Categorization of Rich Internet Applications based on Similitude Criteria

Jaime Muñoz Arteaga[2], Francisco J. Martínez-Ruiz[1,3], Jean Vanderdonckt[3], Alberto Ochoa[1]

[1]Universidad Autónoma de Zacatecas. Departamento de Ingeniería en Computación, Av. López Velarde No. 801, 98600. Zacatecas, México

[2]Universidad Autónoma de Aguascalientes. Aguascalientes, México

[3]Université catholique de Louvain, Louvain-la-Neuve, Belgium

{jamaru,vanderdoncktj}@acm.org, jma070@gmail.com,megamax8@hotmail.com
{martinez, vanderdonckt}@isys.ucl.ac.be

Abstract. *The Rich Internet Applications (RIAs) have been created over a set of well Known Web technologies. The increasing use of Web Rich clients as RIAs requires the definition of development methodologies and categorization criteria to define standards for the developers and the industry. In this paper, the second problem is addressed with the proposal of a classification scheme using a Similitude function to determine a RIA class.*

1. Introduction

The interest in building a Web Rich Client has been increasing since a couple of years. This User Interface type has similar features of those provided in typical desktop applications, e.g., robustness, better responsiveness and visually more appealing than the classic HTML ones. Rich Internet Applications (RIAs) technologies help us to reach this goal [O'Rourke 2004]. RIAs are Web applications that transfer most of the load of processing the user interface to the Web client while the predominant part of data (from control and maintaining to business data) remains on the application server. A standard RIA architecture (Fig. 1) includes an application controller and an application server that control the Web Services Calls that use a XML dialect to transfer data and layout information. Note: Recent Databases can handle also XML with this; the process of translation is pursuit by XQuery language [Berglund 2006].

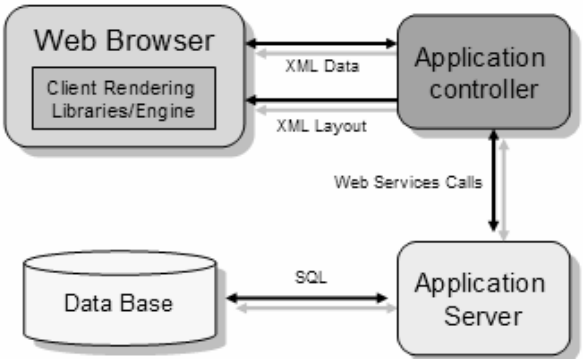

Figure 1: Typical architecture of a RIA application.

2. Problem description

It's difficult to classify RIA applications because they are a compendium of already known technologies in the Web development (JavaScript, CSS, XML, Java, among others) [Diaz *et al.* 1997] besides new features that implies a twist in the traditional way data moves from the client to the server and vice versa, e.g., XMLHttpRequest [Kesteren 2006]. Furthermore, the increasing complexity of Web applications could lead us to assume that classification of RIA applications is a simple task over the Data/Complexity continuum of Web Applications (Figure 2). The use of a RIA technology doesn't imply that the resulting application is going to be an exuberant example of multimedia capabilities, for instance a simple address book application which could be done using the Open Laszlo framework [OpenLaszlo 2006], one of the most popular and vigorous threads of RIA technologies at the moment and without the knowledge of the technologies behind this Web widget we can easily presume a classical Web application.

The Construction of a Rich Internet Application is more difficult than creating a traditional Web application but the payoff is a product with facilities near to desktop application ones like: drag-n-drop, changing shape cursors, and embedded plug-ins to manage video and audio streaming. In [Preciado 2005] the desirable features of RIAs are used to analyze the most suitable Web Development Methodology to build RIA applications but there isn't a proposal to categorize RIAs.

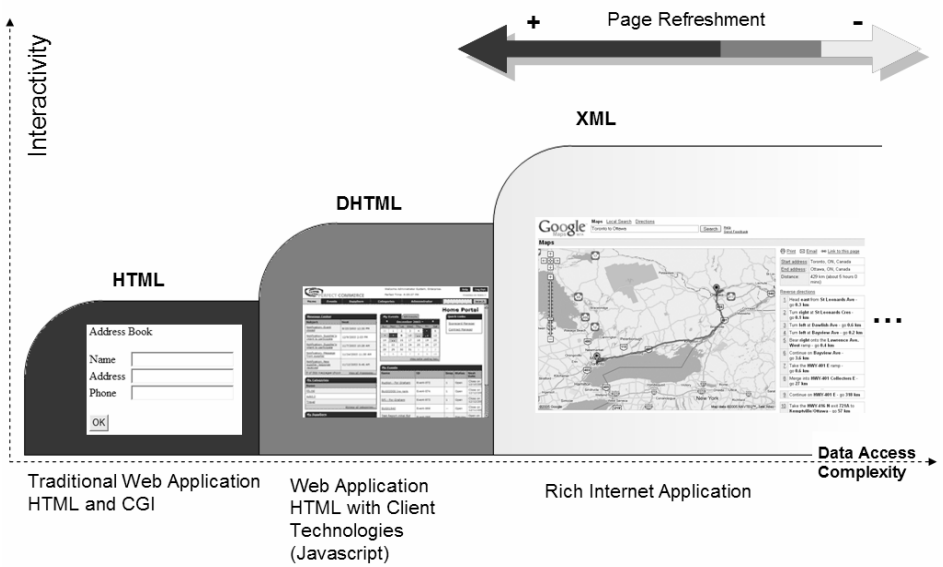

Figure 2: The Data/Complexity continuum in Web Applications

4. Contribution

This work proposes a categorization of RIA applications according to their extension and application domain (see figure 3):

- Complementary applications (Level I). These apps are mini gadgets that could cohabit with others in a more complex Web solution. Typically, their work is to support very specific tasks, e.g., retrieve weather conditions and specific domain calculators (i.e., Currency Rates and quotes of the day, among others).

- Utilitarian Applications (Level II). The main characteristic of these apps is the temporal activation period. Typically a user can use one of these for quick consult of some information (e.g., a Web Search Engine).
- Dominant Apps (Level III). The touchstone of RIA applications. These is an application which activation period is very long, interact with other applications independently from user to update its data from sources in the Web [Crane 2005]. Possible examples of what would be expected of these applications are shown at http://www.openlaszlo.org/demos# Dashboard where a virtual desktop [Payet 2002] is presented or the word processor online at: http://www.ajaxlaunch .com/ajaxwrite.

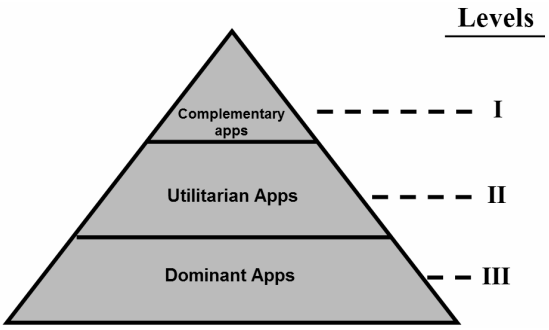

Figure 3: a Rich Internet Applications Categorization.

The classical model of categorization assumes that categories should be mutually exclusive and exhaustive. The result is a classification scheme that unequivocally, can assign new individuals to defined categories. Another way is the implementation of statistical techniques to define a categorization scheme in order to test new RIAs to determine their category according to their features [Chen 2000].

In this paper we propose a categorization method based on similitude techniques. In particular, one used in Case-Based Reasoning (CBR) [Kolodner 1993]:

$$\frac{\sum_{i=1}^{n} w_i \times sim\left(f_i^I, f_i^R\right)}{\sum_{i=1}^{n} w_i} \qquad (1)$$

Where $\boldsymbol{w}_i$ is the importance of dimension $\boldsymbol{i}$, $\boldsymbol{sim}$ is the similarity function and $\boldsymbol{f}^I$ y $\boldsymbol{f}^R$ are the values in the input and retrieved cases. The similarity function for this first attempt of categorization is a Euclidean distance:

$$sim(f^{IV}, f^{MV}) = \sqrt{(InputValue - ModelValue)^2} \qquad (2)$$

Where features $\boldsymbol{f}^{IV}$ and $\boldsymbol{f}^{MV}$ are evaluated to determine their matching ranking. Indeed, the selection of the classical Euclidean distance was made in order to get a comparative baseline to be used in future research and help us to test other distance functions.

4.1 Dimensions of RIAs

The dimensions are the features that we consider relevant to characterize a RIA application. The selection of these features is based on the works of [Preciado 2005] [Crane 2005] and [Bozzon 2006]. It includes the following characteristics: First,

Dynamic data retrieval that is data transmission from client to server and vice versa at execution time. Second, Perceptive continuity is the reduction of page refreshments and freeze situations. Third, Adaptability is the capability of react in an autonomous way to the user necessities. Forth, Multimedia is the capability of handle embedded graphics, video, audio and streaming. Fifth, Collaborative faculties are the capability of cooperation among different users to deal with a common problem or task. Sixth, a User Interface Language (UIL) that describes the UI for multiple contexts e.g., Character User Interfaces, Graphical and Multimodal ones in a form that maintains design independent from specific platforms [Bouillon 2005]. Seventh, Push Technology is the capability of manage unsolicited data to update the information presented to the user [Franklin 1998]. Eighth, The Use of Browser area (Typical Web applications remains attached to the classical model where the navigation bar is an integral part of the application. In contrast, RIAs should use in some cases the whole window and hide the navigation bar).

In order to create a categorization we need to define the range that every feature should cover (see table 1) and the proposed weight given to every feature. The most important characteristics have received a weight near 1 and characteristics not relevant have scored almost 0.

Table 1: Features and Weights needed to categorize a RIA.

Features	Dynamical retrieval		Perceptive continuity			Adaptability			Multimedia			
Feature Attribute	no	yes	none	partial	Full	None	partial	Full	none	animation	sound	embedded streaming video/sound
Values	0	100	0	50	100	0	50	100	0	30	30	40
Dim. Weight	1		0.8			0.8			0.6			

Features	Collaborative faculties			User Interface language		Push Technology		use of Browser area (main or popup one)		
Feature Attribute	none	partial	full	no	Yes	no	yes	minimal	partial	Full
Values	0	50	100	0	100	0	100	30	60	100
Dim. Weight	0.2			0.6		0.2		0.2		

According to equation (1) we need basic cases for each category to test the new prospects and determine which category is more suitable for them. In table 2, three models and their associated features are presented.

Table 2: Features and Weights needed to categorize a RIA.

Level	Dynamical retrieval	Perceptive continuity	Adaptability	Multimedia	Collaborative faculties	User Interface language	Push Technology	Use of Browser window
I	100	0	0	30	0	0	0	0
II	100	50	50	60	50	100	0	60
III	100	100	100	100	100	100	100	100

5. Study cases

5.1 Online word processor: ajaxWrite

The ajaxWrite application is an online word processor that can be accessed at http://www.ajaxwrite.com. Its features include open, read and write in some of most popular document formats. The values that we obtain for the features of this application: Dynamical retrieval = 100, Perceptive continuity = 100, Adaptability = 0, Multimedia = 30, Collaborative faculties = 0, User Interface language = 100 (XUL [Mozilla 2006]), Push Technology = 0 and use of Browser window = 100. The match rankings that we calculate using the prototype models are 65, 73.63 and 63.18, respectively. Thus, the ajaxWrite application's category is level II.

5.2 weather Application

This Weather application is a small gadget that doesn't use the whole navigator window. The application has four screens: one for introduce the zip code of a U.S. City, and three more for describing the weather conditions: wind speed, humidity, radar view, among others that you can test at http://www.laszlosystems.com/partners/support/demos/_weather/. The values that we obtain for the features of this application: Dynamical retrieval = 100, Perceptive continuity = 50, Adaptability = 0, Multimedia = 30, Collaborative faculties = 0, User Interface language = 100 (open Laszlo [Openlaszlo 2006]), Push Technology = 0 and use of Browser window = 30. The match rankings that we calculate using the prototype models are 77.27, 83.18 and 50.91, respectively. Thus, the Weather application's category is level II.

5.3 Google suggest

This application is a modification of the Google Search Engine which gives you alterative spellings in real time of the typed words in order to suggest possible sites. Its URL is http://www.google.com/webhp?complete=1&hl=en. The values that we obtain for the features of this application: Dynamical retrieval = 100, Perceptive continuity = 100, Adaptability = 50, Multimedia = 0, Collaborative faculties = 0, User Interface language = 0, Push Technology = 0 and use of Browser window = 30. The match rankings that we calculate using the prototype models are 68.64, 65.45 and 51.36, respectively. Thus, Google Suggest application's category is level I.

6. Conclusions and Directions for Future Research

In this paper we have proposed a Categorization method based on similitude of RIAs features and the analysis of the elements and technologic architectures behind RIAs. The result is a classification tool to index the modern Web applications. The task of classification is manual but an automated version is in process. This paper is part of a coordinated work to look for an extensive review of RIA technologies using UsiXML (http://www.usixml.org – User Interface eXtensible Markup Language) [Martinez-Ruiz *et al.* 2006] and more classification schemas. Nevertheless, more work is needed to build a more general classification. The inclusion of Ergonomic and IHC criteria would be important in this work in progress. For instance, in the selection of dimensions weights that needs a more quantitative method of definition. Finally, the study cases

besides more samples have shown that current apps are not level III ones, the most suitable candidates (section 4) are still betas and lack some of the features needed to index as dominant applications but in the future this status could change.

References

Berglund, A., Boag, S., Chamberlin, D., Fernández, M. F., Kay, M., Robie J., Siméon, J. (2006), XML Path Language (XPath) 2.0, W3 Consortium, March 15th, 2006, http://www.w3.org/TR/2005/ WD-xpath20-20050404/#id-references.

Bouillon, L., Limbourg, Q., Vanderdonckt, J., Michotte, B. (2005), Reverse Engineering of Web Pages based on Derivations and Transformations, Proc. of 3rd Latin American Web Congress LA-Web'2005 (Buenos Aires, October 31-November 2, 2005), IEEE Computer Society Press, Los Alamitos, pp. 3-13.

Bozzon, A., Comai, S., Fraternali, P., Toffetti Carughi, G. (2006), Capturing RIA concepts in a web modeling language, Proc. of the 15th International Conference on World Wide Web WWW'2006 (Edinburgh, May 23-26, 2006), pp. 907-908.

Chen, H., Dumais, S. (2000), Bringing order to the Web: automatically categorizing search results, Proc. of ACM Conf. on Human Aspects in Computing Systems CHI'2000 (The Hague, April 1-6, 2000), ACM Press, New York, 2000, pp. 145-152.

Crane, D., Pascarello, E., James, D. (2005), Ajax in Action,Manning Publications, USA.

Díaz Pérez P., Catenazzi N., Aedo Cuevas, I. (1997), De la Multimedia a la Hipermedia, Ed. Alfaomega, Spain.

FLEX (2006). http://www.macromedia.com/software/flex/

Franklin, M. and Zdonik, S. (1998), Data in your Face: Push Technology in Perspective, Proc. of ACM SIGMOD International Conference on Management of Data SIGMOD'98 (Seattle, June 2-4, 1998), ACM Press, New York, 1998, pp. 516-519.

Kolodner, J. (1993), Case-based Reasoning, Morgan Kaufmann Publishers, USA.

Martínez-Ruiz, F.J., Muñoz Arteaga, J., Vanderdonckt, J., González-Calleros, J.M. (2006), A first draft of a Model-driven Method for Designing Graphical User Interfaces of Rich Internet Applications, Proc. of 4th Latin American Web Congress LA-Web'2006 (Puebla, October 25-27, 2006), IEEE Computer Society Press, 2006.

Open Laszlo (2006). http://www.openlaszlo.org/

O'Rourke, C. (2004), a Look at Rich Internet Applications, Oracle Magazine, July - August 2004.

Payet, D. (2002), LYPO: vers une perception applicative du WEB, Proceedings of the 14th French-speaking Conference on Human-Computer Interaction IHM'2002 (Poitiers, November 26-29, 2002), Association Francophone de l'Interaction Homme-Machine, 2002, pp. 231-234.

Preciado, J.C., Linaje, M., Sanchez, F., Comai, S. (2005), Necessity of methodologies to model Rich Internet Applications, Proc. of 7th IEEE International Symposium on Web Site Evolution WSE'2005, IEEE Computer Society Press, 2005, pp. 7-13.

W3C, Editor Kesteren, A., XMLHttpRequest (2006), http://www.w3.org/TR/XMLHttpRequest/

XML User Interface Language (XUL) 1.0, Mozilla Foundation, 2006, http://www.mozilla.org/projects/xul/ xul.html.