From Debugging to Authoring: Adapting Productivity Tools to Narrative Content Description

David Pizzi and Marc Cavazza

School of Computing, University of Teesside
TS13BA Middlesbrough, United Kingdom
{D.Pizzi,M.O.Cavazza}@tees.ac.uk

Abstract. Recent progress in Interactive Storytelling has been mostly based on the development of proof-of-concept prototypes, whilst the actual production process for interactive narratives largely remains to be invented. Central to this effort is the concept of authoring, which should determine the relationships between generative technologies underlying Interactive Storytelling engines, and the actual description of narrative content. In this paper, we report the development of an authoring technology on top of a fully-implemented Interactive Storytelling system. Although this system originated as a debugging tool for a Planning system, its interactivity as well as the high-level nature of the formalism it manipulates makes it a candidate to support collaboration between authors and technologists.

Keywords: Authoring Tools, Content Creation, Character-based Storytelling, Planning.

1 Introduction

Whilst the idea of Interactive Storytelling (IS) can be traced back to the seventies, recent progress over the past 10 years can be attributed to the adoption of generative techniques based on Artificial Intelligence (AI) formalisms. One of the principal manifestations of progress has been the multiplication of proof-of-concept IS prototypes [38][21][20][26][25], which have often required considerable effort not just in terms of implementation but also content description. The development of these prototypes has often taken place in its entirety within computing research laboratories: in the absence of a declared interest by potential users (i.e. authors), example narratives have been developed by researchers themselves, often borrowing simple plots from tales, classics or popular culture (e.g. Façade and the Edward Albee's *Who's Afraid of Virginia Woolf* [21]). If we believe IS has reached a certain maturity, it can only refer to the viability of underlying computational techniques. The issue of content creation is lagging behind the early technical developments and their proof-of-concept prototypes. Its cornerstone is likely to be the relation between content creation and the generative formalisms making interactivity possible. This relation is generally described through the concept of *authoring*, which accounts for multiple aspects of the relation between content and technology. These range from the

U. Spierling and N. Szilas (Eds.): ICIDS 2008, LNCS 5334, pp. 285–296, 2008.
© Springer-Verlag Berlin Heidelberg 2008

creative and aesthetic aspects that would be specific to IS, to the more elementary ability of representing narrative content using the AI formalisms of IS prototypes. The growing interest in authoring in the IS community [32] finds its roots largely in the imbalance between the number of technical proof-of-concept prototypes and the number and scale of actual Interactive Narratives.

In this paper, we introduce an approach to authoring which evolved from the development of proof-of-concept prototypes and the need for internal productivity tools. We suggest how these tools can be adapted to a possible production process for IS, which is inspired from the development of computer games and more specifically the relationships between designers and programmers, yet operates at a higher level of abstraction, due to the ability of IS systems to directly formalise narrative actions. We first survey current approaches to IS authoring; after introducing our IS technology, we describe an associated authoring method and its embedding in an authoring software.

2 Related Work

The authoring process[1] remains a bottleneck limiting the further development of IS and the production of interactive narratives. As a result, the IS community has dedicated significant efforts to the identification of requirements for authoring systems and associated methods. Essentially, IS systems require the author to create/generate a large amount of appropriate content [21][20] to serve as a basis for narrative generation. Recently, Medler and Magerko [23] have proposed a list of essential requirements for an Interactive Drama Authoring Tool; they extracted six properties: *Generality* (it should be independent of both game environment and story content), *Debugging* (check for redundancy, dead-ends, poor perplexity in certain decision point, consistency, etc.), *Usability* (ease of learning, efficient to use), *Environment representation*, *Pacing and Timing* and, finally, *Scope*. Conversely, McNaughton et al. [24] identified four major problems being caused by the generative property of their system ScriptEase: *Generality*, *Performance*, *Coverage* and *Evolution*, which could in turn feed back into the definition of an authoring mechanism remediating these.

Previous work has proposed a classification of authoring tools: Szilas [34] associates authoring tools with the paradigm used by the IS engine itself. He classified existing tools amongst three distinct categories: "*character based*", "*plot based*" and finally the last approach which consists in focusing on narrative properties rather than on a course of events or actions.

We propose a classification illustrating the duality between interfaces based on formalism and those based on story representation. The former foster generative aspects but lack content creation representation, the latter offer an intuitive visualisation of the story space and action sequences but force the author to explicitly express all the story alternatives limiting the plot variations.

[1] In this section, we deliberately exclude from our study authoring tools focused on the creation of visual elements to concentrate exclusively on the creation of narrative content.

Figure 1 proposes a classification of previous authoring systems along two main dimensions: *Visibility* and *Generativity*. Systems such as INSCAPE [39][7] and U-Create [28] use story-graph representation which offer a high visibility on the story plot but little generativity. On the other hand, system like ScriptEase [24] use scripted sequences of actions triggered by events which possesses substantial generative power but lacks visibility. The Scenejo authoring tool [33][37] differs sensibly from other systems as it is rather used for the generation of dialogues constitutive of a narrative in a way similar to Façade [21]. To the exception of most recent systems such as Scribe [23], Wide-Ruled [30] or even the authoring environment of the Bowman planning system [35], we can notice that most authoring systems tend to favour one property over the other.

Ideally, an authoring tool should both give access to the generative power of its formalism and allow the representation of the created narrative content. In our previous work [4], we investigated the fundamental relations between planning formalisms and authoring in IS. Despite its high power of generation, HSP [2] is often based on STRIPS formalism [9] which clearly lacks visibility for long-term dependencies between the actions carried out during the plan unfolding. To address this challenge, we have developed, as part of later experimentations, tools that will provide the representation of all the possible unfolding plans ensuring thus a greater visibility on the whole story space.

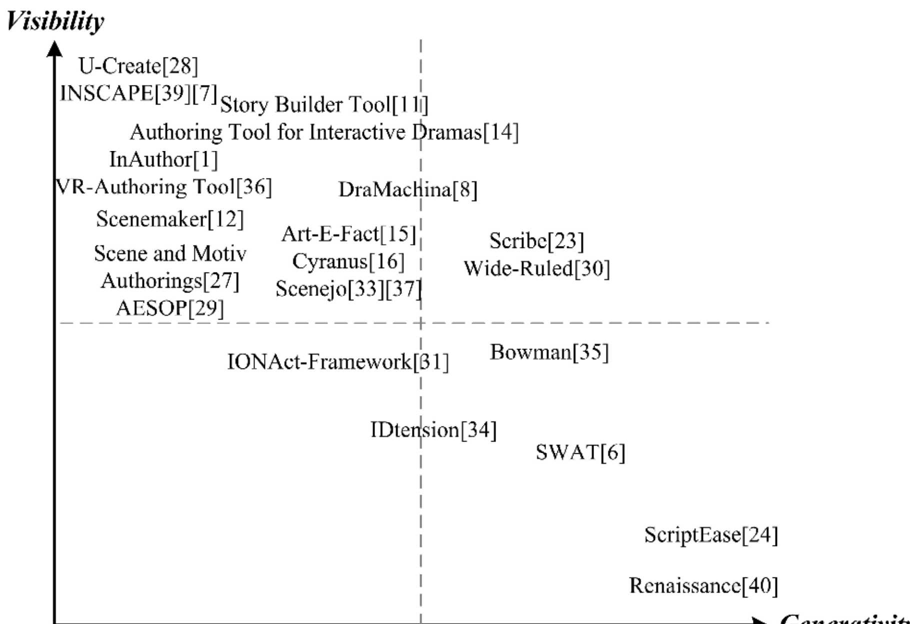

Fig. 1. Classification of existing authoring tools according to their visibility and generativity

3 Overview of Authoring Tools

We have decided to use a classic XIXth century French novel, *Madame Bovary* [10] by Gustave Flaubert, as the background for our interactive storytelling experiments. As described by Flaubert himself in his preparatory work to his novel *Madame Bovary* [19], characters' feelings can be considered as the missing link between the cognitive and a narrative perspective on the story's characters [5] and thus can be used to drive the development of the story plot. Combined with ontology of characters' feelings, our latest work on character-based affective IS [25] aims at reconciling both philosophies and uses a real-time Heuristic Search Planner (HSP) [2] [17] to generate characters' actions consistent with their psychology and allowing anytime user interactions.

Our authoring environment (Figure 2) has been developed subsequently to our proof-of-concept prototype of emotional planning for IS. Its rationale was to support the authoring of a complex planning domain, by checking its completeness and its consistency. However, since this authoring tool was an interface to the narrative formalism itself, and that the narrative formalism determined entirely the interactive narrative it became a candidate for a more generic approach to authoring. The integration of new modules could support the collaboration between authors and developers in designing an interactive narrative.

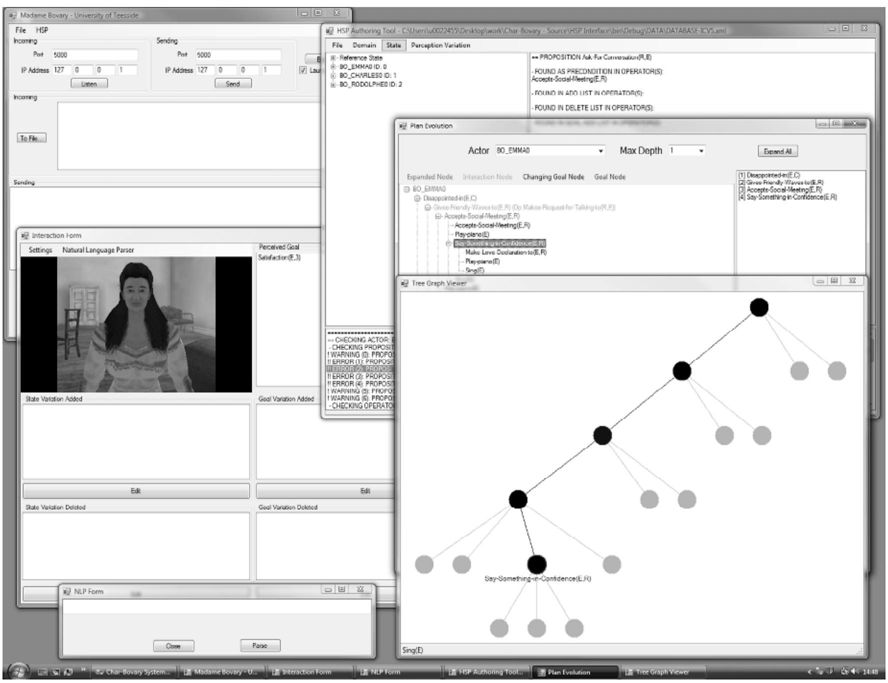

Fig. 2. Overview of our IS environment (run-time interface, interaction and authoring tools)

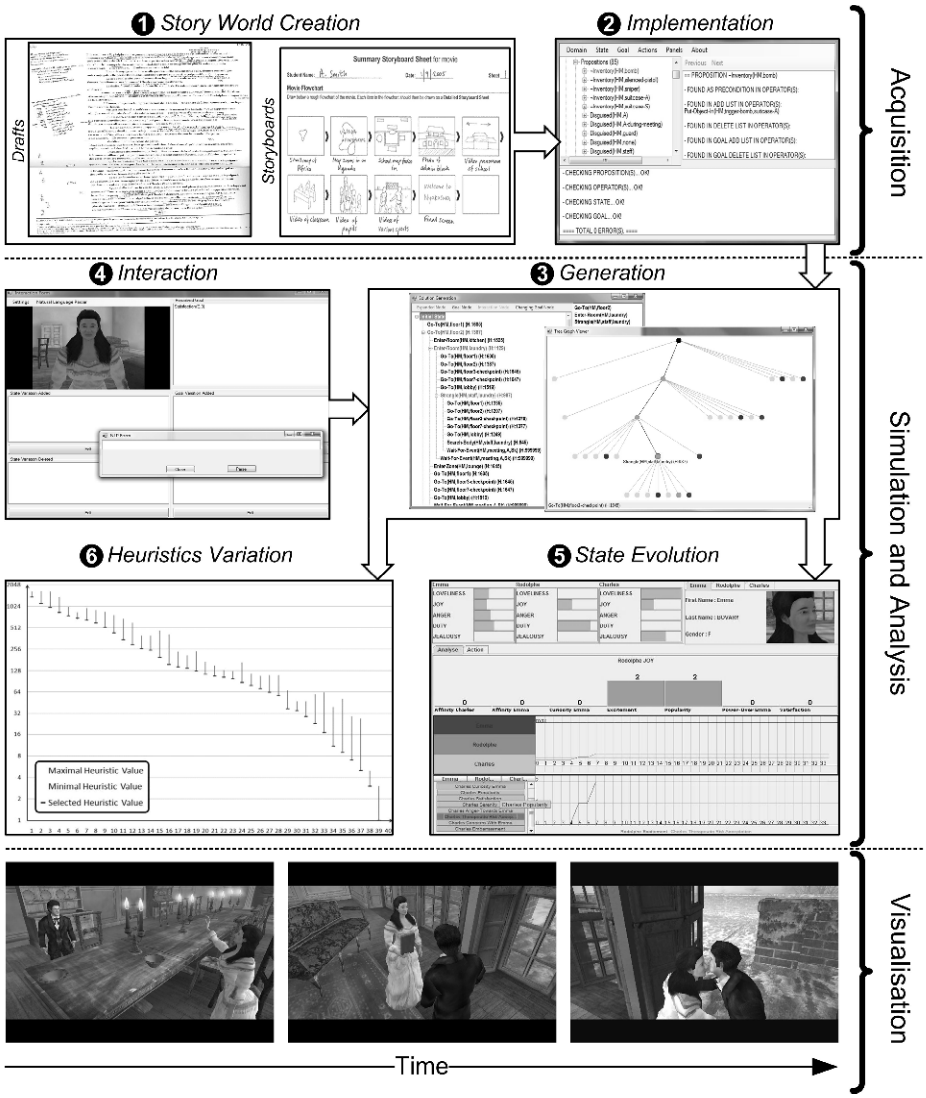

Fig. 3. Overview of the Authoring process and its three main stages (Knowledge Acquisition, Simulation and Analysis and Story Visualisation)

As illustrated by Figure 3, the authoring approach supported by our system follows three main stages: *knowledge acquisition*, *simulation and analysis,* and finally *story visualisation*.

Knowledge Acquisition. The creation of a story world (Fig 3-1) is the initial stage where drafts of story elements are created by the author. They describe diverse story elements (e.g. characters' psychology, representative scenes or environment description). The next step (Fig 3-2) corresponds to the elicitation of all knowledge required to describe the story world, such as the various states (i.e. initial and goal states) and the various actions described through their validity conditions and their consequences. In terms of Planning, this corresponds to *domain implementation* where each part of the planning domain is created (i.e. propositions, operators, states and goal). The domain description also includes a formalisation of the initial state and the goal state, which correspond to the scene's or characters' objectives.

Simulation and Analysis. As an alternative to the offline generation of a complete narrative (formally a solution plan), an interactive mode allows a step-by-step generation of a solution including the visualisation of all possible outcomes (Fig 3-3). Starting from the initial state, the user can expand the plan at each step using a tree representation until the goal state is reached. After each action is selected by the user, the system automatically offers a list of possible subsequent actions. For instance, the system will only offer the solution of Emma accepting an invitation once Rodolphe would have proposed it. This simulation has both a formal component (access to the planning domain, inspection of operators and world states) and a visual component (a tree structure providing a natural visualisation of actions and their consequences). This dual visualisation is meant to support collaboration and explanation between system developers and content creators.

In addition, authors can interact with the solution generation process at any time (Fig 3-4). This module includes also a *dynamic environment simulation* feature. It allows reproducing changes in the world not triggered by the planning system that will normally occur within the story, without having to simulate this in the complete 3D environment.

Then, several analysis tools (Fig. 3-5 and 3-6) can allow the validation of the generated narrative content. For instance, the evolution of the world state through the development of the story plan is an effective way of ensuring its consistency with respect to the characters' psychology by allowing the analysis of how emotions intensities vary along the plan evolution.

Story Visualisation. Finally, when the result has been validated, the last stage is to visualise the final result using the run-time engine.

We can observe that the early step of this production process is a specific case of *knowledge engineering* applied to planning formalisms (i.e. by integrating knowledge into computer systems in order to solve complex problems normally requiring a high level of human expertise).

In the following sections, we develop the above mentioned stages, and illustrate them using an example.

4 Domain Implementation

The first step is the elicitation of all the planning knowledge required and starts by providing a complete propositional representation of the world (e.g. characters' beliefs such as `Affinity(X,Y,`*`intensity`*`)` | *`intensity`* $\in$ `{1,2,3,...,n}`, contextual or interactions flags (e.g. `Is-in-Intimacy(X,Y)`), etc.).

The initial and goal states are then represented by conjuncts of propositions. The planning operators are represented using a STRIPS-like formalism (i.e. a set of propositions as preconditions and effects) [9] and correspond to actions that can be performed by the characters (e.g. `Is-Kissed-by(E,R)`, `Physical-Contacts (E,R)` or `Feels-Guilty-of-Estranging(E,C)`).

Usually, the grounded sequence of elementary actions that will be played by the game engine on the operator selection is also determined at this stage.

The particular choice made on the granularity of the domain representation is not without any impact on the types of solutions that will be generated later. Common pitfalls consist in representing the domain at a level of abstraction too high or conversely using too specific actions, which will limit the generative power of the system. Indeed, a too low level of abstraction would only produce minimal story variations and not real compelling plot twists.

The description of the planning domain is a manual process which becomes error-prone when its size (and the corresponding number of operators) increases. It is one well-known problem to maintain consistency between the predicates used by the various operators, which can become challenging when describing operators manually. Inconsistencies in predicates' labels could also be responsible for errors in the content of operators with other detrimental side-effects. There is thus a need to check consistency of preconditions and effects every time changes to the planning domain are introduced as part of the knowledge elicitation process.

5 Plan Generation

When solutions rely on a sophisticated plan, the various causal dependencies as generated by HSP planning may be difficult to recognise. Therefore, we wanted to explore whether the set of possible plans could be visually represented in order to control the unfolding of the generated content. Moreover, the combinatorial aspect of content generation can quickly overflow the amount of possible paths that can be exploited if done using a brute force approach. For instance, if we consider the first reference milestone set by Mateas [22]: an interactive story is compelling if it presents a 10 minute length with one beat per minute. With an average of one beat every 4 actions (~15 seconds duration per action) it would thus represent a 40 steps long story. Now, with an average perplexity of 5 operators per step, a 40 steps long plan would simply present a total of 5^{40} ($\sim 10^{28}$) possible solutions. For that reason, our system proposes a step-by-step plan simulation, in which the user can manually expand nodes and from there visualise the results using a tree-like hierarchy (see Figure 4).

Every operator in the planning domain is tested for applicability. Whenever an operator preconditions are satisfied, it is applied to create a new state that could further be extended. For efficiency purposes, and in order to avoid redundancy, we only develop new states that diverge significantly from their parent state. The whole tree can then be automatically scanned and all the possible solutions can be listed and visualised.

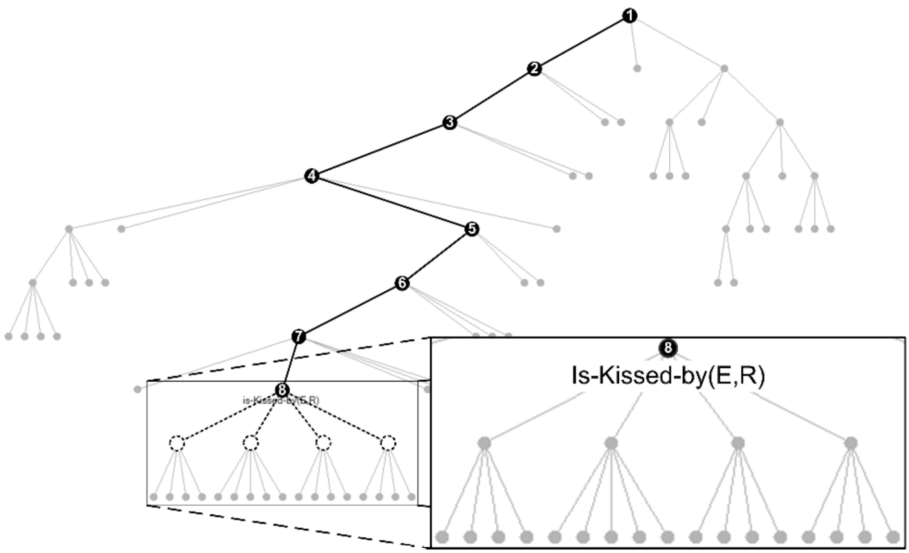

Fig. 4. Tree-based visualisation of the unfolding plan for the narrative

6 Simulating User Interaction during Authoring

We can specify interactions at any time in the form of applied operators or more simply by directly editing the propositions present in the world state. The tool generates from there the new unfolding plans saving a significant time during the domain creation.

Figure 5 shows the impact of an interaction upon the following action selection. In the original plan and once the action `Is-Kissed-by(E,R)` has been executed by the game engine, Emma will have to choose her next action amongst four different operators. According to the best heuristic value [18], the HSP planner will select the operator `Physical-Contacts(E,R)` has it presents a minimum heuristic of 19. Now, it is obvious that one user interaction could induce a variation in the following available actions and thus on the subsequent plan generation. Using the tool allowing the input of user utterance, we enter the sentence "*you shall not be one of these frivolous women*" which will have the effect of adding the proposition `Embarrassment(E,3)` within Emma's beliefs (see [25] for more details). As a consequence, we can observe that following the action `Is-Kissed-by(E,R)` five different operators can now be applied. We can also see that the best subsequent heuristic value (6) belonging now to an alternative operator `Feels-Guilty-of-Estranging(E,C)`, the HSP planner will produce an alternative plan that will lead Emma to return to her husband forgetting about her new love affair.

In a real game engine, the environment is often dynamic (e.g. Non-Player Character (NPC) have autonomous behaviour). For instance, a NPC can keep walking from one room to another on cycles. Consequently, spatio-temporal variations have also to be simulated within the solution generation process. We have introduced the

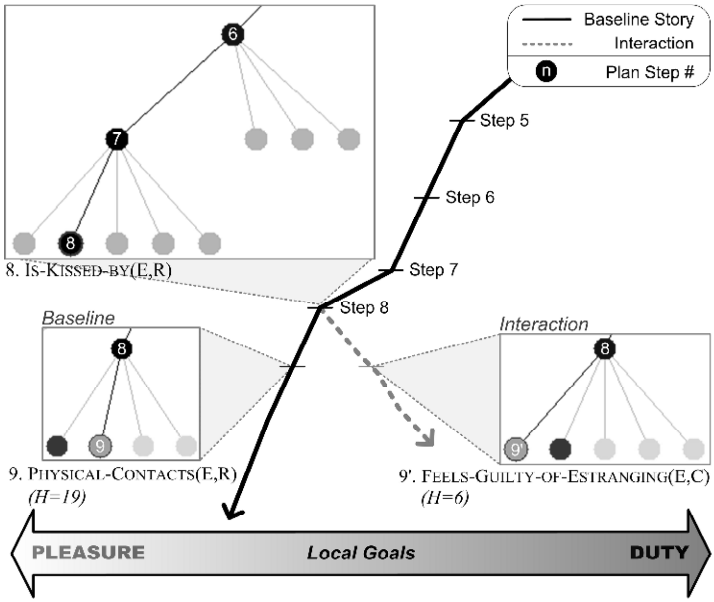

Fig. 5. Representation of the impact of user interaction upon the unfolding plan

possibility to modify the world state at any stage in order to make the appropriate state variations. The plan generation is synchronised using external actions with plan interruptions reproducing characters movements (e.g. `Update-Position`) or world events (e.g. `Character-Level-Arrival`).

7 Analysis and Validation

In line with our commitment to the exploration of aesthetic/dramatic phenomena in the narrative, we are looking for a representation that would describe the overall story progression. The direct analysis of actions' sequence seems to be a simple method of appreciating consistency and can thus constitute a preliminary validation of the generated narrative.

However, if we seek inspiration from narratology, Bremond [3] has proposed to evaluate the characters' situation in terms of improvement and deterioration, including multiple causal descriptions of the reciprocal characters' influence on the process. Dramatic aspects can then occur from such influences and improve the overall plot quality. Reasoning on causal dependencies in plans seems better suited to task models [26] or plan-space search, in which such dependencies can be explicitly formalised (see e.g. the examples discussed in [13]). For state-space search, as implemented when using HSP, the contribution of specific actions to an agent's desires is best measured through the heuristic function. For example, analysing the evolution of the ranges of heuristic function variations can inform about the diversity and quality of story alternatives. The evolution of the world state composition can also inform authors about how the emotional dimensions are exploited during the plan, which can help to ensure the believability and consistency of the characters according to their psychology.

Therefore, the generation of narrative content should not remain the only areas that are assisted by authoring tools. Our initial productivity tools have been refined to support both the complete analysis of generated sequences (Section 5) and the impact of anytime user interactions upon the unfolding story (Section 6). The formalisation of narrative actions into planning operators (STRIPS or PDDL) undoubtedly requires some knowledge of Planning in both description and appraisal of the domain. This limits the use of such tools to either programmers or authors motivated enough to acquire knowledge engineering skills. However, if the ability of the IS community to produce validation method is critical for its maturity, we believe that the experience gained at expert level could constitute a first case study and a starting point to the development of authoring methodologies for designers and writers.

8 Conclusion

We have presented our development of an authoring technology on top of our fully-implemented IS system based on Flaubert's novel *Madame Bovary*. Although this system originated as the debugging tool for our Planning system, its interactivity as well as the high-level nature of the formalism it manipulates makes it a prominent candidate for the support of collaboration between authors and technologists.

One essential question that we need to ask ourselves as developers is whether an authoring tool could exist as fully independent from these underlying IS technologies. The idea of devising perfectly generic narrative formalisms may remain utopical as it is very difficult to extract them from the procedural concepts of narrative content generation. One of the demonstrations of this has been the proliferation of different authoring tools which have been developed over the past few years within the IS community.

Acknowledgments. Jean-Luc Lugrin is thanked for the development of the visualisation environment.

References

1. Barrenho, F., Romao, T., Martins, T., Correia, N.: Authoring environment: Interfaces for creating spatial stories and gaming activities. In: Proceedings of the 2006 ACM SIGCHI international conference on advances in computer entertainment technology (2006)
2. Bonet, B., Geffner, H.: Planning as heuristic search: new results. In: Biundo, S., Fox, M. (eds.) ECP 1999. LNCS, vol. 1809, pp. 360–372. Springer, Heidelberg (2000)
3. Bremond, C.: Logique du Récit. Editions du Seuil, Paris (1973)
4. Charles, F., Lozano, M., Mead, S.J., Bisquerra, A.F., Cavazza, M.: Planning formalisms and authoring in interactive storytelling. In: Göbel, S., et al. (eds.) Proc. TIDSE 2003, pp. 216–225. Frauenhofer IRB Verlag (2003)
5. Christian, D., Young, R.M.: Comparing Cognitive and Computational Models of Narrative Structure. In: The Proceedings of the National Conference of the American Association for Artificial Intelligence (2004)

6. Crawford, C.: Storytron Interactive Storytelling, Project (last accessed July 1, 2008), http://www.storytron.com/
7. Dade-Robertson, M.: Visual Scenario Representation in the Context of a Tool for Interactive Storytelling. In: Cavazza, M., Donikian, S. (eds.) ICVS-VirtStory 2007. LNCS, vol. 4871, pp. 3–12. Springer, Heidelberg (2007)
8. Donikian, S., Portugal, J.: Writing interactive fiction scenarii with draMachina. In: Göbel, S., Spierling, U., Hoffmann, A., Iurgel, I., Schneider, O., Dechau, J., Feix, A. (eds.) TIDSE 2004. LNCS, vol. 3105, pp. 101–112. Springer, Heidelberg (2004)
9. Fikes, R., Nilsson, N.: STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intellligence 2, 189–208 (1971)
10. Flaubert, G.: Madame Bovary. In: revue de Paris, L. (ed.), France (1856) (in French)
11. Fuertes, J., Gonzalez, A., Mariscal, G., Ruiz, C.: Developing Virtual Storytellers for the Virtual Alhambra. In: Cavazza, M., Donikian, S. (eds.) ICVS-VirtStory 2007. LNCS, vol. 4871, pp. 63–74. Springer, Heidelberg (2007)
12. Gebhard, P., Kipp, M., Klesen, M., Rist, T.: Authoring Scenes for Adaptive, Interactive Performances. In: AAMAS-2003. Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 725–732 (2003)
13. Gratch, J.: Why you should buy an emotional planner. In: Proceedings of the Autonomous Agents 1999 Workshop on Emotion-based Agent Architectures (EBAA 1999) (1999)
14. Hartmann, K., Hartmann, S., Feustel, M.: Motif Definition and Classification to Structure Non-linear Plots and to Control the Narrative Flow in Interactive Dramas. In: Subsol, G. (ed.) ICVS-VirtStory 2005, vol. 3805, pp. 158–167. Springer, Heidelberg (2005)
15. Iurgel, I.: From Another Point of View: Art-E-Fact. In: Göbel, S., Spierling, U., Hoffmann, A., Iurgel, I., Schneider, O., Dechau, J., Feix, A. (eds.) TIDSE 2004. LNCS, vol. 3105, pp. 26–35. Springer, Heidelberg (2004)
16. Iurgel, I.: Cyranus – An Authoring Tool for Interactive Edutainment Applications. In: Pan, Z., Aylett, R.S., Diener, H., Jin, X., Göbel, S., Li, L. (eds.) Edutainment 2006. LNCS, vol. 3942, pp. 577–580. Springer, Heidelberg (2006)
17. Korf, R.E.: Real-time heuristic search. Artificial Intelligence 42(2-3), 189–211 (1990)
18. Liu, Y., Koenig, S., Furcy, D.: Speeding Up the Calculation of Heuristics for Heuristic Search-Based Planning. In: The Proceedings of the Eighteenth National Conference on Artificial Intelligence, pp. 484–491 (2002)
19. Leclerc, Y. (Eds.): Plans et Scenarios de Madame Bovary, CNRS Edn., France (1995) (in French)
20. Magerko, B.: Building an Interactive Drama Architecture. In: Göbel, S., et al. (eds.) Proc. TIDSE 2003, pp. 226–237. Frauenhofer IRB Verlag (2003)
21. Mateas, M., Stern, A.: Façade: An Experiment in Building a Fully-Realized Interactive Drama. In: Game Developer's Conference, San Francisco, CA (2003)
22. Mateas, M., Stern, A.: A Behavior Language for Story-Based Believable Agents. IEEE Intelligent Systems 17(4), 39–47 (2002)
23. Medler, B., Magerko, B.: Scribe: A Tool for Authoring Event Driven Interactive Drama. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 139–150. Springer, Heidelberg (2006)
24. McNaughton, M., Cutumisu, M., Szafron, D., Schaeffer, J., Redford, J., Parker, D.: ScriptEase: Generative Design Patterns for Computer Role-Playing Games. In: ASE 2004 (2004)
25. Pizzi, D., Charles, F., Lugrin, J.-L., Cavazza, M.: Interactive Storytelling with Literary Feelings. In: Paiva, A.C.R., Prada, R., Picard, R.W. (eds.) ACII 2007. LNCS, vol. 4738, pp. 630–641. Springer, Heidelberg (2007)

26. Riedl, M.O., Stern, A.: Believable Agents and Intelligent Story Adaptation for Interactive Storytelling. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 1–12. Springer, Heidelberg (2006)
27. Rober, N., Huber, C., Hartmann, K., Feustel, M., Masuch, M.: Interactive Audiobooks: Combining Narratives with Game Elements. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 358–369. Springer, Heidelberg (2006)
28. Sauer, S., Osswald, K., Wielemans, X., Stifter, M.: U-Create: Creative Authoring Tools for Edutainment Applications. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 163–168. Springer, Heidelberg (2006)
29. Silverman, B., Johns, M., Weaver, R., Mosley, J.: Authoring Edutainment Stories for Online Players (AESOP): A Generator for Pedagogically Oriented Interactive Dramas. In: Balet, O., Subsol, G., Torguet, P. (eds.) ICVS 2003. LNCS, vol. 2897, pp. 65–73. Springer, Heidelberg (2003)
30. Skorupski, J., Jayapalan, L., Marquez, S., Mateas, M.: Wide Ruled: A Friendly Interface to Author-Goal Based Story Generation. In: Cavazza, M., Donikian, S. (eds.) ICVS-VirtStory 2007. LNCS, vol. 4871, pp. 26–37. Springer, Heidelberg (2007)
31. Sobral, D., Machado, I., Paiva, A.: Managing Authorship in Plot Conduction. In: Balet, O., Subsol, G., Torguet, P. (eds.) ICVS 2003. LNCS, vol. 2897, pp. 57–64. Springer, Heidelberg (2003)
32. Spierling, U., Iurgel, I.: Pre-Conference Demo Workshop "Little Red Cap": The Authoring Process in Interactive Storytelling. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 193–194. Springer, Heidelberg (2006)
33. Spierling, U., Weiß, S., Müller, W.: Towards Accessible Authoring Tools for Interactive Storytelling. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 169–180. Springer, Heidelberg (2006)
34. Szilas, N.: IDtension: a narrative engine for Interactive Drama. In: Göbel, S., et al. (eds.) Proc. TIDSE 2003, pp. 187–203. Frauenhofer IRB Verlag (2003)
35. Thomas, J., Young, R.M.: Author in the Loop: Using Mixed-Initiative Planning to Improve Interactive Narrative. In: The ICAPS 2006 Workshop on AI Planning for Computer Games and Synthetic Characters (2006)
36. Wages, R., Grutzmacher, B., Conrad, S.: Learning from the Movie Industry: Adapting Production Processes for Storytelling in VR. In: Göbel, S., Spierling, U., Hoffmann, A., Iurgel, I., Schneider, O., Dechau, J., Feix, A. (eds.) TIDSE 2004. LNCS, vol. 3105, pp. 119–125. Springer, Heidelberg (2004)
37. Weiß, S., Müller, W., Spierling, U., Steimle, F.: Scenejo – An Interactive Storytelling Platform. In: Virtual Storytelling, Using Virtual Reality Technologies for Storytelling, Third International Conference, Proceedings, Strasbourg, France, pp. 77–80 (2005)
38. Young, R.M., Riedl, M.O.: Towards an Architecture for Intelligent Control of Narrative in Interactive Virtual Worlds. In: Proceedings of the International Conference on Intelligent User Interfaces (IUI), Miami, FL, USA, pp. 310–312 (2003)
39. Zagalo, N., Göbel, S., Torres, A., Malkewitz, R.: INSCAPE: Emotion Expression and Experience in an Authoring Environment. In: Göbel, S., Malkewitz, R., Iurgel, I. (eds.) TIDSE 2006. LNCS, vol. 4326, pp. 219–230. Springer, Heidelberg (2006)
40. Zancanaro, M., Cappelletti, A., Signorini, C., Strapparava, C.: An Authoring Tool for Intelligent Educational Games. In: Balet, O., Subsol, G., Torguet, P. (eds.) ICVS 2001. LNCS, vol. 2197, pp. 61–68. Springer, Heidelberg (2001)