Issues of Applying Conventional Development Methodologies to World Wide Web Environment

Hsiangchu Lai     Tzyy-Ching Yang
Department of Information Management,
National Sun Yat-sen University, Taiwan, R. O. C.

ABSTRACT

Due to the surprising growth of population of world wide web, doing business on the web becomes in vogue. As world wide web environment is so different from the traditional environment, it has challenged the appropriateness of conventional development methodologies. In this paper, we first introduce two major conventional development methodologies and the characteristics of world wide web environment. None of a single development methodology can work well alone. Based on our experiences, we illustrate how to adapt the conventional development methodologies to world wide web environment. We also address some fundamental technical issues. Finally, we conclude with some future research directions.

Keywords: System development, World wide web (WWW), Electronic Commerce, System development life cycle (SDLC), Prototyping, Web store.

1. Introduction

Since the forbidding by NSF of doing business on the internet was removed in 1991, all societies are very interested in electronic commerce. Among the internet tools, world wide web is the most popular for developing electronic commerce. World wide web’s popularity comes from its multimedia environment, hyperlinked structure, friendly user interface, boundless extensibility and interactive ability, etc. Moreover, it has the potential to turn the prospective buyer from being a passive surfer to an interactive customer (Pitt, et. al., 1996).

Supposedly, web store is created in world wide web for the purpose to proceed business activities. It has many advantages. For example, it is open 24 hours daily. Consumers can do shopping at any time. It allows consumers to visit again and again at any places so long as they can access internet. The potential consumers may come from all over the world. It is much easier to conduct international marketing. Furthermore, it can integrate marketing activities such as promotions, advertising, on-line ordering, customer services and marketing researches to a synergistic effect. It can also implement all these activities in a multimedia environment. Because of the characteristics of web store, the numbers of web store have grown at a terrific pace.

From the viewpoint of developing systems, world wide web is different from the traditional environment in some aspects. First, let’s look through the developing side. World wide web is constructed by numerous hyperlinks. This open architecture is a new challenge for developers. Further, all users have identical interface no matter which homepage he/she accesses. It allows developers to have great freedom to add new functions. In addition, world wide web’s interface is independent to the programs parts. As a result, it is very easy to modify programs without redesigning interface. On the other hand, it lacks an integrated development environment. To update a web store usually needs to use HTML editors, CGI programs, Java and database software in combination. None the less, only in each tool’s own development environment can developer write, compile and test the programs.

Next, let’s examine the user side. Web users can be anyone from any places around the world. They can have different hardware devices, browsers and computer architecture. But due to the current limit of network bandwidth, the transmission speed is still a problem to be improved. Usually, users need great patience if they want to access a wonderful homepage in the context of fantastic multimedia.

Based on the discussion above, you would appreciate that world wide web environment is different from traditional development environment. Therefore, there will be some challenges for developers. The purpose of this paper is to raise the problems of
developing web stores and explore the possible solutions. The rest of this paper is organized as follows:

In Section 2, we review two major conventional development methodologies in traditional environment and point out major components of world wide web. In Section 3, we discuss the problems of applying conventional development methodologies to world wide web environment and propose some solutions according to our experiences. We also indicate some fundamental technical problems need to be solved in Section 4. We end this paper with conclusion in Section 5.

2. Background

2.1 Two Major Conventional Development Methodologies

With the advent of software-development tools, a system plan might recommend that a full function system be developed or that a prototype be built and tested ( Eliason, 1990). The former is SDLC methodology. It proceeds developing work with full understanding of all functions required by users. The latter is a prototype methodology involving an iterative testing, modifying and expanding process of a system model. That is, it starts to develop a system without full understanding of the users requirements. We will describe these two methodologies briefly and make a comparison between them.

(1) SDLC Methodology: Specification oriented

All types of information systems go through a series of phases from birth to death. This process is referred to as the system development life cycle (SDLC). Today, most systems people would agree with the concept of SDLC. But they don’t always agree on the nature or sequence of the life cycle steps (Capron, 1986). The seven stages of a SDLC model proposed by Kendall (1987) are problem recognition, feasibility study, analysis, design, construction, conversion and maintenance. SDLC methodology proceeds stage by stage to develop systems. Each stage will generate specifications which will become the basis of the next stage. This methodology emphasizes on having a full comprehension of each stage. So it discourages changes although some necessary revisions are still allowed. It is used for building systems where user requirements are highly structured and well-defined.

(2) Prototyping Methodology: Prototype oriented

Prototyping methodology is a good alternative to SDLC methodology when user requirements are imprecise and abstract at the beginning of the design process (Challa & Redmond, 1996). Normally, the system developed by this methodology is not structured. A prototype is a working, through abbreviated, version of a system. Prototype methodology is an iterative process through which to refine it again and again until the system matches user requirements (Turner, et. al., 1987). It can build, develop and operate an initial and highly simplified prototype system quickly based on the very preliminary user requirements. This methodology makes use of the prototype to facilitate the interaction and feedback between users and developers. The iterative process will not stop until the version is satisfied. Burch (1992) had compared SDLC and prototyping methodologies based on system characteristics (Table 1).

<table>
<thead>
<tr>
<th>Methodology</th>
<th>SDLC</th>
<th>Prototyping</th>
</tr>
</thead>
<tbody>
<tr>
<td>User requirements</td>
<td>User requirements are generally well defined</td>
<td>Difficulty in defining requirements</td>
</tr>
<tr>
<td>I/O and transactions</td>
<td>High in volume</td>
<td>Low in volume</td>
</tr>
<tr>
<td>Database</td>
<td>Large number of records and elements within records</td>
<td>Small number of records and elements within records</td>
</tr>
<tr>
<td>Controls</td>
<td>Extensive system of controls</td>
<td>Basic editing controls</td>
</tr>
<tr>
<td>Technology</td>
<td>Usually a widespread multi-user computer system, often interconnected by an enterprisewide telecommunication network</td>
<td>Usually a standalone computer with a ‘private’ database</td>
</tr>
<tr>
<td>Number of users</td>
<td>Many</td>
<td>Few</td>
</tr>
<tr>
<td>Type of application</td>
<td>Distributed throughout the organization for general functions</td>
<td>Parochial and ad hoc such as special reports, DSS or EIS</td>
</tr>
</tbody>
</table>
2.2 World Wide Web Environment

The original design purpose of world wide web is to support groups which are in different areas to cooperate to do research and build share knowledge-bases (Lin, et. al., 1995). It adopts client-server structure. HTML, URL, HTTP and CGI play the major roles in this structure. They construct great potential for developing application systems on world wide web. We will discuss them one by one briefly.

(1) HTML (Hypertext Markup Language Document): Homepages on world wide web are constituted by a simple "markup language" called HTML. The basis of HTML is SGML (Standard Generalized Markup Language) which is a much more general language used to define particular markup languages for particular purpose. As a result, HTML inherits the special characteristics of SGML structure, so it has a cross-reference function. Another advantage of HTML is that its functional programs, such as CGI, Java programs and interface design are independent from each other. Consequently, it is very easy to change interface design by modifying HTML only.

(2) URL (Uniform Resource Locator): URL is a draft standard for specifying an object on the internet. It adopts an unique method to describe the location of resources on the internet. By using the rule, new web server will be given a single area name. It allows a faster and efficient search even if the amount of servers increase at a tremendous speed.

(3) HTTP (Hypertext Transfer Protocol): World wide web uses a high-level communication layer called HTTP, which follows a client-server model. It can operate on different computer platforms and operating systems and therefore users can easily access all kinds of resources by a single interface. Independence and openness are the best characteristics of world wide web (Schubert, 1995). HTTP protocol is stateless; in other words, there is no relationship between two consecutive requests to the same server even if the requests come from the same user. This statelessness allows the HTTP server software to impose very little overhead on the server machine and keeps the protocol between the client and server very simple (Ibrahim & Franklin, 1995). However, there will be problems when we try to have two-way interactions or the current action depends on the previous step.

(4) CGI (Common Gateway Interface): CGI is a simple program executed in the web server. When a user generates an event, the related CGI programs will be fired. Then, the returns are either the result of executed programs or a virtual HTML text. Thanks to the speciality of CGI programs, world wide web can provide interactive functions and enlarge its application areas. For example, it can do database search and on-line ordering. This is an important factor that makes world wide web be a good system development environment.

Based on the above understanding of world wide web components, we now discuss how world wide web environment is different from conventional development environment as follows:

(1) Friendly user interface: World wide web integrates texts, graphics, videos, sounds and animation, etc. Such multimedia capability diversifies system environment. It also has an identical interface. All homepages have hyperlinked structure and therefore any user can navigate everywhere by clicking mouse only. These friendly characteristics probably are the main reason for the ever-growing internet population after the world wide web was developed.

(2) Hyperlinked structure: The hyperlinked structure of world wide web makes it very different from traditional systems. It not only allows users browse homepages with great freedom but also lets the developers to add new functions more easily and flexibly.

(3) Boundless extensibility: The hyperlinked structure of world wide web and the ever-growing web servers lead web server to boundless extensibility. For developers, it is very easy to link outside resources to a web store. For users, it allows them to navigate freely, although users may get lost easily in world wide web environment.

(4) Worldwide spread of users: Everyone
is a potential customer of any web stores. It becomes more difficult to locate the target market. It is also more difficult to design a system to meet all user requirements.

(5) Diverse user environment: Users can choose their favorable browsers, computer devices and set-ups to visit homepages. This diverse user environment results in requiring special considerations for developing applications on world wide web.

(6) Users-controlled system: All users have full control on how wide and deep to navigate when they visit web stores. Due to the hyperlinked structure, it can hardly have control of any consumer’s tour around the web stores.

(7) Nonintegrated development environment: Different tools are required jointly to develop web stores. For example, HTML editors are used to write homepages, CGI and Java programs are used to write functions enclosed in homepages. Each tool has its own development environment to allow developers write, compile and test programs. However, there are no integrated development environment which enable us to write, compile and test all kinds of tools in the same environment. Consequently, it takes developers a lot of time to develop and test the applications.

3. Development Methodology for World Wide Web environment

3.1 Problems of Conventional Development Methodologies

Based on our understanding of the characteristics of the two major conventional development methodologies and the characteristics of world wide web environment, we can see there are some problems if we want to apply conventional development methodologies to world wide web environment. In contrast to table 1, table 2 summarizes the characteristics of world wide web environment in terms of system characteristics. First of all, since world wide web is a very open environment, it can be accessed by any navigator around the world. All navigators who have accessed your homepages may become your customers. It becomes very difficult to define the requirements of such diverse users. As for I/O and transactions, most of current web stores focus on advertising only. They have not used CGI or Java programs extensively, so the amount of I/O might be low in volume now. Nevertheless, we anticipate the transaction volume will grow rapidly in the near future. In terms of database, because world wide web is a multimedia environment, it will have very diverse contents of texts, sounds, videos, images, etc. Besides, up to now, because intranet is still unpopular, current applications on the internet have no strong connection with traditional business database. Though, a closer connection can be expected when intranet is getting more popular. Due to the open environment, it is difficult to control a user’s behavior and security becomes an important issue. Moreover, world wide web is constituted by interconnected homepages around the world. The number and contents of homepages almost change every moment. The number of its users is numerous. The most applications are ad hoc and organizational applications.

<table>
<thead>
<tr>
<th>System characteristics</th>
<th>Characteristics of world wide web environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>User requirements</td>
<td>Difficult to define user requirements (based on speculations mainly)</td>
</tr>
<tr>
<td>I/O and transactions</td>
<td>Low volume now; may be higher in the future</td>
</tr>
<tr>
<td>Database</td>
<td>Diverse contents</td>
</tr>
<tr>
<td>Controls</td>
<td>Difficult to control a user’s behavior; importance of security control</td>
</tr>
<tr>
<td>Technology</td>
<td>Easily access homepages with multimedia content in hyperlinked structure connected by internet</td>
</tr>
<tr>
<td>Number of users</td>
<td>Huge</td>
</tr>
<tr>
<td>Type of application</td>
<td>Ad hoc and organizational applications</td>
</tr>
</tbody>
</table>

Table 2: Characteristics of world wide web environment

By comparing table 1 with table 2, we can have better understanding of the appropriateness of applying the two conventional development methodologies to world wide web environment. Let us examine the SDLC methodology first. In
view of the numerous internet users and the popularity of organizational applications, SDLC methodology is suitable. Moreover, it can also fulfill the required sophisticated security controls of world wide web. However, there are still some difficulties.

On table 1, it shows that SDLC methodology is suitable when user requirements are generally well-defined. Normally, when we adopt SDLC methodology, the finished system is not expected to be modified repeatedly once it has become operational. It is often used to develop organizational applications. However, for web application, its users may come from all over the world and therefore it is impossible to have well-defined user requirements. Most of the user requirements are based on speculation, i.e. they may change later. In addition, creative and timely services are critical success factors for web stores. If a web store doesn’t update its services often, it is impossible to keep users visiting the sites continually. In order to gain more strategic advantages, it also has to response quick. To face the rapid innovation of internet technologies, the current systems may have to be enlarged or modified very often and ad hoc applications might be added frequently. Based on these points, SDLC methodology is obviously not sufficient.

Compared with SDLC methodology, it seems that prototyping is a better methodology to develop web stores because of the above reasons which lead to the inappropriateness of SDLC methodology. Unfortunately, there are still some problems too:

For the important issue of security control to web store, the performance of prototyping methodology might be worse than SDLC methodology due to a lack of a complete plan. Besides, prototyping methodology usually has maintenance problems owing to inadequate documentation and the lack of a structured development approach. As the applications are enlarged or modified more often, the maintenance problem will become more serious if we adopt prototyping methodology.

Table 1 indicates that prototyping is often used for a standalone computer system with a private database and for few users. It often requests users to take part in defining their requirements and giving feedback of using the prototype. However, the users on world wide web may spread everywhere. Consequently, it is impossible to ask their full involvement and to modify the prototype based on their feedback. Even if we can survey a group of users or allow users to send comments to us, the sample usually isn’t large enough to represent population. In spite of the fact that prototyping methodology is appropriate for ad hoc and small applications, it may still cause many problems when the functions of a web store are getting more complex, such as to integrate CGI, Java and databases, etc.

From the above discussions, we recognize none of a single methodology is sufficient. In Section 3.2, we’ll discuss how the two conventional development methodologies be adapted to world wide web environment based on our experiences.

### 3.2 Adaptation of Conventional Development Methodologies to World Wide Web Environment

In this section, we will describe how we adapt the two conventional development methodologies to world wide web environment when we helped a university bookstore create a web store. Fu-Wen Bookstore is a university bookstore located in National Sun Yat-Sen University (NSYSU), Taiwan. Its target customers are all teachers and students in NSYSU. We chose it as a pilot web store for developing intelligent customer services on the world wide web because it can conveniently communicate, provide better services to all members in NSYSU and offer us an opportunity to do field experiment.

The homepage structure of Fu-Wen bookstore is depicted as figure 1. There are six main options. ‘Bookstore Introduction’ and ‘Hot News’ are unidirectional presentation in order to provide information about the bookstore, best-seller, special discount, etc. Both of them are pure static HTML documents. All the other four options written by CGI programs can support interactive activities. Figure 2 is the system framework of Fu-Wen web store. On the front end of server, there are two different types of functions. The first is static HTML documents including three types: basic information, periodical updated information and seasonal information. The second is interactive function which is constructed by CGI programs for the purpose...
of providing book search, communication and customized services.

Under our developing experiences, we classified the contents of a homepage into two major types: non-interactive HTML documents and interactive virtual documents resulting from CGI programs. Non-interactive HTML documents provide users static information only. Interactive virtual documents result from an execution of CGI programs which are fired by user’s requests. The resulting virtual document is sent through an interactive interface on homepage to users.

Occasional News
For Members
For New Members
For Guests

Bookstore
Introduction
Hot News
Book Search
On-line Ordering
Communication Room
Intelligent Book Commendation
Interested-based Book Commendation
Bonus Query
On-line Order Query
Personal Data Modification

Main Options

Figure 1 : Structure of Fu-Wen Homepage

To provide only non-interactive HTML documents is the most popular application for current web stores. It provides users all kinds of information regarding a better picture of the business. It can be any information about the company, products and services as well as advertising and promotion activities. Its goal is to provide users information of regular business and occasional activities. It needs to be updated frequently. Since these documents’ life is short and they are independent from other functions, it will not make much sense to have a complete plan in advance. For this point, we adopt prototyping methodology owing to the high flexibility of HTML. HTML gives developers very high freedom to present non-interactive documents and therefore we can modify it and add new documents frequently without any difficulty. We invited the bookstore’s staff and a class of MIS graduated students to evaluate the prototype when we had the first version. After we modified it and finally put it on the web, all modifications were based on the suggestions from bulletin board, our experiences, new ideas from literature and other web stores, occasional events, etc.

In order to provide intelligent customer services, CGI programs were required. We classified the interactive applications into two types: ad hoc applications and organizational applications. Ad hoc applications were developed for special occasions which didn’t relate to organizational database much. The bulletin board, on-line questionnaire, book discussion board, and good book request in figure 1 belong to this type. We have also designed some applications of on-line lottery for promotion. Since these applications are less involved with organizational database and they are for non-regular and non-permanent activities mostly, prototyping methodology seems suitable to these kinds of applications.

Organizational applications meant to be developed for regular business operations and most of them are related to organizational database. On Fu-Wen web store, all applications of on-line ordering, book search and individualized services belong to this type. Book search function is built on the database of books. On-line ordering and individualized services functions are mostly based on customer-related database. The results may update related databases in turn.
Without any experience, we adopted prototyping methodology to develop this type of application in the beginning. It worked well at first. As more and more new applications were added, problems discussed in Section 3.1 appeared gradually, such as the databases have redundancies; the modification of old databases/applications for additional new applications were required; compatible problems of different versions of CGI tools came out. All these troubles not only took the developer a lot of time but also will affect the future development of the web store.

As discussed above, an organizational application is developed for regular business operation. These applications are more stable than non-interactive HTML documents and ad hoc applications because business operations don’t change frequently. In addition, they are involved with databases and often related to other applications. So we believe, in terms of the whole system framework of a web store, that we have to adopt SDLC methodology to have a more complete overall considerations even though it is impossible to have a very precise design due to the ever-changing characteristics of world wide web environment. Since data is a critical resource to an interactive application, current database design might be a vital success factor if we want to integrate all conceivable applications in the future with less difficulty. That is, database design may be a good starting point to allow us to have an efficient and effective integration in the future.

Based on our experiences, we summarized our adaptation of the conventional development methodologies to world wide web environment in figure 3. We suggest adopting prototyping methodology to develop non-interactive HTML documents and ad hoc applications. Meanwhile, we recommend using SDLC methodology to develop organizational applications.

4. Fundamental Technical Issues

In addition to exploring the adaptability of development methodologies to world wide web environment, there are still some fundamental issues deserved considerable attention based on our experiences and observations. In this section, we will discuss these fundamental issues, proposed solutions and the problems to be solved.

Open environment

World wide web is an open environment. Users can visit any homepage and jump to other homepages at any time. In order to provide better services, many web stores link some outside homepages, which are supposed to be interesting to their target customers, to their own homepages. Still, a user may navigate to other homepages and never come back again. In order to reduce such risk, we use frames to reserve our homepages in one of the windows no matter users jump to any other homepages from our homepage. Then the users can jump back to our web store at any time.

Due to the HTTP, HTML, URL, etc. characteristics of world wide web environment, it is more difficult to control and anticipate users’ actions. In order to monitor users’ behavior, we design ‘login’ and ‘logoff’ function. When users enter the web store, system records their names and IP addresses. As CGI program will pass an IP parameter to server upon browsing action, we take IP address as a key to identify who initiates this browsing action. It is the way to provide individualized service. Besides, the system will update related databases. This method allows us to observe users’ behavior and provide customized service without asking users to provide their personal data again and
again.

**Expandable flexibility**

World wide web provides developers boundless extensibility because it needn’t install new software on user site when adding new applications. This characteristic is a very good advantage for system development. In order to provide a greater freedom of navigation to the users, it would better allow cross-reference among all homepages. However, developers may spend a lot of time to modify all old homepages whenever there is a new homepage added.

For example, we first adopted imagemap files to arrange systems options. Imagemap file is a feature of both NCSA and CERN httpd servers. It can turn a GIF into a clickable map by designating polygons, circles and rectangles within the GIF and specifying a URL for each link (Boutell, 1997), but it needs to redraw the picture, reset areas and redesign hyperlinks when options are changed. Our solution is same as that for open environment. We divided screen into several windows. One window is reserved for main menus. Whenever there are new applications, we have to modify that window only. The shortcoming of this method is that the reserved window may be too small to display all applications simultaneously due to the size limit of the computer screen.

**User-oriented system**

Every user of world wide web is a potential customer of a web store and user requirements are therefore mainly based on speculations. There might be a gap between the speculations and true user requirements. The ways we had tried to reduce this gap were prototyping methodology, creating bulletin board, recording users’ behavior and doing survey, etc.

Compared with the users of traditional systems, web users have full control over their navigating processes. As mentioned above, it is difficult to have precise user requirements. Recording user’s navigating process may become a wonderful solution to achieve a real user-oriented system. Use the tracing and analysis of each user’s navigating process as basis to develop customized applications. That is, we can apply AI technology, such as expert systems, machine learning, to provide intelligent customer service. Different user will get different responses and be presented with different information based on each user’s characteristics. For example, at Fu-Wen homepage, the application of interest-based book commendation will present new books to user according to each user’s reading interest recorded in their personal data. In other words, users with different reading interests will get different book commendations. On the other hand, the application of intelligent book commendation will present new books to user based on their recent history of navigating around the new books. So, even for the same user, the commendation changes every time. It is performed by applying learning mechanism.

**Different set-ups**

World wide web is a cross platform. Users can use any browser to visit any websites at any places; besides, world wide web can integrate texts, graphics, animation, videos and sounds, etc. Due to users’ diverse computer environment, our developed applications on world wide web have to be able to adapt to it. The way we have done was to test our systems on different browsers, resolution, computer devices and environment set-ups to make sure it works well no matter what environment.

Although multimedia capability permits us to design a dynamic, vivid applications, the speed of transferring it on the internet may be too slow to keep users waiting. A solution is to design different versions for a new application. For example, one has text only; the other has images, sound, etc. as well as text.

**Synchronous interactive ability**

CGI programs to let world wide web have interactive functions is an important speciality to make world wide web be a system environment. For example, the functions of bulletin board, on-line ordering, survey, book search, etc. provided by Fu-Wen homepage are all implemented by CGI programs. However, all these interactions are unidirectional and asynchronous. If we can develop some synchronous, multidirectionally interactive applications for web stores, maybe we can create an excellent shopping atmosphere by allowing many users to go shopping together and exchange their ideas among each other.
Hence, we need more sophisticated tools to develop synchronous interactive functions.

**Changing tools**

World wide web environment allows us to integrate HTML editors, image processing software, CGI programs, database software and Java applets, etc. to develop web stores. Since these tools change very fast, the different versions of programs and databases may not be compatible all the time. Moreover, it lacks an integrated environment for developers to build world wide web applications. System developers have to write, compile and test programs in each tool’s own development environment and, then, integrate them together to test whether there is any problems. If there is, they have to go back to the tool’s own development environment to modify, recompile and retest the programs. These back and forth processes make the development become very inefficient. So it is an important issue to have an integrated development environment in the near future.

5. Conclusions

World wide web is a global, interactive, dynamic, cross-platform, distributed, hyperlinked multimedia information system that runs over the internet. Because of the amount of users is growing extremely, doing business on the web becomes a hot issue for all businesses. Although there are many advantages in terms of doing business on the internet, there are some problems if we try to apply conventional development methodologies to create web stores owing to the characteristics of world wide web environment. Based on these characteristics, we not only point out the problems of applying conventional development methodologies to world wide web environment, but also provide some solutions based on our experiences. Moreover, we also indicate some fundamental technical issues deserved great efforts in the future.

In addition to the above mentioned direction for future research, there are some other worthwhile studying directions. First, how to increase the system’s flexibility and expandability and to reduce the impact on old system if there is a new application to be added? Second, how to integrate the traditional systems with web systems? Third, how to apply AI technology to provide more customized services? Finally, how to develop an integrated development environment which allows developers to write, compile, test, and integrate all kinds of development tools?

**References**


