TOWARDS A GENERIC AND CONFIGURABLE MODEL OF AN ELECTRONIC INFORMER TO ASSIST THE EVALUATION OF AGENT-BASED INTERACTIVE SYSTEMS

Chi Dung Tran, Houcine Ezzedine and Christophe Kolsky
LAMIH – UMR CNRS 8530, University of Valenciennes and Hainaut-Cambrésis
Le mont Houy, 59313 Valenciennes cedex 9, France

Keywords: Human-computer interaction, supervision, interactive system architecture, evaluation of interactive systems, agent-based interactive systems, electronic informer.

Abstract: This paper presents a generic and configurable model of an electronic informer to assist the evaluation for agent-based interactive systems. In order to propose this model, the current state of the art concerning architectures used for traditional interactive systems is presented, along with that for agent-based interactive systems. In this article, we propose an agent-oriented architecture that is considered as being mixed (it is both functional and structural). By using this architecture as a basis, we propose a generic and configurable model of an evaluation tool called "electronic informer" which assist evaluators in analysing and evaluating interactive systems of this architecture.

1 INTRODUCTION

Nowadays, in spite of the existence of several methodologies for the development of interactive systems, designing, developing and assessing, in terms of *utility* and *usability* (Bastien and Scapin, 1995; Nielsen, 1993; Shneiderman, 1998) an agents-based interactive applicative system is still a difficult task. It is therefore necessary to provide methods, models and evaluation tools to make it easier. A lot of research work has dealt with the specification and design of object-oriented interactive systems as well as agent-based interactive systems. However, the choice of an architecture which is adequate for the system is not on easy task. Furthermore, in the majority of cases, the current evaluation methods do not take into account the specific architecture of an agent-based interactive system (Trabelsi et al., 2004) and there are rarely propositions concerning the coupling between the architecture and evaluation phase (Trabelsi, 2006).

The section 2 presents a brief state of the art concerning the architectures for traditional interactive systems as well as for agent-based interactive systems. The section 3 proposes an architecture which is both functional and structural, and which provides a separation into three functional components (Ezzedine et al., 2003). By using this architecture as a basis, in the section 4, we propose a generic and configurable model of an evaluation tool called "electronic informer"; its aims at assisting the evaluation of agent-based architecture systems.

2 INTERACTIVE SYSTEM ARCHITECTURES

Architecture of an interactive system supplies the designer with a generic structure from which he/she can build an interactive application. It is a set of structures that include: components, the outside visible properties of these components and the relations between them. Researchers have proposed several architecture models over the past twenty years. Two main types of architecture can be singled out: functional (Seeheim, Arch) and structural models (PAC proposed by J. Coutaz, and its variations, such as PAC Amadeus, MVC and its variations, such as MVC2). The functional models split an interactive system into several functional components. For ex., the Seeheim model is made up of 3 logical components (*Presentation, Dialogue Controller, Application Interface)*; the Arch model defines a functional breakdown of an interactive system into 5 components in which: both the *presentation* and *interaction* components are a

Dung Tran C., Ezzedine H. and Kolsky C. (2007).
TOWARDS A GENERIC AND CONFIGURABLE MODEL OF AN ELECTRONIC INFORMER TO ASSIST THE EVALUATION OF AGENT-BASED INTERACTIVE SYSTEMS.
In *Proceedings of the Ninth International Conference on Enterprise Information Systems - HCI*, pages 290-293
DOI: 10.5220/0002377702900293

decomposition of the presentation of the Seeheim model, *the functional kernel* component, the *domain adapter* component and the *dialogue controller* component. Seeheim and Arch provide canonical functional structures with big grain; they are useful as a structural framework for a design or a rough analysis of the functional decomposition of an interactive system (Trabelsi et al., 2004). These decompositions are generally not enough to complex applications; the functionalities are mixed in the too macroscopic components (Tarpin-Bernard and David 1999). The structural models aim at a finer breakdown by using structural components, and in particular those said to be distributed or agent approaches, suggest grouping the functions together into one unit, the agent. The agents of this type of architecture are then organized in a hierarchical manner according to principles of composition or communication. For example, a MVC agent is made up of three facets: Model, View and the Controller. The PAC model defines an agent using three facets: the Presentation, the Abstraction and the Control.

These architecture models recommend the same principles, based on the separation between the functional core of system (application) and the human-machine interface. The architecture therefore has to define a distribution of the interface services and to define an exchange protocol (Hilbert and Redmiles, 2000). This separation makes modifications easier; it allows modifying the interfaces without affecting the application. In the next part, we present an agent oriented architecture.

3 AN AGENT ORIENTED ARCHITECTURE

Our approach is intended to be mixed as its principles borrow from both types of model; it is both functional and structural. In our architecture (Grislin-Le Strugeon et al., 2001; Ezzedine et al., 2003), we suggest using a division into 3 functional components recommended in the Seeheim model which we have called respectively: interface with the application (connected to the application), dialogue controller, and interface or presentation (this component is directly linked to the user). Each of these components can be broken down further in a structural approach in the form of agents. These components are built like three multi-agent systems and they are considered as working in parallel, at least, at a theoretical point of view.

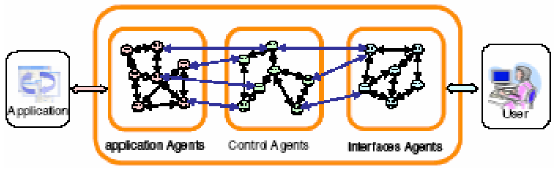

Figure 1: Agent oriented architecture.

The application agents, manipulating the field concepts of the application, cannot be directly accessed by the user. One of their roles is to ensure the real time transmission of the information necessary for the other agents to perform their task. The interface agents are in direct contact with the user (they can be seen by the user). These agents co-ordinate between themselves in order to intercept the user commands and to form a presentation which allows the user to gain an overall understanding of the current state of the application. The control agents in the Dialogue Controller component provide services for both the application and the interface agents in order to guarantee coherency in the exchanges emanating from the application towards the user, and vice versa. Their role, in particular, is to link the two other components together by distributing the user commands to the application *agents* concerned, and by distributing the application feedback towards the interface agents concerned. All these agents communicate amongst themselves in order to answer the user actions. This communication can be considered as being services among agents and can be represented using high level Petri Nets (PN) (Ezzedine et al., 2006).

4 PROPOSITION

An "electronic informer" is a software tool that ensures the automatic collection, in a real situation, of users' actions and their repercussions on the system. The collection of information is done in a discreet and transparent way for the user, who must not at any time feel hampered by the presence of the informer. This is an advantage of such a tool. Objective data collected through interactions can be processed, analyzed and shown in a synthetic shape to the evaluator. This facilitates the analysis of the results. The reader can find in (Hilbert and Redmile, 2000) state of the art concerning tools of this type.

We propose a generic and configurable model of an "electronic informer" which assists the evaluator in analysing and evaluating agent-oriented interactive systems. Such tool takes into account the specific architecture of these systems. It also proposes explicitly a coupling between the architecture and evaluation phase. At the present time, the first version of an electronic informer has been studied and developed (Trabelsi, 2006). However, it is not a generic tool but only a specific tool to evaluate a specific agent-oriented applicative system that is intended to supervise the passenger information on a public transport system. It cannot be used to evaluate other agent-oriented systems because it depends on the number of agents, the structure and the contents of such systems. Furthermore, it shows some inconveniences and shortcomings. We solve such problems by a generic and configurable model of an "electronic informer". It is made up of 7 main modules (Fig. 2).

Module 1 (M1): collecting events in user interface and service level from all agents and users of the concerned interactive system.

M2: associating events in intermediate level (user interface and service events) with each application task. Several events in intermediate level can be realized to obtain a certain application task. For ex., 3 user interface events *TabDriver_click*, *TextBoxMessage_OnChange* and *buttonOK_Click* and 2 services with the same name "Send a message to the driver" of the agent interface Vehicule and of the agent application Vehicule associated with the application task "Send a message to the driver" of a system intended to supervise the passenger information on a public transport system.

M3: processing collected data of a chosen agent in a certain period of time and showing results in comprehensible forms. Here are examples of calculations and statistics: response time for interactions between services; time for a certain user interface event (time for loading an interface agent or for typing an text box…); time for completing a service and furthermore, an application task; time for consulting help or unproductive time (help time + snag time + search time (Bevan and Macleod, 1994; Chang and Dillon, 2006)) that user takes to complete a certain application task, the percentage of services accomplished and furthermore, of application tasks accomplished, the error's percentage, the help's use frequency, the percentage of services and furthermore, of application tasks achieved per unit of time, the ration of failure or success for each interaction between services, the ration of appearance of each user interface event of a certain interface agent, the percentage of use for each service of a certain agent, the average number of user interface events per unit of time, and so on.

M4: generating the Petri Nets (PN) to describe activity process of agents and users in the system from collected data and BSA (Specification Base of Agents). Indeed, it describes process of interactions between services of different agents as well as process of activity of user to complete application tasks. We call them "observed" PN. Generating PN facilitates evaluators because it provides them with the visual views of all the activities of the user and the concerned system.

M5: comparing observed PN created above with the PN that system designer has intended before to complete application tasks. This comparison assists the evaluators in detecting use errors; for ex., the evaluator can perceive that the user has passed redundant state, has realized useless manipulations or takes more time than the one predicted by designer to complete an application task. M5 can also be used to assist the evaluator in comparing the ability of different users to use a system.

M6: using results of processes from M3, the PN generated by M4, the comparison of two PN from M5 and usability characteristics as well as ergonomic criteria as a basis, M6 is responsible of assisting the evaluator in criticising concerned system and advising the designer to improve it. Although the term "usability" has not been defined homogeneously, it exists several definitions (Dix et al., 1993; Nielsen, 1993; ISO/IEC 9126-1); in general, it refers to a set of multiple concepts, such as execution time, performance, user satisfaction and ease of learning ("learnability"), effectiveness, efficiency, taken together (Abran et al., 2003). There are also several sources from different authors and organisations. M6 assists the evaluator in evaluating concerned system on the basis of criteria from several different sources such as the ergonomic criteria of (Bastien and Scapin, 1995), the quality attributes of (Lee and Hwang, 2004) and the characteristics of the consolidated usability model of (Abran et al., 2003). The results of processes (calculations and statistics) from M3 provide the necessary measures for the evaluation of these ergonomic criteria, quality attributes and characteristics. M6 is not yet realized.

M7: configuring electronic members to evaluate different agent-oriented systems. It allows entering the BSA (Specification Base of Agents) that describes the evaluated system, the PN that system designer has intended and some configuration parameters of evaluated system.

5 CONCLUSION - PERSPECTIVE

We have presented a brief state of the art concerning interactive system architectures, and proposed a mixed architecture as well as a generic and configurable model for assisting the evaluation for

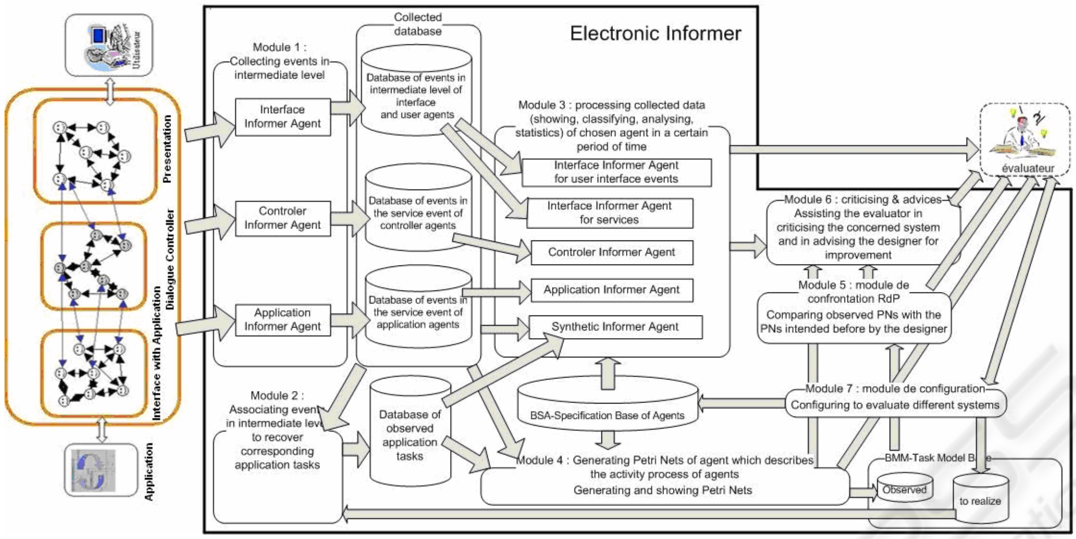

Figure 2: A Generic and Configurable Model.

agent-oriented interactive systems. We intend to combine this "electronic informer" method with other methods (questionnaire, interview…). It needs to combine data collected from "the electronic informer" and data collected from the other methods to evaluate more efficiently such systems.

ACKNOWLEDGEMENTS

The present research work has been partially supported by the "Ministère de l'Education Nationale, de la Recherche et de la Technologie», the «Région Nord Pas-de-Calais», the FEDER (MIAOU, EUCUE, SART), the ANR ADEME (Viatic.Mobilité), the PREDIM (MouverPerso).

REFERENCES

Abran, A., Khelifi, A., Suryn, W., Seffah, A., 23–25April 2003. "Consolidating the ISO Usability Models". In: *Proc. of 11th Int. Software Quality Management Conference,* Springer, Glasgow, Scotland, UK.

Bastien J.M.C., Scapin D.L., 1995. Evaluating a user interface with ergonomic criteria, In: *Int. Journal of Human-Computer Interaction,* vol. 7, p. 105-121.

Bevan, N., Macleod, M., 1994. Usability measurement in context. In: *Behav.Inf. Technol.,* vol. 13, no. 1/2, pp. 132–145.

Chang, E., Dillon, T., March 2006. A Usability-Evaluation Metric Based on a Soft-Computing Approach. In: IEEE Trans. On SMC—Part A: Systems and Humans, Vol. 36, N. 2.

Ezzedine, H., Trabelsi, A., Kolski, C., 2003. Modelling of Agent oriented Interaction using Petri Nets, application to HMI design for transport system supervision. In: *Proceedings CESA*, Lille, France.

Ezzedine H., Trabelsi A., Kolski C., 2006. Modelling of Agent oriented Interaction using Petri Nets, application to HMI design for transport system supervision. In: *Mathematics and computers in simulation,* 70, pp. 358-376.

Grislin-Le Strugeon, E., Adam, E., Kolski, C., 2001. Agents intelligents en interaction homme-machine dans les systèmes d'information. In: Kolski C. (Ed.), *Environnements évolués et évaluation de l'IHM,* pp. 207-248. Paris: Éditions Hermes.

Hilbert D.M., Redmiles D.F., 2000. Extracting usability information from user interface events. In: *ACM Computing Surveys*, 32 (4), pp. 384-421.

Lee, S.K., Hwang, C.S., 2004. Architecture modelling and evaluation for design of agent-based system. In: *The Journal of Systems and Softwares,* 72, 195-208

Nielsen, J., 1993. *Usability Engineering*, Academic Press, Boston.

Shneiderman, B., 1998. *Designing the user interface : strategies for effective human-computer interaction.* Addison-Wesley.

Tarpin-Bernard, F., David, B., Mai 1999. AMF : un modèle d'architecture multi-agents multi-facettes, In: *TSI,* Vol. 18, No. 5, 555-586.

Trabelsi, A., Ezzedine, H., Kolski,. C., October 2004. Architecture modelling and evaluation of agent-based interactive systems. In: *Proc. IEEE SMC 2004, The Hague, pp. 5159-5164*.

Trabelsi, A., 2006. Contribution à l'évaluation des systèmes interactifs orientés agents. Application à un poste de transport urbain. Ph.D. Thesis, University of Valenciennes, France.