Authoring Interactive Narratives with Hypersections

Jean-Hugues Réty
Paragraphe, University Paris 8
140 rue de la Nouvelle France
93100 Montreuil - France
+33 (1) 48 70 37 12
Jh.rety@iut.univ-paris8.fr

Nicolas Szilas
TECFA, FPSE, University of Geneva
CH 1211 Genève 4,
Switzerland
+41 22 379 9307
Nicolas.Szilas@tecfa.unige.ch

Jean Clément
Paragraphe, University Paris 8
2 rue de la liberté
93200 Saint-Denis - France
+33 (0) 1 49 40 67 89
jclement@magic.fr

Serge Bouchardon
University of Technology of Compiègne
Département Technologie et Sciences de l'Homme
Centre Pierre Guillaumat
60200 Compiègne - France
+33 (0) 3 44 23 40 95
serge.bouchardon@utc.fr

ABSTRACT
We present in this paper a framework for interactive narrative authoring. This framework is based on a generalized notion of section that we name *hypersection*. We present a graphical editor and discuss reader softwares for interactive narratives based on hypersections.

Categories and Subject Descriptors
J.5 [**Computer Applications**]: Arts and Humanities – Fine arts.

General Terms
Performance, Design.

Keywords
Interactive narrative, authoring tool, hypersection.

1. INTRODUCTION
Authoring interactive narratives that both provide the user with a significant amount of non-deterministic navigation choices and satisfactory user navigation experience is a difficult challenge. Many researches have tackled this question during the past twenty years with different approaches and within different research domains.

In the domain of virtual storytelling, agents with human capabilities (reasoning, emotions, verbal and non verbal communication) interact within a fictional world. These fictional worlds most often involve mechanisms for plot structuring based on artificial intelligence techniques like for instance planning algorithms. In this context, stories and narratives are represented into the computer system with computing models for interactive narrative. These computing models often rely on an analytical approach: narrative elements like characters, places, characters actions, inter-characters relationships, etc. must be precisely defined in order to author the narrative. Several such computing models were proposed in the literature [3, 6, 8, 17, 21, 24] (see [7] for a review). They typically define partial orders on narrative events and take into account structuring elements like preconditions, assertions, causal constraints and temporal constraints, as well as narrative constraints like for instance narrative tension [15] or Suspense [20]. Some of these computing models are based on Aristotle [1], Vladimir Propp [16] or Claude Brémond's [5] literature and theatre studies (for instance the IDtension project [23] or research by Michael Mateas [14]). All these computing models differ from their degree of abstraction and from their granularity. For instance, beats necessitate from the author to specify high-level narrative units, reducing the scope of user's agency [15]. On the opposite, the IDtension system can generate low-level narrative actions.

Authoring stories in this context is a long and difficult process because it requires the detailed specification of a large amount of basic narrative elements (properties of objects, properties of characters…). Authoring interactive narratives requires taking care of numerous constraints about these narrative elements, as well as taking care of narrative constraints. In order to help authors in this complex process, some research projects are developing authoring environments with integrated sets of authoring tools aiming at helping the author in controlling all of these narrative elements and all of these constraints [9].

We discussed in [21] some of the problems faced by interactive narrative authoring. In our opinion, a key issue is that the analytical computing models mentioned above are not natural devices to authors. Authors primarily reason with linear plots or subplots that do not match easily into the models defined by interactive narrative research.

Electronic literature is another domain of interest with respect to interactive narrative. From the late eighties, authors have been investigating the authoring of electronic literature with softwares like Dreamweaver or Storyspace [4]. Some interesting results

DIMEA'08, September 10–12, 2008, Athens, Greece.

were obtained (let us cite the hypertext novel *Afternoon a story*, by Michael Joyce [11]), but electronic literature failed to produce a large variety of satisfactory narratives. In our opinion, a major reason for this situation is that the hypertext model Dreamweaver and Storyspace are based on hardly allows for controlling narrative constraints. Several proposals were made in order to extend the hypertext model, like for instance preconditions on links [4, 13]. This resulted into some improvements, but it did solve the more general problem of authoring interactive narratives with hypertexts.

Beside the domain of electronic literature, authoring tools for branching narrative have been developed [2, 18, 19, 10]. Some of these tools provide graphical authoring interfaces based on 2D graph representation. These attempts failed in offering solutions to manage large-scale hypernarratives because as the graph grows big, the author hardly maintains a global sense of the overall branching structure.

We think there is a need for interactive narrative authoring tools and engines that do not primarily involve a complex analytical process and that in the same time allow for much more control over the narrative than hypertext-based models and branching narratives. We propose such tools in this paper. Our starting point is the very important role played by structural division in authoring (non interactive) textual narratives. The organization of text into words, sentences, paragraphs, chapters, volumes, etc. helps the author in structuring her writing. This organization helps the author in maintaining a higher-level view on what she is authoring, especially when the text grows long. Whenever considering the text from the bottom of this "structural division hierarchy", the author can focus on writing words and sentences. Whenever considering the text from some higher level, the author can focus on ordering paragraphs, sets of paragraphs, chapters… that is on a higher-level structural organization of the narrative. We propose a generalized notion of section, that we name *hypersection*, in order to provide authors with high level structuring elements dedicated to interactive narrative authoring.

Section 2 in this paper describes hypersections. Section 3 presents an authoring tool based on hypersections we developed in Java. Section 4 shows excerpts from an interactive narrative with our tools.

2. HYPERSECTIONS

This section describes hypersections. In our framework, an interactive narrative is a set of fragments structured with hypersections.

Let us call *fragment* an atomic content element in our framework. Fragment contents are intended to be displayed to the audience. Fragments can contain simple text but can also contain other medias.

Termination of a fragment. Every fragment is associated a termination property. A straightforward example of a termination property associated to a fragment is: the fragment is terminated when it has been displayed to the user who is viewing the interactive narrative (this means that the user will not be displayed twice this fragment). The author of an interactive narrative can specify more elaborated termination properties in such a way that the user will be given access several times to the same fragment.

The aim of the computing model we present here is to afford authors with a simple tool for structuring fragments together. Fragments can be composed together within hypersections, in a recursive manner, as follows:

A *hypersection* is composed of:

- a set of subsections;
- a behavior;
- a termination property.

Subsections are fragments or hypersections. The set of subsections in a hypersection may contain fragments only, hypersections only, or a combination of fragments and hypersections. No circularity is allowed: a hypersection must not contain itself, directly or indirectly. Hypersections are thus organized in a hierarchical way.

Termination of a hypersection. A straightforward example of a termination property associated to a hypersection is: the hypersection is terminated when all its subsections are terminated. The author of an interactive narrative can specify more elaborated termination properties, for instance in such a way that the hypersection is terminated when a certain percentage of the fragments it contains are terminated.

In our framework, a user reads an interactive narrative by successively accessing fragments. At each step, the set of fragments that can be accessed by a user is specified by the hierarchy of hypersections, behaviors of hypersections and by termination of fragments and hypersections. We call this set: *set of successor fragments*. The hierarchical structure of hypersections and fragments specify all the different paths users can experiment among fragments. The specific path one user is going to experiment is determined by user interactions (see section 3).

We define two main classes of hypersection behaviors. These two kinds of behaviours were foreshadowed in a previous work with Robert Kendall [12].

Deterministic behaviours. Reading a deterministic hypersection (i.e. a hypersection with deterministic behaviour) proceeds with respect to the order of its sub-sections: whenever one user accesses a deterministic hypersection, she accesses its first non-terminated sub-section. To be more precise, the set of successor fragments of a deterministic hypersection is the set of successor fragments of its first non-terminated sub-section.

Deterministic hypersections provide a structural organization where subsections are organized into sequence. This kind of hypersection is particularly dedicated to plot structuring. Plot structures require the satisfaction of causal constraints and narrative constraints. The simplest way for enforcing such constraints is sequencing.

Non-deterministic behaviours. Reading a non-deterministic hypersection proceeds without any predefined order: whenever one user accesses a non-deterministic hypersection, she can access any non-terminated sub-section. To be more precise, the set of successor fragments of a non-deterministic hypersection is the union of the sets of successor fragments of its non-terminated subsections.

Non-deterministic hypersections provide a structural organization where no particular order on subsections is specified. This kind of hypersection is particularly dedicated to narrative descriptions: character descriptions, global situation descriptions, descriptions of locations where action takes place, etc.

These two types of behaviour offer basic tools for structural organization of fragments altogether. The expressiveness of the framework we propose in this paper results from their combination. A deterministic hypersection can contain (directly or indirectly) non-deterministic hypersections, and vice versa.

When authoring first examples of narratives within our framework, we found that deterministic behaviours suffered from reading problems. When accessing the last fragments of a subsection, the reader was given a decreasing number of choices. At the end, he only had one choice! We thus introduced a new type of behaviour.

Deterministic behaviours with fuzzy transitions. Reading a deterministic hypersection with fuzzy transitions proceeds with respect to the order of its subsections, except that the reader can be given access to a subsection B before the preceding one, say A, is terminated. A threshold specified by the author defines the moment where the reader can access B. For instance, if this threshold is 70%, this means that whenever one reader has read more than 70% of A, then she can both access A and B. The author can also set up a mechanism that insures that the reader will eventually terminate A before going too far in reading B. This is defined with another threshold. For instance, if this threshold is 20%, this means that if the reader reads more than 20% of B, then she is forced to terminate A before going further in reading B.

3. A GRAPHICAL AUTORING TOOL

Authoring an interactive narrative within our framework is writing a set of fragments together with organizing these fragments into a hierarchy of hypersections. Our assumption is twofold.

First, we assume that this simple framework has yet enough expressive power to allow for authoring interesting interactive narratives (we foresee extensions to what is presented in this paper in order to strengthen further this expressive power). A short interactive fiction with 38 fragments was already authored, and we are currently working on a “real size” experiment with a recognized author.

Second, we assume that authoring within our framework allows for a rather good control of the author over the overall work. In other words, we expect our framework to overcome the “lost in hyperspace” syndrome that authors (and readers…) often encounter whenever the size of an interactive narrative grows high. Our answer to this challenge is based on the one hand on the notions of hypersection and behavior that are simple enough to allow for a good understanding by human of a combination of hypersections into a hierarchy of hypersections. On the other hand, the notion of hypersection (and more specifically hierarchical composition of hypersections) allows for efficient visual representation of an interactive narrative structure into a tree of hypersections.

We designed a visual representation of a hierarchy of hypersections into a 2D tree. This representation was implemented in Java using the Jtree library. Figure 1 shows a part of the hypersection hierarchy of the short fiction we wrote with hypersections.

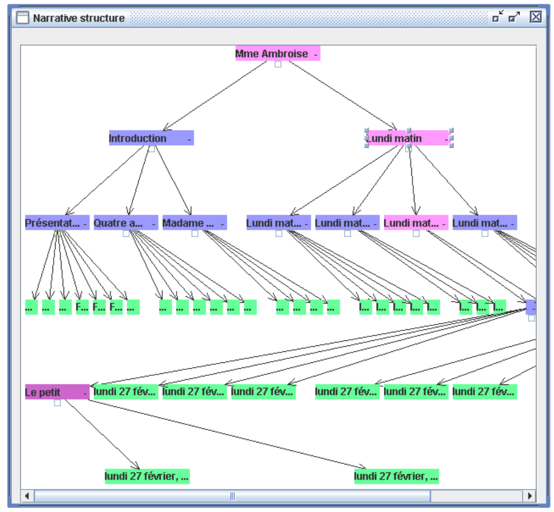

Figure 1. Hierarchy of hypersections represented into a 2D tree. Fragments are in green, non-deterministic hypersections in blue, deterministic hypersections in violet, and fuzzy deterministic hypersections in light violet.

The authoring tool includes a structure editor based on this 2D tree representation. The editor allows for creating, editing and modifying a hierarchy of hypersections. For Every node, a rollover frame gives details as shown in figure 2. Let us notice the *nbChoicesMax* parameter that appears in this figure. This parameter can be associated to any hypersection node. It constrains the cardinality of the set of successor fragments to be less or equals to nbChoicesMax. In other words, the reader tool (see section 4) will propose to the user no more than nbChoicesMax selection choices at each step when accessing the hypersection. This mechanism prevents cognitive overload of the reader with too many alternative choices.

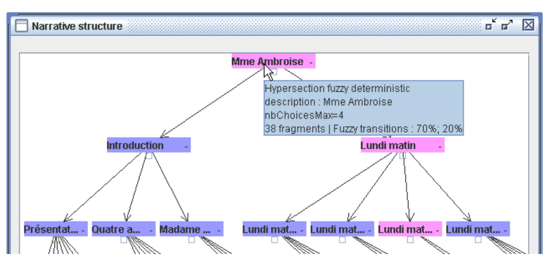

Figure 2. Rollover

Fragments are edited as shown in Figure 3. Fragment contents are coded into HTML. They can be modified with an integrated

HTML editor. Hypersections can be edited in similar way. The interface allows for adding, suppressing and moving nodes.

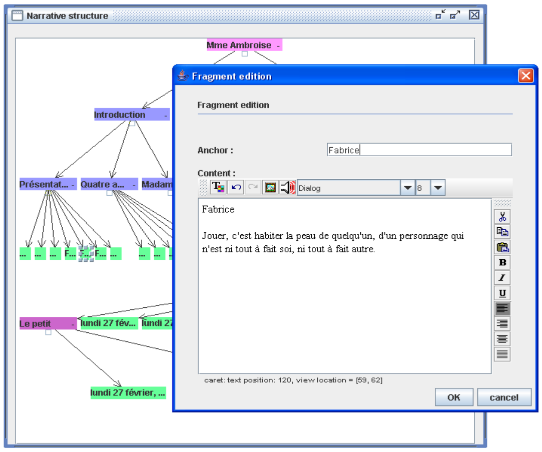

Figure 3. Edition of a fragment

Zooming and folding/unfolding of subtrees are two important features. The author can work on global organization of the narrative with viewing the entire tree or large parts of the tree. She can also focus on small parts of the tree with zooming or folding subtrees. Figures 4 and 5 show these functionalities.

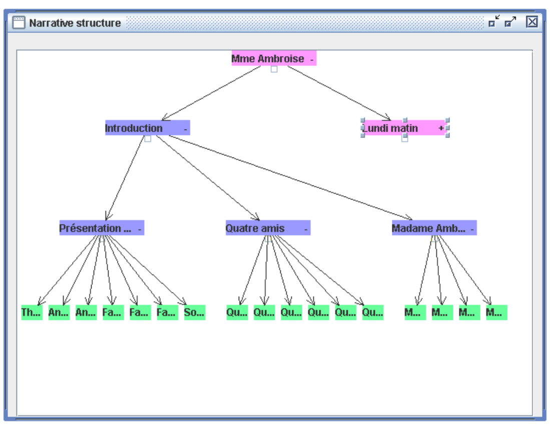

Figure 4. Folding

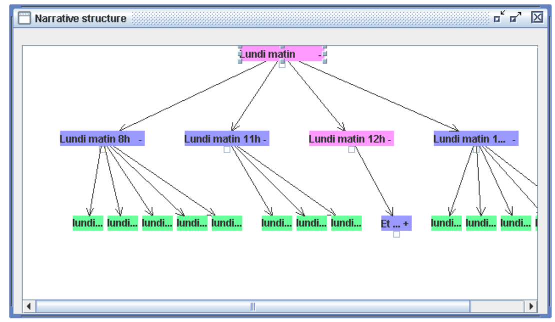

Figure 5. Zooming

Interactive narratives authored with this authoring tool are stored in an XML format. The authoring tool can import and export this format.

4. READERS

Reading an interactive narrative authored with the authoring tool described above requires a reader software. This reader tool imports an XML file and displays fragment contents to the user.

There are many ways reader softwares could manage interactivity with the user and render fragments contents. We describe in this section two reader tools we implemented in Java.

Whenever a user reads a hypersection based interactive narrative, he eventually reads a sequence of fragments. The algorithm that determines what sequence of fragments (what fragments, in what order) a user actually reads is a non-deterministic algorithm. This means that at each execution step (that is: each time a fragment is being selected for being displayed to the user), there is a set of fragments that are candidate for being selected next (this set is the set of successor fragments of the top level hypersection in the hierarchy). Interactivity can occur at this level: the user can be asked to select a fragment among the set of fragments candidates.

4.1 A non-interactive reader

Let us consider a simple reading tool where fragment selection is made by the machine on random bases. No user action is required. We then obtain some kind of a narrative generator: each execution of such a reading tool results into some randomly generated particular ordering of fragments that can then be put together into a single linear document. Let us notice that every ordering this generator can produce is obtained with the non-deterministic algorithm mentioned above. This means that every such ordering is consistent with the hierarchy of hypersections, and especially with behaviors of hypersections.

4.2 An interactive reader

Interactive reading tools let the user, at each reading step, choose among a set of fragments candidates. Such reading tools are thus composed of two parts: a fragment viewing interface and a selection device.

The viewing interface is responsible for displaying fragments. In our current implementation of the hypersection framework, fragment contents are coded in HTML (without links: HTML is here only used to format fragment contents). The viewing interface is thus a simple HTML viewer.

The selection device is responsible for letting the user select the next fragment to be displayed. Many selection interfaces can be imagined. Figure 6 shows the 3D selection devices we implemented in Java3d. The user manipulates (rotates) the 3D sphere with the mouse. She then clicks to select an “anchor”. The corresponding fragment is displayed in the fragment-viewing interface, and the selection device is refreshed with new anchors (from the new set of fragments candidates).

Let us make more precise the notion of anchor. In an interactive narrative setting, the reader must base interaction on some kind of prediction of what effect her selection (and more generally: her action) is likely to produce on the narration or on the fictional world. We decided to associate every fragment with what we call an anchor (by analogy with anchors in hypertext) because anchors bear a meaning that helps the reader making selection choices. In our current implementation, anchors, just like fragment contents, are coded in HTML. We expect however in future work developing a hypersection based framework where contents and “anchors” are actions, in such a way that reader softwares could involve virtual characters.

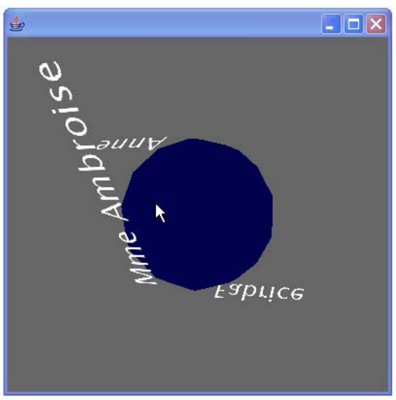

Figure 6. A 3D selection device

5. AN EXPERIMENT

A short fiction was written with our system. This section presents and discusses this experiment. Real-size writing experiments with authors will be an important further step in our work.

The short fiction involves five characters: four students and a professor. The action takes place in a University, during a lesson on Monday morning. The fiction is made of 38 fragments and 13 hypersections.

The fiction is divided into two parts, as shown in figure 7. The first part describes the characters and the general context of the story. The second part tells an event. These two parts are structured together within a deterministic hypersection with fuzzy transitions. Hence, the reader only starts reading the second part when he has gathered enough knowledge about the characters and about the context. The fuzzy transition mechanism gives a feeling of continuity in reading.

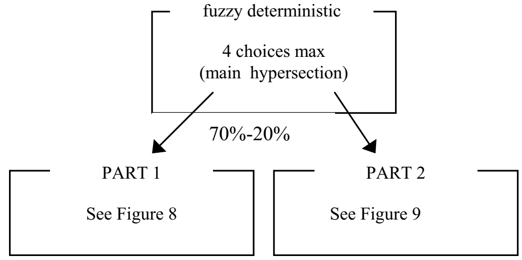

Figure 7. Top-level structure of the fiction

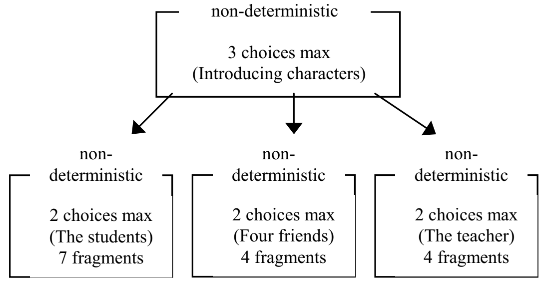

Figure 8. Structure of the first part of the fiction

5.1 First part of the fiction: description of the characters and general context of the story

Figure 8 shows the structural organization of the first part of the fiction. This part is divided into three sets of fragments:

- 7 fragments present the students as individuals (3 fragments for Fabrice, 2 for Anne, 1 for Théo, 1 for Sophie);
- 6 fragments present the students as a group of friends;
- 4 fragments present the professor.

Each of these sets of fragments is structured with a non-deterministic hypersection. These three hypersections are then structured within an enclosing non-deterministic hypersection. When all subsections of a hypersection are fragments, we did not draw them on figures (only the number of fragments is indicated).

Each of the three bottom level hypersections is set to generate two successor fragments only. The enclosing hypersection is set to generate three successor fragments. As a consequence, when three fragments are proposed for navigation to the reader, they cannot all belong to the same subsection. The reader is thus encouraged to navigate between subsections while being free not to follow this incentive if he doesn’t want to.

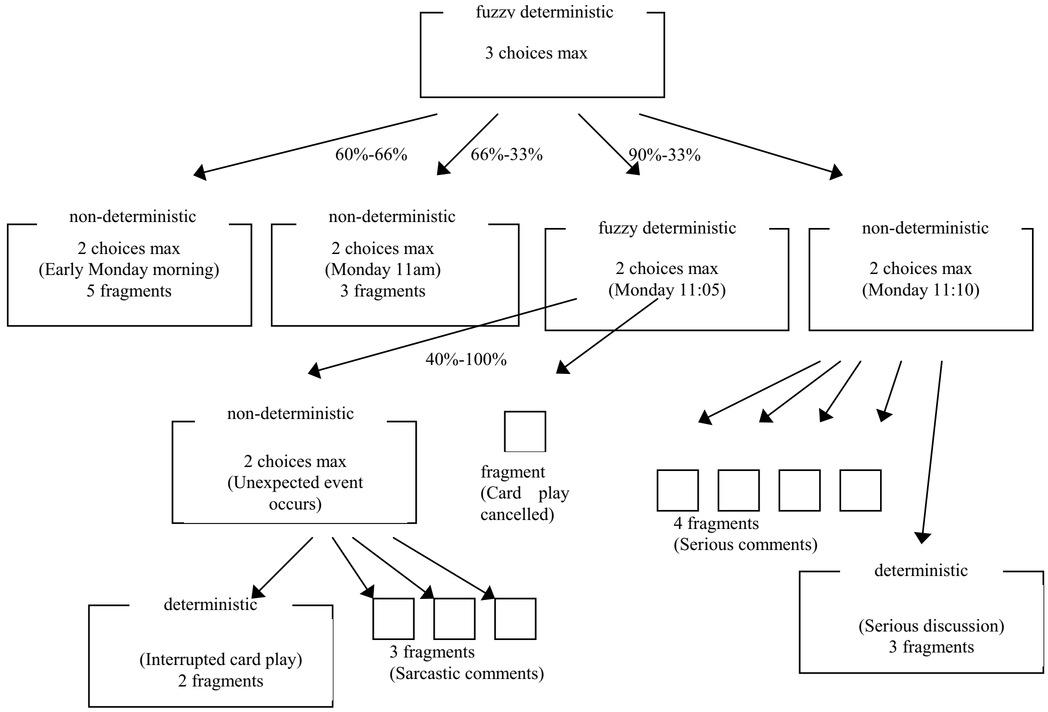

Figure 9. Structure of the second part of the fiction

5.2 Second part of the fiction: a plot line

The main plot line can be summarized as follows:

The action takes place on Monday morning. Four students are attending a lesson. At some point during the lesson the teacher, Mme Ambroise, suffers from an apoplectic stroke. The students try to remember her last words before the attack. They eventually find out that Mme Ambroise went into thought inconsistency, most probably generating global mind inconsistency, and then causing the apoplectic stroke. During the lesson, the four students were playing cards. The card play is interrupted and cancelled because of what happens to the teacher. Events about card play merge with events about the teacher's attack. All these events also merge with comments of the students about the teacher and about the card play.

Figure 9 shows how the second part of the fiction is structured. Let us notice that we drawn (with squares) the fragments contained in "mixed" hypersections (that is: hypersections whose set of subsections is composed both from hypersections and fragments). As above, when all the subsections of a hypersection are fragments, fragments are not developed in the figure.

The consistency of the narrative requires some kind of a global ordering. The steps listed below form a classical narrative schema:

1. *Early Monday morning*. Description of the situation on Monday morning (arrival at University; state of mind of the students).
2. *Monday 11:00*. Main event announcement.
3. *Monday 11:05*. Sarcastic comments of the students about their teacher (they did not realise yet how serious is the situation) crossing other comments about the card play interrupted, and then cancelled.
4. *Monday 11:10*. Resolution: the students debate about what happened to their teacher and build out their own hypothesis.

We wanted the reader to perceive this global organization. This perception however does not require that the reader complete a step before he starts reading the next step. Deterministic hypersections with fuzzy transitions are a convenient tool for this.

Let us consider the first two steps. The description of the situation on Monday morning contains five fragments, and the announcement of the main event contains three fragments. After several informal experiments, it appeared that three out of the five fragments in step 1 (any three out of the five) were providing the reader with enough information so that he could perceive the atmosphere in this Monday morning and then start reading the announcement of the main event (threshold of 60%). After the reader has read 3 fragments in step 1, fragments in steps 1 and 2 can thus interleave. Besides this, we wanted to preserve the perception of the overall organization step 1/step 2/step 3. The second threshold (66%) does this. It insures that reading the steps 1 and 2 terminates with a fragment that belongs to step 2.

Transitions between steps 2 and 3, and then 3 and 4 are organized with fuzzy transitions in a similar way.

The internal structure of step 3 is worth being detailed. This step is composed of three sarcastic comments of the students about their teacher, of a dialog divided into two fragments about the interruption of the card play, and of a fragment where the cancellation of the card play is announced. When authoring this, we wanted to insure the following constraints on the narrative:

- The two fragments in the dialog must be perceived by the reader in a comment/answer schema. We thus needed to enforce their relative reading ordering, but we also needed to avoid situations where the reading of these two fragments would be separated by the reading of too many other fragments from other hypersections. We then used a deterministic hypersection with a maximum number of intermediate fragments set to 1. For sake of brevity, we do not detail in this paper this last notion. It allows the author for enforcing the fact that the reading of fragments in a hypersection is not interleaved with too many fragments belonging to other hypersections.
- For consistency reasons, the comment announcing the cancellation of the card play shall not be read at the very beginning of step 3. Besides this, it could be read after or before the dialog (or even in between the two fragments of the dialog). This is achieved with a deterministic hypersection with fuzzy transitions. This hypersection forces that at least two fragments out of the five other fragments in step 3 are read before the fragment announcing card play cancellation is read.

Step 4 is composed of seven fragments: four comments from the students that describe what happened to their professor, and a discussion composed of three fragments. These three fragments contain the resolution of the narrative: they expose the hypothesis builded out by the students in order to try to explain the apoplexy stroke of their teacher.

These three fragments must be read with respect to a precise ordering. On the other hand, they can be preceded by, followed by (or even interleaved with) the four other fragments. Like for the dialog in step 3, these three fragments are structured within a deterministic hypersection with a number of intermediate fragments set to 1.

6. CONCLUSION

We describe in this paper a framework for interactive narrative. A graphical authoring tool and a reading tool with a simple 3D interaction device were implemented in Java. We authored within this framework a short interactive fiction and we are currently working with a professional author on a longer fiction.

This ongoing project will now develop in several directions. Once the new fiction will be finished, two kinds of evaluations will be conducted. From the author point of view, the usability of the authoring tool and the conceptual model of hypersections will be assessed (via interviews), in order to refine the tool and possibly develop new metaphors of authoring with hypersections. This evaluation takes place in an iterative design, in which the authoring tool and the engine are refined while new art pieces or demos are produced. From the reader point of view, the digital narrative will be tested on a panel of readers, in order to evaluate how the readers perceive the hypersection-based underlying model. In particular, we hypothesize that the user might perceive the difference between a simple graph and the hypersection-based model, the main contribution of the model being for the authoring process.

Further research and development is also needed on the reader tools. Indeed, the way the reader can both navigate the fragments and visualize them has a deep impact on the whole reader experience. Our goal is to provide a panel of different tools, customizable by the author, in order to enhance author's expressive power within the proposed framework.

7. ACKNOWLEDGMENTS

This research was supported by the Maison des Sciences de l'Homme Paris-Nord, the Laboratoire Paragraphe of Paris 8 University, and by the Institut Universitaire de Technologie de Montreuil.

8. REFERENCES

[1] Aristotle, 330 BC. The Poetics. Mineola, New York: Dover, 1997.

[2] Balet O., INSCAPE: An authoring platform for interactive stories, in "Virtual Storytelling. Using Virtual Reality Technologies for Storytelling" (Marc Cavazza, Stéphane Donikian Eds.), Lecture Notes in Computer Science 4871, Springer 2007

[3] Bates, J., Virtual Reality, Art, and Entertainment. In Presence: The Journal of Teleoperators and Virtual Environments, 1(1), MIT Press, Winter 1992.

[4] Bernstein M. Storyspace 1. In Proceedings of ACM Conference on Hypertext and Hyper-media, College Park, Maryland, 2002.

[5] Bremond, C. 1974. Logique du récit. Paris: Seuil.

[6] Cavazza, M., Lugrin J-L., Pizzi, D., and Charles, F., 2007. Madame Bovary on the Holodeck: Immersive Interactive Storytelling. ACM Multimedia 2007, Augsburg, Germany.

[7] Cavazza, M. and Pizzi, D., Narratology for Interactive Storytelling: a Critical Introduction. 3rd International

Conference on Technologies for Interactive Digital Storytelling and Entertainment, Darmstadt, Germany, December 2006.

[8] Cavazza M., Charles F., Mead S.J., Character-Based Interactive Storytelling. In IEEE Intel-ligent Systems 17(4): 17-24, 2002

[9] Donikian, S., Portugal, J.-N. Writing Interactive Fiction Scenarii with DraMachina, in Pro-ceedings of TIDSE'04, 2nd International Conference on Technologies for Interactive Digital Storytelling and Entertainment, Darmstadt, 2004.

[10] Gordon A. S., Authoring Branching Storylines for Training Applications. Proceedings of the 6th international conference on Learning sciences, 2004.

[11] Joyce M., Afternoon, A story, Ed. Eastgate Systems, Watertown (Massachusetts), 1990.

[12] Kendall, R., Réty, J.-H. Toward Organic Hypertext. In Proceedings of ACM Conference on Hypertext and Hypermedia, San Antonio, 2000.

[13] Kendall, R., Réty, J.-H. Connection Muse. Web-based hypertext poetry and fiction authoring system. Developed from 1999 to 2001. http://www.wordcirtuits.com/connect

[14] Mateas M. A preliminary poetics for interactive drama and games. In proceedings of SIGGRAPH, 2001.

[15] Mateas M., Stern A. Structuring Content in the Façade Interactive Drama Architecture. In Proceedings of Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2005), Marina del Rey, 2005.

[16] Propp, V. 1928. Morphologie du conte. Ed. Seuil, Paris.

[17] Riedl, M. O., Young, R. M. An Intent-Driven Planner for Multi-Agent Story Generation. Proc. of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 186-193, 2004.

[18] Silva A., Raimundo G., Paiva A., Tell Me That Bit Again... Bringing Interactivity to a Virtual Storyteller. International Conference on Virtual Storytelling 2003: 146-154

[19] Silverman, B. G., Johns M., Weaver R., Mosley J., Authoring Edutainment Stories for Online Players (AESOP): Introducing Gameplay into Interactive Dramas, In proceedings of International Conference on Virtual Storytelling(ICVS'03), novembre 2003.

[20] Somanchi, S. K. 2003. A Computational Model of Suspense in Virtual Worlds. Technical Report Number 03-002, Liquid Narrative Group, North Carolina State University.

[21] Szilas, N. 2007. A Computational Model of an Intelligent Narrator for Interactive Narratives. Applied Artificial Intelligence, 21(8), 753-801.

[22] Szilas N., Marty O., Réty J.-H. Authoring Highly Generative Interactive Drama. In Pro-ceedings of International Conference on Virtual Storytelling, Toulouse, 2003.

[23] Szilas N. IDtension: a narrative engine for Interactive Drama. In proceedings of 1st Inter-national Conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE), 2003.

[24] Young, R. M., Riedl, M. O., Branly, M., Jhala, A., Martin, R. J., & Saretto, C. J. 2004. An architecture for integrating plan-based behavior generation with interactive game environments. Journal of Game Development, 1(1), 51-70.