

## A framework to author educational interactions for geographical web applications

The Nhan Luong<sup>1</sup>, Thierry Nodenot<sup>1</sup>, Philippe Lopistéguy<sup>1</sup>  
and Christophe Marquesuzaà<sup>1</sup>

IIUT de Bayonne Pays Basque, LIUPPA-DESI, 2 Allée du Parc Montaury  
64600 Anglet, FRANCE  
{thenhan.luong, thierry.nodenot, philippe.lopisteguy,  
christophe.marquesuzaa}@iutbayonne.univ-pau.fr

**Abstract.** This paper focuses on the production of authoring tools that teachers may use to prototype interactive geographical web applications. We present some computational models and a toolset that we designed to address some needs of teachers trying to make use of particular localized documents called “travel stories”. Our research challenge is to enable teachers to design interaction scenarios for such a domain, avoiding any programmer intervention. In the design process, the teacher typically faces three activities: (a) Identification of candidate documents, (b) Evaluation of the adequacy of the document and (c) Production of the learning application making use of the selected document. In this paper, we mainly focus on the (c) Production activity. We highlight the necessary use of an “agile” approach to shorten as much as possible the delay between the design and the evaluation step of a prototype. To address the technological challenges raised by such an aim, we present WIND framework and we discuss its capabilities while considering some examples of interactive scenarios generated with WIND framework.

**Keywords:** geographic information, interaction design, interaction programming, agile approach, web application, authoring framework.

### 1 Introduction

Educational scenarios are particular design artifacts that take advantage of current enhancements in the domain of “Science of Design” [1]. Research works dedicated to the design of educational scenarios propose new paradigms, concepts, approaches, models and theories that promote stronger bases for the design of TEL environments [2]. These bases are foundations enabling to improve the processes of both coding, evaluating and maintaining such type of application. Recent works focus on:

- the definition and role of a pedagogical scenario [3, 4],
- the definition of visual instructional languages [5] and executable [6] scenarios,
- the definition and the evaluation of methodological principles allowing designers to produce and to re-use such scenarios [7].

Nowadays, there is also a strong emphasis on the process of transforming an abstract scenario (that a teacher is able to understand and to design by himself) into an executable scenario (that an execution infrastructure can process). Several works promote model-driven engineering techniques and tools to fully integrate the functions supported by the chosen infrastructure. This approach is interesting not only because of underlying model-driven scientific challenges [8], but also because model-driven transformations must respect educational constraints specified in the scenario produced by the teacher [9].

Most of identified works are still in progress and it is thus difficult to know if they will soon provide teachers with toolboxes fitted to the design and the implementation of constructivist learning situations. Moreover the complexity of required technologies may put the teacher out of the play (when the educational scenario becomes very detailed, when the scenario is deployed on a target infrastructure...). In most works, the author is a pedagogic engineer. Such designer profiles can be found in e-learning firms but not in most classrooms and teaching institutes: though they are not computer-scientists, teachers must fully handle the design process (from the scenarization step to the deployment step).

This paper focuses these particular teachers: we propose a framework that they may use to edit/prototype and to evaluate by themselves an educational scenario. This framework targets the scenarization of interactive resources to be used in inquiry learning activities [10] for a specific applied domain: travel stories.

In the second section, we present the background and the objectives of these research works. This leads us to present in the third section WIND framework that facilitates an “agile” production of interactive scenarios. The conclusion section proposes a synthesis about WIND and our future directions of investigation.

## 2 Background and Objectives

The DESI<sup>1</sup> group aims to propose software architectures and tools to re-vitalize localized documents that generally rest in the depth of archives, museums and libraries. In particular, travel stories offer very challenging revitalization objectives because tourists and teachers could benefit from e-services developed from such localized documents. Travel stories have intrinsic characteristics that make them good teaching-resource candidates. A travel story is a sort of text whose author tells what he discovered while travelling through a territory or a country. On the one hand, the author tries to very precisely present the places he/she visited; on the other hand, he/she tries to tell the events that occurred during his/her trip, he/she reports on his/her activities and explorations. Indeed, the travel story aims at using words to describe the travel reality: the travel story is told day after day, the duration of the trip is often explicitly written in the text in conjunction with the travelled locations. Moreover, travel stories provide an opportunity to ground the design and the operation of systems with text, map and calendar components that require extensive human-machine interaction.

Following the experiment available at <http://erozate.iutbayonne.univ-pau.fr/forbes2007/exp/>, we proposed three authoring steps to assist as much as

1 DESI is a French acronym that means Electronic Documents, Semantics and Interaction.

possible the process of authoring educational applications making use of travel stories. The first teacher's task consists in selecting in a corpus of documents the travel stories that deal with geographical areas [11, 12]. The second task consists in evaluating the adequacy of the document as regards of teacher's pedagogical aims [13]. The third teacher's task consists in producing a highly interactive application based on the semantics of the validated travel story. To this end, the teacher needs an authoring environment that enables him/her to quickly evaluate/correct his/her conceptual choices. This paper focuses on this third task.

Following A. Gibbons' works [14, 15], our approach aims at breaking such a design dichotomy/gap for the particular case of educational applications making use of travel stories. Indeed, Gibbons' instructional design layers are: content, strategy, control, message, representation, media-logic, data-management. Design of each layer can be considered separately from the other layers, providing an important modularization to the design effort. Applied to our application domain (study and design of educational applications making use of geographical information embedded in travel stories), Gibbons' design theory leads us to design interactions at four levels of abstraction, as suggested in [16]:

1. The most elementary level deals with the data and geographical information embedded/retrieved in a text (*cf.* the data-management, media-logic and representation layer) that may be associated with goals and activities (input and output parameters), participants, artifacts provided to participants (*e.g.* map, text, calendar components).

2. The second level (*cf.* the message layer) focuses on the messages exchanged during an interaction, their structure and the way they are generated from the data and geographical information manipulated by a learner. It also enables to control the execution of the interaction model in accordance with the satisfaction of a certain conditional expression. For example, an interaction may become mandatory depending on previous interactions with an icon representing a particular place mentioned in a travel story.

3. The third level (*cf.* the control layer) considers the possibility to introduce decisions and commands that enable to change the behavior of interaction scenarios according to the aims of participants having specific rights for the considered educational unit. For example, a learner can decide to mask the calendar component because he/she wants to ignore the travel story's chronology.

4. The fourth level (*cf.* the strategy layer) considers the use of events to decide the performance of changes: events can dynamically occur during execution, they can be triggered from the evaluation of time conditions, goals achievement, activity reports. For example, if the learner never interacts with the map component, this means that he/she probably needs some support to take advantage of the map component capabilities.

In the next section, we present a framework that addresses interaction design according to these four abstraction levels.

### **3 A Framework to Facilitate an “Agile” Production of Interactive Scenarios**

WIND favors empirical design approaches enabling a teacher to easily formalize and

evaluate his/her educational ideas by using (as a learner) the automatically generated application. Evaluation step is therefore used to check/criticize his/her pedagogical choices. We thus define the concept of an “agile” design tool as a piece of software supporting such an approach. Indeed, we may define an “agile” method as a design approach not only by fully implying the end-user along the whole process but also by rapidly integrating his/her requirements in a technical solution [17]. Final quality of the generated application is ensured thanks to a continuous control all the production process along. As a consequence, each teacher’s pedagogical choice must be fully and automatically traduced into executable code. This constraint implies the use of an applicative model as a design framework. An applicative model is an application generic model that may be instantiated all the design step along; each instance of this model is then automatically traduced into executable code. Our proposed design approach is based on a model-driven approach [18, 19] which is also used for TEL engineering [8]. We distinguish three levels:

1. The *generic applicative model* describes the core concepts of the application classes.
2. The *application model* created by the teacher during his/her design step. This model is an instance of the previous generic model. It describes the characteristics of the application desired by the teacher.
3. The *code* generated from the application model designed by the teacher.

The WIND *generic applicative model* [20] defines the core of the interactive web applications that a teacher will be able to produce. An interaction may be simply defined as a triple  $\langle \text{area}, \text{event}, \text{reaction} \rangle$ . Interaction is the central mechanism which characterizes the applications we wish to develop. The expressive capacity of the interactions may be declined to express not only simple interactions but also more complex ones.

This WIND *generic applicative model* is described in a WIND-XSD schema (available at <http://erozate.iutbayonne.univ-pau.fr/Nhan/WINDv2/schema.xsd>). Each concept of the model is described as a specific element. This XSD schema helps to instantiate WIND *generic applicative model* into XML format that describes the interactions of a WIND *application model*. That is to say it permits to describe a web-based application embedding textual, map and calendar interactive components.

Taking advantages of JavaScript, WIND *generic applicative model* is supported by a WIND-API that implements the different classes as well as their associated methods. WIND-API proposes a homogeneous layer built on lower level APIs, specialized in the handling of text, map and calendar.

To avoid any programmer intervention in the teacher activity devoted to the application production, we have developed a JavaScriptCodeGenerator<sup>2</sup>. The JavaScriptCodeGenerator can parse any WIND-XSD compliant data file (e.g. a WIND *application model* description) in order to generate JavaScript code for interactions that WIND API can execute. These technologies enable us to shorten the delay between the design and the evaluation step of a prototype. The implementation of WIND interactions may simply be done with four main steps<sup>3</sup> :

<sup>2</sup> For example, see the XML file at <http://erozate.iutbayonne.univ-pau.fr/Nhan/WINDv2/data.xml> and the automatically produced web application that the end-user can exploit: <http://erozate.iutbayonne.univ-pau.fr/Nhan/WINDv2/generator.php?file=data.xml>.

<sup>3</sup> A complete example is available at <http://erozate.iutbayonne.univ-pau.fr/Nhan/WINDv2/>

1. Defining the components of the application and their characteristics.
2. Defining reactive areas for each component defined in the previous step.
3. Defining possible reactions for the reactive areas.
4. Defining interactions upon previously defined reactive areas and reactions.

## 4 Discussion and Future Directions

WIND is an operational framework that allows both describing and implementing interactions of web applications mixing texts, maps and calendars. This framework promotes an agile process fitted to designers without computer-science skills: its characteristics make easier the description of interactions. Yet, WIND still needs further developments. We must extend our current framework because WIND does not completely address the design of the four interaction layers presented in section 2.

Current version of WIND framework correctly addresses the data-layer: it enables designers to manage sensible parts corresponding to the main concepts (places, dates, movement-verbs, etc.) automatically retrieved in travel stories. As a consequence, interaction design can take advantage of these specific sensible parts. However, we still need to extend WIND framework to manage the same way more complex concepts of an itinerary.

Current version of WIND framework enables a designer to specify the messages exchanged with a learner, the semantics of the messages and their appearance. Moreover, conditional messages are easy to describe with WIND functionality, thus satisfying the requirements of the second level/layer.

Current version of WIND framework enables a designer to specify who controls an interaction, how it is initiated, what the learner's degrees of liberty are. WIND provides functionality needed to design mixed-initiative interactions, thus satisfying the requirements of the third level/layer.

However, current version of WIND framework fails to completely address the strategic layer because WIND does not provide any functionality to assess cognitive processes. The event-reaction mechanism implemented by WIND provides the required functionality to design reactions on cognitive events, but we do not currently provide designers with any means to detect such cognitive events.

Our first experiments have shown that WIND is really helpful to rapidly design and assess inquiry activities making use of the semantics of travel stories. Directions of research discussed above will certainly enhance the instructional design added-value of WIND framework. We also need to propose an evaluation protocol to determine to which extent can teachers concretely exploit the current WIND framework and its corresponding authoring tools.

## Acknowledgments

This research is supported by the French Aquitaine Region (project n°20071104037) and the Pyrénées-Atlantiques Department ("*Pyrénées : Itinéraires Educatifs*" project).

## References

1. NSF 2007, Science of Design: National Science Foundation 07-505, Program Solicitation, from [www.nsf.gov/publications/pub\\_summ.jsp?ods\\_key=nsf07505](http://www.nsf.gov/publications/pub_summ.jsp?ods_key=nsf07505)
2. Tchounikine, P.: Pour une ingénierie des Environnements Informatiques pour l'Apprentissage Humain. *Information Interaction Intelligence*, vol. 2(1), pp. 59-93 (2002)
3. Pernin, J.-P., Emin, V. and Guérayd, V.: ISiS: An Intention-Oriented Model to Help Teachers in Learning Scenarios Design. In: *Second European Conference on Technology Enhanced Learning*, pp. 338-343 (2008)
4. Dillenbourg, P. and Tchounikine, P.: Flexibility in macro-scripts for computer-supported collaborative learning. *Journal of Computer Assisted Learning*. vol. 23(1), pp. 1-13 (2007)
5. Nodenot, T.: Scénarisation pédagogique et modèles conceptuels d'un EIAH: Que peuvent apporter les langages visuels? *International Journal of Technologies in Higher Education*, vol. 4(2), pp. 85-102 (2007)
6. Ferraris, C., and Martel, C.: LDL for Collaborative Activities. In L. Botturi & T. Stubbs (eds.), *Handbook of Visual Languages in Instructional Design; Theories and Practices*, pp. 226-253: Hershey, PA: IDEA Group (2007)
7. Villiot-Leclercq, E. : Modèle de soutien à l'élaboration et à la réutilisation de scénarios pédagogiques. Doctorat en Sciences Cognitives de l'Université Grenoble 1 (2007)
8. Laforcade, P. ; Nodenot, T. ; Choquet, C., Caron, P.A.: MDE and MDA applied to the Modeling and Deployment of Technology Enhanced Learning Systems: promises, challenges and issues. In *Architecture Solutions for E-Learning Systems* (2007)
9. Caron, P.-A.: Web Services Plug-in to Implement "Dispositives" on Web 2.0 Applications. In: *Second European Conference on Technology Enhanced Learning*, pp. 457-462 (2007)
10. Olson, S., Loucks-Horsley, S.: *Inquiry in the National Science Education Standards: a guide for teaching and learning*, Olson and Loucks-Horsley. National Academies Press (2000)
11. Loustau, P., Nodenot, T., Gaio, M. 2009, Design principles and first educational experiments of  $\pi R$ , a platform to infer geo-referenced itineraries from travel stories *International Journal of Interactive Technology and Smart Education (ITSE)* (2009)
12. Gaio, M., Sallaberry, C., Etcheverry, P., Marquesuzaà, C., Lesbegueries, J.: A Global Process to Access Documents' Contents from a Geographical Point of View. In *Journal of Visual Languages and Computing*, vol.19, pp. 3-23 (2008)
13. Loustau, P., Nodenot, T., Gaio, M. "Spatial decision support in the pedagogical area : Processing travel stories to discover itineraries hidden beneath the surface", 11th AGILE International Conference on Geographic Information Science, pp. 359-378, (2008)
14. Gibbons, A. S.: What and how do designers design? A theory of design structure. *TechTrends*, vol. 47(5), pp. 22-27 (2003)
15. Gibbons, A., & Stubbs, T. Using Instructional Design layers to categorize design drawings. In L. Botturi & T. Stubbs (Eds.), *Handbook of Visual Languages in Instructional Design; Theories and Practice*: Hershey, PA: IDEA Group (2007)
16. Caeiro-Rodríguez, M., Llamas-Nistal, M., Anido-Rifón L.: The PoEML Proposal to Model Services in Educational Modeling Languages. In *Groupware: Design, Implementation, and Use*. LNCS, vol. 4154, pp. 187-202, Springer, Heidelberg (2006)
17. Vickoff, J.P.: *Systèmes d'information et processus agiles*. Hermes Science (2003)
18. Seidwitz, E.: What models mean. *IEEE Software*, pp 26-32 (2003)
19. Bézivin, J., Blay, M., Bouzeghoub, M., Estublier, J., Favre, J.-M.: *Rapport de synthèse de l'Action Spécifique CNRS sur l'Ingénierie Dirigée par les Modèles: Action Spécifique MDA du CNRS* (2005)
20. Luong, T.N., Etcheverry, P., Nodenot, T., Marquesuzaà, C.: WIND: an Interaction Lightweight Programming Model for Geographical Web Applications. In *International Opensource Geospatial Research Symposium (OGRS)* (2009). To appear.