Communications of the Association for Information Systems

Volume 5 Article 7

January 2001

The W Life Cycle Model and Associated Methodology for Corporate Web Site Development

Lei-da Chen
Creighton University, lchen@creighton.edu

Linda B. Sherrell
University of West Florida, lsherrel@cs.uwf.edu

Recommended Citation
Chen, Lei-da and Sherrell, Linda B. (2001) "The W Life Cycle Model and Associated Methodology for Corporate Web Site Development," *Communications of the Association for Information Systems*: Vol. 5 , Article 7.
DOI: 10.17705/1CAIS.00507
Available at: https://aisel.aisnet.org/cais/vol5/iss1/7

Communications of the Association for Information Systems

Volume 5, Article 7
April 2001

THE W LIFE CYCLE MODEL AND ASSOCIATED METHODOLOGY FOR CORPORATE WEB SITE DEVELOPMENT

Linda B. Sherrell
Department of Computer Science
University of West Florida

Lei-da Chen
College of Business Administration
Creighton University

lchen@creighton.edu

ELECTRONIC COMMERCE;
RESEARCH

THE W LIFE CYCLE MODEL AND ASSOCIATED METHODOLOGY FOR CORPORATE WEB SITE DEVELOPMENT

Linda B. Sherrell
Department of Computer Science
University of West Florida

Lei-da Chen
College of Business Administration
Creighton University

lchen@creighton.edu

ABSTRACT

The last few years witnessed the increasing internal and external use of the Internet by organizations. Web sites grew in sophistication from conventional sites composed of a simple collection of Web pages for public relations or marketing to complex Web information systems dealing with business-to-customer transactions or business-to-business networks. However, most organizations still do not have a formal process of Web site development, and corporate Web sites are often developed in an unorganized and uncoordinated fashion. The results of this chaotic situation include slow delivery, conflicting standards, discrepancies with respect to corporate objectives, and redundant development efforts.

To help alleviate this situation, this paper presents the W software life cycle model for corporate web site development along with an associated methodology to guide Web development groups in their endeavors. The proposed process model and methodology are based on insights gleaned by studying development projects for three types of corporate Web sites: intranets, Web-presence sites, and transactional sites. However, because Web information systems are more complex, the article also provides

recommendations for how the methodology can be adapted to handle these types of applications better.

Keywords: W life cycle model, Web site construction, software development methodologies, joint application design, prototyping, incremental process model

I. INTRODUCTION

In recent years, the Internet presented almost unlimited potential to business communities around the world. Many businesses realized the Internet's ability to reach large audiences at a relatively low cost and became more and more active on the Web. A late 1990's survey showed that businesses doubled their usage of the Internet from 1996 to 1997, and approximately forty percent of business owners changed the way that they did business because of the Internet [Facilities Design & Management, 1997]. Ivins [1998] reported that the population of Web users in the U.S. in 1998 was over 57 million, while a 2000 report projects the number of Internet users worldwide as 362 million by the year 2003, with two thirds of these users living outside the U.S. [Yorgey, 2000].

The openness of the Internet creates opportunities for virtually all companies ranging from small start-ups to Fortune 100 companies. It is widely utilized to establish general as well as niche marketsolutions [Terdoslavich, 1997]. The scope of Internet-based applications can be divided into four general categories:

1. Intranets, corporate internal networks,
2. Web-presence or conventional sites, public corporate web sites for marketing, public relations, and other purposes,
3. Electronic commerce systems, transaction oriented business-to-consumer web sites, and
4. Extranets, dedicated business-to-business networks [Isakowitz, Bieber, and Vitali, 1998].

In terms of specific functions, businesses use the Internet as communication, transaction, and distribution channels [DeVille, 1995; Anderson and Choobineh, 1996]. The most popular usage of the Internet is as a communication channel. Because of its broad reach, commercial or noncommercial information is distributed around the world at relatively low cost. The second most popular practice is as a transaction channel to facilitate sales activities between buyers and sellers. The least tapped function of the Internet is as a distribution channel handling the physical exchange of products and services for information goods. These attractive features pressure corporations to develop a presence on the Internet.

However, most organizations do not have a formal process for Web site development, and corporate Web sites are often implemented in an unorganized and uncoordinated fashion. Part of this lack may be the result of only a small amount of the trade literature discussing the development of corporate Web sites (e.g. Millman, 1997; Sallquist, 1996; Fassett, 1995; Strehlo, 1996a, 1996b), and the scarcity of academic journal articles. A notable exception is Wiegers' [1999] paper that describes Kodak's Web development group activities to improve their process.

Our paper attempts to fill this void by defining the W software life cycle model together with an associated methodology for corporate Web site development. The purpose of this new software process model is to furnish web developers with an overall framework that describes the required phases in constructing and maintaining corporate Web sites. The aim of the methodology is to reduce the pitfalls of corporate Web site construction and to increase the completeness, compatibility, and quality of resulting sites. Using the proposed software process model and following the steps of its associated methodology should systematically lead an organization's development team through its various corporate Web site endeavors.

The next section of this paper discusses further issues in corporate Web site development and the importance of using a methodology. Section III describes Web information systems development and its differences from conventional Web site development. Section IV provides the basis for the W software life cycle model, defines the model, and gives a detailed description of each step in the associated corporate Web site development (CWSD) methodology. Finally, Section V presents our conclusions and the implications of this work for both practitioners and researchers.

II. CORPORATE WEB SITE DEVELOPMENT

As the Internet becomes crowded, the question is no longer whether an organization should develop a presence on the Internet, but rather how to create and maintain a site that delivers enough value for both its customers and itself. The national director of retail and consumer products industry services for Ernst & Young (Fred Crawford), identified six criteria for a successful Web site: knowledge, intuition, interaction, community, customization and currency [Lewis, 1998]. However, the development of a corporate Web site that meets these criteria can be quite expensive. Prices quoted for small sites range from $10,325 to $92,500, whereas the costs for some high-end sites are well over $1 million [Advertising Age's Business Marketing, 1996; Gantz, 1996]. In addition, ActivMedia projected total spending of $22 billion for corporate Web development and maintenance for the year 2000 [ActivMedia, 2000]. Therefore, the development process of a corporate Web site must be carried out with great control and careful planning.

At present, the three development options available for corporate Web sites are:

1. Design, implement, and maintain a Web site in-house.
2. Outsource all of the work to an interactive marketing company (IMC).

3. Share the work with an IMC [Fassett, 1995].

Outsourcing and partial outsourcing are attractive options for most organizations in Web site development because of the range of expertise required [Reinbach, 1996]. The choice of the development options depends on such factors as a company's willingness to become involved in the project, the perceived value of the Web site, the availability of expertise, and cost .

The development of a corporate Web site is viewed as either a software development project [Strehlo, 1996a] or as an end-user system [Goff, 1996]. In the software development view, the Information Systems (IS) department should lead the development process and work closely with other departments to implement the vision. End-user system proponents believe that functional business areas should be in charge of development because they know exactly what they want (e.g., promotion of new products, enhancement of the corporate image). In fact, with the help of easy-to-use Web authoring tools, the end-users can develop most of a conventional Web site by themselves. Therefore, the contribution of the IS department is mostly advisory in nature with IS department members providing both an understanding of the Internet and an effective use of the technologies available. While the software view emphasizes the need for rigorous controls over the process, it may result in a serious backlog of projects and lower user satisfaction. The problem with the second view is that most end-users lack the experience in formal systems analysis and development. Moreover, sites developed independently by different users within a firm often do not comply with a corporate standard; this situation increases the likelihood of a lengthy process of integrating the sites into a common corporate Web strategy, infrastructure, and identity in the future [Gantz, 1996].

No matter which view of Web site development a company takes, a well-designed methodology must be in place to overcome the difficulties and to help ensure the success of the project. The methodology should complement the entire system development life cycle (SDLC) model by specifying the step-by-

step tasks for each phase, the roles of the individuals and groups involved, and the development techniques for each task [Whitten et al., 1994]. Specific requirements for a corporate Web site methodology are:

1. It must be applicable to any type of corporate Web site (intranet, Web-presence, transactional, or extranet), and it must address all relevant issues.
2. It must be sufficiently rigorous to ensure the completeness, compatibility, and quality of the Web site, yet easy enough for end-users to follow.
3. It must be flexible enough to fit into any of the development options discussed above.

Most attempts to design a methodology for corporate Web site development [Millman, 1997; Sallquist, 1996; Fassett, 1995; Strehlo, 1996a], merely identify some common steps in building and maintaining Web sites, and some of the approaches are overly simplistic. For example, one article refers to Web site development as a three- phase process: design, implementation, and maintenance [Fassett, 1995]. While this breakdown is useful in the early stages of the exploration of a new topic, it does not constitute a comprehensive and viable methodology that meets the criteria discussed above. Furthermore, with both the increasing pressure for a corporate presence on the Internet due to intense competition and the escalation of Web site development costs, the need for a corporate Web site development methodology to provide process guidelines will become even greater.

III. WEB INFORMATION SYSTEMS DEVELOPMENT

Over the last few years, companies began to use the Web to link buyers, sellers, and business partners in innovative ways [Tenenbaum, 1998]. This movement is expected to continue. For example, Lewis [2000] forecasted that by

2002, the biggest trend on the Internet will involve old-economy corporations (e.g., utility companies) establishing Web sites to handle customer services. A Cap Gemini/International Data Corporation survey of 300 global companies found that 95 percent of these companies will have established Web sites by 2001, with 28 percent of them using electronic payment by 2003 [Trepper, 2000].

To facilitate electronic commerce, a transaction oriented Web site, also called a Web information system (WIS) is needed. Moreover, WISs are predicted to be the systems for the next generation of business data processing [Press, 1999]. A WIS is a system that supports the internal or external work of an organization. The accomplishment of its transactions often involves interacting with traditional systems (databases and applications) [Isakowitz et al., 1998]. A WIS complements a conventional Web site to provide an organization with a number of benefits. Among the most common benefits are enhancing competitiveness or creating strategic advantages, enabling easier access to information, and providing new products or services to customers [Lederer, Mirchandani, and Sims, 1998]. Despite these benefits, the design and implementation of a WIS presents many challenges to its developers because of the differences between a WIS and a conventional Web site. Takahashi [1998] highlights a number of these differences, which can be summarized as:

1) a WIS supports workflow through constant interactions with users, and
2) structured data modeling and referential integrity are vital to a WIS.

Therefore, while following the steps of the methodology proposed in this paper, WIS developers may choose to emphasize certain phases of the development process to address the challenging issues. Specific recommendations for corporate web site development (CWSD) from the literature are discussed below.

Although parts of a corporate Web site can be loosely coupled, a WIS requires seamless integration between workflow, applications and documents. Besides being a technological challenge, developing a WIS is also "an organizational, political, and cultural issue" [Balasubramanian and Bashian,

1998]. Determining the requirements of a WIS is a demanding job, and extra attention must be given to the requirements analysis phase of the software development life cycle. Ultimately, the overall goal of WIS developers is to design and implement a system architecture that supports the distributed collaboration work processes of an entire organization.

The work of a WIS is often accomplished with the help of existing non-Web systems. Therefore, an additional challenge to WIS developers is interfacing a new software system with legacy systems. Kambil and Ginsburg [1998] describe various obstacles that they encountered while interfacing a particular WIS with a legacy application and offer some solutions based on their experience. In particular, to minimize the interfacing problem, they suggest that special attention be paid to the determination of database/program requirements during requirements analysis.

Whether the WIS serves as an electronic shopping environment or a corporate intranet application, the design of the user interface is extremely important due to the high volume of user interactions. Because most users have high expectations regarding the ease of use of all interactive systems, usability tests are especially critical when developing a WIS.

The software process model proposed in this paper is a generic model for virtually all types of corporate Web sites. The methodology describes the steps associated with planning, constructing, and maintaining these sites. However, when developing Web information systems (as opposed to conventional web sites), project teams may wish to emphasize certain phases of the methodology. Furthermore, designers may refer to Lohse and Spiller [1998] or Rossi et al. [1999] for guidelines on improving hypermedia information design for Web information systems.

IV. THE W LIFE CYCLE MODEL AND ASSOCIATED METHODOLOGY

The W model and its associated methodology for corporate Web site development (CWSD) are based on four primary sources:

First, the authors were involved in many Web site or Web-based system development projects during the last five years. These projects took place in both real world organizations and classroom environments.

Second, reviews of additional projects and interviews with their respective project managers revealed a wide range of issues that were taken into account during the development of the W model. The experiences and knowledge generated from these first two sources provided the building blocks of the W model and its methodology.

Third, the analysis by Gillenson, Sherrell and Chen (2000), which is based on a study of 300 commercial Web sites, gave us insights about the varying structures of corporate Web sites, the differences between Web-presence and transactional Web sites, and the contrasts between Web-site and traditional information system development projects. The Gillenson study took a bottom-up, three-level approach to construct a taxonomy of Web site traversal patterns and structures, and the article provides developers with useful information about the inner workings of a Web site to utilize this medium better .

Fourth, software engineering theories provided the glue to integrate our previous experiences and observations in order to form the current W model and methodology. The system development life cycle (SDLC) or waterfall model, the V model [German Ministry of Defense, 1992], rapid application development (RAD), and joint application development (JAD) are just a few of the software life cycle models and techniques that provide a solid theoretical foundation to the W model.

Some of the corporate Web-site development projects from sources one and two used variations of the W model as a guide, and improvements to both

the model and the corresponding methodology were made when problems related to the execution of the methodology were encountered.

Table 1 summarizes the major problems experienced during the development process and our recommended solutions. These solutions are reflected in the W model, because one of our goals is to provide Web developers with a specialized software life cycle model that will reduce the occurrence of the problems listed in Table 1. Among the various types of problems, people issues were observed most frequently. Keeping user expectations at reasonable levels, avoiding scope creep, increasing user acceptance, and resolving political issues were all considered vital to the success of a Web-based development project. In fact, if these issues are not treated cautiously, they inhibit a project from being completed successfully within its allotted time and budget constraints. Moreover, well-crafted corporate Web site standards were found to proactively prevent some project development problems.

Table 1.Summary of Observations from Case Projects

Types of Projects	**Major Problems Observed**	**Solutions**
Corporate Intranet	Non-uniform appearance of organization's Web site pages	Use top-down planning to develop an overall structure for corporate Intranet. Develop and strictly follow corporate Web site standards throughout the development process.
	Creeping requirements	Hold JAD sessions and frequent walkthroughs to keep users informed of system requirements and project progress.
	Unreasonable user expectations	Hold JAD sessions to allow users to communicate freely with development teams.
	Low user acceptance	Use JAD sessions and prototyping techniques to involve users in the development process.
	Political issues	Have well-defined goals and objectives for sites. Hold frequent user meetings and pre-design meetings to resolve conflicts. Include representatives from management and marketing on JAD teams.
Web-Presence Sites	Low usability	Conduct frequent usability tests.
	Browser incompatibility	Program Web sites with cross-platform technologies Enforce corporate Web site standards.

	Excessive attention to detail - causes developers to lose sight of overall project picture	Outline overall structure of the Web site at early stages. Hold pre-design meetings and frequent walkthroughs to keep developers focused on core concepts and project goals.
	Excessive multimedia elements on Web site	Have well-defined goals and objectives for Web site projects. Conduct performance tests frequently.
	Constantly changing requirements	Use incremental development to better react to changing requirements.
	Budget overruns	Conduct extensive feasibility analysis in the planning phase. Keep Web site features in line with goals and objectives.
	Constant content updates	Include detailed updating procedure and schedule in corporate Web site standards. Draft a maintenance agreement when Web site is developed by IMC.
	Project schedule pressures	Use RAD and incremental development techniques. Prioritize Web site features. Allow users to participate in the development process. Set reasonable schedules / milestones.
Transaction Oriented Sites	Integration difficulties with legacy systems	Develop corporate Web site standards to use mainstream development technology. Conduct extensive research in technologies before adoption. Use vendor support effectively.
	Rapid changes in development technologies and tools	Follow corporate Web site standards in choosing development technologies. Keep development tools in line with project goals and objectives.
	Complexity	Use structured process, data and logic modeling. Spend more time and effort in the analysis phase to benefit the latter stages.
	Extensive maintenance (corrective, adaptive, and perfective)	Keep good documentation during development. Develop formal maintenance request guidelines. Keep detailed maintenance reports.
	Development bottlenecks	Use JAD techniques to resolve bottlenecks. Hire external consultants or use IMC.
	Lack of internal technical capabilities	Use packaged development tools. Provide intensive training for developers. Outsource to external consultants / IMC.
	Project schedule pressures	Use RAD techniques for small projects. Adopt packaged development tools. Prioritize Web site features. Set reasonable schedules / milestones.
	Political issues related to data ownership	Follow corporate Web site standards. Hold frequent user meetings and pre-design meetings to resolve conflicts. Include representatives from management and marketing on JAD teams

Because corporate Web projects have several unique features, the W model and CWSD methodology must necessarily deviate from existing software life cycle models and methodologies that were designed for traditional software development.

- First, since the major portion of a corporate Web site displays company information (e.g. financial statements, product catalogs, etc.), each development project is information-driven.
- Second, the user interface design is particularly important since a Web site is an interactive system.
- Third, an incremental development process is more feasible, because the modules (sections) of a corporate Web site are often loosely coupled and therefore minimally dependent on one another.
- Fourth, client involvement in a project is typically high; moreover, the system may even be partially developed by the client.

Because of the high overhead in planning, designing, implementing, and maintaining corporate Web sites, development projects should follow the steps of a formal software development methodology. As in many software projects, the early stages of the SDLC model can provide a foundation for corporate Web site development; therefore, a great deal of work completed beforehand benefits the latter stages [Dykman and Robbins, 1991].

Both the management information systems and the software engineering literature contain many examples of software life cycle models. Some models prescribe recommended procedures for software development, whereas others describe actual customs [Pfleeger, 1998]. The W software life cycle model (Figures 1 and 2) and its associated methodology provide a framework to bridge this gap between theory and practice in the realm of corporate web site development.

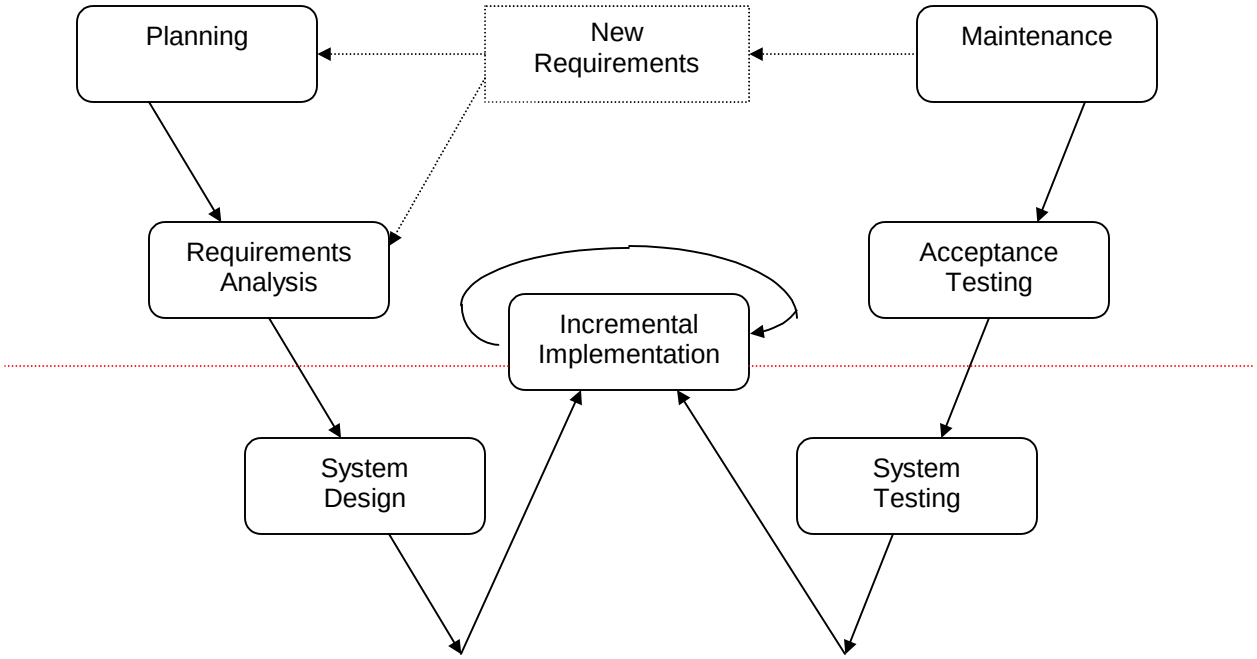

Figure 1. W Life Cycle Model

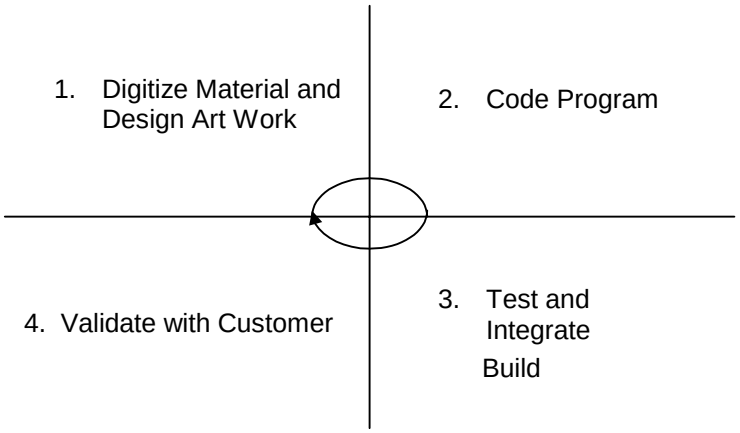

Figure 2. Steps of Incremental Implementation

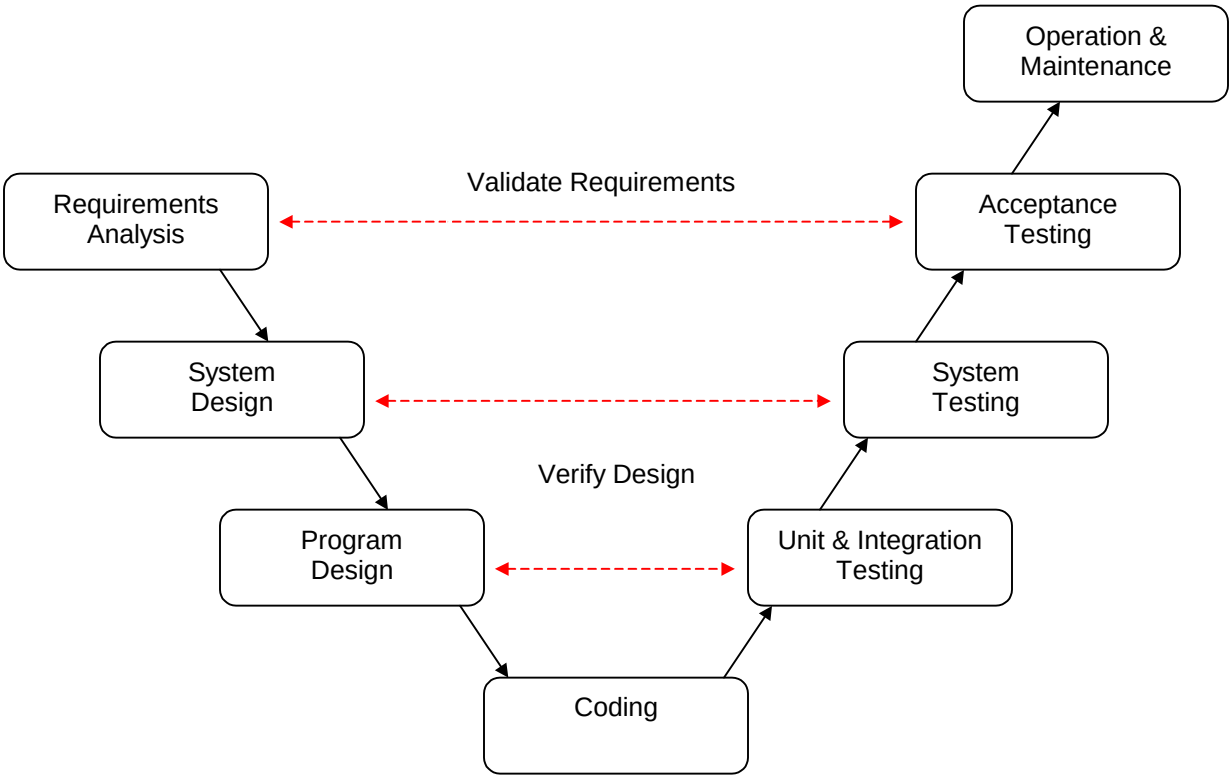

Figure 3. V Model (Adapted from [Pfleeger 1998])

The W model (see Figures 1 and 2) integrates features from several popular process models allowing corporate web site developers to adjust both the model and associated steps of the methodology to meet their specific needs. At its simplest, the W model can be described as an adapted waterfall model for corporate web sites since it explicitly lists deliverables for each step of a project (Table 2). First time corporate web developers may choose to view the W model in this light due to the clarity and discipline enforced by the traditional waterfall model [Boehm, 1981], thereby making it easy to explain to customers. However, the W model is more accurately described as a variation of the V model [German Ministry of Defense, 1992] (Figure 3). Like the V model, the W model emphasizes the correlation between development phases (requirements analysis

Table 2. Corporate Web Site Development (CWSD) Methodology

PHASE	STEP	DELIVERABLE
Planning	1. Identify Web site project	Preliminary work request form
	2. Conduct feasibility study or initial interview	Feasibility report or preparatory definition
	3. Form project team	Personnel task form
Requirements Analysis	4. Outline overall structure	Requirements definition
	5. Filter information and refine requirements	Refined definition + data and process models
	6. Compare with Web site standards	Requirements specification
System Design	7. Hold pre-design meetings	Project schedule
	8. Develop prototype	Prototype evaluation + software specification
Incremental Implementation	9. Construct site using builds	Program code + results of unit tests
System Testing	10. Install and test	Results of test cases
Acceptance Testing	11. Deliver web site	Working site + maintenance request guidelines
Maintenance	12. Maintain web site	Maintenance reports

analysis, system design, and program code) and testing phases (acceptance testing, system testing, and unit testing). The W model includes a prototyping step, which is conducted immediately following the architectural design of the overall system, and an implementation phase that is realized as a series of individual builds as opposed to a staggered incremental model, which is considered to be more risky [Schach, 1997]. For each build, construction follows a 4-step process:

1. digitalization of materials and design of art work,
2. coding of the program,
3. unit testing of models and integration with previous builds, and
4. validation of the product with the customer.

Both the waterfall and V software life cycle models embrace the management aspects of a software project via scheduled deliverables, but neither model adequately addresses the needs of the client or user [Bruegge and Dutoit, 2000]. More specific limitations include the models' dependence on the accurate expressions of requirements by the customer at the start of a project and their

late inspection of a working system. The W model helps to overcome these limitations by advocating a prototyping step following architectural design and the gradual delivery of the system via incremental builds. To emphasize the increased interaction with the client/user, the depiction of the W model includes a dashed line with life cycle phases above the line indicating heavy client involvement and phases below the line denoting little or no client interaction.

This part of the W software life cycle model is similar to the sawtooth model [Rowen, 1990], another variation of the V model. The depiction of the sawtooth model (Figure 4) illustrates the difference in understanding of the system between a client or user and the system developer, while the model itself incorporates two types of prototyping techniques: revolutionary or throw-away

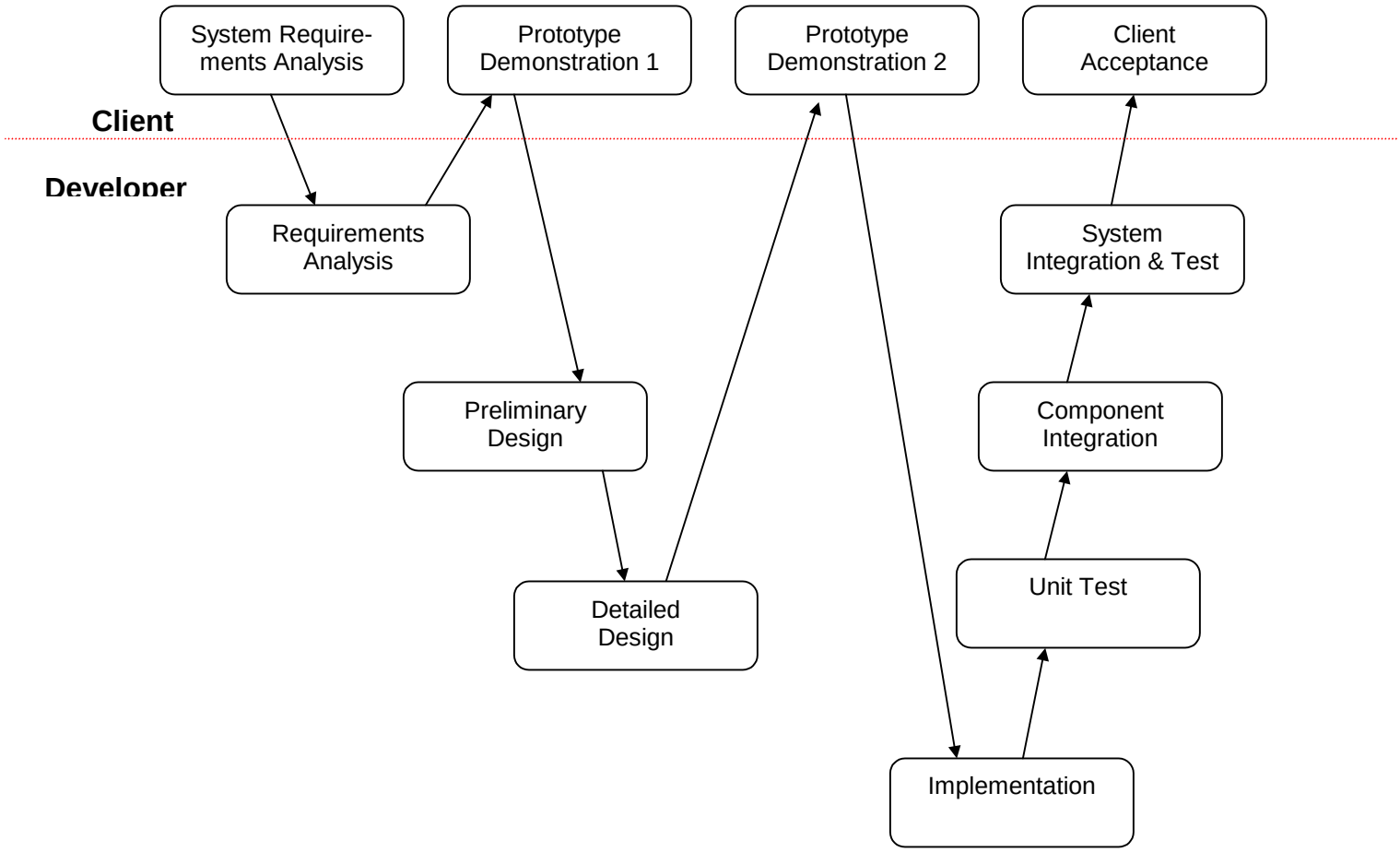

Figure 4. Sawtooth Model (Adapted from Bruegge and Dutoit, [2000])

and evolutionary [Bruegge and Dutoit, 2000]. Despite some similarities in the depiction of the W and sawtooth models, the W model supports the more modern trends of joint application design (JAD) and incremental development. The JAD teams typically includes managers, developers, and other domain experts, as well as client representatives.

Because corporate Web sites can differ enormously in their size, functionality and complexity, the W model is designed so that it can be adapted for particular projects. For example, in the case of small Web site projects or when the client's requirements are especially vague, the model and methodology can be modified to fit into either a RAD software process model [Pressman, 1997] or a rapid prototyping model [Lantz, 1985]. By the same token, when the project risk is high and the development is internal, the process model can be incorporated into Boehm's spiral model [Boehm, 1988].

The next portion of this paper describes the individual steps of each phase of the corporate Web site development methodology. For easy reference, Table 2 delineates the phases, the steps of each phase, and the associated deliverables for each step of the methodology.

PLANNING

Identify Web Site Project

The motivation for a Web site can come from both external and internal sources [Gantz, 1996; Terdoslavich, 1997]. The external sources may be customers' needs or requests caused by pressure from competitors who successfully established their presence on the Internet. Alternately, the increasing demand for information sharing within an organization may lead a company to develop internal Web sites [Friedman, 1998]. In both situations, a company can use the Internet as a marketing tool to promote its products and services. Another type of project request can be for a Web development group (WDG) to add enhancements to a previously developed Web site. At any one

time, there may be several outstanding project requests, so it is helpful if an organization's WDG has a method for prioritizing Web site projects [Wiegers, 1999]. Some possible weighting factors include urgency of need, amount of technology risk, business value, and degree of alliance with the current mission and goals of the organization. The documentation resulting from this development step is a preliminary work request form.

Conduct Feasibility Study or Initial Interview

Once a Web site project is identified, the Web development group needs to decide if a feasibility study should be conducted. The major areas of concern are normally economic justification, technical feasibility, legal considerations, and evaluations of alternative approaches [Pressman, 1997], where the evaluations may also depend on time-related or marketing issues [Berard, 1993]. Probably the most important facet of the feasibility study is cost-benefit analysis. Other areas of interest include long-term corporate objectives and the potential for market growth. After the completion of the feasibility study, the findings and recommendations are documented in a feasibility report. The importance of the feasibility study and corresponding report can not be overstated. For example, a survey of CEOs of companies found that 69% felt that they had not considered the issues sufficiently before undertaking new e-business initiatives, and 34% worried about their efforts failing [Stepanek, 2000].

It should be noted that when the WDG previously developed and successfully maintained a similar type of Web site, a feasibility study might not be warranted. An initial interview with the client, which may be an individual or a department, will still need to be conducted to discover the immediate objectives as well as future goals of the requested site. A simple example of the goal of a Web site would be "to enhance customer services". Both the objectives and goals must be explicitly stated, since they will lead the entire development process [Healthcare Executive, 1997]. Based on these goals, initial requirements for the Web site are identified and documented in a preliminary requirements

document labeled the preparatory definition. In both of these cases, the work request form may also need to be refined at this time.

Form Project Team

Before the development effort gets underway, a project team needs to be formed. Ideally the project team includes a collection of people with the necessary expertise to complete the project successfully. In case there is a lack of expertise in the organization, outsourcing should be considered [Fassett, 1995; Reinbach, 1996]. The size of the project team depends on the scale of the Web site. Generally, the team includes managers from the requesting department to ensure that business goals are met, members of the WDG to serve as technical references, representatives from the software quality assurance team and other specialists such as graphic artists, writers, and programmers. In major Web site development projects, the chief information officer (CIO) and legal counsel are also involved [Healthcare Executive, 1997]. Increasingly, telecommunications managers participate in Web site development and operations [Hamblem, 1997]. Because these managers have substantial experience in supporting networks and in managing customer service functions, they are likely to play an even more prominent role in projects as the distinction between data and voice blurs. It is also helpful to have a risk officer, who is in charge of risk management activities, as a permanent member of the WDG [Wiegers, 1999]. After the project team is formed, a review of both the feasibility report or preparatory definition and the work request form should occur so that tasks can be allocated to specific personnel. The associated deliverable from this step is a personnel task form.

REQUIREMENTS ANALYSIS

Outline Overall Structure

Based on the preliminary requirements specified, which may be part of the feasibility study, the project team outlines the overall structure of the Web site. Furthermore, at this time, the WDG expands the preparatory definition into a

requirements definition, which is a high-level description of the system requirements written primarily for the customer as opposed to a detailed requirements specification targeting the design team [Sommerville, 1996].

The overall structure of the Web site deals with its major functions rather than individual pages. If one considers a Web site as a hierarchy, the general site structure may only decompose to the third level: from Site Directory (Home Page) to Foundation Pages (second level) to third level pages [Gillenson, Sherrell and Chen, 2000]. Since the determination of the overall structure establishes the foundation for further development, it is recommended that all members of the project team participate in this stage of the corporate Web site development process. The participatory or JAD approach not only ensures the completeness of the analysis, but also improves the relationship between the client, who may or may not be a user, the WDG, and other team members [Whitten et al. 1994]. Requirements representation methods for Web site structure vary among different organizations. Figure 5 demonstrates one well accepted representation method. As mentioned previously, the deliverable at this stage is the requirements definition. This document includes the overall site structure and a description of the major functions.

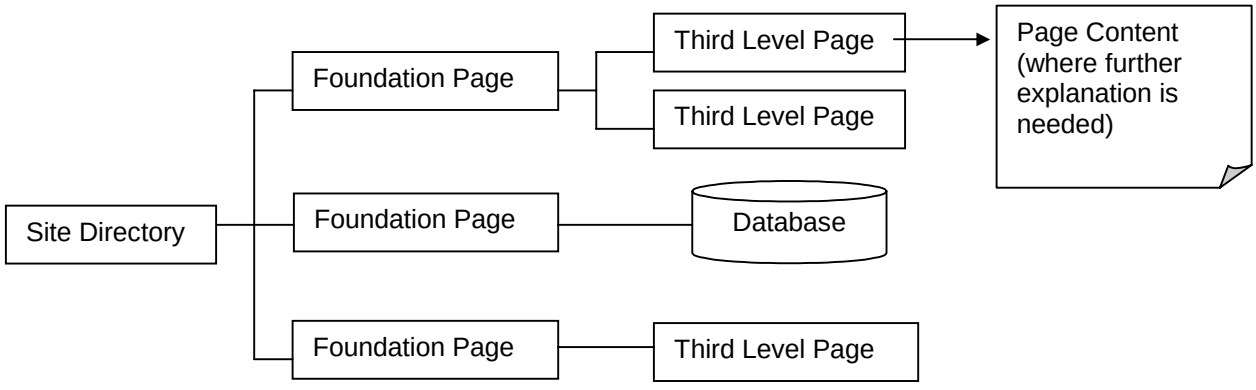

Figure 5. A Sample Web Site Structure

Filter Information and Refine Requirements

Information pertinent to the content of the Web site as delineated in the requirements definition is collected during this stage. The information need not be limited solely to text. Images, audio and video are often used to enhance the content of a Web site [Chau, 1997]. However, managers who represent the user or the client should filter all materials to ensure relevance and suitability.

Some functions of a corporate Web site require the use of specific databases and/or computer programs. For example, a Web-enabled transactional system involves product databases, customer databases, and programs that support the interactions between users and databases. The requirements determination for databases and programs is well covered in the software engineering literature (e.g. [Davis, 1993]), and hence is not described here. The deliverable from this stage of the CWSD methodology is a refined requirements definition containing a section that describes the requirements for databases and programs. Both data and process models often accompany this new section.

Compare with Web Site Standards

A survey of 200 top corporate Web sites found that the majority did not meet basic industry and government standards for access, navigation, and customer interaction [Verton, 2000]. Fundamental features that were missing were privacy statements, action links for customers, text-oriented access, and assistance methods for disabled users. Verton reported that not a single site contained every feature.

Each corporate Web site should meet minimum requirements as outlined in government or industry standards. For example, IEEE Standard 2001 outlines a general set of requirements and procedures to aid Web development groups in engineering their Web sites. It places particular emphasis on practices

to enhance the user value of a site and to automate its maintenance. IEEE Standard 2001 also recommends that each organization adopt and document its own corporate Web site standards [Isaak, 1999]. Documentation can help to ensure that the various parts of an organization's internal and external Web sites share a common corporate strategy, infrastructure, and identity.

If a formal standards document does not already exist, then it needs to be developed at this point in the software life cycle since all standards must be explicitly stated before the actual development work takes place. One standards manual [Thomas & Betts, 1998] for corporate Web development states the following:

> *"This document describes the mandatory guidelines and standards for publishing information via the World Wide Web (WWW). It should be included when creating and designing Web content which will be published on a corporate Internet, Intranet, or Extranet. This document is designed to ensure a level of quality, consistency, and structure of the corporate Web site."*

The Web site standards should minimally include the specifications for hardware and software platforms, design guidelines (e.g., user interface requirements, general layout, logos, and color scheme), testing, and publishing procedures. One important aspect that must be addressed at this point is how the company will handle any disagreements in browser standards. The decision on this issue depends on factors such as user browser preference and the browser dependency level of the Web site content. The corporate Web site standards, which may be a collection of manuals or templates, is developed by the project team with software quality assurance personnel playing a prominent role [Rettig, 1992].

Since future stages of the software project must closely follow the corporate Web site standards, the requirements specification document is written after the standards are reviewed. The requirements specification contains a detailed description of all system functions and serves as a contract between the client and development team [Schach, 1997]. After the requirements specification is written, a specification walkthrough is often performed. The walkthrough team, which consists of client representatives and personnel from the project team and the software quality assurance group, reviews and validates the requirements specification.

DESIGN

Hold Pre-design Meetings

Before the design of the prototype gets underway, pre-design meetings in which all members of the project team participate are important. These meetings allow team members to digest the materials for the Web site, familiarize themselves with the site structure and standards, brainstorm design ideas, allocate responsibilities, and develop the project schedule. Although the deliverable resulting from these activities is the project schedule, both the objectives and functionalities (Figure 6) of the proposed prototype are also discussed at this time.

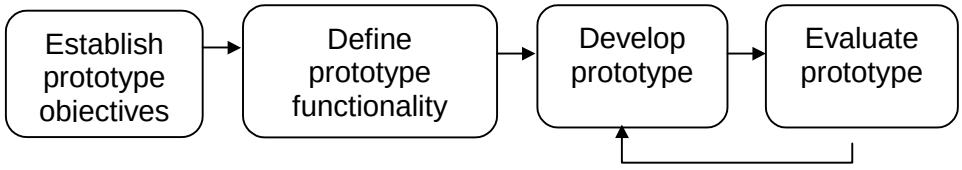

Figure 6. The Prototype Development Process
(Adapted from [Sommerville, 1996)

Develop Prototype.

A prototype is usually a small-scale, representative, or working model of the final system [Whitten et al., 1994] developed to discover or validate user requirements. A prototype can be used to communicate with clients better than

the written specifications developed in the previous stages. This is especially true in the case of Web site development, where user involvement in the development process is high. The prototype increases the clients' understanding of their information needs, reduces communication difficulties, and validates user requirements [Carey, 1990]. Prototyping is suitable for Web site development because:

1. A Web site is an interactive system with increased user involvement in the development process; therefore the builders can respond rapidly to the user's needs.
2. A wide variety of 4GL rapid development tools are available [Terdoslavich, 1997; Hibbard, 1996; Jefferson, 1997].

Prototyping is either evolutionary or throwaway. In evolutionary prototyping, the prototype eventually evolves into the final system, whereas in throw-away prototyping, the objective is to validate or derive the system requirements [Sommerville, 1996]. However, in Web site development, the distinction between these two types of prototyping may not be clear-cut. Both the information content and the user interface, which are often developed using HTML, can always be reused and integrated into the final Web site. However, the program/database portion of the prototype can be either evolutionary or throwaway, depending on the size and complexity of the site. In the case of a small Web site with a relatively simple database function, a database prototype developed using a 4GL language may evolve into the final product. On the other hand, in the case of a major Web site development involving linkages to the corporate product database, the prototype is only used to validate the system requirements and will be thrown away. The final system is developed using tools that are either compatible with the product database or more efficient in execution.

Sommerville [1996] suggests a process model for prototyping (see Figure 6) that can be used to develop the prototype of a corporate Web site. In most

cases, the objectives of the prototype are the construction of the user interface and the validation of functional system requirements. As in conventional development, the prototype for a corporate Web site usually contains a subset of system functions. The prototype is developed using 4GL rapid Web development tools, and clients or users discover errors or the need for new requirements through facilitated prototype reviews, which are held using a JAD format. Some advantages of using formal prototype reviews include the control of project scope by focusing users and developers on distinct levels of requirements, the management of customer expectations via a step-wise refinement or evolutionary approach, and the discovery and handling of problems early in the development process [Fitzpatrick, 1999]. After the prototype has been evaluated, a revised set of requirements including a software specification, which serves as the basis for design and implementation [Sommerville, 1996], is developed. Furthermore, an additional walkthrough may be conducted to finalize the specification documents before the incremental implementation phase.

INCREMENTAL IMPLEMENTATION

Construct Site Using Increments

Upon clients/users' approval of the prototype, the necessary hardware and software for developing the final system needs to be acquired. The following steps of the CWSD methodology parallel an incremental development process model in which individual system components are built and delivered in stages [Sommerville, 1996]. As mentioned earlier, sections of a Web site are often loosely coupled; therefore, the construction of a Web site fits naturally into this type of model. A description of each of the steps follows.

First, the increments or builds of the system must be determined. The Web site structure hierarchy developed earlier provides insight on how to divide the entire Web site into more manageable pieces. The actual construction of each increment includes four major activities:

1. *Digitize Materials and Design Art Work* -- Project team members or specialists digitize the materials that are selected for the Web site, while graphic designers design and develop the specific layout and artistic elements of the Web site with aesthetic appeal in mind. Digitized materials include text, images, audio, and video.
2. *Code Program* -- Programmers write the code (e.g. HTML, XML, JavaScript, Java, or CGI) for the Web site.
3. *Unit Test and Integrate* -- Each system increment goes through a unit test before it is integrated into the Web site.
4. *Validate with Customer* – The resulting integration is validated with the customer.

Most of the time, the first two steps can occur in parallel. Similarly, the last two steps can also occur simultaneously, because unit testing involves two types of tests: tests to determine whether the system increment performs its specified function correctly and the all important usability tests. Usability tests address Web design issues by providing data about the problems that people encounter when interacting with the Web site and by attempting to diagnose the cause of these problems. If a Web site scores high in usability, the chances of the site being accepted and used productively are good. Usability tests depend on a number of measurable variables, such as learning time and end-user satisfaction [Stahl, 1987]. Nielsen [1998] found that the majority of all Web site interfaces are unusable. When user interfaces are difficult to learn and use, users are more likely to make errors resulting in low performance and low satisfaction. He suggests usability tests be performed to ensure that the Web site is user-oriented. For a comprehensive discussion of usability tests, the interested reader may refer to the study of Nielsen and Sano [Nielsen and Sano, 1995]. Finally, if the system increment passes the unit test and customer validation, it can be integrated with the existing Web site. This process iterates until all pieces of the corporate Web site have been completed and integrated into the site.

SYSTEM TESTING

Install and Test

Prior to completing the Web site, all pieces are temporarily stored in a development area. The lessons learned from the past show that nothing irritates users more than broken links, dysfunctional databases, and "under construction" signs in a Web site [Nielsen, 1998]. During this stage, the completed Web site is installed and tested on the designated hardware. To ensure that integration requirements for the Web site are fulfilled, a complete system test is once again conducted. A system test is a comprehensive test that generally includes tests on network connections, database access, and Web site navigation and functions. The system test may result in requiring modifications to the Web site. These modifications prompt the project team to return to the previous activity, and this iteration will continue until the system performs at an acceptable level as prescribed in the requirements and software specifications.

ACCEPTANCE TESTING

Deliver Web Site

The Web site is delivered into production following a conversion plan [Whitten et al., 1994]. Because a corporate Web site can be categorized as an E-system [Lehman, 1995], i.e. a system embedded in the real world, the system will likely undergo continual change. Therefore, clients should receive a copy of the maintenance request procedures upon delivery of the site.

MAINTENANCE

Maintain Web Site

The WDG is responsible for all maintenance performed on the Web site. More specifically, the group handles problems related to daily usage, manages system modifications, improves existing functions, and ensures that the system

performs at an acceptable level [Pfleeger, 1998]. The activities associated with maintenance can be classified as:

1. corrective, changes to correct remaining faults,
2. adaptive, changes necessary to respond to a changing system or environment,
3. perfective, changes to improve the site or documentation, or
4. preventive, changes to defend against system failure.

When new requirements arise because of requests for additional product functionality or changes to accommodate new business objectives, the project team needs to retrace the steps of the corporate Web site development methodology associated with the W software life cycle model. Furthermore, the documentation from previous test cases needs to be reexamined so that regression testing can be performed.

SUMMARY OF THE W MODEL

Each step of the phases of the W software life cycle model was described in this section. Although the steps of the planning phase were described as if they occur sequentially, often these steps take place simultaneously. Simultaneity is especially true when the development or maintenance activity for a particular Web site is an internal request, since the corporate WDG will normally have other ongoing projects. Therefore, all work requisitions as well as the tasks associated with an individual work request may need to be prioritized.

Finally, developers need to understand that web sites tend to evolve rapidly and require frequent updates and redesign [Gellersen and Gaedke, 1999]. When new user requirements require revamping the entire Web site, the project team will need to go through the whole development process again to redesign the site. Furthermore, since Web site development projects differ enormously in their size, functionality, and complexity, organizations should use the W software life

cycle model and associated CWSD methodology creatively to accommodate their own situations.

V. CONCLUSIONS

Due to escalating costs of corporate Web site development in recent years, there is an urgent need for a comprehensive methodology to effectively guide a company through its entire Web development process. The major contribution of this research is the design of a specialized software process model (W model) and an associated corporate Web site development (CWSD) methodology. The combined model/methodology is based on insights obtained from previous corporate Web site development projects; therefore, the combination is not only useful but also practical. It provides the basis for a corporate Web site development paradigm. This combined model/methodology has important implications for both practitioners and researchers.

For practitioners, the W model and associated CWSD methodology provide a means of avoiding ad hoc development, which is unfortunately the norm in many companies. The model illustrates the major phases of a Web site development project while the methodology delineates individual steps and associated deliverables to guide corporate web development groups. Benefits gained by following the steps of the methodology include systematic management of each development phase to ensure an efficient process and to improve the quality of the resulting Web site. By applying corporate Web site standards, the various components of a site will support a common corporate Web strategy, infrastructure and identity. Furthermore, usability tests performed on a regular basis will increase the chances of the Web site being used productively. Finally, due to the careful planning and documentation connected with the methodology, all types of maintenance including future expansion of the Web site will require less effort.

The market for Web site development tools is extremely competitive [Terdoslavich, 1997; Strehlo, 1996c]. While most vendors focus on incorporating a collection of reusable functions in their products in order to make the development process easier and faster, few provide a comprehensive corporate solution to Web site development. By integrating the corporate Web site development methodology into their products, tool developers could add enormous competitiveness to their products.

For researchers, the unique features of Web applications such as corporate Web site design present exciting challenges and research opportunities in the area of software engineering. Since the Internet will serve as the primary decision support delivery platform for intra- and extra-enterprise applications [Ladley, 1998], the demand for Web application methodologies can only increase. As a final comment, the W software life cycle model and its associated methodology will continue to be tested on corporate Web sites, and the methodology will be further refined if warranted.

EDITOR'S NOTE: This article was received on September 11, 2000. It was with the author for approximately 5 months for one revision. The article was published on April 13, 2001

ACKNOWLEDGEMENTS

A previous version of the corporate web site life cycle model and its associated methodology were presented at the *ICSE '99 Workshop on Web Engineering*, which was held in conjunction with the *21st International Conference on Software Engineering*. We would like to thank all the participants for their valuable feedback on the previous model and methodology. Our special thanks go to San Murugesan and Yogesh Despande, organizers of the workshop.

REFERENCES

EDITOR'S NOTE: The following reference list contains the address of World Wide Web pages. Readers who have the ability to access the Web directly from their word processor or are reading the paper on the Web, can gain direct access to these references. Readers are warned, however, that
1. these links existed as of the date of publication but are not guaranteed to be working thereafter.
2. the contents of Web pages may change over time. Where version information is provided in the References, different versions may not contain the information or the conclusions referenced.
3. the authors of the Web pages, not CAIS, are responsible for the accuracy of their content.
4. the authors of this article, not CAIS, are responsible for the accuracy of the URL and version information.

ActivMedia (2000) "Online Companies are Investing $22 Billion on Website Hosting & Development," ActivMedia. Available at www.activMedia.com.

Advertising Age's Business Marketing (1996) "Web Price Index," *Advertising Age's Business Marketing*, 81(7), p. M14.

Anderson, M.D. and J. Choobineh (1996) "Marketing on the Internet," *Information Strategy: The Executive's Journal*, 2(4), pp. 22-29.

Balasubramanian, V. and A. Bashian (1998) "Document Management and Web Technologies: Alice Marries the Mad Hatter," *Communications of the ACM*, 41(7), pp. 107-115.

Berard, E. V. (1993) "Creation of and Conversion to Object-Oriented Requirements", *Essays on Object-Oriented Software Engineering*, Englewood Cliffs, NJ: Prentice-Hall,.

Boehm, B.W. (1981) *Software Engineering Economics*, Englewood Cliffs, NJ:.Prentice Hall,

Boehm, B.W. (1988) "A Spiral Model of Software Development and Enhancement," *IEEE Computer,* 21(5), pp. 61-72.

Bruegge, B. and A. H. Dutoit (2000) *Object-Oriented Software Engineering: Conquering Complex and Changing Systems*, Upper Saddle River, NJ:.Prentice Hall,

Carey, J.M. (1990) "Prototyping: Alternative System Development Methodology," *Information & Software Technology,* 32(2), pp. 119-126.

Chau, C.T. (1997) “The Internet: a World of Opportunity for Accountants,” *National Public Accountant,* 42(2), pp. 20-24.

Davis, A.M. (1993) *Software Requirements: Objects, Functions, and States, Revised Edition*, Englewood Cliffs, NJ: Prentice-Hall,

DeVille, B. (1995) “Internet for Market Researchers,” *Marketing Research,* 17(3), pp. 36-38.

Dykman, C.A. and R. Robbins (1991) "Organizational Success through Effective System Analysis," *Journal of Systems Management,* 42(7), pp. 6-16.

Facilities Design & Management (1997) "Business Internet Usage More Than Doubles in 11 Months,” *Facilities Design & Management,* 16(4), p. 15.

Fassett, A.M. (1995) "Building a Corporate Web Site: Advice for the Hesitant," *Telecommunications,* 29(11), pp. 33-36.

Fitzpatrick, J. G. (1999) “RAD in Summary”, Industry Presentation at the *21st International Conference on Software Engineering*, Los Angeles, CA, May 1999.

Friedman, K. (1998) "Case Study of One Intranet Development Project," *Professional Safety,* 43(10), pp. 16,56.

Gantz, J. (1996) "Is Your Web Site Worth the Effort?" *Computerworld,* 30(5), p. 37.

Gellersen, H. and M. Gaedke (1999) “Object-oriented Web Application Development,” *IEEE Internet Computing*, pp. 60-68.

German Ministry of Defense (1992) *V-Model: Software lifecycle process model*, General Reprint No. 250 Bundesminister des Innern, Koordinierungs- und Beratungstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung.

Gillenson, M.L., D.L. Sherrell, and L. Chen (2000) "A Taxonomy of Web Site Traversal Patterns and Structures," Communications of AIS, (3) Article 17.

Goff, L, (1996) "Wowing 'em on the Web," *Computerworld*, 30(21), p. 102.

Hamblen, M. (1997) "Telecom Managers Want Web Site Voice," *Computerworld,* 31(47), pp. 41-45.

Healthcare Executive (1997) "Web Site Development: Action Steps," *Healthcare Executive*, 12(3), p. 15.

Hibbard, J. (1996) "Web Site Development Tool Takes It a Step Further," *Computerworld,* 30(50), p.10.

Isaak, J. (1999) "IEEE Standard 2001: Web Page Engineering for Intranets and Extranets", *IEEE Internet Computing*, 3(2), pp.94-95.

Isakowitz, T., M. Bieber, and F. Vitali (1998) "Web Information Systems," *Communications of the ACM*, 41(7), pp. 78-80.

Ivins, B. (1998) "Reconciling Audience Measurement Data to Server Logs," Relevant Knowledge, Available at http://www.relevantknowledge.com.

Jefferson, S. (1997) "Cut and Paste Your Way to a Powerful Prototype," *Datamation,* 10(43), pp. 104 - 105.

Kambil, A., and M. Ginsburg (1998) "Public Access Web Information Systems: Lessons from the Internet EDGAR Project," *Communications of the ACM*, 41(7), pp. 91-97.

Ladley, J. (1998) "Data Warehousing Trends," *Data Web Adviser*, pp. 35-36.

Lantz, K.E. (1985) *The Prototyping Methodology*, Englewood Cliffs, NJ: Prentice-Hall,

Lederer, A.L., D.A. Mirchandani and K. Sims (1998) "Using WISs to Enhance Competitiveness," *Communications of the ACM*, 41(7), pp. 94-95.

Lehman, M.M. (1995) "Process Improvement - the Way Forward, *Proc. CaiSE 95*, Lecture Notes in Computer Science, Springer Verlag, pp. 1-11.

Lewis, L. (1998) "Site Unseen," *Progressive Grocer,* 77(8), pp. 83-86.

Lewis, T. (2000) "The Web Fosters a Rust-Belt Renaissance," *IEEE Computer*, 33(7), p.112.

Lohse, G. L. and P. Spiller.(1998) "Electronic Shopping," *Communications of the ACM*, 41(7), pp. 81-87.

Millman, H. (1997) "7 Rules of Site Design," *Computerworld,* 31(22), p. 82.

Nielsen, J. (1998) "That Mess on Your Web Site," *Technology Review*, Available at http://www.techreview.com/articles/oct98/nielsen.htm.

Nielsen, J. and D. Sano, (1995) "SunWeb: User Interface Design for Sun Microsystem's Internal Web," *Computer Network & ISDN Systems,* 28(1), pp. 179-188.

Nielsen, J. (1998) "Seven Deadly Sins for Web Design," *Technology Review.* Available at http://www.techreview.com/articles/oct98/nielsen-sidebar.htm.

Pfleeger, S.L. (1998) *Software Engineering*: *Theory and Practice*, Upper saddle river, NJ: Prentice Hall,.

Press, L. (1999) "The Next Generation of Business Data Processing," *Communications of the ACM*, 42(8), pp. 13-16.

Pressman, R.S. (1997) *Software Engineering*: *A Practitioner's Approach*, , New York: McGraw-Hill.

Reinbach, A. (1996) "Outsourcing the Art of Web Site Development," *Bank System & Technology,* 33(6), p. 47.

Rettig, M. (1992) "Interface Design When You Don't Know How," *Communications of the ACM*, 35(1), pp. 29-34.

Rossi, G., D. Schwabe, and F. Lyardet (1999) "Improving Web Information Systems with Navigational Patterns," *Computer Networks*, 31(11-16), pp. 1667-1678.

Rowen, R. B. (1990) "Software Project Management under Incomplete and Ambiguous Specifications", *IEEE Transactions on Engineering Management*, 37(1), pp. 10-21.

Sallquist, C. (1996) "How to Craft a Web Site That Delivers," *Presentation,* 10(7), pp. 28-33.

Schach, S.R. (1997) *Software Engineering with JAVA*, Chicago: Irwin,

Sommerville, I. (1996) *Software Engineering*, 5th edition, Harlow, UK: Addison-Wesley.

Stahl, B. (1987) "Testing for Usability Can Head off Disaster," *Computerworld,* 21(49), pp. 83-92.

Stepanek, M. (2000) "Are You Web Smart?", *Businessweek Online*, September 18, 2000.

Strehlo, K. (1996a) "Remember: Web Site Development Means App Development," *InfoWorld,* 18(43), p. 62.

Strehlo, K. (1996b) "The Web Is Young, But Your Processes Need Not Crawl," *InfoWorld,* 18(44), p. 69.

Strehlo, K. (1996c) "Three Important Web Tools: Process, Process, Process," *InfoWorld,* 18(45), p. 78.

Takahashi, K. (1998) “Metalevel Links: More Power to Your Links,” *Communications of the ACM*, 41(7), pp. 103-105.

Tenenbaum, J.M. (1998) “WISs and Electronic Commerce,” *Communications of the ACM*, 41(7), pp. 89-90.

Terdoslavich, W. (1997) "Navigating Web Site Development," *Computer Reseller News,* 726, pp. 57, 64.

Thomas & Betts (1998) *Web Standards and Guidelines and Self Validating Site Design Methodology*, Thomas & Betts, p. 2.

Trepper, C. (2000) “CRM Will Provide a Strategic Focus,” *Information Week*, May 15, 2000, p. 64.

Verton, D. (2000) “Big-Name Web Sites Lack Basic Guidelines”, *Computerworld*, October 2, 2000.

Whitten, J.J., L.D. Bentley, V.M. Barlow (1994) *Systems Analysis & Design Methods*, 3rd edition, Richard D. Irwin, Inc.

Wiegers, K. (1999) “Software Process Improvement in Web Time”, *IEEE Software*, 16(4), pp. 78-86.

Yorgey, L.A. (2000) “Reaching the World by Web,” *Target Marketing*, 23(7), pp. 42-46.

ACRONYMS

CWSD	Corporate Web Site Development
IMC	Interactive Marketing Company
JAD	Joint Application Development
RAD	Rapid Application Development
SDLC	System Development Life Cycle
WDG	Web Development Group
WIS	Web Information System

ABOUT THE AUTHORS

Linda B. Sherrell is currently on sabbatical from teaching. She taught mathematics or computer science at the university level for over twenty years and was most recently employed as an Assistant Professor in the Department of Computer Science at the University of West Florida where the work reported here was performed. She received B.S. and M.S. degrees in Mathematics from Auburn University and a Ph.D. in Computer Science from Louisiana State University. Her current research interests include electronic commerce, requirements engineering, rapid prototyping, and formal methods. She has published articles in *Software -- Practice and Experience*, *The Computer Journal*, *The Journal of Information Technology Management*, and the *International Computer Software and Applications Conference*.

Lei-Da Chen is Assistant Professor of Management Information Systems in the College of Business Administration of Creighton University. His research interests are electronic commerce, data warehousing and mining, and end-user computing. He received his Bachelor of Business Administration degree in

Computer Information Systems and Master of Business Administration degree in MIS from the University of Central Arkansas and his Ph.D. degree in Management Information Systems from the University of Memphis. His works have been published in *Information & Management, Communications of AIS, Journal of Computer Information Systems, Journal of Management Systems, Information Systems Management, Electronic Markets, and Journal of Education for MIS.*

Communications of the Association for Information Systems

ISSN: 1529-3181