Get Involved in an Interactive Virtual Tour of Brest Harbour: Follow the Guide and Participate

Mukesh Barange[1], Pierre De Loor[1], Vincent Louis[2], Ronan Querrec[1], Julien Soler[1], Thanh-Hai Trinh[1], Éric Maisel[1], and Pierre Chevaillier[1]

[1] ENIB–UEB; LISYC: Laboratory of Computer Science for Complex Systems, France
[2] Dialonics, Lannion, France

Abstract. Recent cultural heritage applications have been based on rich-content virtual environment (VE), in which virtual humans can communicate with visitors and other agents using natural language (NL). The conceptualisation of these dialogues are dependent on the contents of the application. Hence, we propose to use the semantic modelling of the VE and the activities of agents for the conceptualisation of the dialogue. Meta level semantic information are used as arguments in NLU/NLG rules. The advantage of this approach is that the dialogue rules are independent from the contents of the application and have clear semantics. We applied these principles to develop Brest'Coz, an interactive virtual tour for the learning of shipbuilding techniques used in France in early 18^{th} century.

Keywords: Semantic Modelling, Dialogue Management, Cultural Heritage.

1 Introduction

This study takes place in a general perspective to make the development of rich-content virtual reality (VR) applications more rational, and we address more specifically issues related to the design of conversational agents. As proposed by many authors [1,8], one promising approach is to center the architecture on an abstract semantic layer. The main motivations are as follows: (1) The design of the VE and that of agents, should be independent. That means that the communicative capabilities of an agent, such as a guide in an interactive virtual tour (or an educational agent), should be independent to the environment it is supposed to act in. (2) The semantic model of the environment, both physical and social, can be used as a source of knowledge for agents to make decisions and to support dialogues [4].

This article presents how these principles have been successfully used to develop Brest'Coz, an interactive virtual tour dedicated to the learning of techniques used for shipbuilding in France in early 18^{th} century. The learning is based on a gaming approach. Users receive directions from a virtual guide and have to communicate with non-playing characters to get information and participate to some collaborative activities.

H. Högni Vilhjálmsson et al. (Eds.): IVA 2011, LNAI 6895, pp. 93–99, 2011.
© Springer-Verlag Berlin Heidelberg 2011

Providing NL dialogue capabilities to conversational agents using semantic modelling, rises many scientific issues. One of the issues is *what the agent can say?* In Brest'Coz, agents should be able to describe the structure of the environment, i.e., properties of objects, their behaviors and their spatial relationships. Moreover, the visitor may ask agents to describe their own activities or the activities of other agents. Also, agents can be involved in some collaborative activities. Thus, while communicating with other agents and visitors, agents should take into account the social norms, including their roles and organisational rules.

Another issue is *how does the agent generate the NL dialogue structure?* In many applications [7,11], the dialogue management is based on dialogue acts. However, this approach is highly dependent on contents of the application, and requires extra efforts for the annotation of dialogue acts [3]. In contrast, the rule–based dialogue management approach, like in [13] operates on a hard–coded set of rules, using pattern matching and substitution, to generate the response. The fundamental bottleneck of this approach is that the knowledge is explicitly presented in the form of hard–coded values in dialogue rules. All these approaches do not take into account the semantics of the VE and therefore, are not suitable for the modelling of dialogue behaviors, independent from the application.

To develop Brest'Coz, we used Mascaret framework in which we embedded NabuTalk, a commercial rule-based dialogue engine. We defined the generic queries agents can perform on the semantic model in order to interpret, and to generate utterances. Because these NLU/NLG rules are defined at an abstract level (meta-model) they are independent of the content of the application and have clear semantics.

In Sect. 2, we present the Brest'Coz application. Sect. 3 gives main elements of the semantic layer used as linguistic resources for conversational agents. Sect. 4 illustrates how these semantic information are used in NLU/NLG rules.

2 The Brest'Coz Application

Brest'Coz is an application for interactively visiting the harbour of Brest, France. In early 18^{th} century, it was an important site for the French navy where various specific shipbuilding techniques were used. It is a task-oriented tour. At the beginning, a virtual guide gives some directions about the goal assigned to the user, which is represented by a human-like avatar. The user has to take part to the transshipment of a boat, and can learn how middle-age wheeled cranes were operated for that. To get involved in this collaborative activity (supported by autonomous agents), the user has to communicate with different characters and to interact with the environment using VR peripherals.

The 3D modelling encompassed the docks, some noticeable buildings and various shipbuilding sites (Fig. 1). We simulated different shipbuilding activities (e.g., shipwrighting, transshipment), performed by different categories of workers (autonomous virtual agents).

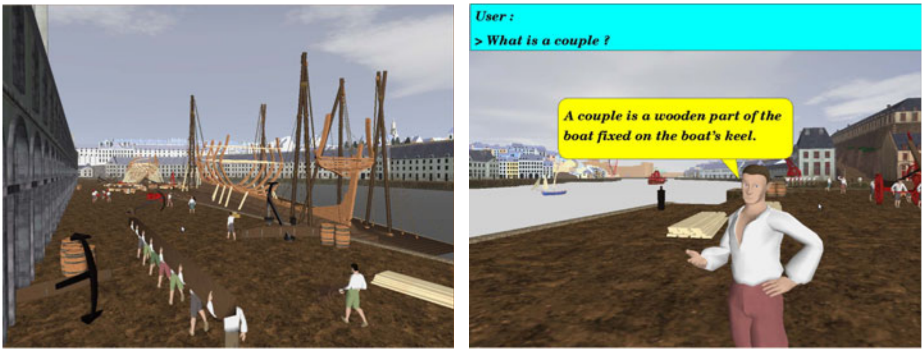

Fig. 1. Brest'Coz: overview of shipbuilding activities on the harbour (left) - example of dialogue with a worker (right)

3 Semantic Modelling Using Mascaret

MASCARET, which stands for *MultiAgent System for Collaborative, Adaptive & Realistic Environments for Training*, is a generic framework that provides necessary abstractions for both content- and system-oriented semantic modelling of VR applications [2]. It provides a modelling language expressive enough for subject matter experts to formulate their knowledge about a specific domain (content-oriented approach). The language has clear operational semantics for models to be interpreted by the execution platform (system-oriented approach). MASCARET is based on two complementary metamodels, VEHA and HAVE. These metamodels have been defined as extensions of those of UML, the Unified Modelling Language. UML is used here as a content description language and not as an abstraction for object-oriented programming.

Modelling the VE Using VEHA. This metamodel is dedicated to the modelling of entities compounding the VE, their categories, internal structures, relationships, and behaviors. Fig. 2 illustrates how cranes have been modelled in Brest'Coz (based on Worldnet[1]). A crane structurally contains one hook that can be logically linked to one container.

Fig. 2 shows the partial model of the behavior of the crane, which is supported in MASCARET by state machines. Here, the crane can grasp an object if it is stopped and when the condition canBeLoaded is satisfied. Such logical constraints are expressed using VRX-OCL [12].

Modelling Human Activities into the VE using HAVE. This metamodel supports the modelling of collaborative activities of agents in the environment, which is supposed to be described using VEHA. Agents are here the autonomous characters for whom the activity is simulated and the users. Defining the activity

[1] http://wordnetweb.princeton.edu/perl/webwn

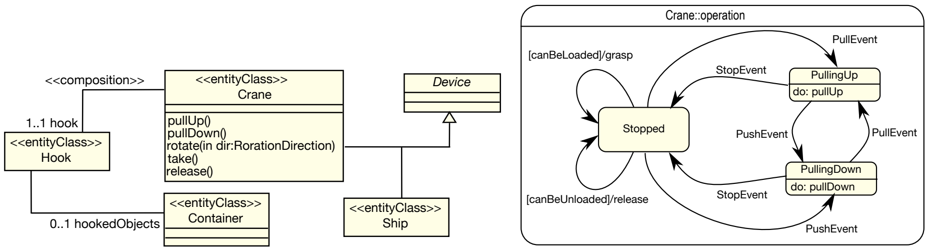

Fig. 2. Partial modelling of the Crane entity class using VEHA. Left: properties and structure; right: behavior.

means to describe how it is organised and what the agents are supposed to do in the environment. These two views are supported in HAVE respectively by the *organisational model* and the *activity model*.

As for existing multiagent models, the organisational metamodel is based on the concept of agent, role and organisational entity. It allows to define social rules that govern agents' behavior within an organisation.

Fig. 3 (left) illustrates how the organisational structure used for the boat transshipment can be instantiated: agents have been associated to roles (e.g., crane operator) and the available resources been defined. HAVE supports the modelling of the collaborative activity (i.e., actions to be performed by an organisational entity) using activity charts similar to those of UML. Each action is defined by its feasibility conditions (precondition), rational effect (post-condition), and action to be done (do: statement). Fig. 3 (right) shows how the user (visitor) can collaborate with other agents to manipulating the crane.

The way the description of the activity is interpreted by agents is defined using behave, which is a generic model of an agent architecture, independent to any domain specific application. The dependency between agents and the environment takes place at the meta level, i.e., the actual model of the environment and of the activities that agents are supposed to perform, are viewed as data available for agents to make decisions and to execute their behaviors.

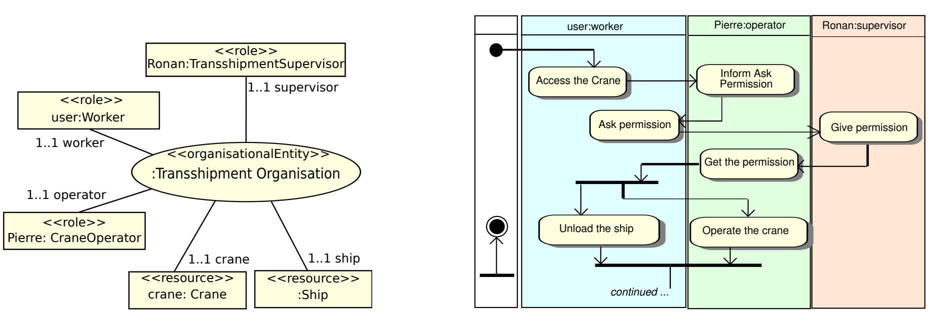

Fig. 3. Modelling of the transshipment activity using HAVE (partial)

4 Conceptualisation of Dialogues Using Semantic Modelling

In this section, we show how semantic modelling presented previously has been used to process NL utterances using generic parameterized rules, that preserve the independence of the communicative behavior of agents from the VE. We also illustrate how it has been implemented using Mascaret and NabuTalk.

NabuTalk is a NL dialogue engine that relies on the concepts of Artimis [10] and the JADE Semantics Add-on [9]. It is based on a high–level agent programming language, named Nabu, that includes appropriate mechanisms to handle NLU/NLG concepts, such as utterance templates, dialogue interpretation and generation rules. It provides a rule–based mechanism to deal with users' utterances. The pattern-matching algorithm relies not only on the matching of usual regular expressions but also on the unification of feature structures, based on the theory of functional unification grammars [6].

Both users and agents can initiate the dialogue. In Mascaret, each agent is associated with a thread of the NabuTalk engine. For each communicative function, NabuTalk selects the best appropriate pattern for the NLU rule. The patterns may contain parameters, which can be substituted with semantic information retrieved from Mascaret models. The properties of model elements depend on their type, which are defined at the meta level. Because the NLU/NLG rules rely only on meta level concepts, they are independent from specific contents of the application.

All the agents share a common set of dialogue rules. The individuality of agent's dialogues comes from the role associated to the agent. Furthermore, agents may use additional NLU/NLG rules to support application-specific dialogues.

In the transshipment scenario (Fig. 3), when the user (playing the role of worker) accesses the crane, the operator agent, named Pierre, initiates the following dialogues with the user:

- 1.1 Pierre: Hey! You do not have the permission to use the Crane.
- 2.1 user: Who can give the permission to use the Crane?
- 2.2 Pierre: The supervisor can do it. His name is Ronan.
- 3.1 user: What can I do with the Crane?
- 3.2 Pierre: You can pull-up, pull-down, rotate, take and release with the Crane.

When the user accesses the crane, the operator executes the communicative action inform(require(Ask_Permission)). The precondition of this action is to verify the physical constraint Crane.state=Available. This vrX-Ocl constraint is evaluated according to the state machine of the Crane provided by Veha (Fig. 2). If it fails, the operator generates the utterance 1.1.

The natural language rule (nlu-rule) in NabuTalk can use predicates implemented as Mascaret connectors to access meta level information. To go further in details, utterance 3.1 is interpreted as a Query-ref on the capabilities of a class (Crane is a class name). The agent looks in the Veha model and retrieves the names of all the operations of the class. The Nabu code to handle this looks like:

```
(nlu-rule {[what-can-do-with] [class($class)]} {
  let (eval $list listClassOperations($class)) {
    if (empty $list) { talk($other {"You can do nothing with a" $class})
     } else {
       talk($other {"You can" list($list "," "and") "with a" $class}) }
  } })
```

NabuTalk performs pattern matching with resources like *[what-can-do-with]*. The NabuTalk resource is a regular expression that can support different ways to express a phrase. Thus, it provides some flexibility for the user to communicate with agents. [class($class)] is a resource implemented with NabuTalk predicates and Mascaret connectors. It represents any class found in the semantic model. If such a pattern matches, the condition (eval $list listClassOperations($class)) is evaluated and if true, the $list variable will contain names of all operations of the class. The listClassOperations predicate is implemented as a Mascaret connector. The code just walks through Mascaret model to get operations of the class:

```
Class c=MascaretApplication::getModel()->getClassByName(className);
vector <Operation> op=c->getOperations(); // returns operation names
```

The result of this method is converted into a NabuTalk object that can be post–processed by the NabuTalk engine to generate the utterance 3.2.

Mascaret agents are able to talk about spatial relationships (topological and directional relationships) between entities using vrX-Ocl. Dialogues may also refer to the activity of agents, based on the information from activity charts. Thus, the agent can reply to user's questions like *What can I do now?* (the name of the next action belonging to the role of the locutor), *Why should I do it?* (the description associated to the post-condition of the referred action) or *How can I do it?* (the description of the do:statement associated to the action), etc.

Thus, using meta level information, agents can talk about the entities of the VE, their behaviors, their social organizations, and collaborative activities. This approach is particularly suitable for complex VEs where many human activities have to be simulated. In such a context, using specific NLU/NLG rules would make the drawing of the scenario far too complex. Our approach does not cover all types of dialogues. It is more suitable to handle inquiry, request–response or question–answering dialogues. A trade-off has to be reached between the genericity of dialogue rules and the ability to deal with specific application conversational styles, like in [5] which uses learning mechanisms and is based on annotated corpus.

5 Conclusion

In this article, we have presented the benefits of semantic modelling for dialogue modelling. Because our approach is anchored to a meta-modelling framework, it ensures the consistency between the model of the VE and behaviors of autonomous agents. It also makes NLU/NLG rules independent from contents of the application and provides a high expressiveness. Our solution has been used to develop an interactive cultural heritage application, Brest'Coz, using Mascaret and the NabuTalk engine. Although, we have not yet performed its formal evaluation, the application has been successfully presented during a video game exhibition at Brest, and feedbacks from users were very positive. Our long term goal is to enrich agents' behaviors, so that they can engage themselves into dialogical interactions to coordinate their collaborative activities. One can expect to significantly lower the complexity of the description of the activity.

Acknowledgments. This work was supported in part by a grant from Région Bretagne and in part by the ANR (Corvette project ANR-10-CORD-012).

References

1. Bogdanovych, A., Rodriguez, J.A., Simoff, S., Cohen, A.: Virtual agents and 3D virtual worlds for preserving and simulating cultures. In: Ruttkay, Z., Kipp, M., Nijholt, A., Vilhjálmsson, H.H. (eds.) IVA 2009. LNCS, vol. 5773, pp. 257–271. Springer, Heidelberg (2009)
2. Chevaillier, P., Trinh, T.-H., Barange, M., Devillers, F., Soler, J., Loor, P.D., Querrec, R.: Semantic modelling of virtual environments using Mascaret. In: Proceedings of SEARIS 2011, Singapore (March 2011)
3. Gandhe, S., Traum, D.: Creating spoken dialogue characters from corpora without annotations. In: Proceedings of Interspeech 2007, Antwerp, Belgium (2007)
4. Ijaz, K., Bogdanovych, A., Simoff, S.: Enhancing the believability of embodied conversational agents through environment-, self- and interaction-awareness. In: Proceedings of ACSC 2011, Perth, Australia (2011)
5. Jan, D., Roque, A., Leuski, A., Morie, J., Traum, D.: A virtual tour guide for virtual worlds. In: Ruttkay, Z., Kipp, M., Nijholt, A., Vilhjálmsson, H.H. (eds.) IVA 2009. LNCS, vol. 5773, pp. 372–378. Springer, Heidelberg (2009)
6. Kay, M.: Functional unification grammar: A formalism for machine translation. In: Proceedings of COLING-ACL, pp. 75–78 (1984)
7. Kopp, S., Gesellensetter, L., Krämer, N.C., Wachsmuth, I.: A conversational agent as museum guide – design and evaluation of a real-world application. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, pp. 329–343. Springer, Heidelberg (2005)
8. Latoschik, M.E., Biermann, P., Wachsmuth, I.: Knowledge in the loop: Semantics representation for multimodal simulative environments. In: Butz, A., Fisher, B., Krüger, A., Olivier, P. (eds.) SG 2005. LNCS, vol. 3638, pp. 25–39. Springer, Heidelberg (2005)
9. Louis, V., Martinez, T.: Jade semantics framework. In: Bellifemine, F., Caire, G., Greenwood, D. (eds.) Developing Multi-agent Systems With Jade, ch. 12, Wiley & Sons, Chichester (2007)
10. Sadek, D.: Artimis rational dialogue agent technology: An overview. In: Weiss, G., Bordini, R., Dastani, M., Dix, J., Fallah Seghrouchni, A. (eds.) Multi-Agent Programming, Multiagent Systems, Artificial Societies, and Simulated Organizations, vol. 15, pp. 217–243. Springer US, Heidelberg (2005)
11. Traum, D.R.: Talking to virtual humans: Dialogue models and methodologies for embodied conversational agents. In: Wachsmuth, I., Knoblich, G. (eds.) ZiF Research Group International Workshop. LNCS (LNAI), vol. 4930, pp. 296–309. Springer, Heidelberg (2008)
12. Trinh, T.H., Querrec, R., De Loor, P., Pierre, C.: Ensuring semantic spatial constraints in virtual environments using UML/OCL. In: Proceedings of VRST 2010, pp. 219–226 (2010)
13. Wallace, R.S.: Be Your Own Botmaster. ALICE A.I. Foundation Inc. (2003)