Briefing Virtual Actors: a First Report on the PRESTO Project

Paolo Busetta[1] and **Chiara Ghidini**[2] and **Matteo Pedrotti**[3] and **Antonella De Angeli** and **Zeno Menestrina**[4]

Abstract. The PRESTO (Plausible Representation of Emergency Scenarios for Training Operations) project, involving industrial and research partners, aims at adding semantics to a virtual environment and modularising the artificial intelligence controlling the behaviours of NPCs to support a productive end-user development environment. This report provides an overview of the approach being followed and a few preliminary results at the end of the first year of work, concerning in particular knowledge representation, the agent framework being developed and its graphical programming environment.

1 INTRODUCTION

In this work, we focus on the behaviours of artificial characters, traditionally called non-player characters (NPCs) in gaming. Our experience is based on the development and use of a BDI multi-agent system with cognitive extensions, CoJACK [10], as artificial intelligence for simulation and serious games [4] and more recently deployed on teams of mobile robots used for live fire defence training. Our aim is dramatically lowering the complexity and cost of programming the decision-making required to a sophisticated artificial character that has to accomplish goals according to a predefined set of procedures while taking in account context (including its own emotional state) and events unfolding during the course of a game, which may affect the character's goals as well. Further, we would like to enable a user to customise and improve ("mod") a game well beyond simply changing the NPCs' appearances or selecting their behaviours from a built-in list. The artificial characters we aim at building would be "virtual actors" because they would be able to "interpret" a part written at a higher level of abstraction than current scripting, with additional modalities (that may correspond to, e.g., levels of skills or psychological profiles) that can be selected at the beginning but changed during a game as a result of the application of rules or by explicit user choice. As a consequence, the game's administrator would be empowered to become a "director" able to "brief" virtual actors, that is, to define the parts the artificial characters have to play by means of a language aimed to non-programmers that composes more fundamental even if potentially very complex behaviours into game-specific sequences.

In the long term, we envisage a very simple programming paradigm for serious game platforms supporting plausible simulated human behaviours that enables a large number of custom-made games at a very low cost, thanks to end-user development tools [9] and the ability to mix and match behavioural components taken off-the-shelf from a market place (similar in principle to asset stores in popular gaming platforms such as Unity).

This ambitious objective is at the core of a three-year project called PRESTO, being run by Delta Informatica (Trento, Italy) in collaboration with AOS Group (UK), FBK, University of Trento and others, including the Trento's public health department, APSS. PRESTO, currently completing its first year, set out to accomplish the following:

- adding semantics to a virtual environment and the actor's behaviours themselves, to enable reasoning;
- modularising behaviours, to enable their composition and selection both statically and during game playing;
- providing powerful end-user development tools for defining the parts played by virtual actors and the overall script of a game.

PRESTO supports a specific virtual reality, XVR from E-Semble, a well known tool in use for emergency management and training (EMT) in a number of schools and organisations around the world. However, PRESTO has already been tested in other gaming platforms (including Unity) and, at least in principle, is agnostic with respect to the environment in use.

This report provides an overview of the current state of progress of PRESTO with respect to the objectives identified above; various examples will be presented orally. We hope that, by the end of the project, PRESTO will have demonstrated the applicability of a cognitive architecture and formal representations to the creation of modular, reusable AI for NPC behaviours and the feasibility of development tools oriented to domain experts.

2 AN OVERVIEW OF PRESTO

In this section we provide a brief overview of the main characteristics of PRESTO, together with a report of their current state of progress.

2.1 Dynamic, moderated composition of behaviours with DICE

The core of the PRESTO project is the development of an agent framework, called DICE (Delta[5] Infrastructure for Cognition and Emotion), that supports a mixed reactive / planned approach on top of a BDI system (AOS' JACK [2, 3]). DICE uses "moderators" as in CoJACK, i.e. the representation of sub-rational factors influencing decision-making, to decide when to switch "behavioural models", that is, clusters of behaviours consistent with a psychological

[1] Delta Informatica Spa, Trento, Italy, email: paolo.busetta@deltainformatica.eu
[2] FBK (Fondazione Bruno Kessler), Trento, Italy, email: ghidini@fbk.eu
[3] AOS Group, Cambridge, UK, email: matteo.pedrotti@aosgrp.co.uk
[4] DISI, University of Trento, Italy, email: antonella.deangeli@disi.unitn.it and zeno.menestrina@studenti.unitn.it

[5] "Delta" stands for Delta Informatica Spa, the company leading the PRESTO project.

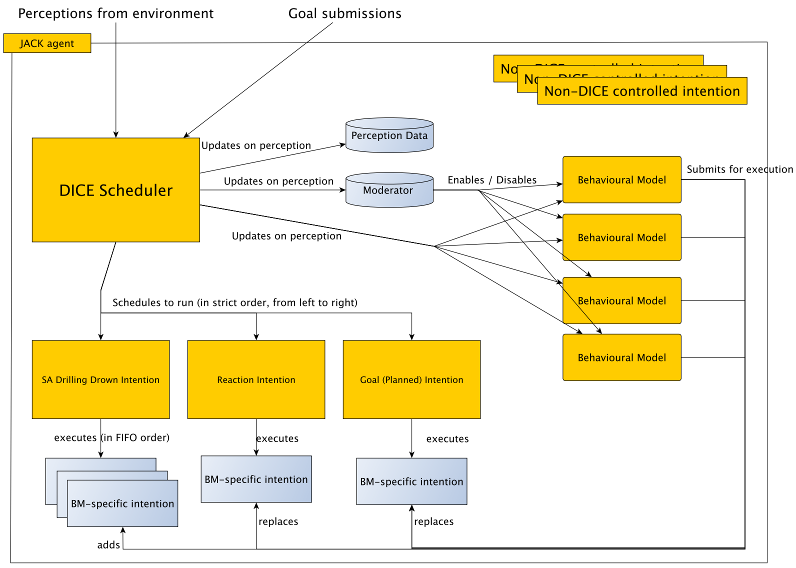

Figure 1. Overview of DICE

profile (see Sec. 2.2 below). DICE defines an architecture for: receiving perceptions from, and performing actions on, the virtual environment; selecting behavioural models according to the moderators' levels (fear and fatigue at the moment); maintaining short-term and medium-term memory; and, feeding monitoring tools that provide a real-time explanation of the choices taken by the virtual actors. Further, DICE supports the development of behavioural models that exploit its introspection capabilities (including ontological-based queries on goals and plans; see Sec. 2.3) and the real-time flow of perceptions to manipulate motivations, intentions and moderators, thus influencing both the current activities and the psychological state of a virtual actor based on a model-specific set of rules typically representing a personality type.

A simplified overview of the computational model of DICE is in Figure 1.

2.2 Behavioural modularisation

One of the challenges of PRESTO is the definition and implementation of behaviours in a way that is flexible and practically viable for a business setting. The approach we take is that of breaking complex behaviours into elementary, reusable components along the lines of both delivered functionality (the "what") and psychological profiling (the "how") that are dynamically combined by DICE. To this end, desired behaviours are analysed and decomposed in a hierarchy of goals satisfied by context-specific plans, according to a standard BDI approach but taking in account variations due to an actor's cognitive state, e.g. concerning its emotions. Plans are then clustered in components called "behavioural models" and implemented as BDI "capabilities" [1]. Goals can be given to a virtual actor from any component as well as from on-line game GUIs and from scripts developed off-line with proper end-user development tools (see Section 2.4); they are satisfied according to the behavioural models adopted at the time of their submission by the agent representing the "brain" of the actor.

Behavioural models being developed are partly universal and partly domain-specific. Universal behaviours concern, for instance, movements satisfying goals such as "go to a location of interest of a certain type" or "follow another character" with models differentiated e.g. for abilities (normal vs disabled) and psychological state (hurried vs calm). Domain specific behaviours are related to the pilot study described later and concern, e.g., patients, visitors and nurses in a hospital, with goals such as "go in the common room and chat with others" or "feed the patient" that imply coordinated behaviour and whose models are, again, differentiated by skills and psychological state. Note that a methodology and a notation for the systematic description of behaviour to be modelled (as captured e.g. from domain specialists) are among the future objectives of PRESTO.

2.3 Semantic knowledge

The ability to describe, and reason about, behaviours, goals, and plans requires the ability to represent a wide range of entities that *exist* in the (artificial) world. Even simple behaviours such as "feed the patient", or "go in the common room and chat with the doctor" require the existence of specific types of characters, spatial environments and the possibility of performing certain actions. Complex plans and goals, such as the goal "black smoke in patient room R handled" and its related plan "investigate potential fire in patient room R" are expressed not only in terms of physical entities (smoke, fire, patient, room) but may also entail or require some knowledge on the fact that they are concerned with, e.g., "urgent matters" that need to be dealt with immediately and / or cannot be suspended, in contrast to goals and plans that handle interruptible routine activities of the artificial character.

The approach taken in PRESTO is to use ontologies to define the terminology used to represent this basic knowledge, in a way that is semantically well specified and independent of a specific game or scenario [7]. This follows a stream of work in computer science, where ontologies and taxonomic representations have been widely proposed and used to provide important conceptual modeling tools for a range of technologies, such as database schemas, knowledge-based systems, and semantic lexicons [8] with the aim of fostering clarity, reuse, and mutual understanding. More specifically, our aim

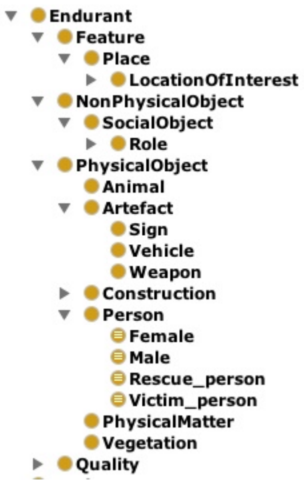

Figure 2. A portion of the PRESTO taxonomy

is to be able to define a clear lexicon to be used for the description of entities, behaviours, plans and goals in a scenario that is independent of the specific game and virtual reality, that is customisable for specific domains, and that fosters clarity, reuse, mutual understanding of code and can facilitate meta-reasoning in the DICE framework.

Rather than building everything from the ground up by using one of the state of the art ontology engineering methodologies such as METHONTOLOGY [5], the process followed in PRESTO has been driven by an attempt to first maximise the reuse of already existing knowledge and then complement it with knowledge elicited from experts by means of more traditional ontology engineering approaches such as the one mentioned above. This has lead us to consider two sources of knowledge:

- state of the art upper level (so called "foundational") ontologies, which provide generic, reusable ontology elements, such as a foundational ontology's categories (e.g., process, physical object, role), domain-independent relations (e.g., part-whole relations, participation), and ontology design patterns;
- the concrete entities existing in the virtual reality.

Our choices for the PRESTO project were the upper level ontology DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) [6], and the classification of elements provided by XVR. DOLCE was chosen as this ontology not only provides one of the most known upper level ontologies in literature but it is also built with a strong cognitive bias, as it takes into account the ontological categories that underlie natural language and human common sense.

The construction of the first version of the ontology of PRESTO was performed by following a middle out approach, which combined the reuse and adaptation of the conceptual characterisation of top-level entities provided by DOLCE and the description of extremely concrete entities provided by the XVR environment. The resulting taxonomy of PRESTO's own top ontology, a portion of which is shown in Figure 2, can be used to represent a wide spectrum of artificial world entities ranging from (physical and non physical) entities "that are" (in DOLCE called Endurants), to entities "that happen" (in DOLCE called Perdurants). Examples of physical and non physical Endurants are provided by the objects that are in the environment (including avatars and, e.g., their professional roles), which are extracted from the classification of XVR objects in a semi-automated way. Candidate Perdurants range from high level plans to low level XVR animations, which have also been used as a starting point for an initial classification.

This ontology has been a good base for the following step of analysis of the knowledge requirements of reasoning entities, which has focused in particular on non-physical objects, that is, cognitive objects that do not trivially correspond to observable physical entities. These are of particular interest for AI-based virtual reality games as they provide powerful novel concepts that help the game director (i.e. the game administrator, or a trainer in XVR) to ascribe knowledge to the virtual actors. An example of such a concept is the "location of interest" , i.e. an arbitrary place or area or volume representing a cognitive / decision-making perspective on the world that may even change during game play. While a location of interest is an Endurant for DOLCE, it turns out that Perdurants are better suited to model dynamic aspects of gaming. For instance, PRESTO classifies animations as "observable actions" (thus usable e.g. for intention recognition) and even classifies a virtual actor's own goals and plans to support sophisticated, component-independent meta-reasoning via introspection.

A substantial development effort has been spent in PRESTO to support the conceptual modeling described above. Algorithms, knowledge bases, APIs and end-user GUIs are required to allow, e.g., the addition of non-physical objects to game scenarios, the real-time extraction of perceptions from the perspective of an individual character, the classification of perceptions according to the ontology, and the conversion of ontological concepts into the appropriate entities and actions in the virtual world.

2.4 End user development

Innovative authoring tools designed for non-technical users allow the easy adaptation of the behaviour of virtual actors to specific requirements, e.g. to reach specific training objectives in a serious game. This is what we call "briefing" in PRESTO, by contrast with the (unavoidably complex and laborious) behavioural model development better left to professional analysts and programmers.

A game's director is enabled to write or adapt "parts" for a virtual actor, i.e. interpreted scripts providing sequences of goals and conditions written with a very simple graphical language but deployed as if they were behavioural model themselves, thus under DICE control. The director also decides which behavioural models must be adopted given a configuration of the virtual actor's moderators. Another tool allows the definition of partial or complete game scenarios by specifying types and number of virtual actors involved, their initial configuration e.g. of moderators, the major events that are relevant to a scenario and their expected order, the parts that specific actors have to follow when a certain event happen, and so on. Figure 3 gives a glance of the editor for the parts of a virtual actor, taken from a preliminary prototype.

3 A PILOT STUDY

The concepts and technology of PRESTO are going to be tested in a pilot study in the emergency training (EMT) domain, specifically to train hospital staff (nurses, doctors, maintenance people and so on) to properly manage a fire. A ward of the main hospital of Trento, Italy, has been reconstructed with XVR. The objective of the training is to learn to recognise situations, apply the proper procedures, and reduce the risks to patients, visitors and (costly and sometimes dangerous) medical equipment. In its standard, off-the-shelf configuration, XVR is used by the trainer to quickly build an initial emergency situation,

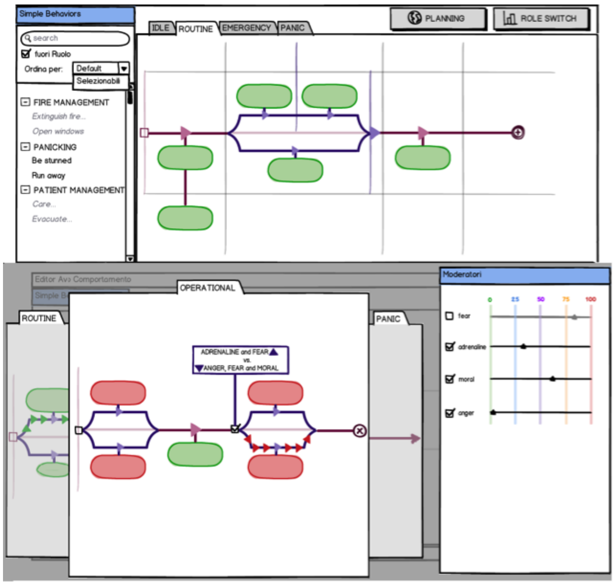

Figure 3. A screenshot from the authoring tool

to let the trainee(s) move in the scene and take decisions communicated by voice to the trainer, and to manually manipulate the scenario in real-time during a training session according to trainees' choices and desired training objectives. This very effective but highly manual procedure hits unavoidable limits when NPCs – representing e.g. patients, visitors, firefighters – need to be involved; XVR's own limited avatar animations and simplicistic automation of reactions are insufficient to provide the richness and variability of scenarios required for a realistic hospital reconstruction. PRESTO starts to provide tools for the creation and control of plausible characters' behaviours with sufficient ease to be appreciated by trainers, who are the main driving force behind the adoption of XVR within organisations. Note that the level of automation provided by PRESTO's authoring tools would move XVR very close to a full serious game development platform for non-programmers, i.e. very close to our long-term objectives.

4 CONCLUSION

PRESTO's first year of work has tackled some substantial fundamental challenges and put the basis for further refinements and extensions. We believe that the approach that has been taken is of general interest for the systematic creation and sharing of plausible decision-making behaviours in serious as well as entertainment games.

ACKNOWLEDGEMENTS

We thank Andrew Lucas of AOS e Frank Ritter of PSU for their comments during the preparation of this paper. PRESTO is partially funded by the Provincia Autonoma di Trento (Italy).

REFERENCES

[1] Paolo Busetta, Nicholas Howden, Ralph Rönnquist, and Andrew Hodgson, 'Structuring BDI Agents In Functional Clusters', in *Proceedings of the Workshop on Agent Theories Architectures and Languages (ATAL-99)*, volume 1757 of *Lecture Notes in Artificial Intelligence*, Orlando, Florida, (15-17th July 1999). Springer Verlag.

[2] Paolo Busetta, Ralph Rönnquist, Andrew Hodgson, and Andrew Lucas, 'Light-Weight Intelligent Software Agents in Simulation', in *Proceedings of The Simulation Technology and Training (SimTecT 99) Conference*, Melbourne, Australia, (29th March to 1st April 1999).

[3] Paolo Busetta, Ralph Rönnquist, Andrew Hodgson, and Andrew Lucas, 'Using Java for Artificial Intelligence and Intelligent Agent Systems', in *Proceedings of the JavAus'99 Conference*, Sydney, Australia, (February 1999).

[4] Rick Evertsz, Matteo Pedrotti, Paolo Busetta, Hasan Acar, and Frank Ritter, 'Populating VBS2 with Realistic Virtual Actors', in *Conference on Behavior Representation in Modeling & Simulation (BRIMS)*, Sundance Resort, Utah, (March 30 – April 2 2009).

[5] M. Fernández-López, A. Gómez-Pérez, and N. Juristo, 'Methontology: from ontological art towards ontological engineering', in *Proc. Symposium on Ontological Engineering of AAAI*, (1997).

[6] Aldo Gangemi, Nicola Guarino, Claudio Masolo, Alessandro Oltramari, and Luc Schneider, 'Sweetening ontologies with dolce', in *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*, pp. 166–181. Springer-Verlag, (2002).

[7] Thomas R. Gruber, 'Toward principles for the design of ontologies used for knowledge sharing?', *International Journal of Human-Computer Studies*, **43**(5–6), 907 – 928, (1995).

[8] Nicola Guarino and Christopher Welty, 'Identity and subsumption', in *The Semantics of Relationships: an Interdisciplinary Perspective*. Kluwer, (2001).

[9] Henry Lieberman, Fabio Paternò, Markus Klann, and Volker Wulf, 'End-User Development: An Emerging Paradigm', *End User Development*, **9**, 1–8, (2006).

[10] Frank E. Ritter, Jennifer L. Bittner, Sue E. Kase, Rick Evertsz, Matteo Pedrotti, and Paolo Busetta, 'CoJACK: A high-level cognitive architecture with demonstrations of moderators, variability, and implications for situation awareness', *Biologically Inspired Cognitive Architectures*, **1**, 2–13, (July 2012).