December 2002

THE EVOLVING WEB DEVELOPMENT LIFE CYCLE: A STAGE MODEL PROPOSITION

Sharon Tabor
Boise State University

THE EVOLVING WEB DEVELOPMENT LIFE CYCLE: A STAGE MODEL PROPOSITION

Sharon W. Tabor
Boise State University
stabor@boisestate.edu

Abstract

Over the past few years we have seen phenomenal growth in the use of the public infrastructure of the Internet and the World Wide Web as the delivery mechanisms for e-commerce applications. During that time, many Web sites evolved from simple information repositories to sophisticated front-end systems that interact with complicated back-end transaction processing systems. Along with this evolution, development tools, processes, and methodologies have changed also. Developers who used deceptively simple tools for rapid Web deployment in the early days of e-commerce have met reality in the form of changing business models, input from diverse stakeholders, and increasingly complex issues of performance, security, and extensibility over a distributed network. Organizations with major commitments to move beyond e-commerce and become fully network enabled have had to adapt processes and methodologies from the traditional systems development world. This research explores Web development methodologies in current practice and seeks to define a differentiated Web Development Life Cycle (WDLC) that incorporates the appropriate level of process to the evolving world of Web applications.

Introduction

Web application development has been perhaps one of the fastest moving targets in recent IT history. From humble beginnings with HTML and text editors, serious Web applications are now being developed in sophisticated environments such as IBM's WebSphere and Microsoft's .net family, with help from database manipulation tools such as ColdFusion, and a multitude of evolving standards. An interesting and challenging characteristic of our time, however, is that many of today's Web projects are initiated outside of IT. The cross-functional nature of 21st century businesses means stakeholders come together from marketing, planning, and various other departments to conceive and implement what Straub and Watson (2001) call the net-enabled organization (NEO). This wider view of what technology can accomplish encompasses the e-commerce activity of yesterday and moves the organization to new levels of electronic relationships with customers, business partners, and suppliers, through increasing use of networking technologies. Many IT organizations are finding they must become more involved in broader business issues to drive IT strategy, rather than being driven, or in some cases, pulled.

It is not difficult to understand how Web projects are viewed as being different from traditional systems. The tools are different and easier to learn, often resulting in non-technical users playing a role in the development, deployment, and ongoing life cycle of these strategic business systems (Meister et al., 2000). The very nature of the final product creates challenges for interface design and usability. In what other development environment do we not have some knowledge and control over our end users' environment? Our developers may not be seasoned programmers, but new comers with hot skills and little production experience. The temptation might be to consider Web development so different that organizations avoid the discipline of traditional systems development methodologies. Over time and with increased Web application complexity organizations are struggling to find a workable modified process, labeled for simplicity, the Web Development Life Cycle (WDLC). Successful organizations are finding that process and discipline is needed in Web development, without removing the flexibility to support our increasingly net-enabled organization.

Current Literature and Research Framework

Various researchers are looking at the differences between Web and traditional application development. Bajaj and Siau (2000) suggest that model-based methodologies from the object-oriented development segment offer a logical choice for Web developers.

Wells (2000) reviews the traditional SDLC and presents a summary of research issues for e-business development for each of the traditional phases. Pant, Sim and Hsu (2001) recommend a new Web application planning methodology based upon process reengineering. Others examine Web development from a systems or engineering perspective. Critics might say that Web development is not new, and that its methodology is a given. It has not been that simple in terms of implementation, however, and particularly in larger organizations, has presented challenges such as deadlines directed by non-technical staff, and incomplete deployment issues.

Truex et al. (1999) elaborate on current development trends in what they call "emergent" organizations (p.119). The significant change and flexibility required by new economic realities causes these organizations to value stability less, so they are continually adapting or emerging from their old environments. System stability was a major goal of the traditional SDLC, emphasizing the need for new development processes. Other goals of traditional development also become problematic for the newly flexible organization:

1.   Where formal, lengthy analysis and design was necessary to minimize maintenance activities, Web systems have a shorter useful life and the maintenance cycle is shortened by the next generation system.

2.   While achieving user satisfaction was a primary goal of traditional systems, Web user needs are always evolving.

3.   Where abstract requirements definitions needed to be determined, Web requirements are expected to evolve.

4.   While we expected complete specifications to flow from requirements, changing e-business models mean the Web system must remain flexible.

5.   Where IS development traditionally required rigorous advance planning to achieve stability, the new reality tells us change is good, and our Web systems only approach stability before the next deployment is necessary.

Analysis and design phases may be reduced in time and cost by about half, according to Truex et al., (1999), and lengthy analysis and design becomes a poor investment for competitive applications in today's organizations.

The phases of the traditional systems development life cycle (SDLC) are represented in various ways, but typically include analysis, design, code and test, implementation, and maintenance, often with a requirements phase on the front, and design split into architectural, logical, and physical (Hoffer et al., 1999; Kendall and Kendall 1999). This traditional waterfall model implies that each phase is a distinct step that results in a deliverable before the next phase begins. As far back as the 1980's, however, Naumann and Jenkins (1982) introduced the concept of prototyping to offer a more flexible development method for the emerging CASE tools of the day. The concept of try and test before the final system is built is a better option for Web development than a waterfall, but perhaps not quite as dynamic as needed to meet changing business needs and the reality of a maturing user population.

Scharl (2000) further notes that the dynamic nature of electronic markets creates a conflict with "the long life span and low volatility of corporate information systems" (p. 33). Today's organizations have to continuously adapt to rapidly shifting environments, resulting, he states, in a need to radically rethink Web development. Scharl's solution is a "continuous and evolutionary….iterative cycle of evaluation and refinement" (2000, p.34). His very thorough work on the topic discusses the transition of Web development from static to interactive, and uses Roger's (1995) diffusion framework to explain its adoption. An ongoing thread in his work is the increasingly important role of communication in the evolution of Web applications.

Countless examples are available that show traditional processes are often no longer appropriate, and that many Web development efforts are initiated or influenced by others outside a technical workgroup. A major clothing manufacturer, for example, spent six months and $500k on an outsourced, marketing-driven Web site project, targeted not toward transactions, but toward building community. The virtual distribution arm of a global memory manufacturer found that an early Web project deliverable date was governed by the marketing department's purchase of advertising space, without their prior input. The Web team of a major corporate development shop forgot to allow time for the DNS network to distribute a new image and stood helplessly by while their upper management unveiled a cached version to the media. The Web opens up new and interesting opportunities, but doesn't let us hide our sins as easily as internal projects. A lack of involvement puts IT at a disadvantage in the organization, and missing processes born of inexperience can be painful.

If we consider the practicalities of developing Web applications, a stage model approach may be most useful for developing a Web methodology in organizations emerging into a new business model. Reflecting back to Nolan's (1972) use of the stage model hypothesis to explain management's desire to exploit technology for the sake of the business, the growing trend toward NEOs shows that we are still striving to maximize the benefits of technology as discussed thirty years ago. Nolan and Gibson further used the stage context to demonstrate application maturity, personnel specialization, and techniques used to manage technology.

It is very possible that the stage of an organization's involvement with Web technologies will govern their internal processes and methods. Transaction cost theory also plays an important role as organizations seek to reduce development costs and maximize value chain relationships.

The WDLC Proposal

Clearly, organizations need something less than continuous change, something more than a prototype that is never finished, and a process more flexible than the traditional SDLC. Developers need discipline and processes to tighten project parameters, but flexibility to adapt to the new realities of e-business application development. In evaluating actual case studies of successful e-Commerce Web sites, it appears that the WDLC evolves as Web applications increase in complexity and exposure, and as they become more of a strategic asset for the organization. Part of the increase in site complexity includes links to legacy databases and ties to ERP systems. Today, we can look at the evolution of Web development in a stage model context and see that several variables impact an organization's ability to define and adopt a workable Web Development Life Cycle. It would appear to be related to the organization's goals and efforts in becoming net-enabled, as well as to various internal characteristics. Table 1 considers the variables that must be evaluated to define relationships that impact the related stages of the WDLC.

Table 1. WDLC Impact Matrix

People	Skill sets - Programming, Web design Experience - Traditional development, Web development
Tools	Technical - Open standards User-oriented - Proprietary development environments
Application complexity	Standalone Web - Information, marketing presence Early interaction - Batch processed transactions Real time transaction - Interface to legacy systems, ERP
Time	Fast development - Less than 6 months to deliverable Continuous improvement - Maintenance begins with implementation Rapid technical innovation - Easier to rebuild than add functionality
Organization Maturity	Degree of emergent or NEO status - WDLC supports strategic importance of Web applications

It is also likely that the definition and adoption of a WDLC will impact the ultimate success of the organization's Web applications and its development into a net-enabled organization. Each of the above variables is further impacted by the organization's maturity, or evolutionary stage in moving to e-business initiatives.

The Research Question

The research question evolves from the prior research and framework discussed, as well as preliminary conversations with organizations in various stages of commitment to Web applications. The question is how a WDLC, affected by people, tools, application complexity, and time, is defined and adapted in relation to an organization's maturity or stage in becoming a net-enabled organization. Graphically, the resultant model might appear something like figure 1, with the organization's stage of maturity as a point on the continuum from Web applications as experimentation to Web applications as strategic asset.

Study Methodology

The study uses both case and survey methodologies to gather data. Case study is appropriate when looking at new or evolving areas according to Yin (1994), and survey methods allow quantitative analysis of key variables. A formal case protocol is applied to each of the three organizations studied, and a Web-based survey is used to expedite data handling. A preliminary interview questionnaire will gather information about past, current, and future Web application development goals in three diverse organizations in the semi-conductor, grocery, and agricultural industries. Interviews with key management provide the background on the organization's progression with Web application development and determine their current goals in moving into an e-business or net-enabled business model. A preliminary Web survey was tested and revised as needed to gather data from the case companies about Web development methodologies and development team characteristics to measure the dependent variables of interest. From the data obtained, the research question will be tested and the model revised as necessary.
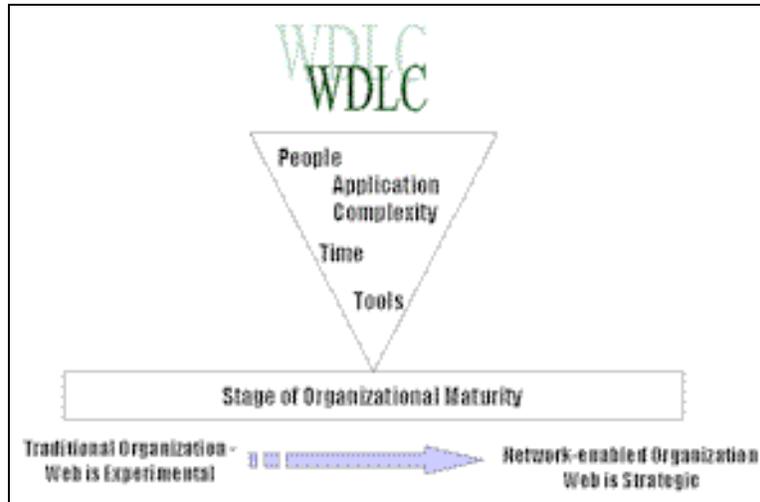
Figure 1. WDLC and Stage of Organizational Maturity

Results of this research should contribute to the IT literature by identifying a workable Web development methodology and life cycle characteristics that can help organizations in early stages of Web experimentation. Additionally, the WDLC can be used in the classroom along with the discipline of tradition methods for those teaching Web application or e-commerce courses. As organizations grow into NEOs they will need new talent that has both technical skills and a thorough understanding of this new era of Web application development, and our challenge as always, is to help prepare those future developers.

References

Bajaj, A. and Siau, K. "E-Commerce Systems: Architecture, Infrastructure, Model and Development Methodology," Proceedings of the 2000 Americas Conference on Information Systems, Long Beach, August, 2000, pp.136-139.

Gibson, C.F., and. Nolan, R.L. "Managing the Four Stages of EDP Growth," Harvard Business Review, January/February, 1974, pp. 76-84.

Hoffer, J.A., George, J.F. and Valacich, J.S. Modern Systems Analysis and Design, Reading: Addison-Wesley, 1999.

Kendall, K.E. and Kendall, J.E. Systems Analysis and Design. Upper Saddle River: Prentice Hall, 1999.

Meister, F., Patel, J. and Fenner, J. "Not Just for Tech Whizzes," InformationWeek (810), October 30, 2000, pp. 83-103.

Naumann, J.D. and Jenkins, A.M. "Prototyping: The New Paradigm for Systems Development." MIS Quarterly (6:3), 1982, pp. 29-44.

Nolan, R.L. "Managing the Computer Resource: A Stage Hypothesis," Communications of the ACM, (16:7), 1972, pp.399-405.

Pant, S., Sim, H.T. and Hsu, C. "A Framework for Developing Web Information Systems Plans: Illustration from Samsung Heavy Industries Co., Ltd.", Information and Management (38:6), 2001, pp.395-408.

Rogers, E.M. Diffusion of Innovations, New York: The Free Press, 1995.

Scharl, A. Evolutionary Web Development, London: Springer-Verlag, 2000.

Straub, D.W. and Watson, R.T. "Research Commentary: Transformational Issues in Researching IS and Net-Enabled Organizations," Information Systems Research (12:4), 2001, pp.337-345.

Truex, D.P., Baskerville, R. and Klein, H. "Growing Systems in Emergent Organizations," Communications of the ACM (42:8), 1999, pp. 117-123.

Yin, R.K. Case Study Research: Design and Methods, 2nd Edition, Sage Publications, Inc., Thousand Oaks, CA, 1994.