New Approaches to Portletization of Web Applications

Fernando Bellas
University of A Coruña, Galicia

Iñaki Paz
University of the Basque, Spain

Alberto Pan
University of A Coruña, Galicia

Óscar Díaz
University of the Basque, Spain

ABSTRACT

Portlets are interactive Web mini-applications that can be plugged into a portal. This chapter focuses on “portletizing” existing Web applications, that is, wrapping them as portlets, without requiring any modification. After providing some background on portlet technology, we discuss two kinds of approaches to portletization: automatic and annotation-based. Automatic approaches make use of heuristics to automatically choose the fragments of the Web application pages to be displayed into the space available in the portlet’s window. In turn, in annotation-based approaches, it is the portal administrator who annotates each page of the portletized Web application to specify which fragments should be displayed. Annotation-based approaches also allow to supplement the functionality of the original Web application. Each approach is explained by using a sample scenario based on the same Web application. We also pinpoint the advantages and shortcomings of each approach, and outline future trends in portletization.

INTRODUCTION

Moving an application to the Web implies facing its multichannel character. The very same functionality can be offered through a variety of “channels”, for example, as a Web application to be accessed from a desktop or a small-screen device (e.g., PDA, smart phone, etc.), or as a
portlet available through a portal. Although the functionality remains the same, the characteristics of each channel determine the implementation. The pressure to have a Web presence and the pace at which technology evolves make most organizations support the desktop-oriented Web application first, and at latter stages, care about adapting it for small-screen devices and portals.

Portlets are interactive Web mini-applications, local or remote to the portal, that render markup fragments that the portal can integrate into a page (usually shared by other portlets). Integrating a Web application as a portlet improves the user experience, since the portal can automatically apply user profiles when accessing the Web application. For example, if the application requires authentication, the portal can store authentication parameters to automatically authenticate the user in the application when she or he logs in the portal. Support for smart bookmarks is another example. The portal can allow the user to store bookmarks to specific screens of the application. Each bookmark stores the navigation sequence to reach such a screen (Anupam, Freire, Kumar, & Lieuwen, 2000).

In principle, to integrate a Web application, a specific portlet must be developed. This implies the need to develop a new presentation layer, which is always a time-consuming task. This chapter focuses on “portletization” existing Web applications, that is, wrapping them as portlets. The wrapping approach allows to expose Web applications as portlets to the portal without requiring any modification to the applications, thus greatly reducing the development time.

The rest of the chapter is organized as follows: First, we provide background on portlet technology and portletization approaches. Then, we exemplify the two main approaches proposed to date by summarizing the portletization techniques which have been the subject of the authors’ work. Next, we compare both approaches. Finally, we outline future trends in portletization, as well as present conclusions.

**BACKGROUND**

**Overview of Portlet Technology**

The classic approach to integrate a Web application into a remote portal consists in defining a Web service (SOAP/REST) that exposes part of the application’s business logic. The problem with this approach is that any portal wishing to integrate such an application must re-implement the user interface. It would be easier if the service was a remote portlet, returning HTML markup fragments rather than plain data. The Web Services for Remote Portlets (WSRP) specification (OASIS Consortium, 2003) standardizes the interfaces of the Web services that a portlet producer (typically, a portal) must implement to allow another application (typically, another portal) to consume its portlets, regardless of the technology which the producer and consumer use (e.g., J2EE, .NET, etc.). Portlet URLs embedded into fragments point to the portal. Whenever the user clicks on a link, the portal receives the HTTP request, invokes an operation on the portlet’s producer that returns the markup fragment corresponding to this interaction, and finally composes a page that includes the response of this portlet and those of the remaining portlets in the page.

The Java Portlet Specification (Java Community Process, 2003) standardizes a Java API (JSR 168) for implementing local, WSRP-compatible portlets. Java portlets run in a portlet container, a portal component that provides portlets with a runtime environment. Apart from the portlet container, the typical architecture of a Java portal server (Bellas, 2004) includes other components:

- The WSRP producer component provides an implementation of the WSRP interfaces, so other consumers can access local portlets; and
- The WSRP consumer component is implemented as a local portlet that acts as a generic proxy for any WSRP producer.
Related Content

Context-Aware Service Provisioning in Next-Generation Networks: An Agent Approach
[www.igi-global.com/article/context-aware-service-provisioning-next/2636?camid=4v1a](www.igi-global.com/article/context-aware-service-provisioning-next/2636?camid=4v1a)

Enterprise Architecture’s Identity Crisis: New Approaches to Complexity for a Maturing Discipline
[www.igi-global.com/chapter/enterprise-architectures-identity-crisis/137419?camid=4v1a](www.igi-global.com/chapter/enterprise-architectures-identity-crisis/137419?camid=4v1a)

Exploring the Influence of Affiliation Motivation in the Effectiveness of Web-Based Courses
Maurício Gregianin Testa and Edimara Mezzomo Luciano (2013). *Web-Based and Blended Educational Tools and Innovations* (pp. 57-77).
[www.igi-global.com/chapter/exploring-influence-affiliation-motivation-effectiveness/68637?camid=4v1a](www.igi-global.com/chapter/exploring-influence-affiliation-motivation-effectiveness/68637?camid=4v1a)

Database High Availability: An Extended Survey
[www.igi-global.com/chapter/database-high-availability/5214?camid=4v1a](www.igi-global.com/chapter/database-high-availability/5214?camid=4v1a)