UNIVERSITÀ DEGLI STUDI DI TRENTO

UNIVERSITY OF TRENTO

DEPARTMENT OF INFORMATION ENGINEERING AND COMPUTER SCIENCE

MASTER DEGREE IN COMPUTER SCIENCE

FINAL THESIS

TITLE

BMCT: A UI FOR HIGH LEVEL ORCHESTRATION OF SERIOUS GAMES

SUPERVISOR:
PROF. ANTONELLA DE ANGELI

GRADUANT:
ZENO MENESTRINA

CO-SUPERVISOR:
DR. PAOLO BUSETTA

ACADEMIC YEAR 2012/2013

BMCT: a UI for High Level Orchestration of Serious Games

Zeno Menestrina

Department of Information Engineering and Computer Science

University of Trento

via Sommarive 14, 38123, Trento, Italy

zeno.menestrina@unitn.it

ABSTRACT

Serious games have been growing in importance over the years, proving to be a viable alternative to formal lectures and training sessions in real life. "Learning by doing" and the opportunity to recreate the same interactive scenarios multiple times make these experiences more stimulating for the students and more affordable for the trainers. Nevertheless, these products suffer from a lack of customization of content; the game is handed to the instructor as a finalized product, providing minimal authoring features to model the sessions according to the needs of trainer and trainee. Core elements of serious games are Non-Player Characters (NPCs) which, depending on the quality of the Artificial Intelligence (AI), can make the scenarios more realistic and involving. This paper proposes a solution as regards the personalization of AIs, allowing a non-technical user to construct and edit complex behavioral models through a new high-level approach. BMCT has been developed following a user-centered design and evaluated by eight users. Results show that the chosen approach is suitable and a good basis for future evolution.

Author Keywords

End-user Development, Serious Game, Interaction Design.

1. INTRODUCTION

Serious game can be briefly described as a game designed for a main purpose other than mere entertainment; the aim is the progression of the users' skills in the real world through practice in a simulated environment [**1**]. Serious games allow the player to improve perception, attention and memory processes through "learning by doing", making the experience more stimulating than the typical formal lecture. The gaming component gives the player a more challenging environment, while the simulation of real scenarios increases the involvement of the user. The possibility to repeat the experiment without limits could potentially allow a high level of mastery of the content.

Serious games are largely used for military purposes [**2**,**3**], both for the training of technical crews as well as simulation of complex military operations. Beyond these, there are many other applications in a variety of fields, such as:

1. Healthcare: the game concept has been integrated into rehabilitation procedures [**4**], used to improve mental well-being [**5**] or more generally to support training of physical activity [**6**];
2. Education: to acquire and strengthen complex cognitive skills [**7**,**8**];
3. Business: to simulate contexts of sale, interaction and, more generally, situations that require direct experience [**9**];
4. Emergency management: to train and evaluate competence and problem solving capabilities of the personnel (e.g., firefighters, police, and hospital staff) [**10**,**11**].

The focus of this study is strictly related to emergency management, which aims to educate and train safety supervisors, firefighters, and other professionals. In this area, serious games usually have a dynamic which involves two main actors: the trainee and the instructor. The first is placed in front of a simulation representing a real emergency scenario with a series of goals to accomplish. During this playing session the instructor is the director of the game: he can stop the simulation, redefine the goals, generate/interrupt events, and more.

Serious games should be flexible, allowing adjusting the training sessions to the players' needs. Despite the importance of flexibility, this requirement is still under-achieved. In most cases the game is handed to the trainer as a finalized product, with none or minimal custom-tailoring authority. The intervention of game programmers is needed for every personalization of the product. It is clear, with these premises, that there is a need for new tools to offer to the instructor the chance to modify the sessions according to a training program and to create an individual training experience specific to each trainee. Serious games should provide End-User Development (EUD) features, allowing users of software systems, who are acting as non-professional software developers, at some point to create, modify or extend a software artifact [**12**].

One main focus of the customization should be the AI of the Non-Player Characters (NPCs). In this way the instructor would be able to modify the NPCs' behaviors according to the different scenarios, creating a more realistic environment and making a more involving training session. In this context, this thesis - part of the PRESTO project in collaboration with Delta Informatica - aims to provide a new approach to the customization of serious games, recapped in a tool called BMCT (Behavioral Modeling Customization Tool).

After several design sessions in collaboration with the Delta team, we have developed a beta version of the BMCT. The interface was evaluated by eight experts in computer science and interaction design. The positive results proved that our approach is applicable and a good starting point for further development.

The paper is organized as follows: Sections 2 and 3 present an introduction to PRESTO and a careful analysis of the state of the art of AIs and serious games; Sections 4 and 5 describe the design and implementation of the interface; Sections 6, 7 and 8 present the evaluation of the interface and the discussion of the results. Finally Section 9 presents reflections regarding the tool and future work.

2. PRESTO

PRESTO is a project by Delta Informatica which aims to develop a serious game for emergency training, including a series of new authoring features, entrusting the instructor with customization powers and making the training sessions more adaptable.

Our project is an add-on to PRESTO, conceived to offer an end-user development tool which allows people with limited experience in programming to easily create or edit complex behavioral models.

Another add-on, currently under development, concerns the construction of a director's timeline. This tool will allow the instructor to define and trigger specific events at specific times during the sessions and manage in an easy way the different NPCs and events in the scenario. PRESTO includes other sub-parts. The visual component relies on XVR, virtual reality training for safety and security professionals developed by E-Semble. The AI is based on CoJACK, a cognitive architecture used for modeling the variations in human behavior [**13**].

3. RELATED WORKS

For this project we analyzed the research literature and the products on the market (video games, serious games, virtual reality in general) related to the concepts of customization of contents, abstraction of the programming language and information visualization.

Customization of AIs

Serious games are widely studied by the academic community, but customization is a subject largely unexplored. More usually this concept is seen as the personalization of the game design according to user requirements. One example are the works made by Gobel [**6**] and Brisson [**14**] that explored new models for the development of customizable games, in the sense that the gameplay dynamics are adjusted according to the user's needs.

We were not able to find any papers on the specific topic of the development of authoring tools for the construction of behavioral models. As to the video game industry, it offers nowadays an almost infinite choice of products, but among them a small percentage offers the possibility to personalize the content.

Looking at the low number of videogames provided with authoring tools, the customization can be considered a minor factor in the quality and appeal of a game (unless the gameplay dynamics are based precisely on this feature). The hit series, whose strengths rely heavily on the complexity and richness of the levels and/or a strong online community, usually allow to create, edit and share new levels and mods[1] to increase the longevity of the game.

The released authoring tools of AIs are usually "light" versions of those used by the developers of the various companies. For ease of implementation AIs are usually developed using paradigms such as FSM (finite state

[1] "A mod is a set of formal or functional changes to a video game, created by professionals or by gamers, in order to maintain, improve or simply make the game different." [**30**]

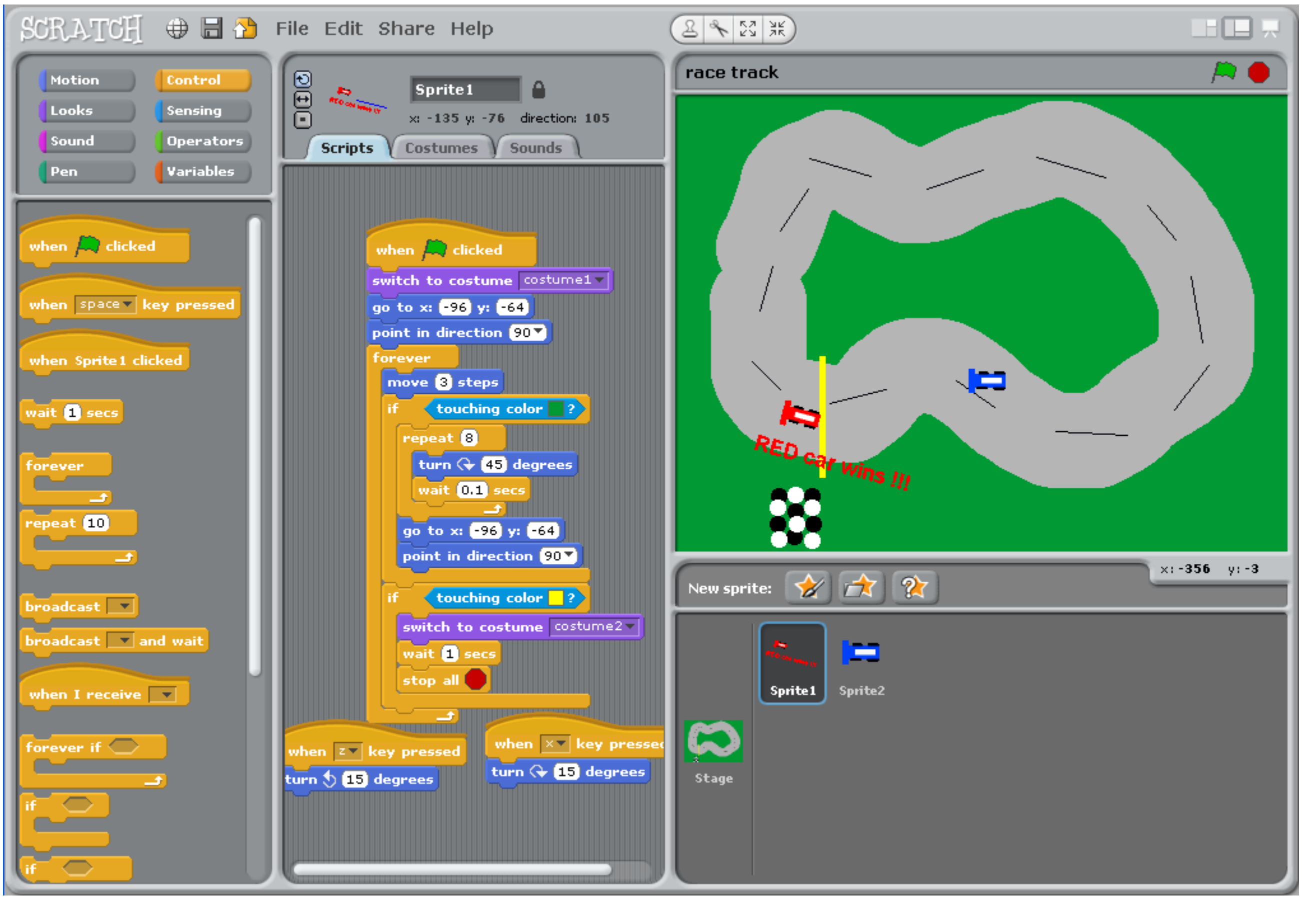

Figure 1. An example of Scratch. The left column shows the "control" category of the toolbox; the central column contains the developed scripts; the right column shows the rendering of the game.

machine) or HFSM (Hierarchical FSM), solutions, for game developers, with very good value for simplicity and effectiveness. Consequently the tools, as in the case of Kismet (part of Unreal Development Kit) or BST (part of Crysis SandBox), are based on more or less complex state diagrams. Although some are easier to learn than others, the interaction approaches are too low-level. The complexity of these tools requires good programming skills and a basic knowledge of AIs' architecture.

The problem persists even in games in which the structure of the AI is simpler, as in the case of F.E.A.R. (First Encounter Assault Recon) – videogame series developed by Monolith Productions - which propose a modular architecture of AIs [**15**]. Even though the use of this sort of black boxes makes the behavioral model much more "readable", the related authoring tool (F.E.A.R. SDK) proves to be far too complex for a non-technical user.

More interesting is the on-going work by Bethesda for the development of the TES (The Elder Scroll) Construction Set, an editing software for the video games TES III: Morrowind, TES IV: Oblivion and TES V: Skyrim. In the latest version the user is able to model some aspects of the behavior of the NPCs through parameters with semantic values (e.g., Confidence = "Cautious"). Customization is achieved by selecting verbal labels describing desired behaviors. It is possible for almost anyone to construct behavioral models; however there are no elaborate behavior structures. Another feature of the tool, the AI Package, allows the construction of detailed action plans through a hierarchical system depending on the NPC's mode (e.g., default, spectator, combat). Although this type of editor is definitely easier than writing the AI's algorithms, it is not sufficiently abstract to be used by a novice user and requires at least a basic programming background.

Levels of abstraction

The studies by Smelik [**16**], van Est [**17**] and Tutenel [**18**] share a common topic in the exploration of new paradigms for high-level development. Even if they do not deal with AIs, their works propose new interesting concepts for the construction of worlds and storylines, which partially inspired our work. The same applies for the authoring of interactive storytelling[2], a topic in

[2] "Interactive Storytelling (IS) is a form of digital entertainment in which users create or influence a dramatic storyline through actions, either by issuing commands to the story's protagonist, or acting as a

which publications have been growing over the last ten years. Scribe [**19**], Art-E-Fact [**20**] and Scenejo [**21**] were developed with the aim to allow a single user to generate a large amount of story content, turning the author into artist and programmer.

These latter products are usually based on a graph metaphor, which are easy interfaces to learn for non-programmers, but give rise to usability problems due to low efficiency (too much work, even for basic results) and the complexity of the final results.

In this analysis, special attention was given to Scratch [**22**], a project of great importance regarding the development of new paradigms of high-level programming. This tool, developed by MIT, has the goal of introducing programming to kids. The classical programming approach is replaced by a modular structure; the division into categories and colors (Figure 1) makes navigation between modules clearer and helps the user build a mental model of a programming language. Through basic units such as cycles, controllers, timers and others, it is possible to construct functioning programs. Although the method is inconvenient for the creation of complex programs, Scratch has proven, with more than three million shared projects, to be very simple and effective with regards to the introduction of the subject of programming [**23**].

As to the videogame industry, the products that best take advantage of high-level programming concepts are the Integrated Development Environments (IDE) for videogames, such as Unity3d, Game Maker, RPG Maker and Game Editor. In addition to these there are video games that integrate into the gameplay the contents' customization, such as Trackmania, a racing game where the tracks are built and shared by the community.

All these products have common elements that make their use simple enough for users with no deep programming knowledge. Above all is the use of the "What You See Is What You Get" (WYSIWYG) approach, which lets users immediately observe the results on the IDE of the performed actions. Another common component is the use of minimalistic toolboxes/palettes, to facilitate learning and memorizing of the meaning of the different controls. Obviously, such simplicity is dependent on the complexity of the product itself. While the IDEs remain quite complex, in the specific case of Trackmania, it is possible to perform any action using a palette reduced to no more than ten elements.

In summary, from the analysis of this field, it is possible to define three requirements that determine the effectiveness of a high-level language for non-technical users:

1. The user shall be able to perceive clearly the result (can easily understand the state of the model for every performed action);
2. The user shall be able to perceive with clarity the value of his/her actions (can easily realize what an action will produce);
3. The user shall be guided in the process of development.

Information Visualization

Mazza [**24**] describes information visualization as the element between the raw data and information, which provides the methods and tools with which to organize and represent the data to finally produce information. This is accomplished by organizing data into a meaningful form, presenting it in appropriate ways, and communicating the context around it.

Effective visual representation can enhance cognitive processes, because they allow inferences that humans are able to do in an easy way. Larkin and Simon [**25**] carried out an empirical study comparing visual and textual descriptions while solving physics problems. They concluded that the first are expressively more effective due to three properties:

1. *Locality*: each element has its place in the physical space. For example, different pieces of data can be represented by different visual elements positioned in the immediate spatial vicinity, allowing the reader to better compare;
2. *Minimizing labeling*: it is related to the human ability to recognize information represented in a visual format, without the need of further description;
3. *Perceptual enhancement*: the ability to carry out a large number of inferences through visual representations, allowing us to distinguish relationships and dependence between data.

Edward Tufte defines a set of rules for "graphical excellence" [**26**], which can be generalized and used as a reference for a good representation of information. Hence a visual representation should:

- Show the data;
- Induce the viewer to think about the substance rather than about methodology, graphic design, or something else;
- Avoid distorting what the data have to say;
- Make large data sets coherent;

general director of events in the narrative. Interactive storytelling is a medium where the narrative, and its evolution, can be influenced in real-time by a user." [**29**]

- Encourage the eye to compare the different pieces of data;
- Reveal the data at several levels of details.

Tufte regards graphical excellence as "complex ideas communicated with clarity, precision and efficiency" [**26**].

Fundamental to the study of a correct representation of information is also the perception that the user has of the visualization. The visual mapping – a process that lies in defining which visual structure to use to map the data - is fundamental in designing a visual representation. Views generated by these processes create strong representations of data which can be rapidly perceived.

A proper transformation from the data to suitable graphical elements is at the base of an effective visualization, in which the most important patterns are clearly distinguishable from their surroundings.

Colin Ware [**27**] defines four properties which can help the user through the process of understanding the information contained in a visual representation:

1. Colors. Color needs particular attention because it strongly depends on cultural and physiological factors. Both hue and intensity can be used for the visual detection of elements that are distinguished from the surroundings;
2. Form. Attributes such as shape, orientation, number and size can be exploited to characterize the different elements;
3. Spatial position. A specific position for each element can help the user to define the space of the visual representation;
4. Movement. Attributes such as flicker and motion can be used to catch the attention of the user to specific elements.

In the design phase, it is important to carefully define the attributes to use, since the number of visual distinctions of a single attribute is limited due to our short-term memory feature that has to process the meaning of each encoding. Ware [**27**] suggests no more than eight different hues, four different orientations, four different sizes, and all the other visual attributes to less than ten distinct values.

Another main concept for the design of the visual representation is the interaction that the user has with the interface. The information visualization aims to support the user to take advantage of his/her perception for exploring data at various levels of abstraction. A weak point of these systems comes out when the amount of data to represent is overwhelming, in which case the manipulation of the visual representation by the user is fundamental. According to Shneiderman a proper representation should consider a specific rule: "First, overview, then, zoom and filtering, finally details on demand" [**28**]. The core of his idea is that information visualization should support the user in the exploration and analysis of the represented data.

We have taken as data points for the design of our interface some of these techniques for the manipulation and transformation of the visual representation:

1. Overview plus details: the basic idea is to show both a detailed part and a global view of the entire representation. In this way the user can have a view of the overall structure while exploring specific parts of the representation;
2. Bifocal view: in the case where the visual representation is too large for the screen, it is possible to represent a distorted version, in which the information of interest is shown in the center of the screen, while the sides are distorted. In this way it is possible to keep the useful information on top and at the same time provide the context in which it is placed;
3. Filter and reorder: data filtering can be used by the user for eliminating any data items that are not considered relevant. Reordering can be useful to understand if there exists any correlation between data.

4. DESIGN

The design process involved a series of seven workshops with the company. The sessions were structured in a simple way: a first meeting with the team of Delta Informatica for updates/changes to the PRESTO project and the requirements of the BMCT was followed by a series of new prototypes in the form of digital static sketches, made using Balsamiq mock-up tool.

The prototypes, which were reviewed collectively, have been useful to identify the strengths and weaknesses of our interface and for the redefinition of the structure of the AI developed by Delta Informatica. The BMCT project was carried out in parallel with the development of the PRESTO project; the latter has undergone many changes and the structure of the AI has been remodeled several times. While the end-user requirements have remained unchanged, the amount and type of data needed for the behavioral models has changed continuously during the design phase, which lasted nearly two months.

The interface's interaction is focused on three sets (routine, operational and panic), used to define the different modes of action of the NPC. The routine mode defines the action plan of the NPC under normal conditions, without any specific event triggered in the scenario (e.g., a nurse on her daily round). The operational mode defines the actions to take in case of

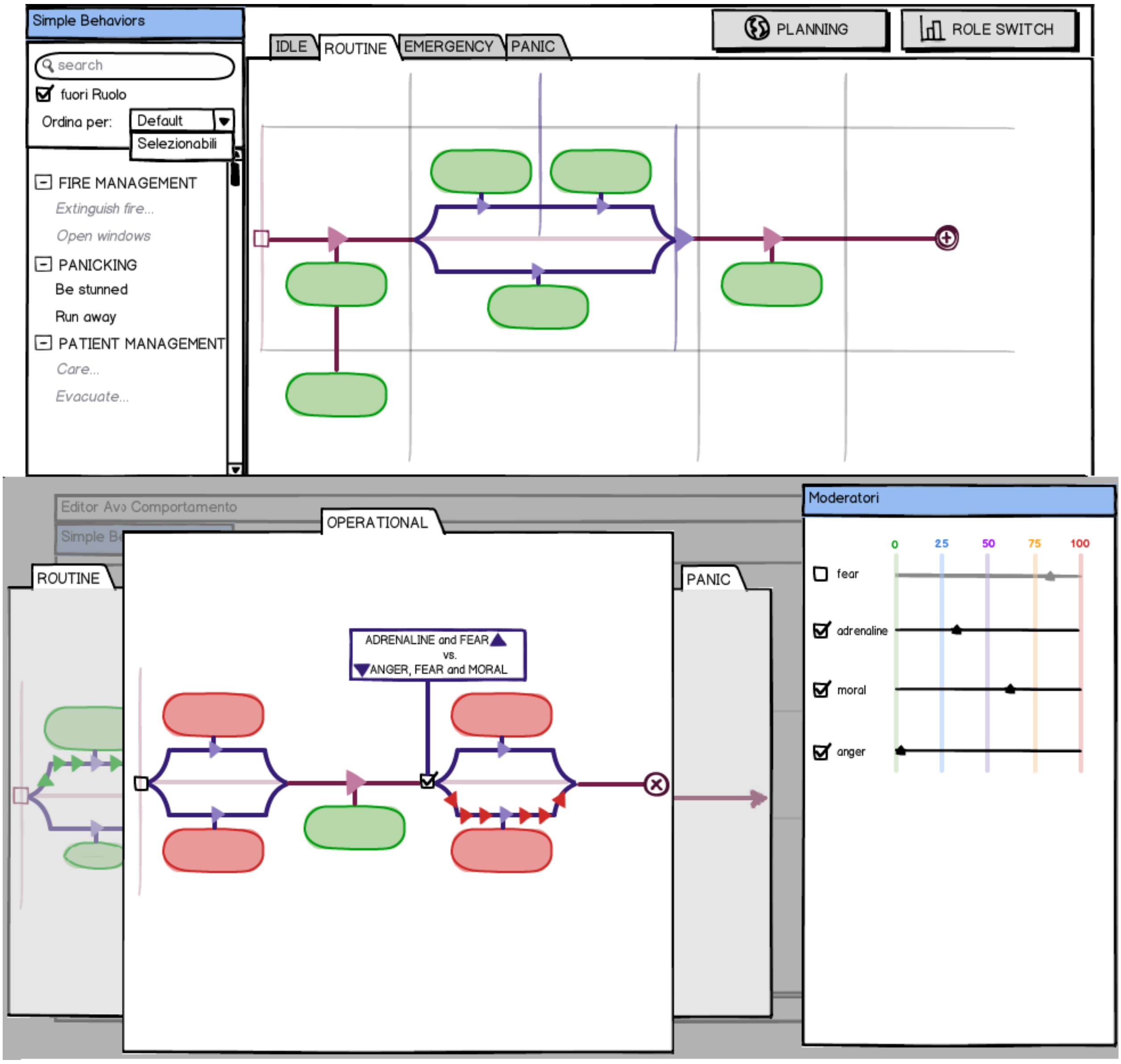

Figure 2. Main view of the BMCT (up) and moderators' view (down)

specific emergency events (e.g., the nurse starts the evacuation plan in case of fire). The panic mode is a situation in which the NPC loses control of its actions and is activated by the internal emotional states of the character (e.g., the nurse is too scared and runs for her life).

At the end of the seventh session of design, we produced the final prototype (Figure 2). In this version, the structure of the behavioral model has been completely redefined with respect to the previous, including the use of SPAL (Sequence, Parallel, Alternative and Loop), a new paradigm designed to allow the user to build complex behavioral patterns through a low-level approach.

Each mode has a dedicated canvas in which the trainer can define the corresponding SPAL model. The actions, chosen from the different categories, are placed on the canvas and each block of actions is executed from left to right. In this way, even though time is not an integrated parameter, the user can identify a sequence in the events. The model is therefore not a true flowchart, but more a path of action that the NPC crosses during the session.

The AI behind PRESTO provides a series of emotional curves which determine the intensity of each single moderator[3] (for each NPC) during the session. The moderators have a specific view where the user can study what actions will be performed by the NPC, in the case of alternative paths in the plan action, according to the value of the various moderators. Thereby it is possible to give a clearer idea of the behavior of the character during the training session in

[3] Emotional parameters (e.g., fear, stress) used to influence the quality of the action.

accordance to the different events, but it is not possible to manually calibrate these values. In fact this is a read-only feature and the user is not allowed to modify the emotional states of AI.

SPAL

The SPAL paradigm is based on four concepts:

1. Sequence: each action or set of actions belonging to the same column is carried out after the one to its left has been completed;
2. Parallel: actions in the same column and in the same quadrant, are performed at the same time;
3. Alternative: actions, or set of actions, in the same column but on different quadrants are performed with the logic of "exclusive OR" (i.e., one or the other);
4. Loop: once a structure of actions is completed, it is possible to define a checkpoint from which to restart the plan of action.

The overall structure of the SPAL is simple, but even if the atomic elements are reduced to four, it is possible to build quite complex behavioral model. In fact the compositionality of SPAL allows constructing multiple layers of nested structures; ideally the only limit is the "readability" of the final structure.

The loop element of the SPAL is a clear example of the importance of a right balance between the complexity of the constructed model and the complexity of the interaction to build it. In a standard flowchart this would be a conditional step, which the user could use to define the necessary conditions for possible branching or looping. Given our target user, this kind of solution has been considered too close to classical programming. The checkpoint system is simpler and yet still retains a certain refinement in the final model.

5. IMPLEMENTATION

After the final prototype was defined, it was implemented in a fully working interface for the evaluation of the usability of the BMCT. This was developed in Microsoft Silverlight. The main reasons for choosing this application framework was dictated by the possibility of developing rich internet applications, allowing reuse in different projects.

During this phase, there were modifications on the interactive version. The interaction with the interface has been enhanced with editing features, such as the deletion of specific elements on the canvas and the insertion of new elements (e.g., insert an action in between a parallel block and an alternative one).

A series of visual feedback have also been added in order to facilitate the use of the interface and to make easier to understand the impact of the actions made on the model by the user: a system of different cursors shows the effects of dropping on the different areas of the canvases, helping to differentiate between sequential, parallel and alternative actions and determine where it is possible to insert new actions.

The algorithms behind the interface allow an automatic reordering of the model, for easy reading. The actions and/or blocks of actions are held in a grid structure: the action sequences are placed in different columns; actions in parallel are in the same column but in different rows below the abscissa; the alternatives are on the same column, but on lines of opposite quadrants. Every action is visually connected to the structure through lines and arrows to help the user visualize the relationships between the elements of the model.

Finally it has been decided to constrain the compositionality, keeping a basic structure for the model. The users are allowed to place actions in sequence or in parallel or in alternative, but cannot combine them (e.g., a sequence within a branching is forbidden). This choice reduces the freedom on behavioral modeling, but avoids the creation of too complex and obscure structures.

The main limitation of BMCT is a real attachment with PRESTO. It was possible, in collaboration with a programmer of Delta Informatica, to write a database, but its construction was made specifically for the BMCT project. This database of actions and moderators contains parameters such as the conflicts between actions, the relationships between moderators and actions and the relationships between actions and modes, but, being made *ad hoc*, it does not fully reflect the current state of the PRESTO project. For the same reason it was not possible to give the user a visual feedback from PRESTO to show a relationship between the constructed model and the result on the simulator. However, this limitation was remedied by inserting a slideshow of actions and structures of actions.

6. EVALUATION

The evaluation involved eight participants, four from the department of Information Engineering and Computer Science (DISI) and four from Delta Informatica. The group was composed by six males and two female, between 25 and 53 years old, with an average of 31 years. The subjects have a deep knowledge in computer science and Human-Computer Interaction (HCI): three of them have a background in information technology, one of them in HCI and three participants have both.

The group has been chosen for the capability to assess both complexity and usability of the interface. Computer experts had the knowledge to evaluate the expressivity of SPAL and determine if it was enough elaborate to model complex behaviors. HCI experts looked at the interface more focused on the user's needs and evaluating if SPAL was easy to understand and learn. For these reasons the background of the participants made them both users and experts.

The evaluation sessions, each lasting about 30 minutes, involved a single user at a time and were divided into two parts. In the first, which lasted about 5 minutes, was presented the context of BMCT, followed by an explanation of the features and commands of the interface. In the second part, the tester was asked to perform a series of tasks in order of increasing difficulty. For example, the first task was to perform a simple "drag & drop" of an action on one of the canvases, while one of the last tasks of the session required the construction of a complex behavioral model based on a textual description. On this last model, the participant was asked to activate the specific view of the moderators, set the parameters and see the resulting change in the plan of action.

We recorded both the screen and the audio, as the participants were asked to externalize their thoughts and problems through the “think aloud” protocol. Every task was assessed with a score from 0 to 2: 0 for failure; 1 for inaccuracy or too slow in accomplish the task (more than 20 seconds for simple tasks); 2 for success. At the end of the session the participant was asked to provide a more detailed description, and suggestions, of the tasks which required more effort.

7. RESULTS

The evaluation by the experimenter according to the users performances have brought to light some usability issues and some problems of the interface.

The main problem, which occurred in seven out of eight cases (Table 1), is related to the representation of the actions on the canvas, and more precisely on their deleting actions. This graphical representation, consisting of a rounded rectangle superimposed by labels and comboboxes proved to be too small for interaction: the users had difficulty to accomplish cancellations and in some cases this issue led to unwanted cancellations (e.g., trying to make a selection in a combobox). In addition, all participants deplored the lack of an option to "clear all", without which the user was forced to manually delete every item on the canvas.

	P1	P2	P3	P4	P4	P6	P7	P8
ERASING ISSUES	✓	✓	✓	✓	✓	✓	✓	
“CLEAR ALL” MISSING	✓	✓	✓	✓	✓	✓	✓	✓
MORE LABELS NEEDED	✓	✓		✓				✓
SCROLLING ISSUES	✓			✓				
FORBIDDEN AREAS NOT VISIBLE	✓							
INITIAL AMBIGUITY ALTERNATIVE AND PARALLEL	✓		✓		✓	✓	✓	
ACTIONS' CATEGORIZATION ISSUES			✓	✓				
“DRAG&DROP” OF INNER ELEMENTS						✓	✓	✓

Table 1. Distribution of the issues experienced during the sessions.

Another editing problem, highlighted in three out of eight cases, is the lack of "drag & drop" inside the canvas. The user cannot move items already in the canvas, and is therefore obliged to delete and insert a new instance of the desired position.

Participants have also found some problems with regard to the navigation. The majority of users complained of the limited use of labels: while the color choice of the different sections of the screen was considered very useful, they noted that a greater use of labels would help to better define the spaces. In addition to this, there were minor problems of navigation in the canvas, for example, the scrollbar being hard to see, and the difficulty in the choice of actions with regards to the categorization.

The difference between parallel and alternative actions created some problems in the early stages. Even though it was explained at the beginning of the session, most of the testers made some errors in performing the first tasks. In fact an average of 46% of testers had a score of 1 (while the other had a 2), for what concerns tasks specifically related on the parallel and alternative concepts. However, users were able to recognize their mistakes, and they all completed correctly the later part of the session. Asked to construct a behavioral model, according to a verbal description of the desired behaviors, using all the elements of SPAL, 100% of participants achieved the task with a score of 2.

8. DISCUSSION

The results are positive in several respects. First of all, the most important problems did not involve the SPAL, which was considered sufficiently simple and intuitive. In fact, the later tasks, in which the participants had to construct complex behavioral models, were carried out in an error-free and rapid manner. Although the testers had some initial problems with the concepts of parallel and alternative actions, once the ambiguity was resolved, they recognized the simplicity of construction of the structures of behaviors. The evaluation proved that once the users understood the meaning of each element and relation of the BMCT, they had no problem with these notions later on. This fact indicates the future need, once the tool will be released, for a practice training session with the instructor.

Another positive factor is the consistency of the encountered issues. The evaluation's results show that the testers usually came across the same issues, which allow us to focus on specific problems of the interface, without having to rethink the entire design. Most of these problems can be almost considered as design oversights (e.g., "clear all" feature), hence can be fixed easily in the code. In general, the suggestions gathered during the evaluation sessions have given useful insights to their resolution.

Limitations

The evaluation lacked the participation of domain experts and it was not possible to make any validation in the field. By the fact that PRESTO and BMCT are still at an early stage, it was considered more appropriate to submit in the near future a more complete product to avoid any reluctance to the transition to PRESTO by the trainer, due to a premature presentation of the product. Such limitations can be considered normal for this kind of study; therefore we consider our results acceptable.

9. CONCLUSION

The current stage of development of BMCT was considered successful. The implementation from the interface reflected the ideas developed during the design and its evaluation gave positive results.

The SPAL paradigm allows the user to build models of some complexity, but does not have the level of details that the PRESTO's Artificial Intelligence grants to the user. The presence of conditionals in replacement of the loop, giving the possibility to build branches and multiple checkpoint, and the addition of a more complex compositionality (e.g., parallel and sequential inside of alternative paths, or alternative path with multiple branches), would allow the construction of more complex models that could fully exploit the potential of AI.

For this reason, the next step after the completion of the BMCT will be the development of multiple layers of abstraction. This will help provide an overview to the user, and potentially be a solution to the problem of getting overly complex models. These layers may be used by the trainer, but will be designed specifically for the role of the modeler, which will be able to easily build more realistic models of behavior to be provided to the instructors.

This work is being conducted in parallel with the PRESTO project. Even now the Delta team has established new requirements for the system and the BMCT needs to be adapted and updated with new features. The collaboration will continue for a period of one year, so the current state of BMCT can be considered an early version, a foundation of the future advancements of the project.

ACKNOWLEDGEMENTS

I want to thank Antonella De Angeli for her guidance and help; Paolo Busetta and Piero Consolati for the opportunity to collaborate on PRESTO; Jacopo Grandi for his cooperation and support throughout the BMCT project; Sylvie Noël for her support during the evaluation phase and for the many advices; Elena Livieri for her patience and precious suggestions; my parents for their constant and priceless support during all my university career; all the members of the HCI group of the department of information engineering and computer science and the Delta team for their help in the evaluation of the interface.

REFERENCES

1 Wikipedia contributors. "Serious game". *Wikipedia, The Free Encyclopedia* (2013).

2 Bohemia Interactive Simulations. *http://products.bisimulations.com/*.

3 U.S. ARMY GAMES FOR TRAINING PROGRAM. *http://www.peostri.army.mil/PRODUCTS/USAGFTP/*.

4 De Mauro, Alessandro. "Virtual Reality Based Rehabilitation and Game Technology". In *EICS4Med* (2011), pp. 48-52.

5 Felix Navarro, Karla et al. "A Dynamic and Customisable Layered Serious Game Design Framework for Improving the Physical and Mental Health of the Aged and the Infirm". In *IARIA Conference* (Guadaloupe 2011), pp. 140-145.

6 Gobel, Stefan et al. "Serious Games for Health: Personalized Exergames". *MM '10 Proceedings of the international conference on Multimedia* (2010), pp.

1663-1666.

7 Nadolski, Rob et al. "EMERGO: A methodology and toolkit for developing serious games in higher education". *Simulation and Gaming*, 39, 3 (2008), pp. 338-352.

8 Muratet, Mathieu et al. "Experimental Feedback on Prog&Play: A Serious Game for Programming Practice". *Computer Graphics Forum* (2010), pp. 61-73.

9 Boinodiris, Phaedra. "Playing to Win: Serious Games for Business". *Winter Issue of The Bridge on Frontiers of Engineering*, pp. 34-39.

10 El Mawas, Nour and Cahier, Jean-Pierre. "Towards a knowledge-intensive serious game for training emergency medical services". In *Proceedings of the 10th International ISCRAM Conference* (2013).

11 Blochel, Kevin et al. "A Serious Game for Measuring Disaster Response Spatial Thinking". *ArcUser* (2013).

12 Lieberman, Henry et al. "End User Development". *Springer* (2005).

13 Ritter, Frank et al. "CoJACK: A high-level cognitive architecture with demonstrations of moderators, variability, and implications for situation awareness". *Biologically Inspired Cognitive Architectures* (2012), pp. 2-13.

14 Brisson, Antonio et al. "Artificial Intelligence and Personalization Opportunities for Serious Games". In *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference* (2007).

15 Orkin, Jeff. "Three States of a Plan: The A.I. of F.E.A.R." In *Game Developers Conference* (2006).

16 Smelik, Ruben et al. "Interactive Creation of Virtual Worlds using Procedural Sketching". In *EG 2010* (2010), pp. 29-32.

17 van Est, Casper, Poelman, Ronald, and Bidarra, Rafael. "High-Level Scenario Editing for Serious Games". In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications* (Vilamoura 2011).

18 Tutenel, Tim et al. "Using Semantics to Improve the Design of Game Worlds". In *Proceedings of the Fifth Artificial Intelligence and Interactive Digital Entertainment Conference* (2009).

19 Medler, Ben and Magerko, Brian. "Scribe: A Tool for Authoring Event Driven Interactive Drama". *TIDSE'06 Proceedings of the Third international conference on Technologies for Interactive Digital Storytelling and Entertainment* (2006), pp. 139-150.

20 Iurgel, Ido. "From Another Point of View: Art-E-Fact". In *Technologies for Interactive Digital Storytelling and Entertainment* (2004).

21 Weiss, Sebastian et al. "Scenejo – An Interactive Storytelling Platform". In *Virtual Storytelling. Using Virtual Reality Technologies for Storytelling* (2005).

22 MIT. "Scratch". *http://scratch.mit.edu/*.

23 Resnick, Mitchel, et al. "Scratch: Programming for All". *Megazine Communications of the ACM*, pp. 60-67 (2009).

24 Mazza, Riccardo. *Introduction to Information Visualization*. Springer-Verlag, London, 2009.

25 Larkin, Jill and Simon, Herbert. "Why a Diagram is (Sometimes) Worth Ten Thousand Words". *Cognitive Science*, 11 (1987), pp. 65-100.

26 Tufte, Edward. *Visual Display of Quantitive Information*. Graphics Pr, 2001.

27 Ware, Colin. *Information Visualization: Perception for Design*. Morgan Kaufmann, 2012.

28 Schneiderman, Ben. "The eyes have it: A task by data type taxonomy for information visualizations". In *Proceedings of the 1996 IEEE Symposium on Visual Languages* (1996).

29 Wikipedia contributors. "Interactive storytelling". In *Wikipedia, The Free Encyclopedia* (2013).

30 Wikipedia contributors. "Mod (video gaming)". In *Wikipedia, The Free Encyclopedia* (2013).