with the number of student logins (NumStuLogin) displays the effect of cluster movement between the two models. Heretofore the examples have called out situations where the differences were contained within the differences in sampling. With this example, the differences between models with Clusters 3 and 4 were caused by the movement of two users from Cluster 4’s prolific producers and five from Cluster 2’s focused functionaries in the 5–90 login model to the popular producers of Cluster 3 in the 10–90 login model as depicted in Table 8 (a net gain of 6 users in Cluster 3 since one also moved from Cluster 3 to Cluster 1’s one-hit wonders).

Indeed, box plots were a useful way to compare and contrast the clusters as they provided a measure of central tendency and also notions of the underlying distribution.
Figure C1. Box & whisker plots of the registration and usage features. A side-by-side depiction of features for Clusters 1–4 for both models.
Figure C2. Box & whisker plots of the resource collection features. A side-by-side depiction of features for Clusters 1–4 for both models.
Figure C3. Box & whisker plots of the project size features. A side-by-side depiction of features for Clusters 1–4 for both models.
Figure C4. Box & whisker plots of the project and resource usage features. A side-by-side depiction of features for Clusters 1–4 for both models.
Figure C5. Box & whisker plots of the project change features. A side-by-side depiction of some of the project creation features for Clusters 1–4 for both models.
Figure C6. Box & whisker plots of the project publish state features. A side-by-side depiction of some project creation and edit features for Clusters 1–4 for both models.
Figure C7. Box & whisker plots of the project usage features. A side-by-side depiction of some project creation and edit features for Clusters 1–4 for both models.
CURRICULUM VITÆ

Bart C. Palmer

1159 E. 850 S.
Spanish Fork, Utah 84660
801.362.3287
bart.palmer@gmail.com

Professional Vitæ

Employment

Research and Evaluation Coordinator, LDS Missionary Training Center  (Aug 2012–Present)
Provo, UT
Design and carry out research and evaluation projects related to missionary training and second language training.
• Write proposals for research and evaluation projects.
• Observe and interview subjects and conduct focus groups both at the MTC and in the field.
• Gather and process qualitative data.
• Perform statistical analyses using SAS.
• Supervise a part-time testing coordinator, language proficiency raters, and other part-time employees as needed.
• Prepare evaluation instruments.
• Write reports, prepare data displays, and present reports and otherwise share information.
• Develop SAS, Flex, and Java Server Pages tools for data gathering, processing, and reporting.
• Design, development, and use of Oracle database for storing and processing data.

Adjunct Computer Science Faculty, Stevens-Henager College  (Aug 2009–Sep 2010)
Boise, ID
Developed and taught courses in a blended environment: Programming Fundamentals and Concepts (C++); Programming Fundamentals (Java); Advanced Programming (C#), Web Programming I, II, and III; Computer Servicing I; and Introduction to Operating Systems.
• Earned consistently high student evaluations.
• Mentored new instructors.
• Constructed and delivered both blended and online courses via Angel LMS (http://www.angellearning.com/)
Institutional Researcher, Treasure Valley Community College (Jan 2009–June 2009) Ontario, OR

Support and inform institutional effectiveness and reporting through meaningful data collection and research.

- Defined and standardized valid assessment of Title III grant projects.
- Identified, collected, analyzed, and reported data meaningful to multiple college stakeholders (internal and external).
- Developed student tracking process data acquisition and analysis, greatly expanding institutional knowledge of student educational activities.
- Began method of capturing and reporting on student participation in various campus and professional activity.
- Assisted with preparation of federal and state reports. Developed an institutional data-mart for consistent reporting.
- Experience with MS SQL Server.


A small educational consulting DBA.

- Instructional design and development.
- Face-to-face and online professional development and facilitation.
- Research services.
- Programming.


Research assistant and programmer (http://IA.usu.edu, http://EDM.usu.edu, and http://DLConnect.usu.edu), funded by the National Science Foundation and the National Science Digital Library.

- Researched primarily web metrics and educational data mining applied to digital libraries and online learning resource use by teachers and students with mixed methods.
- Researched secondarily effective teacher professional development, pedagogical changes with technology adoption, communities of practice, and user-centered design.
- Lead development of a website for teachers to easily (re-)use online learning resources.
- Facilitated face-to-face professional development workshop.
- Programed in PHP 4.x & 5.x with PostgreSQL database.
- Utilized 3rd party interfaces (for integrated search and single sign-on design).
**Instructional Designer**, Brigham Young University—Hawaii *(May 2004–Dec 2004)*  
La‘ie, HI  
Designed and developed instructional products for the service area that includes Eastern Asia, and the Pacific Rim and Basin.  
- Experienced all phases of ADDE across several projects.  
- Designed and developed a distance piano course, an interactive scenario-based leadership course, an entrepreneurship course.  
- Participated in the development of the new instructional technology organization.  
- Developed with Flash and ActionScript and gained video and audio editing experience.

**Web Project Lead**, Brigham Young University—HRC *(Jun 2000–May 2003)*  
Provo, UT  
Programmer converting TICCIT functionality (computer-based learning environment developed in 1970s) from DOS to Windows, then to a web application (http://CLIPSone.byu.edu).  
- Programmed in C++, PHP, SQL, JavaScript, DHTML in Windows and Linux.  
- Developed foreign language DVD media controls in web application for language learning.  
- Developed and maintained mySQL databases.

**Education**

**Ph.D. Instructional Technology and Learning Sciences**, Utah State University *(Jan 2005–May 2012)*  
Logan, UT  
Instructional Theory Research and Development  
- Research Focus: Digital library service visitor study through educational data mining of web usage indicators.  
- Research interests: Educational digital libraries, teaching as design, planned and enacted online learning resources and other curricular material; effective teacher professional development; Learning Communities; perceptions of quality.  
- Informal mentoring and tutoring fellow Ph.D. students.

**M.S. Instructional Technology**, Utah State University *(Sep 2003–May 2005)*  
Logan, UT  
Instructional Design and Development  
- Internship at BYU-Hawaii’s Center for Instruction Technology Department—instructional design with focus areas in Pacific Basin/Rim and Asia.  
- Project leader for three large projects centered on How to Shoot Good Video, Communities of Learning with the Opera by Children program, and ski instructor training.
B.S. Computer Science, Brigham Young University (Sep 1996–Apr 2003) Provo, UT

Learned fundamental computational science principles.

- Knowledge of networking protocols, algorithms, data structures, abstract data types, basic electronics, amplifiers, digital-design, Intel assembly language, Java, C/C++ on Windows, Linux, and UNIX.
- Other favorite classes include philosophy, logic, physics, and technical theater as a lighting technition.

Additional Training

Introduction to Latent Class Modeling (Jul 2008–Aug 2008)

- With Dr. Jay Magidson of Statistical Innovations, Inc.
- Experience with LatentGOLD 4.5

Basic and Advanced SAS Programming (Sep 2010–Dec 2010)

- With Dr. Dale Scott in the Statistics Department of BYU.
- Experience with basic and advanced SAS programming practices with standard SAS procedures and macro programming.

Professional and Academic Service

College Ambassador, Utah State University—College of Education and Human Services (Sep 2005–2007) Logan, UT

One of nine representative for the College of Education and Human Services (http://CEHS.usu.edu) assisting with recruiting new and transfer students, tours, familiarity with all departments in college, go-to help for the dean.

- President of ambassadors for the 2006-2007 school year.
- Recruiting efforts in Idaho and Utah, booth presentations, representation of students at Advisory Council meetings.

Conference Paper Reviewer

- The Cognition and Exploratory Learning in Digital Age (CELDA 2004).

Papers and Publications


**Conference Presentations (coauthored)**


**Honors and Recognition**

2012 *Department of Instructional Technology and Learning Sciences* (May 2012)

*Outstanding Ph.D. Graduate Award*
- Selected for this award by the faculty on the Awards Committee.

*Collaborative Research: Educational Data Mining Approaches for Digital Libraries (NSF Grant #0840745)* (Dec 2008–Dec 2011)
- This funded grant proposal was based upon my dissertation proposal.
- For more information, see http://edm.usu.edu.

*Nominated for the Legacy of Utah State Award* (2008)
- Nominated by the faculty of the Department of Instructional Technology.

**Personal Vitae**

**Hobbies and Interests**
- Time with my beautiful wife and seven children. We homeschool and love to explore this beautiful world in which we live and getting to know the beautiful people both near and far.
- Technology—Automating processes and building tools that help people be more efficient and effective.
- Camping, Fishing (spinner, fly, and spear), Skiing, and Mountain Biking.
- Natural healthcare and nutrition, cooking, and travel.
- Amateur Radio—KC7TMM.
- Constitutional Studies.
- Learning.