

The Essence of Web Engineering

Managing the Diversity and Complexity of Web Application Development

Athula Ginige and San Murugesan
University of Western Sydney, Australia

Although everything has progressed rapidly in the Internet and Web arena, nothing has changed significantly in the way that most people develop Web sites and applications. Web-based systems and applications now deliver a complex array of content and functionality to many heterogeneous end users, and this trend will continue. Unfortunately, however, the practices that developers follow for Web application development today are as poor as they were when the Web was in its infancy.

Many organizations and developers have successfully developed large, high-performance Web sites and applications, but others have failed or face the potential for major failures. The primary causes of these failures are a lack of vision, short-sighted goals, a flawed design and development process, and poor management of development efforts—not technology. The way we address these concerns is critical to realizing the Web's full potential.

The new Web engineering discipline deals with the process of developing Web-based systems and applications. The essence of Web engineering is to successfully manage the diversity and complexity of Web application development, and hence, to avoid potential failures that can have serious implications. (For a brief introduction to Web engineering see our earlier article.¹)

This issue concludes our two-part theme on Web engineering. Part 1, which appeared in the January–March 2001 issue,² gives an overview on Web engineering and how it's helping practitioners develop and maintain high-quality Web systems and applications. It presents a Web design framework that facilitates application reuse, an object-ori-

ented approach to legacy integration, a tool for constructing Web documents with visual simulations, Web metrics, and a case study highlighting experiences in developing flexible Web services. This issue further explores Web-based systems development and practices and presents multidisciplinary perspectives that help shape this dynamic area of Internet and Web application development.

Web application development

In many cases, it's not possible to specify fully what a Web site should or will contain at the start of the development process, because its structure and functionality will evolve over time. Furthermore, the information contained within and presented by a Web site will also change. Thus, the ability to maintain information and scale the Web site's structure (and the functions it provides) are two major factors that should be considered when developing a Web site. These factors make Web application development different from traditional software development.

Need for a process

To better manage Web-based systems design and development, and to do it in a systematic and repeatable manner, we need a process that outlines the various phases, steps, and activities of Web-based systems development. This process should be planned well and clearly define a set of steps that developers can follow. Furthermore, the process should be iterative to cater to the evolutionary nature of Web applications. It should also be measurable and trackable. In addition, the process should facilitate a Web application's continual refinement and evolution based on feedback from users and clients.

A Web development process breaks Web development down into manageable chunks and offers techniques to help developers successfully manage and complete Web projects. Some characteristics of Web-based systems that make their development difficult—and uniquely challenging—include their real-time interaction, complexity, changeability, and the need to provide personalized information. In addition, the work and time required to design and develop a Web application is difficult to estimate with reasonable accuracy.

A sound process helps developers address the complexities of Web-based systems, minimizes the development risks, deals with the likelihood of change, and delivers the Web applications within time while providing feedback for management as the project continues.³

Nontechnical issues

Introducing a Web application in an organization causes a paradigm shift, because it has the potential to significantly change an organization's work practices and procedures. As a result, we need corporate commitment to manage the shift and drive the development and deployment of Web applications against the odds. Unfortunately, many developers aren't aware of, or don't adequately address, many of these nontechnical issues and considerations.

Paradigm shift. Deployment of a Web application or an e-business system in an enterprise can significantly impact the work and the way employees carry out various business functions or processes. To successfully manage this transition, companies must retrain the system's users to enhance their understanding of the Web and related technologies. Other aspects such as reengineering current operational processes, modifying organizational policies and training, and changing human-resources development policies to suit the new Web environment also contribute to successful deployment and use of Web-based systems.

Corporate commitment. We'd like to caution that getting corporate commitment to Web engineering proves challenging, especially when developers and management are under pressure to deliver Web sites and applications quickly. This means reducing the time spent on planning, designing, and testing during the development process. These problems are often compounded in startups as well as in established organizations that have little development expertise and processes in place.

Ten key steps for successful development

Based on our experience in building many Web-based systems and also on our research, we recommend the following 10 key steps to follow for successful Web development and deployment:

1. Understand the system's overall function and operational environment including the business objectives and requirements.
2. Clearly identify the stakeholders—that is, the system's main users, the organization that needs the system, and those who fund the system's development.
3. Specify the technical and nontechnical

requirements of the stakeholders and the overall system.

4. Develop an overall architecture of the Web-based system that meets the technical and nontechnical requirements.
5. Identify subprojects or subprocesses to implement the architecture. If the subprojects are too complex to manage, further divide them until they become a set of manageable tasks.
6. Develop and implement the subprojects.
7. Incorporate effective mechanisms to manage the Web system's evolution, change, and maintenance. As the system evolves, repeat the overall process or some parts of it as required.
8. Address the nontechnical issues such as revised business processes; organizational and management policies; human resources development; and legal, cultural, and social aspects.
9. Measure the system's performance.
10. Refine and update the system.

A proactive approach to building large, complex Web applications is to follow a sound development process and methodology. Such processes and methodologies have been applied in a number of successful Web applications (including the ABC Internet College, 2000 Sydney Olympics, 1998 Nagano Winter Olympics, and Vienna International Festival). We hope these and other such success stories will encourage Web developers to adopt suitable development processes and methodologies as well as motivate academics and researchers to focus their research on better Web development methodologies, techniques, and tools.

As we now place greater emphasis on the performance, correctness, and availability of Web-based systems, the development process will assume greater significance. Furthermore, as systems become larger, companies will need a team of people with different types and levels of skills to work together, necessitating distributed, collaborative development.

Scanning the issue

In this issue, we present five articles dealing with Web application development and mainte-

nance as well as the results of a survey on current development practices and perspectives on Web engineering.

To assist Web developers in modeling and implementing Web applications interfaces, Gómez, Cachero, and Pastor offer an object-oriented approach called the object-oriented hypermedia (OO-H) method. It captures all relevant information to create device-independent, front-end specifications.

To facilitate continuous Web site evolution and maintenance, Ricca and Tonella present ReWeb, a new Web development tool that analyzes a Web site's structure and evolution. This tool extracts a Web site's main structural features and its history of evolution. The structural information helps developers understand the Web site's organization and its Web pages and lets them obtain details of changes by analyzing the Web site's history.

Reinforcing the concerns we raised regarding ad-hoc development, Barry and Lang present a gloomy picture of how multimedia, including Web-based applications, are being developed in practice. Based on a survey they conducted in Ireland, they observe that no uniform approach exists to multimedia systems development and that there's little evidence to suggest that practitioners use any of the known development models. Barry and Lang argue for new, more usable techniques that capture requirements and integrate them with a system development framework.

Klapsing, Neumann, and Conen discuss how to apply the Resource Description Framework (RDF) to a formal model for defining a Web application. The resulting model is interoperable, and hence, a developer can use a suitable tool for each phase of Web application development.

There's now a growing need to make the text and multimedia information on the Web more usable and understandable by humans and machines. Nagao, Shirai, and Squire propose a scheme for annotating Web documents. Such annotations let computers create customized Web content—depending on user preferences—with less effort and greater quality. They claim their work is a step toward a better solution for dealing with information deluge.

The information technology community has readily responded to the needs and challenges arising from rapid changes and advances in IT by creating new fields of study. Deshpande and Hansen, who have been working with us to develop and promote Web engineering, trace its genesis

and revisit the issues we raised regarding Web application development. They present Web engineering as a discipline encompassing computer science, software engineering, information systems, multimedia, and many others that contribute to Web development. To cater to the Web and its developers' needs, they argue that Web engineering needs to be considered as a discipline in its own right rather than subsumed under software engineering or under any other currently existing discipline.

Looking ahead

Although the Web engineering discipline is young, it's gaining the attention of some researchers, developers, academics, and other major players in Web-based systems implementation such as customers and clients. Web engineering needs to evolve and mature to effectively handle the unique challenges that Web-based systems and applications pose. As a new research community, we need to develop new methods, techniques, and other approaches to address the challenges of developing large-scale Web-based systems. The areas that need further study include (not in any specific order)

- requirements analysis and systems design;
- information modeling;
- process and product models;
- testing, verification, and validation;
- performance measures;
- Web metrics;
- configuration and project management;
- user interface and ease of use;
- user-centric design, end-user development, and personalization;
- quality control and assurance; and
- education and training.

Although some of these areas are common to software development, there are certain special characteristics of Web application development that call for further study.

More importantly, we need to promote the Web engineering discipline and the benefits it offers among Web-based system developers.

Web engineering lies at the center of the Web revolution, which is one of the most important technological revolutions to affect our society. As more applications migrate to the Web environment and play increasingly significant roles in business, education, healthcare, government, and many day-to-day operations, the need for an engineering approach to Web application development will only increase.

We believe industry needs to move from a “nice to adopt” to a “must adopt” attitude and strategy for the development of large, high-performance, evolutionary, and/or mission-critical Web sites and applications.

We hope that in a few years Web engineering will become a well-established and mature discipline that’s widely understood and practiced. **MM**

References

1. A. Ginige and S. Murugesan, “Web Engineering: An Introduction,” *IEEE MultiMedia*, vol. 8, no.1, Jan.–Mar. 2001, pp 14-18, <http://computer.org/multimedia/mu2001/pdf/u1014.pdf>.
2. Special issue on Web Engineering, Part 1, *IEEE MultiMedia*, vol. 8, no.1, Jan.–Mar. 2001, pp. 14-65. See also <http://computer.org/multimedia/mu2001/u1toc.htm>.
3. A. Ginige, “Web Engineering: Methodologies for Developing Large and Maintainable,” *Proc. IEEE Int’l Conf. Networking India and the World (CNIW 98)*, 1998, <http://www.aeims.uws.edu.au/athula/publications/ga1998-1.pdf>.
4. S. Murugesan and Y. Deshpande, eds., *Web Engineering* (Lecture Notes in Computer Science—Hot Topics), vol. LNCS 2016, Springer-Verlag, Heidelberg, Germany, April 2001, <http://www.springer.de/comp/lncs/hottopic.html>.



Athula Ginige is Professor of information technology and head of the School of Computing and Information Technology at the University of Western Sydney, Australia. He is also the director of the Advanced Enterprise Information Management Systems Research Center at the university. His research interests are in the areas of multimedia systems; development of large-scale, Web-based information systems; information structures for interactive flexible learning modules; new information retrieval strategies for the Web; and e-business systems for small to medium enterprises. He graduated with a BS in engineering from the University of Moratuwa, Sri Lanka, and received his PhD in computer vision from the University of Cambridge. He is a fellow of the Cambridge Commonwealth Society and a member of the Editorial Board of *IEEE MultiMedia*.



San Murugesan is Associate Professor in the School of Computing and Information Technology at the University of Western Sydney, Australia. He is also the associate director of the Advanced Enterprise Information Management Systems Research Center at the university. His research interests include Web engineering, e-business technologies and applications, e-transformation, intelligent agents, enterprise information systems, and new information-retrieval schemes. He received his PhD in computer science from the Indian Institute of Science, Bangalore, India. He also holds BE and MTech degrees in electrical engineering from the PSG College of Technology, Coimbatore, India, and the Indian Institute of Technology, Chennai, respectively.

Readers may contact Ginige and Murugesan at the School of Computing and Information Technology, University of Western Sydney, Campbelltown NSW 2560, Australia, email {a.ginige, s.murugesan}@uws.edu.au, <http://www.aeims.uws.edu.au>.