Mediterranean Archaeology and Archaeometry, Vol. 16, No 5, (2016), pp. 29-41

DOI: 10.5281/zenodo.204964

VIRTUAL AGORA: REPRESENTATION OF AN ANCIENT GREEK AGORA IN VIRTUAL WORLDS USING BIOLOGICALLY-INSPIRED MOTIVATIONAL AGENTS

Spyros Vosinakis[1] and Nikos Avradinis[2]

[1]Department of Product & Systems Design Eng, University of the Aegean, Greece
[2]Department of Informatics, University of Piraeus, Greece

Received: 20/02/2016
Accepted: 20/07/2016
Corresponding author: Spyros Vosinakis (spyrosv@aegean.gr)

ABSTRACT

Populating virtual worlds with computer controlled characters is a key issue in virtual heritage applications, an argument that can also be held as valid for the majority of virtual world applications. Virtual heritage worlds usually tend to be either devoid of people, or include computer-controlled characters that function as animated props, demonstrating pre-scripted and repetitive behaviour. In more advanced approaches, digital characters in special roles, such as virtual guides, may also be situated in the virtual world. Recent virtual heritage reconstruction works seem to acknowledge the necessity of incorporating non-human controlled characters that include intelligence in order to enhance presence and provide the user with an engaging experience. This paper presents the design and development of Virtual Agora, a virtual heritage application in the Open Simulator environment aiming to replicate daily life in an ancient Greek agora using biologically-inspired motivational agents. The application follows a multi-layered motivational model for agents that includes biological, as well as psychosocial needs. Every agent possesses a set of basic attributes that relate to its biological and physical characteristics, as well as its personality. Furthermore, agents are endowed with a set of behaviours that satisfy particular goals and consist of a sequence of actions towards achieving this goal. In addition to this generic action set, every agent possesses an extra set of actions, based on its assignment of a role or profession. The roles and the respective behaviours have been designed and selected based on available resources regarding life in the ancient agora of classical Athens. In the current implementation visitors can walk around the environment observing daily activities performed by the digital characters and interact with them by asking questions about aspects of their profession.

KEYWORDS: virtual worlds, virtual heritage, intelligent agents, virtual humans, ancient Greece

1. INTRODUCTION

Interactive 3D environments such as Virtual Worlds (VWs), in the form of serious games applications, have the capacity to serve as dissemination and learning platforms for history, culture and archaeology (Anderson et al., 2010; Gaitatzes et al., 2001). They incorporate a number of distinguishing characteristics, including high-quality visualization of digital content, real-time simulation of realistic or imaginary environments, natural and intuitive user interactions, and single- or multi-user embodiment in the 3D space through animated avatars. These characteristics enable interactive 3D environments to represent existing or reconstructed cultural artifacts and places in high detail, enhanced by features such as the superimposition of associated descriptions and media. Added interactivity allows users to freely navigate and explore the content, as well as provide engaging features such as interactive digital stories or mini-games related to the historical and cultural context of the subject. Applications of this kind not only allow people to closely observe cultural heritage artifacts that may be difficult for them to physically approach, for reasons of distance, cost or accessibility but can also serve as a motivating means to supplement people's knowledge and increase their interest in culture.

While high-quality 3D visualization of spaces and artifacts is a desired element in virtual heritage applications, it is not alone adequate to ensure that the user experience will be as engaging and fruitful as expected. There are plenty of 3D applications developed in the last two decades that provide static representations of cultural or historical content and let users freely navigate around and observe the artifacts. The problem with such approaches is that they lack any additional interactive and social features that would further engage and motivate users. On the contrary, in these environments users quickly feel a sense of void and lose interest. After all, understanding the form of constructed objects or buildings is only one aspect of history and cultural heritage; these reconstructions need to be experienced together with a number of other, intangible aspects of the related historic period and culture in order to be immersed in the cultural context (Pujol & Champion, 2012). E.g. visitors might be interested in observing aspects of daily life, activities and rituals taking place in the buildings, typical usage of the artifacts being presented, related stories and events, etc. Even more, they would like to be part of the story themselves, to be able to participate in the environment and interact with the content.

Demonstrating daily life situations and narrating short stories that aim to present social aspects of culture and life in antiquity requires the introduction of digital characters that can act these stories out in the virtual world. These characters can either be human controlled, in the form of human avatars, or computer controlled, with varying levels of adaptability to the environment or human user input (Lombardo & Damiano, 2012; Papagiannakis et al., 2005). The advancement of real-time 3D graphics and animation technology in the past fifteen years or so has allowed for the creation of realistic-looking digital characters that have the ability to move around, communicate and act in the environment, utilizing multiple means of expression, such as gestures, facial expressions or advanced locomotion. Virtual heritage applications have recently started to take advantage of the affordances of digital characters, and have incorporated them in the representation environment.

In the case of computer controlled characters, however high the quality of the visual representation or the detail of animation and other means of expression, there is often failure to achieve the expected results, in terms of user engagement. This is largely due to the fact that the agents' behavior patterns are limited, often predictable, and mainly not always in context, seeming incompatible with the high level of visual quality. This is an inhibiting factor for the acceptance of the virtual character on the part of the user and undermines the whole experience of the virtual world. Due to this fact, it has been argued that computer controlled virtual characters should be totally abolished from virtual heritage applications and any roles in the world should be played by human avatars (Morgan, 2009).

However, the behavior of digital characters can be enriched by adopting approaches stemming from the fields of Artificial Intelligence and Intelligent Agents. Instead of performing pre-scripted behavior, virtual characters can be designed as intelligent agents that possess knowledge about their environment and are endowed with a degree of autonomy that allows them to make decisions, in accordance with their perceived physical, mental and psychological profile as well as their role in the virtual world. Work towards this direction has been introduced to VWs and interactive 3D environments since the early 2000s (Luck & Aylett, 2000), and currently there are a few virtual heritage applications that incorporate digital characters based on intelligent agent architectures in their environment (Bogdanovych et al., 2010).

Adopting this approach, we are presenting a biologically-inspired intelligent agent approach for digital characters, and we are examining its suitability for virtual heritage applications from both the designer's and the final user's point of view. We have developed a platform based on VWs for creating multi-user virtual heritage applications that include multiple intelligent virtual human agents. Each agent can have its own appearance and personality traits, and may adopt one or more typical patterns of behavior (roles) according to his profession and identity in the environment. Their behavior is, however, not repetitive and predictive; they decide about their actions based on their needs and generate suitable plans in order to fulfil them. The implemented platform has been tested in a prototype edutainment application that demonstrates daily life and rituals in an ancient Greek agora. The environment includes typical buildings and roles based on available information about the Agora of Athens. Visitors can walk around, observe daily activities performed by the digital characters, interact with them by asking questions about aspects of their profession, and participate in some of the activities using their avatars.

2. ADDING LIFE TO VIRTUAL RECONSTRUCTIONS

An important element in adding storytelling capabilities and creating rich, interactive content in virtual heritage applications is the addition of computer-controlled characters. These characters can adopt a variety of roles in virtual heritage applications: they may populate virtual cities, demonstrate typical activities, present stories, or communicate with the visitors and offer them guidance to the visitors, to name a few of their potential functions. As such, they can tackle the problem of empty and inanimate places by transforming cultural representations into 'living' spaces and enhancing their interactivity.

In most cases, however, digital characters act using pre-scripted, repetitive patterns. In some cases they simply execute repetitive behavioral patterns based on scripts predefined by the designer, whilst in others they adopt shallow reactive behavior, where incoming stimuli are directly associated to corresponding reactions or short action sequences. In any interaction exceeding a few seconds, this produces repetitive behavior. This mechanistic and predictive behavior of virtual human characters makes them far from believable in the long term. Visitors expect from digital characters to contain at least some of the qualities found in human behavior, such as rationality, decision making, personality traits, emotions, etc.

2.1. *Digital characters in virtual heritage applications*

Examining the various modes of operation in digital characters, one can distinguish three general approaches. The most obvious approach to introduce digital characters in virtual reconstructions is to have them operate as '*animated props*'. Such characters usually execute pre-scripted animation sequences that are sometimes parameterized for greater diversity, and their operation is independent from any changes in the environment or user activity. Their main functionality is to augment the scenery with their presence and to make the digital environment feel livelier for the human visitor. As such, their primary contribution in cultural heritage applications is to have the user observe the appearance and typical activities of indicative people in the place and time of reference. In some cases they may be also used to play back stories, anecdotes or historical events related to the place. Nevertheless, pre-scripted characters can only produce linear, non-interactive narratives, thus leaving no room for user participation or intervention.

More interesting and dynamic character behavior can be achieved with the use of *virtual crowds*. In this case, characters move and act collectively imitating the behavior of real human crowds. The actions of each individual agent depend on the presence and motion of other agents or users nearby, and in some cases it is also affected by locations or elements of the environment that apply positive or negative attraction to them. In most virtual crowd systems, individual agents are automatically generated based on generic rules that define their appearance and properties, and they are assigned a role from a predefined set. E.g., in the work of Maïm et al (2007) a real-time simulation environment presenting a reconstructed district of ancient Pompeii has been populated with a crowd of virtual Romans. A number of human templates has been used to instantiate the crowd members with variations in clothes and body parts. The agents walked around the environment and followed simple behavioral patterns triggered by semantic labels associated with places and objects of the environment, e.g. buy bread from the bakery store, look at the window of a building, etc.

A different utilization of digital characters with more essential contribution to the user experience is to have them operate as *virtual guides*. In that case, the characters have the additional ability to communicate with the users, and their goal is make the experience more lively and pleasant for the visitors by presenting places, objects or related stories to them. There are plenty of virtual guide implementations in virtual museums and reconstructed cultural sites, which vary in terms of the means of communication and the adaptability of the presentation. A simple solution is to have the visitor choose from a list of pre-defined options regarding the content and type of presentation, whilst more sophisticated agent implementations involve communication in natural language using text or speech. Furthermore, some applications include digital characters that can take into account user preferences and context to personalize their presentations. The museum gallery presented by Oberlander et al (2008) included an agent that could communicate with visitors in natural language and adapt the presentation based on the preferences and the visiting history of users.

Finally, digital character implementations based on *intelligent agent* approaches can have a long-term, autonomous, goal-oriented operation in cultural heritage applications. Intelligent agent architectures such as the well-known *BDI (Beliefs – Desires – Intentions)* approach can be used to implement digital characters with the ability to accumulate new knowledge about the environment using the input received by their sensors, to prioritize their next tasks according to their long-term goals, and to plan a sequence of actions to achieve the desired results. This approach leads to a more elaborate behavior that is not as predictive and repeatable as in the case of following pre-scripted orders. Furthermore, such characters can expand the affordances of virtual heritage applications by demonstrating daily life aspects of ancient cultures and letting users not only observe but also interact with them. Given that the characters have an autonomous operation, any changes in the environment, e.g. due to user presence and interventions, do not distort their behavior; rather, they adapt to the new situation by re-planning their actions in order to reach their goals effectively.

The City of Uruk (Bogdanovych et al., 2010) is a virtual heritage application that uses intelligent agents in a VW to present daily life in an ancient city. The characters have been built based on a BDI architecture and follow a daily routine that involves movement, interaction with objects and communication with other characters. The agents' actions are shaped by their beliefs about the environment. They can follow pre-scripted plans in order to perform some standard activities, and they can also update their goals and generate dynamic plans as a result of certain changes in the environment. Finally, they have the ability to communicate with human visitors using natural language. They can talk about their current goal and planned actions, and they can also present information about the surrounding objects and environment. In a study aimed to validate its learning effectiveness (Bogdanovych et al., 2012) the application yielded positive results regarding student performance.

Table I. A categorization of digital characters based on their characteristics and their respective function in virtual heritage applications

Category	Characteristics	Function
virtual props	pre-scripted, repeating behavior	demonstrate appearance and typical actions
virtual crowd	dynamic locomotion, behavioral patterns	populate reconstructions, present daily activities
virtual guides	communication, adaptivity	dialogs with users, present content, provide information
living agents	autonomous, goal-oriented behavior	long-term simulation of daily life, user participation

The four categories of digital characters in cultural heritage applications presented above are summarized in Table 1.

2.2. *Believability of digital characters*

A key requirement for virtual worlds is *believability*, a rather complex concept that involves diverse aspects of virtual worlds and has led to equally diverse approaches of the matter. Before attempting to provide a definition for believability, it is important to distinguish the concept from realism, a close term that is, however distinct from believability. Realism refers to creating high fidelity reconstructions of the physical world. On the other hand, believability has to do with a synthetic character being consistent to essence of the entity it is supposed to embody as well the coherence of this character within the world it is situated in.

An obvious and important aspect of believability is the presentation of the virtual world (Magnenat-Thalmann et al., 2005). Common sense would dictate that the more realistic the presentation of the virtual world, the higher the degree of believability. However, this is not necessarily the case. Very high fidelity reconstructions of the physical world have a high degree of realism, but not necessarily believability, especially if the virtual world is devoid of people. As argued by Loyall & Bates (1997), realism is neither adequate, nor required to ensure believability. It is rather the consistency of the presentation in relation to the thematic context of the virtual world that leads to a greater degree of believability.

In regard to realism, what holds true for the unpopulated virtual world, also holds true for the characters situated within. As Mori (1970) has suggested in the well-known uncanny valley hypothesis, certain levels of realism may seriously undermine the acceptance of the character as real by the audience and produce awkward responses, especially if other constituents of the total experience such as gaze or locomotion are missing.

Believability, in the context of synthetic characters, has been defined by Bates (1994) as the ability on the part of the system to suspend the users' disbelief, by providing an illusion of life. Plainly put, believability in virtual agents is about making the human user accept they are interacting with a living character, whose existence is consistent and coherent in the context of the virtual world it is situated in.

Extending beyond the strictly physical properties of the virtual world and the agents, it has been widely discussed that believability equally involves the agent's behavior (Lester & Stone, 1997; Prendinger & Ishizuka, 2001; Ortony, 2003). Works on behavioral believability are diverse in approach, depending on what aspect of emotion one has is focusing on. A most common approach examines behavior as expressing a synthetic agent's internal state through gaze, facial expression, gesture or posture (Bevacqua et al., 2007). Believability also demonstrates a narrative aspect, both in terms of plot coherence and character expression (Ho & Dautenhahn, 2008; Riedl & Young, 2003). In terms of affect, the importance of emotion and personality has been widely discussed (Becker-Asano, 2005; Lim & Aylett, 2007).

The focus of the current work is on the generation as well as the expression of believable behavior. As argued by Ortony (2003), believability in the behavior of an intelligent virtual agent consists in demonstrating coherence in the agent's reactions and its motivational states and consistency among similar kinds of situations. Adopting a viewpoint on behavior as decision making, we aim to achieve believability in terms of producing behavior that is consistent to virtual characters' goals, state of mind and personality (De Rosis et al., 2003). We are viewing behavior as a process of making decisions and determining the appropriate course of action, taking into account the agent's role and function, as well as its physical, emotional and mental status at the moment. The outcome of this decision making process is a series of corresponding physical or communicative actions that implement the decisions made, in a way consistent to the agent's physical and affective state and traits as well as the holding conditions in the environment at the time of execution. As Dautenhahn (1998) argues, it is a matter of enabling the virtual character to produce behavior that matches what would be expected of the user, and this is something that can be accomplished by blending together various contributing elements, such as rationality, reactivity, personality and emotion.

2.3. *Needs, wants and obligations – a balancing act*

Necessary prerequisites for the development of intelligent virtual characters that demonstrate believable behavior is the incorporation of affective features (Bailey et al., 2012) and elements of autonomy (de Sevin et al., 2005). As already mentioned in the previous section, computer controlled virtual characters can be classified into four generic categories, depending on their role within the virtual world but also the level of autonomy they can demonstrate. Virtual props and crowds show relatively low autonomy or none at all and follow mainly pre-scripted behaviors with limited flexibility, whereas living agents are self-driven and can demonstrate adaptability to the environment and user actions.

As has been argued by de Sevin et al. (2005), in order to achieve high levels of believability, digital characters should be developed as autonomous agents, in the sense that they possess their own set of goals, that they proactively pursue, rather than only follow a set of given instructions. This will allow them to maintain the capacity to demonstrate purposeful behavior and continue acting even where there is no particular extrinsic goal, whereas a prescripted character in a similar case would exhaust its repertoire of behaviors, resulting in repetitive sequences of actions or purposeless trivial behavior, reducing itself to an animated prop. This is of particular importance in persistent open virtual worlds, where the user is allowed to freely explore and is not limited to specific pathways, while also the world continues to exist even after the user exits. In order to achieve autonomy, virtual agents need a mechanism to produce and select their own goals, that is, a motivational subsystem that drives the agent towards particular courses of action, aimed towards satisfying its self-interests. This motivational subsystem acts as a source of intrinsic motivation and complements any directly provided goals or other extrinsic goals resulting from social norms and obligations as well as agent and human avatar interactions within the world.

Since the agent is embodied and situated within a virtual environment where other agents co-exist, the specific driving forces incorporated in the motivational subsystem should include low level, biological factors, along with affective and social factors. Mixing the three and balancing the agent's personal goal agenda with external goals is a hard task; virtual characters should act autonomously and function as if they were living their own lives, but also possess the ability to adapt their behavior when human users interact with them or when external events occur that require their attention. Goal generation has to be complemented with a differentiation of behavior among individual agents, consistent with their particular physical, social, metal or emotional characteristics, which constitute their virtual persona.

3. A MULTI-AGENT PLATFORM FOR VIRTUAL HERITAGE APPLICATIONS

We have developed a platform for cultural heritage applications in VWs that include multiple autonomous agents moving and interacting with the users and the environment. The platform allows the execution of complex virtual heritage scenarios that may include features such as:

- demonstration of activities, habits and rituals of ancient cultures,
- agents communicating with users and presenting locations, elements or activities of interest, and
- scenarios requiring user participation, such as interactive stories and quests

The platform provides a set of high-level tools and reusable components that aim to assist developers in creating new applications and programming the interactive behavior of virtual agents. It includes a *plan definition language* to design and test static agent plans, a *planning engine* to dynamically produce plan sequences in order to achieve a desired goal, a *dialog engine* for the agent-user communication, and a *programmable perception mechanism* to identify elements of interest. A prototype environment of an Ancient Greek Agora with multiple virtual agents demonstrating typical roles has been developed using this platform.

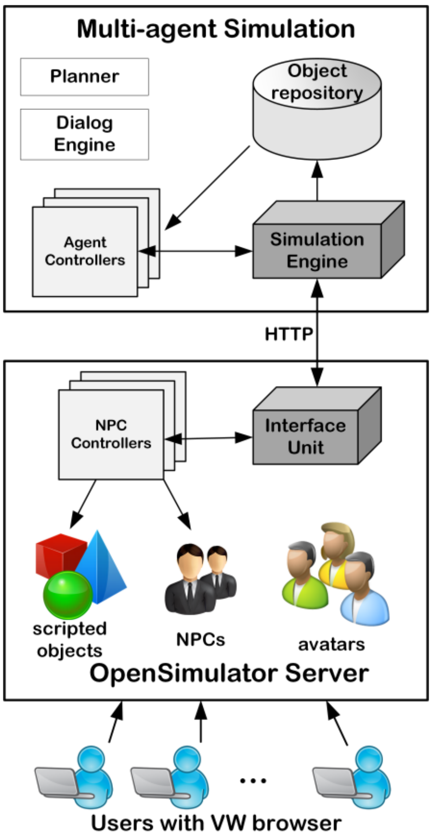

Figure 1. System architecture of the multi-agent platform.

Our platform is based on OpenSimulator, an open source alternative to the VW of Second Life. Users connect to an environment running in an OpenSimulator server using specific VW browser software and have the ability to personalize their appearance, navigate around, meet and communicate with other users, interact with the objects of the environment, and even to construct new content provided that they have the appropriate rights to do so. The environment provides simple and usable tools for in-world building, allows complex geometry models (meshes) to be imported, and produces a good quality rendering of the environment. As such, it is suitable for the collaborative reconstruction, exploration and visualization of cultural heritage sites. Furthermore, it is equipped with a powerful scripting language named LSL, which can be used to program the interactive behavior of the 3D elements contained in the environment. Finally, it allows the introduction and control of multiple digital characters (termed NPCs – Non Player Characters) in the environment.

The architecture of our platform follows a three-tier client-server model based on the OpenSimulator server (Figure 1). Users join the application using a compatible browser that connects to the OpenSimulator server, and they are embodied as avatars in the representation space. The interactive behavior of the environment is orchestrated by a special object named *Interface Unit* that mediates between the objects and NPCs of the VW, and a *multi-agent simulation environment*. The latter is an external application implemented in Java, which handles the agents' perception, decision making, dialogs and high-level actions using an abstraction of the actual VW and its contents. Thus, the agent operation takes place in two parallel layers: the low-level execution layer of the VW, in which the NPCs move and interact with the objects and users of the environment, and the high-level layer of the multi-agent simulation, in which the agents update their beliefs, prioritize their goals, and decide about their next actions.

The elements of the VW that have an active role in the application are three: the Interface Unit, the *NPC Controllers* and the *scripted objects*. The functionality of the Interface Unit is twofold: it constantly updates the multi-agent simulation with any changes happening in the VW, i.e. users and NPCs moving, communicating and interacting, and it also triggers and controls the action execution of individual NPCs. For the first function, it makes use of a specially built extension to the OpenSim server (Region Module), which monitors the placement of all entities of the environment and identifies any changes. For the second function, it makes use of the NPC Controllers. Each NPC created in the environment is attached a scripted object, the NPC Controller, which initiates and controls the execution of the supported actions. When the execution of an action by a specific NPC is requested, the Interface Unit communicates with the respective NPC Controller and transmits the request using the action name and any additional parameters, e.g. lookAt user1. When the NPC completes the action, the controller notifies the Interface Unit that the action execution has finished, and the latter forwards the message to the multi-agent simulation environment. Finally, there are agent actions that involve some interaction between the NPC and an object of the environment, e.g. to grasp an object, or to leave it in a specific position. The objects that support these interactions include a specific script that receives requests from the NPC Controllers and triggers the respective responses.

The multi-agent simulation environment controls and monitors the behavior of the agents that participate as NPCs in the application. The operation of each agent is driven by a respective *Agent Controller*, which updates its beliefs, takes any required decisions and executes the current plan. The agents' sensory input is based on an *object repository* that stores basic geometric information (size, position and rotation) of all elements that actively participate in the simulation, i.e. agents, users and selected objects. The repository is constantly updated by the Interface Object to reflect the active status of the VW. The multi-agent environment is equipped with two additional components to support the agent operations: a *planner* based on PyHOP (Nau, 2013) that can generate dynamic plans according to the status of the environment, and a dialog engine based on AIML (Wallace, 2009) for the agent-user communication.

3.1. A biologically-inspired agent model with needs and personality traits

The agent implementation in our platform follows a multi-layered motivational model that includes biological, as well as psychosocial needs (Figure 2). Every agent possesses a set of basic attributes that relate to its biological and physical characteristics, as well as its personality. Following a needs hierarchy, needs arising at the lowest motivational level are assigned greater priority than needs on a higher level, therefore the corresponding goals are queued first for satisfaction. Need priority is also determined by the intensity of each corresponding motivation, as well as an internal priority index corresponding to the relative urgency of needs within the same level.

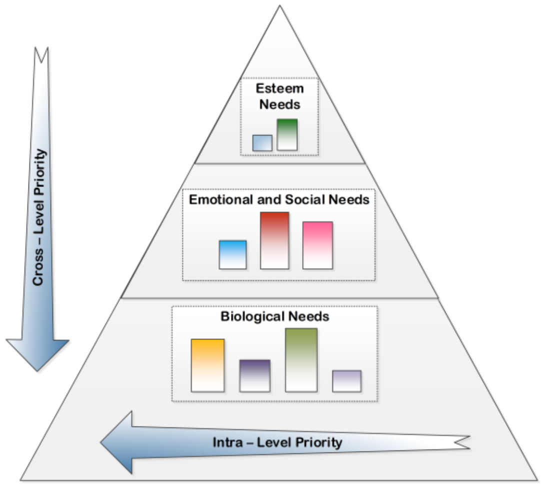

Figure 2. Motivational hierarchy model.

At the lowest level, a biologically inspired model (Avradinis et al., 2013) monitors and regulates the thirst, hunger, hygiene and sleep needs based on the value of corresponding reserve variables. Reserve variables are regulated by means of a homeostatic mechanism that is based on human physiology and tries to maintain the agent in a well-being state. When the value of a reserve variable moves outside a defined comfort zone, a corresponding need rises, in turn activating a behavior aiming to satisfy the particular need.

Every agent is endowed with a set of behaviors that satisfy particular goals. These behaviors are described as task networks that are evaluated to a sequence of actions towards achieving the goal based on the actual status of the environment. In addition to a generic behavior set that is assigned to all agents, selected characters with an assigned role or profession possess an additional set of behaviors to fulfil goals related to their specialization. The agent architecture is presented in Figure 3.

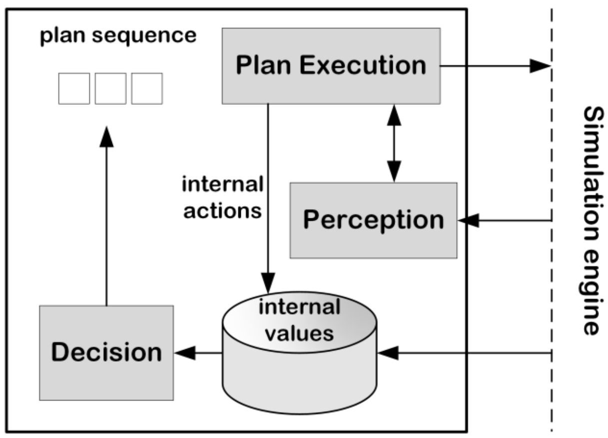

Figure 3. Agent architecture.

3.2. Planning and Action Execution

The execution of a new action sequence is triggered when a certain need fires. In that case, the agent sets as a goal to fulfil the need and decides about its next steps based on the actual status of the environment. An HTN planner (Erol et al., 1994) is being utilized to reach that decision. All possible agent goals, e.g. to drink water in order to quench its thirst, are described as hierarchical task networks, where higher level tasks trigger the execution of lower-level ones based on the properties of the environment. At the lowest level are the primitive tasks which have a direct mapping to the scripted plans that the agent executes in the VW. Whenever a planning operation is requested, the HTN planner is given an appropriate initial state representation that includes appropriate properties and relations of objects of interest based on their representation in the Object Repository. The planner tries to detect an appropriate task sequence based on the initial status and the preconditions and effects of each possible sub-task. The result is a sequence of primitive tasks with respective arguments that are executed by the agent and lead to a number of NPC actions.

The plans that the agent can execute in the environment are described using a dedicated high-level scripting language, the *Plan Definition Language (PDL)*. Each plan defined in this language has a unique name and an arbitrary number of argument variables. The plan implementation is encoded in an imperative programming manner that allows sequential, conditional and iterative execution of commands, based on the definition of PDL. The supported commands fall into the following categories:

- actions that are executed by the NPC in the VW,
- "internal" actions that affect the internal values of the agent,
- message passing between agents,
- dialogs with users,
- memory processes (store / retrieve temporary information),
- perception of specific elements based on criteria

Finally, the execution of a new plan can be called during the plan implementation, thus leading to more complex plan definitions.

Each plan that is defined using PDL must be described as a primitive task in the task network and be assigned appropriate preconditions and effects to be able to participate in the planning process.

3.3. NPC actions in the 3D environment

The NPCs can perform a variety of actions in the VW. They can walk to certain targets, animate their body, interact with objects, and communicate with users. All actions are triggered and controlled by the NPC Controller that is attached to them.

There are four distinct actions for NPC walking. The first is to have the NPC wander around in a given location, i.e. a part of the environment defined by the designer with a unique name. E.g. a number of citizens wander around in the agora when they have no urgent needs to fulfil. The second action is to ask the NPC to walk to an entity (object, agent or user) up to a specific distance from its center. E.g. if the agent decides to talk to a user, the NPC walks to the user's avatar keeping a 2m distance. The third action makes the NPC walk to a specific position related to an existing object. The designer can add an arbitrary number of named spots to the scripted objects of the environment. These spots are useful for having the NPC take correct position when performing an action, e.g. to perform the action of drinking water from the fountain, the NPC has to be in an appropriate position with respect to the fountain before animating its body. Finally, the 'look at' action makes the NPC orient its body towards a target entity.

An NPC can start or stop playing animation sequences using respective actions. The OpenSimulator environment has already a large number of animation sequences, and new animations can also be created and uploaded in the environment. Using these two actions, the NPCs can demonstrate a variety of activities in the VW.

Two NPC actions are provided for communicating with users. The first one causes the agent to say a message in public using the chat channel of the VW. The second triggers a dialog with a specific user. In that case the NPC not only sends the message to the public channel, but also listens for replies from the specific user. The replies are forwarded to the Dialog manager of the multi-agent simulation environment to proceed with the dialog.

Finally, NPCs can attach or detach objects to specific body parts based on respective actions. When an NPC decides to hold an object, e.g. take an apple from the tree, the NPC Controller sends a message to the target object, and it attaches to the requested body part of the NPC, e.g. the right hand. OpenSimulator allows designers to adjust the attachment point of the object with respect to the NPC body part, and thus the objects that participate in the environment can be designed to be correctly attached to the characters. An NPC can also detach an object from its body and leave it in a specified spot.

3.4. Internal Actions

Besides actions that have a direct visual effect in the VW, there are also actions that affect the agents' internal values. E.g. an agent eating an apple is performed as an animation of her holding the apple and bringing it to her mouth, but it also has an effect on her food reserves. The latter is an 'internal' action, which is not executed in the VW but in the multi-agent simulation environment. We have currently implemented a number of internal actions that affect the agents' reserves, such as eating, drinking, resting, working, etc. The final effect of these actions on the agents' internal value is determined by the argument values and also by other relevant attributes of the agent, e.g. sex, weight, age, etc. An example internal action is the 'bite <food>' action, where the type of food determines the calorie increase per bite.

3.5. Inter-agent communication

The implemented multi-agent platform includes a simple message-passing mechanism for the communication between agents. An agent can send a message to any other agent using a designated action, and the message will be instantly submitted to the receiver. Messages are formed as subject-predicate-object triples and are not restricted to a specific vocabulary. It is thus the responsibility of the designer to ensure that messages are properly interpreted. A second action allows an agent to read incoming messages from other agents. When a message is received, the sender id and the message triple are stored in selected variables for further processing. An example of message passing between agents is the following. A citizen meets the pottery seller and wishes to buy an amphora. The message would be 'me wantsToBuy amphora'. The seller identifies that the citizen wants to buy something, checks the product price and replies with a message 'amphora priceIs 5'. The citizen learns the price, checks if she has the required amount of money or an object of equivalent price to trade, and proceeds with the transaction or cancels it accordingly.

3.6. Dialogs with users

Agents can start dialogs with users using the dialog engine, which is based on the Artificial Intelligence Markup Language (AIML). Designers can create a number of AIML bots, each of which is based on its own collection of AIML files that define how it responds to user input. In our platform we use the bots as dialog patterns that can be chosen depending on the type of dialog that the agent decides to start. E.g. some agents might serve as virtual guides and present the various places of the environment, whilst others might wish to present their own profession, or even to assign a quest to the user. The action that initiates a dialog takes as argument the name of the user and the name of the bot to be used as pattern. It is, therefore, possible to have the same agent use more than one patterns during its lifetime, or to use the same pattern for multiple agents.

We have extended the AIML language with custom tags in order to embed dynamic information in the dialog and to trigger new agent actions during the discussion. There are tags that can dynamically insert information from the object repository and from the agents' internal status, like the name of the user with whom the agent is discussing, the current activity and needs of the agent, etc. Furthermore, we have created another tag that can trigger the execution of a scripted plan. When this tag is called, the agent executes the plan and then continues with the dialog. This feature is important, as it can lead to a more natural and human-like behavior that blends dialogs and activities. A simple example would be to look and point at a building while discussing about it. In a more complicated example, the agent could ask the user to follow her, walk to a place of interest and present the activities taking place.

3.7. Memory Processes

Designers can define and use new variables in PDL to store intermediate data while executing a plan. However, there may be cases where long-term storage of information may be needed. Sometimes important knowledge may need to be stored during the execution of one plan and retrieved later on, during another. E.g. an agent whose role is to guide users might need to remember whether he has spoken to a specific user before, or not. Having such information will allow her to direct her attention towards newcomers. A repository for managing such information is included in the platform, and respective actions for storing and retrieving data to and from the memory are provided. Memory clauses follow the same conventions as agent messages: they are formed as subject-predicate-object triples.

3.8. Perception Mechanism

A simple perception mechanism enables agents to identify specific objects of interest in the region in order to execute their actions. The process of perceiving the environment in virtual agent platforms is based on the idea of 'filtering' the surrounding entities based on specific criteria, in order to generate usable information that will facilitate action execution (Bordeux et al., 1999; Vosinakis & Panayiotopoulos, 2003). E.g. if an agent decides to eat an apple to satisfy her hunger, she has to find the nearest apple tree and select a reachable apple from that tree before performing the action of grasping it. Both these decisions can be supported by a perception mechanism that searches for entities matching the required properties.

The perception mechanism can be triggered using a specific command in PDL. It combines custom filters regarding the properties of entities in the environment and returns a single entity that matches the filters, if any. The returned entity is stored in a variable for further processing. The available filters can be about:

- the type of entities (e.g. objects, users, NPCs)
- their description, if any (e.g. apple, basket, temple, etc), applicable only to objects
- their role, if any, applicable only to NPCs (e.g. worker, priest, etc).
- a statement about them in the agent's memory (e.g. me spokenTo).

Filters such as the above can be combined using logical operators (and, or, not) and create more complex filtering mechanism. Finally, the user has to provide the criterion to select a single entity from the list of entities that pass the perception filters, and the variable in which the result will be stored. The implemented selection criteria are two: nearest and any. In the first case, the mechanism selects the entity with the shortest distance from the agent.

4. VIRTUAL AGORA: DAILY CITY LIFE IN CLASSICAL ANTIQUITY

The Virtual Agora is an experimental interactive edutainment environment, aiming to present life in ancient Greece and educate users about common daily activities and important religious rituals. The setting of the Virtual Agora is inspired by actual Ancient Agora sites and includes buildings typically contained in a classical ancient forum. The main focal point of the Agora is the Temple of Hephaestus, along with a sacrificial altar, modeled according to measurements from Hepahestus' Temple in Athens' Agora (Dinsmoor, 1941). The Temple is where commoners gather to satisfy their religious needs. A public water source, modeled after the Enneakrounos source in Athens' Agora, is located nearby the Temple and is where citizens could get water as well as bathe themselves (Figure 4). The setting also contains workshops and stands, where craftsmen and salesmen could create and sell their goods.

Figure 4. The water source alongside the temple of Hephaestus.

Virtual characters in the Agora have appearances that conform to their distinct roles. A priest, a potter, a fruit seller as well as a few commoners inhabit the Agora, each performing their own duties and acting according to their own roles. All characters are free to wander around the place, however, particular installations or buildings can only be used by the corresponding professional. Commoners can interact with the craftsmen and priest and buy goods or otherwise request their services.

A craftsman's typical behavior is working, which involves the use of a raw material (clay, metal or straw), to be processed into pottery, ornament or basket. These items are then put on display, to be sold to prospective clients. A food vendor typically engages in a gathering behavior, picking fruit from nearby trees as well as a selling behavior, selling fruit to clients. The food vendor also sells wine to interested clients, stored in small size amphorae. Food and wine purchased by the food vendor can either be used as an offering in the temple, or directly consumed by the virtual agent, to satisfy its hunger or thirst. Priests stand by the temple, performing their own religious duties, until a worshipper requesting a religious service approaches. The priest can perform libations and food sacrifices, as well as pray on behalf of the worshipper. Figure 5 shows a typical scene in front of the altar.

Characters can interact with human users, in the form of simple AIML dialogue interactions that provide information on the place and the daily activities of each character, as well as directions towards key points of interest.

Figure 5. NPCs in front of the altar waiting for priest to perform libation.

All characters have a set of biological needs that have to be satisfied as the characters get hungry, thirsty, sleepy or tired. The rise of a need generates goals that are satisfied through a series of actions. In addition to these needs, the characters also lose their interest and get bored when they have nothing particular to do, which motivates them to engage in any sort of active behavior. They also have a need of spirituality, which causes them to visit the temple. All these needs are satisfied by a set of behaviors. These are compound plans, with alternative implementations through sequences of actions. Some behaviors and actions are common among all Agora dwellers, however some are specific and linked to each agent's profession. Such is, for example the behavior of performing a libation, for the priest, or the crafting of an item, for the potter.

Table II. Professions and common behaviors of Virtual Agents

Profession	*Sub-role*	Profession Behaviours	Common behaviours	Actions
Priest		-Perform offering -Perform libation	-Acquire offering -Visit temple -Offer item -Request libation ritual -Buy item -Satisfy hunger -Satisfy thirst -Satisfy hygiene -Rest	-pick up item -drop item -put on table/put in place -give item to other agent -take item from other agent -eat -drink -wash -sit -sleep
Craftsman	Metal Worker Basket Weaver Potter	-gather materials -create (craft) item -sell item		
Food Vendor		-pickup food -sell food/wine		

All citizens, craftsmen and commoners alike, have a need of spirituality. When this spirituality need exceeds a predefined threshold, dependent on the agent's personality, the agent initiates the "perform_religious_duty" behaviour, which causes the agent to first cleanse itself by visiting the fountain and then visit a shrine or a temple. At a shrine, the agent can satisfy the spirituality need by praying to the god the shrine is devoted to. At the temple, in addition to praying, the agent can also interact with the priest and worship the god either by offering wine and requesting a libation or by giving the priest a votive offering, such as food, pottery or a metal ornament, to be acquired at the corresponding vendor or craftsman.

For example, let us consider a complex task for a character assigned the role of a potter, which is creating a pottery artifact. The potter first needs to acquire the materials required for the artifact and then actually start making the object itself. The top level tasks (acquire clay, acquire water, make pottery) are further decomposed hierarchically into subtasks, that are implemented by operators, corresponding to actions executed by the agent. A plain language task breakdown into subtasks for the potter scenario is as follows.

```
create Pottery
    acquire clay
        go to clay pit
        pickup clay
        go near potter's wheel
        drop clay on potter's wheel
    acquire water
        acquire jug
        go to nearest fountain
        hold jug near water
        go near potter's wheel
        drop jug on potter's wheel
    make pottery
        pickup jug
        pour jug on clay
        turn wheel
        pickup clay
        put clay on potter's table
end (create Pottery)
```

Gathering the materials requires several trips within the virtual world on behalf of the potter, which causes the consumption of energy, water as well as the fall of its levels of hygiene. This triggers further goals that will have to be satisfied after the pottery artifact has been created. The intertwining of goals regarding biological needs as well as emotional or social needs with obligations such as work or other externally given goals allows for rich and believable behavior, that is largely self-driven rather than mostly dictated.

5. CONCLUSIONS

In this paper we have described the design and early development of Virtual Agora, a virtual heritage application for demonstrating daily life activities and rituals in ancient Greece. Our implementation has been based on a generic platform for multi-agent applications in virtual worlds, which focuses on believable, long term, dynamic behavior of digital characters. The platform allows designers to easily extend and adjust existing scenarios, or even to create new ones and extend the functionality of the application. Furthermore, the platform is not linked in any way to specific objects, appearances or customs of ancient Greece, and, as such, it can be further used in alternative cultural heritage applications. The implemented environment currently supports only a few user activities, but we are planning to extend it with further affordances, such as quests and mini-games.

From our experience with the implemented prototype and the early test scenarios, it seems that the adoption of a generic multi-agent architecture for virtual heritage applications has both prospects and pitfalls. On the positive side is the fact that it gives freedom to the designers to define the functionality and behavior of digital characters that suits the application. An agent can demonstrate an action, simulate daily life, present places of interest to the users, communicate with them, participate in common activities with users, etc. All these possibilities may be a useful starting point to create novel approaches to demonstrating intangible culture. On the other hand, designers have little control on the final outcome, as agent behaviors are a result of complicated planning procedures, and a lot of fine-tuning is needed to achieve the desired effect. It seems that further research is needed regarding the needs of virtual heritage applications and the appropriateness of current agent architectures.

In the future we are planning to conduct a number of user evaluations of the Virtual Agora environment and assess its effectiveness as an edutainment environment.

ACKNOWLEDGEMENTS

We would like to thank Ms Grigoria Dimitrellou for her kind assistance in 3D modeling work.

REFERENCES

Anderson, E. F., McLoughlin, L., Liarokapis, F., Peters, C., Petridis, P. and de Freitas, S. (2010). Developing serious games for cultural heritage: a state-of-the-art review. *Virtual Reality*, Vol. 14(4), 255–275.

Avradinis, N., Panayiotopoulos, T. and Anastassakis, G. (2013) Behavior believability in virtual worlds: agents acting when they need to. *Springerplus 2013*, Vol 2:246

Bailey, C., You, J., Acton, G., Rankin, A. and Katchabaw, M. (2012) Believability Through Psychosocial Behaviour: Creating Bots That Are More Engaging and Entertaining, in Hingston P. (Ed), *Believable Bots*, Springer Berlin, 2012, 29-68.

Bates, J. (1994) The role of emotion in believable agents. *Communications of the ACM*, Vol. 37(7), 122-125.

Becker-Asano C., Prendinger H., Ishizuka, M. and Wachsmuth, I. (2005) Evaluating Affective Feedback of the 3D Agent Max in a Competitive Cards Game, *Proceedings of Affective Computing and Intelligent Interaction*, LNCS 3784, 466–473.

Bevacqua, E., Mancini, M., Niewiadomski, R. and Pelachaud, C. (2007) An expressive ECA showing complex emotions. *Proceedings of the AISB Annual convention, Workshop on Language, Speech and Gesture for Expressive Characters*, Newcastle UK, 208-216.

Bogdanovych, A., Rodriguez-Aguilar, J. A., Simoff, S. and Cohen, A. (2010). Authentic Interactive Reenactment of Cultural Heritage With 3D Virtual Worlds and Artificial Intelligence. *Applied Artificial Intelligence*, Vol. 24(6), 617–647.

Bogdanovych, A., Ijaz, K. and Simoff, S. (2012). The city of Uruk: Teaching ancient history in a virtual world. *Lecture Notes in Computer Science*, Vol. 7502, 28–35.

Bordeux, C., Boulic, R. and Thalmann, D. An efficient and flexible perception pipeline for autonomous agents. *Computer Graphics Forum*, Vol. 18(3).

Dautenhahn, K. (1998) The Art of Designing Socially Intelligent Agents - Science, Fiction, and the Human in the Loop. *Applied Artificial Intelligence,* Vol. 12, 573-617.

De Rosis, F., Pelachaud, C. and Poggi, I. (2003) Transcultural Believability in Embodied Agents: A Matter of Consistent Adaptation. In Payr, S. & Trappl, R. (Eds.), *Agent Culture: Designing Human-Agent Interaction in a Multicultural World*, Lawrence Erlbaum Associates.

De Sevin, E. and Thalmann, D. (2005) A motivational Model of Action Selection for Virtual Humans. *Proceedings of Computer Graphics International 2005 (CGI),* New York, USA: IEEE Computer Society Press.

Dinsmoor, W.B. (1941) Observations of the Hephaesteion. *Hesperia: Journal of the American School of Classical Studies at Athens, Suppl. 5,* Swets-Zeitlinger Amsterdam (repr. 1975), 1-164

Erol, K., Hendler, J. and Nau, D. S. (1994). HTN planning: Complexity and expressivity. *Proceedings of AAAI*, 1123-1128.

Gaitatzes, A., Christopoulos, D. and Roussou, M. (2001) Reviving the past : Cultural Heritage meets Virtual Reality. *Proceedings of the 2001 Conference on Virtual Reality, Archeology, and Cultural Heritage*, 103–110.

Ho, W.C. and Dautenhahn K. (2008) Towards a Narrative Mind: The Creation of Coherent Life Stories for Believable Virtual Agents. In: Prendinger, H, Lester, J, Ishizuka, M (Eds) *Lecture Notes in Computer Science 5208*, Springer Berlin-Heidelberg, 59-72.

Lester, J.C. and Stone, B.A. (1997) Increasing believability in animated pedagogical agents. *Proceedings of the first international conference on Autonomous agents (AGENTS '97)* New York, USA, 16–21.

Lim M.Y. and Aylett R.S. (2007) Feel the difference: A Guide with Attitude! *Proceedings of Intelligent Virtual Agents 2007*, 317–330.

Lombardo, V. and Damiano, R. (2012) Storytelling on mobile devices for cultural heritage. *New Review of Hypermedia and Multimedia*, Vol. 18, 11–35.

Loyall, A.B. and Bates, J. (1997) Personality-rich believable agents that use language. *Proceedings of the First International Conference on Autonomous Agents*. ACM Press, NY USA, 106–113.

Luck, M. and Aylett, R. (2000) Applying artificial intelligence to virtual reality: Intelligent virtual environments. *Applied Artificial Intelligence*, Vol. 14(1), 3–32.

Maïm, J., Haegler, S., Yersin, B., Mueller, P., Thalmann, D. and Gool, L. Van. (2007) Populating Ancient Pompeii with Crowds of Virtual Romans. *Proceedings of VAST: International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage*.

Magnenat-Thalmann, N., Kim, H.S., Egges, A. and Garchery, S. (2005) Believability and Interaction in Virtual Worlds. *Proceedings of the 11th Intl. Multimedia Modelling Conference (MMM'05)*. IEEE Computer Society, Washington, DC

Morgan, C. (2009) (Re)Building Catalhoyuk: Changing Virtual Reality in Archaeology. *Archaeologies: Journal of the World Archaeological Congress*, Vol. 5(3), 468-487.

Mori, M. (1970) The Uncanny Valley, *Energy*, Vol. 7(4), 33–5.

Nau, D. (2013) Game applications of HTN planning with state variables. *Proceedings of ICAPS Workshop on Planning in Games*.

Oberlander, J., Karakatsiotis, G., Isard, A. and Androutsopoulos, I. (2008) Building an adaptive museum gallery in Second Life. *Proceedings of Museums and the Web*, Montreal, Quebec, Canada.

Ortony, A. (2003) On making believable emotional agents believable. In: Trappl, R., Petta, P., Payr, S., (Eds) *Emotions in humans and artifacts*. Cambridge, MA: MIT Press.

Papagiannakis, G., Schertenleib, S., O'Kennedy, B., Arevalo-Poizat, M., Magnenat-Thalmann, N., Stoddart, A. and Thalmann, D. (2005) Mixing virtual and real scenes in the site of ancient Pompeii. *Computer Animation and Virtual Worlds*, Vol. 16(1), 11–24.

Prendinger, H. and Ishizuka, M. (2001) Let's talk! Socially intelligent agents for language conversation training. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, Vol. 31(5), 465–471.

Pujol, L. and Champion, E. (2012) Evaluating presence in cultural heritage projects. *International Journal of Heritage Studies*, Vol. 18(1), 83–102.

Riedl, M.O. and Young R.M. (2003) An objective character believability evaluation procedure for multi-agent story generation systems. *Proceedings of Intelligent Virtual Agents*, 278–291.

Vosinakis, S., & Panayiotopoulos, T. (2003) Programmable agent perception in intelligent virtual environments. *Proceedings of Intelligent Virtual Agents 2003 Conference.*

Wallace, R. S. (2009) The anatomy of ALICE. In Epstein, R., Roberts, G., Beber, G. (Eds) *Parsing the Turing Test*, Springer Netherlands