

Chapter X

Visual Environment for DOM–Based Wrapping and Client–Side Linkage of Web Applications

Kimihito Ito

Hokkaido University Research Center for Zoonosis Control, Japan

Yuzura Tanaka

Meme Media Laboratory, Hokkaido University, Japan

ABSTRACT

Web applications, which are computer programs ported to the Web, allow end-users to use various remote services and tools through their Web browsers. There are an enormous number of Web applications on the Web, and they are becoming the basic infrastructure of everyday life. In spite of the remarkable development of Web-based infrastructure, it is still difficult for end-users to compose new integrated tools of both existing Web applications and legacy local applications, such as spreadsheets, chart tools, and database. In this chapter, the authors propose a new framework where end-users can wrap remote Web applications into visual components, called pads, and functionally combine them together through drag-and-drop operations. The authors use, as the basis, a meme media architecture IntelligentPad that was proposed by the second author. In the IntelligentPad architecture, each visual component, called a pad, has slots as data I/O ports. By pasting a pad onto another pad, users can integrate their functionalities. The framework presented in this chapter allows users to visually create a wrapper pad for any Web application by defining HTML nodes within the Web application to work as slots. Examples of such a node include input-forms and text strings on Web pages. Users can directly manipulate both wrapped Web applications and wrapped local legacy tools on their desktop screen to define application linkages among them. Since no programming expertise is required to wrap Web applications or to functionally combine them together, end-users can build new integrated tools of both wrapped Web applications and local legacy applications.

INTRODUCTION

During the last several years, the main portion of information resources on the World Wide Web (Berners-Lee, Cailliau, Luotonen, Henrik Nielsen, & Secret, 1994) has shifted from handmade HTML pages to server-generated HTML pages, such as those using Common Gateway Interface (CGI), Active Server Page (ASP), Java Servlets, Java Server Pages (JSP), and PHP: Hypertext Preprocessor (PHP). A Web application is an application program that has an HTML-based front end for users to utilize some services provided by a remote HTTP server. There are an enormous number of Web applications on the Web, and they are becoming the basic infrastructure of everyday life. Many companies and researchers provide Web applications, such as search engines, financial services, scientific analysis tools, and various other kinds of database services. Basically, Web applications allow every end-user to use various remote services and tools through their Web browsers.

We have now more than 10^9 pages on the Web, and most of those pages are crawled by major search engines. Although we have seen remarkable developments in information retrieval technologies, such as ranking and clustering of Web pages, it still takes a lot of time to get satisfactory information by trying to retrieve documents from the Web. The more pages we have on the Web, the more time it will take for users to get satisfactory results from the Web.

Our objective is to support users' intellectual activities using the Web. We assume target users to be specialists in a domain, such as finance or bioinformatics, but novices in computer programming. We call them end-users.

The end-users access Web applications through Web browsers. Thus, to combine two Web applications, users need to open these two Web pages, to input some data on the first page, to copy a part of the result, and to paste this copy into the

input-form on the second page. Users need to repeat this process if they want to apply the same processing to other inputs. In UNIX, users can compose command sequences using pipes and redirections. UNIX users can perform a job that is not implemented as a single command. On the other hand, Internet users cannot combine existing Web applications into a single application.

From this point of view, we propose a new framework for the visual wrapping and linking of Web applications in order to dynamically and visually compose an integrated function. The framework allows users to rapidly wrap any Web application with a wrapper component through direct manipulation. The key idea of the framework is the use of the IntelligentPad architecture (Tanaka, 2003; Tanaka & Imataki, 1989), which allows users to combine these wrapped visual components, called pads, by pasting pads on another pad, to compose a composite pad. This composition defines application linkages among Web applications to integrate them into a single composite function. Users can also combine wrapped Web applications together with wrapped legacy local applications, such as spreadsheets, charts, and databases to integrate their functionalities.

The remarkable development of Web engineering makes the revision and approval cycles of Web applications shorter and shorter. Because the providers of Web applications need to increase the quality and competitiveness of their applications, they often revise the format of front-end HTML pages, and sometimes they change URLs.

Many researchers have worked on developing a robust wrapping method for Web applications (Kushmerick, 2000; Phelps & Wilensky, 2000). Some wrapping methods have robustness for format changes. From the viewpoint of computational learning theory, it is impossible to deal with every kind of format change. We do not focus on such robustness in this chapter. We focus on how, instantaneously, users can create

20 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/visual-environment-dom-based-wrapping/24101

Related Content

The Political Use and Abuse of Science

Gabriel R. Ricci (2018). *The Changing Scope of Technoethics in Contemporary Society* (pp. 40-59).
www.irma-international.org/chapter/the-political-use-and-abuse-of-science/202490

Open Content in Libraries: Contractual Issues

Galateia Kapellakou, Marina Markellou and Evangelia Vagenas (2013). *Digital Rights Management: Concepts, Methodologies, Tools, and Applications* (pp. 1010-1029).
www.irma-international.org/chapter/open-content-libraries/71017

Fairness and Regulation of Violence in Technological Design

Cameron Shelley (2011). *International Journal of Technoethics* (pp. 20-36).
www.irma-international.org/article/fairness-regulation-violence-technological-design/62307

Designing a Moral Dilemma Support System

Kieran Mathieson (2005). *Information Ethics: Privacy and Intellectual Property* (pp. 38-52).
www.irma-international.org/chapter/designing-moral-dilemma-support-system/22938

Public Policy Issues and Technoethics in Marketing Research in the Digital Age

Pratap Chandra Mandal (2021). *International Journal of Technoethics* (pp. 75-86).
www.irma-international.org/article/public-policy-issues-and-technoethics-in-marketing-research-in-the-digital-age/269436