Exploring Creative Freedom in Real Time Story Generation

Gonzalo Méndez and **Raquel Hervás** and **Pablo Gervás**[1] and **Alejandro Martín** and **Frank Julca**[2]

Abstract. Interactive storytelling systems have become very popular as they engage users in the creation of narrative. In order to explore a co-creation process where the final result is narrated while the user is playing, we have designed an interactive storytelling system that is able to generate a narration in real time while giving the user the possibility to decide what to do and where to go. The intended result is a tool that allows the author, the user and the system to co-create a narration giving the user, as the ultimate author of the resulting narration, the freedom to decide the course of the events. In this paper we focus on the structures that have been used to represent the stories and in the mechanisms designed to generate the narration so that a coherent and entertaining story can be presented to the user.

1 Introduction

Interactive storytelling systems have become a popular way to engage users in the creation of narrative elements. However, in many occasions, the narration as a voice or textual product takes second place to the interactive experience within the game, as a way to let the user take the role of one of the characters inside the game and influence the story.

One of the most entertaining aspects in a narration is to read or listen to a skilled narrator that wraps the listener into the atmosphere of the story. This is the case of the game *The Stanley Parable* [24], an interactive, narrative-driven video game where the player controls Stanley, the main and only character in the game. The story is presented to the user via the voice of a narrator, who explains the user what he is expected to do. The story splits in several possibilities, and the narrator continues the story according to the player's choices.

All along the game, the player can interact with the elements that surround him, such as buttons or doors. In the points where a choice has to be made, the narrator always suggests the path to follow, but the player can choose differently, producing different narrations and endings to the story. The game is envisioned as a thought-provoking experience about the freedom of choice in video games, since the options given to the user are always within the intentions of the narrator. The original game has a total of six different endings, and it takes about an hour to explore all the possibilities.

Inspired by the game *The Stanley Parable*, our objective has been to design an interactive storytelling system that is able to generate a narration in real time while giving the user the possibility to decide what to do and where to go, in a co-creation process where the final result is narrated while the user is playing the game.

[1] Instituto de Tecnología del Conocimiento, Universidad Complutense de Madrid, Spain, email: {gmendez,rhervasb,pgervas}@ucm.es

[2] Facultad de Informática, Universidad Complutense de Madrid, Spain, email: {alemar07,fjulca}@ucm.es

The purpose is to tell a story, or a set of interrelated stories, to the user, trying to make him follow the path of the story told by the system, while giving him the possibility to change the way and order in which the story is told.

Therefore, our first objective is to give a human author the possibility to easily create the stories that the system has to tell, adding and changing the events of the story from outside the application. Secondly, we need to allow the system to decide how to tell the story according to the user's actions, so that the same event can be narrated in different ways. Third, we want the user to be able to change the course of the events, driving the story to a different course than that originally designed by the author, and the system must be able to narrate the new story as the user is playing it. And finally, we intend to give the user the chance to explore the setting and interlace that into the story.

The intended result is a tool that allows the author, the user and the system to co-create a narration giving the user, whose actions drive the resulting narration, as much freedom as possible to decide the course of the events.

The system has been implemented using the Unity 3D engine and the stories are set in the Computer Science School at UCM (see Figure 3). The game uses a first person perspective where the user takes the role of the main character, who needs to solve several situations in his daily life at the university. When the game starts, the voice of a narrator starts telling the story and tries to direct the user towards his first objective. The objectives of the stories are located at specific checkpoints; when the player reaches one of these points and performs a specific action, if necessary, the objective is accomplished, the narrator tells the part of the story corresponding to that objective and the next objective of the story is unlocked.

At any point of the game, the user can decide to either follow the instructions the narrator provides, so that he plays the story the game intends him to play, or he can choose to disregard these instructions and follow his own path. The first option represents the case where no unexpected actions are taken by the user, so the story the narrator tells can be carefully planned by the author. The second option is the alternate case where the user decides to ignore the story and explore other options. In this case, the designed story will not continue until the user returns to the appropriate location, but the system will have to narrate what the user is doing (see Section 4).

In order to know whether the user is following the story the narrator is telling or not, every time the player achieves an objective and unlocks the next one, the path between both objectives is calculated by the system, represented as a series of physical landmarks the user is supposed to go across. If he does not cross these points, the system accounts for the number of landmarks he is missing, so that it can figure out how much the user is deviating from the intended objective

and can modify the narration accordingly (see Section 5).

In this paper we focus on the structures that have been used to represent the stories and in the mechanisms designed to generate the narration so that a coherent and non-monotonous story can be presented to the user, irrespective of whether he follows the predefined stories or not.

2 Related Work

Storytelling systems have been developed for more than forty years. The first story telling system for which there is a record is the Novel Writer system developed by Sheldon Klein [10]. Novel Writer created murder stories within the context of a weekend party. It relied on a micro-simulation model where the behaviour of individual characters and events were governed by probabilistic rules that progressively changed the state of the simulated world (represented as a semantic network). The set of rules is highly constraining, and allows for the construction of only one very specific type of story.

TALESPIN [18], a system which told stories about the lives of simple woodland creatures, was based on planning: to create a story, a character is given a goal, and then the plan is developed to solve the goal. TALESPIN introduces character goals as triggers for action. The systems allows the possibility of having more than one problem-solving character in the story (and it introduced separate goal lists for each of them). The validity of a story is established in terms of: existence of a problem, degree of difficulty in solving the problem, and nature or level of problem solved.

Lebowitz's UNIVERSE [13] modelled the generation of scripts for a succession of TV soap opera episodes (a large cast of characters play out multiple, simultaneous, overlapping stories that never end). UNIVERSE is the first storytelling system to devote special attention to the creation of characters. Complex data structures are presented to represent characters, and a simple algorithm is proposed to fill these partly in an automatic way. But the bulk of characterization is left for the user to do by hand. It is in a first instance intended as a writer's aid, with additional hopes to later develop it into an autonomous storyteller.

The line of work initiated by TALESPIN, based on modelling the behaviour of characters, has led to a specific branch of storytellers. Characters are implemented as intelligent autonomous agents that can choose their own actions informed by their internal states (including goals and emotions) and their perception of the environment. Narrative is understood to emerge from the interaction of these characters with one another. While this guarantees coherent plots, the lack of author goals does not necessarily produce very interesting stories. However, it has been found very useful in the context of virtual environments, where the introduction of such agents injects a measure of narrative to an interactive setting.

With the use of virtual environments, the field of interactive storytelling has flourished over the last two decades, shifting the focus from actually generating a story towards real-time interactivity, engagement and user involvement in the story [1, 2, 4, 7, 15, 22].

The Virtual Storyteller [23] introduces a multi-agent approach to story creation where a specific director agent is introduced to look after plot. Each agent has its own knowledge base (representing what it knows about the world) and rules to govern its behaviour. In particular, the director agent has basic knowledge about plot structure (that it must have a beginning, a middle, and a happy end) and exercises control over agent's actions in one of three ways: environmental (introduce new characters and object), motivational (giving characters specific goals), and proscriptive (disallowing a character's intended action). The director has no prescriptive control (it cannot force characters to perform specific actions).

Façade [15] is an interactive drama where the player takes the role of a friend of Grace and Trip, a married couple who invite the player to a get-together in their apartment. A conflict arises between the couple and the player gets involved in solving the situation between Trip and Grace. The player can talk to Trip and Grace, gesture, move and use objects, and both Trip and Grace will react to the user's actions according to their psychological state. *Façade* takes advantage of a dialog based interaction that recognizes a large number of commands with which the system reacts to the user's utterances. The story is therefore based on the dialogs that take place between the player and the characters.

In [3] the authors describe an immersive interactive storytelling system where the user can play the role of a character of the novel *Madame Bovary*, written by Gustave Flaubert. The authors use a character-based approach where each character is driven by its own feelings, supported by a planner that determine the next action to be taken. However, much of the storytelling output is based on animations, rather than text or speech, which is limited to short utterances of dialogues extracted from the novel. In addition, the role of the player is limited to responding to Emma Bovary's actions, which narrow the capacity of the user to create its own story. This approach has been furtherly explored in [20], an interactive storytelling system based on Shakespeare's *Merchant of Venice*, where the focus is set on balancing the character's autonomy and the global structure of the plot.

The authors of [16] refer a social simulation game, *Prom Week*, where the aim is to explore social physics among characters to make them achieve their goals. The result is a story that reflects the interactions that have taken place among characters during the simulation. The game is based on the use of a rich set of sociocultural norms and an AI engine, *Comme il Faut*, that determines the results of the characters interactions based on these norms [17]. Although the authorial involvement of the user is noteworthy, the narrative output of the game is left for the player to interpret.

3 Description of the System

The work presented in this paper focuses on how the design of the interface between the user and the system can allow for different degrees of creative freedom and/or perception of such behaviour by the user. To achieve this, a system was designed to explore the interplay between the actual set of options open to a user, the indications provided to the user in the interface, the implicit preferences captured in the way the system communicates with the user, and the paths that the system would prefer the user to follow.

Although some of the interactive systems reviewed in Section 2 do allow the dynamic creation of new alternatives in reaction to unexpected decisions by the user, we are at present considering a system in which the set of alternatives paths open to the user is fixed before hand by the designer. These need not only be paths through a physical space, but may include sets of possible actions taken at particular points in the interaction. This set of alternatives is referred to as the *action space*. If such a system wants to provide the user with a certain sensation of creative freedom, it has to rely on different means for presenting the set of alternatives. The set of ingredients to combine are:

1. the actual set of possible paths through the action space that the user can follow

2. the set of possible paths that are marked to the user as alternatives (either signposted on the display, offered as items in system menus, or somehow marked implicitly in the environment – such as doors in the walls)
3. the set of possible paths recommended to the user by the system
4. the path chosen by the user
5. the actual path that the user's avatar is made to follow as a result of the user's choice

This set of ingredients can be combined in different ways. In basic systems for interactive storytelling, all possible paths are explicitly marked to the user (1 = 2) and the actual path taken by the avatar matches the path chosen by the user (4 = 5). In more elaborate systems, some of the possible paths are not explicitly marked to the user but the user may explore them if he decides not to follow the explicitly marked indications (1 > 2). Another variation occurs when the the path chosen by the user is disregarded by the system in favour of another one. This usually happens because the path chosen by the user would have taken the interaction away from the path desired by the designer. In such cases, a favoured option is to provide the user with a brief explanation of why the chosen action was not taken. A final point to consider is that the border between the set of possible paths that are marked to the user as alternatives (2) and the set of possible paths recommended to the user by the system (3) may become blurred in the case of subtle markings of possible paths or subtle means of recommendation. This opens up an avenue for exploring ways in which the user can be made to experience a certain creative freedom by means of crafty design of the marking and recommendations.

In the following sections we describe how we have structured the stories, so that the user can have the impression of freedom in the way he plays all the available stories, and how we generate the narration of the user's actions outside the predefined paths, so that we can add them to the story.

4 Story Structure

One of the first aspects we had to solve was to decide how to represent the stories the user can play so that, at any given moment, he has a wide range of possible actions to perform. We considered three possible structures, with minor variations, inspired by the ones described in [21]:

- linear story: all the possible actions are intertwined in a single story
- story tree: a set of stories with a common starting point from which different branches emerge and with one or more endings
- inverted story tree: a set of stories with one or more starting points that are divided in different branches and end up in one or more endings.

In the following subsections we analyze their advantages and disadvantages in order to select the most appropriate one for our purpose.

4.1 Linear Story

This is the simplest case and the easiest one to manage, since there are not relationships among different stories, which are merged into a single one (see Figure 1). It is the least costly solution in terms of implementation, and all the possibilities the user is given can be easily controlled in order to decide what the user can or cannot do.

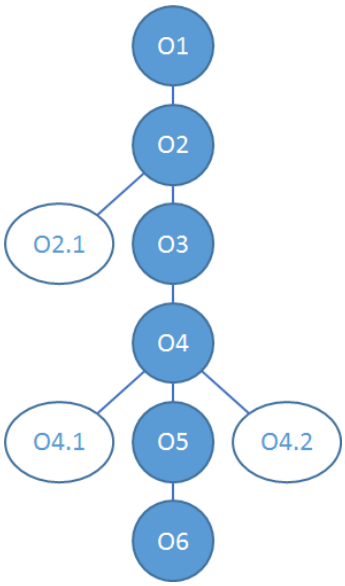


Figure 1: Linear story structure

In order to avoid monotony in the narration, minor ramifications can be added to the story, but the user cannot deviate much from the main branch of the plot. In these cases, in general, the user is led to alternative endings which are different from the one of the main story, unless he decides to return to the main course of action and lead towards the main ending.

This option was discarded due to the lack of flexibility in terms of available user choices and, subsequently, the monotony of the story the user can play.

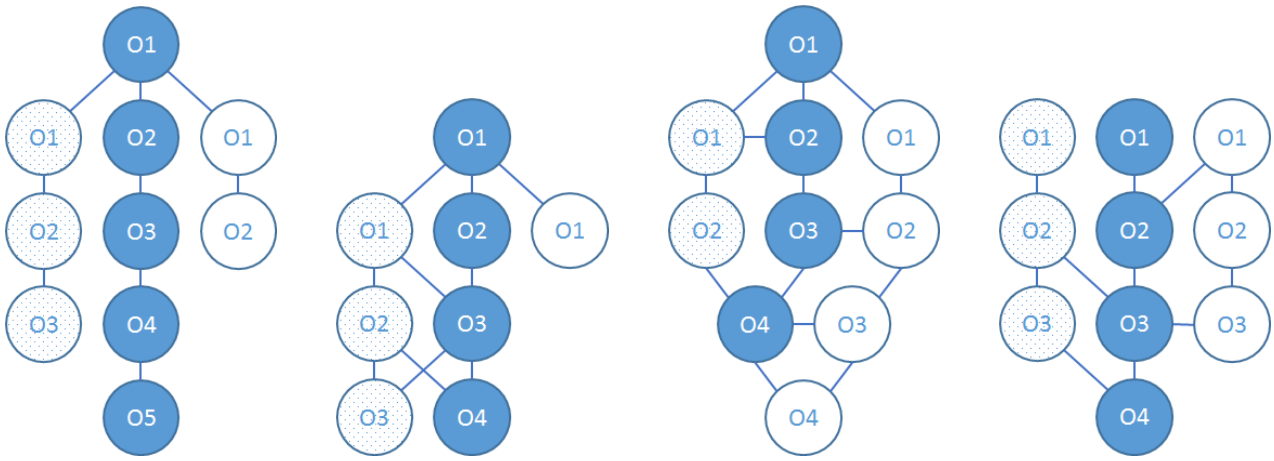
4.2 Story Tree

This is a tree-shaped structure (see Figure 2a) which starts in a single node that subsequently divides in several branches, a common approach that is used by different authors [11]. The relationship among these branches emerges from the initial node. Using this structure we can make the story evolve and end in many different ways (as shown in Figure 2b) and we can make all the stories converge into a single ending (as depicted in Figure 2c).

This structure is more complex than the previous one, and if the number of stories and their length are big enough, the implementation requires loading and unloading parts of the stories dynamically so that all the necessary elements of the stories can be stored in memory. When a user moves from one story to another one, it is necessary to generate the narration in such a way that both stories are smoothly stitched together, so that the narration does not jump abruptly from one topic to another, but is integrated in a comprehensible way instead.

4.3 Inverted Story Tree

This schema is similar to the previous one. The main difference is the initial disposition of the tree nodes, which allows the user to choose the story he would like to start playing (see Figure 2d). The main advantage of this structure is that the user can start playing any of the available stories, without having a default one the user has to start from. For example, using the previous structure, when the game starts the user is immediately pointed to play the default story, while using this structure it is not until the user starts doing something that



(a) Simple Tree (b) Branched Tree (c) Convergent Tree (d) Inverted Story Tree

Figure 2: Variations of the Story Tree and Inverted Story Tree

the systems chooses what story to start narrating depending on the user actions.

Both in the case of the story tree and the inverted story tree, if the user follows the narration without deviation the system assumes that it ends satisfactorily and chooses another one among the available stories so that the user can play all of them. On the contrary, if the user decides not to play the story the narrator is telling, a range of possible options will open, which can be abandoned or resumed at any time, giving rise to a network of interconnected stories.

In terms of implementation, trying to maintain all this stories active a the same time involves a high computational cost, because the extent of the ramification is unknown, depending on the number of available stories and on how much the user deviates for the narrator's instructions. We tested a first prototype of the system trying to play three stories in a four storey building that represents our school (see Figure 3) and the result was a slow system that produced meaningless stories. This was due, on the one hand, to the fact that a lot of structures and information had to be loaded in memory at the same time, slowing the system to a point where it was difficult to play it. On the other hand, having all the possibilities of all the stories available at the same time made the generated story be completely meaningless, since unrelated events where being mixed without further criteria, having even repeated parts because the user was passing once and again through the same spot.

In order to solve these two problems, which were quite related to one another, we decided to restrict the available predefined options the user had at the same time, keeping the possibility to deviate from these predefined paths. Each node of the tree is represented as a hidden object in Unity with which the user can collide. When this collision is effective, the system triggers an event that makes the narrator tell the user the next part of the story. Once he has finished telling it, the object disappears and is replaced by another object in a different location that represents the next objective of the story.

At any moment, only the current objectives of all the available stories are loaded, decreasing considerably the amount of information that has to be loaded in memory, and increasing notably the execution speed. In addition, since no other parts of the stories and ready to be told, we make sure that what the narrator tells is completely meaningful and that one part of a story is not told repeatedly if the user moves around the same spot for a while.

The inverted story tree has proved to be the structure that best suited all of our needs, so this was the chosen one to develop of the system. These inverted story trees are represented as external files that contain the information about the stories that the user can play, so is it fairly easy to add new stories, modify the existing ones or add new relations among the existing stories in order to enrich the user experience throughout the game.

5 Narrative Generation

As it has been previously mentioned, the narrative system has been implemented using the Unity 3D engine, where the user can choose whether to follow the narrator's instructions or not. One of our objectives was to study how the narrator can influence the user's choices in order to make him follow a predefined story while giving him the chance to follow his own path. In order to achieve this, we needed to adapt the messages that were presented to the user to make them attractive and to redirect the user to the story in case he deviated from the predefined course of action.

With this objective in mind, there were two aspects that had to be considered in order to create an attractive narrative that the user would like to follow:

- The messages the narrator offers have to be varied so that the story does not turn repetitive.
- Apart from these variations, if the narrator has to persist in its intent to make the user return to the predefined story, the tone of the messages must change progressively, which involves adding even more variability to the narration.

In order to do this, a simple algorithm has been designed to generate pseudo-random combinations of sentences with the same meaning, changing the way of phrasing them and adding a different tone depending on the situation. This way, we have managed to almost eliminate the repetition of messages throughout the story, specially when the user decides to take alternative paths to the story.

5.1 Adjusting the levels of intensity

As we have mentioned before, when the user decides not to follow the story, the narrator can use different strategies in order to convince the user to return to the predefined path. Several approaches were considered, but only one of them was selected for the first version

of the system, since adding more than one would require the design of a selection strategy or even endowing the narrator with emotional traits in order to change the way in which it talks to the user (which will be taken into account for future work).

Similarly to what is done in [24] we decided to arrange the messages in different levels of intensity, progressively making them less polite and more imperative. Therefore, it is necessary to decide how to manage these levels in order to make the narrator's messages increasingly more cutting. The messages the narrator can provide are read from an .xml file, so that the catalog of messages can be conveniently expanded in order to increase the variability of the sentences that the system offers. The narrator can provide messages using four levels of intensity.

In the lowest level, the narrator will be more polite and will keep a certain distance from the user. The narrator will consider that there has been a slight deviation from the predefined path, but it is not being completely ignored, so he will just remind the user that the path he should follow is different.

On the contrary, in the highest level, the narrator is less friendly and his comments will be more direct. At this point, the user is quite far from the place he is intended to be, so the narrator will remind him where he should go and how to get there.

In order to determine the appropriate level of intensity, the system keeps track of the user's position, and calculates a number of milestones the user must cross before arriving to the intended objective. The level of intensity of the messages will be determined according to the number of milestones that the user misses when he walks through the scene. This numbers can be appropriately calibrated in the corresponding configuration files. If the user goes back to where the narrator suggests, or if he switches to a different story, the messages the narrator provides will become friendly again.

5.2 Sentence Structure

After deciding what levels of intensity we would use, the next step was to decide how to generate the narrator's messages so that there would be enough variability that the messages that form the story would not be repeated.

In order to do this, we decided to use a combinatorial method where the sentences would be divided in different parts that would be combined in order to generate a highly enough number of combinations. Each of these parts are subsequently written in a text file attending to the previously defined levels of intensity, so the same sentence can be generated differently according to the narrator's mood.

Each sentence is currently divided into three parts, according to its position and purpose in the sentence:

- Opening: it is optional and it mainly consists in a way of referring to the user; it is used at the beginning of the message. *"It seems that"* or *"Insisting on his impulsive attitude"* are some possible openings.
- Body: it contains the main message of the narrator and has meaning by itself, like *"Gabe decided to go to the canteen"*.
- Closing: it is optional and contains an additional comment to the narrator's message; it is used at the end of the message. *"Disregarding my directions"* or *"without considering the consequences"* are some possible closings.

Combining these three parts a considerable amount of messages can be generated by the narrator: *"It seems that Gabe decided to go to the canteen disregarding my directions"*, *"Gabe decided to go to the canteen without considering the consequences"*, etc.

5.3 Message Generation

Using the previously described sentence division, we intend to generate both simple and compound sentences. If the pieces we use to build the sentences belong to the same intensity level, we will be sure that the generated sentences will be coherent, as long as we make use of the following rules.

In the case of simple sentences, we have to specify in the configuration file that they end with a period. As for the openings and closings, we can use assertions, exclamations and questions, such as *"Hey!"* or *"You know what?"* and we also have to specify it in the configuration file so that they can be linked appropriately with the rest of the sentence. In addition, in the closings we need to indicate whether they can link directly with the body of the sentence or not, so that they can be correctly written, using a comma or starting them with capital letters.

Compound sentences involve more restrictions about the way in which they can be linked to other sentences. We use two different types of sentence bodies that can be combined with openings and closings, depending on whether they need a nexus with them or not. As in the case of the simple sentences, we need to consider whether the openings and closings are separated by commas or not, so that they can be correctly written (e.g. *"In a show of rebellion,..."* or *"..., as his father used to say."*). The only difference between openings and closings is the place where they link with the sentence body.

In order to allow the insertion of elements of the story into the sentence, the rules to generate the sentence bodies are a little different. A sentence body can be used by itself or in conjunction with openings and/or closings, and it can be completed with context dependent actions (e.g. *"In a show of rebellion, Gabe decided to turn left"*).

The main variations in this case depend on the user's location and the objectives he still has to achieve in the story. If the user prefers to explore the site instead of following the story, the narrator can provide him with information regarding where he is, while other times he can try to lead the user to the next objective. In both cases, the sentence body must end with a verb, so that the sentence can be completed correctly adding information about the place or objective where the system wants the player to go.

The following example summarizes the previous rules to generate compound sentences. Gabe is in a classroom and has decided to disregard the narrator and explore the upper floor of the building. The narrator, annoyed, tries to lead Gabe to the canteen, where his friends are working on an assignment. The sentence body the narrator is going to use is *"I thought I had told you that you had to go to..."*.

Before processing the sentence, the narrator retrieves the current objective in order to integrate it into the sentence. The narrator decides whether to use an opening and/or closing and, in case it does, checks whether it is necessary to use punctuation marks or not. As a result, the narrator decides he has to use *"You may have not understood me well, but"* as an opening and *"Are you sure you want to continue with this nonsense?"* as a closing. The system then selects the appropriate capitalization for the three parts of the sentence and builds the final message: *"Gabe, you may have not understood me well, but I thought I had told you that you had to go to the canteen with your friends. Are you sure you want to continue with this nonsense?"*.

6 Story Generation Example: *Exam Season*

As a complete example of how the system works, we present a sample execution containing two different stories. In the first one, our

main character, Gabe, must go to a specific classroom to take an important exam that he had forgotten he had to take. In the second story, which could be played before or after the first one, Gabe must find a specific professor to ask him to review a previous exam. In both cases, the player would have to interact with different characters and fulfill some partial objectives for the stories to advance.

In order to understand the example, we will first describe the data structures of the stories and the narrative generation, and then we will show an execution where the user is playing one of the stories and jumps from that story to the other before finishing the first one.

6.1 Structure of the Stories

As previously explained in Section 4, stories are organized as a series of objectives that must be fulfilled for the user being able to continue the story. Therefore, objectives can be seen as milestones that mark the advancement of the user through the story. In order to understand the working example, we present here the list of objectives for each of the stories.

List of objectives in Story 1 (Gabe must go to classroom 3 to take an exam):

s1o1 Gabe sees a lot of students in front of classroom 3 and decides to approach them to see what is happening.
s1o2 Gabe realizes that he had an exam today but he had forgotten about it.
s1o3 Gabe decides to go the expending machine to look for something to calm his nerves.
s1o4 Already at the machine, Gabe realizes that he has no money on him.
s1o5 Gabe decides to go to the canteen to see if he finds a friend who could lend him some money.
s1o6 Gabe sees Michael in the canteen, who owes him a favor.
s1o7 Description of Michael, who is unmistakable because of his sun glasses and orange shirt.
s1o8 Michael does not have enough money for Gabe. Gabe decides to come back to classroom 3.
s1o9 Gabe has to hurry because the teacher is already there.
s1o10 Gabe arrives on time and takes the exam.
s1o11 After the exam Gabe feels better and starts to count the days until holidays.

List of objectives in Story 2 (Gabe wants to find Professor Smith to ask him to review a previous exam):

s2o1 Gabe remembers that he has to talk with Professor Smith to ask him to review an exam.
s2o2 Gabe thinks that he can find him in his office, so he goes there.
s2o3 Gabe meets Macarena on his way. He decides to ask her about the professor.
s2o4 She tells him that Smith was with her in a meeting in a different office.
s2o5 Gabe goes to this other office to look for him.
s2o6 Professor Smith tells Gabe that he does not have time to help him. Gabe leaves a bit annoyed.

We will later see how the user can jump from one story to the other at any time. As the user has complete freedom of movement in the 3D environment, he can decide to move from the current objective in one story and find a different one in other story (remember that objectives are triggered by the user when he goes over certain landmarks in the environment). The only restriction is that the system "activates" the next objective of a story only when the previous one has been fulfilled, so the user can only move from the current objective in a story to the current objective in the other story. We will later see an example of this and what are the consequences from the point of view of the narration.

6.2 Narrative Generation Data

From the point of view of the narrative generation, there are two different kinds of messages used by the narrator: objective messages and control messages. On the one hand, each objective has an utterance associated that explains what is happening or what is the next objective (for example *"Gabe wondered why there were so many students in front of classroom 3. He decided to check what was happening."*). Those are the objective messages. On the other hand, the narrator also speaks when the user is not following a path leading to fulfilling the next objective. In this case, the generated messages are intended to remind the user of what he should be doing, and they are expressed in different levels of intensity depending on how far the next objective is. These are the control messages. As the generation of control messages is more creative, we will explain them deeper for the present example.

According to what was explained in Section 5.1, the current example includes four different levels of intensity in the narration. Level 1 (lowest level) represents utterances with a slight annoyance from the narrator (like for example *"Gabe felt curious and decided to deviate from his path"*), whereas in level 4 (highest level) the narrator transmits to be very angry with the user (for example *"You know what? I desist. If you don't want to follow my advice, it's up to you."*). A total of 18 body sentence templates, 10 openings and 9 closings were created for these control messages. Remember that utterances by the narrator are then formed by a body sentence template completed according to the current state of the story, optionally accompanied by an opening and/or closing.

6.3 Sample Story

Gabe, our main character, appears in the entrance hall of the building. At the beginning none of the stories is active, and one or the other would activate when Gabe passes through the landmark of the first objective of one of them. Until then, the user is allowed to explore the 3D environment as he wishes. The narrator is initially off and would start the narration when an objective is activated.

In addition, there is not a prefixed order for the two stories to occur. The user can start and finish any of them before starting the other, or interweave them as he wishes. In this example the user will start with the first story, but during it he will decide to explore the second story, finally coming back to the first one when the second is finished. Although these two stories are reasonably simple in order to have an example that is easy to follow, we can imagine an execution of the system where there are multiple stories and the user "finds" them while exploring the course of other stories.

While exploring the building, Gabe enters one of the corridors in the ground floor, triggering the first objective of story 1 (*s1o1*). He sees a group of students in front of classroom 3 and decides to take a look (see Figure 3). When he arrives, he realizes he had forgotten about an important exam (*s1o2*). The narrator then suggests that Gabe is very nervous and should go to the vending machine to grab a drink (*s1o3*).

Note that in objective *s1o2* the player is not told to go anywhere, and therefore objectives *s1o2* and *s1o3* occur in the same spatial point

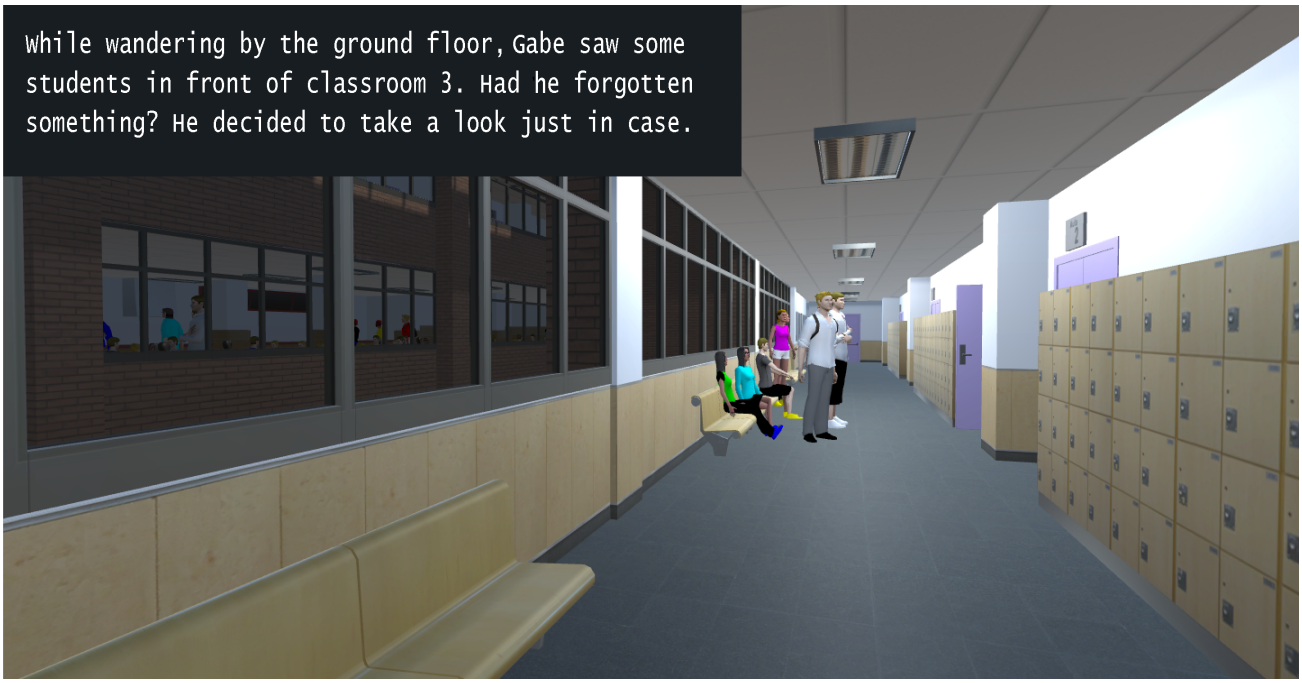


Figure 3: First objective of story 1

and could be considered as only one objective. However, the narration is better managed if they are considered as two different objectives occurring in the same place but with different narration utterances. This will happen again in some other points of the stories.

When Gabe arrives to the vending machine, the narrator informs him that he has no money to pay for the drink (*s1o4*), and that he should go to the canteen to see if he finds someone who could lend him some coins (*s1o5*).

At this point of the execution, the user decides that he will not obey the narrator anymore. Instead of going to the canteen, as he was suggested, the user makes Gabe turn around and go upstairs to the first floor. The narrator detects this deviation from the expected path in story 1, and activates the use of control messages with a low intensity. Therefore, the narrator will suggest Gabe to go back to the expected path with sentences like *"Being aware that this was not the way to go, Gabe decided to explore other places without knowing the possible consequences"* or *"It seems that Gabe decided to go in the opposite direction"*. Openings and closings have been underlined for easier recognition.

As Gabe explores the first floor the system detects that he is getting away more and more from his current objective (*s1o5* - going to the canteen). As a consequence, the narrator gets more serious and the intensity of control messages is increased. If it reached the maximum level, the narrator would say things like *"You think that being the player gives you the right to ignore me. For the last time, you are going to the canteen!"*.

While exploring the first floor, Gabe enters classroom 5 where the first objective of story 2 is located (*s2o1*) and he remembers that he has to talk with Professor Smith to review a previous exam.

At this point, the system stores the current objective of story 1, starts the execution of story 2 by setting the current objective to *s2o1*, and the narrator "forgets" his anger for previous deviations.

The user then continues story 2 by making Gabe go to Professor Smith's office (*s2o2*), meeting Macarena in his way (*s2o3* and *s2o4*), going to the office Macarena mentioned (*s2o5*), and finally finding him (*s2o6*).

After Professor Smith tells Gabe that he has no time to attend him and the narrator explains that Gabe is a bit annoyed, the second story is considered to have finished.

Then the system checks if there are other unfinished stories (story 1 in this case), sets the current objective as *s1o5*, and the narrator activates again the control messages if the user is deviating from the path to the current objective. The user can then finish story 1, and once he does the system detects that there are no more available stories and informs the user accordingly.

7 Conclusions and Future Work

This paper describes a system that has been designed to allow a human author, the system and a human player to co-create stories in a given environment, currently a 3D reproduction of the Computer Science School at UCM. The focus has been set on those aspects of the system that have allowed us to create the stories: the structure to represent the stories, which allowed us to give the user the control over its development, for which an inverted tree structure has been used; the mechanism to generate different ways to narrate an event to the user; and the technique devised to influence the user in order to make him follow the narrator's instructions.

The objective of the system is twofold: first, as a computer game, it aims at providing the user with an engaging experience playing the stories that a human author has created, as well as telling to the user these stories while he plays; second, as a computational creativity tool, it intends to allow the user to influence and modify the system behaviour in order to create stories which are different from the original ones.

Although both objectives have been achieved, we have been able to identify several issues that allow for future lines of work in order to improve the system behaviour.

Even though the variability of the generated story is notable, it requires the intervention of the human author to elaborate the open-

ings, closings and bodies of the sentences. For this reason, a more generative way to create the stories, based on NLG techniques, are to be considered, as long as they meet one of the basic requirements of the system: that it can generate the stories in real time, in order to be able to narrate the actions of the player while he is playing. This way, we will be able to include more elements into the story and generate its content in a more dynamic way.

Another aspect that has to be furtherly explored is the behaviour of the narrator, giving it the chance to use other ways to persuade the user to follow its instructions. To achieve a successful behaviour, an extensive user testing is required in order to study the user's preferences and reactions towards the narrator's messages.

In order to relieve the human author from having to create the contents of the initial stories, we are aiming at using other storytelling systems for this task. Our research group has already developed several systems of this kind (see, for example [8, 12, 14, 19]) and we are currently working in an intermediate representation [5, 6] that enables different storytelling systems to exchange information that allows us to combine them to generate different aspects of a story.

In this same line, we are evaluating the possibility to feed the system with the new stories, so that they can be evaluated an incorporated into the game so that other players can play the new stories that have been considered acceptable using different quality metrics [9].

Finally, one of the aspects that has not been fully explored in the current version of the game are the dialogs with other characters, in a similar way as it is done in *Façade* [15] and *Prom Week* [16], which will give the user the impression of having more influence over the resulting story.

ACKNOWLEDGEMENTS

This project has been partially supported by projects WHIM (611560) and PROSECCO (600653) funded by the European Commission, Framework Program 7, the ICT theme, and the Future Emerging Technologies FET program, and by project IDiLyCo (MINECO/FEDER TIN2015-66655-R), funded by the Spanish Ministry of Economy and the European Regional Development Fund.

REFERENCES

[1] Ruth Aylett, 'Narrative in virtual environments-towards emergent narrative', in *Proceedings of the AAAI fall symposium on narrative intelligence*, pp. 83–86, (1999).

[2] Marc Cavazza, Ruth Aylett, Kerstin Dautenhahn, Clive Fencott, and Fred Charles, 'Interactive storytelling in virtual environments: building the "Holodeck"', in *Proceedings of VSMM 2000*, pp. 4–6, (2000).

[3] Marc Cavazza, Jean-Luc Lugrin, David Pizzi, and Fred Charles, 'Madame Bovary on the Holodeck: immersive interactive storytelling', in *Proceedings of the 15th ACM International Conference on Multimedia*, MM '07, pp. 651–660, New York, NY, USA, (2007). ACM.

[4] Fred Charles, Steven J Mead, and Marc Cavazza, 'Character-driven story generation in interactive storytelling', in *Virtual Systems and Multimedia, 2001. Proceedings. Seventh International Conference on*, pp. 609–615. IEEE, (2001).

[5] Eugenio Concepción, Pablo Gervás, and Gonzalo Méndez, 'Mining knowledge in storytelling systems for narrative generation', in *CC-NLG: Computational Creativity in Natural Language Generation (@INLG2016)*, Edimburgh, UK, (09/2016 2016). ACL Anthology, ACL Anthology.

[6] Eugenio Concepción, Pablo Gervás, Gonzalo Méndez, and Carlos León, 'Using CNL for knowledge elicitation and exchange across story generation systems', in *5th Workshop on Controlled Natural Language (CNL 2016)*, Aberdeen, Scotland, (07/2016 2016). Springer, Springer.

[7] Rui Figueiredo, Joao Dias, Ana Paiva, Ruth Aylett, and Sandy Louchart, 'Shaping emergent narratives for a pedagogical application', *NILE-Narrative and Interactive Learning Environments*, 27–36, (2006).

[8] Pablo Gervás, 'Reviewing Propp's story generation procedure in the light of computational creativity', in *AISB Symposium on Computational Creativity, AISB-2014, April 1-4 2014*, Goldsmiths, London, UK, (04/2014 2014).

[9] Pablo Gervás and Carlos León, 'Integrating purpose and revision into a computational model of literary generation', in *Creativity and Universality in Language*, eds., Mirko Degli Esposti, Eduardo Altmann, and Francois Pachet, Lecture Notes in Morphogenesis, Springer, (2016).

[10] Sheldon Klein, John F. Aeschliman, DavidF. Balsiger, Steven L. Converse, Claudine Court, Mark Foster, Robin Lao, John D. Oakley, and Joel Smith, 'Automatic novel writing: A status report', Technical Report 186, Computer Science Dept., The University of Wisconsin, (December 1973).

[11] Hartmut Koenitz and Kun-Ju Chen, 'Genres, structures and strategies in interactive digital narratives: analyzing a body of works created in ASAPS', in *Proceedings of the 5th International Conference on Interactive Digital Storytelling*, ICIDS'12, pp. 84–95, Berlin, Heidelberg, (2012). Springer-Verlag.

[12] Iván M. Laclaustra, José L. Ledesma, Gonzalo Méndez, and Pablo Gervás, 'Kill the dragon and rescue the princess: designing a plan-based multi-agent story generator', in *5th International Conference on Computational Creativity, ICCC 2014 (Late Breaking paper)*, Ljubljana, Slovenia, (06/2014 2014).

[13] M. Lebowitz, 'Story-telling as planning and learning', *Poetics*, **14**, 483–502, (1985).

[14] Carlos León and Pablo Gervás, 'Creativity in story generation from the ground up: non-deterministic simulation driven by narrative', in *5th International Conference on Computational Creativity, ICCC 2014*, Ljubljana, Slovenia, (06/2014 2014).

[15] Michael Mateas and Andrew Stern, 'Façade: an experiment in building a fully-realized interactive drama', in *Game developers conference*, volume 2, (2003).

[16] Josh Mccoy, Mike Treanor, Ben Samuel, Aaron A Reed, Michael Mateas, and Noah Wardrip-fruin, 'Prom week: designing past the game/story dilemma', in *Proceedings of Foundations of Digital Games (FDG*, (2013).

[17] Josh McCoy, Mike Treanor, Ben Samuel, Noah Wardrip-Fruin, and Michael Mateas, 'Comme il faut: A system for authoring playable social models', in *Proceedings of the Seventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, AIIDE'11, pp. 158–163. AAAI Press, (2011).

[18] James R. Meehan, 'Tale-spin, an interactive program that writes stories', in *Proc. of the Fifth International Joint Conference on Artificial Intelligence*, pp. 91–98, (1977).

[19] Gonzalo Méndez, Pablo Gervás, and Carlos León, *On the Use of Character Affinities for Story Plot Generation*, volume 416 of *Advances in Intelligent Systems and Computing*, chapter 15, 211–225, Springer, 02/2016 2016.

[20] Julie Porteous, Marc Cavazza, and Fred Charles, 'Narrative generation through characters' point of view', in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1 - Volume 1*, AAMAS '10, pp. 1297–1304, Richland, SC, (2010). International Foundation for Autonomous Agents and Multiagent Systems.

[21] Mark O Riedl and Robert Michael Young, 'From linear story generation to branching story graphs', *IEEE Computer Graphics and Applications*, **26**(3), 23–31, (2006).

[22] Nicolas Szilas, 'A new approach to interactive drama: from intelligent characters to an intelligent virtual narrator', in *Proceedings of the AAAI Spring Symposium on AI and Interactive Entertainment*, pp. 72–76, (2001).

[23] Mariet Theune, Er Faas, Anton Nijholt, and Dirk Heylen, 'The virtual storyteller: story creation by intelligent agents', in *Proceedings of the Technologies for Interactive Digital Storytelling and Entertainment (TIDSE) Conference*, pp. 204–215, (2003).

[24] Davey Wreden. The Stanley Parable, 2013. [accessed 1-February-2017; `https://www.stanleyparable.com/`].