Adding Dialogue to the Virtual Storyteller

Erwin Bergervoet
e.j.bergervoet@student.utwente.nl

ABSTRACT
The Virtual Storyteller is an interactive storytelling system in development at the University of Twente. Currently, this system does not include dialogue between agents. For creating interesting stories, dialogue is a very important aspect. This paper tries to find a way of implementing dialogue in the Virtual Storyteller by looking into other interactive storytelling systems such as FearNot! and Façade to see how they handle dialogue. Experiments are done to implement dialogue in the Virtual Storyteller and based on the outcome of these experiments, suggestions are done for a future implementation.

Keywords
Virtual Storyteller, Interactive Storytelling, Dialogue, Emergent Narrative

1. INTRODUCTION

1.1 Storytelling in Commercial Games
Interactive storytelling is a reasonably underdeveloped field of interactive entertainment. Whereas game development companies have invested largely in improving the audiovisual qualities of video games, the story in the game is still as straightforward as it was decades ago. This limits video games in their appeal to the mass market, because they are not capable of coping with many of the topics and themes of human relationships, as was noted by Mateas and Stern in their paper on the ground breaking Façade project [MS03], one of the first big achievements in the field of interactive storytelling.

Interactive entertainment, especially in the form of commercial video games, is booming business. Over the past few decades, the gaming industry grew from the garages of enthusiasts, to companies employing thousands of people all over the world, with revenues exceeding a billion dollars. But still, most high budget games are mainly played by adolescent males. Other mostly cheaper, smaller and simpler games are targeted at the more occasional, mostly older gamer. Compared to other mainstream media such as movies, there are not many games which attract both groups at the same time.

A possible way to get more mass market appeal for video games, is further developing techniques for interactive storytelling. With interactive storytelling video games can become real interactive stories which truly use the storytelling potential of an interactive medium. Currently, most stories in video games are completely linear. The story serves as a framework around the actual action gameplay which is mostly the game challenging the player, trying to prevent him from advancing, instead of telling a story.

With further development of interactive storytelling, such a framework is not needed, because the story becomes the actual game itself. With such a game, challenging the player, for example by having enemies attack him, is not needed anymore. This challenge is the focus point of most games nowadays and makes video games inaccessible for many. When the story is the actual game itself, the user's actions should really influence the story and the characters in the game world should react in appropriate ways to those actions. That alone should deliver the motivation to continue playing.

There are some commercial games available already which do not focus on defeating enemies and have a more story focused approach. The most well-known example is The Sims [Sim08], created by Maxis. The Sims lets players run a household and give orders to characters (called sims) in order to fulfill their daily needs and eventually make a career and become rich. There is no set goal for the game and there is no way to really fail or succeed. The game is about what happens to the sims and how they manage in unexpected situations. For example, their house can burn down, a sim can get abducted by aliens or they can fall in love with their neighbors. While sims can die, this does not actually end the game, because new sims can be born or can come to live in the household, continuing the ongoing story.

The sims can be seen as autonomous agents. When the player does not interact with his sims, they will decide themselves what to do, based on their current needs. Those needs are modeled as a set of numbers, and objects in the environment have the property of changing those values. For example, Sims have the property hygiene with a certain value (the lower, the more in need of hygiene the sim is) and the shower has the property to improve hygiene. The game does not tell the story from a certain perspective, but shows everything that happens around the house to the player. The characters in the game do not communicate in any real language, but speak a made up language while icons represent their general subject.

Another quite different example of interactive storytelling in games is Fahrenheit [Fah05], a game which actually tells a pretty structured story with a set beginning and an end, but lets the player bend the story a bit along the way. While Fahrenheit contains some classical challenges where a sequence of buttons needs to be pressed in time, those are mainly there to increase the suspense, while the real motivation to keep playing is the story and how it evolves. The fact that the player can influence the story, makes it more immersive and interesting, even though the influence of the player is quite limited in the end.

Between The Sims and Fahrenheit, there is still a large gap to fill. The Sims is merely a simulation with agents doing their thing and does not have a strong narrative. Fahrenheit does have the latter, but offers limited freedom in changing the course of events. Ultimately, the ideal game would offer the freedom of The Sims, while keeping the narrative tension of a game such as Fahrenheit, both offering freedom to the player and a greatly told story. In order for this to be possible, lots of research and development still needs to be done in the field of interactive storytelling.

1.2 Motivation
The Virtual Storyteller [TRA04], developed at the University of Twente, is a research project in the field of interactive storytelling. Where many interactive storytelling projects focus on the player influencing the story, the Virtual Storyteller does not include player interactivity yet. But, while most interactive storytelling projects limit themselves to small, controlled environments, the Virtual Storyteller offers a framework which

can operate on many different story domains. Such a story domain can define one of those small, controlled environments, but the underlying system does not know in advance what the environment will look like.

The Virtual Storyteller uses the concept of emergent narrative [ALD05], where autonomous agents are pursuing their own goals within the story world. The idea is that by giving the agents conflicting goals, an interesting story will emerge from the story world automatically.

A difference between the Virtual Storyteller and similar projects, is currently the absence of dialogue in the Virtual Storyteller. For now, the autonomous agents inhabiting the story world can not engage in a conversation with one another, but can only perform physical actions which directly change the state of the world. This is quite remarkable, because in most other interactive storytelling projects dialogue is an important part of the story, if not the most important. Also for stories in general, dialogue is very important and one of the most common ways to change the relation between characters.

This paper tries to find an appropriate way to implement dialogues into the Virtual Storyteller application. Therefore, dialogue techniques of other interactive storytelling projects are analyzed. In FearNot! [ALD05] and Façade [MS03], dialogue is the most common action and physical interactions with the environment rather are the exception. Implementing dialogue in a storytelling system which relies more on physical interaction, might require a different approach.

The Virtual Storyteller consists of two separate levels, being the simulation and the presentation level [TFN02]. The simulation level is where the plot is generated, based on actions of characters upon the virtual environment. In the presentation level, these plot actions are represented from a certain point of view and converted into human readable text, or speech eventually. In the current version of the Virtual Storyteller, the presentation layer is not yet fully implemented.

This paper will solely focus on the simulation level of the Virtual Storyteller, which is currently the most developed aspect of the system. To implement dialogue into the Virtual Storyteller, this should first be done on the level of the simulation, the actions which take place in the world itself, before it can be converted into the appropriate text by the presenter. Dialogue on this level of the story generation process should not be seen as an utterance of a line of text, but instead as a schematic representation of the content of that utterance. Only in the presentation level, this might be converted in a line of text or speech utterance which contains the information from the schematic representation.

1.3 Research Question

The research question is as follows:

RQ: What is the most appropriate way to incorporate dialogue in the Virtual Storyteller?

This question will be answered by answering the following subquestions:

SQ1: What type of dialog acts are expected in the domain of the Virtual Storyteller?

SQ2: What techniques are used for dialogue by other interactive storytelling applications on the market?

SQ3: What are the implications of implementing dialogue for the Virtual Storyteller?

1.4 Methodology

This research will be a combination of a literature study, where current implementations of dialogue in interactive storytelling systems are analyzed and compared, and experimentation with the Virtual Storyteller to see what approaches to dialogue might work and what problems still remain.

First, the current inner workings of the Virtual Storyteller will be elaborated further, giving an idea of its functionality and possibilities. Then, a very broad overview is given of dialogue in general, pointing out some important dialogue acts which are suitable for stories in general.

Next, current systems for interactive storytelling are evaluated, especially looking at the way dialogue is implemented in those systems. After that, a practical experiment is done trying to implement dialogue in the most straight-forward way in the Virtual Storyteller. Based on that, suggestions for future implementations of dialogue are made.

2. THE VIRTUAL STORYTELLER

The Simulation part of the Virtual Storyteller uses three different agents to generate the plot: the World Agent, the Plot Agent and various instances of Character Agents, as can be seen in Figure 1: The structure of the Virtual Storyteller. The World Agent handles the operation of actions on the world and transmits the outcome of the actions and events back to the Plot Agent. The Character Agents perceive the current state of the world via the Plot Agent, and then decide for a certain action which is then transmitted to the Plot Agent which sends it forward to the World Agent. While the Plot Agent is more or less an intermediary between the World and Character Agents, there can also be some mutual interaction between the Character Agents and the Plot Agent. The Plot Agent can decide what characters see of the world and in that way influence their plans. The Plot Agent generates the information necessary for the story and forwards this to the Presentation part of the system, which changes a representation of the actions in the world to an human readable story. This part of the system is outside the scope of this paper.

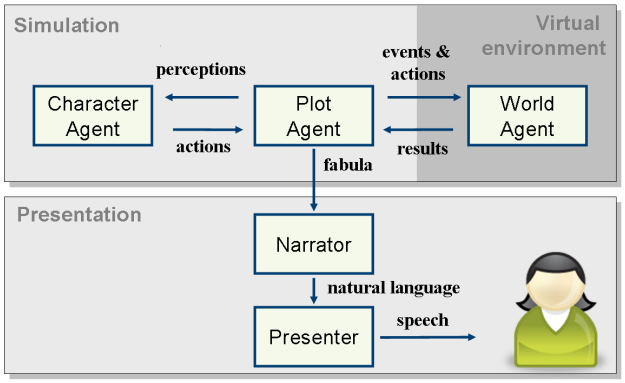

Figure 1: The structure of the Virtual Storyteller

Each domain within the Virtual Storyteller has an ontology, describing for instance the objects, actions and goals in a given world. Character Agents choose a certain action by looking at their current goals and deciding what actions lead to this goal. Each action has a set of preconditions, which have to be true in order to be able to perform the action. The actions also have a set of effects, being the changes made to the story world by performing the action. Objects and other agents can serve as a means necessary to perform a certain action. For example, to kill another agent with a knife, he needs to be in the same spot

as the agent performing the action, and the agent performing the action needs to have a knife.

Each Character Agent has a goal, which is selected from a set of possible goals for which the preconditions are true for a given agent. A goal also has a success condition. The goal is completed when this success condition is true. Agents having a certain goal will plan actions which lead to the success condition being met.

2.1 Possibilities for Dialogue Acts in the Virtual Storyteller

The field of dialogue management, speech acts and dialogue acts is a very broad field of research starting from a philosophical point of view in the 1960's, lead by Searle's famous essay about Speech Acts [Sea69]. A large focus of the dialogue management field is on the interpretation of spoken language and the subtleties therein. The ambiguity of spoken messages is a big focus point in the research of speech acts, but is not so relevant in the communication between agents in an artificial language, as stated by Traum [Tra99] However, the actions needed for the agents to perform will be roughly the same as those occurring in natural interaction.

There are many ways to categorize speech acts, or dialogue acts, from very detailed including all the subtleties, to very generic. A well known categorization of dialogue acts is the DAMSL dialogue act coding scheme, consisting of a set of forward communicative functions and backward communicative functions. Considering the forward-looking functions, the main categories of the DAMSL scheme are statement, influencing-addressee-further-action, committing-speaker-further-action, conventional, explicit-performative and exclamation [CA97]. Most of these dialogue acts are done to influence the other agent in the conversation. The speaker either wants the addressee to know or believe something (statement), wants him to do something for him (influencing-addressee-further-action) or tells the other he will do something (speaker-further-action). The backward looking functions are responses directly related to a forward looking function, such as agreements, an answer to a question or signals of understanding.

For a story environment, the influencing of other characters is a very important aspect in the dialogue. Convincing another agent to do something for you, promising them rewards when they do so, such constructions are frequently seen in various stories and should eventually be relevant for the Virtual Storyteller.

Currently, the Virtual Storyteller works with ontologies based around fairy tales and pirates. Characters in such stories are expected to be flat or at least emotionally not very complex. Therefore, having conversations about the personal emotions of the characters, are not a priority. Instead, dialogue used as a means of knowledge transmission (statement) might be a useful starting point of the implementation. Utterances which try to influence the behavior of other characters in a more complex way, for example by transmitting false information, could be added later.

The information transmitted through the dialogue can be anything about the story world the addressed character does or does not know about, or about events which took place in the story world in the past. The information can also incorporate the internal state of the speaking character, such as his opinions and beliefs about certain subjects.

Before information can be transmitted through dialogues, a conversation needs to be set up. For this purpose, actions such as greeting other characters to engage in conversations, and saying goodbye to close conversations, could be needed. Some conversations also start without an explicit greeting, for example by making eye contact. These are the so called conventional forward communicative functions from the DAMSL scheme. By asking questions, characters can request information from the other character. Acknowledgements and denials can be used to confirm or deny information uttered by the other character.

The question remains how to implement such dialogue acts. One option is to implement them in a similar way as physical actions. Such physical actions have a set of preconditions which need to be true in order for the action to be possible. Whether the agent really performs the action, depends on the goals of the agent and whether the action leads to such a goal. Take for example the physical action OpenHatch. This action requires that a hatch is closed and the agent is at the location of the hatch, in order to be possible. Whether the agent executes the action, depends on whether opening the hatch leads to the agents current goal.

One can imagine a speech act similar to OpenHatch called OpenSesame, which has the same preconditions and results as OpenHatch, but is represented as an utterance by an agent on the narrative level. It is a kind of speech act which is closely related to a physical action. The same is true for actions such as insulting, which differ from a physical slap in the face merely on the narrative layer, whilst both can have the same effect on the world.

But are such actions sufficient for having more complex dialogue, in which agents share their knowledge of the world? Can such actions be implemented by having a set of preconditions and a general goal, and will that lead to interesting and dynamic dialogues? Or should there be a separate system for handling dialogue, which allows agents to generate any kind of speech acts based on their knowledge of the world?

Before answering these questions for the Virtual Storyteller, other interactive storytelling systems currently available are analyzed.

3. CURRENT SYSTEMS

This section will look into several of the current interactive storytelling systems. The different approaches used for dialogue in these systems will be discussed based on how they might fit into the Virtual Storyteller.

3.1 FearNot!

FearNot! [ALD05] is a interactive storytelling application used to teach children about bullying. Just as the Virtual Storyteller, FearNot! uses the concept of emergent narrative to generate stories. The player acts as a imaginary friend of a girl or a boy who gets bullied as a daily routine, and has to give him or her advice on how to act the next time they get bullied. FearNot! relies heavily on dialogue. Giving advice to the bullying victim is done through a natural language based text interface and also the bullying itself is for a large part verbally. Nonetheless, characters can interact with their virtual environment and for example push other characters or drop books on the ground.

FearNot! uses a fairly constrained environment for interactive storytelling and even though there is interactivity, it is a repetition of the same two scenes over and over again. First, there is a scene where the person gets bullied, followed by a scene where the player gives this person advice. How these scenes work out is different based on the character's internal motivations influenced by player input, but every scene follows

the same kind of structure and has the characters continuously engaged in some sort of dialogue with each other.

In FearNot!, all dialogue acts are implemented as normal actions, on par with actions such as 'push'. An action selection mechanism [ALD06] chooses the most appropriate action at a certain moment, whether it is an utterance by a character, or a physical action.

While they are both represented as actions, a physical action could also be seen as a dialogue act. An act of pushing someone can imply the same as saying "I do not like you" to the character. The pushing itself has no influence on the world, other than the internal state of the characters, just as with dialogue acts. Characters do not get wounded or hospitalized, resulting in a major turn in the storyline. Instead, the actions which are present (eg. books falling off tables, characters pushing each other) only affect the internal state of characters on a longer term. The change in internal state of the character might lead to a major turn in the storyline eventually, but this could also be achieved by a speech act having the same result to the internal state.

Therefore, scenes in FearNot! can be seen as an ongoing dialogue between characters until either of the characters decides to go away. This is different to the Virtual Storyteller, where characters are moving freely through the story world and are not sure the other character hear what they are saying. In FearNot!, the characters relevant for the story, always pay attention to each others utterances. For the Virtual Storyteller, a mechanism for agents to engage in conversations with each other, might be required, because not every agent is part of a certain conversation all the time.

3.2 Friends

Another interactive storytelling system is the one developed by Cavazza et al. [CCM02], which uses the characters and environments from the popular sitcom Friends to generate interactive stories. They use Hierarchical Task Networks (HTN) planning to represent the different options characters have to reach their preset goals.

In the original system, dialogue was implemented as an action in the HTN, which usually is used to request some kind of information or transmit a message. Dialogue acts can, just as normal actions, have preconditions. For example, the character which is spoken to, needs to be listening before the dialogue act can be done.

In a later study, Cavazza et al. looked specifically into the incorporation of more extensive dialogue in their interactive storytelling system [CC05]. They especially pay attention to transmitting information implicitly, instead of explicitly as is the case with the dialogue acts which were already implemented. For example, when asking if someone wants to come to a party, instead of directly inviting them (explicitly), a character could also just tell them there is a party (implicitly). The different sentences will be generated based on different properties for a sentence, such as the addressee, the actor, the type of event and the affinity of the addressee with the subject matter (positive or negative).

While this type of representation of dialogue allows for quite complicated and subtle dialogues between virtual characters, which open up many opportunities for interesting stories, it also requires a lot of information to go with the dialogue act from a plot level. While eventually the approach of Cavazza et al. is interesting and offers a good framework for different types of dialogue acts, for the current version of the Virtual Storyteller a more straight-forward approach should be sufficient.

3.3 Façade

Another interactive storytelling system which, like FearNot!, relies heavily on dialogue is Façade. Contrary to the FearNot! system as well as the Virtual Storyteller, Façade does not use the concept of emergent narrative, in which the characters have certain (conflicting) goals they try to achieve. Instead, Façade uses a set of story beats which try to get the flow of the story going. These story beats which occur are chosen based on the interactions of the player with the virtual characters.

Just as with FearNot!, characters in Façade are almost constantly engaged in dialogue. In each beat, the characters have a set of goals including the dialogue they have to speak during that beat. Also, each beat has a set of discourse acts which allow to react on the players interactions. This set up is quite different to the approach used in the Virtual Storyteller, where the internal motivations and goals of the characters drive the story forward. In such a situation, you do not know in what situations the story might end up, which makes it impossible to attach certain lines of dialogue to certain parts of the story. Therefore, the method used in Façade is not really applicable in the Virtual Storyteller.

4. SUGGESTED APPROACH

There are two distinct approaches to implement dialogue in the Virtual Storyteller, which will be shortly outlined below.

4.1 Domain-independent approach

With the first approach, which will be referred to as the domain-independent approach, a separate system for dialogue handling is developed which runs alongside the system which handles the actual actions. The dialogue system can generate any type of dialogue action to request or give information based on the facts it knows from the story world. It can also be expanded to cope with more complex dialogue acts, such as promising other agents to do something, or convincing other agents to do something for the speaker.

Such a dialogue system can be very powerful. By defining any speaking agent in the story world, it can communicate, using at least the basic dialogue acts, with other speaking agents without having to define new actions. It can request information it wants to know, for example information needed to perform certain actions or reach certain goals. Once you have such a system in place, you automatically have dialogue actions for any ontology you develop. The more objects and properties this ontology has, the more possible dialogue acts you get.

The major disadvantage of the domain-independent approach is that it requires a huge investment before it pays out. An entire system needs to be developed, which needs to be integrated with the goal management of the agents in the Virtual Storyteller world. There has been some research into general communication between agents, such as [Ahn01] and [All94], which show it can be complex. Still, it should be noted that both research papers focus on communication between agents which do not necessarily share the same concepts, while in the Virtual Storyteller, the way facts are defined are the same for both agents. This should make communication a whole lot easier, considering one of the main concerns [Ahn01] points out, is getting the agents to have the same vocabulary. When agents share the same concepts, as noted in section 2.1, the ambiguity of their utterances is no longer a problem.

But still, especially in the current state of the Virtual Storyteller, such a separate system is not a good idea. Most of the current functionality has still a lot of development to be done. By adding another system to a not yet completely functioning system, all the weaknesses of the different parts add up together to a system which is not capable of as much as each of the separate parts of the system would suggest. It would be better to first invest in getting the system more robust and dynamic, while keeping the dialogue acts defined as normal actions.

4.2 Action-based approach

A better approach for the time being would be a action-based approach, where dialogue acts are defined just as normal actions, with the main difference they do not change anything in the physical world, only the internal state of the characters. The advantage of this approach is that it can be quickly implemented and tested, because it does not require major changes to the system. A disadvantage is that it is not a long term solution. For every different story world, new dialogue acts need to be defined which fit that story. Some generic dialogue acts could be defined in the story world core, a set of basic facts which apply for every domain, so it does not need to be reproduced for every ontology. Still, it requires to have many different, detailed dialogue acts referring to specific subjects, instead of a more information oriented approach.

By specifying dialogue acts as normal actions, the preconditions of the actions do not merely describe the physical limitations for executing the action. The preconditions in dialogue acts are more or less politeness limitations, which describe how a dialogue evolve and what rules apply. Physically, an agent can say anything at any time, but when an agents gives an answer before a question is asked, the dialogue feels less realistic. By using the preconditions, a somewhat natural structure of the dialogue can be guaranteed. Still, there is a risk the structure gets to rigid, for example requiring agents always to greet each other, while that is not necessary in every situation.

5. IMPLEMENTATION

To test the action-based approach in practice, dialogue actions have been added to an existing ontology in the Virtual Storyteller called Lollipop. This small domain was developed for testing purposes and includes two agents, a girl called Linda and a ice vendor called Otto. Linda has a goal to buy ice cream. To test if dialogue can be implemented in the Virtual Storyteller, die 'Buy' action from the Lollipop domain was divided in several sub actions.

5.1 Cooperation in dialogue

In the initial implementation of Lollipop, the action 'Buy', in which Linda buys ice cream from Otto, is defined as a single action. As long as Linda has money, Otto has ice cream and both are at the same location, the 'Buy' action takes place, resulting in the fact Linda has the ice cream, and Otto has the money.

While 'Buy' seems like a very plausible action in a story world, it undermines the concept of autonomous agents. The 'Buy' action is initiated by Linda, who wants to buy something from Otto, but there is no way for Otto not to cooperate in this action. Linda can just perform the action, no matter if Otto wants to sell the ice cream or not. Still, a 'Buy' action cannot be seen as an entirely autonomous action either. It requires the cooperation of both agents and some steps in the action, such as handing over the ice cream, even require both agents to synchronize their actions at the same time. The best way to define 'Buy', is as a joint action.

In his book 'Using Language', Clark [Cla96] states that all dialogue can be seen as a joint action, requiring the intention of both persons to participate. He illustrates this with the example of the purchase of a product in a drug store. Both persons work towards the same goal and are aware of the other person's role in the conversation.

Still, the fact that a dialogue is a joint action, does not mean there is only one scenario in which it can unfold, it merely means that both participants are in a process towards a given goal. The separate steps within a joint action, can still be defined as separate actions which sometimes are only performed by one of the agents while the other agents are passive. What action is done next is decided by the individual agent. For example, when buying something, there can be a scenario where the customer insults the vendor for taking too long, which makes the vendor not give away the product and send the customer away. Also, such a joint action could be interrupted, for example when a robber with a gun enters the shop – something you see relatively more often in stories than in real life.

Dividing an action such as 'Buy' is a useful thing to do, especially when the action can be a crucial one in the storyline. But more importantly, dividing the action is a good way to test an ongoing dialogue towards a clear goal in the Virtual Storyteller by using dialogue acts which are implemented in the same fashion as the physical actions.

5.2 Adding dialogue for buying ice

To test the implementation of dialogue in the Virtual Storyteller, the 'Buy' action was divided in several sub-actions, which is potentially useful if you want the buy action to have different scenario's and outcomes. Each of those actions represent one step in the buying process, eventually resulting in the entire buy action being completed. The scenario of the buy action could be as follows. Before buying ice cream, Linda will say something to Otto, such as 'I want some ice cream', maybe even after she greeted Otto and the ice vendor greeted back at Linda. Next, Otto will hand Linda the ice cream, while afterwards Linda gives Otto the money. Finally, they will say goodbye to each other, ending the conversation.

For testing the dialogue in the Virtual Storyteller, such separate actions have been added to the Lollipop world. Instead of having one 'Buy' action, it was divided into the actions 'Greet', 'AskFor', 'Give' and 'Goodbye'. The idea was to have a general goal, such as 'Linda wants ice cream', and have the different actions imply each other. For example, in order to ask for ice, Linda and Otto should first have greeted each other, and before Otto would give ice to Linda, she would have to ask for ice first.

5.3 Agents thinking about other agents

When implementing the first couple of actions, a clear problem of the current version of the Virtual Storyteller emerged: agents do not take into account actions of other agents. Take for example the just introduced action AskFor, which Linda uses to ask Otto for ice. The action was intuitively defined as follows (code simplified):

```
%AGENS asks TARGET for PATIENS
Action AskFor(Agens, Target,
Patiens),[
PreConditions,[
       (Agens, isNot, Target),
       (Agens, at, Location),
       (Target, at, Location),
       (Agens, greeted, Target),
```

```
            (Target, greeted, Agens),
            (Target, has, Patiens)
    ],
    Effects,[

    ]]
```

As this piece of code shows, before the action can take place, Agens (the agent performing the action) and Target (the other agent, which is asked something in this occasion) should have greeted each other, as can be seen in the highlighted piece of code. But because the two agents only consider their own actions, this way of implementing it does not work. When Linda tries to search a path to her goal, which is having ice, she does not have an action which makes Otto greet her. Instead of already doing her own actions of a necessary sub-action to reach her goal, she decides to do nothing because there is no direct path leading to her current goal. But if she would have greeted Otto, and Otto would have greeted her (because in order for him to sell ice, he would have to greet as well), the story would have evolved as planned automatically.

In reality it is not required to greet an ice vendor in order to buy ice, but this is merely an example to illustrate the problem. The greeting could also be seen as an implicit action, such as getting eye contact, merely a way of setting up the conversation. Agents should know what actions might lead to their goal, taking into account the way other agents respond to those actions.

5.4 Quick solutions

There are some ways to get around this problem and have the scenario unfold, even though the agents do not consider the action of the other agent. The most easy one is to define actions the other way around. Instead of having just the action 'Greet', an action 'getGreeted' can be added, which makes Linda enforce the Greet-action for Otto, adding the fact `Linda, greeted, Otto` to the fabula. But this is not really a nice way of solving the problem, because it undermines the idea of emergent narrative where autonomous agents do what is in their best interest, instead of conform to the actions the other agent wants them to do.

Another solution is to add new goals which make the agents behavior more reactive. For example by having a goal 'Greet back' which is active after an agent got greeted, agents directly react on another agent who greets them, even though this action might not lead to their main goal. When having Linda greet anyone she thinks is nice, she will greet Otto even though she does not know that will eventually lead to her goal of getting ice. By defining many of these sub-goals, it is possible to get through the conversation and reach the main goal, but the separate actions are not done in order to reach that main goal, which should actually be the reason to do such actions. Also, it requires to define reactions on every piece of dialogue, risking that only the 'right path' through the dialogue will be possible, instead of a more dynamic dialogue which one would prefer in a Virtual Storyteller.

To implement dialogue in the Virtual Storyteller, it is required to make agents think about each other's actions. This can be done in several ways, for example by looking directly into the knowledge base of the other agent, looking at its goals and preconditions. In order to reach a goal for which actions of another agent are needed, agents can look at the preconditions of those actions. An agent can sometimes fulfill a preconditions of an action by another agent by doing a different action itself. When doing so, this might lead eventually to the agent's goal and might be a clever action to do at that time.

While this seems as a plausible solution, it is not very realistic when agents know everything about other agents in a story context. Instead of knowing how other agents think and what their goals are, it is better to consider some general politeness rules which give a probability of other agents doing agents in response to your own actions. Also, goals of other agents could be predicted by looking at a situation. Instead of knowing Otto sells ice, he probably wants to sell ice because he is an ice vendor.

5.5 Recommendations

FearNot! [ALD05] uses a probabilistic approach to reason about other agent's behavior, where agents plan considering there is a certain chance of the other agent doing a certain action. Using this probability, agents choose the action which has the highest likeliness of giving them a positive outcome in order not to get bullied.

A similar implementation can be used for the Virtual Storyteller. In addition to having direct effects for certain actions, those actions can also have a set of expectations, which are what the agent thinks might happen when he performs such an action. For example, an expectation associated with greeting, might be that he gets greeted back by another agent. When the planning algorithm takes into accounts these expectations and plans towards the goal considering the expectations are true, agents can make plans which do not only take into account their own actions. There should also be a mechanism in place to change the expectations if the current expectations are not fulfilled when performing the action. Otherwise, agents will keep doing the same action over and over again, even though the other agent is not doing the expected behavior.

Especially for dialogue, it is important that agents can think about the actions of the other agents. This whole idea is currently missing from the Virtual Storyteller and makes it impossible to implement real dialogue, even with the action-based approach, without having to define almost every step in the process. Once the agents can plan to their goals using the expectations, it should be possible to implement basic conversations between agents by defining dialogue as actions, but in the long run, a more generic approach to dialogue seems like the best solution.

6. CONCLUSION

Contrary to the Virtual Storyteller, in most interactive storytelling systems dialogue is a central aspect. Those systems are more or less designed around the idea of having dialogue and could even be seen as a long, ongoing conversation. The Virtual Storyteller is currently focused on physical actions. Implementing dialogue within such a system is something new and requires a different approach than used by the other systems.

It is possible to add some simple dialogue acts which are not part of an ongoing conversation, but those differ mainly from physical actions on a narrative level. Whether you kick someone or insult him, as long he does not get hurt, the actual effect on his internal state can be the same.

But before dialogue can really be implemented, other changes need to be made to the Virtual Storyteller first. Most importantly, agents need to be able to think about the actions and responses of other agents, in order to plan towards the goal of the conversation. This can be done through adding expectations to actions. Those expectations are facts which do not directly result from the action, but the agent believes will happen eventually by executing the action.

When such a system is in place, dialogue can be added in two ways: by defining specific dialogue acts for the situations needed in the story, or by setting up a general dialogue handling system which can generate dialogue acts based on knowledge from the used ontology.

Whatever approach is chosen, it is important that dialogue evolves in a natural way. The agents need to consider their goals during the dialogue, but also need react in a realistic fashion to the utterances of the other agent. There should be a combination of dialogue planning and reactive behavior, where the agent tries to get the right information or the right things done, while at the same time staying polite to the other agent and trying to fulfill his goals. When this reactivity is guaranteed to a certain level, agents can plan knowing that the other agent will probably cooperate in achieving his goal.

The latter requires some improvements to the goal management as well. Instead of having only one active goal, agents should have one goal which they pursue for the long run, while dialogue acts of other agents can trigger short term goals to get the right reactivity from the agent.

REFERENCES

[Ahn01] Ahn, R.M.C. Agents, Objects and Events: A computational approach to knowledge, observation and communication. *Thesis, University of Eindhoven,* 2001.

[All94] Allen, J.F. Natural Language Understanding. *Redwood City, CA. Benjamin/Cummings*. 1994.

[ALD05] Aylett, R.S, Louchart, S., Dias, J., Paiva and A., Vala. FearNot!: An Experiment in Emergent Narrative. In *Proceedings of the Fifth International Conference on Intelligent Virtual Agents,* 305-316.

[ALD06] Aylett, R.S, Louchart, S., Dias, J., Paiva, A., Vala, M., Woods, S. and Hall, L. Unscripted Narrative for affectively driven characters. In *IEEE Computer Graphics and Applications, 26, 3, (May/June 2006), 42-45.*

[CA97] Core, M.G. and Allen, J.F., Coding Dialogues with the DAMSL Annotation Scheme. In *Proceedings of the AAAI Fall 1997 Symposium on Communicative Action in Humans and Machines.* American Association for Artificial Intelligence, 1997.

[CC05] Cavazza, M. and Charles, F. Dialogue Generation in Character-Based Interactive Storytelling. In *Proceedings of the AAAI First Annual Artificial Intelligence and Interactive Digital Entertainment conference*. American Association for Artificial Intelligence, 2005.

[CCM02] Cavazza, M., Charles, F. and Mead, S.J. Character-Based Interactive Storytelling. In *IEEE Intelligent Systems, July/August 2002, 17-24.*

[Cla96] Clark, H.C. Using Language. *Cambridge University Press, 1996.*

[Fah05] Fahrenheit, developed by Quantic Dream, published by Atari in September 2005. Website: http://www.atari.com/fahrenheit/, visited on 26 june 2008.

[MS03] Mateas, M. and Stern, A. Façade: An Experiment in Building a Fully-Realized Interactive Drama. In *Proceedings of the Game Developers Conference, Game Design track, 2003.*

[Sea69] Searle, J.R. Speech Acts: An Essay in the Philosophy of Language. *Cambridge University Press American Branch*, 1969.

[Sim08] The Sims, The Sims 2, developed by Maxis, published by Electronic Arts in 2000 - 2008. Website: http://thesims.ea.com/, visited on 26 june 2008.

[TP98] Traum, D., Poesio, M., Towards an Axiomization for Dialogue Acts. In *Proceedings of TWENDIAL, the Twente Workshop on the Formal Semantics and Pragmatics of Dialogues*, May 1998, 207-222.

[Tra99] Traum, D. Speech Acts for Dialogue Agents. In *M. Wooldridge and A. Rao, Foundations of Rational Agency*. Kluwer, 1999, 169-201.

[TRA04] Theune, M., Rensen, S., Akker, op den, R., Heylen, D. and Nijholt, A. Emotional Characters for Automatic Plot Creation. *Technologies for Interactive Digital Storytelling and Entertainment (TIDSE 2004)*, in *Lecture Notes in Computer Science 3105, Springer-Verlag Berlin Heidelberg*, 2004, 95-100.