

Chapter 1

A Framework for Multimedia Software Engineering

In December 1998 *USA Today* carried an interesting article, "Birth of a New Order", talking about the year that the world's lines of time and space collapsed. The most incisive paragraphs are excerpted below:

The global, time-crunched market driven by electronic information "forces things to get bigger and smaller at the same time," says Nicholas Negroponte, author and technologist at the Massachusetts Institute of Technology. "And that's so ironic, when things want to do both but not stay in the middle. There will be an increasing absence of things that aren't either very local or very global". Oil and cars aren't much suited to being small and local. So they're moving to become gigantic and cross-border. As for being small and local, that's where the Internet, or World Wide Web, comes in -- and it works in two ways. It lets little companies be global, so a start-up in a garage can put its goods or services on a Web site and sell world-wide, competing against midsize or big companies, wiping out disadvantages (such as distribution and scope) that once had to do with distance. And since little companies can change direction faster than bigger ones, they have an advantage in time. Big companies used to have time and distance on their side. Increasingly, little ones do. And so in 1998, we had the phenomenon of Amazon.com, which has become such a symbol of small beating big that business people have turned it into a verb: to be "amazoned". (December 31, 1998, USA Today)

It is interesting to study how Internet and multimedia technology might help the "little guys" compete against the "big guys". Indeed, this

investigation may lead to a better understanding of the roles of multimedia software engineering (MSE) in this new Internet-based industrial revolution.

1. HIGH PRESENCE AND HIGH TOUCH

Internet and multimedia are changing the rules of the economy and redefining our businesses and our lives. It is destroying solutions such as mass production, segmented pricing, and time and distance for big businesses. A company can develop a web page and advertising campaign and quickly compete in the world market. This has led to the flattening of the economy, whereby established companies and individuals doing business on their own can compete on an equal plane. The small companies that succeed in challenging the large companies are the ones who can maintain a global presence and yet make people feel that they are personal and easy to deal with.

(1) Small companies can interact closely with their customers, so that the customers feel that they are able to communicate to the small company what they need, as opposed to the customers merely accepting the mass-produced product that large companies will sell and not give much ground for derivation from the product.

(2) Web changed from just a means of advertising, to a medium to rapidly exchange ideas with potential customers. Since the small company listens to what they say, it not only results in having a satisfied (and probably a faithful) customer but increases sales significantly with time.

(3) The Internet's primary advantage in advertising is not so much in attracting attention and conveying a brief message (the tasks assigned to traditional advertising media), but lies instead in delivering in-depth, detailed information. Its real power is the ability to provide almost infinite layers of detail about a product or service, interactively, at the behest of the user.

However, small companies have to work smarter and respond more quickly [Murr98]. They have to avoid mistakes and make the best of possible use to everything. Corporations with big budgets can afford to lose their investments, while a small company looks at web as survival, not as an investment.

The small businesses also need to realize having a web site does not automatically mean that the company will reach millions of potential customers. It simply means that there is the potential to reach millions of potential customers. Company has to promote the site through advertisements, e-mail, links to other sites, and cutting-edge multimedia technology to attract lots of visitors. For a new start-up small company, a new idea is always crucial. Second, multimedia technology should be used to

provide various kinds of services on the web site. Third, once the site starts catching on and e-mails start rolling in, more and more person hours should be put into keep up with it all.

2. WHAT BUSINESSES WANT FROM MULTIMEDIA TECHNOLOGY

2.1 How Small Businesses View Technology

There are significant differences in how large and small businesses view technology [JBR95]: 1) Affordability - small businesses have to be extremely cost conscious, while big businesses have a larger capital to invest in technology. 2) Scalability - all small businesses have ambitions to become big and this is an important requirement in the technology that they buy. 3) Fast return on investment - while large companies can wait up to 12-18 months for returns, small businesses want instant gratification. 4) Simplicity - most small businesses want 'plug and play' products such as, for example, the Unix multiserver networks, or the peer-to-peer networks.

2.2 Advantages of Multimedia Technology

From the perspective of a small company, the advantages of multimedia technology are perceived as follows:

- (1) Helps develop advertising that could be used in many different media, thus cutting advertising costs.
- (2) Cuts down on the amount of time the development staff needs to deal with customer service issues.
- (3) Gives the appearance of having all the customer service support of a larger company.
- (4) Facilitates out-sourcing [Neil98].
- (5) Keep clients aware of progress in almost real time by allowing them access to the site in development.

2.3 Wanted: Flexible MSE Tools

What businesses want from multimedia, in the above context, become quite clear:

- (1) Affordability \Rightarrow software tools
- (2) Scalability \Rightarrow scalable software tools
- (3) Fast return on their investment \Rightarrow prototyping tools
- (4) Simplicity \Rightarrow easy-to-use tools

- (5) Helps develop advertising that could be used in many different media
⇒ adaptive multimedia tools
- (6) Cuts down on the amount of time the development staff needs to deal with customer service issues ⇒ customer-service-oriented tools
- (7) Gives the appearance of having all the customer service support of a larger company ⇒ scalable tools
- (8) Facilitates out-sourcing ⇒ specification tools
- (9) Keep clients aware of progress in almost real time by allowing them access to the site in development ⇒ incremental development tools

3. INTERNET AND MULTIMEDIA TECHNOLOGY TRENDS

To support the design of such flexible MSE tools, we note the following Internet and multimedia technology trends:

- (1) The Browser will become the preferred universal interface.
- (2) Java, already known as the de facto standard, will offer more attractive features in reducing the cost of Internet development to that of typical client/server projects.
- (3) Event-based modeling will provide software developers significant advantages over in-house development.
- (4) Sophisticated tools will monitor events as they change. Agents will be used to post events and make their own decisions about how to process events [Blak98]. Window dialogs will assist engineers with dynamics of objects.
- (5) Businesses will find they will not be equipped to keep-up with new technology. To compensate, they will defer to out-sourcing to obtain Internet and multimedia technology.
- (6) The proliferation of the Internet will give rise to data centers that will decentralize data and provide multiple companies access to data on a global basis [Neil98]. This in turn will promote the development of higher speed communication lines with remote management software systems. Internet 2 will replace existing multimedia standards with higher speed and enhanced video-conferencing.
- (7) Embedded wireless communication will find its way into the Internet. This technology will facilitate remote access to the Internet, rendering further proliferation of its use [Patr99].
- (8) Personal digital assistant (PDA) and/or palm top computer will become popular because of their cheap price and small size. People will use PDA to connect to Internet or do personal information processing. It can display video clip, play audio file or control household equipment. Mobile

agent software will be the important application for the PDA. Such agent software can do various activities such as downloading or finding interesting information from Internet, exchanging information among them, etc.

(9) There will be multimedia components in the software engineering process.

(10) Multimedia software in the future will be multi-lingual in order to gain widespread usage rather than specific in any particular language.

4. WEB SITE LIFE CYCLE

A web site in many ways resembles other types of corporate information systems. Each web site has a limited life span, similar to the water fall software life cycle model. One major difference is the emphasis on content development in multimedia applications, which can be iterative because the content must be continuously updated. The phases of web site development are as follows: idea formulation, general web site design, detailed design of web site, testing of an implementation and maintenance.

(1) Idea Formation: During the idea formulation phase, specific target-marketing program, content goals and objectives must be set. Since a web site development project can become very time consuming and a major capital investment to owners of small businesses, it may be more effective to identify opportunity of specialized markets big companies have ignored. Furthermore, the profile of netizens must be carefully studied [Choi99] to find out who is surfing the net and what these people are looking at. Small businesses should be aware of the dynamics of the on-line market place and develop strategies and plan accordingly. The ideas of this phase can pave the foundation for developing a comprehensive plan for web site design.

(2) Web Site Design: Web site should be integrated into the company's backbone information system so that the web site can grow along with the business. To be successful, companies must integrate e-commerce into their overall business strategies and processes. Moreover, content needs to be targeted to specific user's needs. Visitor's information should be collected so that the company will be able to tailor the web pages to the specific needs of the interested customers. Furthermore, it is important that the web site can be surfed fast and efficiently. In addition, the users should be involved by providing an opportunity for them to input suggestions and complaints. The development of navigational cues and the user interface is of critical importance. The actual design tasks can be out-sourced for a small company. Also a new web site should be linked to as many search engines as possible. This can increase the chance that the web site is visited. Financial infrastructure should be developed properly as well.

(3) Testing: Once the implementation is complete, the company should conduct a pilot to test its integrity and effectiveness. The pilot provides an opportunity to obtain feedback from functional groups, customers and business partners. It ensures the quality and usability of the site.

(4) Maintenance: It is essential that new content is developed and the web site is kept refreshed. Timeliness is the key on the web. Moreover, appointing a web master to manage the site on a day-to-day basis is imperative. Web master can trouble-shoot any error such as a link to a defunct web address, track the traffic of the web site, use reader feed back to build a loyal following and ensure server maintenance and security. Also, this person or persons should make sure that the company's web site supports the latest versions of popular browsers. Moreover, maintenance includes the continuous expansion of the functions and capabilities of a web-based system.

It can be seen the Web Site Life Cycle is essentially iterative. Therefore, the agile methodology, which emphasizes the iterative nature of software processes, may be more appropriate than the classical Waterfall methodology.

5. DUAL ROLES OF MULTIMEDIA SOFTWARE ENGINEERING

Having discussed what businesses want from multimedia technology and the web site life cycle, we can now discuss the roles of multimedia software engineering. We can view MSE in two different, yet complementary, roles: 1) to apply multimedia technology to the practice of software engineering; and 2) to apply software engineering principles to the design of multimedia systems.

Multimedia has two fundamental characteristics that can be expressed by the following formula: *Multimedia = Multiple Media + Hypermedia*.

How can software engineering take advantage of these two characteristics? Will these two characteristics pose problems in multimedia systems design?

In Chapter 2 we will give a focussed survey of current research in MSE to apply multimedia technology to the practice of software engineering, or to apply software engineering principles to the design of multimedia systems. From the focussed survey of Chapter 2 it will be seen that multimedia is useful in software engineering, but whole-hearted incorporation of multimedia in software engineering has not yet happened [Hira99]. There is an ongoing paradigm shift -- from business orientation to entertainment orientation [Hira99], which has evolved to the meta-universe orientation.

New software process models and paradigms, such as object-oriented approach, are needed in multimedia systems design. Other interesting approaches include model-based approach to define navigation and access primitives, virtual multimedia objects approach to construct complex multimedia objects using virtual links, and identification of patterns (of navigation, news, landmark, etc.) to facilitate multimedia design. A long-term goal of MSE should be to design multimedia systems by multimedia.

6. A CONCEPTUAL FRAMEWORK FOR MSE

A conceptual framework for MSE based upon the notion of *multidimensional language* (ML) will now be presented.

A multidimensional language is a language where the primitives are objects of different media types and the operators comprise of spatial and temporal operators. Because of the importance of such spatial/temporal operators, we prefer to call such languages multidimensional languages rather than multimedia languages, although the multidimensional languages can be used to specify multimedia applications.

This viewpoint enables us to describe the various aspects of multimedia applications with conceptual clarity. The corresponding framework for MSE thus provides a principled approach, a set of scalable, adaptive tools and an environment for the specification, design, testing and maintenance of multimedia applications.

6.1 Syntactic Aspect

A multimedia application is constructed from a collection of multimedia objects. The primitive objects are media objects of the same media type. The complex multimedia objects are composed from these primitive objects and in general are of mixed media types. The syntax of ML describes how the complex multimedia objects are constructed from the other multimedia objects. Spatial and temporal composition rules must be taken into consideration.

6.2 Semantic Aspect

Multimedia applications nowadays are seldom passive. A passive multimedia application can be specified by a static ML, but a dynamic multimedia application requires the system to take actions in response to user input or internal/external stimuli. The semantics of ML describes how the dynamic multimedia objects are derived from other multimedia objects

when certain internal/external events occur. Since an important characteristic of multimedia is the ability to create links and associations, the semantics of ML must take that into consideration.

6.3 Pragmatic Aspect

Multimedia applications are heavily content-based and require a lot of manual hard work to put together. Tools are needed to assist the designer in building a multimedia application in a timely fashion. The pragmatics of ML can be based upon the patterns for various multimedia structures or sub-structures, such as navigation structures, content-based retrieval structures, etc. Once such structures and sub-structures are identified, they can be used as building blocks in putting together a multimedia application.

6.4 Systems Aspect

Last but not least, the systems aspects of multimedia applications must be considered. Multimedia applications require the support of distributed multimedia systems. The systematic design, specification, analysis and optimization of distributed multimedia systems will improve the performance of multimedia applications. Both QoS (quality of service) and QoP (quality of presentation) must be considered in systems design.

6.5 The Multimedia Software Life Cycle

With the above described framework, the multimedia software life cycle can be seen to essentially consist of three phases, which can be iteratively applied: 1) **Syntactic Phase**: Gather user's requirements to specify the syntactic structure of the multimedia application. 2) **Semantic Phase**: Design the actions to be performed by the multimedia application. 3) **Pragmatic Phase**: Identify the basic building blocks and utilize tools to implement and test the application.

In any one of these three phases, the designer must always pay attention to the systems issues and to optimize the performance of the application in a distributed multimedia system environment.

If the three phases are followed in a sequential order with no repetition, we have something similar to the classical waterfall model. If the three phases are iterated fairly quickly in a software development environment with a set of integrated tools, we have something akin to the iteration model of the agile methodology. If the three phases are iterated using increasingly sophisticated tools with more and more emphasis on scaled-up operations, we have the spiral model. The spiral model can adopt the slow intelligence

methodology to form a complete model for environment aware software engineering.

To summarize, there are essentially three software process models:

The Waterfall model (the classical methodology)

The Iteration model (the new agile methodology)

The Spiral model (the slow intelligence methodology)

Sensor-based applications lead to Environment-Aware Software Engineering (EASE). The spiral model can be regarded as a generalization of the iteration model: The software life cycle is iterative, but can move upward or downward into another software life cycle due to changes in the environment. The recognition of environmental changes can be accomplished by the Slow Intelligence methodology.

7. PATTERNS

Software engineering can be regarded as the discipline for the systematic specification, design, substitution and verification of patterns. In multimedia software engineering the patterns are almost always visual, or can be made highly visual, due to the very nature of multimedia. Visual notation can also help in detecting and/or discovering patterns.

According to Christopher Alexander, a pattern is a morphological law that establishes a set of relationships in space. In the most general form, a pattern can be express as: $X \rightarrow r(A, B, \dots)$, which means: within a context of type X, the parts A, B, ... are related by the relationship r.

Later he expanded his definition as follows: Each pattern is a three-part rule, which expresses a relation between a certain context, a problem, and a solution. The 'problem' part can be understood as the semantics of the pattern. Then he developed a theory of beauty that transcends architecture, other human crafts, and structures in nature, based on **centers, structural features, and process**.

A center is something that draws our eye as a focus. It could be the top of a column, the center of a geometric design, or a well-proportioned bit of space at the side of a room. Centers have at least one axis of symmetry, and

are reinforced by the centers around them. Structural features enhance and strengthen centers to increase the level of wholeness in a system.

James Coplien applied Alexander's theory to the design of software patterns. In a software pattern, there are **interacting centers**. The interaction can be specified using **protocols**.

From the viewpoint of IC theory, the **center** can be specified by an **IC cell**. The interaction protocol is specified by the state transition and message exchange of the IC cells. Therefore from this viewpoint, a software pattern can be specified as a collection of interacting IC cells, or an IC system. What need to be added, is how to transform patterns defined by relations into an IC system.

8. ORGANIZATION OF THE BOOK

This book is organized according to the conceptual framework described in Section 6.

8.1 Overview

Chapters 1 and 2 give an overview of multimedia software engineering. These two chapters are introductory in nature and can be read with ordinary effort. Managers, software engineers, programmers and people interested in gaining an overall understanding of multimedia software engineering can read mainly these two chapters and secondarily the following two chapters, chapters 3 and 4, on the syntactic aspect of multimedia software engineering.

8.2 Syntax, Semantics and Pragmatics

The next six chapters, chapters 3 to 8, present the multimedia software engineering according to the proposed conceptual framework. Practitioners, system developers, multimedia application designers, programmers and people interested in prototyping multimedia applications can read these six chapters.

The syntactic aspect of multimedia software engineering is presented in chapters 3 and 4. Chapter 3 discusses the elements of a visual language, and Chapter 4 shows how the fundamental concept of a visual language can be extended to multimedia, so that multimedia interfaces can be designed. This chapter also gives a preliminary introduction to the concept of active index.

The semantic aspect of multimedia software engineering is the focus of the next two chapters. In Chapter 5 the active index is introduced. Chapter

6 then introduces the tele-action object, which combines the syntactic aspect (the multidimensional language for user interface), with the semantic aspect (the active index cells for performing actions).

Chapters 7 and 8 cover the pragmatic aspects of multimedia software engineering. The software tools useful in multimedia applications development are presented in Chapter 7. Chapter 8 describes the Multimedia Information Custom Engineering (MICE) environment in sufficient detail so that multimedia applications can be prototyped in the MICE environment. Such details can be skipped if the reader is not interested in using MICE to do prototyping work.

8.3 Research Issues

The next three chapters are more research oriented and intended for researchers working on the specification, modelling and analysis of distributed multimedia systems. Scientists, researchers and software engineers interested in the systems and theoretical aspect of multimedia software engineering can read these three chapters.

Chapter 9 discusses how multimedia languages can be designed systematically by adopting a principled approach. The design and analysis of distributed multimedia systems following a transformational approach is the subject matter of Chapter 10. Last but not least, Chapter 11 addresses the issues of formal specification of multimedia applications.

8.4 Exercises and Project Suggestions

The book can be used as a textbook in a course on software design or in a course on software engineering where the emphasis is on multimedia applications. To serve that purpose, exercises and project suggestions are included in Chapter 12.