

Staging Choreographies for Team Training in Multiple Virtual Worlds Based on Ontologies and Alignments

Emanuel Silva^{1,2}, Nuno Silva², and Leonel Morgado³

¹University of Trás-os-Montes e Alto Douro, Vila Real, Portugal

²School of Engineering, Polytechnic of Porto, Porto, Portugal
{ecs,nps}@isep.ipp.pt

³INESC TEC (formerly INESC Porto) / Universidade Aberta, Lisbon, Portugal
leonel.morgado@uab.pt

Abstract. In this paper we present an approach that makes possible the staging of choreographies for education and training purposes in potentially any virtual world platform. A choreography is seen here as the description of a set of actions that must or may be executed by a group of participants, including the goals to be achieved and any restrictions that may exist. We present a system-architecture and the formalization of a set of processes that are able to transform a choreography from a platform-independent representation into a specific virtual world platform's representation. We adopt an ontology-based approach with distinct levels of abstraction for capturing and representing multi-actors and multi-domain choreographies to be staged in virtual world platforms with distinct characteristics. Ontologies are characterized according to two complementary dimensions – choreography's domain (independent and dependent) and virtual world platform (independent and dependent) – giving rise to four ontologies. Ontology mappings between these ontologies enable the automatic generation of a choreography for virtually any target virtual world platform, thus reducing the time and effort of the choreography development.

Keywords: virtual worlds, training, choreography, multi-user, model-driven, ontology, mapping.

1 Introduction

Virtual worlds have achieved significant levels of interest for supporting teaching and learning activities [1], [2] since they provide the creation of immersive environments where multiple elements of a team sharing a common virtual space can develop competencies in a simulated context [3], [4]. Choreographies of virtual actors are a specific type of content that represent the set of actions that can be performed simultaneously by human-users and virtual computer-controlled actors thus enabling human trainees/students to play roles as part of teams or within a simulated social context. In this sense, a choreography is the description of a set of actions that must or may be executed by a group of participants, including the goals to be achieved and any restrictions that may exist.

Because designing a choreography is a resource-intensive effort, it would be desirable for the result not to be hostage to a specific virtual world platform (VWP) but rather deployable in any VWP. However, as virtual platforms are very heterogeneous in terms of (e.g.) functionalities, data models, execution engines and programming/scripting languages or APIs, deploying a platform-based choreography into another VWP is difficult and time-consuming [5]–[8].

We believe that the approach presented in this paper provides a contribution that facilitates the development, sharing and adaptation of choreographies aimed to be staged in different virtual platforms. For this, we suggest an approach where the conceptual representation model of the choreography is captured in the form of ontologies, and its adaptation to a particular virtual world follows a set of models transformation processes, similar to that suggested by the Model Driven Architecture (MDA) paradigm [9]. The proposed ontology-based definition of choreographies can capture not only the physical aspects as objects but more complex content such as procedures, consisting of sets of actions and conditions in which the actors can perform them.

Thus, this paper presents an approach that deals with the design and representation of platform-independent multi-user choreographies, and their deployment to different VWPs with minimal effort and time using a set of transformation processes based on ontologies and alignments. The rest of the paper comprehends four more sections. In section 2 we present the proposed approach and the description of the system architecture. Section 3 describes the developed experiments. Section 4 compares the related work with the proposed ideas. Finally, Section 5 summarizes the proposal and suggests future directions.

2 Proposed Approach

To deal with VWP with different characteristics, we argue that choreographies should be clearly separated from the technical characteristics of the execution in the VWP.

To this end, the core of the proposal is a “generic high-level ontology” that captures the choreography in a conceptual and abstract fashion, so it is independent from the staging/deployment VWP. Thus, the data model of every virtual world must be captured/represented by the so-called “platform-specific ontology”, and a mapping between the generic high-level ontology and the platform-specific ontology must be defined. The mapping will provide the means to transform the platform-independent choreography into a platform-dependent choreography.

To address this the MDA software-development paradigm [9] is adopted and adapted. MDA specifies three default models of a system corresponding to different layers of abstraction: a Computation Independent Model (CIM) the most abstract, which represents the view of the system without any computational complexities; a Platform Independent Model (PIM) that describes the behavior and structure of the system, but without technological details, and a Platform Specific Model (PSM) that combines the specifications in PIM with the specific details of a specific platform.

Based on the concept of model independence and model transformation of MDA, we adopt an approach based on two first-class citizen dimensions: the VWP

dimension and the choreography’s domain dimension. In fact, unlike in MDA, in our approach the model is not only independent from the VWP but also independent from the (choreography’s) domain. Figure 1 depicts the MDA one-dimensional approach (Fig.1 a) in comparison with the two-dimensional envisaged approach Fig. 2 b).

The nomenclature O_{PxDx} refers to Ontology, Platform and Domain, with “x” assuming “d” and “i” values for “dependent” and “independent”, respectively. E.g. O_{PiDd} stands for “Ontology, Platform-independent, Domain-dependent”.

Taking into account the characteristics of ontologies, and considering they are the best way to represent conceptual information in order to bring the intelligent systems

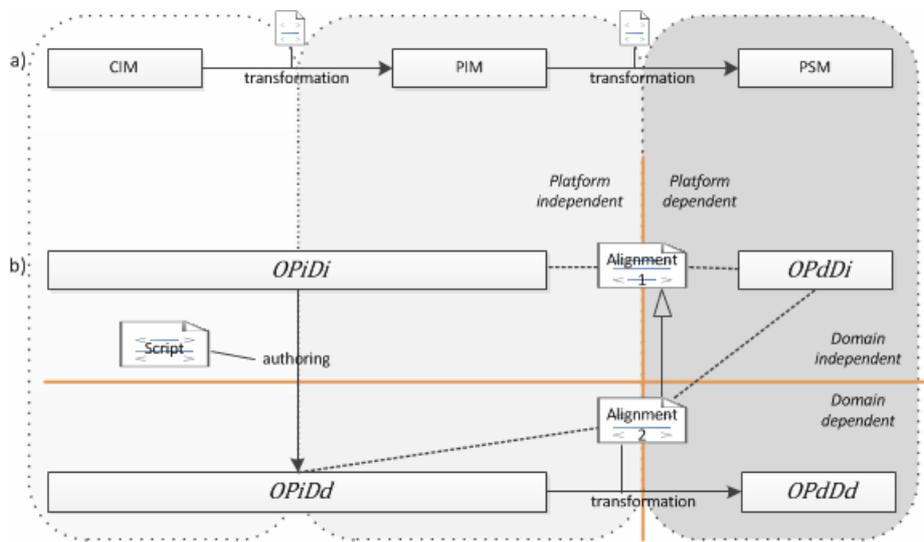


Fig. 1. Model-Driven representation according to: a) MDA and b) Our approach

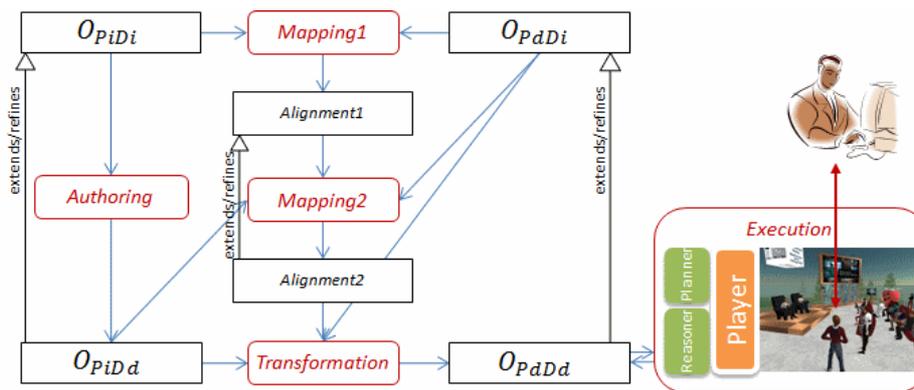


Fig. 2. The system architecture with the processes of authoring, mappings, transformation and execution

closer to the human conceptual level [10], we claim that ontologies are then the adequate knowledge representation model for bridging the gap between the human requirements and the computational requirements [10]–[13], thus able to play the role of both Computation-Independent Model (CIM) and Platform-Independent Model (PIM). Following MDA, the ontology-based choreography representations are successively transformed through a set of processes until the final, platform-specific choreography (O_{PDD}) that is executed in the specific VWP. The proposed architecture is depicted in figure 2, and comprehends four ontologies and five processes.

2.1 Ontologies

The following four ontologies are representational models of the different choreography abstractions:

- O_{PiDi} (platform-independent and domain-independent) is the generic high-level ontology representing the core concepts of a choreography independent of any implementation environment, also designated as the foundational ontology. Figure 3 presents its current status, whose motivations and design decisions have been described in a previous paper [14].
- O_{PDDi} (platform-dependent and domain-independent) represents the core concepts of a choreography for a specific VWP. Despite this is a platform-dependent ontology, it remains independent from any application domain, and therefore is developed only once (eventually requiring adaptations in face of mutations in the VWP due to any evolution). Each virtual world can have their own interpretation of a choreography describing the concepts in a private way in order to best fit its characteristics, but this ontology must capture and represent every concept corresponding to those defined in the foundational ontology to capture the same semantic knowledge, so that is possible to establish semantic relations between them. Thus, we consider that in order to apply this approach, it is necessary to develop, for each target VWP, an ontology to represent that virtual world platform’s particular interpretation of the fundamental ontology. Moreover, this ontology can incorporate additional concepts considering the specific characteristics of that virtual world and using its own terminology.

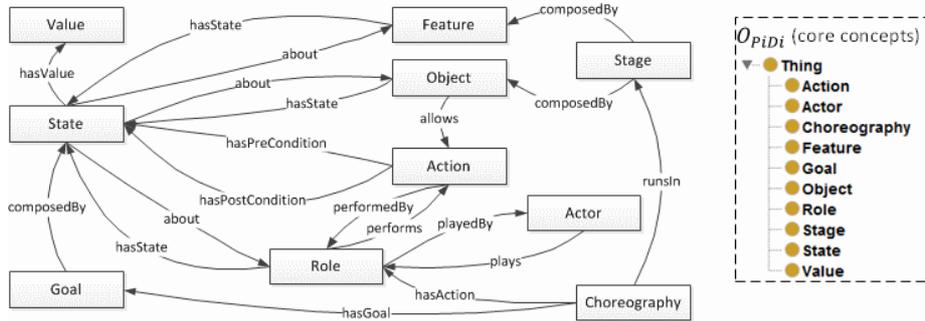


Fig. 3. The Foundational Choreography Ontology (O_{PiDi}): representation of concepts, properties and relations between concepts

- O_{PiDd} (platform independent and domain dependent) is a choreography resulting from the authorship of O_{PiDi} . It captures the representation of a complete choreography for a specific application domain, without any concern about the technical specificities of any platform.
- O_{PdDd} (platform dependent and domain dependent) is a choreography represented in/for a specific VWP. This is the choreography that is intended to serve as a reference for the staging in the virtual world.

2.2 Processes

In the proposed architecture we apply five processes to conduct a successive transformation of models representing the various abstractions of a choreography to adapt its specification to a particular virtual world.

To illustrate the explanation, we will consider a generic Instant Messaging platform for which a choreography will be adapted. This is a very simple platform (developed by the authors for testing purposes, cf. Fig. 7) where there is only the representation of an action called write.

Authoring. Authoring is a user-based process in which the choreographer authors a domain dependent choreography in the form of an ontology. This process is typically performed by an expert that manually builds the ontology. But end-user tools can also be developed to allow people without knowledge or training in ontologies to specify the choreography through simple and intuitive interfaces. That is, end-user tools can in a transparent manner, build ontologies instead of human users directly, facilitating the authoring process.

The foundation ontology (O_{PiDi}) is extended and refined semantically to describe the choreography entities specific to an application domain, giving rise to O_{PiDd} . The authoring process must ensure the set of changes applied does not change the semantics defined in O_{PiDi} . For that, the following assumptions must be guaranteed:

- No axioms defined in O_{PiDi} can be removed;
- No contradictions can be added, i.e., O_{PiDd} must be logically consistent;
- No new root elements are allowed. I.e. new entities (classes and properties) are defined as sub entities of those existing in O_{PiDi} and should not create new root elements.

Figure 4 depicts an excerpt of the ontology resulting from authorship (O_{PiDd}), and more formally its representation using Description Logics (DL) syntax.

New concepts are added as well as restrictions that will define boundaries to the ability of actions execution by the actors. The restrictions are based on the definition of associations between roles (that actors can play) and actions. Commonly, the following two types of restrictions are defined:

1. To constrain the actors allowed to perform an action based on the user's role, i.e. to perform an action, the actor must have a specific role previously defined. The relation between the concepts referenced by the action and role is defined by the property *performedBy*;

2. To assign the role(s) of an actor based on the action(s) s/he performs. Thus, the roles are dynamically assigned during the choreography according to the actions performed by the actor. This association can be seen in a dual perspective. Using a small example: on the one hand, if an actor plays the role Role1 and performs actions Action1 and Action2, s/he shall automatically plays the role Role2 thereafter. On the other hand, to play the role Role2 it is a necessary condition that the actor plays the role Role1 and performs the actions Action1 and Action2.

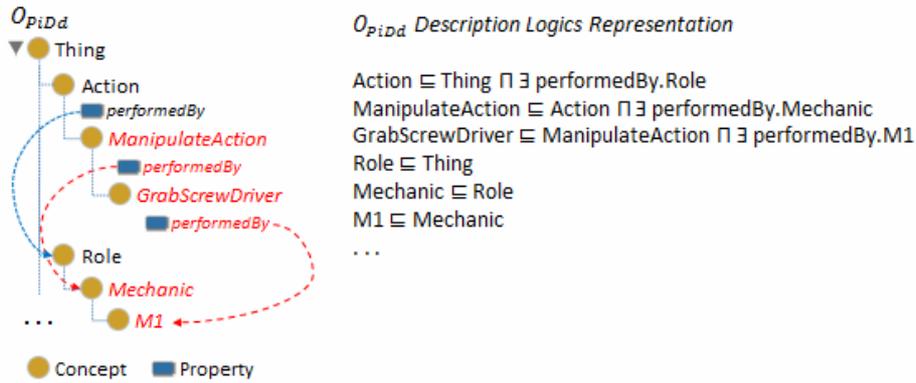


Fig. 4. An excerpt of the (O_{pIdi}) Ontology resulting from the authoring process

During the authoring process, the author can take advantage of all the semantic expressiveness of the DL language to elaborate more complex choreographies. For example, if an actor during the choreography cannot play two roles, one can specify using DL that these two roles are disjoint.

Mapping1. Mapping1 is the process that establishes correspondences between the representations of the two choreographies abstractions represented by O_{pIdi} and O_{pAdi} (Alignment1).

An alignment is a set of correspondences (semantic bridges in this paper). A semantic bridge is a set of information elements describing what entities from both source and target ontology are semantically related, and what conditions must hold in order to be considered [15]. Two types of semantic bridges are considered:

1. Concept Bridge (CB), used to describe the semantic relations between (source and target) concepts;
2. Property Bridge (PB), used to specify the semantic relations between (source and target) properties, either relations or attributes.

In this process all the core concepts (concepts that are direct subclasses of the Thing concept, depicted in Fig. 3) of the foundational ontology (O_{pIdi}) should be mapped to concepts of the platform specific ontology (O_{pAdi}) to ensure that there is full correspondences between both ontologies. Correspondences between properties are defined between properties of two mapped concepts. In order to facilitate understanding,

a relation is established between the concept-concept correspondence and the property-property correspondence (Fig. 5).

Mapping2. Mapping2 is the process that establishes correspondences between the domain choreography (O_{PiDd}) and a VWP ontology (O_{PdDi}), i.e. Alignment2. Alignment2 profits from (and extends) Alignment1, thus promoting reuse and reducing efforts (Fig. 5).

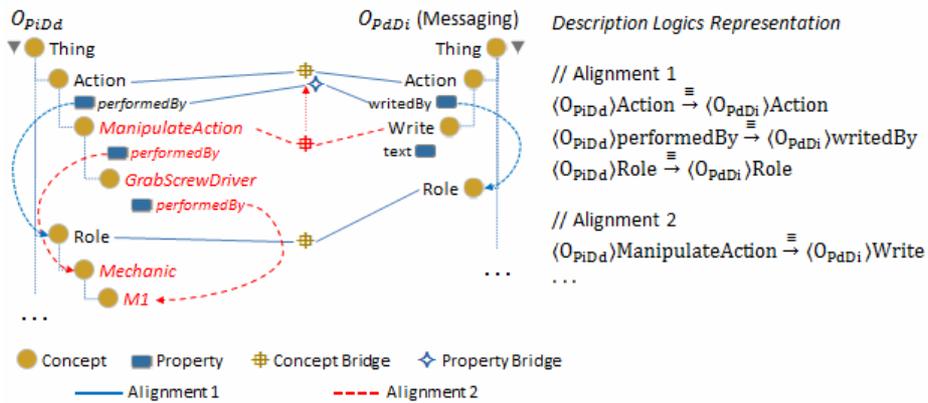


Fig. 5. A partial view of the alignments resulting from Mapping1 and Mapping2

Transformation. Transformation is the process that creates the VWP choreography (O_{PdDd}) from O_{PiDd} and O_{PdDi} and according to the Alignment2.

This is a fully automatic process that “copies” the O_{PiDd} classes and constraints (Fig. 4) to the O_{PdDd} (Fig. 6).

Despite this being an automatic process, choreographers can intervene and edit the resulting O_{PdDd} ontology for additional adjustments. Thus, the O_{PdDd} ontology may be further finely tuned to better fit the implementation platform.

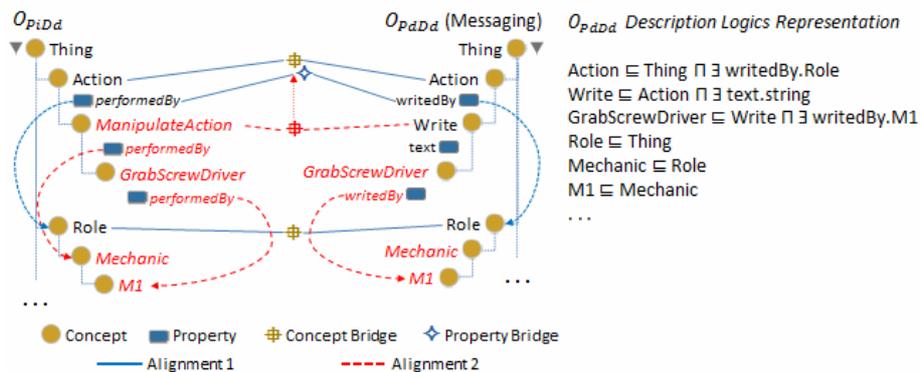


Fig. 6. An extract of O_{PdDd} ontology

Execution. Execution is the process that stages the choreography in a VWP through a Player (computer program) compatible with the VWP who has the ability to schedule actions according to the choreography and control its execution by virtual characters. Further, it monitors the human-user performance by comparing the executed actions with those described by the choreography, and reacts accordingly. This process uses a reasoner mechanism to evaluate whether it is possible to perform the actions, verifying if all necessary conditions are met. When virtual-users are present, a planner is used to calculate a plan of actions for them.

3 Experiments

For the evaluation of our approach we deployed several real-world choreographies that were staged in two different multiuser platforms with very distinct characteristics with human-users only. We used the VWP OpenSimulator¹ (OpenSim) to create a realistic multiuser 3D environment; as a counterpart system, we developed for testing purposes the aforementioned messaging platform. It is a simplified version of text-based virtual worlds of the Multi-User Dungeons era, following Morgado's definition [16]. This messaging platform has very different characteristics from the OpenSim, since it does not allow the representation of scene objects, but enables the development of a team's choreography nonetheless. Its interface provides the actions of the choreography in the form of buttons, the interaction is done by pressing buttons, and when an action is performed successfully by each team member, it is communicated to all other team members by means of a text log (Fig. 7).

Authoring is obviously the most time-consuming and creative process, while semi-automatic Mapping1 and Mapping2 processes require reduced time and effort. Once these processes are done, the transformation and execution processes are fully automatic.

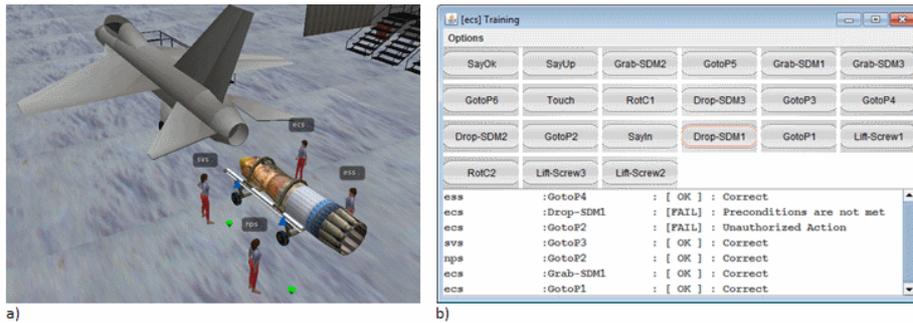


Fig. 7. Staging the choreography in a) OpenSim and b) messaging platform

¹ opensimulator.org/

4 Related Work

There is prior relevant related work addressing the description of plans to represent training procedures to be staged by a single actor as well as by teams with several elements and how actions are distributed among them. But, most approaches design a choreography aiming it to be staged on a particular VWP. This creates strong dependencies with this VWP, making it difficult or even impossible to apply to other virtual worlds. Thus, related work can be categorized according to three dimensions: modeling independence, VWP independence and number and type of the actors.

Some approaches use separate models to represent the specification of procedures and scene [7], [17], [18]. They address team training scenarios but they are strongly dependent on the characteristics of the VWP. Some other approaches attempt to bridge the gap between the representation of procedures and its execution in distinct VWP. However, such approaches are only focused on a single user not allowing the representation of teamwork [19]–[21].

Instead, our approach is capable of representing teamwork choreographies involving multi-users played either by human and virtual-characters. Also, the actions and scene are captured conceptually using a unique choreography model that is converted to potentially any VWP.

5 Conclusions and Future Work

In this paper we propose an approach that allows the development of choreographies and its adaptation and staging in potentially any VWP. For that, based on the concept of MDA and the assumption that the use of ontologies is the best way to represent the conceptual information to approximate the intelligent systems to the human conceptual level, we propose an ontology to capture the semantics of a generic choreography independent of any application domain and VWP. Further, for each VWP is adopted an ontology representing its specific terminology and functionalities, and is mapped with the generic one.

Using a set of alignments between the ontologies we describe a complete sequence of processes that allow adapting a choreography of a specific domain (but independent of any VWP) into a choreography suitable and capable of being staged into a specific virtual world. We also describe the execution process that monitors, manages the staging of the choreography, and uses reasoning engines to aid in the evaluation and validation of actions.

Moreover, ontologies allow the integration in the same model all the modeling information related to the choreography, i.e. the definition of procedures related to teamwork and the information about the scene.

Using alignments between ontologies enables the automation of adaptation of the generic ontology to the specific target ontology, hence contributing to reduce development time and resources.

In future work the Mapping1 and Mapping2 processes can be refined to incorporate automatic matching mechanisms. So, it would be possible to increase the ability to automate these processes while at the same time it reduces the need for user intervention.

Acknowledgments. This work is supported by FEDER Funds through the “Programa Operacional Factores de Competitividade - COMPETE” program and by National Funds through FCT “Fundação para a Ciência e Tecnologia” under the project AAL4ALL (QREN13852).

References

1. De Freitas, S.: Serious virtual worlds, Scoping Guide JISC E-Learn. Programme Jt. Inf. Syst. Comm. JISC UK (2008)
2. Morgado, L., Varajão, J., Coelho, D., Rodrigues, C., Sancin, C., Castello, V.: The attributes and advantages of virtual worlds for real world training. *J. Virtual Worlds Educ.* 1(1) (2010)
3. Kapahnke, P., Liedtke, P., Nesbigall, S., Warwas, S., Klusch, M.: ISReal: An Open Platform for Semantic-Based 3D Simulations in the 3D Internet. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 161–176. Springer, Heidelberg (2010)
4. Pinheiro, A., Fernandes, P., Maia, A., Cruz, G., Pedrosa, D., Fonseca, B., Paredes, H., Martins, P., Morgado, L., Rafael, J.: Development of a Mechanical Maintenance Training Simulator in OpenSimulator for F-16 Aircraft Engines. *Procedia Comput. Sci.* 15, 248–255 (2012)
5. Media Grid: Open File Formats Technology Working Group (OFF.TWG) Charter, <http://mediagrid.org/groups/technology/OFF.TWG/> (accessed: October 14, 2013)
6. Mollet, N., Arnaldi, B.: Storytelling in Virtual Reality for Training. In: Pan, Z., Aylett, R., Diener, H., Jin, X., Göbel, S., Li, L. (eds.) Edutainment 2006. LNCS, vol. 3942, pp. 334–347. Springer, Heidelberg (2006)
7. Gerbaud, S., Mollet, N., Ganier, F., Arnaldi, B., Tisseau, J.: GVT: A platform to create virtual environments for procedural training. In: IEEE Virtual Reality Conference, VR 2008, pp. 225–232 (2008)
8. Vernieri, T.M.: A web services approach to generating and using plans in configurable execution environments (2006)
9. Alhir, S.: Methods & Tools - Understanding the Model Driven Architecture (MDA). Martinig & Associates, fall (2003)
10. Obrst, L., Liu, H., Wray, R.: Ontologies for Corporate Web Applications. *AI. Mag.* 24(3), 49 (2003)
11. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowl. Acquis.* 5(2), 199–220 (1993)
12. Fensel, D.: Ontologies: A silver bullet for knowledge management and electronic commerce. Springer, Heidelberg (2004)
13. Gruninger, M., Lee, J.: Ontology Applications and Design-Introduction. *Commun. ACM* 45(2), 39–41 (2002)
14. Silva, E., Silva, N., Paredes, H., Martins, P., Fonseca, B., Morgado, L.: Development of platform-independent multi-user choreographies for virtual worlds based on ontology combination and mapping. In: 2012 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), pp. 149–152 (2012)
15. Silva, N., Rocha, J.: MAFRA—an ontology MApping FRamework for the semantic web. In: Proceedings of the 6th International Conference on Business information Systems (2003)

16. Morgado, L.: Technology Challenges of Virtual Worlds in Education and Training - Research Directions. In: 2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES), pp. 1–5 (2013)
17. Edward, L., Lourdeaux, D., Lenne, D., Barthes, J., Burkhardt, J.M.: Modelling autonomous virtual agent behaviours in a virtual environment for risk. *IJVR Int. J. Virtual Real.* 7(3), 13–22 (2008)
18. Lopes, A., Pires, B., Cardoso, M., Santos, A., Peixinho, F., Sequeira, P., Morgado, L.: System for Defining and Reproducing Handball Strategies in Second Life On-Demand for Handball Coaches' Education
19. Young, R.M., Riedl, M.O., Branly, M., Jhala, A., Martin, R.J., Saretto, C.J.: An architecture for integrating plan-based behavior generation with interactive game environments. *J. Game Dev.* 1(1), 1–29 (2004)
20. Young, R.M., Thomas, J., Bevan, C., Cassell, B.A.: Zócalo: A Service-Oriented Architecture Facilitating Sharing of Computational Resources in Interactive Narrative Research (2011)
21. Cash, S.P., Young, R.M.: Bowyer: A Planning Tool for Bridging the gap between Declarative and Procedural Domains. *Artif. Intel.*, 14–19 (2009)