University of Alberta

HTML5, Tablet Computers and the Emerging Web Paradigm

by

Amy Jeannine Dyrbye

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Arts

Humanities Computing

Fall 2013
Edmonton, Alberta

Abstract

In recent years, consumers have embraced a range of portable web-ready devices, and with them, new means of accessing and using the web. Yet, despite consumer adoption of smartphones and tablet computers for personal and work-related activities both within and without the academy, academic projects' websites still use specifications like XHTML and CSS best suited for a desktop or laptop computer user. This thesis examines the design choices behind academic project websites using XHTML and CSS, and, through a framework of Thomas Kuhn's concept of the paradigm shift and a hands-on project, demonstrates how a shift toward the W3C's newest specifications, HTML5 and CSS3, enables projects to expand their websites to include devices like tablet computer.

Preface

This thesis includes detailed discussion of a hands-on web design project centered on a fictional academic project. The website is public and readers are encouraged to supplement their reading with it.

The XHTML and CSS version is available at

> http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home.php

This version is intended for viewing on a desktop or laptop computer.

The HTML5 and CSS3 version is available at

> http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home_v2.php

This version is intended for viewing on a desktop or laptop computer, and is expanded to include Apple's iPad tablet computer in both landscape and portrait orientations.

Acknowledgements

This thesis benefitted from the advice and feedback of my advisor, Dr. Harvey Quamen. You encouraged me to take on a challenging project and helped me through the worst snags. This thesis would not be as tight or read as smoothly without your input.

My thanks to Dr. Patricia Demers, who took a raw new Masters student and entrusted her with the upkeep of the venerable Canadian Women Writing and Reading from 1950 project. Your mentorship and the challenges CanWWR encouraged me to confront were instrumental to my choice of thesis.

Additional thanks to Dr. Geoffrey Rockwell, who has, in bringing me on to the TAPoR and JWDTD? projects, pushed me to expand my skills and knowledge base further still. I am also thankful for the encouragement and advice you have given me throughout the thesis process.

Last but far from least, I am grateful to Matthew Dixon for his support, affection, encouragement and listening ear throughout this degree. You helped me maintain perspective and kept me going throughout.

Table of Contents

List of Figures

Chapter 1 - Introduction

Web access is a defining feature of twenty-first century North American life. It is integral to how many people find information, hear news, conduct their work and keep in contact. It is never more than a hand's reach away from those who desire it, and wireless web access is now common in public spaces like coffee shops, airports and libraries, supplementing home wireless routers and universities' campus-wide networks. For those out of reach of wireless, cellular networks support web traffic to customers' phones.

Over the last decade, technology companies like Apple have taken advances to computer technology such as high-density hard drives to move small, light, powerful and portable computers out of science fiction and into consumers' hands. Companies market their laptops on their thinness and lightness as much as computing power. The same innovations make it possible for many people to carry smartphones, themselves conveniently pocket-sized computers in miniature. In the last three years, Apple's aggressive marketing of their tablet computers, the first devices of their class on the market, has convinced consumers to adopt them as a practical, affordable bridge between the convenience of smartphones and the basic functions traditionally confined to laptop computers. Between Apple's iPad and comparable tablets from companies like Samsung and Amazon, many consumers now regularly using this class of device to facilitate their daily lives, be they at home, within businesses or within academia, and their applications owe as much to users' innovations as marketing.

People have been finding new and interesting ways to use the internet since the first person thought to use networked computers with Teletype connections to message other users in the 1970s (Hafner and Lyon 187-188). As they became comfortable with the networks, users developed and spread innovations that facilitated common tasks -- electronic mail, document markup, hypertext and so on -- some of which proved so useful that later developers standardized and refined them. The computers that supported the networks and innovations underwent their own refinements, which developers also took full advantage of for further innovations.

As access to computers opened up to lay users outside universities and research and development companies, both within organizations and as individual consumers, consumer's choices had greater influence on the directions computer technology took. For example, Apple's Macintosh computer was more attractive to Humanities scholars than its contemporaries because it was the first to implement a graphical user interface (Hockey, "History"). The technology, the practices of the users and the programming users developed built on one another. In the present, the net result of these influences is a large number of users demanding ready web access, and accessing web resources with a variety of portable web-ready devices, each suited to a particular range of tasks.

Though for most web users the concepts and practices underlying their restaurant search or Twitter feed remain obscure, without them there would be no table reservations or quick means to pass on venue observations to dozens of friends and followers. At the same time, users' desire to book reservations without making a phone call or to make a pithy observation to a broad audience spurred the development of the applications facilitating them. The allure of the smartphone is the way it facilitates these activities without requiring its user to be seated at a computer, while the appeal of the smartphone app is the way it broadens the utility of the smartphone beyond email, text messaging and basic web search. Common reasons cited for purchasing a smartphone include social networking functions, facilitating video or music consumption, as a gaming platform, and for the apps available (IBTimes).

The ways in which the Internet and the portable devices used to access it are influencing daily life extend into work activities. Within academia, websites have become a prominent means of presenting academic work, serving multiple purposes for the projects behind them. At its most basic, a project's website is a place for the project to present its findings. For some, it is also a means for contributors to add to its dataset, and to give other academics a means to view or interact with the project's dataset. In short, they represent the project, act as a workspace and provide a means of disseminating findings.

Academic projects' website developers can no longer safely assume their users sit strictly at a desktop or laptop computer, as "mobile browsing is now expected to outpace desktop-based access within three to five years" (Marcotte). Though the basics behind a web page -- a server to store it, an arrangement of page content, and a device with the means to interpret it -- have remained the same, the devices used to access them have multiplied. Devices are smaller, smaller devices more powerful, and users' desire to take full advantage of their portability established. However, the devices only interpret a website's content. It is the methods developers use to provide content to devices, whether desktop or tablet, that affect how those devices display websites. While the devices are the most visible sign of change to the web experience, developers work to expand what markup can accomplish is quietly enabling the website functions users rely on to accomplish their tasks.

The problem of presenting content on the internet is as old as the CERN computers where web markup originated. Tim Berners-Lee created HTML in 1989, a simplified derivation of SGML,[1] as a method of defining and formatting documents on the CERN network (Raggett et al.). By the late 1990s, HTML was in its fourth iteration and the general public had enthusiastically embraced the internet, electronic mail and web pages as a medium of communication and expression -- indeed, Elizabeth Castro proclaimed "[t]he World Wide Web is the Gutenberg press of our time. Just about anyone can create their own website and then present it to the Internet public" (Castro 11). Though HTML 4 and its younger cousin XHTML solidified many of the characteristics of modern web pages, they introduced a host of problems to web markup.[2]

[1] Standard Generalized Markup Language.
[2] Please refer to Figure 1.

HTML 4	XHTML	Both
• Lingering deprecated elements (ex: `marquee`, `blink`) (Henick 286) • Lingering deprecated attributes (ex: `align`, `style`) (Henick 287) • Browsers default to being overly permissive of broken code (Pilgrim 10) • XForms not compatible with HTML forms (Pilgrim 13)	• Draconian error handling requires a loophole to resolve (Pilgrim 10) • Not backwards compatible with HTML (Pilgrim 13) • Lack of support for properly served XHTML (Henick 271)	• Developer must choose a degree of error-handling rigidity (ex: strict, transitional) (Henick12) • Prone to <div>-itis (Henick 288)[3]

Figure 1: Problems with HTML 4 and XHTML

Two major groups worked actively to resolve these issues from 2004. That year, participants of a W3C workshop laid out seven guiding principals for the successor of HTML 4 (Pilgrim 11-12), while the Web Hypertext Applications Technology (WHAT) Working Group began work on extending HTML 4's forms and their backwards compatibility (Pilgrim 12). By 2006, the WHAT Working Group had developed a variety of new features for HTML, including expanded controls for forms and native multimedia support (Pilgrim 13). That year, the W3C's HTML Working Group agreed to collaborate with the WHAT Working Group and dubbed their collective work 'HTML5' (Pilgrim 13-14). In 2009, the W3C officially shut down the XHTML 2 Working Group, abandoning the XHTML specification in favour of exclusive focus on HTML (Pilgrim 14). As the product of the joint work of the WHAT and W3C HTML Working Groups,

[3] An overuse of <div> containers when implementing a site's CSS layout.

HTML5 not only contains resolutions to many issues with HTML 4 and XHTML, but extends the specification into features for developers working with portable devices like tablets and smartphones.

The changes to web markup, from Tim Berners-Lee creating HTML from the model of SGML, to the problems inherent to HTML 4 and XHTML prompting frustrated web developers to collaborate on resolutions that laid the foundation for the W3C's HTML5 specification, can be understood through a theory first developed to explain changes to the practice of science within a discipline. Where practitioners, such as web developers, develop and adopt a revised set of practices that resolve the problems at hand, the process of moving from one set of practices to another can be understood via the concept of the paradigm shift. First introduced by physicist Thomas Kuhn in his 1962 book *The Structure of Scientific Revolutions* and originally used to describe the ways in which scientific understanding and practice change over time, Kuhn understood the paradigm shift as a notion extensible to other fields, suggesting its application to philosophy, psychology, linguistics or art history (Kuhn 121). Here, it is useful for describing the changes to technology, the structures underlying the web environment, and to the social practices that drive how users employ those technologies to access the web.

Despite the implications of web browser updates, devices' product refresh cycles, or terms like 'backwards compatibility', existing web content still requires developers' active efforts to keep pace with developments to web specifications. Prior to the W3C's official rollout of HTML5 in 2010 (Gilbertson, "Drink Up HTML5"), developers experimented with new methods of adapting web pages. This included modifying a site's header to automatically detect the user's device, as described in papers such as Guadalupe Ortiz and Alfonso García De Prado's “Improving device-aware Web services and their mobile clients through an aspect- oriented, model-driven approach” or Chimay Kulkarni and Scott R. Klemmer's article "Automatically Adapting Web Pages to Heterogenous Devices". Others chose to focus on a set of techniques for dynamically adjusting a site's characteristics via its CSS styling collectively referred to as 'responsive design', for which Ethan Marcotte has become the chief advocate following his influential article "Responsive Web Design."

Each of the purposes behind an academic website has a set of needs attributed to it. Here, the focus is less on the precise mechanisms underlying a computer-based project, and more on maintaining its relevance to the researchers and its users. Nicholas Gold's "Service-Oriented Software in the Humanities: A Software Engineering Perspective" addresses the problem of maintaining digitally housed projects, suggesting focusing on the services the project provides, such as acting as an interface to the project's research. Daniel V. Pitti's contribution to *A Companion to Digital Humanities*, "Designing Sustainable Projects and Publications," emphasizes the close relationship between a project's research, the technology used to express it for others, and the need for a project's developers to maintain their familiarity with the technology available. With websites positioned as popular means for projects to create a common workspace for team members while also storing their contributions to its research, and to publicize its findings between journal publications and conferences, maintaining a project includes maintaining its markup.

Where audience use of the web environment or the technology associated with it has changed, developers need to innovate practices or adopt others' innovations to accommodate. An academic project aiming to maintain its utility in the face of developments to devices, web standards and their users' choices must make active observation of such developments a priority. Kuhn's paradigm shift is a tool for conceptualizing this type of change, considering how it happens and understanding its implications. Any one component may not herald a paradigm shift, but several taken as a whole may form a pattern deserving of attention.

HTML5 is one such component, which on examination reveals a larger pattern at work. It is not just the most recent variation on the theme of web markup, it reconciles the problems with HTML 4 and XHTML, and removes the need for third-party solutions to adapting a site to new devices. It is a facilitator developed to smooth the use of web content for users of portable computers such as smartphones and tablets. Users within and without the academy are increasingly integrating portable computing into their work activities. These compact devices differ in attributes such as display, interface and computing power from desktop and laptop computers, requiring new approaches to a website's content to display appropriately. While projects can choose to implement a device targeting technique like Ortiz and De Prado's or responsive design with their existing HTML 4 or XHTML markup rather than update their markup to HTML5, this is a spot fix to a larger problem. Even beyond the release of HTML5, the W3C ceased development of XHTML web markup in 2009 (Pilgrim 14). In addition, the CSS specification has expanded to include a module referred to as CSS3, including a device detection capacity of its own designed to work with HTML5 (Henick 96). Taken together, HTML5 and CSS3 offer more to projects' web development than an independent solution can. Even on its own, HTML5 integrates features that can facilitate Gold and Pitti's recommendations for digital academic projects.

I am concerned with how academic projects' websites can actively participate in the current web environment. Kuhn's conceptual framework illuminates the impact of new technologies and web practices on academic projects' websites. Each of these developments, whether updates to markup specifications or the devices users and project members are relying on to access project material, affect how academic projects can present their research on the web. With respect to HTML5, the W3C's choice to develop it rather than continuing to focus on XHTML, combined its promotion by companies like Apple and Google for devices running their operating systems, poises this version of markup to be a major part of the web development practices underlying the internet paradigm in the coming years.

The rollout of HTML5 and CSS3, with their built-in capacity to directly assist adapting sites for multiple devices, makes this the time to adopt these specifications for existing websites. It is no longer necessary for projects' web developers to seek solutions further afield to accommodate users' portable devices. The combination of technological advancement in consumer devices with the updates to web languages represented by HTML5 and CSS3, though only components of a larger web environment, parallels Thomas Kuhn's paradigm shift. Consider the convergence of the increasing variety of portable computing devices now available, users' applications of the technology, such as choosing tablets over laptops to access web content, and the rollout of HTML5 and CSS3 as the most suitable versions of their respective standards to offering web content to portable devices. Together, they form a definition of computer usage held together by mobile web access, and form a set of considerations for web design that demands new strategies to reach the same audience. This environment is ripe for developers to innovate approaches and lay the ground for a new paradigm, one markedly different from the one that characterized the late 20th century with stationary computers that must be physically connected to a network.

Indeed, this is hardly the first time the Humanities in particular have experienced a shift in practices, nor one precipitated by technology. The work of Father Roberto Busa, creator of the ambitious *Index Thomisticus* concordance project, offers a prime example. He was the first to enter into a partnership with a technology company to facilitate his work, securing IBM resources to accelerate the process of concording a large corpus (Busa, "Annals" 84). Father Busa set the example for processing Humanities research data by computer, earning him a place as a key founder of the discipline of Digital Humanities (Hockey, "History"). Even traditional Humanities work is now reliant on computer technologies and resources, whether for communication between distant colleagues, for drafting syllabi or articles, or for accessing research materials without requiring a physical copy of a journal in hand. Whether due to Father Busa's trailblazing or the acceptance of desktop computers as a standard piece of equipment for individual academics, conducting research activities in the 21st century integrates methodologies and resources beyond reach at the turn of the last century, or in 1974 when Father Busa's first published edition of the *Index Thomisticus* went to print (Hockey, "History").

My own project is a product of the paradigm shifts that preceded it. As there are a broad range of mobile computing devices and practices to serve them in use, I have chosen to focus this thesis specifically on the application of HTML5 and CSS3 to tablet computers, and on the relationship between an academic project's website's markup and its comparative representation on desktop or laptop computers and tablets.

However, focusing on HTML5 and CSS3 in this way meant sacrificing a fuller picture of not only the mobile computing paradigm, but how to design adaptable websites. Firstly, the approach to HTML5 and CSS3 I take in this thesis is by necessity stripped to its essentials. Taking full advantage of the adaptive qualities of these specifications requires fuller attention to the principals of responsive design, a topic large enough to merit its own study. Second is the need for user testing. A site meant to accomplish a specific purpose, as an academic project's site is, must identify its users. This includes collecting information on how users interact with the site, and what devices they are visiting from. Without this information, the best a web designer can do is guess. Lastly, I was unable to delve into one of the major draws of the HTML5 specification, its inclusion of a number of features designed to reduce dependency on third-party solutions such as embedding video or drawing images directly into the page. These features bear closer examination for their utility to projects integrating embedded media, reliant on data caching, integrating geospatial information and so on.

Even with these restrictions, HTML5 and CSS3 are an elegant solution to the issue of adapting a site to suit the device viewing it. My aim in investigating their use is twofold. Firstly, I seek to situate these specifications within a larger tradition of web markup by paying particular attention to the forces driving the change from one approach to the next as directly analogous to the elements of scientific revolution identified by Thomas Kuhn, with the emerging web practices HTML5 and CSS3 represent as indicative of a larger paradigm shift centered around mobile computing. Secondly, by putting these specifications to practice and demonstrating not only the process of converting a site from XHTML and

CSS to HTML5 and CSS3, but their impact on tablet users, I seek to advocate for their adoption by current academic projects integrating a web presence. Together, they lend weight to the need to include changes to technology and web practices within an academic project's long-term planning website planning.

Chapter 2 - Father Busa, Thomas Kuhn and SGML

Computers, the Humanities and Father Busa

A flurry of innovation characterizes the modern world. Ideas ripple in cascades as they build on one another to affect areas of life from the sociological to the technological to the purely scientific. The emergence of new disciplines has been among the effects, and with this the transmutation of longstanding ways of understanding the world. The convergence of computer technology and the traditional Humanities has been particularly fruitful, permitting an entirely new perspective on and approach to longstanding research activities.

Though it is tempting to regard the use of computers to address Humanities problems as a new development, the affiliation is a long-standing one, nearly as long as the history of post-World War II computation. It began with Father Roberto Busa, the Jesuit priest generally credited as the first to see and pursue the potential of computers as a means of more effectively delving into text analysis. Though his story is hoary with age and repetition, it bears revisiting: Busa was engaged in the creation of a linguistic analysis of the writings of Thomas Aquinas, and first began exploring the possibility of mechanizing the process during World War II (Busa, "Perspectives"). In 1949, he travelled from Italy to the United States to approach IBM in the hope of gaining the company's support and resources for his project, and met personally with CEO Thomas J. Watson, Sr. (Busa, "Annals" 84). Busa succeeded in persuading Watson to back his work, securing access to the equipment and personnel of IBM's Milan office (84).

Thus began the age of computers in the humanities. Over the decades, Busa converted his project, now known as the *Index Thomisticus*, first to paper punched cards encoding one word and its associated references per card, then to reels of spooled iron magnetic tape capable of storing many cards' worth of data (Busa, "Annals" 84-85). Though they never eliminated the need for human processing and sorting, these early forms of computer storage dramatically increased the speed and flexibility of analysis, enabling users to sort 30,000 to 60,000 cards per hour by 1958 (Winter 13). The combination of cards and magnetic tape storage lasted until 1987, an impressive 38 years, when the process of transferring all the data to CD-ROM began (Busa, "Perspectives").

This is the story as known. It has some interesting lacunae -- though the medium of processing is well documented, the details of how the computational aspects were undertaken are curiously missing from most accounts. The incompleteness has the effect of deemphasizing the impressive technical expertise behind the project. We know the raw number of cards and reels of magnetic tape used, and the importance of the *Index Thomisticus* to advances in modern linguistic analysis and concordances, but remarkably little is readily available about the algorithms themselves, the capacity of the equipment, or the data encoding used. Some details emerged about the programming in the 1980s, for example that "[m]uch of the actual 'programming' was still in the design of the card, and the essential overall program was still visibly the process for producing a concordance laid out by Father Busa in *Varia Specimina*" (Winter 14). However, this has not been the focus of most papers written about Father Busa or the *Index Thomisticus*. The absence of technical details gives the impression that the project sprung, fully formed, from one person's guiding vision, stacks of cards and many adeptly sorting hands, changing only the medium of dissemination as the bound multivolume concordance gave way to CD and CD in turn to web hosting. Thomas Kuhn thought that, when viewed in retrospect, the work of past researchers is often most interesting where it can be understood as contributing directly to the traditions behind current practice (Kuhn 138). This suggests that Father Busa's work has been primarily understood as important for its contributions to concordances and linguistics, thereby deemphasizing his technical methodologies despite the fact that the one was necessary for the other. From that point of view, it is important that he used computers, but the retrospective view privileges the results over the algorithms or code that facilitated them. Quoting numbers of cards becomes flavour for the story rather than a tantalizing hint of some deeper processes.

Reducing focus on the method of data processing to narrow in on the replacement of computer devices as the technology improved implies a smooth continuity over the lifetime of the project. This narrative tack glosses over a number of interesting developments: What of the changes in computer equipment required for each, the interfaces that go with them, the capacity for internal storage or the impact of the emerging ability to handle algorithms without direct human involvement every step along the way? Punched cards, though more efficient than their handwritten equivalent, were still a problematic medium for storage and processing. Father Busa recounted, "I still remember how difficult it was to calibrate the lines between the punched holes, as the paper plate stretched progressively during operation" (Busa, "Annals" 85). Compounding the vagaries of the individual card, the scope of the project meant creating and storing a mass composed of hundreds of thousands of punched cards, which required manual sorting into sequence for processing (85). By contrast, magnetic tape allowed sequences of data to be stored compactly and run through the computer repeatedly without constant worry of the steadily degrading cards sticking in the mechanism (85). After this point, however, details become scarce.

Much is available about the linguistics research that later iterations of the project produced, but little about the programs, equipment and processes underlying it aside from Father Busa's commentary on the vagaries of punched cards in his 1980 article "The Annals of Humanities Computing: The Index Thomisticus." There are three principal sources for Father Busa's process outside of himself. The oldest of these is IBM employee Paul Tasman's 1957 article "Literary Data Processing," which describes the benefit of mechanizing a concordance, and details the phases of automation required to concord the *Index Thomisticus* (Tasman 254). The next article on the topic did not appear until 1981, when D.M. Burton published his "Automated Concordances and Word Indexes: The Fifties," describing the technical challenges Father Busa faced first in his Dead Sea Scrolls concordance, and then in the *Index Thomisticus* (Burton 3-4). The last and most recent proved to be Thomas N. Winter's 1999 article, which includes a summary in layman's terms how stacks of punched cards were processed into a complete concordance stored on magnetic tape (Winter 14-15). Both Burton and Winter rely heavily on Tasman's article for their information.

Father Busa wrote little on his process himself, aside from a volume entitled *Inquisitiones lexicologicae in indicem Thomisticum*, of limited circulation and best known via others' citations.[4] Indeed, he tells of how the task of recording and standardizing methodologies was something he preferred to leave to others:

> In 1967, Professor Antonio Zampolli, up to then my assistant, founded in Pisa the laboratory for computational linguistics which has made him internationally famous. In Tübingen at the final session the audience wanted me to accept the task of standardizing codes and methods. I refused. "My Big Boss above wanted to standardize religion. See how He succeeded? Could I be expected to be more efficient than He?" (Busa, "Annals" 86)

This display of professional modesty, though couched in religious conviction, has had the lasting effect of leaving the details of his early methods forever obscure, and suggests that the information about it is missing in accordance with his wishes.

This choice of what to record and what to leave out is remarkable not for its lapses, but for its ordinariness, both amongst researchers and a discipline's historians. An account reconstructing the paths a given discipline has taken contains the writer's decisions about what to emphasize and what to set aside to best serve the needs of the audience. It is simpler to imply progression by the accumulation of knowledge, as "additional research makes it harder, not easier, to answer questions like: When was oxygen discovered? Who first conceived of energy conservation?" (Kuhn 2). For this reason, new practitioners introduced to the foundational achievements of their field may be missing contextual elements such as the sociological environment or details of the supporting technology in

[4] http://www.worldcat.org/title/inquisitiones-lexicologicae-in-indicem-thomisticum/oclc/468279521&referer=brief_results

favour of an account pared down to those aspects judged most immediately relevant to understanding the element of the discipline under exploration. Whether he had the flow of history in mind or not, Father Busa chose to focus on his results, judging them more important than codifying the mechanical processes he employed to obtain them. Though this assists with constructing a retrospective narrative around the achievement, and to situate it within a larger narrative traced by related achievements, the algorithms, problem-solving and day-to-day refinements that produced those results end up glossed over. When the original researcher deemphasizes the long-term value of their own process, as Father Busa did, it is easier to understand the results as emerging linearly from the traditions underpinning current processes.

Thomas Kuhn's Paradigm Shift

A framework for understanding this peculiarity of history is provided by one of Father Busa's contemporaries, Thomas Kuhn, in his book *The Structure of Scientific Revolutions*. In the preface to the 1962 edition, Kuhn reveals that he first began the work that became the book as a graduate student in theoretical physics nearly fifteen years earlier (Kuhn v), concurrent to Father Busa's initial forays into computer-assisted concordance. Though written specifically to address the ways in which scientific revolutions occur, *The Structure of Scientific Revolutions* has contributed a versatile means of understanding the ways in which new ideas and practices emerge, of which Kuhn’s term ‘paradigm shift’ has become emblematic.

Understanding what Kuhn meant by a 'paradigm shift' requires the definition of two closely related terms, 'paradigm' and 'normal science'. Kuhn selected 'paradigm' to "suggest that some accepted examples of actual scientific practice -- examples which include law, theory, application and instrumentation together -- provide models from which spring particular coherent traditions of scientific research" (Kuhn 10). These models function to prepare students for membership within their field, while researchers working from shared paradigms may be confident that their colleagues share consensus on rules and standards of practice

(Kuhn 11). Such a consensus is required for the creation and continuation of normal science, which can then also be understood as research tradition (11). The concept of the paradigm can be applied to disciplines beyond the sciences, illuminating the activities of such disparate areas of research as philosophy, psychology, linguistics and art history, and proving a framework to understand the effect when their normal research goes askew (Kuhn 121).

With the establishment of a paradigm, normal science may take place. Kuhn defines normal science as "research firmly based upon one or more past scientific achievements, achievements that some particular scientific community acknowledges for a time as supplying the foundation for its further practice" (Kuhn 10). This offers a definition of the available methods and types of problems that may be addressed by a discipline's practitioners (10), one that need not be confined strictly to the scientific community. A set of practices defined in this way tends to endure where it can "attract an enduring group of adherents away from competing modes of scientific activity [while remaining] sufficiently open-ended to leave all sorts of problems for the redefined group of practitioners to resolve" (10). Together, the concepts of the paradigm and normal science provide a framework for understanding how a discipline's practices function, whether within the sciences or the humanities or non-academic fields such as web development.

They also contain the elements required for a change from one paradigm to another. A paradigm is only as good as its capacity to facilitate research. Kuhn asks, "[i]f the paradigm represents work that has been done once and for all, what further problems does it leave the united group to solve?" (Kuhn 23). Defining a paradigm as "an accepted model or pattern" obscures how limited a paradigm can be in its scope and precision (23). A paradigm need only be more successful than its predecessor in addressing the challenges of its field, and will not necessarily address all the problems the researchers working under it face (23). Though the paradigm will be refined over time, its shortcomings will perpetually leave the door open for a new way of doing things (Kuhn 24). Eventually, the limit of the research that may be accomplished under the paradigm is exhausted, or it is revealed to be unable to offer the solution to a pressing problem. Kuhn found that

> normal science possesses a built-in mechanism that ensures the relaxation of the restrictions that bound research whenever the paradigm from which they derive ceases to function effectively. At that point scientists begin to behave differently, and the nature of their research problems changes. In the interim, however, during the period in which the paradigm is successful, the profession will have solved problems that its members could scarcely have imagined and would never have undertaken without commitment to the paradigm. And at least part of that achievement proves to be permanent. (Kuhn 24-25)

This is what happened when Father Busa approached IBM. He found that the traditional hand-written method of producing a concordance was an immense and painfully slow task when applied to a corpus as large as the works of Thomas Aquinas. Dissatisfied with the methodology of his predecessors, he questioned whether this was truly the best way to go about creating a concordance and sought a way to mechanize the process. With the aid of IBM, Father Busa took an accepted practice within the literary studies paradigm, the 'normal science' of creating an indexed concordance, and discovered how it may be accomplished far more efficiently and effectively by utilizing punched cards for computer processing rather than relying solely on hand-written index cards and manual processing. In exploring the utility of mid-20th century computers to his research, he established the precedent for further Humanities work with computers. By the 1960s, there was sufficient interest in accomplishing related tasks with the aid of computers to support a dedicated academic journal, *Computers and the Humanities*, which continued publication until 2004 and persisted thereafter as *Language Resources and Evaluation* ("Computers and the Humanities"). Though computer technology, programs and programming have changed substantially in the decades since Roberto Busa began the *Index Thomisticus*, he was instrumental to establishing their utility to the Humanities. Since then, subsequent researchers have broadened the scope of research questions possible to explore far beyond concordances and linguistic analysis.

The change Father Busa instigated not only contributed to the paradigm of utilizing computers in Humanities research practices, it demonstrated a how a paradigm shift could take place. In Kuhn's terms, the process by which one paradigm gives way to another can be understood as a crisis, but one need not consciously recognize a crisis as such for it to have a lasting effect (Kuhn 84). For Father Busa, the crisis was recognizing the monumental nature of creating the *Index Thomisticus* entirely by hand. Kuhn identified three possible resolutions by which the end of a crisis may be known:

> Sometimes normal science ultimately proves able to handle the crisis-provoking problem despite the despair of those who have seen it as the end of an existing paradigm. On other occasions the problem resists even apparently radical new approaches. Then scientists may conclude that no solution may be forthcoming in the present state of their field. The problem is labelled and set aside for a future generation with more developed tools. Or, finally, the case that will most concern us here, a crisis may end with the emergence of a new candidate for paradigm and with the ensuing battle over its acceptance. (84)

Father Busa's work on the *Index Thomisticus* became an example of the third kind of resolution by offering a new paradigm for creating concordances and, more broadly, an example of new possibilities for undertaking research in the Humanities.

SGML and the Humanities

By the late 20th century, computer-based research had established a cadre of adherents within the Humanities. For these scholars, the issue at hand had become how best to handle specific kinds of research, such as applying metadata to literary texts and processing it effectively. The Standard Generalized Markup Language, commonly abbreviated to SGML, was commonly utilized by literary scholars as a means of doing so. SGML was first described as a formal standard in International Standard 8879 and released December 1986 by the International Organization for Standardization (ISO) (van Herwijnen ix).[5] It described a solution to an environment filled with proprietary programs for handling text corpora, each with its own interface and query language (Hockey, "An Agenda" 531). With SGML, one could create a document containing both the text and a description of its structure that is independent from computer hardware or software (ix). SGML documents resided in a plain ASCII file with markup elements embedded directly into the text (Hockey, "An Agenda" 533), making them highly transferable. Prior efforts, many of which were commercial, operated to their own standard and had their own hardware dependencies, posing a substantial challenge to separating one's data from the software and thus to archiving one's work or even transferring it from program to program (531). Ensuring data persistence became a high priority; academic researchers wanted some assurance that their work would continue to be accessible and readable into the future, not be effectively lost as the computer needed to run a program, or the program needed to read a file type, or even the storage device on which it resided, fell from use (531).

The challenge became to integrate the advantages of these document-handling programs while removing or minimizing the problem of carrying the data forward past the lifetime of an academic project. Some programs of this type integrated text markup, an innovation many found highly valuable as a means of including metadata right alongside the source material, but this frequently came at the cost of a human-readable document structure (Hockey, "An Agenda" 532). The value of markup, the desire to make it human readable, and the need to move away from proprietary formats contributed to the development of SGML. Most importantly for Humanities scholars, the Text Encoding Initiative (TEI) began producing guidelines for encoding humanities, social sciences and linguistics texts in 1994

[5] This standard can be obtained directly from the ISO. http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=16387

(TEI, "Text Encoding Initiative"). The TEI, which describes itself as "a consortium which collectively develops and maintains a standard for the representation of text in digital forms," swiftly became influential within academia and related disciplines, as libraries, museums, publishers and scholars adopted their guidelines for textual markup (TEI, "Text Encoding Initiative"). The TEI Guidelines "defin[ed] an ‘encoding scheme’ rendered in a formal markup language. The original TEI language (P1 through P3) used SGML syntax. With P4, users were given a choice of using SGML or XML; with P5, SGML is no longer an option" (TEI, "Introducing the Guidelines").

SGML users, whether utilizing the TEI's guidelines or not, found its markup not only added a new dimension to information in a manner independent of medium, but opened up new methods of processing (van Herwijnen ix). SGML documents organized information in such a way that easy retrieval was possible without tying the data to any one application, while allowing a document to behave much like a database (van Herwijnen 207) by creating a structure analogous to records and fields that could be queried (Bradley 5). Freed from ties to a proprietary program, operating system or type of computer, SGML documents were regarded as an effective means to transfer data between legacy and replacement systems, and to improve data quality through adherence to a predetermined structure (Travis and Waldt 5). Among its advantages was its utility as a system of structuring data that could be used with several others in the creation of a larger data system (Travis and Waldt 10). These traits made SGML attractive not only to businesses seeking an information capture, management and delivery system (Travis and Waldt 7), but to textual scholars as well. The Text Encoding Initiative (TEI) provided best practice guidelines on how to mark up a humanities-related text, including which features to encode and how to go about doing so, while leaving how much or how little to integrate into a given text at the discretion of the researcher (Hockey, "An Agenda" 533). These guidelines also included a "comprehensive list of features likely to be found in humanities-related material" (533), and, through the flexibility of the encoding, the means for researchers to represent and document multiple interpretations of a text (Hockey, "An Agenda" 534). Part of SGML's appeal was the ability to cross-reference points to others within documents, to points between documents, and to images via hyperlinks (534), thereby enriching documents through direct intertextuality. With the establishment of SGML as an ISO standard, the TEI's guidelines functioning as a model, and the standard's offer of a tenable solution to the problem of sustainable data formats, SGML resulted in a new paradigm for Humanities research, one that made longstanding and internationally respected endeavors like *The Orlando Project*[6] possible.

Though SGML offered a brilliant solution for encoding texts and carrying them forward for archival or future research purposes, users eventually began to chafe at its restrictions. Over time, the markup paradigm shifted in favour of its daughter languages, notably HTML and XML. The principals underlying SGML -- plain text storage, hierarchic structure, ability to interweave metadata into a text, concept of semantic elements and so forth -- have been retained within HTML, XML and related markup languages. This change poses a new question: What happened to make SGML, an effective and widespread innovation, the less desirable solution?

As users encountered SGML's limits, such as its inability to create relationships between documents or lack of a formally defined element hierarchy, some developed offshoots in the attempt to refine or add to its capabilities. This is a classic example of a paradigm failing to meet the needs of practitioners, paving the way for a paradigm shift as individual users attempted to work around these limitations with alternate solutions based on but diverging from the paradigm. The resultant offshoots expanded on SGML with functions such as enhanced hypertext linking, automated transformation into a document ready for publication, or a focus strictly on the creation of interlinked hypertext documents on the burgeoning internet. HyTime, DSSSL and SPDL were notable amongst those that coexisted alongside SGML for a time (Bradley 8). HyTime used existing SGML structures to add hypertext links, including targeting positions in a document by a

[6] http://www.arts.ualberta.ca/orlando/

specific point within an element, or by identifying its position in a document hierarchy (Bradley 98). By contrast, DSSSL (Document Style and Semantics Specification Language) offered a means of defining an SGML document's structure, enabling the data to be restructured or used with another application (Bradley 8; Travis and Waldt 488). SPDL transformed DSSSL output by describing where data should be placed on a page, applying font information and page area details to the final document (Bradley 8). Each offshoot fulfilled a specialized need that could not be accommodated with SGML alone, but required a basis in SGML to accomplish.

While the offshoots enhanced what one could accomplish with SGML, none represented the substantial departure from the standard that HTML did. HTML was developed in response to the need for a compact, efficient alternative to SGML for hypertext documents within the CERN lab in the early 1990s, and it proved a sufficiently popular means of document encoding that it moved out of the lab and into the Web at large (Ray 1.2.1). CERN, the European Laboratory for Particle Physics in Geneva, Switzerland, is a highly collaborative research institute, and in 1989, computing services employee Tim Berners-Lee began developing and implementing a method for enabling remote researchers to pool and access resources such as research documents (Raggett et al.). When Berners-Lee wrote his prototype for HTML, he borrowed extensively from SGML, keeping it independent of format for text display and repurposing semantic elements such as `<title>`, `<p>` and header elements `<h1>` through `<h6>`, while supplementing them with hypertext links of his own invention and combining these links with anchor elements (Raggett et al.). By 1994, HTML had been not only adopted by users outside CERN but so important to the emerging Web that the World Wide Web Consortium (W3C), headed by Berners-Lee, formed as a body to develop open standards (Raggett et al.) However, HTML's predefined, all-purpose set of markup elements meant a large part of what made SGML so flexible was lost, and so the W3C began work on XML (Ray 1.3) as a means to blend SGML's powerful user-defined semantic elements and nested hierarchic structure with the needs of the Web and web browsers (Ray 1.2.1).

This environment, in which what was once a single unified standard must compete with a wide variety of more specialized and refined standards, is the type of situation Kuhn referred to when he stated that "research under a paradigm must be a particularly effective way of inducing paradigm change" (Kuhn 52). As SGML matured, its users extended the language beyond its native capacities according to the needs of the businesses, publishers, researchers and so on that had become reliant on it. Referring to such innovations as discoveries, Kuhn observed "they are not isolated events but extended episodes with a regularly recurrent structure. Discovery commences with the awareness of anomaly" (52). Here, 'anomaly' is the recognition that, for example, SGML's hyperlinking capacity is admirably designed to accomplish linking within a document (Bradley 91-92), but users, such as the researchers of the CERN lab, needing to link between documents, a natural and logical extension, have no means of doing so within the standard without devising some customized modification of it. This is why "to be admirably successful is never...to be completely successful" (Kuhn 68) -- there are always problems to be solved that even the best solution cannot wholly anticipate.

These kinds of developments collectively contribute to paradigm change. Once the community assimilates such changes, they permit researchers "to account for a wider range of natural phenomena or to account with greater precision for some of those previously known" (Kuhn 66). In the case of the Humanities and SGML, activities such as textual analysis on corpora marked up to a common standard will have to substitute for 'natural phenomena'. Regardless of whether a researcher is operating under the sciences or the Humanities, it holds that by "discarding some previously standard beliefs or procedures and, simultaneously, by replacing those components of the previous paradigm with others" (66) researchers can shape a new paradigm that better suits the questions facing the discipline. For example, SGML offered an innovative and effective means of marking up, adding metadata to and analyzing complex documents such as literary works. However, with the rise of the internet in scholarly circles, some researchers found web dissemination preferable for sharing marked-up documents.[7] The desire to represent complex documents on the web led to HTML and XML, arguably the best known of the standards underlying the current web development paradigm, as specialized offshoots of SGML. XML became a popular means of storing and transmitting marked up documents on the web (Goldberg xi), as it combines SGML's flexibility with HTML's capacity to display the results in a web browser. Once again, the end of a crisis of normal science through the emergence and triumph of a new paradigm candidate (Kuhn 84) can be seen both in researchers' adoption of SGML as a method of document markup and in their later preference for XML where they required a web-ready means of accomplishing similar tasks.

HTML, XML and the Humanities

It is worth noting that some scholars in the Humanities long accustomed to SGML regarded both HTML and XML with trepidation. In her 1998 paper "An Agenda for Electronic Text Technology in the Humanities," Susan Hockey comments on the rise in importance of the World Wide Web to computing activities, and has the following to say of HTML:

> The Hypertext Markup Language (HTML) used on the Web, however, is rather limiting. It is a simple application of SGML and was designed mostly for the display of material on screen (or in print). It is much less suitable for analysis of the data. For example, with an HTML file, it would not be easy to carry out many of the literary and linguistic computing functions described earlier in this article. The linking mechanisms on the Web are also rather weak in comparison with those which can be built with SGML. (Hockey, "An Agenda" 537)

She speaks somewhat more favourably of XML:

[7] Though SGML remains in use. The University of Victoria's *Internet Shakespeare Editions* maintains tagging guidelines for SGML, HTML and XML, on the basis that "[t]he three schemes of tagging--HTML, XML, and SGML--can conveniently be employed to reach different audiences and satisfy different scholarly needs" (Best).

> A new markup language XML (Extensible Markup Language) has recently been approved by the World Wide Web Consortium and is likely to be supported in the next generation of Web browsers. Briefly, XML is a cut-down version of SGML which will run directly on the Web. It enables the creators of Web pages to define their own markup tags so that they are no longer restricted to those tags which are defined in HTML. The impetus for XML has come from industry, but it is pleasing to note that XML has adopted some of the mechanisms originally defined by the TEI. (537)

Hockey's discussion of HTML, XML and web technology betrays a tension between the need for markup languages better suited to the web environment and an overriding awareness of these languages' shortcomings in comparison to SGML. This illustrates one of the ways in which "[t]he transition from a paradigm in crisis to a new one from which a new tradition of normal science can emerge is far from a cumulative process," (Kuhn 84) as those invested in the prior paradigm may not be swiftly persuaded to switch.

Should a discipline split into two groups, one allied with the old paradigm and one with the new, their research practices effectively occupy two different worlds (Kuhn 150). Switching from one paradigm to another has been likened to a change in visual gestalt, as happens when one views the famous optical illusion that first looks like a duck, and then like a rabbit, but cannot be perceived as both at once (Kuhn 85).[8] However apt the comparison, Kuhn cautions that the "parallel can be misleading. Scientists do not see something *as* something else; instead they simply see it...the scientist does not preserve the gestalt subject's freedom to switch back and forth between ways of seeing" (85). Gestalt nevertheless remains a useful way of understanding what Kuhn terms "full-scale paradigm shift" (85).

Crisis, as occurred when SGML's limitations were reached subsequently supplemented with modifications and additions to the standard, is fundamental to setting the stage for paradigm shift; a crisis need not even develop far at all or be

[8] http://mathworld.wolfram.com/Rabbit-DuckIllusion.html

recognizable as such before triggering the emergence of a new paradigm (Kuhn 86). In the same paper in which Hockey acknowledges the need for HTML and XML in the emerging web environment, she describes the ways in which the value of SGML is still being extended within Humanities research. In her view, this includes its use within text editors such as WordPerfect, and how it can, with only a few small modifications, be used to describe images and other non-textual objects (Hockey, "An Agenda" 535). This demonstrates how one paradigm may have substantial overlap with another, even within one person's research practices. In this way, the crisis and new paradigm may emerge by small increments, moving the researcher gradually toward experiencing a full-on paradigm shift. Hockey, though still advocating for SGML at the time of writing, had already recognized that environmental changes had limited the context of its usefulness, and despite how HTML and XML suffered in comparison to SGML, these new markup languages were more appropriate to representing Humanities content on the web. Viewing it in retrospect, Hockey's 1998 comments mark a point at which the markup paradigm was tipping toward web-based variations.

Assisting the emergence of a new paradigm is its incorporation of aspects of the old, albeit redefined or reshaped from their original form:

> Since new paradigms are born from old ones, they ordinarily incorporate much of the vocabulary and apparatus, both conceptual and manipulative that the traditional paradigm had previously employed. But they seldom employ those borrowed elements in quite the traditional way. Within the new paradigm, old terms, concepts, and experiments fall into new relationships one with the other. (Kuhn 149)

Consider the way HTML is described in Elizabeth Castro's *HTML 4 for the World Wide Web (4th ed.)*, released in 2000:

> All Web pages are written with HTML. HTML lets you format text, add graphics, sound and video, and save it all in a Text Only or ASCII file that any computer can read...[t]he key to HTML is in the *tags*, keywords

> enclosed in less than (<) and greater than (>) signs, that indicate what kind of content is coming up. (Castro 11)

Regarding the file format, she adds:

> [Because HTML documents are saved as ASCII or Text Only files, virtually any computer can read a Web page. It doesn't matter if your visitors have Macintosh or Windows machines, or whether they're on a Unix box or even a hand-held device like a Palm. The Web is open to all. (Castro 12)

Compare Castro's description of HTML to Kevin Howard Goldberg's of XML in his *XML: Second Edition*, released in 2009:

> XML was specifically designed for data storage and transportation. XML looks a lot like HTML, complete with tags, attributes and values...[b]ut rather than serving as a language for displaying information, XML is a language for storing and carrying information...it is easily extended and adapted. You use XML to design your own custom markup languages, and then you use those languages to store your information...XML can also be used to share data between disparate systems and organizations...an XML document is simply a text file and nothing more. (Goldberg xiii)

Both descriptions emphasize universality, though Castro promotes HTML as a means of representing data on the web while Goldberg promotes XML as a means of storing and transferring data. This emphasis is a direct carryover from SGML, as can be seen in Neil Bradley's description from his 1997 book *The Concise <SGML> Companion*:

> The SGML format is designed to operate on any computer platform, and all popular platforms are now well supported by SGML-aware software products...SGML is an international standard...[e]xisting and forthcoming standards either complement SGML or overlap its functionality in a limited respect...SGML enables a high degree of control over information

> content. It ensures that documents conform to agreed structural standards by providing a mechanism that allows a document to be compared with rules defined for a specific class of documents. (Bradley 5)

Bradley also emphasizes that "SGML owes its strength and flexibility in part to its use of ASCII (Bradley 9), and that "[t]he elements available for use in a particular document are determined by the author of the DTD [Document Type Definition] that is applicable to that document" (Bradley 15).

These descriptions show that, though all three languages utilize markup to represent data, XML is more closely related to SGML in purpose. All are described in common language, and utilize common conventions, so that someone familiar with one could conceivably transfer their knowledge of one to learning the other two with little fuss. Yet, beyond these overlaps, HTML is poorly situated for data storage, as is XML for display, and SGML for use in a web environment. Where the internet is the most common means of sharing and presenting data, HTML and XML are the clear choices due to the specialization they have integrated over the generalization their predecessor SGML was once so successful for.

Making the shift from SGML to HTML and XML required recognition of a changing environment, and that changing environment's impact upon one's field. The open-mindedness Susan Hockey displayed in the early days of HTML and XML cannot always be relied upon when such new developments emerge. There will frequently be holdouts within a research community, who do not see the value in the new paradigm and persist in advocating for the paradigm to which they are accustomed. In Kuhn's view,

> neither proof nor error is at issue. The transfer of allegiance from paradigm is a conversion experience that cannot be forced. Lifelong resistance, particularly from those whose productive careers have committed them to an older tradition of normal science, is not a violation of scientific standards but an index to the nature of scientific research

> itself. The source of resistance is the assurance that the older paradigm will ultimately solve all its problems. (Kuhn 151)

Resistance is not a fault of the adherents of the old paradigm, but rather a trait that makes research under paradigms possible, no matter how stubborn it may seem to researchers who have moved on to another paradigm (Kuhn 152). As he noted before, "[s]ometimes normal science ultimately proves able to handle the crisis-provoking problem despite the despair of those who have seen it as the end of an existing paradigm" (Kuhn 84). In the case of SGML, it was not.

Paradigm Shifts and HTML

The process of paradigm shift is once again in motion. Even as SGML gave way to XML and HTML, HTML is now transmuting into something substantially different from where it started out, in no small part due to the expanded variety of means now available to access web-based data. Delving into the elements of a paradigm shift reveals some striking parallels with the present state of computers and the web. In just the last few years, accessing computational and web-based functions, long confined to a desktop or laptop machine on the general consumer level, has expanded to include a variety of highly portable devices priced for wide consumer appeal capable of accomplishing many of the same tasks. At the same time, the methods used to create and present web content optimized for desktop and laptop computers are now giving way to ones adaptable to multiple devices.

The most notable development has been the recent introduction of the first widely affordable, lightweight and powerful tablet computer to the consumer market in the form of Apple's iPad. Since the iPad's release in early 2010 (Apple, "iPad Arrives"), the impressive sales figures of tablet computers, devices occupying the niche between laptop computers and smartphones, have shown them to be the most enthusiastically embraced of these new consumer electronics (Gartner). Concurrent with the release of Apple’s iPad, the W3C released new versions of HTML and CSS, known as HTML5 and CSS3 respectively, optimized for handling web content through such devices while expanding the multimedia and scripting functions included within their standards (Gilbertson, "Drink Up

HTML5"). These developments, taken together, form a paradigm shift that is changing how we, individually and as a society, relate to the computer devices and information at our disposal.

Chapter 3 - An Ordinary Website: XHTML, CSS and the Desktop Paradigm

XHTML's Place in the Web Paradigm

The alchemical transformation developers enacted on the base of SGML to produce HTML and XML did not end with its transmutation into these derivative languages. The process is an ongoing one, with users continually innovating new ways to use the languages and the web. How developers use HTML in particular has undergone dynamic changes as the ways in which everyday users relate to the web has expanded. The widened array of web-based technological innovations available to users and developers is one such expansion, facilitating new methods of accessing and viewing web content.

The early days of CSS, the web specification now used to create the presentational aspects of web pages, illustrates the multiplicity. The W3C created CSS to separate the content of a page from its formatting, with the aim of solving the problems of formatting websites with HTML markup alone (Castro 16).[9] In 2000, when HTML 4 was the most current edition of the W3C's recommendations, cascading stylesheets were largely unsupported by most browsers, with the exception of Opera, and though some developers embraced the possibilities stylesheets opened up, others regarded them as problematic (16). This same problem affected markup as various organizations released their own web browser to the market, with each competing browser supporting a slightly different and sometimes proprietary set of tags (Castro 14). By the time the W3C rolled out HTML5 2010, HTML markup had settled into several widely supported standards. There are now five permutations of HTML 4.01 in common use (Henick 10) in addition to XHTML, a variation on HTML 4.01 designed to be XML-conformant (Henick 11). As a whole, HTML has resisted a single standard. While HTML 4's looser standards can easily result in poorly formed markup,

[9] Before CSS, tables were commonly used to arrange a website's content (Henick 163).

XHTML is too strict as originally defined (11). Despite the W3C's intention that XHTML replace HTML 4.0 as an interim standard in alignment with XML, issues like its lack of backwards compatibility with some browsers proved a stumbling block to universal adoption (Pilgrim 9). For this reason, it is difficult to pinpoint one definitive version of web markup.

XHTML is a favorite of instructional books for its strict adherence to rigid syntax, which makes it easier than its HTML counterparts to both read and write (11) and attractive to designers wherever the consistency that comes from such syntax is paramount, such as in academic projects' websites. As a result, XHTML, generally combined with visual styling via CSS, is broadly representative of web markup up to the release of HTML5. XHTML's place in the web paradigm has persisted alongside HTML5, as can be expected from Kuhn's observation that old and new often coexist for a time until either old or new is proved to best solve the problem at hand (Kuhn 53). XHTML is well suited to creating web pages intended for viewing on desktop or laptop computers, and will continue to be useful for sites where that is the primary means of access.

All of this makes XHTML a useful measure against which to hold HTML5. Early in my development of the web project portion of this thesis, I realized that directly incorporating XHTML into my project offered a useful prelude to my HTML5 website. The challenges faced by the developers of a preexisting website adapting the markup are distinct from those creating a website from scratch. I determined that comprehension derived from personal experience offered me a stronger grounding than distilling others' accounts, which would be particularly important for my thesis to effectively advocate for the value of adopting HTML5 and CSS3 as a strategy for extending a website's long-term sustainability. As this thesis is arguing for updating websites' markup to these new specifications, I determined that I could not fully understand that process and its impact without attempting, documenting and analyzing a real-life shift from XHTML to HTML5. Taking this approach also laid the groundwork for testing the resulting site on a tablet, allowing for a direct comparison of the site's behavior as constructed in XHTML and HTML5.

I settled on HTML5 and CSS3 in particular after considering several approaches to the problem of developing web pages capable of dynamically adapting to their viewing medium. The issue of adaptation, to create one website that will display appropriately on multiple devices rather than several independent but related websites each geared to a particular device, predates tablet computers. The advent of the smartphone as a common consumer device offered a notable spur to investigating the adaptability of academic websites to devices beyond the desktop or laptop computer within the field of computer science. Articles such as Guadalupe Ortiz and Alfonso García De Prado's "Improving device-aware Web services and their mobile clients through an aspect-oriented, model-driven approach" and Chimay Kulkarni and Scott R. Klemmer's article "Automatically Adapting Web Pages to Heterogenous Devices" explore the options available to web designers to accommodate a user base increasingly reliant on these devices.

Ortiz and García report on the results of their research into Aspect-Oriented Programming and model-driven development, aimed at the creation of flexible web services that will display appropriately on a variety of devices beyond desktop or laptop computers. Their article is a direct recognition of the increasing prominence of mobile device users as consumers of web services, and the importance of targeting web pages to device type due to the considerable difference between viewing web content from a desktop computer than a mobile phone (Ortiz and García 1080). Their solution was to develop a methodology for targeting devices based on fixed characteristics such as screen size, and tailoring the web page elements that the user sees accordingly (Ortiz and de Prado 1082). They recommended targeting via modifying a web page's SOAP header -- defined by the 24 June 2003 W3C Recommendation as "a lightweight protocol for exchanging structured information in a decentralized distributed environment" (W3C, "Soap Version 1.2 Part 1") -- element as "the most appropriate for the Web service environment due to its transparency, loose coupling and nonintrusiveness" while having the advantage of not requiring any action on the part of the user (Ortiz and García 1084). Device targeting meant that changes to one aspect of the page did not entail regenerating the full system (Ortiz and García 1092), so that desktop users would be unaffected by changes aimed at mobile phone users.

Chimay Kulkarni and Scott R. Klemmer's article "Automatically Adapting Web Pages to Heterogenous Devices" offers a related approach to adapting web pages. They dislike approaches that aim for a "lowest-common denominator device and create a minimal website that is accessible from it" as this approach results in a poor experience on other devices (Kulkarni and Klemmer 1574). Much like Ortiz and García, they recommend device-specific versions optimized for the desired devices (1574). Their approach relies on identifying those elements of a web page's design that will remain consistent across devices and mapping the relationships between elements on the desktop version of the page (Kulkarni and Klemmer 1574-5). This includes logos, navigation bars or article text (Kulkarni and Klemmer 1575). However, pages can only be customized for known devices, structure-based adaptation does not account for the effect on the content of those structures, and they could not include data on how users interact with adapted pages in their calculations (Kulkarni and Klemmer 1577). To offset these limitations, they emphasize the importance of user studies and an ongoing need to reassess user needs as new classes of web-ready user electronics become widely adopted.

Kulkarni and Klemmer's approach is similar to a trend in web design commonly referred to as 'responsive design.' Defined as "the approach that suggests that design and development should respond to the user’s behavior and environment based on screen size, platform and orientation," responsive design entails "a mix of flexible grids and layouts, images and an intelligent use of CSS media queries" (Knight) within a site's design.

Ethan Marcotte describes responsive web design as analogous to responsive architecture:

> Recently, an emergent discipline called “responsive architecture” has begun asking how physical spaces can respond to the presence of people passing through them. Through a combination of embedded robotics and tensile materials, architects are experimenting with art installations and wall structures that bend, flex, and expand as crowds approach them. Motion sensors can be paired with climate control systems to adjust a room’s temperature and ambient lighting as it fills with people. Companies have already produced “smart glass technology” that can automatically become opaque when a room’s occupants reach a certain density threshold, giving them an additional layer of privacy. (Marcotte)

He suggests that these same principals are directly applicable to within web design.

> Rather than tailoring disconnected designs to each of an ever-increasing number of web devices, we can treat them as facets of the same experience. We can design for an optimal viewing experience, but embed standards-based technologies into our designs to make them not only more flexible, but more adaptive to the media that renders them. In short, we need to practice *responsive web design*. (Marcotte)

However, neither device targeting nor responsive design speaks to the concerns underlying academic projects' websites. These sites serve multiple purposes. At their most basic, a project's website is a place for the project to present its findings. For some, it is also a means for contributors to add to its dataset, and can also offer a means for other academics to view or interact with the project's dataset. In short, they represent the project, act as a workspace and provide a means of disseminating findings to members of the larger academic community.

For the purposes of this thesis, I have chosen to focus my work specifically on academic websites affiliated with literary and textual projects, as there are both longstanding and recent examples that will be affected by shifts in the larger web environment toward accommodating mobile computing. As the web edition of

Father Busa's *Index Thomisticus* and Susan Hockey's interest in XML as a web-ready successor to SGML for textual markup demonstrate, presenting literary material on the web is a priority within the Humanities community. With their example in mind, there are direct historic reasons to focus on project websites of this type.

However, the move toward widespread mobile computing is not a concern unique to literary academic projects. Tablets and smartphones are an established popular means of accessing the web, and ensuring a site's design is mindful of these users is a topic of broader relevance. The approaches I employ are focused on using markup and styling to adapt websites, from a start position of common combinations of XHTML and CSS, for mobile device users. As such, this thesis is applicable to a wide variety of websites utilizing similar markup and structural choices, despite its specific grounding in academic projects centering on literary studies.

Each of a website's purposes has a set of needs attributed to it, needs which I aim to preserve and safeguard via conversion to web specifications designed to not only expand what a website can natively offer, but accommodate contributors and colleagues who may be using a wide variety of devices to access it. As a whole, starting my own website's design with XHTML offers greater understanding of how a site will behave before and after conversion, the better to guide projects seeking straightforward adaptation and those seeking to adapt with the aim of including the needs of tablet users into their site design.

In keeping with this aim, I planned and executed the site in two stages. I designed an initial version utilizing XHTML and CSS, to represent the state of normal science underlying a preexisting academic project's site. I followed this with a second version, created in parallel to the initial version, in which I took the base of XHTML and CSS and converted it to minimalist HTML5 and CSS3 with additional basic tablet functionality. Together, these two variations of the same site and content represent the paradigm of the academic project website designed for exploration on a desktop or laptop computer, and the emerging web paradigm designed to also accommodate users of emerging web-ready devices, represented here by the iPad tablet computer. Putting them side-by-side offers a way to see the transition period moving from one state of normal science to the next in action. The juxtaposition demonstrates how "[d]uring the transition period there will be a large but never complete overlap between the problems that can be solved by the old and by the new paradigm. But there will be also a decisive difference in the modes of solution" (Kuhn 85). Enacting the adaptation of my site from XHTML and CSS to HTML5 and CSS3 places the site within the overlap. However, the state of overlap also situates it to displays the opportunities adaptation opened up for the site to break from the interaction of legacy structures and decisions with the new framework; these would have been difficult to replicate in a from-scratch HTML5 and CSS3 build.

In the early stages of this thesis, I identified two options for adapting a site. The first was to approach the lead of an existing project and gain permission to utilize its data and webpage. At that time, I already had considerable experience, including work on the web infrastructure, on such a project -- Dr. Patricia Demers' *Canadian Women Writing and Reading from 1950* (CanWWR).[10] Leveraging this experience and my relationship as a project team member with Dr. Demers would allow me to take advantage of CanWWR's infrastructure. However, ongoing projects such as this come with their own set of design constraints, the complexity that comes from years of development, and the challenges of understanding programming structures and design decisions that emerge when team members move on to other opportunities while leaving varying amounts of documentation behind. Adapting CanWWR would therefore entail traversing a high learning curve, despite my experience, before adaptation could be planned or proceed. It would additionally mean negotiating the goals of my thesis with the goals of the project, and accepting the risks of making major changes to a live website with years of research invested in it.

[10] http://www.canwwrfrom1950.org/

My second option was to create a new website modeled on an existing project, and then adapt that website. This approach requires a host of infrastructure, from server to data storage method to content. Though it comes loaded with creating from scratch all the elements found ready-built into the website of a project like CanWWR, it offers the benefit of direct experience with the design considerations required to build this kind of site from the ground up. This trades learning the structure of a preexisting site for spending time and effort learning how to create that infrastructure in the first place, while gaining direct experience with the design considerations involved. It affords greater control over the scope of the dataset and site, and more freedom to explore or experiment with design options without the potential for conflict with an existing project's desires or goals, or worries about adversely affecting an active public website.

The second option proved most attractive, as I realized it was valuable to establish a precedent with a proof of concept, rather than requesting a leap of faith on the behalf of CanWWR. My aim, in designing my site, is for it to function as an argument for the validity of making a conversion to HTML5 and CSS 3 on its own merits, and make it accessible to small projects that may not have a large workforce behind them. As this project was additionally designed to test my web design, programming and data management skills, creating a stand-alone website unconnected to active research contained the risk of modifying its infrastructure to a site without years of research data invested in it. I thereby minimized any harm I could potentially incur in learning to making the changes I needed, while establishing the practicality of converting. This approach foregrounds creating a precedent, in the hope of better proving the merits of converting to HTML5 and CSS3 to established projects that may be risk-averse or hesitant to commit their budget to an unproven infrastructure alteration.

CanWWR is an attractive model on which to base my own project as a living example of how a simple research idea can flower into a rich project with a supporting website. Its strength is the way it puts recent Canadian women writers and their works to the forefront, with the core focus on "the immense productivity of women writers within the rapidly changing culture of the mid-twentieth century to the present day" (Demers). Likewise, Dr. Demer's choice to expand the definition of 'writer' beyond prose to include composers, singer-songwriters, screenwriters and graphic novelists offers a unique lens through which to understand the creative works of Canadian women.[11] Since 2008, CanWWR's website has offered a comprehensive listing of writers, works, genres, awards won and chronologies, expanding over time to include interviews, surveys, a bibliography of critical studies on many of the writers and works, a listing of writers' alternate names, and biocritical entries (Dyrbye). Save for the interviews, studies and biocritical entries, which are encoded as plain text XHTML files, this information has been stored in a relational MySQL database, with PHP scripts mediating between that database and the public website. CanWWR displays the listing of works associated with a particular writer, chronology or genre in a simple HTML table, generated as needed by applying rules to the results of appropriate database queries with PHP scripts. The site itself is marked up with XHTML 1.0 Transitional, in accordance with Henick's observation of its greater ease of reading and writing in comparison to HTML 4.01 (Henick 11).

As a mature, active project, CanWWR is a prime candidate for conversion to HTML5 and CSS3. It can be taken as representative of an academic project website in literary studies in the Humanities. In total, CanWWR contributed several key characteristics to my website:

1) Markup in XHTML 1.0 Transitional, with presentational formatting achieved via CSS stylesheet.
2) The use of a modular page structure, in which sections that repeat from page to page such as the header and navigational sidebar are stored in a separate file and called via PHP as needed.

[11] See CanWWR's subsection Genres: Other Written Arts (http://www.canwwrfrom1950.org/index.php?page_id=3&show=Other%20Written%20Arts&genre=Other%20Written%20Arts).

3) The assumption that users will be accessing the site from a desktop or laptop computer.
4) A dataset composed of and structured by a specific literary interest, modeled in how it catalogues the names, works and resources related to Canadian women writers active from 1950 to the present.
5) An underlying MySQL database to store its dataset.
6) Reliance on PHP to query the database and automate the XHTML formatting of the resulting data.

These characteristics offered a variation on the website paradigm within which I could work, one that suggests a dataset, a means of storage, a version of markup and a basic visual aesthetic to emulate. Though the means of data storage and scripting vary across websites, CanWWR is not unique in its presentation amongst academic project websites. For example, a source view of several other sites, including the *Canadian Writing Research Collaboratory* (CWRC), [12] *The Orlando Project*[13] and *Witchcraft in Early Modern England* (WEME), [14] reveals a preference for XHTML 1.0 Transitional. All three also integrate a structure with modular parts, sharing the use of a header feature and columns set to the right or left containing navigational features that carry across all pages of the site. The modular structure additionally extends to projects like the *Text Analysis Portal for Research* (TAPoR), which offers a set of text analysis tools.[15] With the sole exception of TAPoR, as a research facilitation project, all of these projects are built around gathering and presenting research centered on texts, those who produce them, and the conversations around them, requiring pages supported by databases directly expressing their research interests. It is these characteristics I seek to emulate in my own project.

[12] http://www.cwrc.ca/en/
[13] http://orlando.cambridge.org/
[14] http://witching.org/
[15] http://taporware.ualberta.ca/

For the purposes of this thesis, my choice of markup and styling are most important. These two characteristics are directly affected by adaptation to HTML5 and CSS3. However, the method of data storage chosen, such as a relational database or XML, is independent of the web markup on a project's public site. Data storage choices will influence the programming stage that takes raw data from a database or XML file and transforms it into HTML or XHTML suitable for viewing in a web page. Whether querying a database or pulling information out of XML tagsets, web presentation requires that the relevant data be transformation into a browser-readable form of markup.

Choosing and Structuring the Dataset

The literary focus of CanWWR's dataset inspired me to utilize a similar dataset for my own website. I did not wish to leverage another project's dataset, as I wanted a small but broad selection of books to work from to maximize the variety of data I could represent. As I am interested in the interaction between readers and works, and the conversations diverse subjects can engage in through individual readers, I turned to my personal book collection to populate my pages and chose a bookcase of nonfiction to focus my efforts on.

I composed my dataset's purpose by combining my knowledge of other, similar work with fiction. This produced a fictitious academic project entitled "Page by Page: 21st Century Lives Revealed by the Books of the House of the Books," and a frame story in which to situate it. While CanWWR is dedicated to collecting information on the productivity of Canadian woman writers from 1950 to the present day, including full lists of published works in a wide variety of types with such details as awards won (Demers), the project I have loosely modelled on it is organized around exploring a personal collection of books. My project is not intended to be a slavish reproduction, but rather a reinterpretation of the same general model of data presentation, using a comparable dataset. As a more intimate dataset, it is interested not only in identifying volumes by writer and subject, but also in clues to the mind of the reader or readers such as marginalia, flagged pages and the wear of multiple readings. To keep the dataset manageable, I collected this information from the contents of a single bookcase, while to maximize the variety, the texts chosen were primarily nonfiction with a smattering of canonical historic literary texts and annotated editions of classic children's fiction. All details noted about each, such as the presence of marginalia, flagged pages, or characteristics such as wear or library discard status, are accurate. Only the project's frame story is fictional.

The frame story took seed from a book entitled *Motel of the Mysteries*, an archaeological parody aimed at children blending the destruction of Pompeii with Howard Carter's discovery of Tutankhamen's tomb. As in David Macaulay's tale, my frame story is set in the future after a catastrophic event, and centers around the rediscovery and excavation of habitations -- in Macauley's case, the close examination of a single guest room in a motel complex; in mine, a single home within a subdivision. In this context, the contents of a private home could easily be of academic interest. The combination of books to provide data and a frame story to justify their use enabled me to proceed with constructing my dataset, and from there, to building the web page that would display it.

My experience working on CanWWR influenced not only the type of data I collected but also my choice to store it in a MySQL relational database. My experience on CanWWR working with a similar dataset suggested that this type of data structure is a good fit for the type of project I had chosen. This is in part because relational databases work well for storing information that falls into regular, predictable patterns, as is the case when working with books and authors. In addition, MySQL is supported by a suite of pre-set PHP functions that can streamline integrating and managing database content in a website. As CanWWR also utilizes PHP to dynamically generate its individual pages, and this structure is one I identified as important to my own page, a MySQL database proved a good fit for my overall goals. If I were working with large blocks of text, such as correspondence analysis or biographic information, and seeking to collect detailed information about that text for analysis, instead of collecting specific, discrete pieces of information about physical objects, XML would be the better choice. In that case, I would be looking to CWRC as a model, for their combination of XML and a database that has enabled the project's researchers to flexibly represent the information density of their texts with inline tagging and annotation (Brown). The examples of CanWWR and CWRC show that, for websites reliant on literary datasets, the choice of whether to use a relational database or XML is dependant on the type of information the project is working with. Both are well-suited for data storage and retrieval; it is the questions a project wishes to ask of its dataset that makes one choice more appropriate for the needs of the project. The principles of adapting XHTML to HTML5 and CSS to CSS3 represented in this thesis will be equally applicable regardless of storage method, as they each work as a method to serve up data that can be then modified for presentation with markup.

The data I collected on each volume includes standard bibliographic details, supplemented by ISBN, subject-based keywords for use as tags, the presence or absence of marginalia, and general observations such as a handwritten dedication or unique wear. This enabled me to create a representation of each volume sufficient for identification. A full academic version of this project would extend the dataset to include basic analysis of the volumes in relation to one another, such as inferences on what the whole says about the interests of the owner, which volumes appeared to have attracted the most interest, and details suggestive of how they may have read it. As for the marginalia, Mark Woodhouse's article on the marginalia Samuel Clemens wrote in William Prime's book *Tent Life in the Holy Land* offers a model for the treatment of such books. Woodhouse describes what Clemens chose to comment on, the tone thereof, an assessment of Clemens' overarching opinion of the work, and images of the pages in question (Woodhouse); the juxtaposition of multiple books treated thusly via a website would offer further value for academic inquiry.

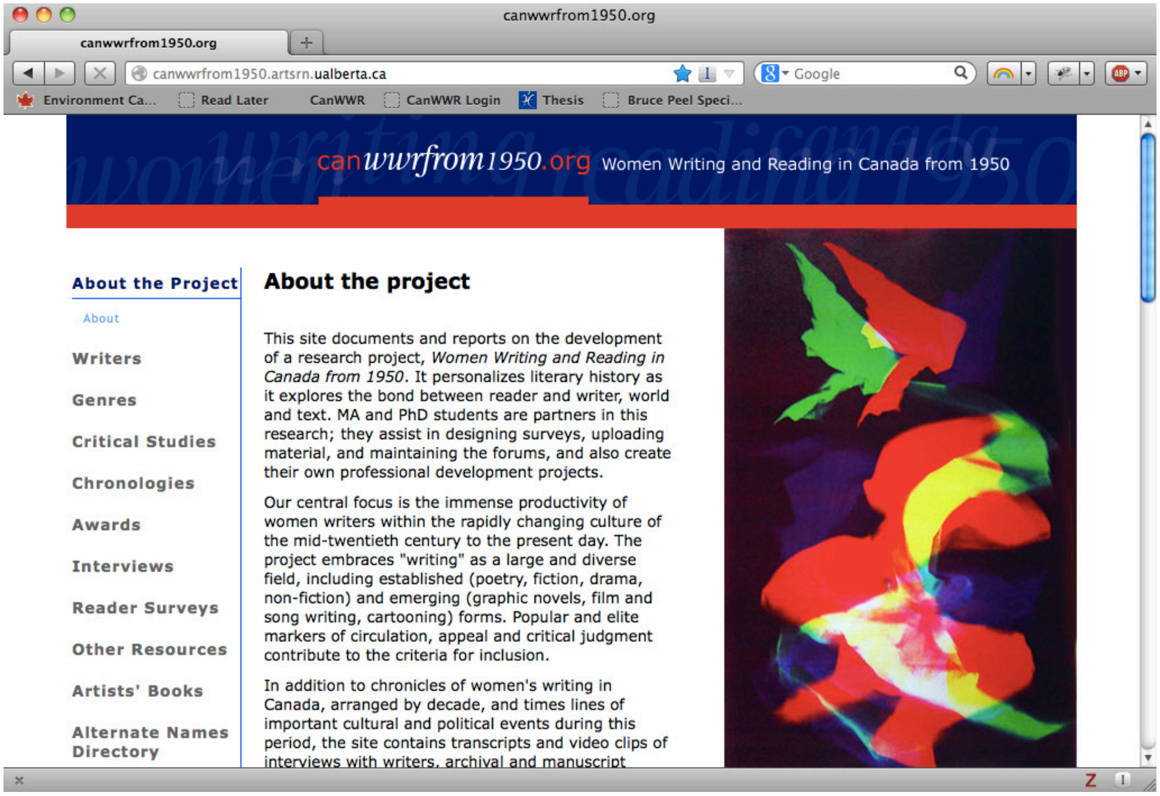
An example of what this could look like is available via the University of Alberta's Bruce Peel Special Collections Library, which recently hosted an exhibit with an accompanying catalogue entitled *The Spacious Margin: Eighteenth-*

Century Printed Books and the Traces of Their Readers. Curated by Sylvia Brown and John Considine as a physical exhibit that ran from October 15, 2012 to February 15, 2013, *The Spacious Margin* showcased a number of volumes from the Bruce Peel collection containing "an array of readerly interactions with books in the form of annotations, improvements, corrections, ornamentation, and suggestive wear and tear" while the catalogue served to "describe and contextualize the notable physical traces of readership and circulation" (Brown and Considine).

As a whole, taking the books I have available at hand and combining them with others' approaches to understanding and presenting similar material provides a substantial dataset to populate my website, and a means of hinting at its place within a framework of related projects centered around the same excavation. The data is, moreover, of a type simple to represent in a relational MySQL database similar to the one underlying CanWWR, and to present in an analogous fashion via pages populated by database queries.

Structuring the Data

Though the focus of my work is on website markup and its impact on tablet users, it is difficult to fully separate a site's data from its markup. Though users of the website will never see the database, how the data is stored directly affects how easily it can be transformed from its stored state to output suitable for formatting with markup. A data-driven website can do this most efficiently by taking care to have a well-structured method of data storage. In the case of a relational database, this entails taking the time to normalize data by breaking it down into smaller units, eliminating duplication and applying logical and consistent connections reduces the chance that the database will require restructuring at a later date (Beighley and Morrison 462). This serves the dual purpose of easing the complexity of the programming required to query and display the data, and



About the project

This site documents and reports on the development of a research project, *Women Writing and Reading in Canada from 1950*. It personalizes literary history as it explores the bond between reader and writer, world and text. MA and PhD students are partners in this research; they assist in designing surveys, uploading material, and maintaining the forums, and also create their own professional development projects.

Our central focus is the immense productivity of women writers within the rapidly changing culture of the mid-twentieth century to the present day. The project embraces "writing" as a large and diverse field, including established (poetry, fiction, drama, non-fiction) and emerging (graphic novels, film and song writing, cartooning) forms. Popular and elite markers of circulation, appeal and critical judgment contribute to the criteria for inclusion.

In addition to chronicles of women's writing in Canada, arranged by decade, and times lines of important cultural and political events during this period, the site contains transcripts and video clips of interviews with writers, archival and manuscript

Figure 2: CanWWR Main Page with Navigation

ensuring that future additions to the dataset as its scope and purpose expands can be easily accommodated without major changes to the existing data and structure. Though users will not be able to infer a website's data structure from a page's source view thanks to the scripting mediating between raw data and final site, that data structure nevertheless influences the overall shape of the website. In an academic project, the database is the reason the website exists in the first place, and the decisions about how to represent that data for storage and processing influence how it's presented for outside interaction. The questions driving the data collection, such as 'which topics are represented on this bookcase', will come out in the final website structure. Taking CanWWR as an example, the sub-pages offered reflect choices to collect critical studies about the represented writers' works, in addition to awards received, and chronological representation of publications. For a project like Dr. Kirstin Uszkalo's WEME, subpages such as the Data Browser reflect an interest in witchcraft documentation's authorship, the

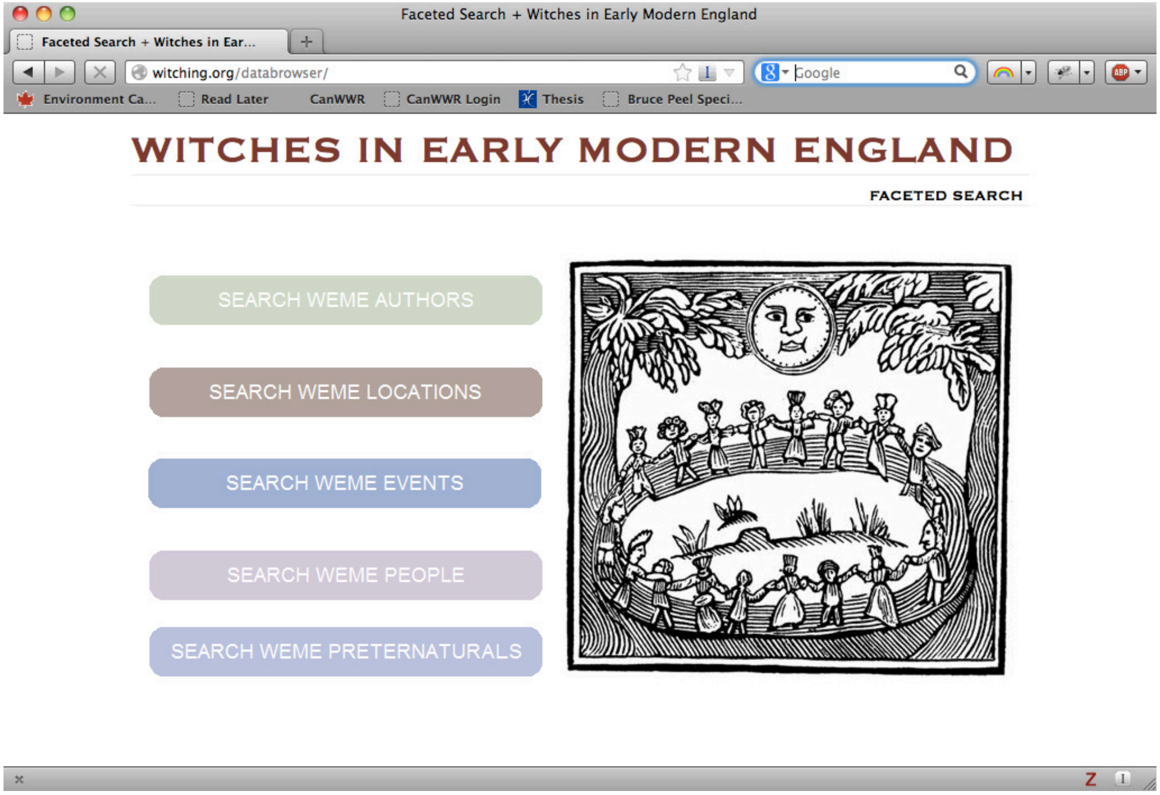


Figure 3: WEME Faceted Search Portal

locations affiliated with events, the people linked to an incident, and the preternaturals alleged to appear. In both CanWWR and WEME's websites, the ability to offer this information is reliant on a data structure underlying the site collecting this information, and forming it to ensure it can be presented in this way. With this in mind, it is necessary to briefly turn to the database to understand the choices behind my site's design.

My dataset required multiple interconnected MySQL tables. It consists of one main table with subsidiary tables mapped back to it either directly or via junction tables, and every line in every table has an auto-incrementing primary key and a timestamp assigned to it. The core table, 'Books', holds the title, format, year of original publication, edition year, publisher name, ISBN and a notes field for researcher commentary. This comprises the information I judged unique to a given volume for this project. It is, for example, not critical to the questions my project asks to know how many books in the collection came from the same

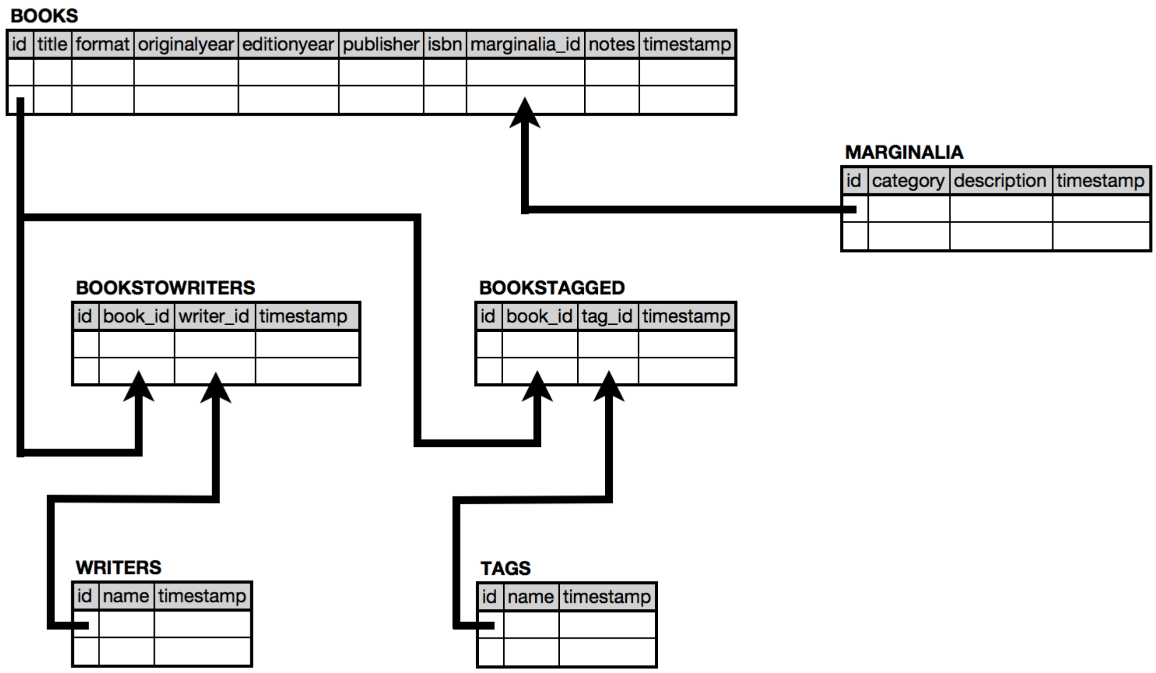

BOOKS
id | title | format | originalyear | editionyear | publisher | isbn | marginalia_id | notes | timestamp
MARGINALIA
id | category | description | timestamp
BOOKSTOWRITERS
id | book_id | writer_id | timestamp
BOOKSTAGGED
id | book_id | tag_id | timestamp
WRITERS
id | name | timestamp
TAGS
id | name | timestamp

Figure 4: Database Structure

publisher, but it is critical to know how many book extant are on the same topic or by the same writer.

The 'Books' table is cross-referenced to several other tables. The 'Marginalia' table is the only one of these mapped directly to it; each entry in 'Books' has a foreign key assigned to it corresponding to one of five lines in 'Marginalia' indicating whether a book has marginalia and to what degree. I required two junction tables, 'Bookstowriters' and 'Bookstagged', to cross-reference entries to subjects and writers, and to associate an entry with multiple subjects or writers. These two tables enable volumes entered in 'Books' to be affiliated with the authors and subject keywords in the 'Writers' and 'Tags' tables. While this same information could have been represented by including additional columns to the core 'Books' table, this practice is generally considered to be not only unwieldy but a contributing factor to poor database scalability (Beighley and Morrison 436). By contrast, splitting out this information reduces duplication and ensures that the expanding the dataset is no more complicated than adding new data and junction tables. As described, my project could expand later to gather additional details about each volume. At that point, it could become important to identify which bookcase or shelf a volume is located on, or what is next to it, or further details about the marginalia within it. For each category of new information, a new data table and junction table is all that is required to add it to the database.

This approach ensures that the database's structure can grow as the project does with a minimum of disruption to the existing tables, a strategy important for not only scalability but the long-term sustainability of the dataset in its present form. The result is a database containing details about individual volumes that I can query to correlate books to authors, information about marginalia, and subject-based keywords. These correlations provided the basis for the website's pages. Similarly, the attention to scalability built into the database has a direct impact on how the website could later expand to match additions to the data set.

Structuring the Initial Site with PHP Includes, XHTML and CSS

With the MySQL database in place, I could then use PHP scripts to query my data and reshape it into a format compatible with display on a webpage.[16] I relied on a handful of PHP's native MySQL functions to open the connection to the database, pass SQL queries to the server, store the results in PHP array variables, and then process the arrays to simultaneously format the data and add the appropriate XHTML elements. For each specific database query, I planned a page in my website to display the results. These pages, such as 'Bookshelf,' are generated by a unique series of PHP scripts that submit queries to the database, process the results and transform them into discrete chunks wrapped by XHTML elements.

The combination of data, database and queries form the structure over which the website as a whole could be constructed. The site's page structure combines the example set by CanWWR's page structure and the recommendations from Lynn Beighley and Michael Morrison's book *Head First PHP & MySQL*. All of CanWWR's pages contain at bare minimum a header banner image at the top of the site and a left-hand navigational column with links to the various parts of the

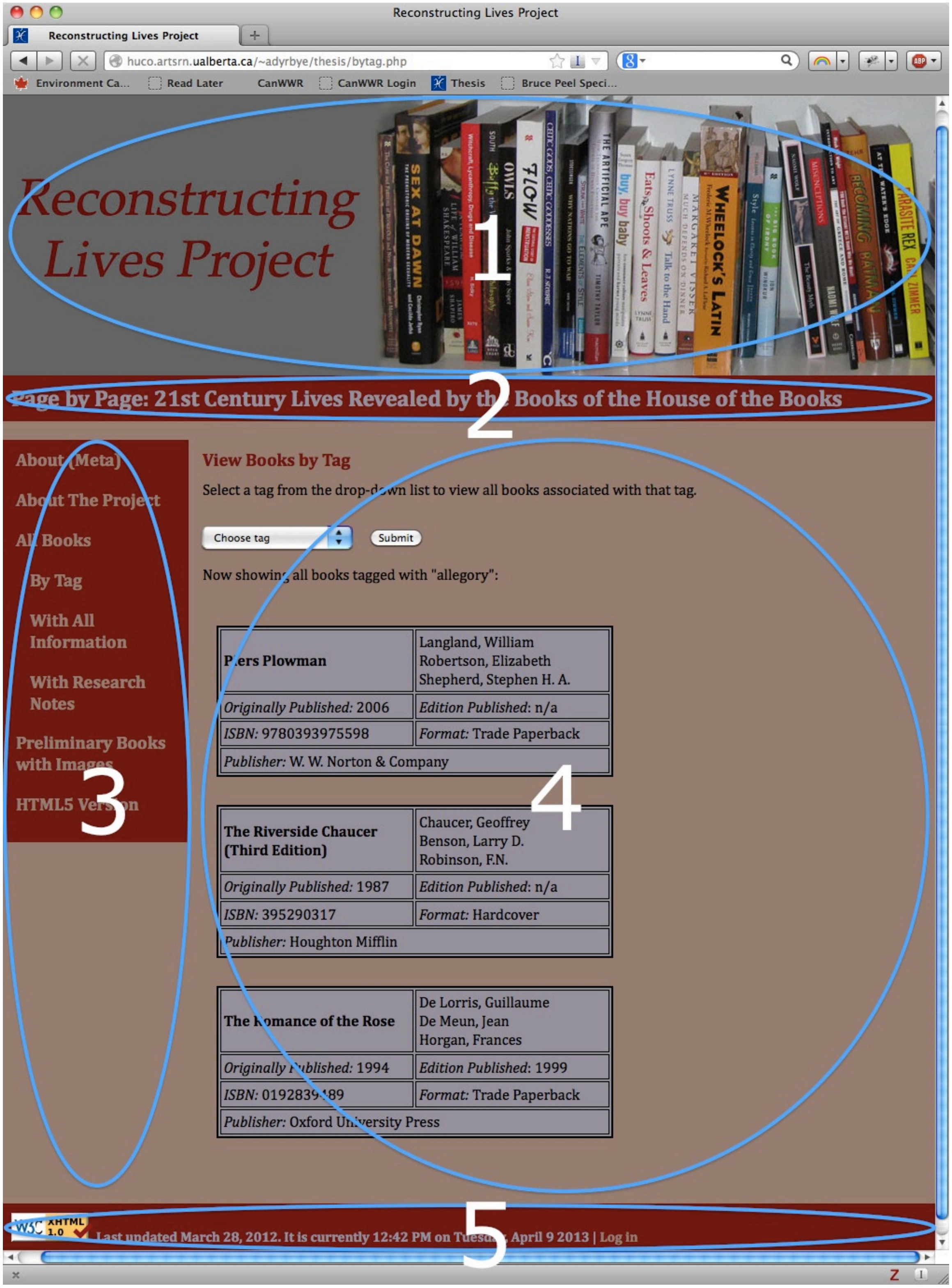
[16] See Appendix One, section bookshelf.php.

site. The main page also has a discrete footer, consisting of a graphic declaring copyright, the date range of the project, and the site URL. A right-hand column displaying a featured piece of artwork and an artist's statement is included on the main page, 'Reader Surveys', 'Artists' Books' and 'Other Resources' pages, while all other pages maintain a two-column layout. This influenced the decision to implement a two-column layout with a discrete header and footer throughout the site.

Beighley and Morrison were particularly helpful as a guide for implementing my site's design. They recommend splitting every component that carries over from page to page off into a reusable template (Beighley and Morrison 422). This highly practical approach to design allows each page to be stitched together from multiple consistent sources via PHP `include` statements, reducing the effort needed to create and maintain pages with unique content while ensuring a unified aesthetic across the site (422). I composed each section of my website I wanted to carry over across pages of PHP scripts interspersed with XHTML markup, to enable sections to be called with `include` statements as needed in the appropriate order on each unique page. I split out the head of the page, left navigational column and foot of the page in this fashion.[17] Designing these first and keeping them separate not only reduced duplication, but freed my efforts to focus more on the unique page content later on by eliminating the need to make tedious systematic page-by-page alterations of these sections as I applied modifications to the site.

From the perspective of good page structure, the header and footer files are the most important. The header file contains all the critical early XHTML tags, such as the opening `<html>` tag, the document type declaration, the `<head>` and `<title>` tagsets, and the opening `<body>` tag. Information about the character set is also found here, as is the link to the site's stylesheet. This is also the most

[17] See Appendix One, sections thesisheader.php, sidebar.php home.php and thesisheader.php.



Page by Page: 21st Century Lives Revealed by the Books of the House of the Books

View Books by Tag

Select a tag from the drop-down list to view all books associated with that tag.

Now showing all books tagged with "allegory":

Piers Plowman	Langland, William Robertson, Elizabeth Shepherd, Stephen H. A.
Originally Published: 2006	*Edition Published*: n/a
ISBN: 9780393975598	*Format:* Trade Paperback
Publisher: W. W. Norton & Company	

The Riverside Chaucer (Third Edition)	Chaucer, Geoffrey Benson, Larry D. Robinson, F.N.
Originally Published: 1987	*Edition Published*: n/a
ISBN: 395290317	*Format:* Hardcover
Publisher: Houghton Mifflin	

The Romance of the Rose	De Lorris, Guillaume De Meun, Jean Horgan, Frances
Originally Published: 1994	*Edition Published*: 1999
ISBN: 0192839489	*Format:* Trade Paperback
Publisher: Oxford University Press	

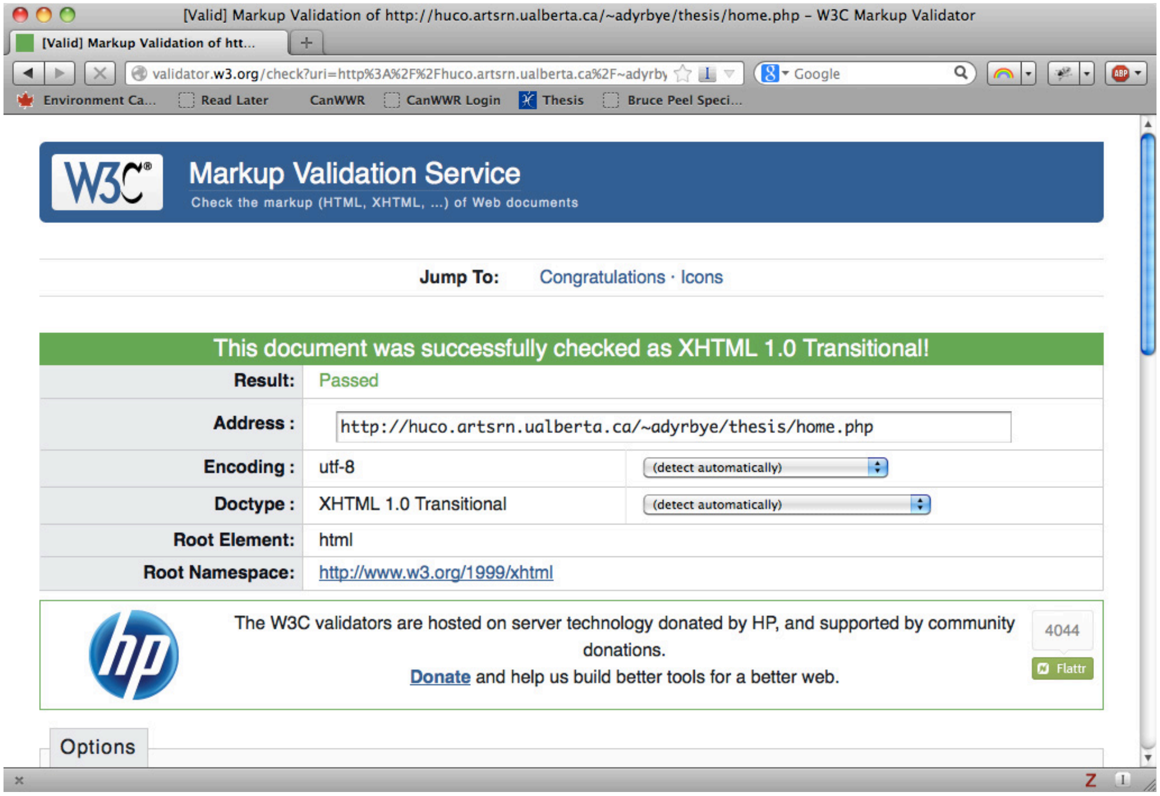
Last updated March 28, 2012. It is currently 12:42 PM on Tuesday April 9 2013 | Log in

Figure 5: Project Website with Structure

1) Banner image 2) Header 3) Navigational sidebar 4) Main body column 5) Footer convenient place to embed the PHP script opening a connection to the MySQL database, as this will be needed on the majority of the site's pages. Conversely, the file containing the footer is used to close the `</body>` and `</html>` and is the most logical place for a PHP function closing the MySQL database connection, as these too must carry over to every page of the site. I have also used the footer to communicate basic page information to visitors, such as W3C compliance via their validation icon, a 'page last updated' statement, and the current date and time as generated by a simple PHP script. Lastly, I have included a discrete link to the mocked up login page.

The last piece split off into a separate file is the left navigation column, containing the links to the various pages within the site. This is the only modular section of the website that does not contain any PHP scripts, as its only purpose is to provide users with links to other parts of the site. I quickly realized that keeping this section in a separate file is particularly beneficial to the maintenance of a growing website. As I added to and modified individual pages, the links to and descriptions of them needed to change in the navigation column to match. From the example of CanWWR, which has added pages for critical studies, interviews, surveys, an artists' books exhibit and an alternate name directory since its initial implementation (Dyrbye), such additions and alterations can be expected to recur over the lifetime of a project.

With the modular structure, any changes I make to the head, foot or navigation column automatically carry over across all affected pages, without having to make the same change to each page. These changes can be updates to the markup, modifications to the programming scripts, updating links to direct to a new stylesheet, updates to the page text and so on. As a strategy, it minimizes redundancies before they can become an issue for a dynamically growing project. As the project grows in depth and the website grows to reflect it, there may be dozens or hundreds of pages each requiring the same information to be displayed on each, and all requiring update from time to time; compartmentalizing these areas reduces the maintenance load and effort required to add new pages.


[Valid] Markup Validation of http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home.php - W3C Markup Validator
[Valid] Markup Validation of htt...
validator.w3.org/check?uri=http%3A%2F%2Fhuco.artsrn.ualberta.ca%2F~adyrby
Google
Environment Ca...
Read Later
CanWWR
CanWWR Login
Thesis
Bruce Peel Speci...
W3C®

Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Congratulations · Icons

This document was successfully checked as XHTML 1.0 Transitional!

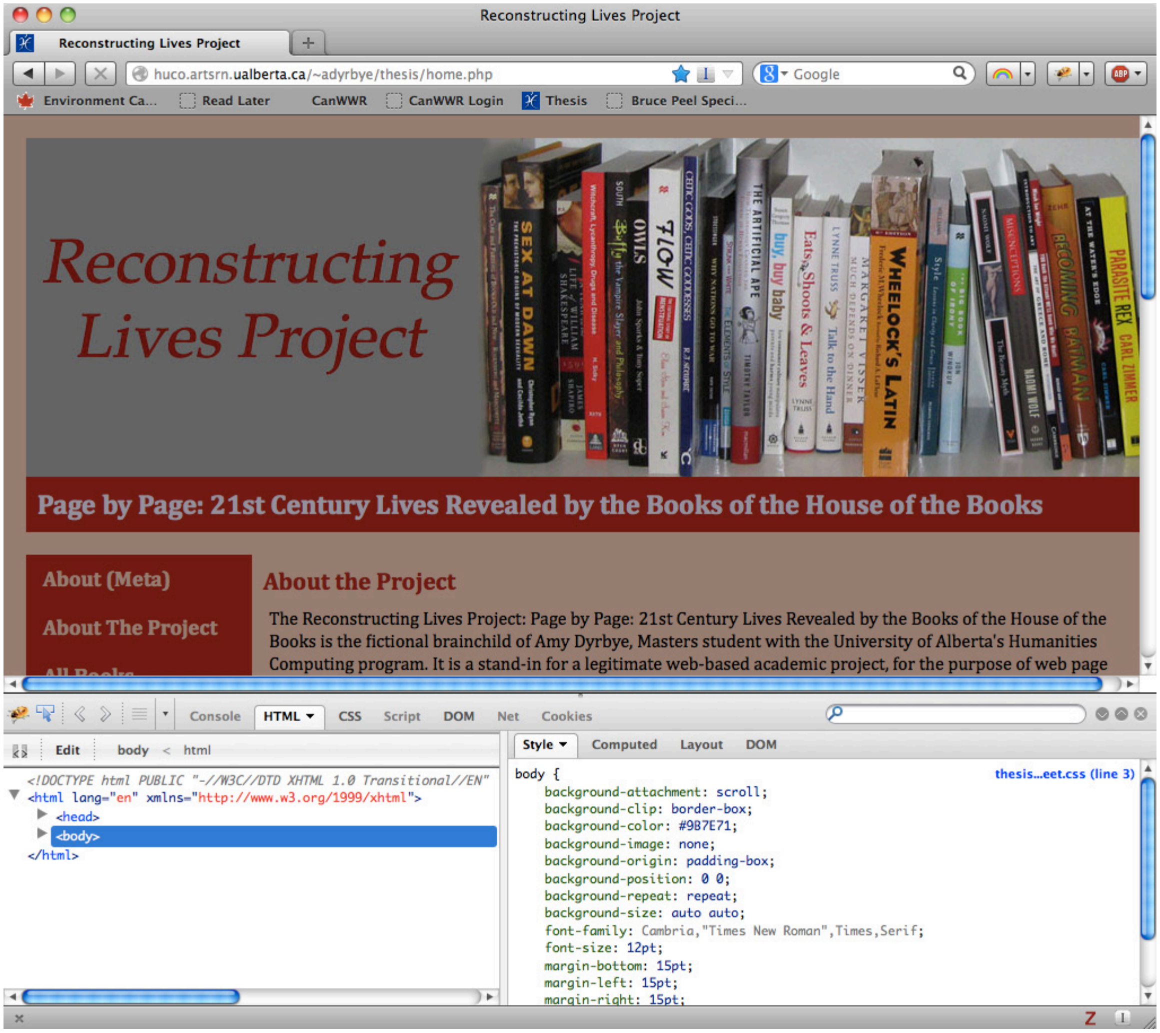
Result:	Passed	
Address :	http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home.php	
Encoding :	utf-8	(detect automatically)
Doctype :	XHTML 1.0 Transitional	(detect automatically)
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

The W3C validators are hosted on server technology donated by HP, and supported by community donations.
Donate and help us build better tools for a better web.

Figure 6: W3C Markup Validation Service

Implementing a modular structure at the outset ensures any updates that must carry across the whole site can be effected easily. This extends to structural changes, such as changing to a new markup language, enabling the site designer to streamline the process. Designing its website to include a modular structure is therefore a way in which a project can plan for updates and alterations in the site's future, regardless of what shape that future takes.

With the modular framework established, I could implement the parts of the website representing the data from specific database queries. I marked up all content in keeping with the XHTML 1.0 Transitional standard, and tested each page via the W3C’s Markup Validation Service before moving on to the next. The W3C's Markup Validation Service is a free, publicly accessible tool for testing web pages or documents containing markup for compliance with a variety of W3C-supported standards, including XHTML 1.0 Transitional, and provides a detailed report of problematic elements (W3C, *Validation Service*). I used the



Reconstructing Lives Project

Page by Page: 21st Century Lives Revealed by the Books of the House of the Books

About (Meta)

About The Project

All Books

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
  <head>
  <body>
</html>
```

```
body {
    background-attachment: scroll;
    background-clip: border-box;
    background-color: #9B7E71;
    background-image: none;
    background-origin: padding-box;
    background-position: 0 0;
    background-repeat: repeat;
    background-size: auto auto;
    font-family: Cambria,"Times New Roman",Times,Serif;
    font-size: 12pt;
    margin-bottom: 15pt;
    margin-left: 15pt;
    margin-right: 15pt;
```

Figure 7: Firebug Panel in CSS View

Markup Validation Service in conjunction with the Firebug plugin for the Firefox browser, which provides real-time feedback on markup elements and their applicable CSS when viewing a webpage (Mozilla). I leveraged this combination to locate problems and bring the first iteration of the site markup into full compliance with the XHTML 1.0 Transitional standard.

For each page with content drawn from my database, I nested the data within the `<div id="main">` tagset. The markup I used for this content was chosen to integrate common website elements into my pages, so that I could test their behaviour to determine through which display erratically or become challenging to use effectively when the site is viewed on an iPad. As such, individual pages include images, tables containing database information of a variety of lengths, tables including both images and textual data, unordered lists, and a variety of form elements such as text entry fields, drop selectors and radio buttons. At this stage, I anticipated tables, forms and text size to be those aspects most likely to display erratically, appear at a size too small for comfortable reading, or become difficult to interact following conversion to HTML5 when viewed on a tablet computer. All of these elements are found in use by actual academic project websites, as can be seen by examining the web browser view source data of CanWWR's website in addition to other projects' sites such as the *Text Analysis Portal for Research* (TAPoR) 2.0 (Rockwell et al.) or *Witches in Early Modern England* (WEME) (Uszkalo).

Following the example set by all three of these projects, I supplemented the 'public' website representing project data and findings with a mockup of a password protected 'back end' containing data entry forms. This public/private structure is critical to projects reliant on the efforts of research assistants to contribute new data without the experience or need to do so with direct database access; it is also a major function of a data-driven research site that must carry over across devices according to the needs of the project's primary researcher. A protected section of a website is important wherever there is the potential for users to access project data. It is dangerous to assume all visitors are trustworthy, and taking steps to secure that access is critical to ensuring data integrity (Beighley and Morrison 295). This can be accomplished in a number of ways, including but not limited to adding password protection to select pages, implementing a user registration system with administrative privileges assigned only to specific accounts, permitting administrative privileges only to visitors from a specific computer's IP address (Beighley and Morrison 298) or combinations thereof.

Following the examples of CanWWR and TAPoR, I added an administrative login page to my site, and a data entry page that could only be accessed by successfully entering a username and password at the login prompt. To keep my focus on the behaviour of the markup elements, I elected not to implement a fully functional data entry structure. Instead, I designed the mockup solely to test the behaviour of various form elements in XHTML, and then later in HTML5 and on a tablet. A user's ability to successfully interact with form elements is in this case independent of their connection to a database, thereby negating the need for a genuine password-entry structure fully connected to the database.

Once I had populated the individual pages with database content marked up according to XHTML 1.0 Transitional, I added formatting to each element with a single CSS stylesheet.[18] This stylesheet contains the page's colour scheme, font and text sizes, adds the banner image, implements margins, and arranges the layout into two columns with a discrete header and footer. I carried the same stylesheet across pages to contribute visual unification to the site. Its completion comprised the last piece in the first stage of the mock project site and its conversion.

Situating Tablets

At this point, I had created a fully functional website, integrating a relational MySQL database, W3C compliant XHTML 1.0 Transitional markup, clear page structure and unified styling -- everything I required to convert it to HTML5 and CSS3, and to conduct basic testing on a non-desktop or laptop device. I chose a second-generation Apple iPad as the alternate device, due to the iPad's status as the first tablet of its class, the market dominance Apple has enjoyed and the role Apple has played in the adoption of HTML5 and CSS3 as web standards to date. This decision was reinforced by the availability of device-specific guides such as Warner, LaFontaine, and Andron’s *iPhone & iPad Web Design for Dummies*, and the focus on iPads in particular in papers such as Czapracki and Burrows' "Bringing Students and Faculty Together Through Mobile Devices." While the iPad remains effectively synonymous with ‘tablet computer,’ it remains the best model of its class for design purposes.

[18] See Appendix Three.

Here, the iPad stands in not only for tablets as a class, but for the wide variety of alternate web devices that have emerged and enjoyed wide adoption in recent years. Alternate web devices will only increase in prominence, exerting new pressures on the web paradigm both in terms of designing web content to accommodate them and in terms of how users relate to web content. Although the long-term impact of tablets on desktop or laptop computer use is still developing, Czapracki and Burrows show that many users have already shown a strong interest in integrating tablets into their routine activities, demonstrating their emergence as a distinct niche within the web paradigm. As I have seen with projects like CanWWR, TAPoR, CWRC and WEME, long-term academic projects now commonly integrate a web presence, whether a simple descriptive webpage detailing the scope and chief investigators, or an elaborate site built to disseminate data and findings. For projects heavily invested in their web presence, changes to web technologies can have a measureable long-term impact over their lifetimes.

As for projects' audiences, it has additionally become common to observe academics integrating newer devices like tablet computers in their daily lives, a tangible manifestation of the paradigm shift underway. As the level of alternate web device adoption increases among projects and users, the need to convert academic project websites to web standards that can flexibly accommodate those users will become more urgent. Both tablets and mobile phones have achieved sufficient ubiquity amongst academic staff that university technical teams are concerned with adequately supporting their users (Czapracki and Burrows 215), though formal studies are still needed to determine a more precise measure of adoption. The identified need for formal support corroborates informal observation. At the University of Alberta, the online help for the wireless network authentication system now includes directions for Apple's iOS in addition to Google's Android operating system for versions 2.2 and 4.0/4.1, indicating the Academic Information and Communication Technologies (AICT) department is aware of smartphone and tablet users among their user base (Academic Information and Communication Technologies). Where project websites are concerned, this lends support to the likelihood that tablet users accustomed to using their device in the classroom and in their regular research activities are also employing these devices to access their own and others' project websites.

Despite clear documentation of tablet computer usage among academic staff, little has been published advocating for academic projects' websites to convert to HTML5 and the closely related update to CSS, CSS3, or other standards that can accommodate tablet users. These issues directly interrelate. Updating to the new web standards facilitates integrating tablet-ready versions of the same pages thanks to features such as the ability to have multiple stylesheets for the same site, each targeted to and optimized for particular devices, included in CSS3 (Warner, LaFontaine, and Andron 120). That said, HTML5 and CSS3 have only been a concern for web developers for a few short years; the web paradigm they represent is at best in mid-shift. The distribution is still a mix of early adopters, cautious observers and the blissfully unaware. A check of page source data for a few major websites on November 24 of 2012 shows Apple's official webpage to be HTML5 compliant, [19] *Wired* to be using a combination of XHTML 1.0 Transitional and Javascript for device detection, [20] while the CBC maintains a combination of XHTML 1.0 Transitional and an HTML5-based mobile site aimed at smartphone users.[21] This represents a small selection of the strategies used to reach websites' user bases currently employed across the web. With such a variety of competitors in play, HTML5 requires a closer look if it is to be successfully promoted as a solution.

There is a common driving force behind the divergent strategies used by prominent websites: the users. All wish to keep their site accessible across platforms, but there is no clear consensus to be found in how they have done so. Yet, all recognize that in order to remain relevant, a website has to remain mindful of how its users are accessing their content. This ties directly to a site's

[19] http://www.apple.com/
[20] http://www.wired.com/
[21] See http://www.cbc.ca/ and http://www.cbc.ca/m/info.html respectively.

usability, which is an attribute that determines whether "people who use the [website] can do so quickly and easily to accomplish their own tasks" (Dumas and Redish 4). Usability is, however, separate from functionality. The VCR was once a prime example of this, as "VCRs may have high functionality (the feature works as it was designed to work), but they have low usability (people cannot use them quickly and easily to accomplish their tasks)" (Dumas and Redish 5). Websites that do not take into account the technologies used to access them risk frustrating rather than assisting their users. This is why "[u]sers, not designers or developers, determine when a product is easy to use" (5).

The need to ensure usability is not confined to new services and businesses. The interest in tablet computers amongst academics noted by technological support staff like Czapracki and Burrows indicates that academic projects cannot assume their peers will use desktop or laptop computers to access project web sites. I suggest that HTML5 offers an elegant solution to the problem, one situated from its inception as well suited for tablet computers, and integrating native functions advantageous over XHTML.

Chapter 4 - This Changes Everything: Tablets, HTML5 and the Emerging Web Environment

Apple, the iPad and the Promotion of HTML5

As a class of device, tablets have had a direct role in promoting HTML5 as an important new web markup language. Though not solely responsible, they have been the most visible HTML5-optimized device. Apple's vocal stance in favour of HTML5, coinciding with their iPad's release, played no small part in its heightened profile. The novelty of both tablets as a class and HTML5 as a markup language was intensified when Steve Jobs spoke out in favour of HTML5's open standard over proprietary plugin software like Flash Player (Jobs). With Apple's iPhone still holding about 32% of the American smartphone market (Reed) and the iPad continuing to lead the tablet market with a share just shy of 70% (Fingas), Apple's support for HTML5 carries considerable weight even now:

> Though the operating system for the iPhone, iPod and iPad is proprietary, we strongly believe that all standards pertaining to the web should be open. Rather than use Flash, Apple has adopted HTML5, CSS and JavaScript – all open standards. Apple's mobile devices all ship with high performance, low power implementations of these open standards. HTML5, the new web standard that has been adopted by Apple, Google and many others, lets web developers create advanced graphics, typography, animations and transitions without relying on third party browser plug-ins (like Flash). HTML5 is completely open and controlled by a standards committee, of which Apple is a member. (Jobs)

With this statement, Jobs declared HTML5 to be the markup language for the smartphone and tablet markets and elevated it from one of many versions of HTML-based web markup to a full-on web standard, while spinning Apple's rejection of Flash as an advantage. This set a clear tone for web developers working within the new web environment. Furthermore, Jobs' assertion of HTML5 as a standard is not simply a rhetorical device to lend authority. The body responsible for HTML5's development, the Web Hypertext Application Technology Working Group (WHATWG), has declared HTML5 a living standard and maintains the authoritative description on its site (Hickson).

Apple's choice of HTML5 as the web language of choice for its suite of iOS products[22] situates HTML5 as the means of reconciling new hand-held devices with existing web-based activities. The iPad debuted in the midst of a steady stream of new web-ready devices, ranging from miniaturized, inexpensive laptops marketed as 'netbooks', to gaming consoles, to smartphones more computer than phone. Interest in designing websites tailored to such devices prefigured the iPad and extended beyond the smartphone, evident in the response to the introduction of the Nintendo Wii gaming console's Internet Channel.[23] With the Wii, as now with the iPad, researchers noted that it has "web surfing and other communication facilities that open new perspectives to facilitate the collaboration among people," but that the consoles "have some limitations respect [sic] to a personal computer, in particular the screen resolution, that impact the access to the information through the Web" (Franzoni et al. 105). The accumulation of devices such as web-enabled gaming consoles, smartphones and tablets have incrementally contributed to the drive toward a new web paradigm invested in augmenting their strengths, as their increased variety and degree of consumer adoption have tipped the inadequacies of prior approaches to web design into crisis. The iPad, and tablets as a class, are just the most visible example of a longstanding trend, while HTML5 and its open community of developers offer an approach to markup capable of integrating new devices above prior web markup.

The paradigm shift toward mobile computing and ubiquitous web access has been percolating for years. It appears a recent thing in no small part because "[t]he resultant pool of facts contains those accessible to casual observation and experiment" with the details of the discipline-specific developments that make those observations possible remaining obscure to that casual observer (Kuhn 15).

[22] The iPad, iPhone and iPod Touch.

[23] http://www.nintendo.com/wii/built-in-entertainment/#/internet-channel

The development of HTML5 is a prime example. As a markup language, HTML5 has roots going back to 2004, when the participants of the W3C's Workshop on Web Applications and Compound Documents first laid out the seven guiding principals for a successor to HTML 4 (Pilgrim 11). These principals identified the most critical requirements for the future of HTML, addressing many of HTML 4.01's and XHTML's shortcomings. This included backwards compatibility with a clear migration path, well-defined error handling, a clearly identified practical use for each feature, integration of scripting, the avoidance of device-specific profiling, and an open development process (Pilgrim 11-12).

At the same time, the Web Hypertext Applications Technology (WHAT) Working Group, an unofficial and open collaboration of web browser developers and other interested parties, was working on extending HTML 4's forms while maintaining backwards compatibility (Pilgrim 12). The WHAT Working Group sought to make XForms, a forms type used with XHTML reliant on its strict error handling, compatible with HTML by documenting browsers' 'forgiving' error handling methods (Pilgrim 13). They used this documentation to successfully create a method for parsing HTML compatible with existing web content (13). WHAT Working Group also developed new controls for HTML forms, in addition to new features such as a native drawing canvas and audio and video support (13), all of which ultimately found a home in HTML5 when the W3C's HTML Working Group and the WHAT Working Group agreed to pool their efforts in 2006 (Pilgrim 13-14). This collaboration, taking the form of a rechartered W3C HTML Working Group, formally named their collective work 'HTML5' (Pilgrim 14).

Their work paid off, culminating in the formal 2010 debut of HTML5 with the publication of guides such as Mark Pilgrim’s *HTML5: Up and Running*. Pilgrim describes it as a specification encompassing the features necessary for integrating web applications into a site, in addition to and standardizing features that not been vetted under XHTML or HTML 4 (Pilgrim ix). Most modern web browsers, such as Apple's Safari, Google's Chrome, Mozilla's Firefox and Opera, now support many HTML5 features, as do the mobile browsers included with Android devices and Apple's iPhone and iPad (ix). In acknowledgement of the many users of older and otherwise non-compliant web browsers, HTML5 is designed for a high level of backwards compatibility (Pilgrim x). It has far more to offer than its predecessors, integrating such features as semantic elements for headers, footers and sections, a drawing surface compatible with JavaScript, direct video embedding, geolocation and persistent local data storage (Pilgrim ix). As a whole, it offers a direct means to accomplish tasks that previously required unofficial features or third-party software.

CSS3, a similarly new cascading style sheets specification popularly associated with HTML5, is a mix of standardized and experimental specifications; most modern browsers have now integrated support for the standardized specifications and may or may not support the experimental ones (Gasston 1). Peter Gasston's *The Book of CSS3* describes it as a set of specifications under active development, with a large community of committed developers creating new features and ensuring their support within web browsers (Gasston 1.1.1). As a modular specification, it enables web developers to integrate features as they are completed without having to wait for CSS3 to be released in its entirety (Gasston 1.1.3). Web developers do not need to update their markup to HTML5 to take advantage of CSS3's visual effects, which include rotations, scaling, animation, text effects such as shadows and box element effects such as gradients and rounded corners (Gasston 1.1.4). It has particular advantages for websites integrating images, as HTML elements may now have multiple images assigned to them, and they can be resized at will rather than requiring visitors to download the same image once per size (Gasston 8.1).

Together, HTML5 and CSS3 offer a complementary set of web markup and styling features all in one place, with particular strengths well suited to web design for tablets. Resources such as Janine Warner, David LaFontaine and Lee Andron's *iPhone and iPad Web Design for Dummies* therefore emphasize these newest web standards and offer guidance on how to optimize web pages for viewing on tablets while maintaining the site's desktop experience (Warner, LaFontaine, and Andron 12). Though specific-purpose apps receive a lot of attention, the authors emphasize that "[n]ot everyone needs an app, but every website on the Internet should be mobile friendly and designed to look good to the growing audience of web surfers on the iPhone and iPad" (Warner, LaFontaine, and Andron 14). This applies to academic projects' websites as much as to sites for general consumers. They argue that tablets are more critical to focus on than mobile phones, for "[t]hough early usage stats vary widely, the bigger screen of the iPad has resulted in users surfing 2½ times more often than iPhone users" (Warner, LaFontaine, and Andron 15). Existing sites created to older HTML or XHTML standards with CSS are strong candidates for to HTML5 and CSS3, as the same principals of separation of content from design apply (Warner, LaFontaine, and Andron 22). Integrating CSS3 is particularly key for maintaining both tablet and desktop ready websites, as its media query function not only allows multiple stylesheets for the same webpage, but enables each one to be targeted to a particular device (Warner, LaFontaine, and Andron 120).

HTML5 and CSS3 have become a major part of the web specification landscape. Jobs' boost to HTML5 in particular proved a boon to its debut, but his endorsement would not have had the same degree of impact had smartphones and tablets not captured the consumer imagination to the degree they have. These types of device have achieved a clear ubiquity on the consumer market. As adopters become more accustomed to relying on hand-held computer devices to stay connected within their daily lives, they correspondingly expect to be able to use their devices in a variety of situations. These users have become a subset web-based projects can ill afford to ignore to best serve their audiences. Consumers' interest in new web-ready devices reinforces the need for web infrastructure that supports their choices. Their activities provide the pool of facts that developers require to determine the shape of the emerging paradigm. In Kuhn's view,

> early fact gathering is usually restricted to the wealth of data that lies at hand. The resulting pool of facts contains those accessible to casual observation and experiment together with some of the more esoteric data

> retrievable from established crafts like medicine, calendar making, and metallurgy. Because the crafts are one readily accessible source of facts that could not have been causally discovered, technology has often played a vital role in the emergence of new sciences. (Kuhn 15-16)

In this case, the established crafts of computer science, computer hardware development and web design have combined to produce mobile web-ready devices, while users' activities have determined their rate of adoption and applications. The interaction between the crafts and the users' application of their products is the driving force behind the web paradigm shift.

Mobile devices have already had a measureable impact on their users' lives and expectations. The phenomenon of the social media 'backchannel' is a well-documented example. The 'backchannel' is a form of social participation in a public event such as a conference, political convention, sports event and so on in which "comments from potentially many thousands of individuals participating directly or remotely in an event can be made immediately visible and searchable" via Twitter or Facebook (Hansen, Smith, and Schneiderman 1). In particular, Twitter activity during conferences is followed "both by people attending the event and those monitoring it from afar" (Dörk et al. 1136). Mobile devices directly facilitate in the moment social media activity among event participants (Hansen, Smith, and Schneiderman 1).

This phenomenon is an example of a change to the technological paradigm triggering a related shift where it overlaps with its users' social paradigm. The adoption of personal mobile devices combined with the growing ubiquity of web access has moved the integration of social media activities such as live-tweeting and live-blogging into academic events from novelty to accepted practice. Social media is sufficiently integrated into the social landscape that users' focus is now on working out the etiquette to govern it. The issues surrounding social media at conferences are now loaded with questions of attribution, accurate representation, the benefits and pitfalls of the openness inherent to broadcasting the event in this fashion and more:

> As social media usage becomes a common feature at conferences, anxieties about authority, control, attribution, originality and privacy are likely to haunt the theory and practice of scholarly social media use for some time to come. But embracing openness and transparency, through Twitter or blogging, can coexist with the existing expectations for academic rigor, ethics and civil behaviour.
>
> Academics who are comfortable with and enthusiastic about social media need to communicate that live-tweeting at conferences is not a threat to scholarly activity but the contrary: an ally. (Priego)

This conversation is a direct consequence of the ubiquity of portable web-ready devices within the academic population, and demonstrates the importance of the constant web access they enable to many participants in the current academic culture.

The combination of the variety of devices now in popular use, the expansion of what it means to use the internet and the years-long process behind the release of HTML5 is strongly suggestive that there are a number of closely related processes at work. The emergence of HTML5 as a component of the web paradigm extends established practices more so than it offers an entirely new method of web design. Many of the assumptions web developers could once make when designing a site -- such as that their users would access web content from a desktop computer, would use a mouse to interact with it, that including multimedia required users to install third-party software and so forth -- no longer hold true. User expectations, the variety of devices from which to access computer-based services and the available web development options have all changed. Users supplement their home or work computers with portable devices, touchscreens interfaces are common, and HTML5's native multimedia support has replaced Flash for serving content to iOS devices. Many of the change to the web and access to it stem from the desire to encompass new means of accessing web content under existing web standards, and to expand the integration of web access into everyday life, a realization of Kuhn's observation that "research under a paradigm must be a particularly effective way to induce paradigm change" (Kuhn 52).

Furthermore, developments of this sort are directly analogous to Kuhn's conception of discoveries as "not isolated events but extended episodes...commenc[ing] with the awareness of anomaly, i.e., with the recognition that nature has somehow violated the paradigm-induced expectations that govern normal science" (Kuhn 52-53). The variety of devices now available, the methods developers have devised to serve content to these devices, how users access content, and what kind of content users tend to access (such as e-mail versus web search versus social media) are just a few of the obvious ways the web paradigm has developed anomalies in recent years. XHTML relies on the assumption that web content will be accessed in a consistent set of circumstances, namely at a computer terminal with a mouse and keyboard, with reliable commonalities to the operating system, software compatibilities and so forth. The variations to interface, operating system and capabilities found in smartphones alone -- consider the Blackberry versus the iPhone versus a Samsung device running Google's Android operating system -- challenges the supremacy of these assumptions and the web practices supporting them.

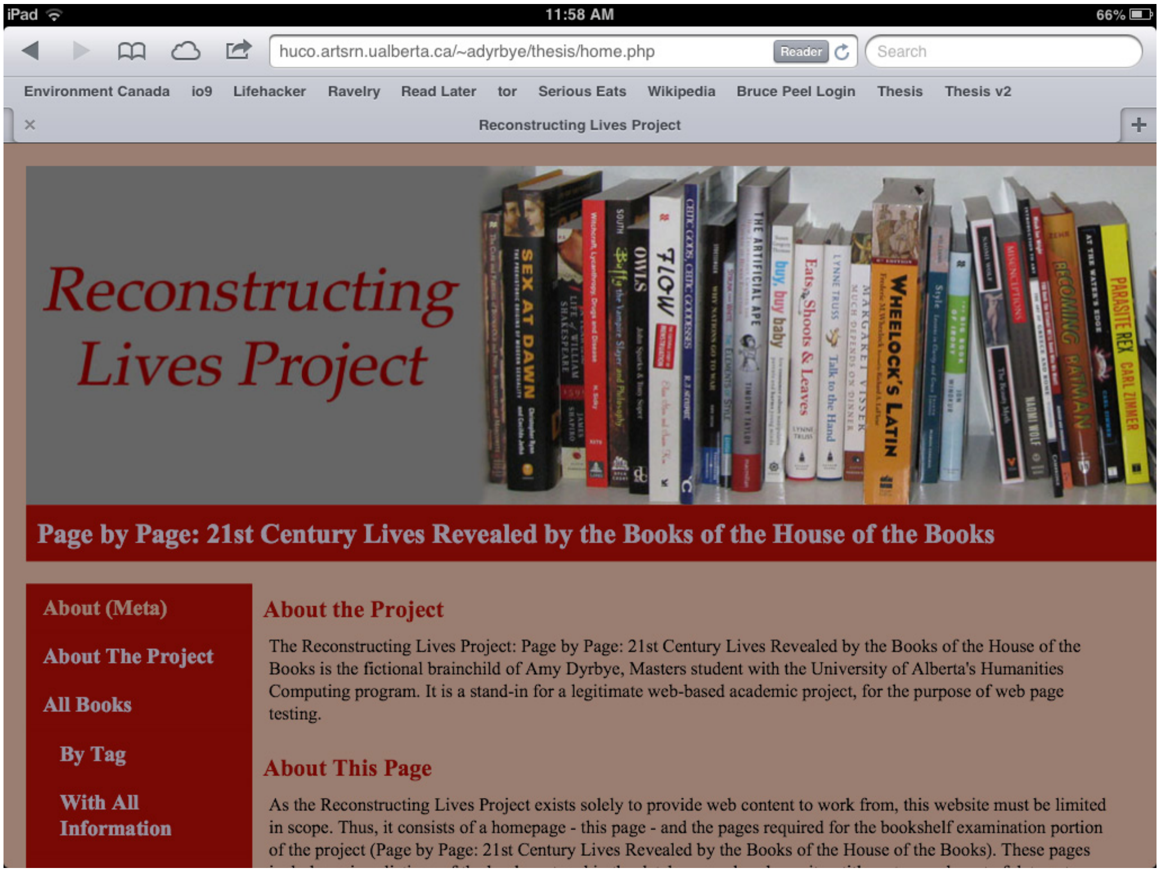
Though sites built using widely accepted web languages -- HTML 4.01, XHTML, CSS and so on -- can be viewed in a limited fashion through many of these devices, the experience is far from ideal. Even in the case of tablet browsers designed to accommodate content presented via older versions of web markup, there is only so much it can do when asked to render a site designed for a viewing on a large screen, assuming a mouse interface, and integrating content reliant on users downloading third-party applications such as Flash. The ability to access the web without being anchored to a desktop or laptop computer has changed user expectations, driving the need for a new approach to web design and enabling the paradigm to shift into something at once immediately recognizable and markedly different. This process makes another step from one paradigm to another with every user who chooses a tablet over a laptop for mobile computing, and every website that makes a conscious effort to accommodate these users.

XHTML, Tablets and the Display Space in Between

Apple's championing of HTML5 effectively established tablets and smartphones as integral to HTML5's web standard, but it does little to explain how and why the various flavours of HTML 4.01 and XHTML fall short. Yet, this information is crucial to building an HTML5 site that truly improves on the prior standard. Without it, developers such as myself risk reproducing the same issues under a different standard. With this in mind and the XHTML version of my website fully in place, my next logical step was testing its behaviour on a tablet computer, and integrating my findings into the HTML5 iteration of the site.

I utilized an iPad 2 for the testing process, and kept my testing confined to looking for common issues noted in tablet web design best practices guides. This included whether navigation features are sized appropriately for fingertip interaction, whether the text size is legible, whether navigational elements are easily accessible with a minimum of scrolling, and how the page handles rotation from landscape to portrait view (Derr). I also checked the padding and density of links and form elements, as these can directly affect click accuracy on a touch screen (Sherrett). This brevity is due to the limited focus of this thesis; I recommend a more thorough examination, including integrating user testing, for established projects' sites seeking a better understanding of the impact their site design has on tablet users, in addition to the users of other devices, such as smartphones, that may be of concern.

As all of these elements are presentational rather than content-based, my preliminary impression was that that they would primarily affect the creation of the CSS3 stylesheet rather than my choice of HTML5 markup. It appeared that HTML5 would come into play only where adjusting markup choices would assist in applying styling. As one of the advantages of using HTML5 is the capacity to



Reconstructing Lives Project

Page by Page: 21st Century Lives Revealed by the Books of the House of the Books

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page testing.

About This Page

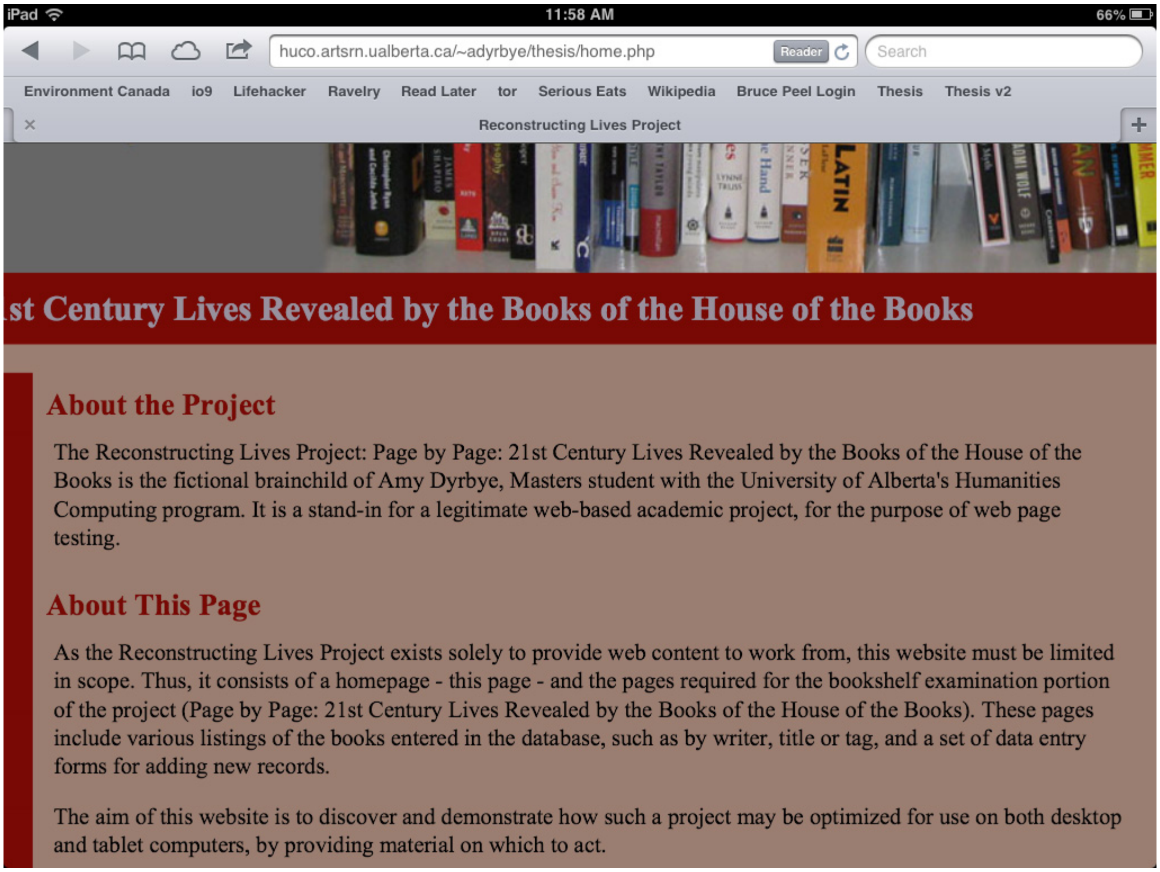
As the Reconstructing Lives Project exists solely to provide web content to work from, this website must be limited in scope. Thus, it consists of a homepage - this page - and the pages required for the bookshelf examination portion of the project (Page by Page: 21st Century Lives Revealed by the Books of the House of the Books). These pages

Figure 8: XHTML Version on iPad (Landscape Orientation)

integrate multiple stylesheets according to predetermined conditions (Warner, LaFontaine, and Andron 34-35), my findings would assist me in deciding how many stylesheets I required for the HTML5 and CSS3 version of my site.

The iPad 2[24] is designed for use with a proprietary cover that folds into a triangular stand along the left edge, encouraging users to orient the device in landscape mode and to treat the edge where the cover hinges as the top (Apple, “Smart Cover”). Thanks to this design choice, the web experience in landscape orientation was the first one I tested. I chose to confine my testing to the native Safari browser, and to compare it to the site's appearance in my primary browser, Firefox, on a 2009 iMac with a 24-inch display.

[24] iOS 5.1.1 at the time of initial testing.



st Century Lives Revealed by the Books of the House of the Books

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page testing.

About This Page

As the Reconstructing Lives Project exists solely to provide web content to work from, this website must be limited in scope. Thus, it consists of a homepage - this page - and the pages required for the bookshelf examination portion of the project (Page by Page: 21st Century Lives Revealed by the Books of the House of the Books). These pages include various listings of the books entered in the database, such as by writer, title or tag, and a set of data entry forms for adding new records.

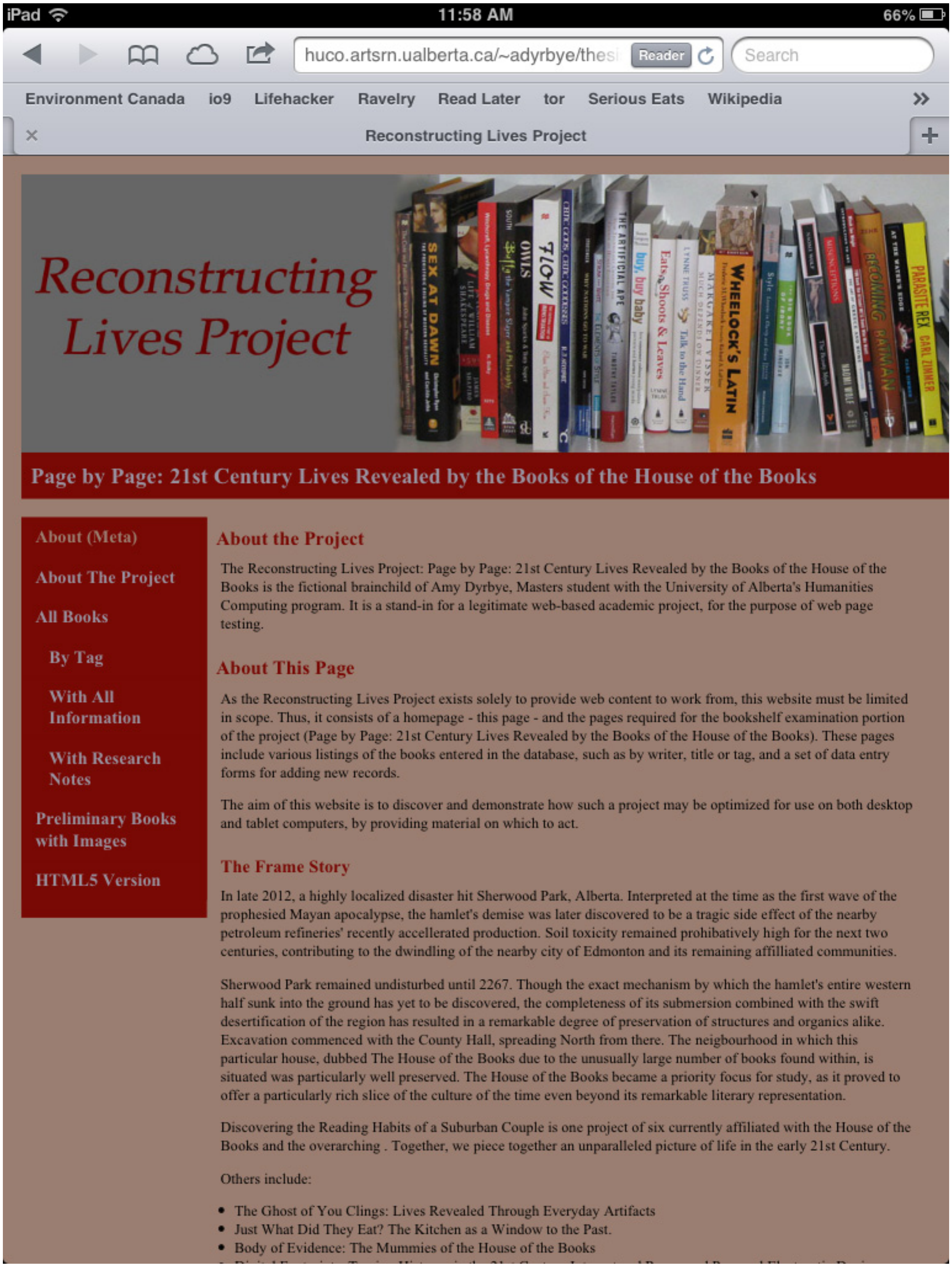
The aim of this website is to discover and demonstrate how such a project may be optimized for use on both desktop and tablet computers, by providing material on which to act.

Figure 9: XHTML Version on iPad (Landscape Orientation, Zoom Mode)

I started with the main page, 'About (Meta)'. The first thing that stands out is the 15px page border, which forces page content down and to the right. It is also immediately apparent that the banner image and header box take up over half of the page. This has the effect of forcing me to scroll down to see all of the options on the left navigational column and distracting away from the body text. Though they dominate the screen, the banner and header text appear noticeably smaller on the iPad 2 than on the screen of my primary computer. Both still display at a reasonable reading size despite the reduction. The body text also appears at a smaller scale. Characters styled to appear in 12pt Cambria on the desktop, measured directly to have a height of 4mm on the iMac, are a mere 2mm tall on the iPad 2's display. This reduction may be due in part to the absence of Cambria from the iPad’s font library; the text defaults instead to the device's generic serif font (Warner, LaFontaine, and Andron 111) in accordance with the site's stylesheet. While users can utilize the iPad's double-tap gesture to enlarge the text, this is a zoom feature rather than a means of re-scaling the page as a whole. As a result, the view of the page resizes to centre on and enlarge the text at the expense of any features to the right or left of the text column. The combination of these issues reveals some substantial flaws in my original page design in the iPad's portrait orientation.

When I changed to the portrait orientation, these problems worsened. While the entire height of the About (Meta) page fits on the screen in this mode, the far right of the page is cut off by the edge of the screen, requiring the user to choose between the ability to read the whole body text column or view the whole left navigational column. Only the banner text remains at a legible height in this view. The body text renders at the virtually illegible height of 1.5mm, while the navigational column’s bold-text links appear at a blobby 2mm that is not much easier to read. Only the banner and header show an improvement over landscape view, as they consume a far more modest slice of the page closely analogous to the desktop view. Zooming in on the main text column produced some improvement in body text legibility, bringing it up to 3mm in height, though once again at the expense of cutting off the left navigational column from view. From these observations, my assessment was that that this desktop-optimized website lose clarity and legibility when viewed on the iPad 2. Further testing would be required to determine what effect Apple's proprietary high-density Retina display, available on more recent models (Apple, "Apple Launches New iPad"), has on this experience, or how competing tablet models' displays interpret the same type of content.

I drew a handful of general conclusions from this exercise. If the landscape orientation offers a close-up of a given desktop-optimized webpage, the portrait offers a view from a distance. The result is that users contend with a webpage that shows either too little or too much. Fonts sized appropriately for comfortable


iPad
11:58 AM
66%
huco.artsrn.ualberta.ca/~adyrbye/thesi
Reader
Search
Environment Canada
io9
Lifehacker
Ravelry
Read Later
tor
Serious Eats
Wikipedia
Reconstructing Lives Project
Reconstructing Lives Project
Page by Page: 21st Century Lives Revealed by the Books of the House of the Books
About (Meta)
About The Project
All Books
By Tag
With All Information
With Research Notes
Preliminary Books with Images
HTML5 Version

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page testing.

About This Page

As the Reconstructing Lives Project exists solely to provide web content to work from, this website must be limited in scope. Thus, it consists of a homepage - this page - and the pages required for the bookshelf examination portion of the project (Page by Page: 21st Century Lives Revealed by the Books of the House of the Books). These pages include various listings of the books entered in the database, such as by writer, title or tag, and a set of data entry forms for adding new records.

The aim of this website is to discover and demonstrate how such a project may be optimized for use on both desktop and tablet computers, by providing material on which to act.

The Frame Story

In late 2012, a highly localized disaster hit Sherwood Park, Alberta. Interpreted at the time as the first wave of the prophesied Mayan apocalypse, the hamlet's demise was later discovered to be a tragic side effect of the nearby petroleum refineries' recently accellerated production. Soil toxicity remained prohibatively high for the next two centuries, contributing to the dwindling of the nearby city of Edmonton and its remaining affilliated communities.

Sherwood Park remained undisturbed until 2267. Though the exact mechanism by which the hamlet's entire western half sunk into the ground has yet to be discovered, the completeness of its submersion combined with the swift desertification of the region has resulted in a remarkable degree of preservation of structures and organics alike. Excavation commenced with the County Hall, spreading North from there. The neigbourhood in which this particular house, dubbed The House of the Books due to the unusually large number of books found within, is situated was particularly well preserved. The House of the Books became a priority focus for study, as it proved to offer a particularly rich slice of the culture of the time even beyond its remarkable literary representation.

Discovering the Reading Habits of a Suburban Couple is one project of six currently affiliated with the House of the Books and the overarching . Together, we piece together an unparalleled picture of life in the early 21st Century.

Others include:

- The Ghost of You Clings: Lives Revealed Through Everyday Artifacts
- Just What Did They Eat? The Kitchen as a Window to the Past.
- Body of Evidence: The Mummies of the House of the Books

Figure 10: XHTML Version, iPad (Portrait Orientation)

reading in a desktop browser environment are substantially smaller on the iPad 2’s screen, which may pose legibility problems, while images and other large features such as page banners tend to dominate. The relative sizes of common page design features like headers, navigational columns or body text columns translate poorly due to the differences in screen ratio, particularly as the iPad 2’s Safari browser is designed to fit the screen rather than permitting users to resize an application window. Though the portrait orientation fares worst in terms of general legibility when viewing a desktop-optimized page, its height to width ratio offers the closest visual analogue to a desktop web browser window. Conversely, the landscape orientation offers the best reading experience, but at the cost of a crowded screen.

These observations gave me a starting point from which to adjust my website for tablet viewing during conversion to HTML5 and CSS3. In particular, they show the places where it is difficult to reconcile a desktop computer and tablet's respective display and interface. For my conversion to be effective, I would need to design multiple CSS3 stylesheets to play up the strengths of each type of display. I would additionally need to be mindful of how my HTML5 markup can or cannot transfer between device types, such as my use of attributes adjusting how links behave on mouse hover or click.

My observations hint at the frustration that can plague the experience of viewing a site on a device it was not intended for, emphasizing the importance of web designers ensuring web content can be viewed appropriately on the devices their site's audience is likely to utilize. However, testing a site in this fashion has its limitations. I was aware from the outset that it is overly optimistic to expect that I will be able to anticipate all use situations or design element interactions. For this reason, I turned to formal guidelines and recommendations for guidance when undertaking the task of designing iPad-specific stylesheets. An active project's site would benefit from supplementing guidelines with even small-scale user testing, including a survey of which devices their users most commonly access the site from, to further refine its presentation and ensure the correct devices are targeted.

So it Looks a Bit Different. Why Convert My Site?

Testing my XHTML-based project website on an iPad 2 highlighted some deficiencies, but nothing came to light that would actively prevent a user from successfully navigating the site. If a site is less than ideal but navigable, what is the point in putting energy and resources into changing the site's markup? The iPad in particular is already designed to consume web-based content, whether a webpage has been designed for tablets or not (Apple, "iPad Arrives"). What about this makes it necessary to tinker with an existing website to accommodate tablet users? What benefits would be conferred by doing so, that are not already present for these users?

The reasons for converting can be traced back to XHTML's notoriously draconian error handling. XHTML as originally described utilized a new `application/xhtml+html` MIME type (also known as a content type) with the result that "[i]f there was a even a single error in your XHTML page, web browsers would have no choice but to stop processing and display an error message to the end user" (Pilgrim 10). This led to an unacceptable estimated 99% of existing XHTML pages to display errors, to the point that Appendix C of the XHTML 1.0 specification included a loophole recommending that developers use XHTML syntax, but substitute HTML's `text/html` MIME type (10). Without Appendix C's loophole, XHTML lacks backwards compatibility with HTML, stifling the usefulness of features such as XForms (Pilgrim 13).[25] Consequently, web markup before HTML5 forced developers to commit to the limitations of a single variation of markup, or to independently develop entirely new solutions as the WHAT Working Group did. HTML5 contains the backwards compatibility XHTML lacks, in addition to the canvas and media extensions the WHAT

[25] A version of web forms developed for use with XHTML. From the W3C:

> Traditional HTML Web forms don't separate the *purpose* from the *presentation* of a form. XForms, in contrast, are comprised of separate sections that describe what the form does, and how the form looks. This allows for flexible presentation options, including classic XHTML forms, to be attached to an XML form definition. (W3C, *The Forms Working Group*)

Working Group originally developed as 'Web Applications 1.0' prior to their merge with the W3C HTML Working Group (Pilgrim 13-14). Finally, the W3C formally ceased development on XHTML in 2009, choosing to let the XHTML 2 Working Group's charter expire in favour of focusing the W3C's efforts on HTML (Pilgrim 14). XHTML, regardless of MIME type, is on its way out. Between XHTML's status as no longer under development and the combination of HTML5 and CSS3 offering developers a range of new media features and a simple means to tailor a site's appearance to multiple devices, conversion away from XHTML looms on the horizon.

On a more immediate level, there are some critical differences between a desktop and a tablet computer that problematize viewing a site intended for a desktop computer on a tablet despite the fact that tablets are designed to access a wide swath of existing content. When an iPad user calls up a webpage designed for a desktop environment, everything from its aspect ratio, to its dual screen orientations, to its pixel density, to its touchscreen has a direct impact on how a user will interact with that website. For example, the hands-on navigation imposed by the touchscreen, while intuitive, can make a site difficult to interact with where its users are accustomed to clicking with a mouse. Features and links appropriate to the fine precision of a mouse pointer can be too small for effortless interaction from a human finger. On a tablet, hyperlinked text must not only be readable, but also large enough that a fingertip can trigger it accurately. Multiple links in a cluster, as occurs in navigation bars, must also be spaced far enough apart that users don't accidentally trigger the wrong one. This requires enlarged navigational elements, which must in turn avoid dominating the screen. The places where a tablet's interpretation of a website diverges from a desktop computer's can therefore become critical to the user's level of frustration, which directly impacts their willingness to continue interacting with that website.

The most obvious difference is in the screen characteristics. A desktop computer’s display is fixed into a single orientation, while aspect ratios can vary from user to user depending on their monitor. Older monitors were designed with an aspect ratio of 4:3, while the recent trend toward widescreen LCD displays has shifted the most common aspect ratio to 16:9 (Tyson and Carmack). Conversely, the iPad's aspect ratio is a return to the older 4:3 standard (Warner, LaFontaine, and Andron 40), and this ratio swaps from 4:3 to 3:4 depending on whether one is holding the device in landscape or portrait mode. While computer monitors range from pixel densities of 72 to 92 pixels per inch (ppi), the iPad 2 has 132ppi (40), and newer devices utilizing Apple's Retina display or comparable technologies can have screen resolutions as high as 264 ppi (Apple, "iPad - Design"). Consequently, websites aiming at tablet users must be able to accommodate changes in orientation, while sizing up content for the iPad's smaller screen (Derr). These differences alone account for the degraded experience I noted when viewing my inflexible XHTML project website on an iPad 2.

The iPad's touchscreen simply adds another wrinkle to the display issues. Where a mouse cursor permits precise interaction with a website's navigational elements, effective touchscreen navigation requires clickable elements be sized for finger or thumb interaction (Derr). For this reason, Apple recommends that links be at least 44 pixels by 44 pixels in size to ensure users can trigger them effectively (Warner, LaFontaine, and Andron 44). The nature of the iPad as a hand-held device also affects how users will interact with the screen. While the iPad can be rested on a surface or used with a stand, users tend to hold it in one or both hands while using their free hand or their thumbs to navigate, making the top of the screen least likely to be obscured even as the sides are easiest to touch (Derr). By inference, key information should appear at the top of the screen, keeping navigational features to the right or left. While a desktop site may integrate some of these characteristics, such as placing navigational features on the sides, as part of its overall aesthetic, it is unlikely to have them all.

How well a website translates from a desktop computer's monitor to a tablet is swiftly becoming a mainstream issue. As early as 2010, Ethan Marcotte noted that "[m]obile browsing is expected to outpace desktop-based access within three to five years," adding that mobile phones have been joined by web-ready devices up to and including video game consoles (Marcotte). Web access has become a major feature of a wide variety of devices; it is now difficult to be confident that a website's audience will primarily use just one. Tablets, with their portability and suite of purpose-driven applications, are simply the most visible. As a class, they have become emblematic of lightweight computing on the go – even as Czapracki and Burrows found that faculty were regularly inquiring about how to integrate iPads into their regular academic activities (Czapracki and Burrows 215), the iPad has also made substantial headway into the business world. In their article “Consumer Toy or Corporate Tool: The iPad Enters the Workplace,” Megan Geyer and Frances Felske report that, much as seen in academia,

> The iPad’s entry into the business world is a textbook example of consumerization—the process of bottom-up integration into the workplace by consumers and employees versus a traditional, top-down corporate-leadership push. Employees started asking for it from their companies, bringing their own iPads in to work and then showing leadership how it was useful. (Geyer and Felske 45)

This is further indicative of a larger cultural shift, an expression of the "[c]umulative acquisition of unanticipated novelties" that drive changes to science and technology (Kuhn 96). It contributes to the pool of facts at hand that in turn contributes to the emergence of paradigm. Not only has web markup matured, but portable web-ready devices have become part of the technological environment, and users have become accustomed to the ubiquitous access they provide.

Even as a student must learn how to think within a particular academic tradition before they can fully participate in it (Kuhn 111), web developers must position themselves as students to their audience. A paradigm "is determined jointly by the environment and the particular normal-scientific tradition that the student has been trained to pursue" (Kuhn 112) -- it is not sufficient to be aware that web-ready mobile devices exist, or that there are new developments to web markup. Instead, a developer must understand how each piece contributes to the web paradigm, and how they affect how adopters already use the web, to facilitate their work. While "[i]n the absence of a paradigm or some candidate for a paradigm, all facts that could possibly pertain to the development of a science are likely to seem equally relevant" (Kuhn 15), the available information has narrowed to show web-ready devices to be a major part of the web landscape. The current web development environment and the bottoms-up adoption by people within both academia and the business world has reached a point where designing a site that can accommodate tablet computers, or at minimum iPads, is increasingly important to reaching one's user base. Likewise, HTML5 not only represents an update to web markup, but facilitates integration of multiple devices into web design.

One of the ways in which web design is adapting to this new environment is through integrating responsiveness. Responsive design is an approach in which, "[r]ather than tailoring disconnected designs to each of an ever-increasing number of web devices, [designers] treat them as facets of the same experience" (Marcotte). For designers already using HTML5 with CSS3 stylesheets, the media query capability included with the stylesheet specifications is a powerful way of implementing some responsiveness by targeting stylesheets based on device characteristics such as maximum screen widths (Marcotte). Media queries make it possible to have a default stylesheet plus additional stylesheets triggered only when the viewing device meets a predefined set of criteria, all for the same underlying website content and markup (Marcotte). One page can be adapted for multiple devices simply by creating additional stylesheets and targeting them appropriately, thereby removing the need to direct visitors to, for example, a separate mobile website optimized for smartphone visitors. Combining HTML5 with CSS3 changes the problem from one of duplicating or modifying one's infrastructure to overlaying styles that take a given device's particular characteristics into account -- a far more manageable undertaking. This is the great advantage of pairing HTML5 with CSS3 rather than mixing and matching with older versions.

For all that, using an iPad is not so different from the desktop experience as to pose a sticky design challenge -- it is primarily a matter of proportions, scale and stylesheets. Even Apple's choice to promote HTML5 over Flash for multimedia content (Jobs) is only a problem on a case-by-case basis. From an academic project's standpoint, this restriction chafes only where one's project website is reliant on multimedia in the first place. While it may directly and immediately impact users of, for example, a project integrating a large number of video interviews such as Histoires de vie Montreal/Montreal Life Stories (High et al.), the exclusion of Flash-based content has no impact whatsoever on a database-driven website like CanWWR. Yet, the differences in the website's general display are substantial enough that tablet users, a group that increasingly includes academics, will find their experience of using an academic project's website is detrimentally limited if that site's design stubbornly clings to a desktop-only model. The restrictions inherent to tablets' smaller screen size, variable display resolutions and touch interface are characteristics of the class, not confined to Apple-manufactured devices. The questions tablets raise are thus less 'why bother accommodating tablets in a project's site design?' and more 'how will not taking tablets into account impact the project's user base?'

The (Not So) Gaping Chasm Between XHTML and HTML5, and CSS and CSS3

Changing an existing website from XHTML to HTML5, and its CSS to CSS3, without substantially altering the site's look or functionality is a process of adapting the existing markup and styles rather than necessarily rewriting both from the bottom up. Before proceeding, it is useful to identify what carries over from old to new and where these sets of standards differ.

Many of the changes to web markup from HTML 4.0 or XHTML in HTML5 stem back to its interactions with CSS. Though HTML is at root a simple markup language, the relationships between HTML elements and CSS styles has led to some complications, particularly where the `<div>` and `<span>` elements are concerned (Lerner). These elements are used to divide a webpage's content into discrete chunks of varying sizes without the need to co-opt another element such

`<article>`	Encloses a self-contained composition such as a blog post, comment, or article, or other independent item of content, like a widget, within a web page.
`<aside>`	Encloses content tangentially related to the surrounding page content, such as pull quotes, sidebars or web ad.
`<footer>`	Encloses content generally appearing at the foot of a web page, such as attribution information or links to related pages.
`<header>`	Encloses section headings (an `<h1>` to `<h6>` element or an `<hgroup>` grouping), table of contents, logos or banners, or any other information generally appearing at the head of a web page.
`<hgroup>`	Encloses a section heading for the purpose of grouping a set of `<h1>` to `<h6>` elements where the heading has multiple levels.
`<mark>`	Encloses text within a document to be highlighted or otherwise marked for reference purposes.
`<nav>`	Encloses a section of a page provided for navigational purposes, particularly where a block of navigational content is required.
`<section>`	Encloses a generic section of a document or application, such as a thematic grouping of content, so that a page can be split into discrete sections.
`<time>`	Encloses a time, either represented on a 24-hour clock or as a precise date in the proleptic Gregorian calendar.

Figure 11: Selected HTML5 Semantic Elements (Pilgrim 41)

as `<h6>`, and the habit of relying on a large number of specialized `<div>` elements in order to achieve one's desired layout is rampant to the point of earning the name 'div-itis' (Henick). Whether used minimally or excessively, `<div>` elements are a widely accepted means of defining broadly useful page divisions such as headers, footers and sidebars (Henick). While `<div>` elements are still semantically correct in HTML5 (Pilgrim 45), HTML5 has introduced a number of new semantic elements that directly replace the need for `<div>` elements to create common partitions, such as `<header>`, `<footer>` and `<nav>` (Pilgrim 41). For websites as simple as mine, these new page division elements constitute the most immediately visible change to the site's markup.

The 'Document Type Definition' ('doctype' or 'DTD'), a declaration used to tell a user's browser how to interpret the webpage's markup (Pilgrim 32), and the `<meta>` element, which contains the metadata telling the browser what type of document it's looking at and what character set it's using (Pilgrim 35), both contain the most dramatic changes to the HTML standard. Between the various versions of XHTML and HTML 4.01, there are 15 doctypes that can be applied to markup, each indicating it to be a particular version and then telling the browser whether that version is to be interpreted in Quirks mode, Almost Standards mode, or Standards mode (Pilgrim 32). Under this system, the doctype corresponding to the XHTML 1.0 Transitional standard utilized for the mock project website is

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
```

By contrast, HTML5 has a dramatically simplified doctype:

```
<!DOCTYPE html>
```

This collapses the multiplicity of possible interpretations down to one.

The `<meta>` element is similarly streamlined in HTML5. Where it was once necessary to declare a document to be HTML text encoded with the appropriate character set, all that is now required is a declaration of character set (Pilgrim 35). The previous standard required a statement similar to

```
<meta http-equiv=“Content-Type” content=“text/html;
charset=utf-8”>
```

while the HTML5 standard has been shortened substantially to

```
<meta charset=“utf-8” />
```

though the previous version of the `<meta>` element can still be used if needed (35). Redundant statements are now implicit wherever practical. These two changes typify the way HTML5 promotes semantic clarity and streamlines the expression of its standards (Pilgrim 31).

This commitment to streamlining is also evident in the new features of HTML5. Even the most basic integration of HTML5 standards enables the later use of a variety of functions that used to require external solutions, such as third-party software plugins, to accomplish. Most famously, HTML5 now includes a `<video>` element. This element enables video to be embedded directly into a webpage according to a predefined standard (Pilgrim 81), allowing one to declare its source, width, height, display controls, and whether to autoplay, preload along with the rest of the page, or wait to load until the user interacts with it (Pilgrim 110-111). For those projects integrating multimedia content, `<video>` may be used to natively manage and present video content without the need to rely on third-party applications like Flash Player, thereby removing the need to keep the application updated, and moving toward a web environment in which the only thing users require to view a video is a browser that supports HTML5.

HTML5 also integrates persistent local storage capabilities, a function historically missing from web applications with significant advantages over relying on browser cookies (Pilgrim 127). Cookies, data packets included with every HTTP request made to the website's server, slow down page responses by submitting the same information over and over again, contain unencrypted data, and are limited to a measly 4 KB of information (127). By contrast, HTML's Web Storage function, sometimes called 'Local Storage' or 'DOM Storage,' allows a webpage to store data on the user's local computer within their browser (Pilgrim 129). This data, now native to most desktop and mobile device web browsers including the historically reluctant Internet Explorer,[26] remains on the user's computer even

[26] While Microsoft's IE enjoyed a deserved reputation as the best web browser in 2001 (Henick 265), it had fallen well behind by 2005 as Microsoft's update cycle

after they quit their browser, and is never transmitted to a remote web server unless the user does so intentionally (129). Under HTML5, it is therefore no longer necessary to re-download content from a site on every visit until an update requires it. This expanded capacity for data storage is highly valuable to websites where users would, for example, benefit from the capacity to save settings, as with the Voyant textual analysis toolset where persistent URLs are the only way to save users' textual corpus analyses or custom skins for future use (Sinclair and Rockwell).

With the inclusion of the `<canvas>` element, the new functions in HTML5 even extend to graphics and animation. This element creates a rectangular space on a webpage for embedded JavaScript to render graphs, game graphics, images and more (Pilgrim 57). The specification includes fonts and Unicode characters, and functions that enable developers to size and place them as needed (Pilgrim 64-5). It can also incorporate preexisting images, either through the `<img>` element or JavaScript's Image object (Pilgrim 72). Click events can even be defined to make the canvas space interactive (Pilgrim 76). The `<canvas>` element would therefore be particularly beneficial to academic projects integrating interactive data visualizations, such as the TAPoR and Voyant textual analysis toolsets.

These are only a selection of the functions now integrated directly into HTML5. My own site only uses a small fraction of its potential, but even though I have not included multimedia or animations, the presence of these features would simplify the process of expanding in that direction at a later date by ensuring they do not require additional infrastructure. It is no longer necessary to worry that desired content of that type will require proprietary software, require users to have particular versions thereof installed, or will encounter incompatibilities with particular categories of devices, as Flash Player has with Apple devices. Instead, all that a developer needs is familiarity with HTML5 and an up to date browser for testing.

had no analogue to Apple and Mozilla's steady incremental improvements to their respective offerings (Henick 266).

The changes to CSS with the transition from CSS2 or 2.1 to CSS3 are more subtle than those introduced to HTML with HTML5. The presentation options available have expanded with the addition of opacity, gradients, text and box element shadows, box element corner rounding, and the capacity for multiple borders and backgrounds applied to box elements (Teague 13). These changes primarily offer a wider variety of aesthetics to a website from within the standard.

For my purposes, I found two new features to be of immediate practical interest: the media queries function, and the ability to integrate web fonts into a site design. Media queries make it possible to create multiple stylesheets for the same webpage and to tell the user's browser which one should be used to display the page on their device (Warner, LaFontaine, and Andron 120). They also enable webpages to change stylesheets dynamically, as is needed when a tablet user rotates the device between landscape and portrait modes (120), or to allow users to print a simplified version of a page (Warner, LaFontaine, and Andron 121). Characteristics like screen aspect ratios, height, width and resolution are used to target individual stylesheets (Teague 100-101). The user's browser will only apply a given stylesheet if the device meets the stated criteria (Teague 100). This offers a great deal of flexibility and granularity, removing the need to rely on a single compromise-laden one-size-fits-all stylesheet.

Where consistency of appearance is concerned, the web fonts feature alleviates the problem of font availability on the user's device. There are only five fonts that a developer can be reasonably assured will be installed on a given user's desktop computer – Ariel, Georgia, Verdana, Trebuchet MS and Times New Roman – which substantially limits the options available (Teague 126). While these same fonts are also available on the iPad, in addition to a few others such as Marker Felt, Zapfino and American Typewriter, font selection is an important part of a website's visual presentation and this small selection will not suit every site or every device (Warner, LaFontaine, and Andron 111). When a font is not available on the user's device, the browser will substitute an alternate, which can throw off the site's appearance. For example, Cambria, the font I used in the XHTML version of my website, is converted to a generic serif font when unavailable on the user's device (Henick 218), altering the text size, spacing and any cascading styles I set to utilize the text as a unit of reference. Drawing from a web font circumvents this problem altogether. Using the `@font-face` option in CSS3 circumvents this problem by telling the user's browser to load a font hosted on an external website instead of one preloaded on the user's device (Warner, LaFontaine, and Andron 111). This ensures the site's aesthetic will carry over regardless of the user's device of choice, though it is reliant on the font's web host and location remaining persistent.

These changes to both HTML and CSS expand the ways sites integrating HTML5 and CSS3 can reach their audiences. Even should an academic project's website only do so for sustainability reasons by updating only the most basic parts of the new web standards, that bare-bones level of adaptation still lays the groundwork for utilizing HTML5's multimedia capacities or CSS3's media queries for device detection as the project's needs grow and change. The focus here on adding tablet functionality is a basic, immediate and utilitarian example of what these standards are capable of, one that will help a site maintain its long-term sustainability.

The value of adapting a site to integrate HTML5 and CSS3 comes down to three things:

1. HTML5 streamlines markup and enhancing its human-readability even over XHTML 1.0 Transitional.
2. HTML5 and CSS3 work together to ease the integration of multiple devices into a website's design.
3. HTML5 natively integrates functions that previously required third-party solutions.

These additions to web markup and styling are strictly utilitarian, with immediate practical use. Project websites that integrate HTML5 and CSS3 but do not take full advantage of their capabilities nevertheless lay the groundwork for doing so later as the project and its goals grow.

Users have already embraced the new web-ready devices and demonstrated their interest in applying them to their personal and professional lives, but web content creators such as academic projects have not kept pace, holding on to proven web languages like XHTML 1.0 Transitional suited to desktop computer access. This creates a divide between users and providers where each demonstrably favours a different technological paradigm, one new and one old. In a circumstance where "the proponents of competing paradigms practice their trades in different worlds" (Kuhn 150), the practitioners of the older paradigm risk losing an audience that has converted to the new. In the case of XHTML versus HTML5, transitioning from one to the other does not carry with it the loss of users who prefer the desktop computer paradigm, while holding onto XHTML makes it that much harder to meet the needs of users who have embraced tablets, smartphones and other such web-ready devices. There is more to be gained from adopting HTML5 and CSS3, and, as I discovered in adapting my own site, it is far from an arduous process to convert.

Chapter 5 - The New Website, A Lot Like the Old Website: Making the Transition

A Small Step to Hide a Big Change

Changing a website from one version of markup to another is a small step to take for a large new vista. Kuhn observed that "during revolutions scientists see new and different things when looking with familiar instruments in places they have never looked before" (Kuhn 111), and HTML5 is a means by which web-based projects can alter their view. The web environment has not changed as quickly as the means of accessing it have expanded. HTML5 offers a streamlined solution to mediating between old data and new devices. In turn, it opens the door for those devices to change how users can relate to that data, suggesting a network of interlocking changes. Indeed, "though the world does not change with a change of paradigm, the scientist afterwords works in a different world" (Kuhn 121).

My work converting my site soon revealed that HTML5 is not half as revolutionary as the devices it supports. Instead, the paradigm shift HTML5 represents is not tied to the markup itself, but from its interaction with new technologies. HTML5's strengths lie in taking existing web practices, be they structural or related to common add-ons like video or Javascript, and making them coexist natively within its markup (Pilgrim ix) such that these additions become accessible to devices that do not support the plugins previously used to deliver, for example, video content. Now that the web is accessible on an array of devices, it works alongside CSS to facilitate that too (Warner, LaFontaine, and Andron 123). Likewise, CSS3's strengths lie in extending existing practices and accommodating the widening range of web browsers (Warner, LaFontaine, and Andron 105). For this reason, the website I ended up with looked and behaved much the same before and after transition to HTML5 and CSS3, and that the changes I used to do so proved entirely straightforward.

This is the beauty of it. My XHTML site, though legible, did not display at its best on a tablet computer alone. Adding further devices, such as a smartphone or a gaming console, would doubtless have revealed further deficiencies in its design. Converting to HTML5 alleviated many of those issues, and opened the door to do the same for other devices as well. The revolutions that result from paradigm shift are frequently invisible (Kuhn 136) -- as long as a website continues to do what it's expected to, the user has no reason to stop and ask why regardless of how that site is accessed. Yet, it is remarkable that users can access a website in as many ways as they now do, but it happened so gradually that it could be taken for granted.

Covering the seams in the user experience is why it is so important to convert. It is not because HTML5 and CSS3 offer anything truly astonishing, but because they permit a site's users to continue to focus on the data rather than the limitations of the technology used to access it. It can take the form of a minor change or major, but it makes all the difference to a project's web users. As I will demonstrate, HTML5 and CSS3 function to maintain the illusion that lets users continue to take it for granted.

Converting XHTML to HTML5

My conversion began with making a copy of the XHTML site. I duplicated the page components and stylesheet, and renamed the files to end in "_v2" to mark them as belonging to the second implementation of my site. I then updated all links and PHP includes in the copies to refer to the new files. [27]

This created a separation between my original XHTML version of the site and the HTML5 modifications. Maintaining two live implementations of the site gave me the freedom to modify my markup and any other aspects of the site required without risking the original site. The XHTML version and original stylesheet could therefore act as controls should I need to revert my changes, in addition to enabling me to offer the final versions of each side by side for comparison purposes.

[27] See Appendix Two.

None of this affected the underlying database or the PHP scripts I used to access and query the database. My only changes were to the markup and stylesheet. Neither the base infrastructure provided by the MySQL database nor the data invested in it required any changes to implement the conversion. This is due to the structure of a website paired with a database, in which the database, the website's server containing its page files and interpreting the PHP scripts they contain, and the user's web browser, form three distinct layers that together form the site (Beighley and Morrison). As my changes are strictly to the information interpreted by the user's web browser, both the database and the PHP scripts remain untouched.

I took on the changeover from XHTML to HTML5 a file-by-file basis, starting with the files designed for integration into a page via PHP include statements. My modifications began with the header file (`header_v2.php`), replacing the XHTML doctype declaration and `<html>` element attributes with their HTML5 counterparts. The original XHTML opening markup

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
```

thus became the far briefer HTML5

```
<!DOCTYPE html>
<html lang="en">
```

I left the `<link>` element directing the page to the appropriate stylesheet unchanged at this time, except to update the file name. Changes to the stylesheet, including implementing the device detection feature, required first updating the markup. Finally, the header file contains my first regular element replacement. Where I used `<div id=”header”>` to demarcate the page header in the XHTML version, the new HTML5 semantic element `<header>` replaced this `<div>`

element entirely. This section could then be set aside until I was ready to integrate additional stylesheets.

I addressed the footer file (`footer_v2.php`) next. As with the header element, HTML5 has a new semantic element that can be used to replace the `<div id="footer">` element, `<footer>`. I also replaced the `<h5>` element used to style the footer text with the new semantic element `<small>`, intended specifically for the small-print text often found in footers (Warner, LaFontaine, and Andron 82). As I had used `<h5>` as workaround in the original page, the `<small>` element proved more appropriate to the purpose of the footer text declaring the last updated information, current date and time, and links to the mock login page. This replacement also freed up `<h5>` for use in a page body should a later addition require it, rather than using another `<div>` element or needing to modify another header element such as `<h4>` with an attribute.

The file containing the left navigational column (`sidebar_v2.php`) required the least change. I made a single element substitution here, exchanging the `<div id="sidebar">` demarcating my left navigational column with its HTML5 equivalent, the `<nav>` element.

Of these three files, only the update to the document type declaration was strictly necessary to convert the page to HTML5. My replacement of the `<div>` elements I had used to create the header, footer and left navigational column with their HTML5 equivalents, or the `<h5>` element with `<small>` for the footer text, was strictly for improving the readability of my markup. I found that it is more concise and precise to utilize these new semantic elements rather than retain the previous XHTML structure. The `<div>` element is still semantically correct under HTML5, though the W3C now advises its use only as a last resort (Berjon et al. 4.5.13). I chose to adhere to the W3C's guideline, leaving only those `<div>` elements in my markup that do not have an HTML5 equivalent.

Fewer changes were required in the remaining website files, those containing the unique content of each page in the site. I made one last substitution of a `<div>`

element for a semantic element, replacing `<div id="main">` with its HTML5 equivalent `<article>,` used to define a self-contained piece of text within a document (Pilgrim 41).

The last major change to my markup is to the forms on the login page (`admin_v2.php`) and the mock data entry page (`form_v2.php`). The login page acts as a portal to the data entry page, standing in for the authentication pages commonly used by project such as TAPoR to enable members to edit content.[28] Here, I made modifications to the `<input>` elements containing the `type="text"` and `type="password"` attributes, signifying text and password fields respectively, plus the `<textarea>` element. To both, I added the attributes

```
    autocorrect="off" autocapitalize="off"
```

When an iOS device like an iPad or an iPhone is used to access a form, these two attributes prevent the web browser from autocorrecting or autocapitalizing text entered into those fields (Warner, LaFontaine, and Andron 103-4). Adding these attributes makes it easier for a user to input usernames, passwords and other potentially idiosyncratic text without interference from the device's typing assistance algorithms. This reduces the barrier to data entry via such devices.

All other elements, including the remaining `<div>` container demarcating the site banner and all header elements used outside the footer, carried over from the XHTML version of the site. The small number of changes made updating to HTML5 simple and quick to implement. Where changes were required, it was largely due to the introduction of a new element to accomplish a function that previously required a `<div>` element to accomplish. A large number of elements carried over between the XHTML and HTML5 standards. I found the ease of update was thanks to a combination of the modular web page structure I had implemented for the XHTML version, and the small number of pages affected. Websites without a modular structure or a large number of pages with unique content will find the alterations more time consuming, though still far less so than

[28] http://entry.tapor.ca/user/login

restructuring a site from the ground up. Nevertheless, updating markup from XHTML to HTML5 consists primarily of clearly defined substitutions, such as the new doctype and the one-to-one exchanges of some `<div>` containers for semantic elements, with some optional additions such as the `<small>` element.

Once a conversion is made, there are also a variety of optional new semantic elements that may be added to enhance a website. A project's website may find value in elements such as `<section>`, used to create groupings within a document; `<aside>`, used to mark pull quotes or content tangential to the main document; `<hgroup>`, used to wrap multiple levels of header elements such as in a table of contents; or `<mark>`, used to mark or highlight text for reference purposes (Pilgrim 41). These types of elements assist in organizing a page without requiring a `<div>` element as a workaround and would be particularly valuable to a project representing large amounts of text within its pages.

Converting CSS to CSS3

As with the page files for my website, I worked from a duplicate of the stylesheet to update it to CSS3. Unlike changing over XHTML to HTML5, I could add CSS3 directly into copied stylesheet without the equivalent of changing to a new document type declaration. Instead, CSS3 coexists peacefully within the same document, as it extends the specification rather than replacing it (Henick 96).

Updating the stylesheet proved the most intensive part of updating the site. The simplest change was a straightforward update of all element names replaced by HTML5 semantic elements. As with the markup, this entailed a one-to-one replacement of elements, such as `small` for `h5`, and element-attribute pairs, such as `header` for `div.header`. I could have stopped my update with this, preserving the previous desktop experience of the website in its entirety, and proceeded to creating an alternate, tablet-friendly version. Instead, I continued to integrate a selection of features characteristic of the new CSS3 standards. This not only lent the second iteration a visual distinctness, but also left room to enhance it with a feature that permits greater unity across versions.

My first new addition was the web fonts function, a feature that enables a site's font aesthetic and all related proportions to remain consistent regardless of whether the site has a stylesheet optimized for the user's device. I chose to integrate this as a ready solution to the problem of differing font sets across devices. Instead of relying on whatever fonts are installed on or packaged with a device, web fonts causes the web page to load its display font from an external website. This can be either a licensed font or one hosted freely by services such as the Google Font Directory or Font Squirrel (Warner, LaFontaine, and Andron 112). For my site, I drew a font from the Google Font Directory, also known as Google Web Fonts, due to their wide variety of explicitly open source options (Google, "About Google Web Fonts").

A `<link>` element containing the web font's address must appear within the `<head>` container of a webpage to be available to the site's stylesheet (Warner, LaFontaine, and Andron 113). This directs the page to load the font from the host URL, while adding a `rel="stylesheet"` attribute indicates the link relates to the stylesheet rather than directly to the page contents (Henick 134). I inserted

```
<link
href='http://fonts.googleapis.com/css?family=Noticia+Text:40
0,700,400italic' rel='stylesheet' type='text/css'>
```

into the `<head>` container of my header file (`header_v2.php`), so that I could access Google-hosted font Noticia Text within my stylesheet.[29]

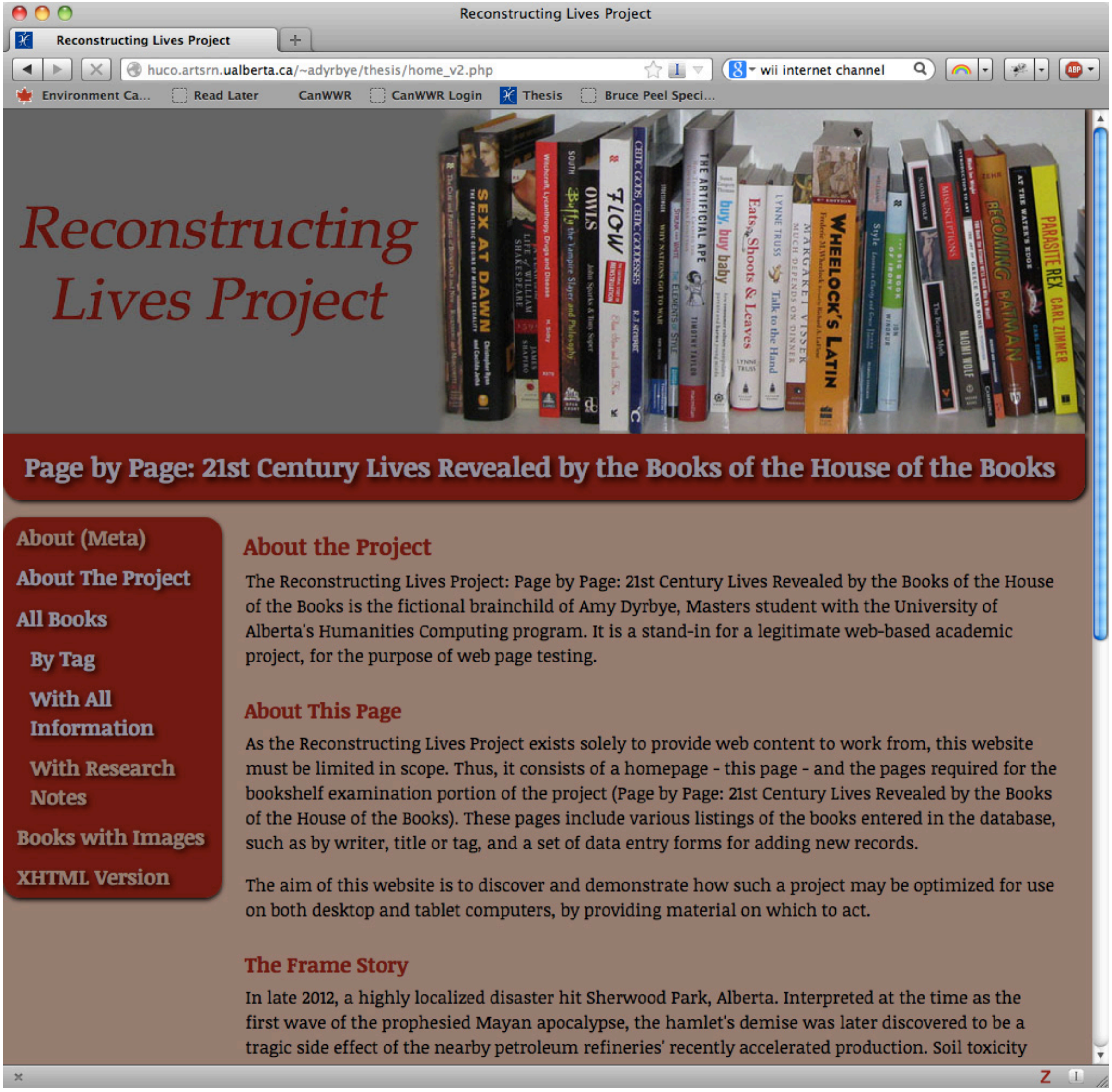
Even with a web font specified in the stylesheet, it is still good practice to indicate a font family as a backup in case the site hosting the primary font goes down (Warner, LaFontaine, and Andron 114). This will ensure a ready hierarchy of desirable backup fonts. For my own stylesheet, I set the alternate to the default serif font:

[29] `Noticia+Text:400,700,400italic` indicates the font is Noticia Text utilizing the font styles (weights) Normal 400, Bold 700 and Normal 400 Italic respectively (described at http://www.google.com/fonts#UsePlace:use/Collection:Noticia+Text).

```
      font-family: 'Noticia Text', serif;
```

Choosing an externally hosted font buttressed another means of providing a consistent experience for users. In my original stylesheet (`thesis_stylesheet.css`), I expressed the majority of styles related to size in points (pt) or pixels (px). While this is an acceptable means of scaling a page intended for desktop viewing, these units are absolute and do not scale as relative units ems (em), exes (ex) and percentages (%) do (Henick 216). For the second iteration of my stylesheet, I changed out all possible units from points and pixels to em, ex[30] and percentage. I kept only the font-size style applied to the body element in pixels, changing it from 12pt, which is relative to an inch and traditionally refers to type (Henick 36), to 15px, which is expressed instead in terms of the pixels a device's display is made up of (Henick 34). This change allowed my page's default font size to be a consistent measure against which the ems and exes used in the rest of the stylesheet could be scaled. Combined with a font selected to displays the same way across devices, this strategy ensured that all sizes, whether applied to text or box elements, remained consistently proportioned regardless of what device my page is viewed with (Warner et al 109). In the event that the Google Web Fonts server goes down or the font removed, all of the page's proportions expressed relative to the base font size will also adjust accordingly to the generic serif font specified in the stylesheet. To further the consistency of my pages, I set the text column within `<article>` to a fixed percentage of the page width, enabling the page to respond to desktop users adjusting the width of their browser window, as this unit is sensitive to context (Henick 34). Of all the units of size, only the width and height declarations for the `div.banner` and the width of the header and footer box elements remained set in units of pixels, as I wanted to keep them matched to the size of the site's banner image.

[30] One em is equivalent to two ex.



Reconstructing Lives Project

Page by Page: 21st Century Lives Revealed by the Books of the House of the Books

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page testing.

About This Page

As the Reconstructing Lives Project exists solely to provide web content to work from, this website must be limited in scope. Thus, it consists of a homepage - this page - and the pages required for the bookshelf examination portion of the project (Page by Page: 21st Century Lives Revealed by the Books of the House of the Books). These pages include various listings of the books entered in the database, such as by writer, title or tag, and a set of data entry forms for adding new records.

The aim of this website is to discover and demonstrate how such a project may be optimized for use on both desktop and tablet computers, by providing material on which to act.

The Frame Story

In late 2012, a highly localized disaster hit Sherwood Park, Alberta. Interpreted at the time as the first wave of the prophesied Mayan apocalypse, the hamlet's demise was later discovered to be a tragic side effect of the nearby petroleum refineries' recently accelerated production. Soil toxicity

Figure 12: HTML5 Desktop Version

Defining the webfont and adjusting the units used within the stylesheet relative to the default font size laid the groundwork for further additions. At this point, I integrated a few of the strictly aesthetic features new to CSS3.[31] Unlike the previous alterations, which serve the purpose of setting up a consistent appearance across devices even without device detection, these changes serve primarily to visually distinguish the HTML5/CSS3 version of the website from the

[31] See Appendix Four, thesis_stylesheet_v2.css.

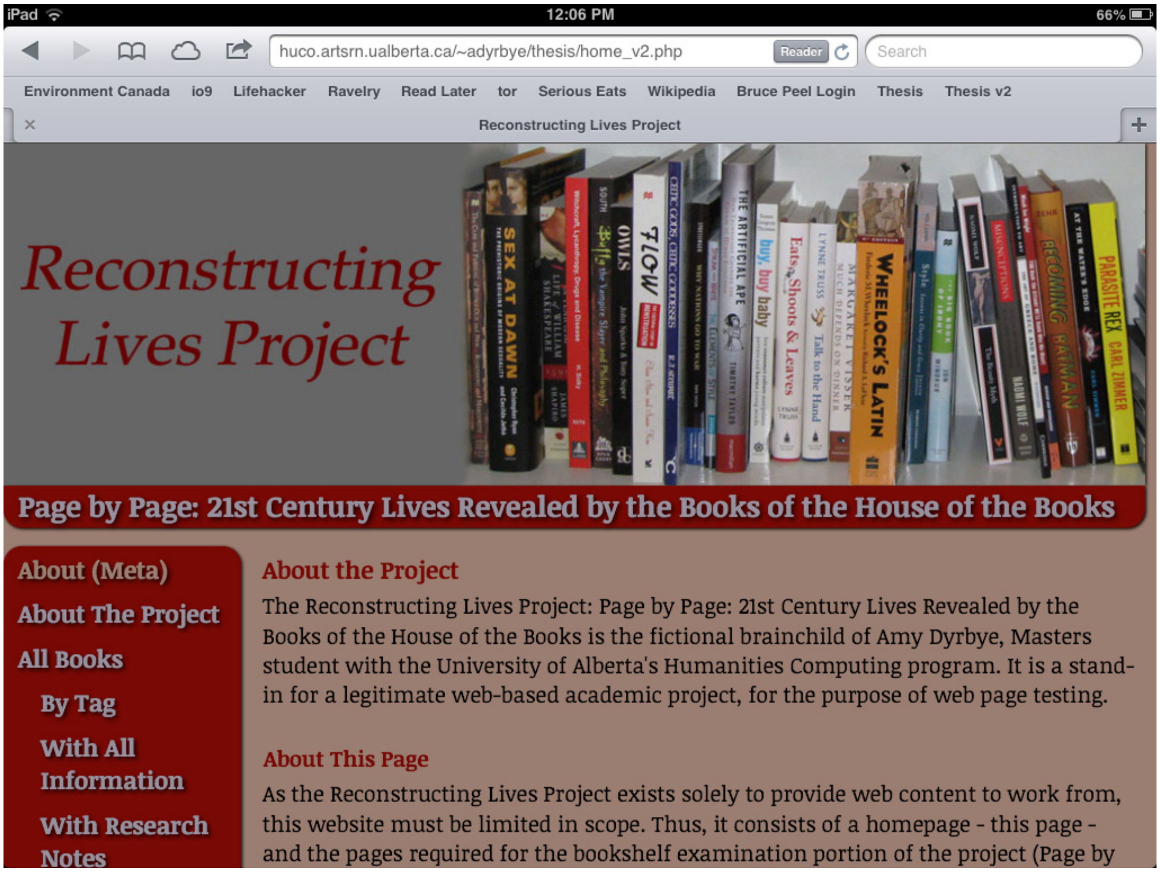
XHTML/CSS version. As an added benefit, they demonstrate a few basic ways CSS3 can be used to adjust a site's aesthetics. I confined these to the box elements `<header>`, `<footer>` and `<nav>`, and to the text within the `<h1>`, `<h6>` and `<h6 class="indent">` elements that appear within those box elements. My changes here were elementary, simply adding rounded corners to boxes and drop shadows to selected text. These few changes make the HTML5 and XHTML versions of my site instantly distinguishable, and add to the polish of the site.

I next targeted my adjustments to specific browsers, another beneficial feature included in the CSS3 specification. Following recommendations, I included targeting for the Safari, Firefox, Chrome, Internet Explorer and Opera browsers, plus the iOS version of Safari as a fallback for the stylesheets I would be targeting to the iPad (Warner, LaFontaine, and Andron 107-108). The following ensures rounded lower right and left corners will appear correctly on a box element when viewed on these browsers:

```
-webkit-border-bottom-left-radius: 1em; /* Safari 3-4 */

-webkit-border-bottom-right-radius: 1em; /* Safari 3-4 */

-moz-border-bottom-left-radius: 12em; /* Firefox 1+ */

-moz-border-bottom-right-radius: 1em; /* Firefox 1+ */

border-bottom-left-radius: 1em; /* Opera 10.5, IE 9, Safari
5, Chrome */

border-bottom-right-radius: 1em; /* Opera 10.5, IE 9, Safari
5, Chrome */
```

Should a user view the site with a browser that does not support these CSS3 styles, the drop shadows and rounded corners will simply not appear (Warner, LaFontaine, and Andron 62), and the page will look much as it does in the original XHTML/CSS version.

The updates to my original stylesheet became the default stylesheet for the HTML5 version of my site. To optimize the site for the iPad, I used this modified



Reconstructing Lives Project

Page by Page: 21st Century Lives Revealed by the Books of the House of the Books

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page testing.

About This Page

As the Reconstructing Lives Project exists solely to provide web content to work from, this website must be limited in scope. Thus, it consists of a homepage - this page - and the pages required for the bookshelf examination portion of the project (Page by

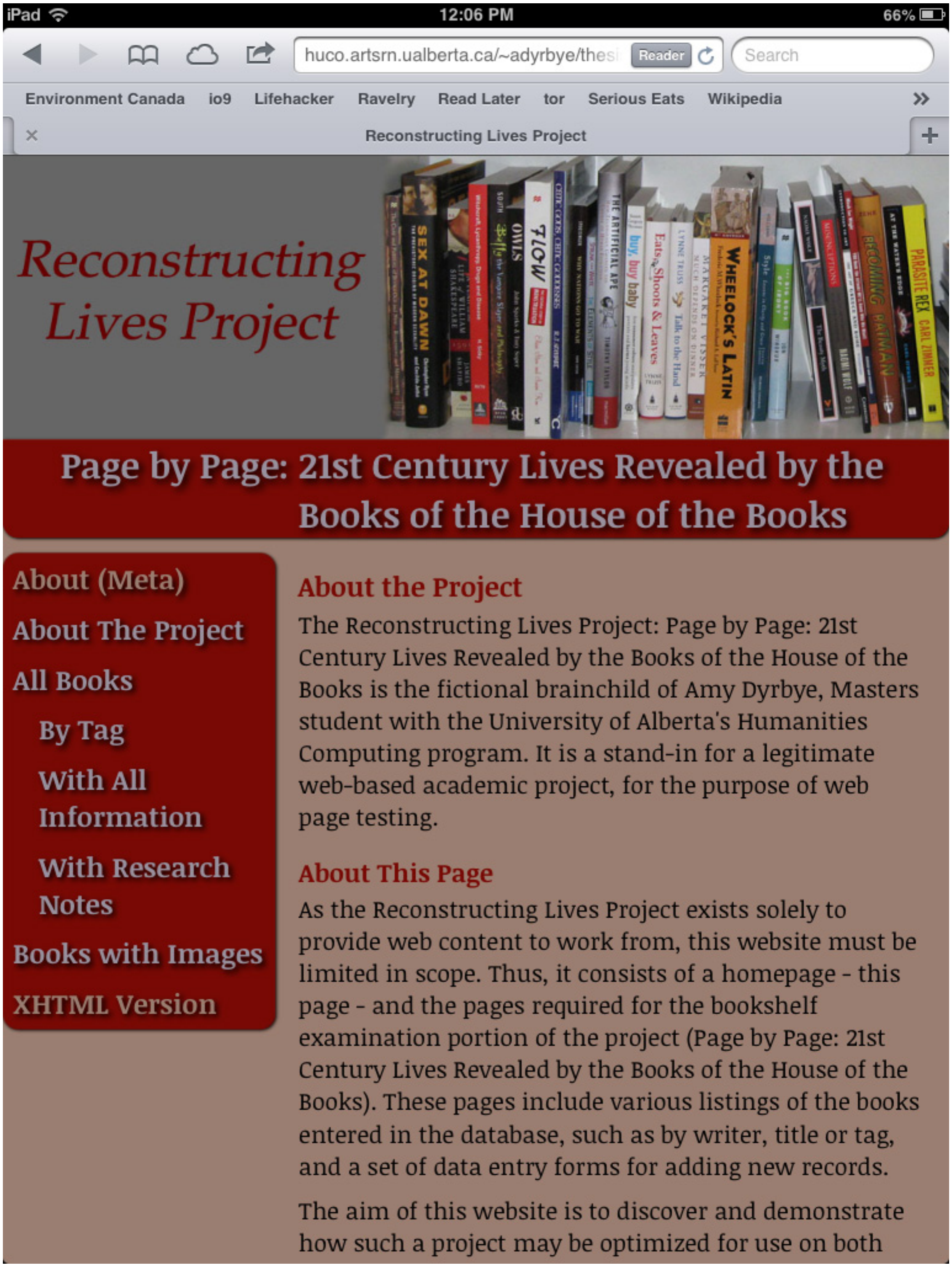
Figure 13: HTML5 iPad Version (Landscape Orientation)

stylesheet as a base for two additional stylesheets, one for each of the iPad's screen orientations. I made two copies and renamed them `ipad_portrait.css` and `ipad_landscape.css`. I could then adjust them for the iPad in portrait and landscape modes respectively.

The bulk of the changes to each iPad stylesheet involved enlarging text to ensure legibility for the orientation, and adjusting the box elements such as `nav` and `header` to fit the newly adjusted text. I also removed the style applied to links (`a:hover { text-decoration: overline underline; }`). This style, which displays an overline and underline on link text when a user hovers their mouse over it in the XHTML and HTML5 desktop versions, is superfluous on touch-screen devices as mouse hover properties are automatically converted to on-click events, negating them entirely (Warner, LaFontaine, and Andron 45). I also increased the spacing between links, and the size of form elements such as radio buttons, check boxes and submit buttons. These changes are cosmetic adjustments, not radical changes to the site design.

To reduce the problems with the text size I noted in my informal test of the XHMTL version of my site on the iPad, I needed to increase it to a more comfortable size in both orientations. I left the default text size of the `<body>` element at 15px, and specified all other sizes in relation to it. In the landscape orientation stylesheet, I increased the main header text element, `<h1>`, to 3ex, and the main text headers `<h2>` and `<h3>` set to 2.5ex and 2.25ex respectively. I set the `<p>` text element, and the text within the table elements and form elements, to 2.25ex as well. To keep tables proportionate to the text, I scaled column elements to a width of 10em, leaving height unspecified to allow for variations in the quantity of text such as title length or number of associated writers. My `<body>` font size also became the measure against which I sized box elements `<nav>` and `<footer>`, with `<nav>` set to height of 25.5em and a width of 13em, and the `<footer>` set to a height of 2.5em. The `<footer>` width remained 1000px to match the banner image width. These adjustments brought the text up into a more comfortable size and improved the spacing of links for better touch accuracy.

My adjustments to the portrait stylesheet were much the same, diverging only in font sizing. As the text in the XHTML version had appeared miniscule viewed in this mode, I scaled them up to a far greater extreme to compensate. I found the `<p>` element's text alone required sizing to 3.25ex for comfortable reading, and the `<h1>` text to 4.5ex. While this narrowed the width of the main column of text to approximately two-thirds of the screen, its effect on legibility offset the loss of screen real estate to navigation.



Reconstructing Lives Project

Page by Page: 21st Century Lives Revealed by the Books of the House of the Books

About the Project

The Reconstructing Lives Project: Page by Page: 21st Century Lives Revealed by the Books of the House of the Books is the fictional brainchild of Amy Dyrbye, Masters student with the University of Alberta's Humanities Computing program. It is a stand-in for a legitimate web-based academic project, for the purpose of web page testing.

About This Page

As the Reconstructing Lives Project exists solely to provide web content to work from, this website must be limited in scope. Thus, it consists of a homepage - this page - and the pages required for the bookshelf examination portion of the project (Page by Page: 21st Century Lives Revealed by the Books of the House of the Books). These pages include various listings of the books entered in the database, such as by writer, title or tag, and a set of data entry forms for adding new records.

The aim of this website is to discover and demonstrate how such a project may be optimized for use on both

Figure 14: HTML5 iPad Version (Portrait Orientation)

To test the iPad variants before implementing them officially, I used a simple PHP script to switch between stylesheets in HTML5 version of the header file:

```
$altcssA= '<link href="../thesis/ipad_portrait.css"
rel="stylesheet" type="text/css" />';

$altcssB= '<link href="../thesis/ipad_landscape.css"
rel="stylesheet" type="text/css" />';

echo $altcssA;
```

This enabled me to confirm that each iPad stylesheet displayed as intended, while keeping the other in easy reach. This was the last piece I required before implementing and testing the targeted stylesheets feature. Once I was satisfied with their appearance, I implemented all three at once by adding `<link>` elements directing to each desired stylesheet in the `<head>` container of the `header_v2.php` file. To target them, I then set the conditions under which a given stylesheet would be applied within the appropriate `<link>` element:

```
<!--Desktop stylesheet-->
<link href="../thesis/thesis_stylesheet_v2.css"
rel="stylesheet" type="text/css" />

<!--iPad (portrait) stylesheet-->
<link href="../thesis/ipad_portrait.css" rel="stylesheet"
type="text/css" media="only screen and (min-device-width:
768px) and (max-device-width: 1024px) and (orientation:
portrait)" />

<!--iPad (landscape) stylesheet-->
<link href="../thesis/ipad_landscape.css" rel="stylesheet"
type="text/css" media="only screen and (min-device-width:
768px) and (max-device-width: 1024px) and (orientation:
landscape)" />
```

This insertion to the HTML5 markup causes my iPad-optimized stylesheets to trigger when the minimum device width matches 768px and maximum width matches 1024px, then narrows the selection further by the device's orientation

(Warner, LaFontaine, and Andron 123). While the `media` attribute has been part of CSS since CSS2, CSS3 has extended it for more precise labelling (Bidelman). The range of attributes that can be targeted extends beyond the device width and orientation I used to include aspect ratio, resolution, monochrome, range of colours available on a device and more (Rivoal et al.) for a greater range of specificity. An untargeted stylesheet, such as my desktop stylesheet, therefore becomes the default if the device viewing the page does not meet the criteria for any of the targeted stylesheets. I confirmed this to be the case by first viewing the HTML5 site on a MacBook Pro in Firefox, Safari and Chrome, and then by viewing it on an iPad 2 in both orientations. When viewed on the iPad 2, the site indeed dynamically switched between the two stylesheets in response to rotation.

This exercise clearly shows how simple and effective it is to implement multiple, optimized stylesheets on a website. Implementing only the most basic characteristics of HTML5 and CSS3 was sufficient to bring my website to a state in which it is useful and accessible on both a desktop computer and an iPad. As these changes do not require any alterations to a project's underlying infrastructure, such as its database or any of the programming scripts that may be supporting the page, the time and effort invested is minimal for the benefits gained.

The limited modifications to my site nevertheless offer some clear improvements. When viewed on an iPad, the changes ensure text readability, the ability of the user to click accurately with their finger when navigating the site, and adjust the layout to better suit the constraints of the medium. I found text size to be particularly important -- though Safari for iOS allows users to double-tap to zoom in on text, doing so has the heavy drawback of promoting the text at the expense of navigational features at either side. Removing the need to zoom in to read the text therefore also removes the need for the user search for navigational features, allowing them to focus on the content rather than the page itself. Likewise, the larger the link or web form feature and the better they are spaced, the less likely the user is to require multiple tries to successfully interact with that link or

XHTML with CSS	HTML5 with CSS3
• `<!DOCTYPE>` declaration includes details such as version of markup used and page language • `<meta>` declaration includes document type, text type and character set • `<div>` containers used to create and position higher-level box containers (ex: footer, navigational column) • One stylesheet, optimized for desktop computer viewing • Fonts chosen are commonly installed on most computers • Sharp corners on box elements; no drop shadows • Stylesheet uses absolute units (pt, px) to size and position page elements • Form elements are spaced for mouse interaction • Links are styled to react to mouseovers • Style declarations are general	• `<!DOCTYPE>` declaration simplified to include only the markup used • `<meta>` declaration simplified to include only the character set • dedicated semantic elements used to create and position higher-level box containers (ex: `<footer>`, `<nav>`) • Three stylesheets, optimized for desktop computer viewing plus iPads in both landscape and portrait • Fonts chosen are hosted remotely at the Google Font Directory • Rounded corners and drop shadows added to box elements • Stylesheet uses relative elements (%, em, ex) to size and position page elements • Form elements are updated for touch interaction (ex: adjusted spacing, suppression of the iPad's autocorrect and autocapitalization features) • Mouseover styles removed • Style declarations are targeted to multiple browsers

Figure 15: Summary of Changes between XHTML and HTML5 Versions

feature, or worse still, become frustrated because they ended up on a page they didn't intend. These simple changes are the building blocks of a functional, tablet-friendly page; additional features such as navigational sections designed to remain static on screen as the user scrolls down the page are a nicety, but one that works best when it supplements such utilitarian adjustments.

Further Considerations

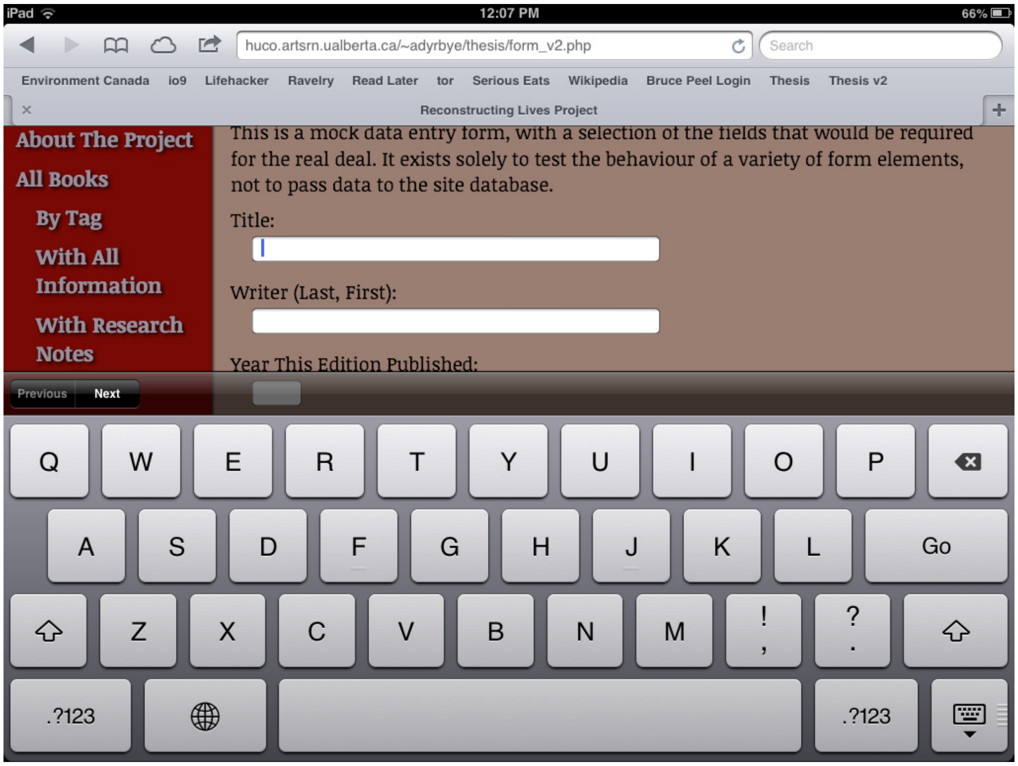
Though I successfully converted my site from XHTML to HTML5, and from CSS to CSS3, there is far more to effective site design than I could implement here. I could not make the best use of HTML5 and CSS3, or present the site to best advantage on the iPad, or even integrate much by way of responsive design into the desktop version. I created my website only to demonstrate conversion. While it can show how few changes are needed to make to a preexisting site to lay the groundwork for taking advantage of the new web standards, and to enable tablet users to better use the site, the constraints of a thesis project have left it lacking the additional work required to bring it up to its best potential.

Where tablet computers are concerned, there is more to site design than simply resizing elements to better suit the screen. The hands-on navigation enabled by a touchscreen has deeper implications than conforming page elements to the proportions of the human hand. It also influences what kinds of page layouts will be most effective -- as Kirsten Derr notes, tablet users tend to pay the most attention to the top of the screen while having the easiest time using navigational elements located at the sides (Derr). However, navigational elements need to always be in plain sight, and should not require scrolling to reach or they risk escaping notice (Derr). My site falls short here. It places its navigational features in a left-hand column, but these are fixed in place -- on pages including long lists of information called from the database such as `kitchensink.php`, the navigational elements are quickly left behind. Furthermore, my site banner graphic still occupies a disproportionately large amount of the screen when viewed on the iPad, a problem compounded by the height of the `<header>` box element that forces the meat of any given page down to the lower half of the page in landscape mode, and the middle of the page in portrait. These issues may seem small, but they take the user's attention away from the content that matters.

The treatment of images is another concern. HTML5 has introduced features that remove the need to create banner images separately through programs like Photoshop, and allows one to instead render them directly in the site, including any text one requires, by using a combination of the `<canvas>` element and JavaScript (Pilgrim 73). Alternately, CSS3 allows one to create a banner by adding a background to an HTML box element superimposing text from within a header or other text element (Ilic). Replacing images with a dynamically generated graphic is not appropriate for all circumstances, as is the case with the photos on my Books with Images (`images_v2.php`) page. Here, the individual file sizes are a major concern when not viewed on a desktop computer -- larger images slow down the page load time, and this compounds for each successive image (Warner, LaFontaine, and Andron 154). This is even more of a problem for tablet users accessing the website with a 3G data rather than over wifi. They experience not only a slower network speed but may also be subject to data usage caps; for this reason, it is strongly recommended that a page be limited to a cumulative total of no more than 75-150kb to avoid losing these users (154). Attention to these details can be critical to an academic project's site, where slow load times will negatively affect its utility to research activities.

My mocked-up data entry page demonstrates the problems form elements pose on a tablet. When a text-based form element is touched on an iPad, it prompts the device to display a keyboard on the screen, immediately reducing how much of the web page can display on screen. The keyboard itself is limited in size and defaults to the bare minimum needed for alphabetical input; to input numbers and most symbols, users must toggle between three separate keyboard configurations. Thus, fields where users must provide input with a mix of alphanumeric characters and symbols can be a source of frustration. Larger fields, such as text boxes, can combine with the keyboard to obscure most of the web page, requiring users to obscure the form to view the text field's context. Though radio buttons and check boxes are by nature small and close together, posing problems of accurate input, this can be addressed by changing their size and spacing and interacting with them does not trigger the keyboard. Drop forms, by contrast, trigger a popup list preset to a comfortable reading size. Where a website expects forms will be used from an iPad, the behavior of these elements must be considered to ensure the intended task can be accomplished without impediment.

In addition, there is more to integrating responsive design into a website than CSS3 media queries. Responsive web design is a philosophy of web development geared toward sites that respond based on user behavior, viewing device, screen characteristics and spatial orientation (Knight). Principals such as flexible grid-based layouts, which permit a page to react to the user resizing their browser



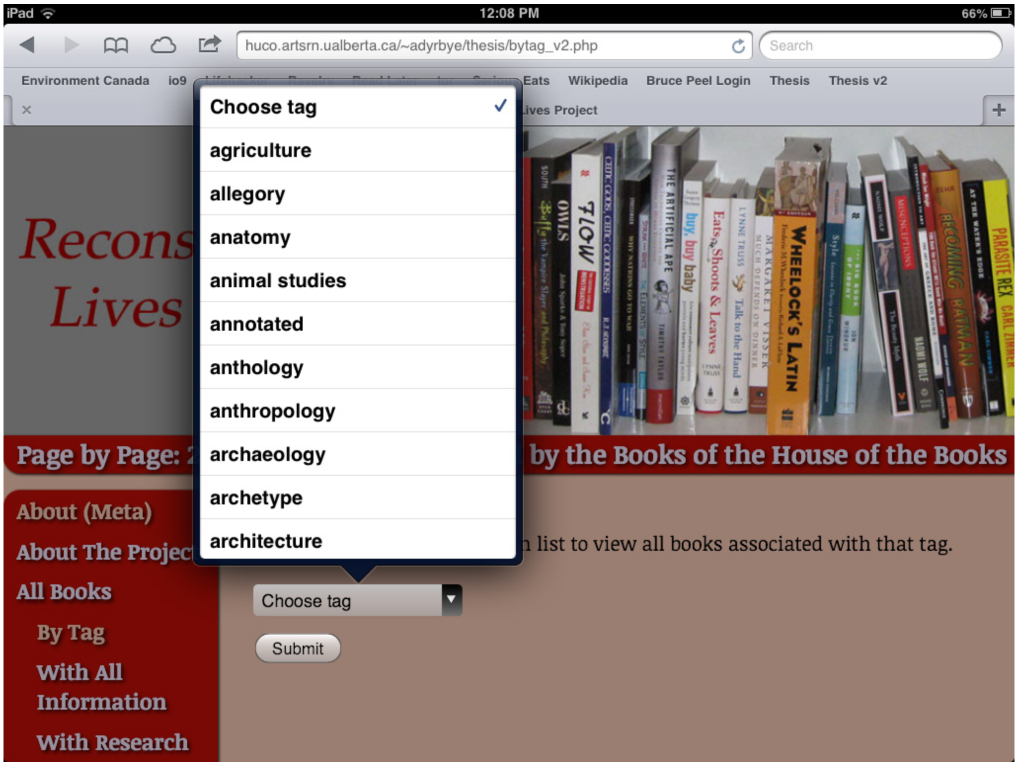
This is a mock data entry form, with a selection of the fields that would be required for the real deal. It exists solely to test the behaviour of a variety of form elements, not to pass data to the site database.

Title:

Writer (Last, First):

Year This Edition Published:

Figure 16: Data Entry Form with iPad Virtual Keyboard



list to view all books associated with that tag.

Figure 17: iPad Drop List

window, are an element of responsive design, but they don't adapt well on their own from desktop computer to netbook, let alone to a tablet (Knight). Dynamic image resizing allows images to automatically adjust to the device through scaling by implementing the CSS property max-width: 100% or by cropping them via the `overflow: hidden property` (De Graeve). The combination of media queries, flexible layouts and dynamic images goes a long way toward a site that can adapt to multiple devices and browsers, but what aspects to implement will still vary based on the needs of one's user base, and the best way to go about doing so is currently being actively debated (De Graeve). Other options include using the CSS declaration `display: none` to selectively hide content within an HTML block element on smaller screens, though this just prevents it from being displayed, not prevents it from downloading to the user's device altogether (Knight). Yet, as complex as this can get, there are substantial drawbacks to creating separate versions of one's site for desktop and mobile or tablet traffic, such as the problem of complex browser redirects, or the confusion of multiple URLs for the same content (Breheny et al.), never mind the doubled maintenance load. As the responsive design conversations continue, new methods will emerge from the developer community; it is an area of web design well worth paying attention to when upgrading one's site for long-term sustainability.

It has been some time since the Internet could be summed up as "an ever-changing, kaleidoscopic collection of hundreds of millions of documents" (Castro 12). The original web paradigm has expanded dramatically. It is still true that people view web content from a myriad of setups, requiring designs that will accommodate the largest number of one's target users in a form as close to that intended as is permitted (Castro 13). However, it is no longer a matter of Mac or PC, Internet Explorer or Netscape; the user environment now includes a wide variety of devices from desktops and laptops to netbooks, to tablets, and to a zoo of mobile phones with of variable capacities. Furthermore, many academic projects exist in a state of tension between the needs of the researchers and the needs of their user base:

> Analyzing objectives relies on identifying and defining the intended user community (or communities) and the uses. Are the intended users peers, sharing the same or similar research interests? Are they students, or perhaps the interested public? [...] For the intended user community, is the project primarily intended to facilitate research, scholarly communication of research, reference, pedagogy, or perhaps a combination of one or more of these? [...] As important as users are, those responsible for creation and maintenance of the project must also be considered. [...] Control, management, and production methods are easily overlooked if the designer focuses exclusively on the publication and end user. (Pitti)

When these projects, such as CanWWR, TAPoR or WEME, are invested in computer technologies such as databases, websites or custom programming, there is a further need to balance the time, effort and budget available between advancing that research and maintaining its technological infrastructure. This is where user studies have their value. They remain the best way to determine how a website's audience actually uses the resources on offer, and how they would like to be able to use it. This information can go a long way toward deciding what new web design techniques and standards are worthwhile directing effort toward. That said, my own project is too limited to include user testing. Instead, my efforts were constrained to finding the most obvious problems, which at best lays the groundwork for more thorough testing.

Despite the small scope of my project, I can confidently assert that integrating HTML5 and CSS3 to an existing web-based project's website has clear benefits. Firstly, this new set of markup and styling standards are poised to supplant HTML 4, XHTML and CSS as the accepted web markup and styling. Converting at this time not only anticipates the need, it eases the way to take advantage of HTML5 and CSS3's multimedia and presentation features as the site's needs grow and change. For example, HTML5 replaces the need for video or audio plugin software (Pilgrim 81), while CSS3 now permits banner images to be generated dynamically in the stylesheet rather than inserted as a static image that must be created with software such as PhotoShop (Ilic). As web-ready devices with browsers supporting HTML5 and CSS3 become commonplace among a website's user base, as tablets are becoming amongst academics, the device detection feature inherent to these standards will permit projects to take them into account.

However, it is clear from informal observation, academic publications addressing the adoption of iPads by faculty and students such as Czapracki and Burrows', and Geyer and Felske's commentary on the iPad in the workplace that tablets have become a major part of the web landscape. As Dörk et al. note, the constant web accessibility offered by mobile devices and tablets has had a direct effect on how people expect to be able to use web content, generating phenomena such as the social media backchannel at conferences and public events (Dörk et al. 1136). The impact of mobile web-ready technologies is also evident in the ongoing conversation on the impact of expectations of availability via work-issued smartphones on employees' work-life balance (Sarker et al. 143), a conversation in which tablets may soon figure. In this environment, it is no longer reasonable to expect that academics will continue to access web content strictly from desktop or laptop environments, particularly when their day to day activities and conference obligations often make carrying a laptop with all its various accessories difficult or inconvenient. Scholars preferring to carry tablets over laptops when navigating between instruction, meetings and other commitments who encounter tablet-ready websites will correspondingly find them of higher value. For those who use tablets when travelling to conferences, sessions promoting web-based projects designed to accommodate these devices will ensure that their audiences can take full advantage during the presentation. This opens the door for a higher quality of peer feedback during that critical question period, and in intersession networking opportunities.

The more tablets become a compromise solution for those days when web access and portability trump the computing power and full keyboard a laptop offers, the more important web-accessible academic projects that accommodate tablets will become. Even if the iPad's limited touch keyboard is not ideal for extended data entry sessions, it is still an acceptable tool for making smaller updates in times of need. At conferences, meetings or chance encounters with peers, its small size makes it fast and easy to retrieve from a bag to interact with web content faster and with less fuss than it would take to pull out a laptop to accomplish the same task. Unlike a laptop, tablets do not require a ready surface for use, making them particularly ideal for facilitating a brief encounter centered around web content without requiring a search for a suitable surface or the need to postpone the conversation to a time when using a desktop or laptop computer is more practical. The more academics use tablets, or see their peers using tablets, and their academic projects correspondingly anticipate the need for tablet access, the more useful tablets will become in these scenarios.

Even beyond the iPad, HTML5 and CSS3 enable a greater range of design elements, ranging from aesthetics such as rounded corners on box elements, to new elements for headers, footers, navigational features and articles, and direct responsive design integration such as dynamic targeting of style sheets based on the aspect ratio of the device viewing a web page. As HTML5 is designed to degrade gracefully on browsers that do not support it (Pilgrim x), updating one's markup does not carry the risk of losing users who are using older or otherwise non-compliant web browsers. This combination of factors strongly suggests that academic projects integrating web development that wish to make the best use of the medium are well advised to implement HTML5 and CSS3.

The advantages to HTML5 and CSS3 are compounded by, not exclusive to, tablet technologies. Projects adapting their content to these new standards will enjoy the complementary benefits of an enriched desktop experience, and the capacity to simultaneously deliver that enriched experience in a format that serves the needs of an audience embracing the portability of a tablet over a lightweight laptop or netbook. As a whole, there are far more advantages to embracing HTML5 and CSS3 than there are to remaining committed to older standards.

The Shape of the Web Paradigm

My changes are at best crude, only hinting at the potential HTML5 and CSS3 offer. But, "[t]he early versions of most new paradigms are crude. By the time their full aesthetic appeal has been developed, most of the community has been persuaded by other means" (Kuhn 156). At that point, academic projects' websites will be well behind the expectations of their users. Instead, I aim to draw attention to their potential early, so that projects like CanWWR or CWRC or TAPoR can see the value of updating their markup and styling before their users' frustrations have much opportunity to mount.

The benefits I highlight are tailored to an iPad, but it is far from the only web-ready device now in regular use. My exercise with creating multiple stylesheets alone can repeat for as many devices as needed. That is only part of the point:

> [P]aradigm debates are not really about relative problem-solving ability, though for good reasons they are usually couched in those terms. Instead, the issue is which paradigm should in the future guide research on problems many of which neither competitor can yet claim to resolve completely." (Kuhn 157)

HTML5 in particular is the markup language of choice for the mobile web paradigm. Its ability to integrate multiple devices is a benefit, but it is first and foremost the favoured solution, with advocates not only the W3C, but technology giants Apple and Google. For good or ill, it is the markup language poised to carry the web over the coming years.

Chapter 6 - Conclusion

Humanities research has changed substantially in the decades since Father Roberto Busa originally collaborated with IBM to create his *Index Thomisticus*. Many researchers now integrate computers into their daily activities, whether to keep in contact with distant colleagues, maintain a bibliography of resources or draft articles. Some take this further, presenting their work online through public websites, as the CanWWR, TAPoR, WEME and CWRC projects do. Such webpages enable projects to raise their profile and communicate findings to a broad audience, a year-round supplement to conference presentations and journal publications.

Within the *Index Thomisticus*, Father Busa's willingness to adopt new practices and innovate contributed to the longevity and scope of the project. In his lifetime, Father Busa transferred his work concording the works of Thomas Aquinas from hand-written index cards to punched cards processed by computer to a digital project hosted on the internet (Busa, "Perspectives"; Hockey, "History"). He also guided the project through multiple formats, enabled by his willingness to adopt new practices as companies like IBM refined their computer and data storage technologies. Once solely available in a multi-volume print edition, Father Busa released the *Index Thomisticus* on compact disk in 1989 (Busa, "Half a Century" 125), and debuted a web-based edition in 2005 (Busa, Bernot and Alarcón). In order to adopt these formats, he first had to update his practices from people feeding hand-sorted punched cards into a machine to an entirely digital version requiring programmers, web developers, linguistic processing and more (Busa, "Perspectives"). However, this conception obscures the process in favour of a clearly delineated progression. It elides the technologies the project teams may have tried only to abandon as a dead end, the methods they tested that seemed to offer a clear new path in one direction and instead led to another entirely.

This is why Thomas Kuhn observed that "[p]erhaps science does not develop by the accumulation of individual discoveries and inventions" (Kuhn 2). The realization that "science has included bodies of belief quite incompatible with the ones we hold today" (2) defies the construction of a smooth, unbroken progression. It is difficult to predict from their outset which practices will persist, or which technologies will fulfill their promise. At the same time, Kuhn's line of reasoning suggests that it is not the person, but the context. In the 1950s, companies and institutions were beginning to replace traditional methods of data processing with methods dependent on the faster processing of analog computer technologies (Mahoney 21); if not Father Busa, it is entirely possible another humanist would have begun the first computer-assisted concordance at this time.

The research tradition of the concordance, part of the 'normal science' in Thomas Kuhn's parlance underlying the paradigm of humanist research, shifted when Father Busa identified computers as a solution to the slow, meticulous work of hand-concording a large body of text, and inspired others to try similar methods. It also resulted in a paradigm shift within his project, both in terms of his process of concording the text, but also setting a precedent for revising the project's practices based on the available technology rather than holding fast to historic methods. Humanities projects presenting their work in digital forms, such as via project websites, follow in Father Busa's tradition even as they embody the very type of paradigmatic change that Kuhn described when compared to their predecessors.

Over the course of my thesis, I have employed Kuhn's paradigm shift in two ways. Firstly, it is useful for understanding how a community innovates and adopts new practices, which I observe in the development of new iterations of web markup, in addition to how developers have applied it. Secondly, it is a means of understanding the act of transitioning from one set of practices to another. My work changing over a website from the combination of XHTML and CSS to HTML5 and CSS3 serves to illustrate what such a transition can look like, while providing a view of its impact such a website.

Both cases call back to Thomas Kuhn's point that the most persuasive argument for a new paradigm is the solved problem, and why he positioned solved problems as the unit of scientific achievement (Kuhn 169). Going back to early web markup, Tim Berners-Lee's development of HTML from SGML solved the problem of representing documents on the internet, while the W3C introduced XML to combine the strengths of web-compatible markup with SGML's flexible capacity to describe data and metadata (W3C, *Extensible Markup Language*). Likewise, the WHAT Working Group and W3C HTML5 Working Group developed HTML5 to reconcile the respective and incompatible strengths of HTML 4 and XHTML, and web markup with an array of new web-ready devices.

New paradigms must resolve longstanding problems that cannot be met another way while preserving much of the problem-solving capacities of its predecessors (169), qualifications that HTML5 in particular typifies. The combined efforts of the WHAT Working Group and the W3C's HTML5 Working Group produced in HTML5 a web specification capable of tailoring a single website multiple devices with the best traits of HTML 4 and XHTML. Their work smoothed out HTML 4 and XHTML's shortcomings, and extending the usefulness of markup by reducing the need for third-party software to accomplish common tasks, such as embedding multimedia or locally storing web content. These capabilities are why the W3C has backed HTML5 as the successor to HTML 4 and XHTML for building web content, and why companies like Apple promote HTML5 as their preferred solution to web development for its portable devices.

Though other methods for adapting websites for mobile web-ready devices exist, such as modifications to the site's SOAP headers or flexible CSS layouts created with responsive design techniques, HTML5 accomplishes both when paired with CSS3. The combination includes a method for creating as many targeted stylesheets as required for a single website's markup, while remaining compatible with responsive design for greater flexibility. Between their improved capabilities and the backing of companies like Apple, HTML5 and CSS3 directly contribute to the normal science of a new web paradigm, satisfying Kuhn's requirement that it solve problems that defy other solutions and preserve the problem-solving capacity of the previous web paradigm.

My assessment of these new specifications is directly informed by the website I constructed. Designing the original XHTML and CSS version and taking it through the process of conversion to HTML5 and CSS3 enabled me to place two versions of the same website built under different specifications side by side, so others can observe directly how the change affects how it can be viewed on desktop versus tablet computers.

Though my work accomplished what I set out to in enacting the conversion, it necessarily truncated my capacity to go beyond that narrow focus. There are three key areas still in need of investigation. Most important is the need to integrate user testing. The scope of my thesis required I set this aside, but an academic project aiming to update its website would benefit greatly from inviting its members and audience to give feedback, both to guide the changeover and to ensure the changes enacted accomplish what they are intended to. Second is the need to investigate in depth how HTML5 and CSS3 can be supplemented with responsive design techniques to further improve the site's functionality. Updating to these two specifications alone, as I did, can do much to smooth the functioning of a website on tablets, smartphones and other web-ready devices, but there are a variety of responsive design techniques that can carry it further still. Finally, the HTML5 specification contains a number of features I was unable to integrate or investigate in detail. This includes persistent local data storage, media integration and a drawing canvas, all of which promise to be directly valuable to an academic project currently relying on solutions like third-party software plugins to accomplish similar tasks. Direct testing of these features would illuminate their usefulness and clarify the contexts in which they can put to best use.

The final website I created shows that updating an academic project's site to HTML5 and CSS3 is a practical, accessible and minimally disruptive task with clear benefits to users visiting the site from tablet computers, benefits easily extended to any other device. Though some avenues opened by these specifications remain unexplored, even the minimal approach I have modeled offers an immediate path to accommodating website visitors using devices other than desktop and laptop computers. As devices such as tablets and smartphones are here to stay, with additional web-ready devices surely on the horizon, taking the step of converting to HTML5 and CSS3 ensures that a project's website will not be left behind as the weight of users and developers adopting new devices and web specifications shifts further toward the new mobile paradigm. Making this move is a means of ensuring a web-based project's user base, whether project members or colleagues, can continue to make the best use of its resources, while laying the groundwork to take advantage of HTML5's additional features as the project's goals grow.

Bibliography

Academic Information and Communications Technologies (AICT). "Connect to UWS." *University of Alberta: Academic Information & Communication Technologies*. Web. Feb. 17 2013. < http://aict.ualberta.ca/selfhelp/howto/uws-howto >

Apple. "Apple Launches New iPad." *Apple - Press Info*. 7 Mar. 2012. Web. 15 Feb. 2013. < http://www.apple.com/pr/library/2012/03/07Apple-Launches-New-iPad.html >

Apple. "HTML5." Web. 2 Jul. 2012. < http://www.apple.com/html5/ >

Apple. "iPad Arrives This Saturday." *Apple - Press Info* (29 Mar. 2010). Web. 28 Jun. 2012. < http://www.apple.com/pr/library/2010/03/29iPad-Arrives-This-Saturday.html >

Apple. "iPad - Design." *Apple.com*. Web. 15 Feb. 2012. < http://www.apple.com/ipad/design/ >

Apple. "iPad 2 Technical Specifications." Web. 14 Jul. 2012. < http://www.apple.com/ipad/ipad-2/specs.html >

Apple. "Preparing Your Web Content for iPad." *Safari Development Library*. Web. 1 Jul. 2012. < https://developer.apple.com/library/safari/navigation/ >

Apple. “Smart Cover.” Web. 10 Jul. 2012. < http://www.apple.com/ipad/smart-cover/ >

Bansod, Aditya. "iPad 2: The HTML5 Developer Scorecard." *Sencha Blog*. 12 Mar. 2011. Web. 11 Nov. 2011. < http://www.sencha.com/blog/ipad-2-the-html5-developer-scorecard/ >

Beighley, Lynn and Morrison, Michael. *Head First PHP & MySQL*. Sebastapol: O’Reilly, 2009.

Berjon, Robin, Steve Falkner, Travis Leithead, Erika Doyle Navera, Edward O'Connor, Silvia Pfeiffer and Ian Hickson. "HTML 5.1 Nightly: A vocabulary and associated APIs for HTML and XHTML." *W3C Editor's Draft*. 22 Feb. 2013. Web. 22 Feb. 2013. < http://www.w3.org/html/wg/drafts/html/master/Overview.html >

Best, Michael. "Internet Editions of Shakespeare Principles of Tagging." *Internet Shakespeare Editions*. 8 Aug. 2010. Web. 25 Jun. 2013. < http://internetshakespeare.uvic.ca/Foyer/Tagging.html >

Bidelman, Eric. "'Mobifying' Your HTML5 Site." *HTML5 Rocks Tutorials*. 3 Mar. 2011. Web. 27 Feb. 2013. < http://www.html5rocks.com/en/mobile/mobifying/ >

Bowman, R. W., G. Gibson, D. Carberry, L. Picco, M. Miles and M. J. Padgett. “iTweezers: optical micromanipulation controlled by an Apple iPad.” *Journal of Optics* 13.4 (2011): n. pag. Web. 28 Jun. 2012. doi: 10.1088/2040-8978/13/4/044002.

Bradley, Neil. *The concise <SGML> companion*. New York: Addison-Wesley Longman, 1997.

Breheny, Rupert, Edward Jung and Matt Zürrer. “Responsive Design – harnessing the power of media queries.” *Google Webmaster Central Blog*. 30 Apr. 2012. Web. 18 Jul. 2012. < http://googlewebmastercentral.blogspot.co.uk/2012/04/responsive-design-harnessing-power-of.html >

Brown, Susan. "CWRC Platform." *The Canadian Writing Research Collaboratory*. 9 Jan. 2011. Web. 12 Feb. 2012. < http://www.cwrc.ca/uncategorized/cwrc-platform/ >

Brown, Sylvia and John Considine. *The Spacious Margin: Eighteenth-Century Printed Books and the Traces of their Readers*. Web. 18 Feb. 2013. < http://www.library.ualberta.ca/specialcollections/exhibits/current/speacious-poster-web-text.jpg >

Burton, D. M. "Automated Concordances and Word Indexes: The Fifties." *Computers and the Humanities* 15 (1981): 1-14.

Busa, R. "The Annals of Humanities Computing: The Index Thomisticus." *Computers and the Humanities* 14 (1980): 83-90.

Busa, Roberto. "Half a Century of Literary Computing: Towards a »New« Philology". *Historical Social Research* 17.2 (1992): 124-133.

Busa, Roberto A. "Foreword: Perspectives on the Digital Humanities." *A Companion to Digital Humanities*. Ed. Susan Schreibman, Ray Siemens and John Unsworth. Oxford: Blackwell, 2004. Web. 8 Aug. 2012. < http://www.digitalhumanities.org/companion/view?docId=blackwell/9781405103213/9781405103213.xml&chunk.id=ss1-1-2 >

Busa, Roberto, Eduardo Bernot and Enrique Alarcón. *Index Thomisticus (web edition)*. 2005. Web. 9 May. 2013. < http://www.corpusthomisticum.org/it/index.age >

Castro, Elizabeth. *HTML 4 For the World Wide Web (4th ed)*. Berkeley: Peachpit Press, 2000.

Cobârzan, Alin. "Consuming Web Services on Mobile Platforms." *Informatica Economica* 14.3 (2010): 98-105.

Czapracki, Alison and Cyd Burrows. "Bringing Students and Faculty Together Through Mobile Devices." *SIGUCCS '11: Proceedings of the 39th ACM annual conference on SIGUCCS (12-17 Nov 2011)*: 215-218. Web. 28 Jun. 2012. DOI: 10.1145/2070364.2070420

De Graeve, Katrien. "Responsive Web Design." *Sitepoint* (30 Jan 2012). Web. 18 Jul. 2012. < http://www.sitepoint.com/responsive-web-design/>

Demers, Patricia. *Canadian Women Reading and Writing from 1950*. Web. 3 Jul. 2012. < http://canwwrfrom1950.org/ >

Derr, Kristen. "Tablet Design Best Practices." *CraneMorley Lunch and Learn*. 23 Sept. 2011. Web. 5 Jul. 2012. < http://cmlunchandlearn.wordpress.com/2011/09/23/tablet-design-best-practices/ >

Deveria, Alexis. *When can I use... Support tables for HTML5, CSS3, etc*. Web. 1 Jul. 2012. < http://caniuse.com/# >

Dörk, Marian, Daniel Gruen, Carey Williamson and Sheelagh Carpendale. "A Visual Backchannel for Large-Scale Events." *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010): 1129-1138.

Dumas, Joseph S. and Janice C. Redish. *A Practical Guide to Usability Testing*. Portland: Intellect Books, 1999.

Dyrbye, Amy. "Expanding the CanWWR Framework." *The Canadian Writing Research Collaboratory (Blog)*. 25 May 2012. Web. 24 Nov. 2012. < http://www.cwrc.ca/blog/expanding-the-canwwr-framewor/ >

Fingas, Jon. "IHS iSuppli: Apple iPad takes 69.6 percent of tablet brand market share in Q2, reader tablets take a bruising." *Engadget*. 14 Aug. 2012. Web. 2 Dec. 2012. < http://www.engadget.com/2012/08/14/ihs-isuppli-apple-ipad-takes-69-6-percent-of-tablet-brand-share/ >

Franzoni, Vanentia, Osvaldo Gervasi, Sergio Tasso and Simonetta Pallottelli. "Web Usability on the Nintendo Wii Platform." *Computational Science and Its Applications - ICCSA 2008: International Conference, Peruguia, Italy, June 30 - July 3, 2008, Proceedings, Part II*. Ed. Gervasi, O., B Murgante, A. Lagana, D. Taniar and Y. Mun. Berlin; New York: Springer, 2008.(Amy's note: The title is exactly as it appears on the book)

Gartner. "Gartner Says Worldwide Media Tablets Sales to Reach 119 Million Units in 2012." *Gartner Newsroom*, 10 Apr. 2012. Web. 1 Jul. 2012.

Gasston, Peter. *The Book of CSS3*. San Francisco: No Starch Press, 2011. Web. 30 June 2012. <

http://proquest.safaribooksonline.com.login.ezproxy.library.ualberta.ca/book/web-development/css/9781593272869/copyright/kurt2safari_copyright_id#X2ludGVybmFsX0ZsYXNoUmVhZGVyP3htbGlkPTk3ODE1OTMyNzI4NjkvaW50cm9kdWNpbmdfY3NzMw== >

Geyer, Megan and Frances Felske. “Consumer Toy or Corporate Tool: The iPad Enters the Workplace.” *interactions* 18.4 (2011): 45-49.

Gilbert, Jason. "Chrome for iPhone And iPad: Google Announces Browser for iOS Devices, Available For Download Soon." *The Huffington Post*, 28 Jun. 2012. Web. 26 Jul. 2012. < http://www.huffingtonpost.com/2012/06/28/chrome-iphone-ipad-google_n_1634756.html >

Gilbertson, Scott. "Drink Up HTML5, It's Last Call." *Webmonkey*, 20 May 2011. Web. 1 Jul. 2012. < http://www.webmonkey.com/2011/05/drink-up-html5-its-last-call/ >

Gilbertson, Scott. "Building a Responsive, Future-Friendly Web for Everyone." *Webmonkey*, 12 Jan. 2012. Web. 1 Jul. 2012. < http://www.webmonkey.com/2012/01/building-a-responsive-future-friendly-web-for-everyone/ >

Gold, Nicholas. “Service-Oriented Software in the Humanities: A Software Engineering Perspective.” *Digital Humanities Quarterly* 3.4 (2009): n. pag. Web. 26 Jun. 2012. < http://www.digitalhumanities.org/dhq/vol/3/4/000072/000072.html

Goldberg, Kevin Howard. *XML: Second Edition*. Berkeley: Peachpit Press, 2009.

Google. *HTML5 Rocks*. Web. 1 Jul. 2012. < http://www.html5rocks.com/en/ >

Google. "About Google Web Fonts." *Google Web Fonts*. Web. 16 Jul. 2012. < http://www.google.com/webfonts/#AboutPlace:about >

Hafner, Katie and Matthew Lyon. *Where Wizards Stay Up Late: The Origins of the Internet.* New York: Touchstone, 1998.

Hansen, Derek, Marc A. Smith and Ben Shneiderman. "EventGraphs: Charting Collections of Conference Connections." *Proceedings of the 44th Hawaii International Conference on System Sciences - 2011*.

Henick, Ben. *HTML & CSS: The Good Parts*. Sebastopol: O’Reilly, 2010.

Hickson, Ian (Ed.). *HTML Living Standard*. 13 Feb. 2013. Web. 15 Feb. 2013. < http://www.whatwg.org/specs/web-apps/current-work/ >

High, Steven et al. *Histoires de vie Montreal/Montreal Life Stories*. Web. 14 Jul. 2012. < http://www.lifestoriesmontreal.ca/en/ >

Hockey, Susan. "An Agenda for Electronic Text Technology in the Humanities." *The Classical World* 91.6 (1998): 521-541.

Hockey, Susan. "The History of Humanities Computing." *A Companion to Digital Humanities*. Ed. Susan Schreibman, Ray Siemens and John Unsworth. Oxford: Blackwell, 2004. Web. 5 May 2013. < http://nora.lis.uiuc.edu:3030/companion/view?docId=blackwell/9781405103213/9781405103213.xml&chunk.id=ss1-2-1&toc.depth=1&toc.id=ss1-2-1&brand=9781405103213_brand >

IBTimes Staff Reporter. "Top Ten Reasons Why People Buy Smartphones." *International Business Times*. 3 June. 2012. Web. 5 May. 2013. < http://www.ibtimes.co.uk/articles/348123/20120603/smartphone-features-camera-browsing-gaming-apps-gps.htm >

Ilic, Manoela (Mary Lou). "CSS3: Creating Web Site Banners." *Codrops*. 6 Sept. 2009. Web. 20 Jul. 2012. < http://tympanus.net/codrops/2009/09/06/css3-creating-web-site-banners/ >

Jobs, Steve. “Thoughts on Flash.” *Apple – Hot News*, Apr. 2010. Web. 9 Jul. 2012. < http://www.apple.com/hotnews/thoughts-on-flash/ >

Knight, Kayla. "Responsive Web Design: What It Is and How To Use It." *Smashing Magazine* (12 Jan. 2011). Web. 18 Jul. 2012. < http://coding.smashingmagazine.com/2011/01/12/guidelines-for-responsive-web-design/ >

Kuhn, Thomas S. *The Structure of Scientific Revolutions (2nd Ed)*. Chicago: University of Chicago Press, 1970.

Kulkarni, Chinmay and Scott R. Klemmer. "Automatically Adapting Web Pages to Heterogeneous Devices." *CHI 2011* (2011): 1573-1578.

Lerner, Reuven M. "At the Forge: HTML5." *Linux Journal* 201 (2011). Web. 14 Jul. 2012. < http://delivery.acm.org.login.ezproxy.library.ualberta.ca/10.1145/1930000/1924409/10926.html?ip=129.128.216.34&acc=ACTIVE%20SERVICE&CFID=126785812&CFTOKEN=79880607&__acm__=1342305075_b97e3f74ab36e4dbf1ff1d5db8b8c7df >

Macauley, David. *Motel of the Mysteries*. New York: Scholastic, 1979.

Mahoney, Michael Sean. *Histories of Computing*. Cambridge: Harvard University Press, 2011.

Manian, Divya, Paul Irish, Tim Branyan, Connor Montgomery and Jake Verbaten. *HTML5 Please*. Web. 1 Jul. 2012. < http://html5please.com/ >

Marcotte, Ethan. "Responsive Web Design." *A List Apart* 306 (2010). Web. 14 Jul. 2012. < http://www.alistapart.com/articles/responsive-web-design/ >

Mozilla. *Firebug*. Web. 10 Jul. 2012. < http://getfirebug.com >

Ortiz, Guadalupe and Alfonso Garcia De Prado. “Improving device-aware Web servies and their mobile clients through an aspect-oriented, model-driven approach.” *Information and Software Technology* 52 (2010): 1080-1093.

Pilgrim, Mark. *HTML5: Up and Running*. Sebastopol: O'Reilly, 2010.

Pitti, Daniel V. “Chapter 31: Designing Sustainable Projects and Publications.” *A Companion to Digital Humanities*. Ed. Susan Schreibman, Ray Siemens and John Unsworth. Oxford: Blackwell, 2004. Web. 26 Jun. 2012. < http://www.digitalhumanities.org/companion/ >

Priego, Ernesto. "Live-tweeting at academic conferences: 10 rules of thumb." *Guardian Professional: Higher Education Network*. 3 Oct. 2012. Web. 2 Dec. 2012. < http://www.guardian.co.uk/higher-education-network/blog/2012/oct/03/ethics-live-tweeting-academic-conferences >

Raggett, Dave, Jenny Lam, Ian Alexander and Michael Kmeic. "Chapter 2." *Raggett on HTML4*. Harlow: Addison Wesley Longman, 1998. Web. 9 Dec. 2012. < http://www.w3.org/People/Raggett/book4/ch02.html >

Ray, Erik T. *Learning XML*. Sebastopol: O'Reilly, 2001. Web. 26 Aug. 2012. < http://proquest.safaribooksonline.com.login.ezproxy.library.ualberta.ca/book/xml/0596000464/introduction/learnxml-id-1.2 >

Reed, Brad. "Android still has over 50% of U.S. smartphone market share, says comScore." *BGR*. 2 Jul. 2012. Web. 2 Dec. 2012. < http://bgr.com/2012/07/02/android-market-share-us-smartphone-iphone/ >

Rivoal, Florian, Hakon Wiun Lie, Tantek Celik, Daniel Glazman and Anne van Kesteren. "Media Queries." *W3C Recommendation*. 19 Jun. 2012. Web. 27 Feb. 2013. < http://www.w3.org/TR/css3-mediaqueries/ >

Rockwell, Geoffrey et al. "Sort: Visualization." *TAPoR 2.0.* Web. 28 Jun. 2012. < http://entry.tapor.ca/tools/filterbyattribute?att_val_id=4 >

Sarker, Suprateek, Xiao Xiao, Saonee Sarker and Manjo Ahuja. "Managing Employees' Use of Mobile Technologies to Minimize Work-Life Balance Impacts." *MIS Quarterly Executive* 11.4 (2012): 143-157.

Sherrett, James. "Starting Guidelines for Tablet and iPad Website Design." *Mobify*. 21 Sept. 2011. Web. 5 July. 2012. < http://blog.mobify.com/2011/09/21/starting-guidelines-for-tablet-and-ipad-website-design/ >

Sinclair, Stéfan and Geoffrey Rockwell. *Voyant Tools: Reveal Your Texts*. Web. 15 Jul. 2012. < http://voyant-tools.org/ >

Tasman, P. "Literary Data Processing." *IBM Journal* (1957): 249-256.

Teague, Jason Cranford. *CSS3: Visual Quickstart Guide*, Fifth Edition. Berkeley: Peachpit Press, 2010. Web. 15 Jul. 2012. < http://proquest.safaribooksonline.com.login.ezproxy.library.ualberta.ca/9780131389892 >

TEI. *TEI: Text Encoding Initiative*. Web. 9 Dec. 2012. < http://www.tei-c.org/index.xml >

TEI. "TEI: Introducing the Guidelines." *TEI: Text Encoding Initiative.* Web. 9 Dec. 2012. < http://www.tei-c.org/Support/Learn/intro.xml >

The PHP Group. "What can PHP Do?" *PHP Manual*. Web. 9 Jul. 2012. < http://www.php.net/manual/en/ intro-whatcando.php >

Travis, Brian E. and Dale C. Waldt. *The SGML Implementation Guide: A Blueprint for SGML Migration*. Berlin: Springer-Verlag, 1995.

Tyson, Jeff and Carmen Carmack. "How Computer Monitors Work: Aspect Ratio and Viewable Area." *HowStuffWorks*. Web. 14 Jul. 2012. < http://computer.howstuffworks.com/monitor2.htm >

Uszkalo, Kirsten C. *Witches in Early Modern England: A Digital Humanities Project for Unveiling Witchcraft Narratives*. Web. 10 Jul. 2012. < http://witching.org >

van Herwijnen, Eric. *Practical SGML*. Boston: Kluwer Academic Publishers, 1990.

W3C. *Extensible Markup Language (XML) 1.0*. 10 Feb. 1998. Web. 8 May. 2013 < www.w3.org/TR/1998/REC-xml-19980210 >

W3C. "Soap Version 1.2 Part 1: Messaging Framework." *W3C Recommendation*. 24 June 2003. Web. 28 June 2012. < http://www.w3.org/TR/2003/REC-soap12-part1-20030624/ >

W3C. *The Forms Working Group*. 4 Apr. 2013. Web. 24 Jun. 2013. < http://www.w3.org/MarkUp/Forms/ >

W3C. *The W3C Markup Validation Service*. Web. 10 Jul. 2012. < http://validator.w3.org >

Warner, Janine, David LaFontaine and Lee Andron. *iPhone & iPad Web Design for Dummies*. Wiley Publishing, Inc.: Indianapolis, Indiana (2011).

Winokur, Danny. "Flash to Focus on PC Browsing and Mobile Apps; Adobe to More Aggressively Contribute to HTML5." *Adobe Featured Blogs*. 9 Nov. 2011. Web. 1 Jul. 2012. < http://blogs.adobe.com/conversations/2011/11/flash-focus.html >

Winter, Thomas N. "Roberto Busa, S. J., and the Invention of the Machine-Generated Concordance." *The Classical Bulletin* 75.1 (1999) 3-20.

Woodhouse, Mark. "'Innocents' and Experience: Mark Twain's Marginalia in William Prime's 'Tent Life in the Holy Land', 'The Innocents Abroad' and the Resilience of Literature." *The International Journal of the Book* 8.2 (2011): 171-186.

"Computers and the Humanities." *SpringerLink*. Web. 4 Nov. 2012. < http://www.springer.com.login.ezproxy.library.ualberta.ca/education+%26+language/linguistics/journal/10579>

Appendices

Appendix One - XHTML 1.0 Transitional Version

This is a representative selection of sections from the XHTML version of the model site.

thesisheader.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
      <head>
            <title>
<?php
$title='Reconstructing Lives Project';
$subtitle='Page by Page: 21st Century Lives Revealed by the Books
of the House of the Books';
echo $title;
?>

            </title>
            <link href="../thesis/thesis_stylesheet.css"
rel="stylesheet" type="text/css" />
            <meta http-equiv="content-type" content="text/html;
charset=UTF-8" />
      </head>
      <body>
            <div id="banner"></div>
            <div id="header">

                  <?php
                        $dbpipeline = mysql_pconnect('localhost',
'adyrbye', 'PW_REDACTED')
                        or die('Error! Error!');

                        $db = mysql_select_db('adyrbye');

                        if ($db == true) {
                        echo '<h1>' . $subtitle . '</h1>';
                        } else {
                        echo '<p><h1>The ' . $title . ' is
currently unavailable. Please try again later.</h1></p>';
                        }
                  ?>

            </div>
            <br />
```

sidebar.php

```
<div id="sidebar">
<h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home.php">Abo
ut (Meta)</a></h6>
<h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/about.php">Ab
out The Project</a></h6>
```

```
<h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/bookshelf.php
">All Books</a></h6>
<h6 class="indent"><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/bytag.php">By
Tag</a></h6>
<h6 class="indent"><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/kitchensink.p
hp">With All Information</a></h6>
<h6 class="indent"><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/withnotes.php
">With Research Notes</a></h6>
<h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/images.php">P
reliminary Books with Images</a></h6>
<h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home_v2.php">
HTML5 Version</a></h6>
</div>
```

home.php

```
<?php
include_once '../thesis/thesisheader.php';
include_once '../thesis/sidebar.php';
?>
            <div id="main">
                  <h2>About the Project</h2>
                  <p>The <?php echo $title . ': ' . $subtitle; ?>
is the fictional brainchild of Amy Dyrbye, Masters student with
the University of Alberta's Humanities Computing program. It is a
stand-in for a legitimate web-based academic project, for the
purpose of web page testing.</p>
                  <h2>About This Page</h2>
                  <p>As the <?php echo $title; ?> exists solely to
provide web content to work from, this website must be limited in
scope. Thus, it consists of a homepage - this page - and the pages
required for the bookshelf examination portion of the project
(<?php echo $subtitle; ?>). These pages include various listings
of the books entered in the database, such as by writer, title or
tag, and a set of data entry forms for adding new records.</p>
                  <p>The aim of this website is to discover and
demonstrate how such a project may be optimized for use on both
desktop and tablet computers, by providing material on which to
act.</p>
                  <h3>The Frame Story</h3>
                  <p>In late 2012, a highly localized disaster hit
Sherwood Park, Alberta. Interpreted at the time as the first wave
of the prophesied Mayan apocalypse, the hamlet's demise was later
discovered to be a tragic side effect of the nearby petroleum
refineries' recently accellerated production. Soil toxicity
remained prohibatively high for the next two centuries,
contributing to the dwindling of the nearby city of Edmonton and
its remaining affilliated communities.</p>
                  <p>Sherwood Park remained undisturbed until
2267. Though the exact mechanism by which the hamlet's entire
western half sunk into the ground has yet to be discovered, the
completeness of its submersion combined with the swift
desertification of the region has resulted in a remarkable degree
of preservation of structures and organics alike. Excavation
commenced with the County Hall, spreading North from there. The
neigbourhood in which this particular house, dubbed The House of
the Books due to the unusually large number of books found within,
is situated was particularly well preserved. The House of the
Books became a priority focus for study, as it proved to offer a
particularly rich slice of the culture of the time even beyond its
remarkable literary representation.</p>
                  <p>Discovering the Reading Habits of a Suburban
Couple is one project of six currently affiliated with the House
of the Books and the overarching <?php $title ?>. Together, we
piece together an unparalleled picture of life in the early 21st
Century.</p>
                  <p>Others include:</p>
                  <ul>
                  <li>The Ghost of You Clings: Lives Revealed
Through Everyday Artifacts</li>
                  <li>Just What Did They Eat? The Kitchen as a
Window to the Past.</li>
                  <li>Body of Evidence: The Mummies of the House
of the Books</li>
                  <li>Digital Footprints: Tracing History via the
21st Century Internet and Recovered Personal Electrontic
Devices</li>
                  <li>The Work of These Hands: The Practice of
Handcrafting in the Early 21st Century, As Evidenced in the House
of the Books</li>
                  </ul>
                  <br />
                  <br />
            </div>
<?php
include_once '../thesis/thesisfooter.php';
?>
```

bookshelf.php

```
<?php
include_once '../thesis/thesisheader.php';
include_once '../thesis/sidebar.php';
?>
            <div id="main">
                  <h2>All Books</h2>
                  <p>This is a listing of all books found on the
first bookcase to be catalogued from the House of the Books.</p>
                  <p>Some books have research notes associated
with them. To view the list of books with notes, please choose
'With Research Notes' from the menu at left.</p>
                  <p>All volumes have also been tagged according
to subject matter. To view the list of books with associated tags,
please choose 'With Tags' from the menu at left, or, to search for
all books with a particular tag, choose 'By Tag'.</p>
<?php
$query = 'SELECT books.id AS b_id, books.title, books.format,
books.originalyear, books.editionyear, books.publisher, books.isbn
FROM books';
$cloud = mysql_query($query, $dbpipeline);

//adapting the 'bytag' solution to echoing out the entire
bookshelf
while ($array = mysql_fetch_assoc($cloud)){
      if (!empty($array)) {
            echo '<table class="bookinfo"><tr><td
width="200"><strong>' . str_replace('&', '&amp;', $array['title'])
. '</strong></td><td width="200">';
            //adding second query for writer names
            $select_writers = 'SELECT books.id, writers.name AS
wname FROM books, writers, bookstowriters WHERE
books.id=bookstowriters.book_id AND
writers.id=bookstowriters.writer_id AND books.id="' .
$array['b_id'] . '"';
            $writers_selected = mysql_query($select_writers,
$dbpipeline);
            while ($writers_array =
mysql_fetch_assoc($writers_selected)){
                  if (!empty($writers_array)){
                        echo $writers_array['wname'] . '<br />';
                  }
            }
                        echo '</td></tr><tr><td ><em>Originally
Published:</em> ';
            if ($array['originalyear'] != NULL) {
                  echo $array['originalyear'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td><td><em>Edition Published</em>: ';
            if ($array['editionyear'] != NULL) {
                  echo $array['editionyear'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td></tr><tr><td><em>ISBN:</em> ';
            if ($array['isbn'] != NULL) {
                  echo $array['isbn'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td><td><em>Format:</em> ' . $array['format'] .
'</td></tr>';
            echo '<tr><td colspan="2"><em>Publisher:</em> ' .
str_replace('&', '&amp;', $array['publisher']) .
'</td></tr></table>';
      }else {
            echo $sampleentry;
      }
}
?>
            </div>
<?php
include_once '../thesis/thesisfooter.php';
?>
```

bytag.php

```
<?php
include_once '../thesis/thesisheader.php';
include_once '../thesis/sidebar.php';
$query = 'SELECT tags.name FROM tags ORDER BY name';
$cloud = mysql_query($query, $dbpipeline);
?>
<div id="main">
<h3>View Books by Tag</h3>
<p>Select a tag from the drop-down list to view all books
associated with that tag.</p>
<form id="selecttag" method="post" action="bytag.php">
      <select name="tag" id="tag">
            <option value="choose_tag" selected="selected">Choose
tag</option>
```

```
            <?php
                  $i = 0;
                  while ($array = mysql_fetch_assoc($cloud)){
                        do {
                              if ($array != NULL) {
                                    echo "<option value='" .
str_replace('&', '&amp;', $array['name']) . "'>" .
str_replace('&', '&amp;', $array['name']) . "</option>";
                                    break;
                                    } else {
                                    echo "<option
value='invalid'>Invalid!</option>";
                                    break;
                              }
                        } while ($i < count($array));
                        $i++;
                  }
                        ?>
            </select>
            <input type="submit" name="Submit" value="Submit" />
</form>
<?php
$select_books = 'SELECT books.title, books.format,
books.originalyear, books.editionyear, books.publisher,
books.isbn, tags.name AS tname FROM books, tags, bookstagged WHERE
books.id = bookstagged.book_id AND tags.id = bookstagged.tag_id
AND tags.name = "' . $_POST['tag'] . '"';
$books_selected = mysql_query($select_books, $dbpipeline);

if (!empty($_POST['tag'])){
      echo '<p>Now showing all books tagged with "' .
$_POST['tag'] . '":</p>';
}
while ($books_array = mysql_fetch_assoc($books_selected)){
      if (!empty($books_array)) {
            echo '<table class="bookinfo"><tr><td
width="200"><strong>' . $books_array['title'] . '</strong></td><td
width="200">';
            //adding second query for writer names
            $select_writers = 'SELECT books.title, writers.name AS
wname FROM books, writers, bookstowriters WHERE
books.id=bookstowriters.book_id AND
writers.id=bookstowriters.writer_id AND books.title="' .
$books_array['title'] . '"';
            $writers_selected = mysql_query($select_writers,
$dbpipeline);
            while ($writers_array =
mysql_fetch_assoc($writers_selected)){
                  if (!empty($writers_array)){
                        echo $writers_array['wname'] . '<br />';
                  }
            }
                        echo '</td></tr><tr><td ><em>Originally
Published:</em> ';
            if ($books_array['originalyear'] != NULL) {
                  echo $books_array['originalyear'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td><td><em>Edition Published</em>: ';
            if ($books_array['editionyear'] != NULL) {
                  echo $books_array['editionyear'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td></tr><tr><td><em>ISBN:</em> ';
            if ($books_array['isbn'] != NULL) {
                  echo $books_array['isbn'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td><td><em>Format:</em> ' .
$books_array['format'] . '</td></tr>';
            echo '<tr><td colspan="2"><em>Publisher:</em> ' .
str_replace('&', '&amp;', $books_array['publisher']) .
'</td></tr></table>';
            } else {
            echo $sampleentry;
      }
}
?>
<br />
<br />
</div>
<?php
include_once '../thesis/thesisfooter.php';
?>
```

admin.php

```
<?php
include_once '../thesis/thesisheader.php';
include_once '../thesis/sidebar.php';
?>
            <div id="main">
                  <h2>Log In</h2>
                  <p>This is the entry page for the mock data
entry form.</p>
                  <form name="mocklogin" action="form.php"
method="post">
                        <p>Username:
                              <input name="user" type="text"
id="user" size="5" maxlength="5" />
                        </p>
                        <p>Password:
                              <input name="pwd" type="password"
id="pwd" size="12" maxlength="12" />
                        </p>
                        <p>
                              <input type="submit" name="submit"
value="Log in" />
                              <input name="reset" type="reset"
id="reset" value="Clear form" />
                        </p>
                  </form>
            </div>
<?php
include_once '../thesis/thesisfooter.php';
?>
```

form.php

```
<?php
include_once '../thesis/thesisheader.php';
include_once '../thesis/sidebar.php';
$username = 'admin';
$password = 'arglefraster';
```

```
$form = '
            <div id="main">
                  <h2>Enter a New Book</h2>
                  <p>This is a mock data entry form, with a
selection of the fields that would be required for the real deal.
It exists solely to test the behaviour of a variety of form
elements, not to pass data to the site database.</p>
                  <form name="mockentry" action="form.php"
method="post">
                        <p>Title: <br />
                              <input name="title" type="text"
id="title" size="50" maxlength="100" />
                        </p>
                        <p>Writer (Last, First): <br />
                              <input name="writer" type="text"
id="writer" size="50" maxlength="50" />
                        </p>
                        <p>Year This Edition Published: <br />
                              <input name="edition_year"
type="text" id="edition_year" size="4" maxlength="4" />
                        </p>
                        <p>ISBN: <br />
                              <input name="isbn" type="text"
id="isbn" size="13" maxlength="13" />
                        </p>
                        <p>Format: <br />
                              <input type="radio" name="format"
value="hardcover" /> hardcover<br />
                        <input type="radio" name="format"
value="trade paperback" /> trade paperback<br />
                        <input type="radio" name="format"
value="mass market paperback" /> mass market paperback</p>
                        <p>Tag with: <br />
                        <input name="allegory" type="checkbox"
id="allegory" value="yes" />allegory<br />
                        <input name="biology" type="checkbox"
id="biology" value="yes" />biology<br />
                        <input name="cultural studies"
type="checkbox" id="cultural_studies" value="yes" />cultural
studies<br />
                        <input name="digital humanities"
type="checkbox" id="digital_humanities" value="yes" />digital
humanities<br />
                        <input name="ethics" type="checkbox"
id="ethics" value="yes" />ethics<br />


                              <input name="feminism"
type="checkbox" id="feminism" value="yes" />feminism<br />
                              <input name="genetics"
type="checkbox" id="genetics" value="yes" />genetics<br />
                        <input name="history" type="checkbox"
id="history" value="yes" />history<br />
                        <input name="literary criticism"
type="checkbox" id="literery_criticism" value="yes" />literary
criticism<br />
                        <input name="middle ages" type="checkbox"
id="middle_ages" value="yes" />middle ages<br />
                        <input name="nature" type="checkbox"
id="nature" value="yes" />nature<br />
                        </p>
                        <p>
                        <textarea name="notes" cols="50" rows="4"
id="notes">This space provided for research notes.</textarea>
```

```
                        </p>
                        <p>
                              <input type="submit" name="submit"
value="Submit" />
                              <input name="reset" type="reset"
id="reset" value="Clear form" />
                        </p>
                  </form>
            </div>';
if ($_POST['user']==$username){
      if ($_POST['pwd']==$password){
            echo $form;
      }else{
            echo '<div id="main"><p><strong>Bad username or
password.</strong></p><p>Use the log in link in the footer to try
again, or try the navigation bar at left to choose a different
page.</p></div>';
      }
}
include_once '../thesis/thesisfooter.php';
?>
```

thesisfooter.php

```
<div id="footer">
<h5><a href="http://validator.w3.org/check?uri=referer"><img
      src="http://www.w3.org/Icons/valid-xhtml10" alt="Valid XHTML
1.0 Transitional" height="31" width="88" /></a>  Last updated
March 28, 2012. <?php
echo strftime('It is currently %H:%M %p on %A, %B %e %G');
?> | <a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/admin.php">Lo
g in</a></h5></div>
</body>
</html>
<?php
mysqli_close($dbpipeline);
?>
```

Appendix Two - HTML5 Version

Only those segments with notable representative changes to the markup have been included in this appendix.

header_v2.php

```
<!DOCTYPE html>
<html lang="en">
      <head>
<?php
$linkcss= '
            <!--Desktop stylesheet-->
            <link href="../thesis/thesis_stylesheet_v2.css"
rel="stylesheet" type="text/css" />
            <!--iPad (portrait) stylesheet-->
            <link href="../thesis/ipad_portrait.css"
rel="stylesheet" type="text/css" media="only screen and (min-
device-width: 768px) and (max-device-width: 1024px) and
(orientation: portrait)" />
            <!--iPad (landscape) stylesheet-->
            <link href="../thesis/ipad_landscape.css"
rel="stylesheet" type="text/css" media="only screen and (min-
device-width: 768px) and (max-device-width: 1024px) and
(orientation: landscape)" />';
$altcssA= '
            <link href="../thesis/ipad_portrait.css"
rel="stylesheet" type="text/css" />';
$altcssB= '
            <link href="../thesis/ipad_landscape.css"
rel="stylesheet" type="text/css" />';
echo $linkcss;
?>
            <!--Meta information for character set, search engine
optimization (SEO)-->
            <meta charset="UTF-8" />
            <!--Link to a free web font from Google Web Fonts-->
            <link
href='http://fonts.googleapis.com/css?family=Noticia+Text:400,700,
400italic' rel='stylesheet' type='text/css'>
            <title>
<?php
$title='Reconstructing Lives Project';
$subtitle='Page by Page: 21st Century Lives Revealed by the Books
of the House of the Books';
echo $title;
?>
            </title>
      </head>
      <body>
            <div id="banner"></div>
            <header>
                  <a href="home_v2.php">
                  <?php
                        $dbpipeline = mysql_pconnect('localhost',
'adyrbye', 'PW_REDACTED')
                        or die('Error! Error!');
                        $db = mysql_select_db('adyrbye');
                        if ($db == true) {
                        echo '<h1>' . $subtitle . '</h1>';
                        } else {
                        echo '<p><h1>The ' . $title . ' is
currently unavailable. Please try again later.</h1></p>';
```

```
                        }
                  ?>
                  </a>
            </header>
```

sidebar_v2.php

```
            <nav>
                  <h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home_v2.php">
About (Meta)</a></h6>
                  <h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/about_v2.php"
>About The Project</a></h6>
                  <h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/bookshelf_v2.
php">All Books</a></h6>
                  <h6 class="indent"><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/bytag_v2.php"
>By Tag</a></h6>
                  <h6 class="indent"><a
      href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/kitchen
sink_v2.php">With All Information</a></h6>
                  <h6 class="indent"><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/withnotes_v2.
php">With Research Notes</a></h6>
                  <h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/images_v2.php
">Books with Images</a></h6>
                  <h6><a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/home.php">XHT
ML Version</a></h6>
            </nav>
```

home_v2.php

```
<?php
include_once '../thesis/header_v2.php';
include_once '../thesis/sidebar_v2.php';
?>
            <article>
                  <h2>About the Project</h2>
                  <p>The <?php echo $title . ': ' . $subtitle; ?>
is the fictional brainchild of Amy Dyrbye, Masters student with
the University of Alberta's Humanities Computing program. It is a
stand-in for a legitimate web-based academic project, for the
purpose of web page testing.</p>
                  <h3>About This Page</h3>
                  <p>As the <?php echo $title; ?> exists solely to
provide web content to work from, this website must be limited in
scope. Thus, it consists of a homepage - this page - and the pages
required for the bookshelf examination portion of the project
(<?php echo $subtitle; ?>). These pages include various listings
of the books entered in the database, such as by writer, title or
tag, and a set of data entry forms for adding new records.</p>
                  <p>The aim of this website is to discover and
demonstrate how such a project may be optimized for use on both
desktop and tablet computers, by providing material on which to
act.</p>
                  <h3>The Frame Story</h3>
                  <p>In late 2012, a highly localized disaster hit
Sherwood Park, Alberta. Interpreted at the time as the first wave
of the prophesied Mayan apocalypse, the hamlet's demise was later
```

```
discovered to be a tragic side effect of the nearby petroleum
refineries' recently accelerated production. Soil toxicity
remained prohibatively high for the next two centuries,
contributing to the dwindling of the nearby city of Edmonton and
its remaining affiliated communities.</p>
                  <p>Sherwood Park remained undisturbed until
2267. Though the exact mechanism by which the hamlet's entire
western half sunk into the ground has yet to be discovered, the
completeness of its submersion combined with the swift
desertification of the region has resulted in a remarkable degree
of preservation of structures and organics alike. Excavation
commenced with the County Hall, spreading North from there. The
neighbourhood in which this particular house, dubbed The House of
the Books due to the unusually large number of books found within,
is situated was particularly well preserved. The House of the
Books became a priority focus for study, as it proved to offer a
particularly rich slice of the culture of the time even beyond its
remarkable literary representation.</p>
                  <p>Discovering the Reading Habits of a Suburban
Couple is one project of six currently affiliated with the House
of the Books and the overarching <?php $title ?>. Together, we
piece together an unparalleled picture of life in the early 21st
Century.</p>
                  <p>Others include:</p>
                  <ul>
                        <li>The Ghost of You Clings: Lives
Revealed Through Everyday Artifacts</li>
                        <li>Just What Did They Eat? The Kitchen as
a Window to the Past.</li>
                        <li>Body of Evidence: The Mummies of the
House of the Books</li>
                        <li>Digital Footprints: Tracing History
via the 21st Century Internet and Recovered Personal Electrontic
Devices</li>
                        <li>The Work of These Hands: The Practice
of Handcrafting in the Early 21st Century, As Evidenced in the
House of the Books</li>
                  </ul>
                  <br />
                  <br />
            </article>
<?php
include_once '../thesis/footer_v2.php';
?>
```

bookshelf_v2.php

```
<?php
include_once '../thesis/header_v2.php';
include_once '../thesis/sidebar_v2.php';
?>
            <article>
                  <h2>All Books</h2>
                  <p>This is a listing of all books found on the
first bookcase to be catalogued from the House of the Books.</p>
                  <p>Some books have research notes associated
with them. To view the list of books with notes, please choose
'With Research Notes' from the menu at left.</p>
                  <p>All volumes have also been tagged according
to subject matter. To view the list of books with associated tags,
please choose 'With Tags' from the menu at left, or, to search for
all books with a particular tag, choose 'By Tag'.</p>
<?php
```

```
$query = 'SELECT books.id AS b_id, books.title, books.format,
books.originalyear, books.editionyear, books.publisher, books.isbn
FROM books';
$cloud = mysql_query($query, $dbpipeline);
//adapting the 'bytag' solution to echoing out the entire
bookshelf
while ($array = mysql_fetch_assoc($cloud)){
      if (!empty($array)) {
            echo '<table class="bookinfo"><tr><td><strong>' .
str_replace('&', '&amp;', $array['title']) . '</strong></td><td>';
            //adding second query for writer names
            $select_writers = 'SELECT books.id, writers.name AS
wname FROM books, writers, bookstowriters WHERE
books.id=bookstowriters.book_id AND
writers.id=bookstowriters.writer_id AND books.id="' .
$array['b_id'] . '"';
            $writers_selected = mysql_query($select_writers,
$dbpipeline);
            while ($writers_array =
mysql_fetch_assoc($writers_selected)){
                  if (!empty($writers_array)){
                        echo $writers_array['wname'] . '<br />';
                  }
            }
                        echo '</td></tr><tr><td ><em>Originally
Published:</em> ';
            if ($array['originalyear'] != NULL) {
                  echo $array['originalyear'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td><td><em>Edition Published</em>: ';
            if ($array['editionyear'] != NULL) {
                  echo $array['editionyear'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td></tr><tr><td><em>ISBN:</em> ';
            if ($array['isbn'] != NULL) {
                  echo $array['isbn'];
                  } else {
                  echo 'n/a';
                  }
            echo '</td><td><em>Format:</em> ' . $array['format'] .
'</td></tr>';
            echo '<tr><td colspan="2"><em>Publisher:</em> ' .
str_replace('&', '&amp;', $array['publisher']) .
'</td></tr></table>';
      }else {
            echo $sampleentry;
      }
}
?>
            </article>
<?php
include_once '../thesis/footer_v2.php';
?>
```

footer_v2.php

```
            <footer>
                  <small>Last updated May 21, 2012. <?php
echo strftime('It is currently %H:%M %p on %A, %B %e %G');
```

```
?> | <a
href="http://huco.artsrn.ualberta.ca/~adyrbye/thesis/admin_v2.php"
>Log in</a></small>
            </footer>
      </body>
</html>
<?php
mysqli_close($dbpipeline);
?>
```

Appendix Three - Stylesheet for XHTML Version

thesis_stylesheet.css

```css
body {
      background: #9B7E71;
      font-family: Cambria,"Times New Roman",Times,Serif;
      font-size: 12pt;
      margin: 15pt;
      text-align: left;
}

#banner {
      background-image:url(../Pictures/thesis_banner.jpg);
      background-repeat: no-repeat;
      background-position: left top;
      height: 300px;
      width: 1000px;
}

#header {
      background-color: #780904;
      clear: both;
      margin: 0;
      width: 1000px;
}

h1 {
      color: #99919E;
      font-weight: bold;
      font-size: 18pt;
      margin: 0;
      padding: 10px;
}

#sidebar {
/* Height must be adjusted if the sidebar links change */
      background-color: #780904;
      float: left;
      height: 27em;
      width: 200px;
}

h6 {
      background-color: #780904;
      color: #99919E;
      font-weight: bold;
      font-size: 14pt;
      margin: 0;
      padding: 10px 0 10px 15px;
}

h6.indent {
      background-color: #780904;
      color: #99919E;
      font-weight: bold;
      font-size: 14pt;
      margin: 0;
      padding: 10px 0 10px 30px;
}

a:active {
      color: #99919E;
      text-decoration: none;
}
```

```
a:hover {
      color: #99919E;
      text-decoration: overline underline;
}

a:link {
      color: #99919E;
      text-decoration: none;
}

a:visited {
      color: #9B7E71;
      text-decoration: none;
}

#main {
      margin: 0;
      padding: 0 0 0 200px;
      width: 800px;
}

h2 {
      color: #780904;
      font-weight: bold;
      font-size: 16pt;
      margin: 0;
      padding: 10px;
}

p {
      margin: 0;
      padding: 0 0 15px 15px;
      width: 750px;
}

h3 {
      color: #780904;
      font-weight: bold;
      font-size: 14pt;
      margin: 0;
      padding: 10px 0 10px 15px;
}

table {
      background-color: #99919E;
      border: 2px solid #000;
      margin: 30px;
}

td {
      border: 1px solid #000;
      padding: 2pt;
}

ul {
      margin: 0 0 15px 30px;
      padding: 0;
}

select {
      margin: 10pt;
      padding: 5px;
}
```

```
img.in_table {
      display: block;
      margin: 1px auto 1px auto;
}

img.feature {
      margin: 0;
      padding: 10px 75px 10px 85px;
}

#footer {
      background-color: #780904;
      clear: both;
      height: 50px;
      margin: 0;
      width: 1000px;
}

h5 {
      color: #99919E;
      font-size: 11pt;
      margin: 0;
      padding: 10px 0 10px 10px;
}
```

Appendix Four - Stylesheets for HTML5 Version

thesis_stylesheet_v2.css

```
/* This is the stylesheet modified for the version of the
fictional project's website integrating HTML5 elements. It also
includes some select CSS3 styles. Sizes have also been adjusted
from px to em, ex and percentage where appropriate.
Font courtesy of the Google Font Library (URL:
http://www.google.com/webfonts) */
article, body, footer, header, nav, h1, h2, h3, h4, h5, h6, small,
img, p, select {
      /* This resets all the listed HTML elements to ensure
consistent styling. Table and list elements have been deliberately
excepted as resetting causes all cells to collapse into a single
column and removes bullet points. The final line here prevents
Safari from applying text resizing. */
      border: 0;
      margin: 0;
      padding: 0;
      display: block;
      -webkit-text-size-adjust: none;
}

body {
      background: #9B7E71;
      font-family: 'Noticia Text', serif;
      font-size: 16px;
      text-align: left;
      max-width: 1000px;
}

#banner {
      background-image:url(../Pictures/thesis_banner.jpg);
      background-repeat: no-repeat;
      background-position: left top;
      height: 300px;
      width: 1000px;
      /* The following adds a shadow to box elements. */
      -webkit-box-shadow: .2ex 0ex .4ex black; /* Safari 3-4 */
      -moz-box-shadow: .2ex 0ex .4ex black; /* Firefox 1+ */
      box-shadow: .2ex 0ex .4ex black; /* Opera 10.5, IE 9, Safari
5, Chrome */
}

header {
      background-color: #780904;
      clear: both;
      margin: 0;
      width: 1000px;
      /* The following rounds corners on box elements */
      -webkit-border-bottom-left-radius: 1em; /* Safari 3-4 */
      -webkit-border-bottom-right-radius: 1em; /* Safari 3-4 */
      -moz-border-bottom-left-radius: 12em; /* Firefox 1+ */
      -moz-border-bottom-right-radius: 1em; /* Firefox 1+ */
      border-bottom-left-radius: 1em; /* Opera 10.5, IE 9, Safari
5, Chrome */
      border-bottom-right-radius: 1em; /* Opera 10.5, IE 9, Safari
5, Chrome */
      /* The following adds a shadow to box elements. */
      -webkit-box-shadow: .2ex .2ex .4ex black; /* Safari 3-4 */
      -moz-box-shadow: .2ex .2ex .4ex black; /* Firefox 1+ */
      box-shadow: .2ex .2ex .4ex black; /* Opera 10.5, IE 9,
Safari 5, Chrome */
}
```

```
h1 {
      color: #99919E;
      font-weight: bold;
      font-size: 1.55em;
      margin: 0 .5ex 0 .5ex;
      padding: 1ex;
      /* The following adds a shadow to text. The numbers
correspond to horizontal/vertical/blur radius/color */
      -webkit-text-shadow: .2ex .2ex .4ex black; /* Safari 3-4 */
      -moz-text-shadow: .2ex .2ex .4ex black; /* Firefox 1+ */
      text-shadow: .2ex .2ex .4ex black; /* Opera 10.5, IE 9,
Safari 5, Chrome */
}

nav {
/* Height must be adjusted if the sidebar links change */
      background-color: #780904;
      float: left;
      height: 22em;
      margin: 1em 0 1em 0;
      width: 12.6em;
      /* The following rounds corners on box elements */
      -webkit-border-radius: 1em; /* Safari 3-4 */
      -moz-border-radius: 1em; /* Firefox 1+ */
      border-radius: 1em; /* Opera 10.5, IE 9, Safari 5, Chrome */
      /* The following adds a shadow to box elements. */
      -webkit-box-shadow: 0 .2ex .4ex black; /* Safari 3-4 */
      -moz-box-shadow: 0 .2ex .4ex black; /* Firefox 1+ */
      box-shadow: 0 .2ex .4ex black; /* Opera 10.5, IE 9, Safari
5, Chrome */
}

h6 {
      font-weight: bold;
      font-size: 2.25ex;
      margin: 0;
      padding: .5ex 1.25ex .5ex 1.25ex;
      /* see h1 for more on the shadow style */
      -webkit-text-shadow: .2ex .2ex .4ex black;  /* Safari 3-4 */
      -moz-text-shadow: .2ex .2ex .4ex black; /* Firefox 1+ */
      text-shadow: .2ex .2ex .4ex black; /* Opera 10.5, IE 9,
Safari 5, Chrome */
}

h6.indent {
      font-weight: bold;
      font-size: 2.25ex;
      margin: 0;
      padding: .5ex 1.25ex .5ex 2.5ex;
      /* see h1 for more on the shadow style */
      -webkit-text-shadow: .2ex .2ex .4ex black;  /* Safari 3-4 */
      -moz-text-shadow: .2ex .2ex .4ex black; /* Firefox 1+ */
      text-shadow: .2ex .2ex .4ex black; /* Opera 10.5, IE 9,
Safari 5, Chrome */
}

a:active {
      color: #99919E;
      text-decoration: none;
}

a:hover {
      color: #99919E;
      text-decoration: overline underline;
}

a:link {
      color: #99919E;
      text-decoration: none;
}

a:visited {
      color: #9B7E71;
      text-decoration: none;
}

article {
      margin: 0;
      padding: 1em 0 0 13em;
}

h2 {
      color: #780904;
      font-weight: bold;
      font-size: 2.5ex;
      margin: 0;
      padding: 1ex 0 .5ex 1.25ex;
}

p {
      margin: 0;
      padding: 0 0 1em 1em;
      width: 95%;
}

h3 {
      color: #780904;
      font-weight: bold;
      font-size: 2.25ex;
      margin: 0;
      padding: 1ex 0 .5ex 1.5ex;
}

td {
      border: 1px solid #000;
      padding: 2pt;
      width: 12em;
}

ul {
      margin: 0 0 0 3em;
      padding: 0;
      width: 65%;
}

select {
      margin: 2em;
      padding: 0.5ex;
}

textarea {
      margin: 0 0 0 2em;
}

input {
      margin: 0 0 1ex 2em;
}
```

```
input.check {
      margin: 0 .5em 0 2em;
      width: 1.5em;
      height: 1.5em;
}

input.radio {
      margin: 0 .5em 0 2em;
      width: 1.5em;
      height: 1.5em;
}

table {
      background-color: #99919E;
      border: 2px solid #000;
      margin: 1.75em;
}

img.feature {
      margin: 0;
      padding: 1em 10% 1em 10%;
}

img.in_table {
      display: block;
      margin: 1% auto 1% auto;
}

footer {
      background-color: #780904;
      clear: both;
      height: 2.5em;
      margin: 0;
      width: 1000px;
      /* The following rounds corners on box elements */
      -webkit-border-top-left-radius: 1em; /* Safari 3-4 */
      -webkit-border-top-right-radius: 1em; /* Safari 3-4 */
      -moz-border-top-left-radius: 1em; /* Firefox 1+ */
      -moz-border-top-right-radius: 1em; /* Firefox 1+ */
      border-top-left-radius: 1em; /* Opera 10.5, IE 9, Safari 5,
Chrome */
      border-top-right-radius: 1em; /* Opera 10.5, IE 9, Safari 5,
Chrome */
      /* The following adds a shadow to box elements. */
      -webkit-box-shadow: .2ex 0ex .4ex black; /* Safari 3-4 */
      -moz-box-shadow: .2ex 0ex .4ex black; /* Firefox 1+ */
      box-shadow: .2ex 0ex .4ex black; /* Opera 10.5, IE 9, Safari
5, Chrome */
}

small {
      color: #99919E;
      font-size: 1.75ex;
      padding: .75em;
      text-align: right;
}
```

ipad_landscape.css

```
/* This stylesheet is optimized for the iPad in landscape
orientation, based on thesis_stylesheet_v2.css. The CSS3 styles
are from iPhone & iPad Web Design for Dummies unless otherwise
```

```
noted. Sizes have also been adjusted from px to em, ex and
percentage.
Font courtesy of the Google Font Library (URL:
http://www.google.com/webfonts) */
article, body, footer, header, nav, h1, h2, h3, h4, h5, h6, small,
img, p, select {
	/* This resets all the listed HTML elements to ensure
consistent styling. Table and list elements have been deliberately
excepted as resetting causes all cells to collapse into a single
column and removes bullet points. The final line here prevents
Safari from applying text resizing. */
	border: 0;
	margin: 0;
	padding: 0;
	display: block;
	-webkit-text-size-adjust: none;
}

body {
	background: #9B7E71;
	font-family: 'Noticia Text', serif;
	font-size: 16px;
	text-align: left;
	max-width: 1000px;
}

#banner {
	background-image:url(../Pictures/thesis_banner.jpg);
	background-repeat: no-repeat;
	background-position: left top;
	height: 300px;
	width: 1000px;
	/* The following adds a shadow to box elements. */
	box-shadow: .2ex 0ex .4ex black;
}

header {
	background-color: #780904;
	clear: both;
	margin: 0;
	width: 1000px;
	/* The following rounds corners on box elements */
	border-bottom-left-radius: 1em;
	border-bottom-right-radius: 1em;
	/* The following adds a shadow to box elements. */
	box-shadow: .2ex .2ex .4ex black;
}

h1 {
	color: #99919E;
	font-weight: bold;
	font-size: 3ex;
	margin: 0;
	padding: 0 0 .25ex 1ex;
	/* The following adds a shadow to Safari browsers. The
numbers correspond to horizontal/vertical/blur radius/color */
	text-shadow: .2ex .2ex .4ex black;
}

nav {
/* Height must be adjusted if the sidebar links change */
	background-color: #780904;
	float: left;
	height: 25.5em;
```

```
      margin: 1em 0 1em 0;
      width: 13em;
      /* The following rounds corners on box elements */
      border-radius: 1em;
      /* The following adds a shadow to box elements. */
      box-shadow: .2ex .2ex .4ex black;
}

h6 {
      font-weight: bold;
      font-size: 2.5ex;
      margin: 0;
      padding: .5ex .5ex .5ex 1.25ex;
      /* see h1 for more on the shadow style */
      text-shadow: .2ex .2ex .4ex black;
}

h6.indent {
      font-weight: bold;
      font-size: 2.5ex;
      margin: 0;
      padding: .5ex .5ex .5ex 3ex;
      /* see h1 for more on the shadow style */
      text-shadow: .2ex .2ex .4ex black;
}

a:active {
      color: #99919E;
      text-decoration: none;
}

a:link {
      color: #99919E;
      text-decoration: none;
}

a:visited {
      color: #9B7E71;
      text-decoration: none;
}

article {
      margin: 0;
      padding: 1em 0 0 13em;
      color: #000;
}

h2 {
      color: #780904;
      font-weight: bold;
      font-size: 2.5ex;
      margin: 0;
      padding: .5ex 0 .5ex 1.75ex;
}

p {
      font-size: 2.25ex;
      margin: 0;
      padding: 0 1em 1ex 1em;
      width: 99%;
}

h3 {
      color: #780904;
      font-weight: bold;
      font-size: 2.25ex;
      margin: 0;
      padding: 2ex 0 .5ex 2ex;
}

table.bookshelf {
      background-color: #99919E;
      border: 2px solid #000;
      font-size: 2.25ex;
      margin: 1em auto 1em auto;
      width: 24em;
}

table.withimage {
      background-color: #99919E;
      border: 2px solid #000;
      font-size: 2.25ex;
      margin: 1em auto 1em auto;
      width: 36em;
}

td {
      border: 1px solid #000;
      color: #000;
      padding: 2pt;
      width: 12em;
}

ul {
      font-size: 2.1ex;
      margin: -1ex 0 0 2em;
      padding: 0 0 1em 1em;
      width: 85%;
}

select {
      margin: 2ex 0 2ex 4ex;
      padding: 0;
      font-size: 2ex;
}

img.feature {
      margin: auto;
      padding: 1em;
}

img.in_table {
      display: block;
      margin: 4% auto 4% auto;
}

textarea {
      font-size: 1em;
      margin: 0 0 0 1.3em;
}

input {
      font-size: .75em;
      margin: 0 0 1ex 1.5em;
}

input.button {
      font-size: 1em;
      margin: 0 -1em 0 2em;
}

input.check {
      height: 2em;
      margin: 0 .5em 1em 1.5em;
      width: 2em;
}

input.radio {
      height: 2em;
      margin: 0 .25em 1em 1.5em;
      width: 2em;
}

input.tagbutton {
      font-size: 1em;
      margin: 0 0 2em 2.25em;
}

footer {
      background-color: #780904;
      clear: both;
      height: 2em;
      margin: 0;
      width: 1000px;
      /* The following rounds corners on box elements */
      border-top-left-radius: 1em;
      border-top-right-radius: 1em;
      /* The following adds a shadow to box elements. */
      box-shadow: .2ex 0ex .4ex black;
}

small {
      color: #99919E;
      font-size: 1.5ex;
      padding: .75ex .75ex 0 0;
      text-align: right;
}
```

ipad_portrait.css

```
/* This stylesheet is optimized for the iPad in portrait
orientation, based on thesis_stylesheet_v2.css. The CSS 3 styles
are from iPhone & iPad Web Design for Dummies unless otherwise
noted. Sizes have also been adjusted from px to em, ex and
percentage.
Font courtesy of the Google Font Library (URL:
http://www.google.com/webfonts) */
article, body, footer, header, nav, h1, h2, h3, h4, h5, h6, small,
img, p, select {
      /* This resets all the listed HTML elements to ensure
consistent styling. Table and list elements have been deliberately
excepted as resetting causes all cells to collapse into a single
column and removes bullet points. The final line here prevents
Safari from applying text resizing. */
      border: 0;
      margin: 0;
      padding: 0;
      display: block;
      -webkit-text-size-adjust: none;
}
```

```
body {
      background: #9B7E71;
      font-family: 'Noticia Text', serif;
      font-size: 16px;
      text-align: left;
      max-width: 1000px;
}

#banner {
      background-image: url(../Pictures/thesis_banner.jpg);
      background-repeat: no-repeat;
      background-position: left top;
      height: 300px;
      width: 1000px;
      /* The following adds a shadow to box elements. */
      box-shadow: .2ex 0ex .4ex black;
}

header {
      background-color: #780904;
      clear: both;
      margin: 0;
      width: 1000px;
      /* The following rounds corners on box elements */
      border-bottom-left-radius: 1em;
      border-bottom-right-radius: 1em;
      /* The following adds a shadow to box elements. */
      box-shadow: .2ex .2ex .4ex black;
}

h1 {
      color: #99919E;
      font-weight: bold;
      font-size: 4.5ex;
      margin: 0;
      padding: 0 .25ex 0 15.5ex;
      text-indent: -12.5ex;
      /* The following adds a shadow to text. The numbers
correspond to horizontal/vertical/blur radius/color */
      text-shadow: .2ex .2ex .4ex black;
}

nav {
/* Height must be adjusted if the sidebar links change */
      background-color: #780904;
      float: left;
      height: 31.5em;
      margin: 1em 0 1em 0;
      width: 18em;
      /* The following rounds corners on box elements */
      border-radius: 1em;
      /* The following adds a shadow to box elements. */
      box-shadow: .2ex .2ex .4ex black;
}

h6 {
      font-weight: bold;
      font-size: 3.4ex;
      margin: 0;
      padding: .5ex .5ex .5ex .75ex;
      /* see h1 for more on the shadow style */
      text-shadow: .2ex .2ex .4ex black;
}
```

```
h6.indent {
      font-weight: bold;
      font-size: 3.4ex;
      margin: 0;
      padding: .5ex .5ex .5ex 2.5ex;
      /* see h1 for more on the shadow style */
      text-shadow: .2ex .2ex .4ex black;
}

a:active {
      color: #99919E;
      text-decoration: none;
}

a:link {
      color: #99919E;
      text-decoration: none;
}

a:visited {
      color: #9B7E71;
      text-decoration: none;
}

article {
      margin: 0;
      padding: 1em 0 0 18em;
      color: #000;
}

h2 {
      color: #780904;
      font-weight: bold;
      font-size: 3.4ex;
      margin: 0;
      padding: 1ex 0 .25ex 1.6ex;
}

p {
      font-size: 3ex;
      margin: 0;
      padding: 0 1em 1ex 1em;
      width: 95%;
}

h3 {
      color: #780904;
      font-weight: bold;
      font-size: 3.2ex;
      margin: 0;
      padding: .75ex 0 .25ex 1.75ex;
}

table {
      background-color: #99919E;
      border: 2px solid #000;
      color: #000;
      font-size: 3ex;
      margin: 1em auto 1em auto;
      width: 24em;
}

/* This class attribute modifying <table> would not detect in
portrait mode. In addition, the ISBN field of that table displays
```

```
as the same colour as the background, despite specifying the
colour as #000, on the iPad. When this stylesheet is viewed on a
desktop or laptop computer, the ISBN views as intended. */
table.bookshelf {
      background-color: #99919E;
      border: 2px solid #000;
      color: #000;
      font-size: 3ex;
      margin: 1em auto 1em auto;
      width: 24em;
}

table.withimage {
      background-color: #99919E;
      border: 2px solid #000;
      font-size: 3ex;
      margin: 1em auto 1em auto;
      width: 28em;
}

td {
      border: 1px solid #000;
      padding: 2pt;
      width: 11em;
}

ul {
      font-size: 3ex;
      margin: -1ex 0 0 2em;
      padding: 0 0 1em 1em;
      width: 85%;
}

select {
      margin: 2ex 0 2ex 4ex;
      padding: 0;
      font-size: 2.4ex;
}

img.feature {
      margin: auto;
      padding: 1em;
}

img.in_table {
      display: block;
      margin: 4% auto 4% auto;
}

textarea {
      font-size: 1em;
      margin: 0 0 0 .9em;
}

input {
      font-size: .75em;
      margin: 0 0 1ex 1em;
}

input.button {
      font-size: .85em;
      margin: 0 0 0 2em;
}
```

```
input.check {
      margin: 0 .5em 0 1em;
      width: 1.75em;
      height: 1.75em;
}

input.radio {
      margin: 0 0 0 1em;
      width: 1.75em;
      height: 1.75em;
}

input.tagbutton {
      font-size: 1.25em;
      margin: 1ex 0 0 2.25em;
}

footer {
      background-color: #780904;
      clear: both;
      height: 2.5em;
      margin: 0;
      width: 1000px;
      /* The following rounds corners on box elements */
      border-top-left-radius: 1em;
      border-top-right-radius: 1em;
      /* The following adds a shadow to box elements. */
      box-shadow: .2ex 0ex .4ex black;
}

small {
      color: #99919E;
      font-size: 2.5ex;
      padding: .25em 1em 0 0;
      text-align: right;
}
```