Streamlining Complexity: Conceptual Page Re-modeling for Rich Internet Applications

Andrea Pandurino[1], Davide Bolchini[2], Luca Mainetti[1], and Roberto Paiano[1]

[1] University of Salento, Department of Innovation Engineering
Graphics and Software Architectures Lab
Via Monteroni - 73100 Lecce (LE), Italy
{andrea.pandurino,luca.mainetti,roberto.paiano}@unisalento.it
[2] Indiana University, School of Informatics at IUPUI
User Simulation and Experience Research Lab
535 W. Michigan St., 46202 Indianapolis, IN, U.S.A.
dbolchin@iupui.edu

Abstract. The growth of Rich Internet Applications (RIAs) calls for new conceptual tools that enable web engineers to model the design complexity unleashed by innovative interaction (with increasing communication potential) and to carefully consider the impact of the design decisions on the optimal flow of the User Experience (UX). In this paper we illustrate how is particularly relevant for RIA engineering not only to capture existing RIA technologies with suitable design artifacts but also to model an effective dialogue between users and RIA interfaces. Through a case study, we propose a set of conceptual design primitives (Rich-IDM) to enable web engineers to characterize the fluid, smooth and organic nature of the user interaction, and to take design decisions which meet both usability and communication requirements.

Keywords: User Experience, Rich Internet Application, Dialogue Modeling, Information Architecture, UX Requirements.

1 Introduction

The technologies enabled by RIA offer designers the opportunity to experiment with a novel interaction grammar that is radically changing the *dialogue rules* between the application and the user. The metaphor of the dialogue seems appropriate to represent the new communication aspects enabled by RIA technologies; in fact, for long time, Human-Computer Interaction (HCI) researchers assume that a sort of dialogue is established between the user and the interactive application during its use [2]. On this basis, every element of the application interface can be considered as a dialogue fragment that can be built using several "dialogue types and techniques" (such as form filling, menu selection, icons, direct manipulation, etc.) [8]. It is our opinion that RIAs are completely changing the core vocabulary of the dialogue for two reasons:

A. Harth and N. Koch (Eds.): ICWE 2011 Workshops, LNCS 7059, pp. 289–301, 2011.

(i) the set of interface primitives is raised through the introduction of new widgets; (ii) the interface primitives respond to more interaction events (such as mouse-over, drag-and-drop) than the corresponding primitives in a standard Web Application (WA). These new events allow users to enrich the dialogue with the application.

Two principal flaws could affect the design of RIAs if they are modeled using standard WA methods: (i) *underutilization* of the features of the RIA interface because the used methodology does not consider expressly the new primitives. Thus, the application interface has the same behavior of a standard WA and simply the system is re-written using a new technology; (ii) *weak use* of the RIA interface primitives: the designer models the new features of RIAs considering only the technological aspects, without evaluating the impact on the UX. This can cause serious problems to the interaction quality of the entire web application. To better understand this important aspect, in section 2 we report some examples of this defect using a commercial web site. Here we propose a design approach that can mitigate these flaws. Our approach, called Rich-IDM, can help designers to consider properly the RIA interface features, taking under control the communication aspects of the application. Rich-IDM can improve the dialogue between users and application, and, thus, the design of the UX because its primitives are characterized by a strong semantics (based on the metaphor of web-as-dialogue) derived by the information model. The dialogue is the bridging metaphor between the need to plan a product (design) and its UX, as defined in [11] "a representation of designers' hypotheses on experiences of the user needs or wants to have with the product in the future" or in [12] "a dynamic, context-dependent and subjective concept, which stems from a broad range of potential benefits users may derive from a product", which is in line with the UX definition proposed by ISO (2008) [10]: "A person's perceptions and responses that result from the use or anticipated use of a product, system or service."

The paper is organized as follows: section 2 provides the reader with a brief introduction to the poor RIA design problem caused by the introduction in RIA features considering only as technological improvements; section 3 gives a brief presentation of our methodological approach in order to address these issues; section 4 reports on key related work in the area of RIA engineering approaches and User eXperience (UX) requirements design; finally, in section 5 the conclusions summarize our key messages and sketch future research directions.

2 Potential and Weaknesses of RIA Design

The features of RIAs tend to exhibit potential flaws that can negatively affect the usability of the interaction. In detail: (i) at the micro-interaction level, the aesthetic impact of the presentation layer can obscure the real intent of the page at any given moment of the interaction, diverting the user's attention; (ii) recurring attention tunneling can easily bring users to misplace the saliency of the overall message of the page content. This problem can be summarized by implicit questions of the user: What is the designer showing me now? What is the main intent of this page? What is the message of this page?; (iii) users may have difficulty in capturing the underlying conceptual model of the designers, which should ideally match the user's mental model, thus starting an errant mental model; and (iv) the massive use of animations (such as sliding windows), that trigger continuous changes of the interface, greatly stresses user working memory by forcing users to recall their specific position in the local and global information architecture, the affordance of specific controls, and their location. In a previous work [16], we presented a set of case studies providing examples of flaws derived by the analysis of real web sites. For lack of space, here we limit our analysis to only one example.

Case study. A clear example of user working memory stressing can be found in the homepage of the Verizon corporate (Fig. 1), a communication carrier of the North America. The intent of the page is to show the features of the offered services. The page content is not so dense, but the page is long and requires the use of the mouse scroll to be completely viewed. The page presents various mechanisms that allow users to change (hide/unhide) completely the provided information without a page reload: (i) the menus in the highlighted area 1 (see Fig. 1) are composed of several items that inside have multiple columns with buttons and advanced options; (ii) the images of the area 3 allow to change completely the content showed in the area 2; (iii) the area 4 contains a set of messages "what's HOT" automatically updated or by user choice through the specific button in the same section; (iv) the area 6 controls the vertical banners, which are sliding elements with contents.

Fig. 1. www.verizon.com, homepage (April 2010)

Summarizing, there are 6 areas and 17 contents that can contain other dynamically showed elements.

The information presented *one-shot* to the user is few if compared to the global quantity of information of the page but, to have a complete schema of the page content, the user should access all the hided elements, which are mutually exclusive. Then, to reach specific product, the user must remember its position and the corresponding path; moreover, in order to compare two products (not displayed at the same time), the user must remember the features of the first while he/she is reading the other one. This situation is a clear example of poor user experience caused by a weak application of the RIA features.

In order to prevent the flaws described in the case study and the UX defects analyzed (but not reported) in this paper, in the following paragraph we present our conceptual approach called Rich-IDM.

3 Disciplined RIA Modeling for Improved UX Requirements

The new features of the RIA have changed radically the UX and put new questions that must be carefully evaluated during the design time. From this point of view, as described in the related work section, the existing approaches reach the goal to formalize the technological aspects of RIAs but they do not consider how to evaluate the changes in the UX and how it evolves. Thus, it is clear that it is necessary to define a methodological layer in which the single primitive has a well-defined communication semantics able to model the interaction and navigation paradigms of RIAs.

On the basis of these needs, in the following we present a conceptual approach (Rich-IDM) based on the Interactive Dialogue Model (IDM) [3]. First, we give a brief introduction to IDM (see Table 1), then we present all the primitives of Rich-IDM considering their notation and semantics (see Table 2). To give an example of the effectiveness of the Rich-IDM primitives to prevent and to correct UX poor usability situations, we report an artifact of the reengineering activity we did on the homepage of www.verizon.com, already used to describe the interface flaw.

3.1 A Brief Introduction to IDM

The idea to use the concepts of the dialogue as basis to describe the human-computer communication is not new. For long time, the research in the field of the Human–Computer Interaction (HCI) assumes that between the user and the interactive application a sort of dialogue is established during its use [1]. Often, in HCI literature, “dialogue” is used as synonymous “interaction”. On this basis, every element of the application interface such as the information retrieved from a database, the pop-up windows, the buttons, and other widgets, can be considered as dialogue fragments that can be built using several “dialogue types and techniques” (such as such form filling, menu selection, icons, direct manipulation, etc. [7]). Hence, the design of the interaction is often called “dialogue design” and, therefore, it is defined as the activity of modeling the structure of the conversation between the user and the system.

A complete model of a dialogue must describe all its aspects: the information and its structure, the relationships among information pieces and how this information must be showed and delivered to the user. Starting from this perspective and considering that often the new features of RIAs affect the quality of the dialogue, raveling the user interface and forcing the user to understand the interaction paradigms, we choose to extend IDM that is a dialogue-based design technique for shaping the communicative structure of information-intensive interactive applications.

IDM is based on proven hypermedia/web design concepts and dialogue theories. It can be used to describe the essential interactive and navigation features of information-intensive applications at the proper conceptual level, by focusing on the dynamics of the dialogue. The main advantages of IDM may be summarized as follows: a) easiness of use and understandability of the design primitives employed with respect to their expressive power; b) primitives semantics based on dialogue concepts, thus more accessible by novice designers without a technical background; c) separation between channel-independent (or technology-independent) design (determining the expected deep structure and dynamics of the dialogue) and channel-dependent design (conceptual specification for the applications available on different devices). IDM primitives are organized in two main design layers: Conceptual IDM (C-IDM) and Logical IDM (L-IDM).

C-IDM is used to describe the "conceptual schema" of the application. It is simple to grasp and effective enough in representing the most relevant features of the application, defining the topics of the dialogue and relations between its elements; in other word, it is used to shape the deep dialogic structure of the interaction. Starting from the C-IDM design, the logical design models the decisions that are typically dependent on a specific fruition channel through which the application may be conveyed. The conceptual schema is unique in the application because defines the overall interaction strategy; while, the designers can develop one or more logical schemas, one for each specific channel they want to design the application for.

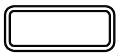
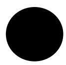
The L-IDM is used to shape the application dialogic features specific of a given channel or technologies of fruition such as standard web browsers, mobile devices, screen readers, etc. IDM breaks down the application information (according to its semantics) defining the topics (core content entities) and dialogue acts (interaction units) in a L-IDM schema. Considering the goal of this paper, the main L-IDM primitives (used in the case studies) are described in the Table 1.

Readers interested in a complete introduction to IDM can refer to [3]. Currently, IDM is being used in several research and industrial projects allowing capturing the dialogue features of the applications, and providing a valid design to project the non-technical aspects.

3.2 The Rich Extension of IDM

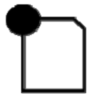
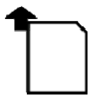
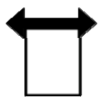
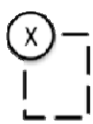
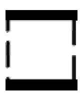
In the rest of this section we provide readers with a brief introduction to Rich-IDM, which is our extension of IDM to cover RIAs. Table 2 shows all Rich-IDM design primitives at a glance; after, for each primitive, we describe its semantics and its specific features.

Table 1. The IDM design primitives.

Name	Notation	Design Semantics
Topic / Multiple Topic	 Topic  Multiple Topic	It is the dialogue subject: the argument of a dialogue between user and application. A topic should contain information with a precise sense for the final user independently from the application and from the arguments presented inside. If the topic has more instances, it is a multiple topic.
Content Dialogue Act		A piece of dialogue that represents contents for users. The information of a topic can be structured using the content dialogue acts.
Transition Dialogue Act		A piece of dialogue that allow users to navigate from a topic to an other one. Its goal is to enable users to change dialogue arguments, following semantic relationships.
Introductory Dialogue Act / Parametric Dialogue Act	 Introductory  Parametric	A piece of dialogue that allows starting a dialogue from a specific topic. The main message of an introductory act is a list of instances of the same topic. It may be multiple.
Relevant Relation		A relevant semantic relation represents the possibility to move the attention from a topic to other one that is semantically related to the same argument.

RIA-Page Element. In Rich-IDM, the minimum piece of information is called RIA-Page Element and it is managed as a unique block. In detail, the RIA-Page Element is defined as a coherent atomic fragment of RIA page, which displays a specific content with its proper meaning for the users. It could be specialized in: (i) Introductory RIA-page Element which main goal is to introduce the specific content and, often, is related to an introductory dialogue act; (ii) Content RIA-page Element that, mapping one or more content dialogue acts, displays to the user the payload of the dialogue; and (iii) Transition RIA-page Element that shows the semantic information (taken from transition dialogue act) that links two topics. The specific semantics of this set of primitives allows establishing a direct link with the functions of the contents inside. Thus, it is possible to evaluate in the early phase of the design the balance between the different types of delivered contents. The UX model is improved because the designer has the tools to assess into the page the correct quantity of information, avoiding pages without contents and stuffed with links represented by graphical elements that are more usual in RIAs.

Table 2. The Rich-IDM design primitives

Name	Notation	Design Semantics
Content RIA-Page Element		A coherent, atomic fragment of RIA page, which displays a *content* unit, as directly mapped from IDM content dialogue acts.
Introductory RIA-Page Element		A fragment of a RIA page which displays mechanisms to enable *access* to multiple instances of a dialogue topic, as directly mapped to an introductory dialogue act of the IDM logical design.
Transition RIA-Page Element		The reification of an IDM transition dialogue act on the RIA page. It allows users to follow the semantic relation of two dialogue topics.
RIA-Handle		An interaction affordance, which enables users moving within two or more page elements of the same User Experience Core.
User Experience Core		A connected composition of page elements, which communicates the semantic nucleus of what is offered to the user at a given moment.
Context View		A set of User Experience Cores, which maintains navigational context, orientation, organic, and fluid transition between the cores.
Default Element		Indicate the default RIA-Page Element showed to the user.

RIA-Handle. After the definition of the RIA-Page Elements, that are the interaction objects, it is necessary to define a new primitive able to model the mutual relation between the elements of the page. This new concept is the RIA-Handle, which main goal is to model all the dynamic aspects of the UX. The RIA-Handle is a directional relationship between the RIA-Page Elements involved in a user action. The RIA-Handle captures the syntactic of RIA interaction, on top of the semantics modeled by the other elements. From the methodological point of view, the RIA-Handle allows designers to represent all the relations contained in the information architecture.

User Experience Core. In RIAs, many information elements (that could not be semantically directly connected) can be collapsed in the same page. Thus, the core of the dialogue is not directly related with the displayed elements. To satisfy this need, the User Experience Core is defined. Its main goal is to model clearly the elements of the Rich-IDM design that must be the heart of the dialogue with the users. The

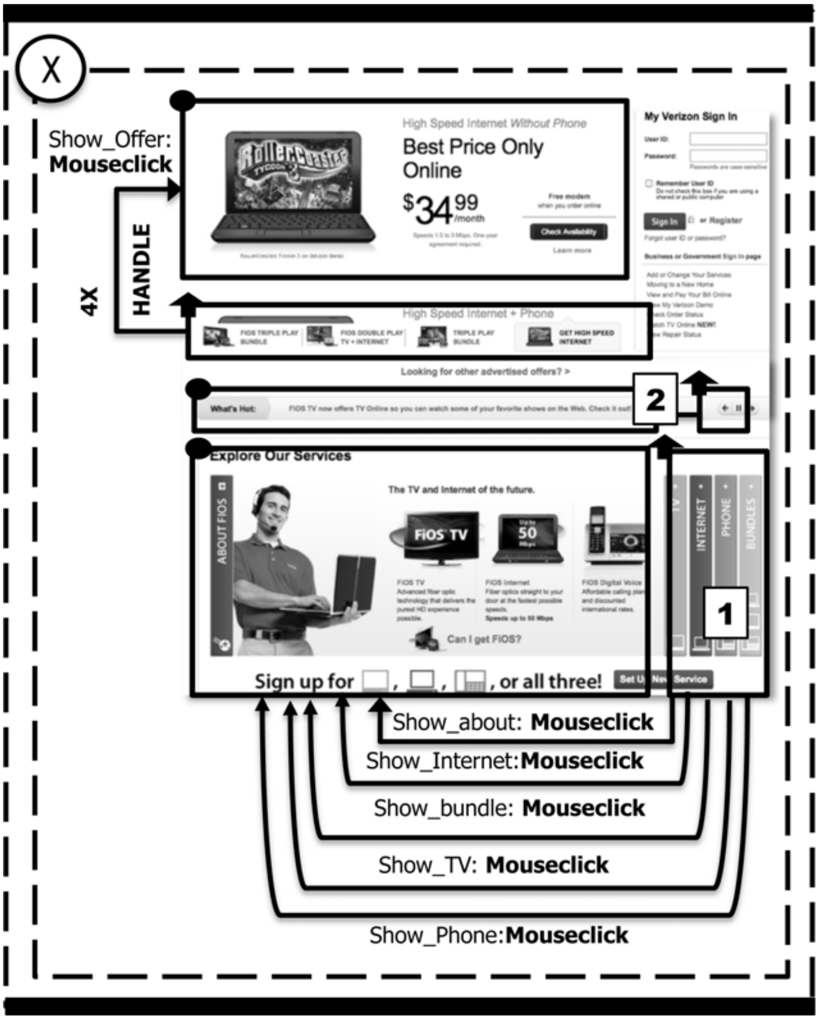


Fig. 2a. The AS-IS model of the www.verizon.com homepage (April 2010) described using the Rich-IDM notation

designer must carefully manage the elements contained in the User Experience Core because their perception affects strongly the sense and the quality of the message delivered to the user. On this basis, the semantics of the User Experience Core is to define the unit of perception of the dialogue. Formally, the User Experience Core is a container of the RIA-page Elements. At the start of the navigation, the default RIA-page Elements showed to the user is marked with the Default Element described in Table 2. The RIA-Handle mechanism is used to model the navigation between the User Experience Cores independently if they are (or not) part of the same Context View.

Context View. The look&feel of the RIA is relevant and often the visualization aspects are used to define specific areas of the application and delimitate related arguments. To capture these characteristics crucial for the quality of the dialogue flow, designers need to define the User Experience Cores that must be shown in the same way to the user. The Context View enables designers to define a specific navigational context allowing harmoniously connecting related User Experience Cores.

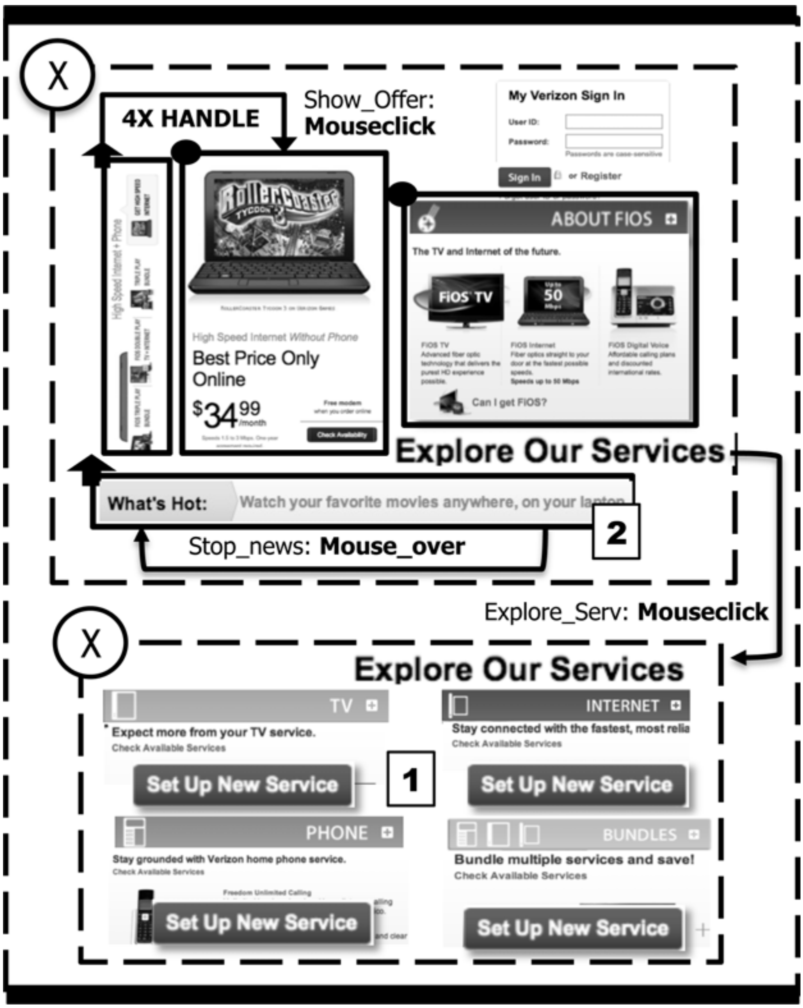


Fig. 2b. The TO-BE model of the www.verizon.com homepage reengineered using Rich-IDM. The UX has been transformed reducing the stress of the user working memory to compare options. This model avoids the sliding elements (lower UX Core) of the AS-IS page (Fig. 2a) and allows the user to compare the services in a one-shot view (upper UX Core). Both UX Cores are contained in a unique Context View that guarantees a uniform navigational context.

Formally, the Context View is a container of User Experience Cores. Its name is due to the idea that the User Experience Cores of the same view are shown to the user in uniform manner, thus providing a common (and stable) visualization environment to the user. In order to better explain the usefulness of Rich-IDM to improve the UX, we report in Fig. 2b the Rich-IDM reengineering of a homepage fragment of the Verizon web site; then we compare the TO-BE model with the AS-IS model showed in Fig. 2a.

At the beginning of the reengineering phase, in the TO-BE model we created two User Experience Cores: the first one groups the information about the company and the news; the second one (in the bottom of Fig. 2b) shows the information contained in the sliding elements of the Fig. 2a (marked with box "1"). Then, we focused on the middle of the homepage where there is the "What's up" news (marked with box "2") with a button to stop them. In Fig. 2b, we have removed the control button and we have implemented a "mouse over" event that the user can use to block the news.
At the end of the reengineering activity, we have removed six navigation links and one button, and we have introduced a new User Experience Core that allows users to compare all the carrier's products.

4 Related Work

Researchers approach the RIA design mainly modeling: (i) the information of the application, the core objects and their behavior; (ii) the navigation through the information nodes; (iii) the interface as "what the user perceives"; and (iv) the interaction between the user and the application, specifying the available events on the interface.

The Object Oriented Hypermedia Design Method [19] (OOHDM) proposes a model process structured according five steps: requirements modeling, conceptual modeling, navigation design, interface design, and implementation. The interface design is defined using the Abstract Data View (ADV) [5] that enables specifying the status and the behavior of each interface objects using state charts.

WebML for RIAs [4] extends the WebML method considering two aspects: (i) a well-defined separation between the client side and the server side; (ii) a better definition of the application interface. The data design is based on the Entity-Relationship (E-R) model that is extended considering the levels of persistence. The business logic model provides the specification of operations at the client side and server. WebML adopts the Rich Internet Application User eXperience (RUX) model [13] to design the interface aspects.

The RUX method defines the interface of an application through four levels: concepts and tasks, abstract interface, concrete interface, and final interface. The concepts and tasks level describes the data and business logic, and can be modeled using a web design methodology such as WebML. The abstract interface level describes the aspects of the interface common to all RIA technologies. The concrete interface level is the implementation of the abstract interface and it defines three presentation sublevels: spatial presentation, temporal presentation, and interaction presentation. The final interface level translates the designed model into the specific RIA concrete technology.

UML-based Web Engineering [17] (UWE) is a method for systematic and model-driven development of web applications. It exploits an UML profile to provide a specification of the domain-relevant information of a web system. To design RIAs, UWE integrates the RUX method. UWE propose to apply patterns [18] at a high abstraction level to minimize the design efforts and to maximize the expressiveness of model artifacts, describing the behavior of the RIA features. UWE-R [14] is a light-weighted extension of UWE for RIAs, covering navigation, process and presentation aspects.

OOH4RIA [15] has the main goal to cover the entire development lifecycle of RIAs. It is based on model-driven approach that specifies the artifacts to model a complete RIA for the GWT framework [9]. The starting point of the OOH4RIA design is the definition of the OOH domain model to represent the domain entities and the navigational connections. Also, OOH4RIA enable transforming the navigation model into the presentation model.

OOWS [20] is a methodological approach to develop web applications in an OO modeling oriented software development environment. It integrates appropriate models to capture the structure, behavior, navigation and presentation requirements of a web application. Also, it proposes an extension to support Web 2.0 application development.

ADRIA [6] is method for designing RIAs departing from the results of an object-oriented analysis; it employs interaction spaces as the basic abstraction mechanism coherently throughout all the design activities; its notation is based on UML.

Internet Application Modeling Language (IAML) [21] aims to provide modeling support for all of the fundamental concepts of RIAs. Along with operations and domain objects, it models events and conditions as first-class citizens, also promoting users and security as first-class. It uses concepts from existing languages where appropriate, such as ECA rules, ER diagrams, and UML Activity and Class diagrams.

Whereas these approaches provide support for abstracting existing RIA technologies and to design (and generate) the development artifacts, they lack in bridging the fluid, smooth and organic nature of the user interaction and navigation in RIAs to the design.

To meet this challenge, we propose to extend this perspective to examine the connection between RIA interface modeling and the requirements for the user experience. This perspective is only partially covered by existing works (RUX and ADV, in particular).

5 Conclusions

Due to the superior user interaction unleashed, RIAs require web engineers to balance the potential sophistication of the user interface and the need to ensure proper usability, cognitive workload, and efficiency. To meet this challenge, we have proposed a set of high-level modeling constructs, which bridges UX requirements and RIA design. Through a case study and extending the IDM method, we have shown some relevant features of our approach: expressiveness to capture interaction grammars and semi-formality to facilitate the establishment of a common ground between UX designers and web engineers. In detail, we proposed new primitives with a strong semantics: the User Experience Core, the Content / Introduction / Transition RIA-Page Element, the RIA-Handle, and the Context View.

Future research will concern the execution of measurements exploiting model metrics we are working on, in order to provide quantitative feedbacks on the level of transformative improvements the Rich-IDM can introduce in UX design of RIAs.

References

1. Andersen, P.B.: A Theory of Computer Semiotics. Cambridge Univ. Press, Cambridge (1997)
2. Andersen, P.B.: A Theory of Computer Semiotics. Cambridge University Press (1997) [2]
3. Bolchini, D., Paolini, P.: Interactive Dialogue Model: A Design Technique for Multi-Channel Applications. IEEE Trans. on Multimedia 8(3), 529–541 (2006)
4. Ceri, S., Fraternali, P., Bongio, A.: Web Modeling Language (WebML): a modeling language for designing Web sites. Computer Networks: The International Journal of Computer and Telecommunications Networking 33(1-6), 137–157 (2000)
5. Cowan, D., Pereira de Lucena, C.: Abstract Data Views: An Interface Specification Concept to Enhance Design for Reuse. IEEE Trans. on Software Eng. 21(3), 229–243 (2005)
6. Dolog, P., Stage, J.: Designing Interaction Spaces for Rich Internet Applications with UML. In: Baresi, L., Fraternali, P., Houben, G.-J. (eds.) ICWE 2007. LNCS, vol. 4607, pp. 358–363. Springer, Heidelberg (2007)
7. Hewett, T., Baecker, R.M., et al.: Dialogue Techniques. In: ACM SIGCHI Curricula for Human-Computer Interaction ACM Special Interest Group on Computer-Human Interaction Curriculum Development Group, `http://sigchi.org/cdg/cdg2.html`
8. Hewett, T., Baecker, R.M., et al.: Dialogue Techniques. ACM SIGCHI Curriculum Development Group, `http://www.sigchi.org`
9. Houben, G.J., Van der Sluijs, K., Barna, P., Broekstra, J., Casteleyn, S., Fiala, Z., Frasincar, F.: Hera: Chapter 10. In: Web Engineering: Modelling and Implementing Web Applications. HCI Series, pp. 263–301. Springer, Heidelberg (2008)
10. ISO DIS 9241-210:2008. Ergonomics of human system interaction - Part 210: Human-centred design for interactive systems (formerly known as 13407). International Organization for Standardization (ISO), Switzerland (2008)
11. Kankainen, A.: Thinking model and tools for understanding user experience related to information appliance product concept. In: Dissertation of Degree of Doctor of Philosophy, pp. 1–59. Helsinki University of Technology, Espoo (2002)
12. Lai-Chong Law, E., Roto, V., Hassenzahl, M., Vermeeren, A., Kort, J.: Understanding, scoping and defining user experience: a survey approach. In: Proc. of the 27th International Conference on Human Factors in Computing Systems (CHI 2009), pp. 719–728. ACM, New York (2009)
13. Linaje, M., Preciado, J.C., Sánchez-Figueroa, F.: A Method for Model Based Design of Rich Internet Application Interactive User Interfaces. In: Baresi, L., Fraternali, P., Houben, G.-J. (eds.) ICWE 2007. LNCS, vol. 4607, pp. 226–241. Springer, Heidelberg (2007)
14. Machado, L., Filho, O., Ribeiro, J.: UWE-R: An Extension to a Web Engineering Methodology for Rich Internet Applications. WSEAS Trans. Info. Sci. and App. 6(4), 601–610 (2009)

15. Meliá, S., Gómez, J., Pérez, S., Diaz, O.: A Model-Driven Development for GWT-Based Rich Internet Applications with OOH4RIA. In: Proc. of ICWE 2008, pp. 13–23. IEEE, New York (2008)
16. Pandurino, A., Bolchini, D., Mainetti, L., Paiano, R.: Rich-IDM: Extending IDM to Model Rich Internet Applications. In: Proc. of 12th ACM iiWAS Int. Conf., pp. 145–152 (2010)
17. Preciado, J.C., Linaje, M., Morales-Chaparro, R., Sanchez-Figueroa, F.: Designing Rich Internet Applications Combining UWE and RUX-Method. In: Proc. of ICWE 2008, pp. 148–154. IEEE, New York (2008)
18. Preciado, J.C., Linaje, M., Morales, R., Sánchez-Figueroa, F., Zhang, G., Kroiß, C., Koch, N.: Designing Rich Internet Applications Combining UWE and RUX-Method. In: Proc. of 8th Int. Conf. on Web Engineering (ICWE 2008), pp. 148–154. IEEE, New York (2008)
19. Schwabe, D., De Almeida Pontes, R., Moura, I.: OOHDM-Web: an environment for implementation of hypermedia applications in the WWW. ACM SIGWEB Newsl. 8(2), 18–34 (1999)
20. Valverde, F., Pastor, O.: Facing the Technological Challenges of Web 2.0: A RIA Model-Driven Engineering Approach. In: Vossen, G., Long, D.D.E., Yu, J.X. (eds.) WISE 2009. LNCS, vol. 5802, pp. 131–144. Springer, Heidelberg (2009)
21. Wright, J.M., Dietrich, J.B.: Requirements for Rich Internet Application Design Methodologies. In: Bailey, J., Maier, D., Schewe, K.-D., Thalheim, B., Wang, X.S. (eds.) WISE 2008. LNCS, vol. 5175, pp. 106–119. Springer, Heidelberg (2008)