Universiteit Twente
de ondernemende universiteit

Human Media Interaction

University of Twente
Department of Electrical Engineering,
Mathematics & Computer Science
Human Media Interaction Group

Comics

__C__reative __O__ntology-based __M__achine __I__llustrating __C__omic __S__tories

Comic Generation from Story Content Graphs

by

René Zeeders

July 2010

Examination Committee:	Dr. Ir. I. Swartjes
	Dr. M. Theune
	Prof. dr. ir. A. Nijholt

Preface

Ever since I was a child I wanted to become a comic artist. I watched educational programs on TV about comics and bought books which explained the various techniques of drawing comics. Using this information I practiced intensively for years. When I was twelve and went to primary school, I even gave a lecture about comics. But as I got older, the dream of becoming a comic artist began to fade, until two years ago, when I started this project.

Though my child's dream has not been fulfilled through this project, I enjoyed working with comics again. Through this project I learned a lot about comics in general and gained a deeper respect for it; its expressive power, the way it involves its audience, how more senses are involved than just sight, and so much more. Now, at the end of this project, I see what the future could bring in comics generation, and have great hopes for its development. I'm looking forward to see what kind of comics computers will be able to generate. Another fun aspect of this project has been the fact that I had to draw a lot of comical characters and such. You will see this both in the system I implemented, as at the beginning of every chapter and in the various figures throughout this thesis.

During the first part of this project, I often thought this project would not teach me anything useful with regard to my future, as I would only apply everything I had already learned throughout my education. But as I look back, I now see all the things I have learned and, would I restart this project, all the things I would do differently. People often say that you will only learn driving after you earned your driver's permit, and I'm starting to believe this also regards graduating.

I want to extent my thanks to some persons who have been important to me during this project and were a great source of relaxation every time I needed it. First I want to thank God and his Son, Jesus Christ, for being with me every day of my live. Secondly my brother, who has helped me relax at the end of each workday, which was particularly helpful in the last few months, as I was often making workweeks of seven days. Next I want to thank my parents, for always being a big support, and for having me over once a week for dinner, which saved me the trouble of cooking once a week. I'm also grateful for both my sisters and friends for all the relaxing moments I had with them, which have been and always will be important to me. Then, of course, I want to thank my mentors, Ivo Swartjes, Mariët Theune and Anton Nijholt. I want to thank Ivo and Mariët especially for their intensive guidance and support throughout this project. I have enjoyed their company during our meetings, which has made it an even more pleasant experience.

Finally I want to thank you, the reader of this thesis, for taking the time to read the result of two years of work. It would be a shame if it only would be a dust catcher in the bookcases of my mentors. I hope you will enjoy reading it and, through it, will receive both a deeper understanding and respect for comics, as increased knowledge in how to design a system that generates comics.

Yours sincerely,

René Zeeders

Table of Contents

Preface 2

Chapter 1 Introduction 6

1.1 Goal 6

1.2 Research Questions and Evaluation Criteria 6

1.3 Thesis Overview 8

Chapter 2 The Traditional Comic 9

2.1 History of Comics 9

2.2 Elements in Comics 10

2.2.1 Graphical terms 10

2.2.2 Abstract Terms 19

2.3 Summary 21

Chapter 3 Automating the Comic Creation Process 22

3.1 Comic Generation Systems 22

3.1.1 Comic Chat 22

3.1.2 Pink Panther™ 22

3.1.3 Comic Diary 23

3.1.4 Game Session Comic 24

3.1.5 Carrie the Cartoonist 24

3.1.6 Cartoon Face 24

3.1.7 Comics2D 25

3.1.8 Three Steps to Creating a Comic 25

3.2 Translation of Story Content to Comic Content 26

3.3 Graphics Creation 27

3.3.1 Screenshot vs. Composition Method 27

3.3.2 Creating and Composing the Graphics 29

3.4 Summary 30

Chapter 4 Design: Describing Comics 31

4.1 Introduction 31

4.2 The Virtual Storyteller 31

4.3 A Comics Ontology 33

4.4 Describing Comics 34

4.4.1 Existing Comics Formats: CBML, ComicsML and CSDL 34

4.4.2 Fabula Elements in Comics 37

4.4.3 A New Comics Format: ComicsDL 38

4.5 Summary 41

Chapter 5 Design: Fabula to ComicsDL 42
5.1 Creating the Story from Fabula 42
5.1.1 Dead Ends 43
5.1.2 One Unified Story to Tell 43
5.1.3 Which Story to Tell 44
5.1.4 Perceptions and Beliefs 45
5.1.5 Examples 46
5.2 Implementation 46
5.2.1 Global Overview 46
5.2.2 Jena and Named Graphs 47
5.2.3 Filter 48
5.2.4 Combiner 48
5.2.5 Converter 49
5.2.6 Example 50
5.3 Summary 51
Chapter 6 Design: ComicsDL to Comic 52
6.1 Introduction 52
6.1.1 Three Kinds of Actions 52
6.1.2 Two Tools 52
6.2 Libraries 52
6.3 ActionMaker - Creating the Action 53
6.3.1 Third Party Software 53
6.3.2 The ActionMaker 53
6.4 ComicsDL2Comic 54
6.4.1 Global Overview 54
6.4.2 Laying Out the Panels 55
6.4.3 Creating the Panels 55
6.5 Design Issues 56
6.5.1 Simple Character Representation 57
6.5.2 Semantic Relationships between Panels 57
6.5.3 Missing Action Definitions 58
6.5.4 Mismatch ComcsDL Action and Comics Action 58
6.5.5 Anchor Points and Complex Objects 58
6.5.6 Missing Visual Elements 59
6.5.7 Emotions and Facial Expressions 59
6.5.8 Shots 59
6.6 Summary 59
Chapter 7 Evaluation 61

7.1 Research Goal 61
7.2 Method 61
7.3 Comic 1: The Chase 62
7.3.1 Results 63
7.3.2 Discussion 64
7.4 Comic 2: The Canon Launch 66
7.4.1 Results 66
7.4.2 Discussion 67
7.5 Conclusion 69
7.5.1 Missing graphical elements 69
7.5.2 Badly drawn graphical elements 69
7.5.3 Semantics 69
7.5.4 Wrong Inferences 70
7.6 QUEST 70
7.7 Summary 71
Chapter 8 Conclusions & Recommendations 72
8.1 Research Questions 72
8.2 Evaluation Criteria 74
8.3 Recommendations for Future Research 76
8.3.1 Comics Layout Composition 76
8.3.2 Action Definitions 77
8.3.3 Intelligent Behaviour 78
8.4 Conclusion 80
Chapter 9 References 81
Chapter 10 Figures 83

Chapter 1 Introduction

This chapter introduces and motivates the goal for the research I describe in this thesis. Secondly it states the research questions and evaluation criteria that will be discussed throughout this thesis, and thirdly it gives an overview of all the chapters this thesis consists of.

1.1 Goal

This thesis describes my graduation project, which is part of the Virtual Storyteller project at the research group of Human Media Interaction at the University of Twente. This project aims at designing and (partly) implementing a system that generates comics, given story content that is created by the Virtual Story Teller.

The "Virtual Story Teller" (VST) is a multi-agent framework that can generate the content of stories automatically. Multiple agents participate in a simulation as either characters, or as facilitators to the simulation. The result of a simulation is saved as a story content graph called *fabula*. Fabula is a network in which the characters' actions, perceptions, goals, etc. are causally linked. To better understand the role of fabula in the context of storytelling, we need to look at the three layers of information that together form the narrative. In Swartjes & Theune (2006) a distinction is made between the *fabula* layer, the *plot* layer and the *presentation* layer, which are sometimes also referred to as the fabula layer, the story layer and the narrative layer respectively. The fabula layer is that what happens in the story world and why. The plot layer is a selection from the fabula which forms a consistent and coherent whole, and finally the presentation layer is the actual presentation of the plot layer through some medium, for example natural language, video, or a comic, etc. Slabbers (2006) designed and implemented the Narrator: a system that creates a Dutch text given a fabula. Te Brinke et al. (2008) implemented Carrie the Cartoonist: a system that generates comics given a fabula. Van Hurne et al (2007) created Annie the Animator: a system that generates an animation given a fabula. Learning from the results of these and other research projects, I will design a system that translates fabula generated by the VST to comics.

Why comics? Comics[1] is a medium widely used and accepted in many cultures all around the world. Alves (2007) stated that comics is an ideal medium to create summaries of events that happen in virtual worlds (p. iii), which, in my opinion, is true for both virtual worlds with and without graphical interface. But more importantly to me, comics is a medium with great appeal that speaks to people, from young to old, from east to west, in a unique way through combination of graphics and text, conveying motion and time through static images, showing emotions and tickling the reader's senses through use of widely accepted symbols for e.g. pain, dizziness, smell, sound, etc., and has power to involve its audience as readers identify themselves with the comics' characters (see McCloud, 1993).

1.2 Research Questions and Evaluation Criteria

In this section I will present a number of research questions, as well as evaluation criteria (questions regarding the performance of the Comics system). Both these categories of questions will be discussed throughout this thesis, which will lead to a better understanding of how to design a system that can generate comics.

To understand how to generate comics, we first have to understand what comics are; its definition and the elements comics consist of, which will be addressed in the next chapter. After we better understand what comics are, the first research question (RQ) that will be looked into is:

[1] I deliberately use the word *comics* (plural), because McCloud (1993) defines comics as the medium, while *a* comic is an instance of this medium.

RQ I. What are the steps that need to be taken to generate comics?

Distinguishing these steps helps me to subdivide the process into smaller steps. We will see that the comics generation process generally takes three steps: content generation, translation of content to a formal comics description, and finally graphics creation. Because content generation is handled by the Virtual Storyteller, the two latter steps lead to additional research questions, which fall into two categories: about translating fabula to comics and about creating graphics that form a comic. First of all we need to select a plot from the fabula, as the fabula layer contains all events that happened in the virtual world during a simulation, and the plot layer contains a consistent and coherent whole (see above). This leads us to the second research question:

RQ II. What makes an interesting story and how do you determine a plot to present from a fabula graph?

When we are able to extract a plot from the fabula, we need to know how to translate this to a comic, and how each fabula event translates to elements a comic consists of, i.e. how to go from the plot layer to the presentation layer:

RQ III. How does the plot, which consists of fabula events, translate to comics and how do these fabula events map onto comics elements?

When we are able to translate stories from fabula to comics, the graphics need to be created. First multiple ways of how comics graphics are currently created by computer systems need to be distinguished. After this I will propose a way to create comics graphics for the Comics system. This leads us to the fourth research question:

RQ IV. What are the various ways to create comics graphics and what is the best way for the Comics system?

After addressing research question RQ III we have an idea of how to go from fabula events to comics elements and we will see that actions and events are displayed through panels, while other events are narrated through a narrator box (a comics element explained later). Therefore we need to formalize rules of how actions and events map on graphical comics representations of them:

RQ V. How do generic fabula actions and events classes translate to specific graphical comics representations in which e.g. some specific character performs some specific action?

When all of these questions have been addressed, we have a better understanding of how to go from a fabula graph to the graphical comics representation of it.

Next to addressing these research questions there are some evaluation criteria (EC) that also need to be dealt with. These are about the way the Comics system is designed, as well as about the performance of the system. For future researchers to be able to keep working on the Comics system and improving it, the system needs to be extendable, so that specific parts of the system can easily be replaced or improved (so that e.g. a certain step in the plot selection part of the system can be improved in the future):

EC I. **Extendibility**: How can the system be extended so that specific steps in the comic creation process can be extended or replaced, without affecting other parts of the system?

Because the system deals with multiple external sources of information, it needs to be flexible, so that various, unforeseen things can be handled, as for example a missing character in an action: character A shakes character B's hand, but character B was not given in the fabula.

EC II. **Flexibility**: How flexible is the Comics system? In particular: how is missing information handled?

Then there is the matter of system expressiveness, i.e. how well the story extracted from a fabula graph is translated, so that the story is correctly conceived by the reader:

EC III. **Presentation Expressiveness**: Does the system create comics that are a good representation of the story content they need to convey to the reader? I.e. is the fabula correctly translated?

Another evaluation point is how much the comics generated by the Comics system can vary in artistic style, form, genre, etc. I.e. from e.g. pirate stories to western stories, from action comics to funny comics, from eastern artistic style to western artistic style:

EC IV. **Generativity**: How generative is the system? How many varying comics can the system generate?

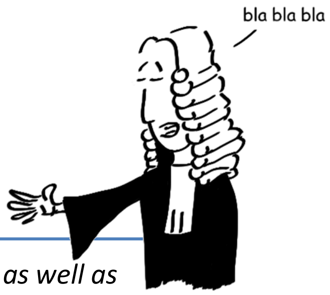
Finally we will find that there is a trade-off between generic and specific. Using generic rules the Comics system will generate specific comics, and, as we will see throughout this thesis, a multitude of problems arise, as for example consequences of actions in one panel are not visible in the next. Or the problem in which actions can be carried out by various characters, using various objects that each can vary in representation (e.g. various representations of a sword), in which, for example, not all possible restrictions of the participating objects can be taken into account. So the last evaluation criterion is:

EC V. **Generic vs. Specific Trade-off**: How does the Comics system handle the trade-off between generic and specific?

These research questions and evaluation criteria will play a central role throughout this thesis. In the next section I will give a short overview of the chapters this thesis consists of.

1.3 Thesis Overview

This thesis describes the research that led to the design of the Comics system. First in Chapter 2 the history of comics is described, and I will give a definition of comics. Next the various elements that comics consist of are addressed. In Chapter 3 we take a look at automating the comic creation process: research that has been done in the field of comic creation, and in particular the system designed by Alves (2007), as his system also depends on a story generation system. In Chapter 4 I propose ComicsDL; an XML based format in which comics can be described. Then in Chapter 5 I will describe how the Comics system will translate fabula (the output of the story generation system) to ComicsDL and in Chapter 6 how the Comics system will translate this ComicsDL file to a graphical representation of that comic. In Chapter 7 I will describe an experiment I performed to test the quality of the generated comics, and thus, in some measure, the quality of the system. The quality is determined by how well the generated comics are interpreted by the readers, given the original content the comics need to convey. Finally in Chapter 8 I will discuss the results, present conclusions, and give recommendations for future research.



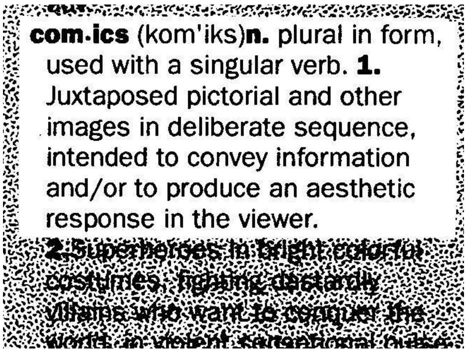
Chapter 2 The Traditional Comic

This chapter is about comics is general; a global overview of its history is given, as well as a definition of comics is stated, and the various elements that comics consist of are described.

2.1 History of Comics

Comics is a medium in which information can be transferred to the reader, as can also be done with text, music, poems, paintings, etc. Each medium has its own advantages and disadvantages. *A painting says more than a thousand words,* but it's hard to briefly describe world events in a newspaper only with pictures. Some songs are poems set in a melody, but the music shapes the words and can amplify their meaning, or negate the possibility of certain interpretations that were possible when the music wasn't there. Comics combine images and words to convey a story or an idea to the reader. It is an underappreciated art form, and Will Eisner and Scott McCloud (amongst others) try to correct this under appreciation.

Will Eisner and Scott McCloud are kind of pioneers in giving the art form of *comics* a more prestigious position in art. Will Eisner called comics 'sequential art' (McCloud, 1993, p. 5). Given this description, McCloud created a dictionary like statement to give a more formal definition of comics (McCloud, 1993, p. 9):



com·ics (kom'iks)**n.** plural in form, used with a singular verb. **1.** Juxtaposed pictorial and other images in deliberate sequence, intended to convey information and/or to produce an aesthetic response in the viewer.
2. Superheroes in bright colorful costumes, fighting dastardly villains who want to conquer the world, in violent sensational

With this definition the domain of comics is extended beyond the general idea about comics, and thus even includes the old cave drawings found in Middle America and the famous Egyptian hieroglyphics. Also the 230 foot long Bayeux Tapestry about the Norman Conquest beginning in 1066 is a form of comics. After the invention of printing, comics continued to appear as for example in 1460: The Tortures of Saint Erasmus. William Hogarth created a 6 plate picture story "A Harlot's Progress" published in 1731, hereafter he created the sequel "A Rake's Progress". But the father of

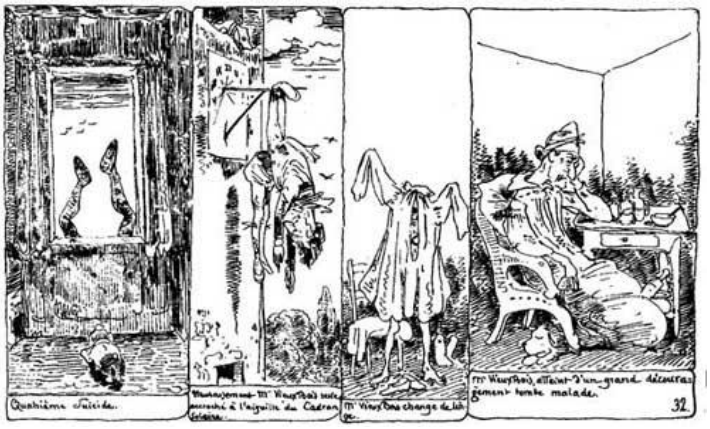

Quatrième Suicide.
Malheureusement Mr Vieux Bois reste accroché à l'aiguille du Cadran Solaire.
Mr Vieux Bois change de linge.
Mr Vieux Bois, atteint d'un grand découragement tombe malade.
32

Figure 1 Comic by Rodolphe Töpffer

modern comics in many ways is Rodolphe Töpffer starting with his work in mid 1800's (see Figure 1). He created light satiric picture stories and employed cartooning and panel borders. He also featured the first interdependent combination of words and pictures seen in Europe, i.e. what we presently classify as comics. British caricature magazines kept the traditions alive and during the 20th century the modern comics took their place in the world. (McCloud, 1993)

2.2 Elements in Comics

To get a good understanding about comics, we need to create a more formal description of all the elements that can occur in comics, as their definitions help us differentiate between the various elements and minimalize certain ambiguities. McCloud describes a lot of the elements comics are composed of. Alves (2007) created an ontology (see Figure 2) given McCloud's explanation. The elements he describes in his thesis (which are integrated in his ontology) and some additional elements are explained in the following sections. These explanations are based on his thesis and, even more, McCloud's book "Understanding Comics" (1993). I divided the elements in two separate groups: **Graphical terms** and **Abstract Terms**.

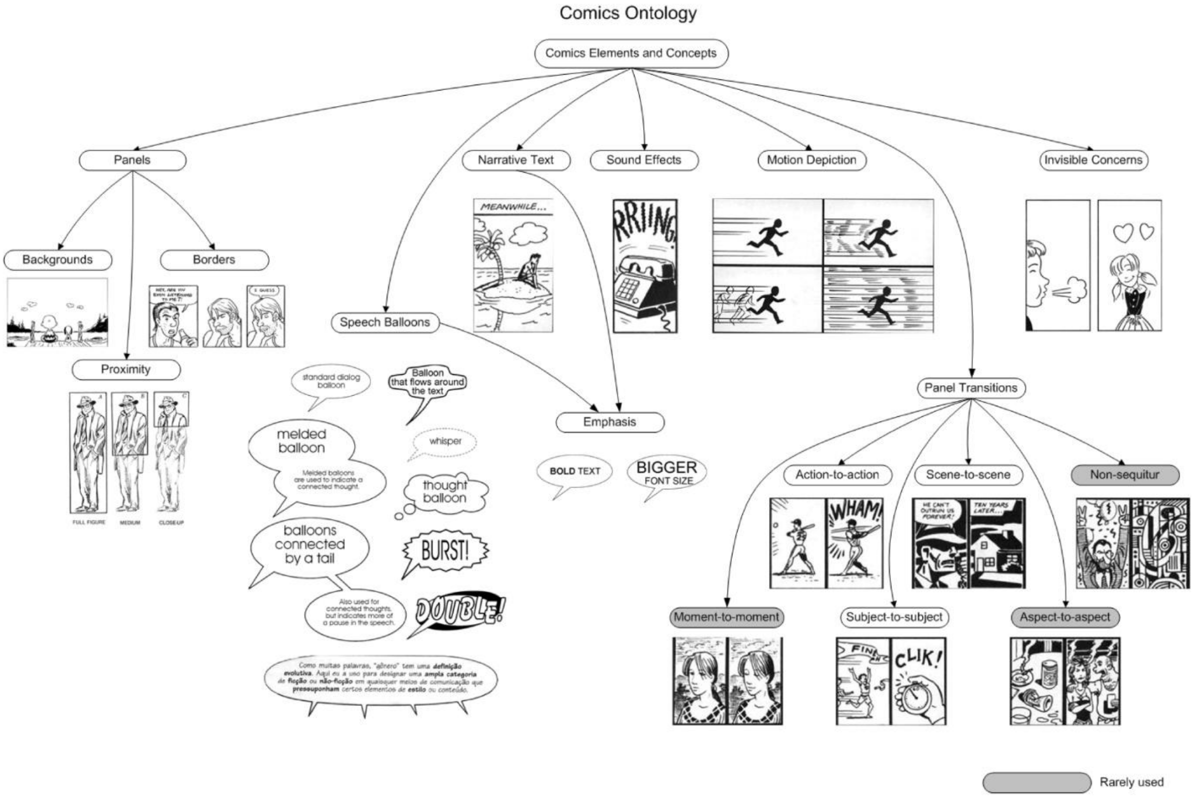

Comics Ontology
Comics Elements and Concepts
Panels
Narrative Text
Sound Effects
Motion Depiction
Invisible Concerns
Backgrounds
Borders
Proximity
FULL FIGURE
MEDIUM
CLOSE-UP
MEANWHILE...
RRIING
Speech Balloons
standard dialog balloon
Balloon that flows around the text
melded balloon
Melded balloons are used to indicate a connected thought.
whisper
thought balloon
balloons connected by a tail
BURST!
Also used for connected thoughts, but indicates more of a pause in the speech.
DOUBLE!
Como muitas palavras, "gênero" tem uma definição evolutiva. Aqui eu a uso para designar uma ampla categoria de ficção ou não-ficção em qualquer meio de comunicação que pressuponham certos elementos de estilo ou conteúdo.
Emphasis
BOLD TEXT
BIGGER FONT SIZE
Panel Transitions
Action-to-action
WHAM!
Scene-to-scene
HE CAN'T OUTRUN US FOREVER!
TEN YEARS LATER...
Non-sequitur
Moment-to-moment
Subject-to-subject
CLIK!
Aspect-to-aspect
Rarely used

Figure 2 Part of the Comics Ontology created by Alves

2.2.1 Graphical terms

To convey abstract ideas (like emotion or time) or information that is transferred through other senses than the eye, the comics artist needs a way to express himself. In comics there are no rules about what is and what is not possible in graphic representation of some content, though by general adaption of cartoonists we can couple meaning to the various icons in comics. For example the speech balloon; when the speech balloon is drawn with a dotted line, we automatically interpret it as whispering. Some artist once thought that this would be a good representation of whispering. Other artists adopted the technique and today it's practically a rule that this representation indicates whispering. Nevertheless an infinite range of possibilities of graphically communicating ideas remain.

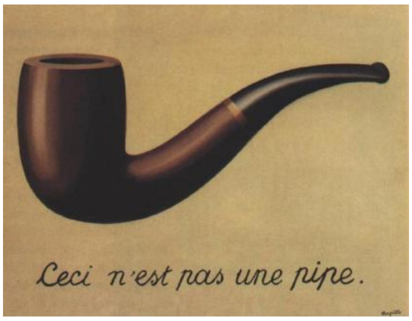


Figure 3 This is not a pipe. It's a copy of a painting of a pipe.

Icon

In his book “Understanding Comics” (1993) McCloud explains the *icon* in great depth. Here I will summarize his thoughts on icons.

An icon is any image used to represent a person, place, thing or idea. The font, in which this text is formatted, are all images of letters of the alphabet; icons of invisible ideas. The pipe in Figure 3 is not really a pipe; it’s not even a painting of a pipe. It’s a printed copy of a digital copy of a painting of a pipe. At least I think it is. Maybe it’s a printed copy of a digital copy of a scan of an analog photograph of a painting of a sculpture of a pipe. Anyhow, it is ultimately an iconic representation of a pipe. If a drawing becomes more abstract, it becomes more *iconic*. The more iconic a drawing is, the more universal it becomes. The reader will identify him- or herself with iconic characters. The most iconic drawn face could represent any human being.

McCloud (1993) explains how the icon has helped comics obtain its position in so many cultures:

> “Storytellers in all media know that a sure indicator of audience involvement is the degree to which the audience identifies with a story’s character. And since viewer-identification is a specialty of cartooning, cartoons have historically held an advantage in breaking into world popular culture.” (p. 42)

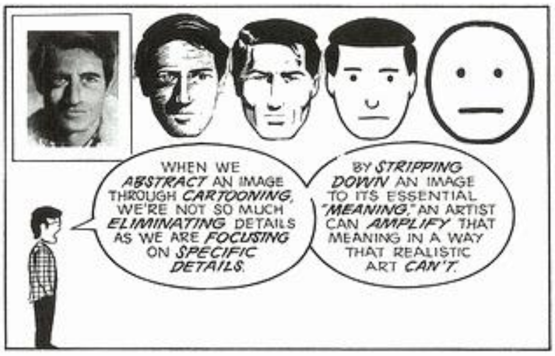

WHEN WE ABSTRACT AN IMAGE THROUGH CARTOONING, WE'RE NOT SO MUCH ELIMINATING DETAILS AS WE ARE FOCUSING ON SPECIFIC DETAILS.
BY STRIPPING DOWN AN IMAGE TO ITS ESSENTIAL "MEANING," AN ARTIST CAN AMPLIFY THAT MEANING IN A WAY THAT REALISTIC ART CAN'T.

Figure 4 Going from realistic to iconic

Even more iconic than the most iconic drawing of a face (as the right most face in Figure 4) is the word “FACE”, which also represents the face (or for example Figure 5). Further down the axis of reality to meaning is an explanation: “two eyes, one nose, one mouth” (p. 49), and then a poem, describing the face; “Thy youth’s proud livery, so gaz’d on now”[1] (p. 49).

[1] From Shakespeare’s sonnet II

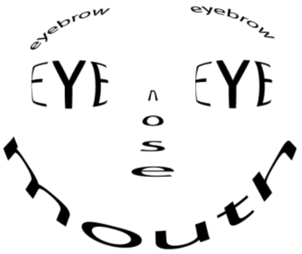


Figure 5 A very iconic face

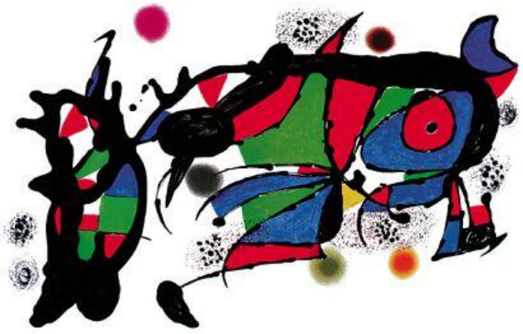


Figure 6 "Obra de Joan Miró" - an (non-iconic) abstract painting by Joan Miró

This axis, from reality to meaning, is one of two axes McCLoud describes. The other leads to non-iconic abstraction, where “no attempt is made to cling to resemblance *OR* meaning” (p. 50, see for example Figure 6). Given these two axes McCloud creates the Picture Plane that represents the total pictorial vocabulary of any of the visual arts (see Figure 7).

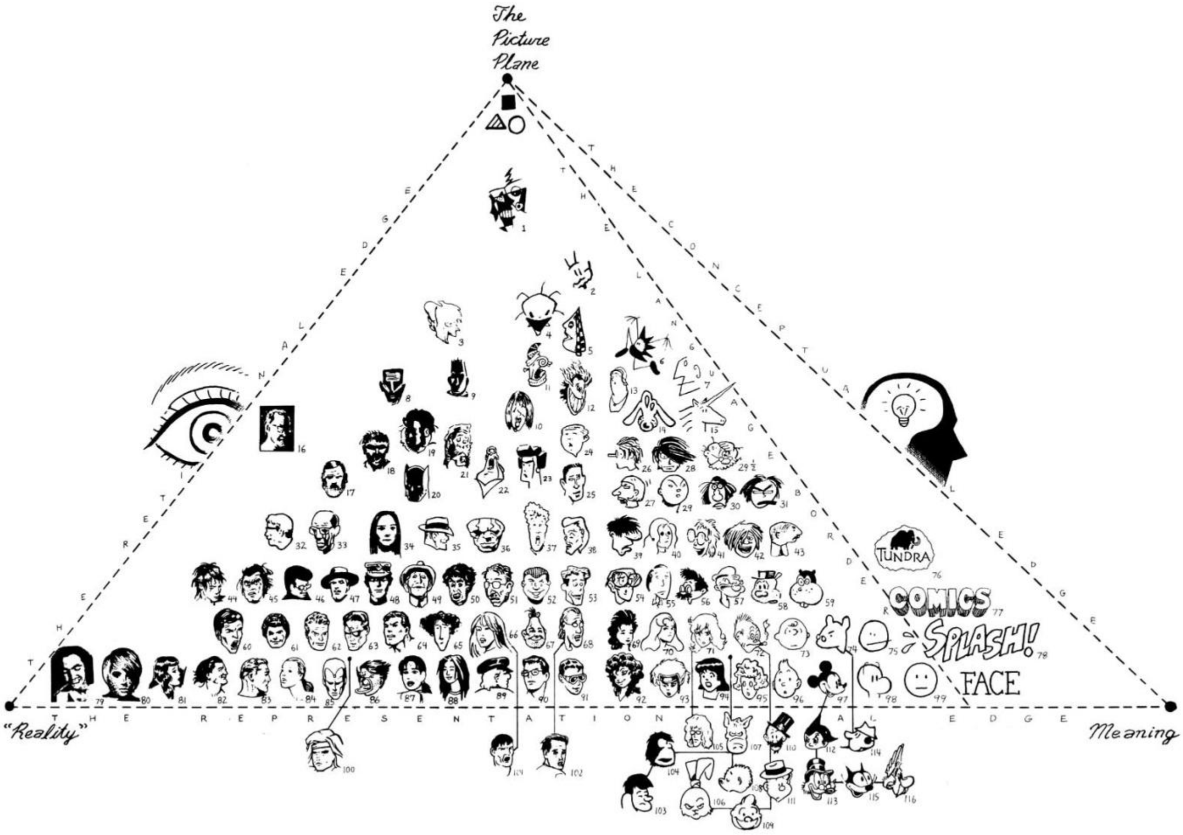


Figure 7 The language of comics

Panel

According to the definition of comics (see section 2.1), every comic contains juxtaposed pictorial and other images, put in deliberate sequence. This means that every comic contains two or more images, which can be distinguished from each other, in order to see the deliberate sequence. To create this distinction, panels are used. Though the panel is often overlooked as one of comics’ most important icons, it divides a flow of events in time and/or space into separate moments (McCloud, 1993, p. 98, 99). Often a panel doesn’t depict a single moment as in photographs, because each sound (through speech or sound effect) and action-reaction has a certain duration. See for example Figure 8 where a panel is drawn, which contents clearly doesn't encompass one single moment. Instead each sound and movement takes time, and each event causes another event until finally the rightmost man is hit by the bowling ball. In fact almost each panel in a comic conveys something that takes time to complete, some maybe milliseconds, others maybe hours or days. But of course there are panels that are like still photographs, capturing one single moment.

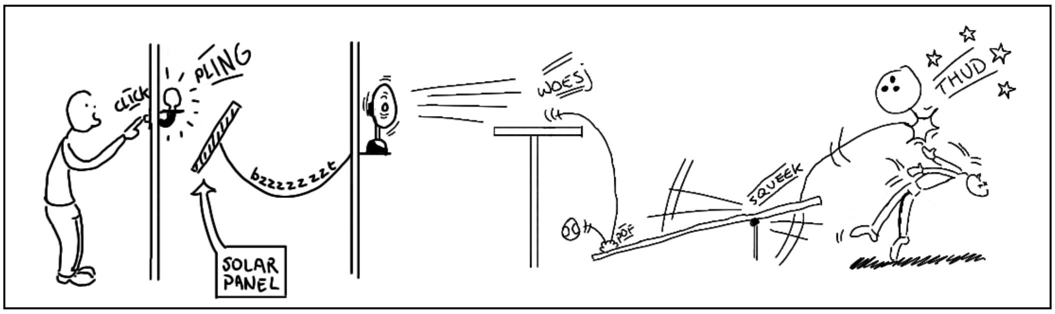


Figure 8 In this panel time passes from left to right

So the *contents* of a panel define the duration of time and the dimension of space, rather than the panel itself. Though panel shapes can vary greatly and affect the reading experience. As said earlier, there are no explicit rules in comic creation, and so there are no rules for border creation. But a sense of timelessness can be created when (part of) the border is omitted. An effect called bleeding, where the contents of a panel are no longer contained by the panel itself, but run off the edge of the page, can set the mood, or a sense of place for whole scenes. The size of a panel also influences the importance of the panel in the story. For example see Figure 9, where a change of scenery is

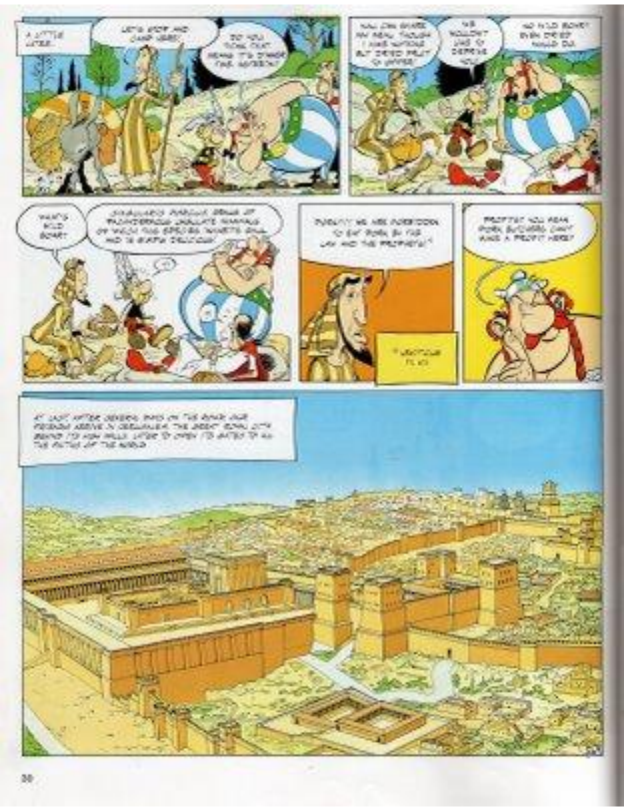


Figure 9 A page of the comic "*L'odyssée d'Astérix*" (1981) of Asterix^{TM}

introduced with a half page sized panel. This panel introduces a new scene, where Asterix, Obelix and two other characters finished their journey to Jerusalem, and now continue their search for oil. The size creates a sense of timelessness which will linger in the reader's mind. Its presence may be felt in the panels which follow it (based on McCloud, 1993).

Background

Backgrounds of a panel are multi-purpose. They can show the environment where the story currently takes place, or they can express invisible ideas, like emotions (see Figure 10).

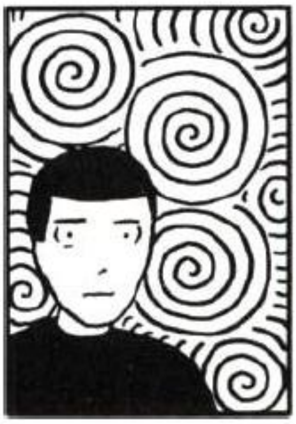


Figure 10 Background used to show the emotion of the character (McCloud, 1993, p. 312)

Border

The border of a panel has two characteristic properties: shape and character. Normally the shape of a panel is rectangular, and its character a single line with some constant thickness, but of course these two can be changed. Both shape and character of a border influence the way a reader interprets its content (see Figure 11).

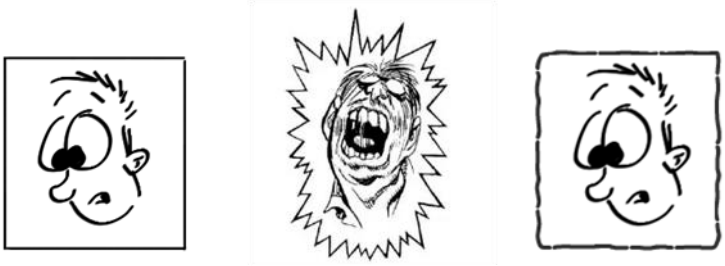


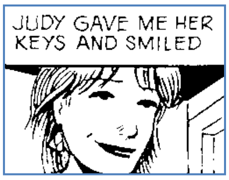
Figure 11 Changing shape and character of the border

Text

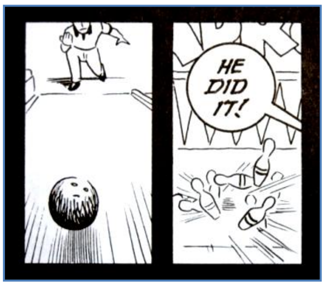
Text is used in many ways in comics; in speech balloons, in narrative boxes, in sound effects, and more. McCloud describes some combinations how word and pictures can be combined:

Word specific

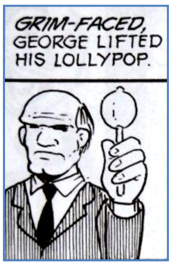
Pictures illustrate but don't significantly add to a largely complete text.



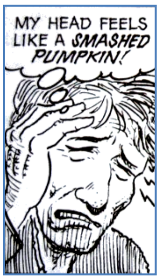
Picture specific
Words do little more than add a soundtrack to a visually told sequence.



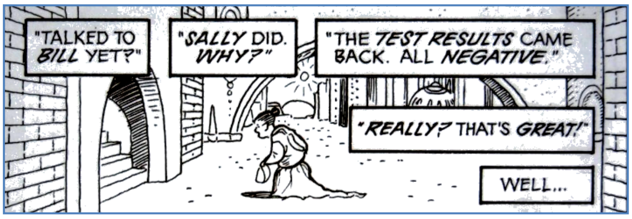
Duo-specific
Both pictures and words tell essentially the same story.



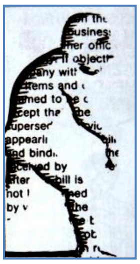
Additive
Words amplify or elaborate on an image, or vice versa.



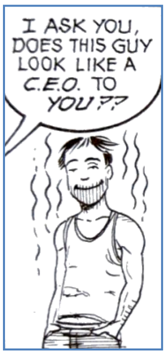
Parallel
Words and pictures seem to follow very different courses, without intersecting.



Montage
Words are treated as an integral parts of the picture.



Interdependent
Words and pictures go hand in hand to convey an idea that neither could do alone.



The interdependent combination is perhaps the most common in comics.

Sound

Sound is anything that we perceive as sound through reading comics; speech balloons and sound effects.

Speech Balloon

When a character says something in comics, this is caught in speech balloons (or word balloons). The variations of speech balloons are virtually infinite. Borders can differ according to style, or to emphasize its content or function (see Figure 12). Also the content of a speech balloon can vary. Bolder font can show that these words are pronounced louder, or that they are shouted. Symbols can be used to cover cursing or the non-verbal, like sleeping sounds, or when a character is speechless or amazed. Even coloring is used to create a certain idea about the emotion in which the speech act is performed (e.g. green when someone is envious).

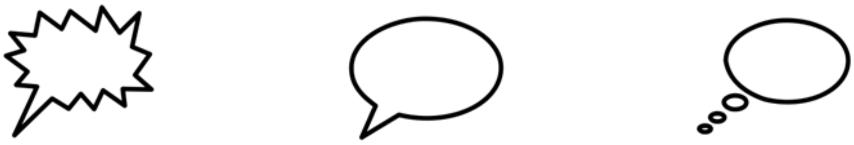


Figure 12 Various border styles to distinguish function. From left to right: shouting, talking, thinking.

The placing of the balloons indicates the order of occurrence. Balloons can be connected, or melted to indicate the flow of a conversation. As shown in Figure 13 the melted balloons create a little pause in the reader's mind between the two phrases, and the connected balloons give a different idea about the flow of the conversation between two characters than if the two balloons were separated.

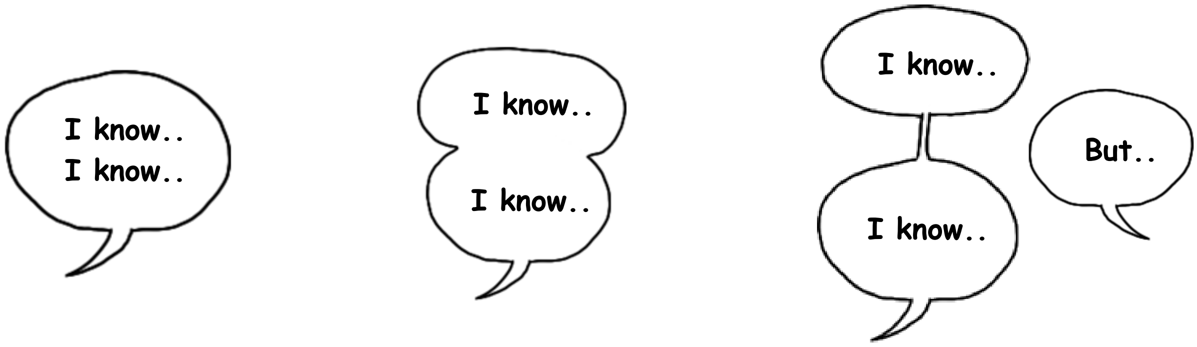


Figure 13 Changing the flow of a conversation by melting and connecting balloons. From left to right: normal balloon, melted balloons, connected balloons.

Sound effects

Sound effects are generally used outside speech balloons, but can also be used inside balloons. They describe the sound that is generated by some source. A lot of sound effects are generally accepted, like “poof” and “pow” in American action comics. Position, coloring, form, style, etc. are important to give a good feel about essence of the sound (see Figure 14).

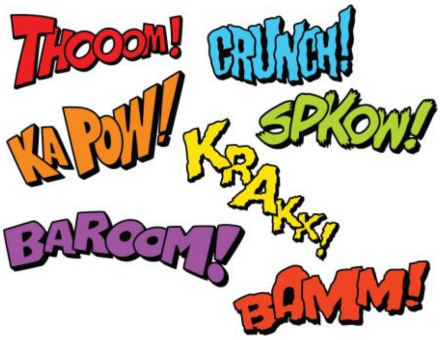


Figure 14 Various sound effects

Narrative Text

Sometimes a panel is introduced by a narrator, creating the context in which the events, contained by that panel, have to be placed. As in movies the style of narration is totally dependent on the creator’s desires; e.g. an objective narrator, talking in third person, or one of the characters, telling his own story. The narrative text is often placed at the top of the panel in a rectangular box (see Figure 15).

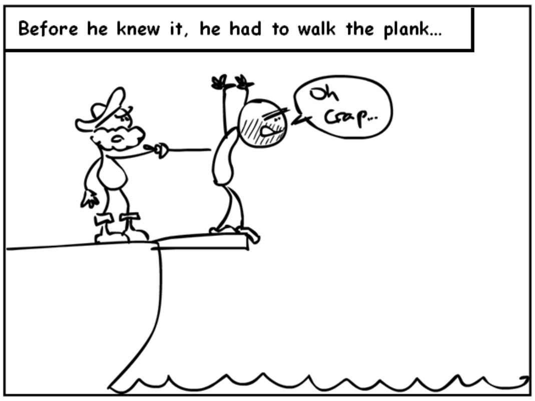


Figure 15 *A panel with narrative text*

Motion Line

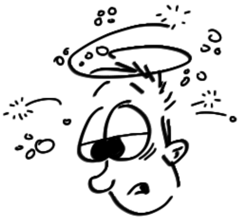
To show motion, comics contain motion lines or zip-ribbons, as they are sometimes called (McCloud, 1993, p. 111). There are multiple approaches to showing motion, but all involve the use of lines to show the path of motion of a certain object or character.

Symbolia

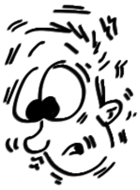
Symbolia is the language of comics used for invisible ideas. American cartoonist Mort Walker wrote the book "*The Lexicon of Comicana*" [1] in 1980, intended as a satirical piece for the National Cartoonists Society, though it eventually became the source of the lexicon of the symbols used in comics. Sometimes the term *Emanata* is used instead of Symbolia, which is in fact just one of many possible forms of Symbolia.

A few terms of this lexicon (Wikipedia, The Lexicon of Comicana):

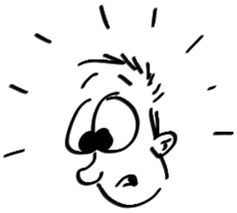
Squeens – Little starbursts or circles that signify intoxication, dizziness, or sickness



Agitrons – Wiggly lines around an object that is shaking



Emanata – Lines drawn around the head to indicate shock or surprise



Character

A character is a part or a role in a drama, story, play, film, etc. Dictionary.com gives the following description (only relevant information is stated):

char·ac·ter /kærɪktər/ [kar-ik-ter]
-noun

…

10. a person represented in a drama, story, etc.

11. a part or role, as in a play or film.

…

-adjective

23. Theater.

a) (of a part or role) representing a personality type, esp. by emphasizing distinctive traits, as language, mannerisms, physical makeup, etc.

b) (of an actor or actress) acting or specializing in such roles.

To distinguish between characters and other elements in a comic, we need a definition. Based on the definition given by dictionary.com I state that a character performs any part or role that has any form of influence on the story in an autonomous, self-decisive manner. This means that falling trees, that influence the stories, are not characters, because they didn't choose to fall.

Object

An object is everything that is not background or a character. Objects can be manipulated and/or participate in events, as opposed to backgrounds. A crowd, not self-decisive, could be classified as an object given this explanation.

[1] I found the information I used in this section on Wikipedia, as I could not get a copy of the original work.

Text Character

Many comics also contain text; in speech balloons, narrator boxes, sound effects, etc. Text contains text characters, which make up the words. These text characters can have any font, shape and color and therefore each text character is an icon in itself; a (partial) representation of the abstract idea of language.

2.2.2 Abstract Terms

These terms describe the various abstract things in comics; those not visible, but nevertheless vital to the idea of comics.

Closure

Closure is the phenomenon of observing the parts, but perceiving the whole. What we do not perceive physically, we complete mentally, based on past experience. If you can't see your whole car, because a van is parked in front of it, blocking your sight partly, you mentally complete the information. You do not assume that the part you do not see, does not exist (see Figure 16).

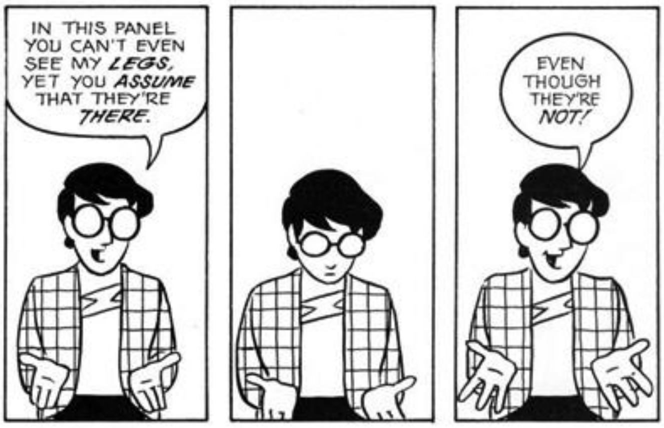

IN THIS PANEL YOU CAN'T EVEN SEE MY LEGS, YET YOU ASSUME THAT THEY'RE THERE.
EVEN THOUGH THEY'RE NOT!

Figure 16 Closure; mentally completing the missing information

Gutter

It is closure that helps the reader to understand relations between subsequent panels, to fill in missing information; the space between subsequent panels is called "*the gutter*". Though the gutter is visible, its abstract function makes it interesting; it's where closure takes place over time and space, it's where we connect the separate images and make it a whole (see Figure 18).

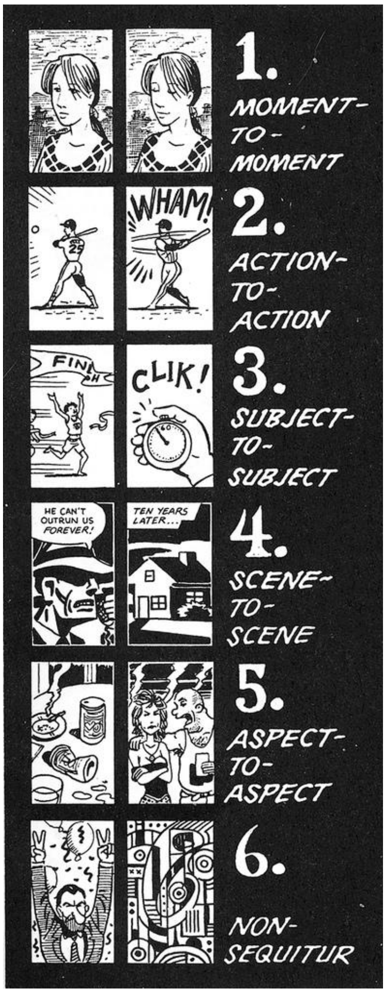


Figure 17 Panel to panel transitions

McCloud (1993) explains the function of the gutter as follows:

> "Comic panels fracture both time and space, offering a jagged, staccato rhythm of unconnected moments. But closure allows us to connect these moments and mentally construct a continuous, unified reality." (p.67)

This gutter is placed between every subsequent panel. These panel to panel transitions have been identified.

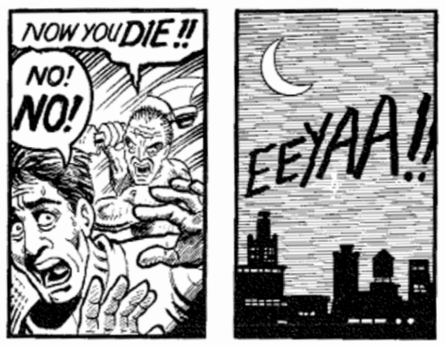


Figure 18 *The reader is responsible for what happens in the gutter*

Panel to Panel transition

Each panel transits into the next, causing some relationship between the two panels. These panel to panel transitions (as shown in Figure 17) are described in McCloud (1993):

1. ***Moment-to-moment***: describes two subsequent moments in time with little difference, requires very little closure
2. ***Action-to-action***: features a single subject in distinct action to action progression
3. ***Subject-to-subject***: shows different subjects within the same scene or idea. Note the degree of reader involvement necessary to render these transitions meaningful
4. ***Scene-to-scene***: transports the reader across significant distances of time and space. Often requires deductive reasoning.
5. ***Aspect-to-aspect***: shows different aspects of a place, idea or mood, while bypassing time.
6. ***Non-sequitur***: offers no logical relationship between panels whatsoever.

McCloud questions the existence of the relationship 'non-sequitur', as the possibility that there is totally no relationship at all between two panels seems so unlikely. Nevertheless I think that in an endless possibility of juxtaposed panels, the possibility *does* exist that two panels are completely unrelated, even though in practice we'll probably never see it.

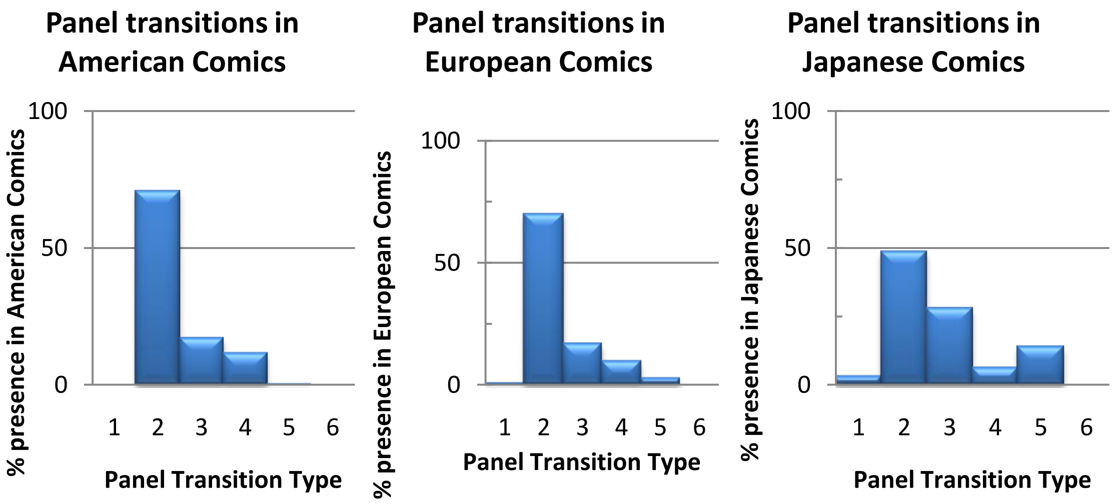


Figure 19 Panel transition type presence in American, European and Japanese comics

McCloud looked at how often each transition is used in comics. He looked at American comics, European Comics and Japanese Comics (see Figure 19). It seemed that the usage percentage of the various transition types in American and European comics are almost the same, though in Japanese comics other transition types are used more often, and others less often. The subject-to-subject transition is used far more often compared with the western comics. And though not much, more than in western comics, the moment-to-moment transition is used. But the biggest difference is the substantial presence of the aspect-to-aspect transition in Japanese comics.

Comic

A comic is an instance of comics. It is a whole story told in juxtaposed pictorial and other images in deliberate sequence.

Scene

A scene is a subdivision in the story being told. It's a sequence of events that occur in a certain time-span and in a certain space. When the reader is transported a significant distance in time and/or space, a new scene has been initiated.

Invisible Ideas

Invisible ideas are important aspects of a story that cannot be sensed by any of the 5 human senses. They are emotions, concerns, thoughts, tensions, moods, settings, motions, etc. Though some of course can be deducted from expressions, for example tears may indicate a character is sad. Though tears and sadness are not the same. These invisible ideas are described by writers of books, told by narrators in movies, and shown in comics through various channels; the choice of background and border, through narrative text, or by use of Symbolia.

2.3 Summary

In this chapter we looked at the history of comics, and stated a definition of comics. We also looked at all the elements comics consist of, and now we are able to distinguish between the various elements. In the next chapter I propose an ontology which consists of these elements, this will aid me further on in this research to design the system.

Chapter 3 Automating the Comic Creation Process

In this chapter research that has been done in the field of automatic comic creation is looked at. First we look at some comic generation systems, then at some formats in which comics can be described or annotated, and finally at some graphical methods of how to create the graphical contents of comics. Research questions RQ I and RQ IV will be addressed here.

First a word of notice: The words comic and cartoon are often used interchangeably, but there is a difference. A comic is, as described by McCloud, a story consisting of two or more pictures. A cartoon on the other hand consists of only one panel (p. 154, Le Roux and van Muylwijck, 1993). Nevertheless, various researchers that are quoted in this thesis interchangeably use the word cartoon and comic, and I will quote them literally nonetheless.

3.1 Comic Generation Systems

Since some time computer science is also getting involved in the world of comics. In many different ways researchers try to get a grasp on the many aspects of automatically creating comics; which content is used, how is that content created, how are the graphics created, etc. In this section I will discuss various systems that have been designed that generate comics or comic elements.

3.1.1 Comic Chat

Kurlander et al (1996) created a system called Comic Chat (see Figure 20) that visualizes chat sessions as comics. In order to realize this, the system chooses which avatars of the chat participants to include in each panel. Depending on the conversation it determines the placement and orientation of the avatars. Then it constructs various word balloons and places these balloons in such a matter that reading order is preserved. The system also chooses an appropriate camera zoom factor for each panel, and decides when to begin a new panel (i.e. how many balloons a panel can contain before a new panel must be created). Depending on some relatively simple text analyses, it chooses gestures and expressions for each character, which the participants can override. Lastly it can adjust the background or scene elements to reflect the topic of the conversation (Kurlander et al, 1996).

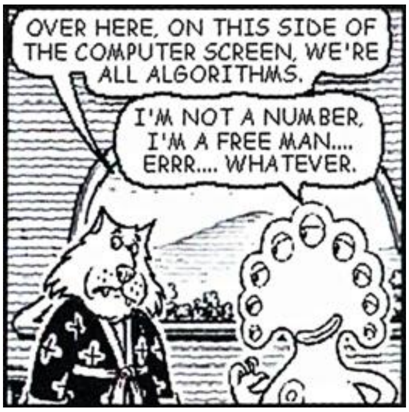

OVER HERE, ON THIS SIDE OF THE COMPUTER SCREEN, WE'RE ALL ALGORITHMS.
I'M NOT A NUMBER, I'M A FREE MAN.... ERRR.... WHATEVER.

Figure 20 A panel from a comic generated by Comic Chat

3.1.2 Pink Panther™

In the field of Digital Storytelling, Cavazza et al. (2003) aim at creating comic situations for which they use comics as an investigation field (see Figure 21). They investigate the form of planning needed to create the content of a comic. They state that many cartoons "resort to dramatization of problem solving, or the pursuit of a single goal (e.g. catching the "roadrunner") as a main narrative mechanism" (p. 1). Their hypothesis is that comic situations arise from plan failure which is applied to creating comics of the Pink Panther™. The graphical content of each panel is created by using screenshots of the 3D environment in which the stories take place.

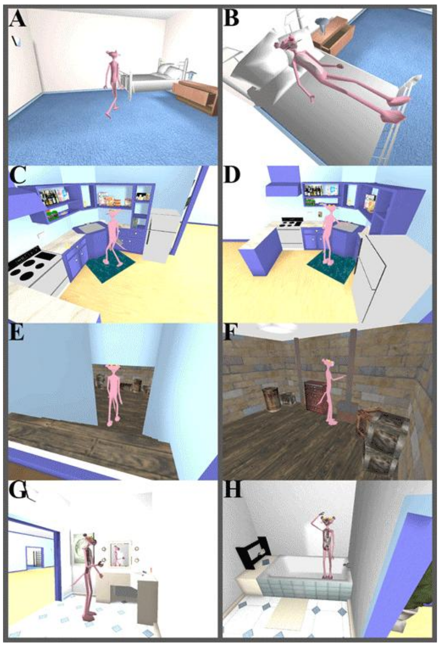
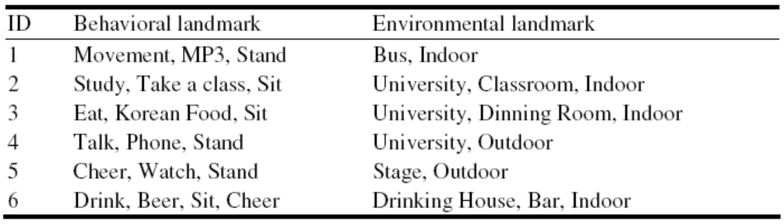


Figure 21 A comic created by the system of Cavazza et al. (2003)

3.1.3 Comic Diary

Song et al. (2006) describe a system that creates comics given inferred context data of someone's behaviour, emotions and surroundings, which are obtained via various sensors, such as GPS, visual and audio sensors, etc. They generate "cartoon cuts by applying semantic similarity between the context events and the images. These cartoon cuts are arranged into a cartoon story by using consistency constraints" (p. 451). The function of the created comics is to share one's daily life with other people, as people do now via web logs and such. An example of such a comic is given in Figure 22 which is a summary of the events summarized in Table 1.

Table 1 *Summary of events of some individual*



ID	Behavioral landmark	Environmental landmark
1	Movement, MP3, Stand	Bus, Indoor
2	Study, Take a class, Sit	University, Classroom, Indoor
3	Eat, Korean Food, Sit	University, Dinning Room, Indoor
4	Talk, Phone, Stand	University, Outdoor
5	Cheer, Watch, Stand	Stage, Outdoor
6	Drink, Beer, Sit, Cheer	Drinking House, Bar, Indoor

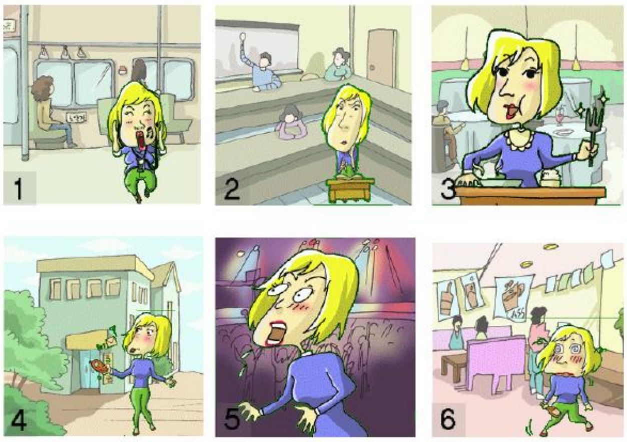


Figure 22 *Generated comic given the events summarized in Table 1*

3.1.4 Game Session Comic

Shamir et al. (2006) created a system that converts a computer game session (Doom[1]) to a comic that summarizes the story's main events. Using certain algorithms they developed on a log of a game session, they determine scenes and important interactions within such a game session. These interactions are then converted to comics using the game's own engine and image processing techniques to give it a more comic feel. Text balloons are added when a conversation occurs in the log. Finally the page layout is performed using a "heuristic image-layout algorithm that breaks the symmetry to create a comic look and feel" (p. 59). To simplify the problem the rows are of the same height, and the columns are resized as stated before. For more in-depth information about this heuristic image-layout algorithm I refer to page 60 in aforementioned paper. A resulting comic of this system is shown in Figure 28 on page 28 in this thesis.

3.1.5 Carrie the Cartoonist

A group of students at the University of Twente have created a comic generation system for a design project (te Brinke et al., 2008). It is called "Carrie the Cartoonist" and is based on content created by the Virtual Storyteller (more on the Virtual Storyteller in section 4.2). Their design is based on another design project called "Annie the Animator" in which van Hurne et al. (2007) attempted to create a system that generates animations based on content created by the Virtual Storyteller. Both use a director agent and a presenter to translate the fabula to the ultimate presentation form (either comic or animation). The director receives the fabula and creates a script file which is passed on to the presenter. The presenter then converts the script into the presentation form. The animator was never completely finished, but "A small subset of the available fabula actions will be accepted by the system and correctly animated" (Hurne et al, 2007, p. 46). Carrie the Cartoonist is able to create comics, though with certain limitations; as shown in Figure 23 actions are not defined, and the characters are completely pre-drawn (where e.g. hair colour or eyeglasses can be altered, see te Brinke et al., 2008, p. 13). Therefore the characters do not have the correct poses to depict actions.

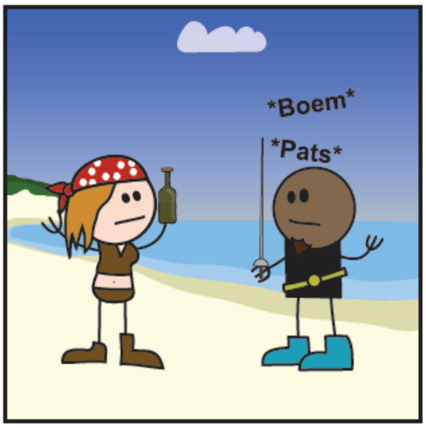


Figure 23 *A fight action in Carrie the Cartoonist*

3.1.6 Cartoon Face

Though these approaches create complete comics, specific parts of the process of comic creation can also be researched. Liu et al. (2005) presented a prototype system for use on a mobile phone; it creates a cartoon face from a real facial photograph and translates text message to multimedia animation. To do this, the system first extracts the feature points of the face. Then, by extracting facial edges, the illustration is generated (see Figure 24). The third step is caricature generation from the illustration (see Figure 25). This is sent to a mobile phone, followed by text and animation data to animate the caricature.

[1] see http://zdoom.org

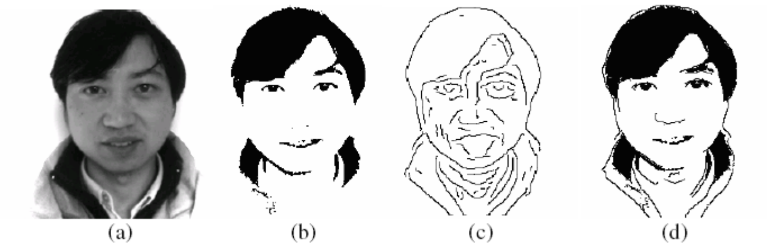


Figure 24 Liu et al. (2005): extracting facial edges from image (a) results in image (d) by multiplying results (b) and (c)

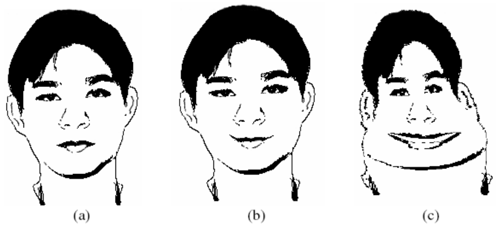


Figure 25 Liu et al. (2005): caricature generation given original image (a)

3.1.7 Comics2D

Another system which incorporates the whole process of comic creation is presented by Alves et al. (2008). The system FearNot! is used as the source for the comic creation process. FearNot! is an application in which autonomous agents create improvised bullying situations. A user must then help the bullied character by advising him what to do next (Aylett et al., 2007). Alves' system "analyses story logs [resulting from a user session], looks at the characters' emotional information to understand their actions and their importance in the story, selects the most important events and creates comic strips" (Alves et al., 2008, p. 1269). Everything leading to, or resulting from a strong emotion is considered important to the story and thus visualized in the comic. The content created by FearNot!, is translated to an XML-based mark-up language, which is then transformed into graphics.

The work in automatic comic creation done by Alves is extensively described in his thesis (Alves, 2007), and more briefly described in Alves et al. (2007). His approach lies close to the approach I will describe in this thesis, as his system, called Comics2D, is also dependent on a content/story generation system.

In his thesis, Alves describes that his motivation to make a system that generates comics was that comics are a privileged medium for the visual representation of summaries (p. 1). His goal was to make a system that could generate comics independent of the source of the story (p. 2), being capable of producing them in different visual styles (p. 2). To reach this goal Alves designed a new comics description language which is able to describe a certain comic in high detail. This language is called CSDL: Comic Strip Description Language.

3.1.8 Three Steps to Creating a Comic

When we look at these various systems, we see that all need some content to portray; E.g. the system designed by Kurlander et al. (1996) uses a chat session as content, Cavazza et al. (2003) use Heuristic Search planning techniques to create content (p. 2), Song et al. (2006) use collected data about someone's daily behaviour, Shamir et al. (2006) use a 3D game session log as content and the Comics2D system designed by Alves (2007) relies on content created by the system FearNot!. Next is translating the content to a comic; E.g. Kurlander et al. (1996) use chat-specific conventions like emoticons, chat acronyms (e.g. LOL, ROFL, BRB, IMHO, etc), typesetting (all caps), punctuation, greetings, self references and references to others to determine what emotion and what pose should be chosen for the character to accompany a certain text unit. The last step is graphic creation, e.g.

the system created by Shamir et al. (2006) uses screenshots as the graphical content of each panel. To summarize; comic creation involves three steps:

1. content creation
2. translation of content to a (formal) comics description
3. graphics creation

These steps form an answer to research question RQ I. To look at the first step, content creation, in great depth is beyond the scope of this thesis, though the story content creation system which is used for the Comics system is presented in the next chapter. The second and third steps are looked at in more detail in the next sections.

3.2 Translation of Story Content to Comic Content

The content of a story is a very abstract idea and its representation can vary enormously; it can be in your head, written in plain English or formally described in some XML based language. To create a comic depicting the content of a certain story, a solution would be to first translate the story content into a formal comics format, which can ultimately be transformed into graphics.

The story content for Alves' system (Comics2D) is created by FearNot!. Everything leading to, or resulting from a strong emotion is considered important to the story and so must be visualized in the comic. To be able to do this, he designed a description language for comics. But first let's take a deeper look at existing markup languages which are created to describe comics too:

- ***CBML***: Comic Book Markup Language (www.cbml.org)
- ***ComicsML***: Comics Markup Language (comicsml.jmac.org)

These two languages are both intended to describe already existing comics for digital cataloging.

As described on the website, CBML is a very descriptive XML-based language which is intended to describe comic books in their totality: ads, advisories, publication information, the characters of the comic, the panels in the comic, etc. It is primarily intended for existing, traditional, American style comic books.

ComicsML is also an XML-based language intended to describe (digital) comics. Information about the comic in general, the creator, or other persons involved, the comic itself and more can be described. A description in ComicsML is linked to the original comic, and so not intended to replace it. In ComicsML, as opposed to CBML, actions can be described, though in an informal manner.

Based on findings from the book Understanding Comics by Scott McCLoud (1993), Alves designed CSDL: Comic Strip Description Language; an XML-based mark-up language:

```
<comic>
      <scene>
            Arguments: ID, Description
            <panel>
                  Arguments: Border, Duration, Importance
                  <background>
                        Argument: ID
                        <object>
                              Arguments: ID, Value
                              <location>
                  <character>
                        Argument: ID
                        <emotion>
                        <pose>
                        <gaze>
                        <location>
                        <action>
                        Argument: Value
                        <target>
                            Argument: Value
                    <object>
                        Arguments: ID, Value
                        <location>
                    <concern>
                <narrative>
                    <phrase>
                        <text>
                            Argument: Type
                        <owner>
                <balloon>
                    Arguments: Type, ID
                    <owner>
                    <phrase>
                        <text>
                            Argument: Type
                    <connection>
                        Argument: Target
                <soundEffect>
                    <owner>
                    <text>
                        Argument: Type
                <camera>
                    Argument: Type
                        <target>
                            Argument: value
            <transition>
                Argument: Value
```

Figure 26 *Overview of the CSDL format, designed by Alves (2007)*

A comic described in CSDL is divided in scenes, which are divided in panels. Panels are composed of a background, one or more characters, narrative boxes, balloons, sound effects and camera information (zoom-level and target, e.g. face or pocket). A character is described using emotions (happy, sad, etc.), pose (standing, sitting, etc.), gaze (where is the character looking at), location of the character in the panel, action the character is performing, objects the character has, and an invisible concern (dizzy, smelly, etc.). Finally there is a transition between the panels (see section 2.2.2). CSDL follows a loose typing approach, which means that the values of the various parameters are not pre-defined.

3.3 Graphics Creation

When generating comics a number of decisions have to be made in the approach of how to generate the graphics. Two methods of creating the panels are distinguished: the Screenshot method and the Composition method.

3.3.1 Screenshot vs. Composition Method

Alves distinguished between two graphical methods: the *composition method* and the sc*reenshot method*. The composition method makes use of composing multiple individual images into one comic's panel. A good example of a system that uses this method is *Comic Chat* (Kurlander et al, 1996), see Figure 27. The screenshot method makes use of existing dynamical graphics of the content to be visualized. Shamir et al (2006) presented a system that creates a summary of a 3D computer game session using a comic, non-photorealistic rendering style to create a comic, see Figure 28.

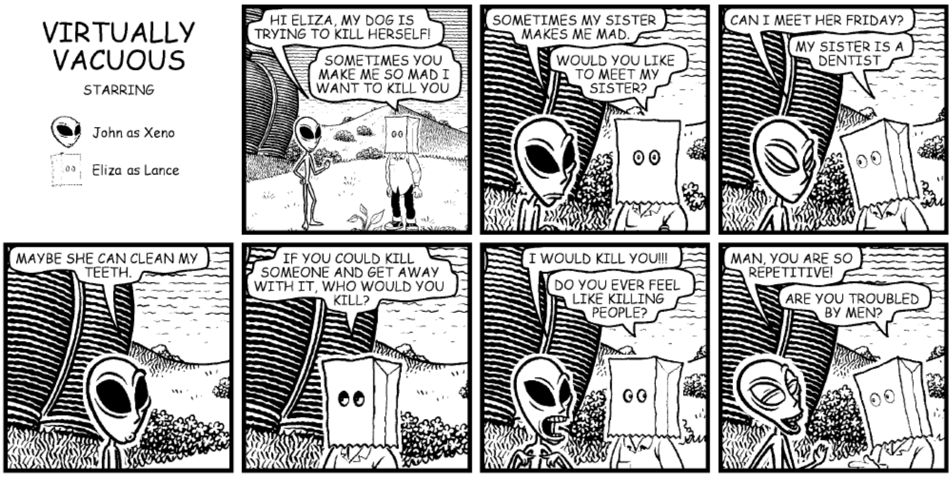

VIRTUALLY VACUOUS
STARRING
John as Xeno
Eliza as Lance
HI ELIZA, MY DOG IS TRYING TO KILL HERSELF!
SOMETIMES YOU MAKE ME SO MAD I WANT TO KILL YOU
SOMETIMES MY SISTER MAKES ME MAD.
WOULD YOU LIKE TO MEET MY SISTER?
CAN I MEET HER FRIDAY?
MY SISTER IS A DENTIST
MAYBE SHE CAN CLEAN MY TEETH.
IF YOU COULD KILL SOMEONE AND GET AWAY WITH IT, WHO WOULD YOU KILL?
I WOULD KILL YOU!!!
DO YOU EVER FEEL LIKE KILLING PEOPLE?
MAN, YOU ARE SO REPETITIVE!
ARE YOU TROUBLED BY MEN?

Figure 27 An example of a translation of a chat session by Comic Chat

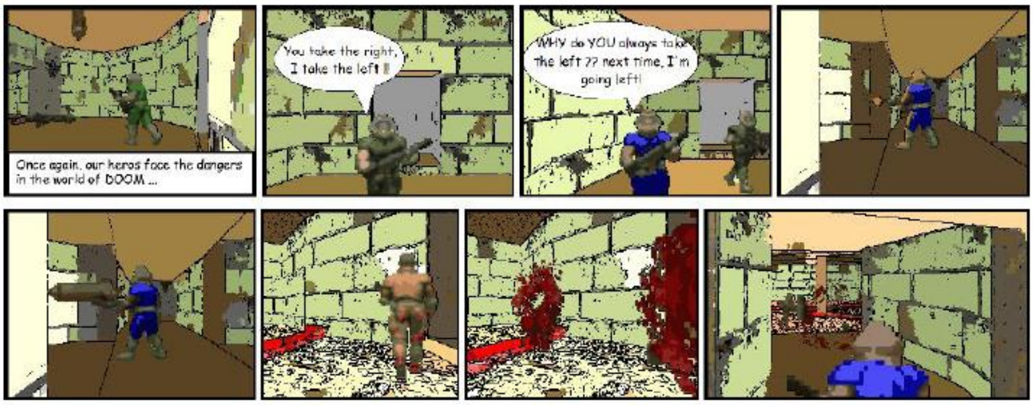

Once again, our heros face the dangers in the world of DOOM ...
You take the right, I take the left !!
WHY do YOU always take the left ?? next time, I'm going left!

Figure 28 An example of a translation of a game session by the system of Shamir et al

Alves makes an analysis of both these methods and concludes with the following advantages and disadvantages of the Composition Method (CM) and the Screenshot Method (SM):

	CM	SM
• ***Independence*** CM can be independent of whatever system generates content; SM however is restricted to the graphic engine it is applied to.	+	-
• ***Modification*** CM can generate cartoons in various drawing styles (western vs. eastern style, sketchy, Asterix & Obelix like, Walt Disney like, etc). SM is again limited to the graphic engine available. Even with being able to apply some image alternation techniques, the CM will given more ease in modification.	+	-
• ***Object and Character placement*** CM must make use of placement algorithms and make use of a state history to record the positions. SM has all scenes element already placed in the world and has access to their positions.	-	+
• ***Background*** SM automatically has the correct (right position and perspective) background given a certain camera position. CM needs to have different backgrounds for every perspective or apply algorithms to manipulate the backgrounds	-	+

according to the perspective		
• ***Camera placement*** In CM it's relatively easy, because the objects are placed before the camera. The SM method on the other hand needs a camera placement system, which is a disadvantage. Though more variety is available.	+/-	+/-
• ***Visual appeal*** CM combined with a good artist will have more appeal than images from the SM	+	-
• ***Event timing*** A certain event takes time, and picking the right moment in the SM is hard. This problem does not occur in the CM	+	-

Another aspect that can be added is:

• ***Graphic creation*** With CM all graphical elements need to be pre-drawn, though with the SM all graphics are already present.	-	+

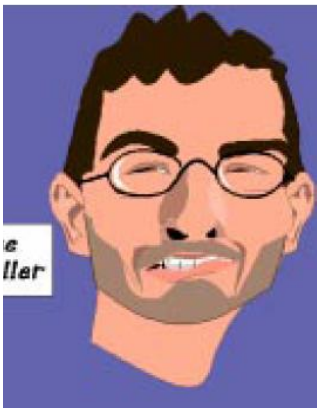


Figure 29 A face in CharToon

The advantage of camera placement in the Composition Method over camera placement in the Screenshot Method is arguable. When Alves discussed this, he probably based his theory of how relatively easy it is on another statement he makes in his thesis, namely: "In comics, the characters are usually placed unidimensionally along the base line of the panels" (p. 28). Though in a lot of (digital) comics (e.g. seen on the internet and in newspapers) this is (mainly) true, it is not true for a whole lot of comics, which makes camera placement a bigger issue for the Composition Method. Nevertheless I agree that for the manner of composition he chose, placing the characters unidimensionally along the base line, this is an easier method. And because I also use this restriction, the advantage is the same as it is for Alves. These two methods of generating graphics partially address research question RQ IV.

3.3.2 Creating and Composing the Graphics

Using the Composition Method, graphics creation can be divided into two separate steps:

1. Creating the separate graphical objects
2. Composing these separate objects into a comic

The first step can be done offline, by creating a library composed of all objects which can occur in the comics you want to create. This can be done by an artist with drawing skills to create appealing drawings, making the end result more appealing. It can also be done using software intended to help the user in making the separate objects.

The first step can also be done on the fly, creating an object when it is needed. Third party software can be used for this purpose, as for example the system presented by Liu et al (2005), which transforms a photograph of a face to a cartoon like caricature of that same face (see Figure 24). Another system that could be used is CharToon (see Figure 29), presented by Ruttkay and Noot (2000). This system is a tool to design 2½D drawings with dynamic potentials (and so one can also make animations). Though it is primarily intended to create and animate faces, this tool, and the one presented by Liu et al., can be used to generate the face of a character at the moment it is needed, as opposed to using a pre-generated library containing all the faces.

3.4 Summary

In this chapter I first gave an overview of research that has been done in the field of comics creation. From the described research and the previous chapter I concluded that the complete comics creation process involves three main steps: 1) content creation, 2) translation of content to comic and 3) graphics creation. With these three steps I addressed research question RQ I: What are the steps that need to be taken to generate comics? Then I explained these three steps, gave examples and discussed various methods. For content creation I showed the various contents that are used in some systems I discussed in this chapter. For the translation process I discussed CBML, ComicsML and CSDL. Finally for the graphics creation part I showed that there are two main possibilities: the Screenshot Method and the Composition Method. For the composition method I also discussed some possibilities of creating the graphical components. With these two methods and the discussion about the Composition Method research question RQ IV has partially been answered: What are the various ways to create comics graphics and what is the best way for the Comics system?

Chapter 4 Design: Describing Comics

In this chapter we look at how comics can be described in a formal way. I will discuss why some existing formats for describing comics are unsuitable for the Comics system. Finally I will propose a new format: ComicsDL.

4.1 Introduction

As explained in Chapter 3 Automating the Comic Creation Process (p. 26) I have broken down the comic creation process into three steps:

1. Content creation
2. Translation of content to a (formal) comic description
3. Graphics creation

The first two steps will be explained in this chapter. The third step is explained in Chapter 6 Design: ComicsDL to Comic.

The first step, content creation, is performed by the Virtual Story Teller, a system designed and in continuous development at the Human Media Interaction Group at the University of Twente (more about the Virtual Story Teller in section 4.2) .

The second step needs an intermediary format that describes the resulting comic in a formal manner. In section 3.2 I looked at some intermediary formats and in section 4.4 I'll show that these formats are not suitable to use in my system. Therefore I designed a new format which I will propose in this chapter based on an ontology which I will describe in section 4.3. Steps two and three are further examined in chapters 5 and 6 respectively.

4.2 The Virtual Storyteller

The content for the system that is presented here is generated by the Virtual Storyteller. The Virtual Storyteller (VST) is a Multi Agent System that can generate stories through use of a virtual world in which Character Agents pursue their goals. The story emerges from the actions and events that occur in the simulated virtual world (p. ii, Swartjes, 2006).

Swartjes & Theune (2006) distinguish three layers: the *fabula layer*, the *plot layer* and the *presentation layer*. Fabula is the collection of causally linked events that occurred in a virtual world during a simulation. From this fabula a consistent and coherent plot can be determined, which can ultimately be presented to a reader through some medium, in this case through comics. Swartjes (2010) designed a model which he calls the fabula model (see Figure 30). It consists of elements and relations, which connect the elements. Those elements are:

- ***Goal*** – a goal is something a character wants to achieve, e.g. eat an apple
- ***Event*** – an event is something that occurs without the intent of some character, e.g. a bridge that collapses
- ***Action*** – an action is something a character intentionally does, e.g. eat the apple
- ***Internal Element*** – an internal element is a belief or an emotion of some character
- ***Outcome*** – an outcome is something that resolves a goal, e.g. the apple is eaten. It is also possible to have a negative outcome, when the goal is not achieved.
- ***Perception*** – a perception is something a character senses (sees, hears), this helps him to make beliefs (an internal element). E.g. a character sees someone lying on the floor (a perception), and concludes he is dead (belief)
- ***Setting Element*** – a setting element introduces something in the world as a fact. E.g. a ship lies in the harbour

These fabula elements are connected by causal relations, which I will explain using examples:

- ***physically causes*** – an event can physically cause another event, for example a bolt of lightning that sets a house on fire during a thunder storm
- ***psychologically causes*** – a perception can psychologically cause an internal element, for example Juliet who gets really sad because she sees Romeo is dead
- ***motivates*** – a goal can motivate an action. For example a Leprechaun has as goal to find a treasure of gold, so he runs to the end of the rainbow
- ***enables*** – an internal element can enable a goal. For example little Linda believes they sell ice at the ice truck, so this enables the goal to go buy ice at the ice truck

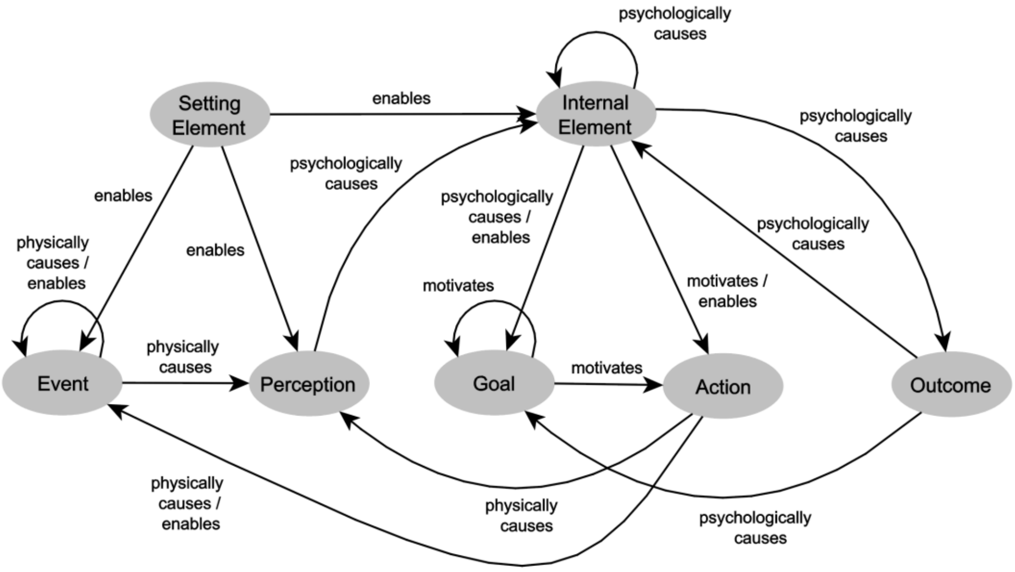


Figure 30 the fabula model; showing how elements are related

In this fabula model, all that happens in the fabula world is captured. This is the story content for the system that is presented in this thesis.

A short example of the fabula of a short story (fragment) is shown in Figure 31. The main character is called Linda and this story starts with a goal Linda has: ‘`HaveIce`’. This results in two actions; firstly to ‘`ToddleOffTo`’ the ice cream truck which causes her to see she is there, which causes her to believe she is there. This enables the second action (which is also motivated by the goal to ‘`HaveIce`’), namely to ‘`Buy`’ the ice cream. This final action causes her to see she has bought the ice cream, which causes her to believe she bought the ice cream, which ultimately results in the state of the successful achievement of the goal to ‘`HaveIce`’. If this fragment would be told in plain English this would (roughly) result in:

> *Linda wanted to have ice cream, so she toddled off to the ice cream truck. After she arrived, she bought an ice cream.*

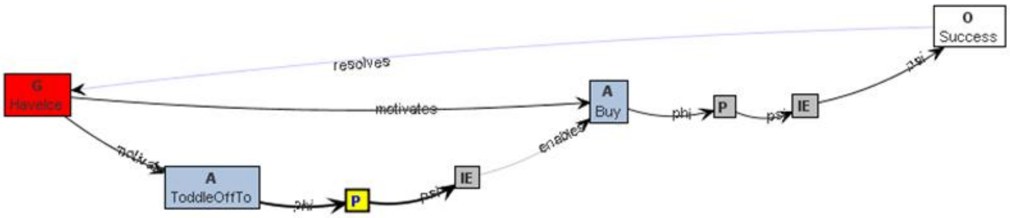


Figure 31 *A fabula graph*

4.3 A Comics Ontology

To describe a comic in some intermediate format, a formal representation of the various elements comics consist of should be formed. As a first step I propose an ontology in which the comics elements described in section 2.2 are hierarchically placed. Alves also proposed an ontology which I will first shortly discuss, after which I will propose a revised version for use in Comics. A part of the comics ontology created by Alves is shown in Figure 2 on page 10 in this thesis.

An ontology is a descriptive diagram in which various elements are connected through relationships, e.g. is-a, consists-of, part-of, etc. In Alves' ontology, presented in his thesis, the relations between the various elements are not described. The top node 'Comic Elements and Concepts' has seven children, all of which are subclasses of this parent node. One of its children, 'Panel' has three children, two of which are elements a panel consists of, and the third is a way content is shown. Another child of the top node is called 'Narrative Text' which is in fact always part of some panel, and thus better placed under 'Panel', using a consists-of relationship. More relationships are unclear and/or flawed, but I will not further discuss these.

I propose an ontology, based on Alves', but with some improvements (see Figure 32). The most important improvement is that the hierarchy of elements is clear; each relationship is a 'is-a' relationship. Next I created a second diagram (Figure 33) where other relations between the various elements are completely clear.

So together with all the terms described in section 2.2, I propose the following ontology:

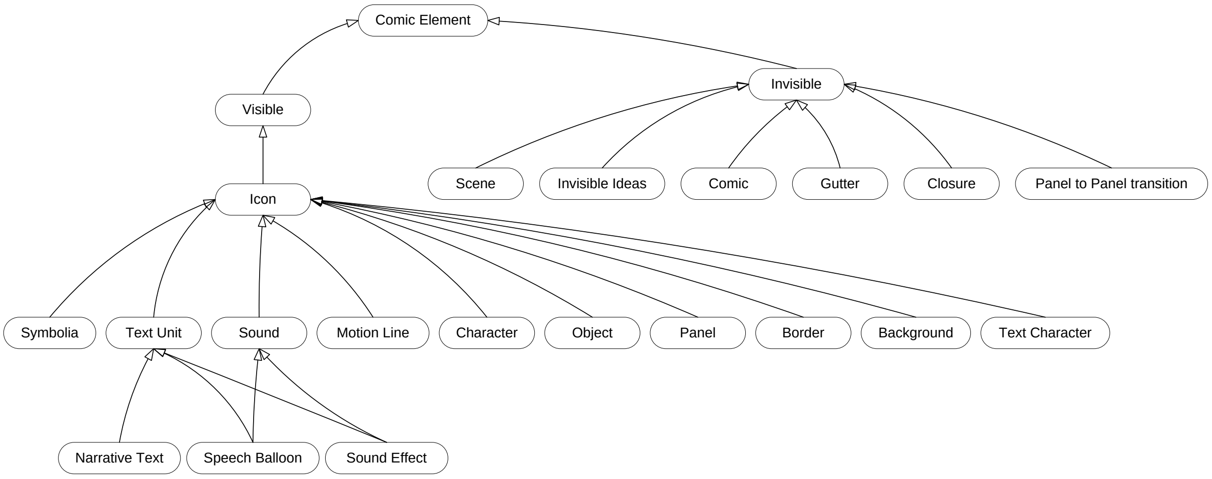


Figure 32 Comics ontology. Each relationship is a 'is-a' relationship.

Each comic consists of only Comic Elements, which are either visible or invisible. The visible elements are icons[1], which is the general super class of all graphical elements. The icon is therefore further subdivided in graphical elements (as explained in section 2.2.1). The invisible elements are those invisible things that are conveyed to the reader (see section 2.2.2 for a more in depth explanation); they cause the reader to be further involved in the process of storytelling. Other relations between these elements are shown in Figure 33.

[1] The reason for not skipping the 'Visible Element' super class altogether, is for making the distinction between visible and invisible elements apparent

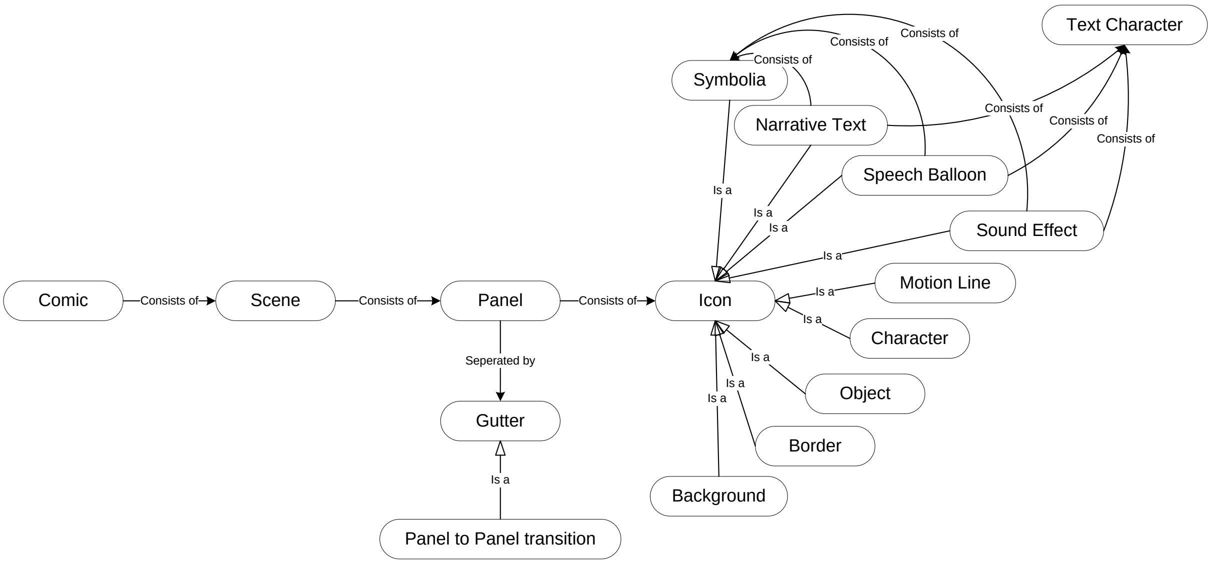


Figure 33 Relations between the various comics elements

4.4 Describing Comics

To describe a comic formally we need a certain format. In 3.2 three kinds of comics formats are described: CBML, ComicsML and CSDL. I will shortly explain why these three formats are not useful for my purpose, after which I'll propose a new format: ComicsDL.

4.4.1 Existing Comics Formats: CBML, ComicsML and CSDL

The goal of CBML is to digitalize comics, and is intended to accompany the original as its annotation. However ways of describing important information are missing: which action is performed, which emotions are displayed, which background is shown? In other words, actions, locations, poses, and the like cannot be described. To see if I could use CBML for the Comics system, I described the comic of Figure 34 in CBML (Figure 35), through which it became clear that this language is (indeed) not descriptive enough for the goal I intended to use it for. Nevertheless it can be used to describe the various elements a comic consist of in great depth: participating characters, title, publisher information, etc. Then for each panel the visible characters are listed; what they are saying and the location and size of the panel are given.

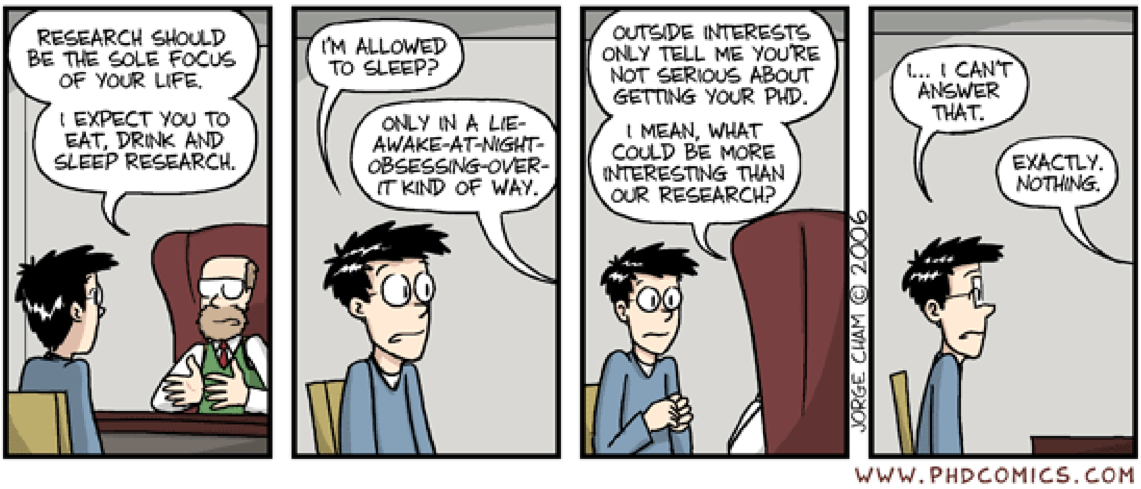

RESEARCH SHOULD BE THE SOLE FOCUS OF YOUR LIFE.
I EXPECT YOU TO EAT, DRINK AND SLEEP RESEARCH.
I'M ALLOWED TO SLEEP?
ONLY IN A LIE-AWAKE-AT-NIGHT-OBSESSING-OVER-IT KIND OF WAY.
OUTSIDE INTERESTS ONLY TELL ME YOU'RE NOT SERIOUS ABOUT GETTING YOUR PHD.
I MEAN, WHAT COULD BE MORE INTERESTING THAN OUR RESEARCH?
JORGE CHAM © 2006
I... I CAN'T ANSWER THAT.
EXACTLY. NOTHING.
WWW.PHDCOMICS.COM

Figure 34 Example comic (from www.phdcomics.com) described in CBML

```
<cbml>
        <cbmlHeader>
                <fileDesc>
                        […]
                </fileDesc>
                <profileDesc>
                    <characterList>
                      <characterItem>
                        <persona id="unknown" type="hero">
                          ???
                        </persona>
                        <persona id="prof_smith">
                          Professor Smith
                        </persona>
                      </characterItem>
                    </characterList>
                  </profileDesc>
        </cbmlHeader>
        <comic>
                <panel n="1" characters="unknown prof_smith" x="5" y="5" width="138"
                height="234">
                        <balloon type="speech" who="prof_smith">
                                <p>Research should be the sole focus of your life.</p><p>I
                                expect you to eat, drink and sleep research.</p>
                        </balloon>

                </panel>
                <panel n="2" characters="unknown" x="155" y="5" width="138" height="234">
                        <balloon type="speech" who="unknown">
                                <p>I’m allowed to sleep?</p>
                        </balloon>
                        <balloon type="speech" who="prof_smith">
                                  <p>Only in a lie-awake-at-night-obsessing-over-it kind of
                                way.</p>
                        </balloon>
                </panel>
                <panel n="3" characters="unknown prof_smith" x="305" y="5" width="138"
                height="234">
                        <balloon type="speech" who="prof_smith">
                                <p> Outside interests only tell me you’re not serious about
                                getting you PHD. </p><p> I mean, what could be more interesting
                                than our research?</p>
                        </balloon>
                </panel>
                <panel n="4" characters="unknown" x="455" y="5" width="138" height="234">
                        <balloon type="speech" who="unknown">
                                <p>I… I can’t answer that.</p>
                        </balloon>
                        <balloon type="speech" who="prof_smith">
                                  <p>Exactly. Nothing.</p>
                        </balloon>
                </panel>
        </comic>
</cbml>
```

Figure 35 *Comic of Figure 34 annotated in CBML*

In ComicsML actions can be described (as opposed to CBML), however, only in an informal manner, by describing what happens in a panel in normal text, making it less usable for my purpose. To give an example see the comic in Figure 36 and the ComicsML description in Figure 37 which were given on the website[1].

Alves’ CSDL format can formally describe the contents of the various panels in a comic, through use of describing elements, such as e.g. ‘gaze’, ‘concern’ and the combination of ‘action’ and ‘pose’ (see section 3.2). Though Alves’ CSDL is descriptive enough for my purposes, the more difference between two formats, the more can get lost in translation, so a form more closely related to fabula would help translate story content, described by some fabula, more precise. Therefore I chose to use Alves’ CSDL and fabula’s definitions of actions and events to create a new format: ComicsDL.

[1] http://www.jmac.org/projects/comics_ml/about.html

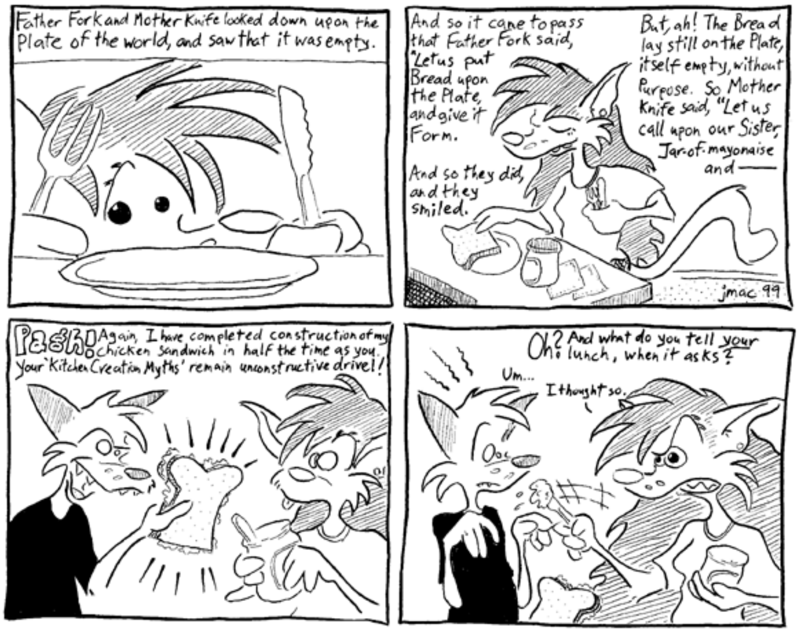

Father Fork and Mother Knife looked down upon the Plate of the world, and saw that it was empty.
And so it came to pass that Father Fork said, "Let us put Bread upon the Plate, and give it Form.
And so they did, and they smiled.
But, ah! The Bread lay still on the Plate, itself empty, without Purpose. So Mother Knife said, "Let us call upon our Sister, Jar-of-mayonaise and —
jmac 99
PAGH! Again, I have completed construction of my chicken sandwich in half the time as you. Your 'Kitchen Creation Myths' remain unconstructive drivel!
Oh? And what do you tell your lunch, when it asks?
Um...
I thought so.

Figure 36 *A comic to be described in ComicsML*

```
<strip id="disbroken-10">
    <date>Jan 1, 2000</date>
    <title>Creation Myths</title>
    <panels>
      <panel>
        <url>http://jmac.org/xml/dcs-xml/panel1.gif</url>
        <panel-desc>
          <action>Martha, holding a knife and a fork, peers at an
              empty plate.</action>

          <narration>Father Fork and Mother Knife looked down upon
              the Plate of the World, and saw that it was
              empty.</narration>
        </panel-desc>
      </panel>
      <panel>
        <url>http://jmac.org/xml/dcs-xml/panel2.gif</url>
        <panel-desc>
          <action>Martha, smiling serenely, places a slice of
              bread upon the plate.</action>

          <narration>And so it came to pass that Father Fork said,
              "Let us put Bread upon the Plate, and give it Form. And
              so they did, and they smiled.</narration>
          <narration>But, ah! The Bread lay still on the Plate,
              itself empty, without Purpose. So Mother Knife said,
              "Let us call upon our Sister, Jar-of-Mayonaise
              and--</narration>
        </panel-desc>

      </panel>
      <panel>
        <url>http://jmac.org/xml/dcs-xml/panel3.gif</url>
        <panel-desc>
          <action>Burt interrupts Martha, brandishing a sandwich
              at her. Martha looks confused.</action>
          <speech>
            <character>Burt</character>
            <text><strong>Pagh!</strong>Again, I have completed
                construction of my chicken sandwich in half the time
                as you. Your "Kitchen Creation Myths" remain
                unconstructive drivel!</text>
          </speech>
        </panel-desc>
      </panel>
      <panel>
        <url>http://jmac.org/xml/dcs-xml/panel4.gif</url>
        <panel-desc>
          <action>Martha, a wild look in her eye, suddenly
              ripostes with her mayonaise-covered butterknife,
              disarming Burt, who drops his sandwich.</action>
          <speech>
            <character>Martha</character>
            <text><strong>Oh?</strong>And what do you
                tell<emphasis>your</emphasis> lunch, when it
                asks?</text>
          </speech>
          <speech>
            <character>Burt</character>
            <text><soft>Um...</soft></text>
          </speech>
          <speech>
            <character>Martha</character>
            <text><soft>I thought so.</soft></text>
          </speech>
        </panel-desc>
      </panel>
    </panels>
  </strip>
```

Figure 37 *Comic of Figure 36 annotated in ComicsML*

4.4.2 Fabula Elements in Comics

As explained in the previous section I combine Alves' CSDL and fabula to create a formal way to describe comics. Though ComicsDL is partly based on fabula, the fabula elements cannot be copied one on one when translating fabula to ComicsDL; the various elements have to be interpreted in some way. Therefore we will look at the fabula elements and discuss how to interpret them for use in comics.

In 4.2 I explained fabula; a way to describe contents of stories formally through use of graphs with various elements that have causal relationships with each other. In fabula the following elements can occur:

- Event
- Action
- Internal Element
- Goal
- Outcome
- Perception
- Setting

Actions and events are what change the world state. An internal element is something that is part of a character. It can be a motivation for actions, a mood, or another internal state. A goal is a state to be achieved by some character, which can be done by planning actions. An outcome determines whether a goal was successful or not, and a perception helps each character to observe world changes. Finally a setting is a fact that is stated in the world, e.g. a ship that is moored in the harbor.

To translate these elements to ComicsDL I had to make some assumptions about how to interpret the various functions these elements can have in comics. Goals are abstract ideas of a state that the character wants to achieve and can almost always be translated to 'character X wanted to Y'. Because a goal is an abstract idea, but important for the reader to understand the motivations for certain actions a character performs, these will be narrated. Almost always a goal is a consequence of something that has happened (which could also be some initial setting the simulated world started with). Best would be to combine the element that led to this goal in narrating the goal, e.g. '*Because Linda was hungry, she wanted to buy ice cream*'.

Outcomes are (normally) only interesting if a goal was not successful, so when an outcome is successful, it is better left implicit, e.g. '*he found the treasure*' instead of '*he tried to find the treasure,* *and he succeeded*'. Unsuccessful outcomes can be caused by unsuccessful actions, or by missing states which are the preconditions for a goal to be fulfilled. Unsuccessful actions should be defined next to the successful actions (e.g. opening a door, and then it opens, vs. opening a door and seeing that it is locked). Missing preconditions should be narrated, as these are invisible ideas as they do not exist in that story (e.g. Linda could not buy the ice cream because she had no money). This is not yet implemented in the current state of the Comics system, and is left for future work.

In a fabula graph a perception is often followed by a belief: "*John took the apple, he saw that he took the apple, therefore he believed that he took the apple*". As one can see, these paired perceptions and beliefs are not interesting to tell. Other perceptions or beliefs could be interesting to tell, but a good algorithm would be needed to distinguish these, and a way to visualize them in a comic. Another element that falls in this category is the internal element; e.g. being in love or being hungry could be shown through Symbolia, but also be told through narrative text.

Actions and (some) events are portrayed though the panels. How a certain action or event is represented has to be defined by the author. When looking at a number of fabula stories, I noticed there are two kinds of events which I will call external events and internal events. External events are for example a volcano erupting; something that can be shown through the graphical contents of a panel. An internal event is for example someone getting hungry; something that is not (always) possible to show through the graphical contents of a panel, but can be told, for example, through narration. Therefore external events should be displayed through panels, and internal events through narrative text.

In the fabula model, an action has the following parameters:

- Agens – the agent doing the action, e.g. the person that walks
- Patiens – the object of the action, e.g. the shoes that are picked up
- Target – the target of the action, e.g. the house that the person walks to
- Instrument – anything facilitating the action, e.g. the horse that the person uses for transport
- Time – the time at which the action was selected (in number of rounds)
- Starttime – the time that the action is started (in number of rounds)
- Endtime – the time that the action is ended (in number of rounds)
- isSuccessful – whether the action succeeded

An event has the following parameters:

- Agens – the agent undergoing the event, e.g. the person that falls asleep
- Patiens – the object of the event, e.g. the rock that someone trips over
- Target – the target of the event, e.g. the bottom of a pit that someone falls into
- Instrument – anything facilitating the event, e.g. the gun that fires unintentionally
- Time – the time at which the event was selected (in number of rounds)
- Starttime – the time that the event started (in number of rounds)
- Endtime – the time that the event ended (in number of rounds)

4.4.3 A New Comics Format: ComicsDL

Using what we discussed in the previous sections and the ontology given in section 4.3 I propose the *Comics Description Language* (ComicsDL) as shown in Figure 38. I used the same global categorization of elements as Alves (see Figure 26, p. 27) combined with the action/event definition of the fabula model. The biggest difference is that in ComicsDL action/event is the most important element in a panel, and the characters are part of the action/event, whereas in CSDL each character is the most important element, and each character is autonomously described with the action (and more) it is performing. Furthermore I chose not to add information like zoom factor and camera placement as I chose to put this responsibility at the graphical generation part of the system. The reason for this is that ComicsDL is to contain the content of the comic, but no graphical information, as this can differ at graphical generation time, depending on which library (and such) is chosen. For this same reason the location of a character on the background is left out; this way each action/event can occur on each background, without having to take into account the placement of characters and objects (later we will see that this choice creates a new problem, as interaction with the background is now impossible). The only relationship between actions/events and backgrounds is the ground line which is defined for both actions/events as for backgrounds.

```
<comic>
   <scene>
          Arguments: ID, Description
          <panel>
                 Arguments: Border, Duration, Importance
                 <background>
                        Argument: ID
                 <object>
                        Arguments: ID, Value
                 <action>
                        Argument: Value
                        <Agens>
                              Argument: Value
                        <Patiens>
                              Argument: Value
                        <Target>
                              Argument: Value
                        <Instrument>
                              Argument: Value
                 <event>
                        Argument: Value
                        <Agens>
                              Argument: Value
                        <Patiens>
                              Argument: Value
                        <Target>
                              Argument: Value
                        <Instrument>
                              Argument: Value
                 <narrative>
                        <phrase>
                              <text>
                                     Argument: Type
                 <balloon>
                        Arguments: Type, ID
                        <owner>
                        <phrase>
                              <text>
                                     Argument: Type
                        <connection>
                              Argument: balloon_id
                        <sequence number>
                 <soundEffect>
                        <owner>
                        <text>
                              Argument: Type
                        <graphic>
          <transition>
                 Argument: Value
```

Figure 38 *ComicsDL format*

The arguments of a panel determine the border type (through use of predefined styles, e.g. fat, no-border, dashed, etc.), the duration (determined by the number of rounds an action or event takes to complete) and its importance relative to the other panels (e.g. some important action that has a lot of consequences). The duration determines how long the content of a panel takes to complete, e.g. a long stare by a character can be translated by creating a borderless panel containing the staring character (see section 2.2.1). A background has an ID, identifying which background to use (beach, forest, etc). Actions and events have a descriptive value (for example kick, run, fall, trip, etc) and participants of that action (which in turn can all have internal elements). A narrative box consists of phrases of text which can have a type, which determines the graphical form of the text (including font name, bold/italic, etc.). A balloon belongs to a character, contains text and can be connected with another balloon. The sequence number is for reading order. The type of the balloon determines its graphical representation (different balloons for talking, whispering, shouting, etc). Using the ID balloons can be connected. A sound effect has an owner, from which the sound originated. It has a text or a graphic, which depicts the sound. Finally the transition element is the type of transition between panels. Because these transition types are interesting for annotating purposes, but not usable for converting a ComicsDL document to a comic, I will not use these in the current version of the system. It is added nonetheless for possible use in the future. Transitions can have multiple types as described in section 2.2.2:

- Moment-to-moment
- Action-to-action
- Subject-to-subject
- Scene-to-scene
- Aspect-to-aspect
- Non-sequitur

An example of a comic in ComicsDL is given in Figure 39 and its resulting comic is given in Figure 40 (generated by Comics). This comic has one scene which consists of two panels. Both panels have a standard border, a duration of 1 round and an importance of 5 (on the scale of 1 to 10). They both also have a background and an action. The background is defined by its ID and is determined by the location of the agens at the moment he performed the action that is also contained in that panel. The first panel also has a narration box which narrates the internal event (Billy got thirsty) and the resulting goal (he wanted to get drunk). Finally each action is defined by its value (hold and drink rum) and the various roles in the action; the agens (Billy) and patiens (rum bottle) in this example.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<comic>
  <scene>
    <panel border="std" duration="1" importance="5">
      <background id="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oBeach_1"/>
      <action value="http://www.owl-ontologies.com/FabulaKnowledge.owl#Hold">
        <agens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#billyBones"/>
        <patiens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oRumBottle_1"/>
      </action>
      <narrative>
        <phrase>
          <text type="normal">
            Suddenly Billy Bones got thirsty, so he wanted to get drunk...
          </text>
        </phrase>
      </narrative>
    </panel>
    <panel border="std" duration="1" importance="5">
      <background id="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oBeach_1"/>
      <action value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#DrinkRum">
        <agens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#billyBones"/>
        <patiens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oRumBottle_1"/>
      </action>
    </panel>
  </scene>
</comic>
```

Figure 39 *The ComicsDL description of a comic*

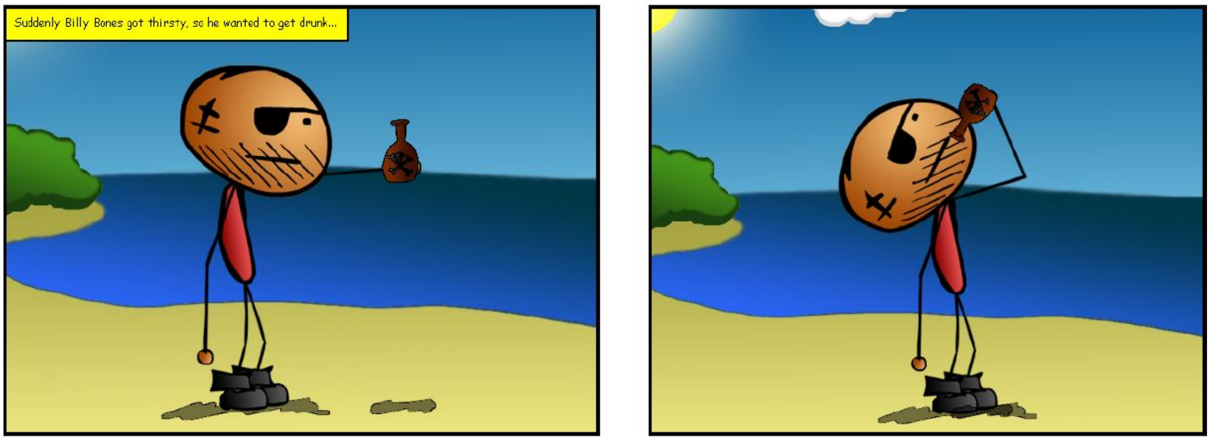
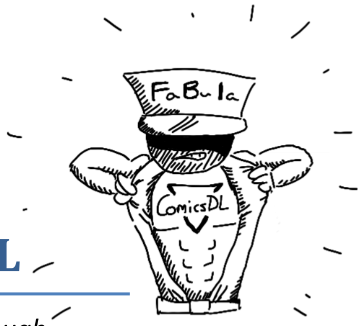

Suddenly Billy Bones got thirsty, so he wanted to get drunk...

Figure 40 *The Comic that is generated by Comics given the ComicsDL content shown above*

4.5 Summary

Based on the fabula, which is generated by the Virtual Storyteller, and the proposed ontology, I have proposed a format to represent comics: ComicsDL. This format describes the actions and events that are performed in each panel. It also describes sound effects, narrator boxes, speech balloons, extra objects and the background for each panel. Transitions between panels can also be described.

Chapter 5 Design: Fabula to ComicsDL



This chapter addresses the challenge of translating fabula to ComicsDL. Through literature and argumentation I'll suggest some solutions to certain problems and give arguments for each decision. These decisions will lead to a set of rules which will guide the translation process. After discussing the possibilities, I will explain the design of this part of the system. Research questions RQ II and RQ III will be addressed here.

5.1 Creating the Story from Fabula

To create a – hopefully – interesting story from the fabula, some assumptions about storytelling have to be made. These are partly based on findings by Trabasso and others (Trabasso et al., 1982, Trabasso and van den Broek, 1985, Trabasso and Nickels, 1992) and additional argumentation. These assumptions will not be formally tested in this study, but used directly in the system.

Trabasso et al. (1982) present the causal network model to represent stories. In their study they looked at how readers use causal reasoning to connect events and what memory representation results from this reasoning. To understand a story and to learn from a story readers must understand the individual story events and organize and store these events in a memory representation. This way they can retain that knowledge and recall it when needed. The main idea of the causal network is cause and effect; each event in a story can have causes and effects. When the comprehender is reading a story, he or she will not receive all necessary information to justify the occurrence of certain events (e.g. Jim eating an apple; motivations and other preconditions for this action like being hungry, holding the apple, having teeth, etc. are not mentioned). Therefore (uncertain) predictions are made to justify these stated events, and in case of a new event the comprehender "instantiates an expectation by a backward inference from the focal event to those events which are causally prior to it" (Trabasso et al., 1982, p. 8).

For example: "*John kissed his mother goodbye, and closed the door*". The comprehender can now make an uncertain prediction that John went out, because he kissed his mother goodbye and closed the door. Another prediction that can be made is that John loves his mother; this is a backward inference, to justify the fact that John *kissed* his mother goodbye.

Each causal relation between two events can be tested with both necessity and sufficiency (Trabasso et al., 1982, p. 7):

1. A is **sufficient** for B means that if A is put into the world and the world runs on from there, B may occur.
2. A is **necessary** for B means that if A were kept out of the world and the world were allowed to run on from there, B would not occur.

A causal network is thus a network consisting of events that are causally related. There is a distinction between 'dead-end' chains and 'causal chain events'. Dead-end chains are pathways which do not continue and do not lead to goal satisfaction (or failure). Causal chain events are those pathways which do lead to goal satisfaction or failure. The end of an episode is not when no further expectations occur, since dead-end chains lead to no further expectations. The episode ends when the desired state of change occurs or clearly fails (Trabasso et al., 1982, p. 10), in other words when goal-satisfaction occurs. Goal failure usually gives way to new consequences.

Causal cohesion is important for (immediate and delayed) recall of events by the readers. It is "quantified in terms of the percentages of events in the story which are contained in the causal chain and as the percentage of events with causal chain connections" (Trabasso et al., 1982, p. 5). In other words: fewer dead-end chains and more causality between events lead to increased cohesion. The more cohesive the elements are in a story, the more easily relationships are found between the events and a coherent representation is constructed (Trabasso et al., 1982, p. 3). This improves the reader recall of the events at a later time.

Trabasso and van den Broek (1985) analyzed, using an empirical study, how story-grammar category, causal chain and causal connectivity factors influence factors like recall, summarizing stories and judging importance of individual events (Trabasso and van den Broek, 1985, p. 615). There are multiple story grammar categories that categorize and event, like e.g. setting, initiating event, internal state, attempts, etc. It seems that readers recall settings, initiating events, goals and consequences better than other categories (Trabasso and van den Broek, 1985, p. 613). The causal chain starts with those events that introduce the protagonist(s), sets the time and place, and initiates the story with some action. The ending of a story is related to the protagonist's goal(s); if the goal has successfully been achieved, the causal chain ends with statements that indicate goal attainment. If it has failed, the chain ends with the direct consequences of the failure. The events contained in this chain from beginning to end form the causal chain. And finally causal connectivity is the number of direct causal dependencies of each event. Trabasso and van den Broek (1985) showed that events in a causal chain that have a lot of direct causal dependencies are better recalled (immediate and delayed).

Trabasso and Nickels (1992) looked at the narrative skill development from childhood to adulthood. Four groups of children (3 to 4, 4 to 5, 5 to 6 and 9 to 10 years in age) and adults (20 years and older) were asked to read a booklet of pictures and tell the story. The causal networks composed from these results were compared to the causal networks composed from the original story and several results were found which give information about the narrative skill development. Though the exact findings are beyond the scope of this thesis, summarized Trabasso and Nickels found that the results showed that "children from 3 years onward progressively move from identification and description of states to actions and then to explanation of actions carried out according to a goal plan of action" (Trabasso and Nickels, 1992, p. 273). The interesting part of this research for me is the empirical research in which persons are asked to narrate a picture story. Because a picture story can be interpreted as a comic (see comics definition), the form of this study can be used for my purposes as well. Chapter 7 goes further into necessary details.

In the following sections I will state a number of assumptions based on the above research and logical reasoning which are needed to translate fabula to ComicsDL.

5.1.1 Dead Ends

Dead Ends are those pathways in a causal network that do not continue and which do not lead to goal satisfaction or failure (Trabasso et al., 1982, p. 9), i.e. those pathways which are not in the causal chain. Dead ends are easily forgotten (Trabasso et al., 1982, Trabasso and van den Broek, 1985, Trabasso and Nickels, 1992), in contrast to the events in the causal chain. Therefore I state the following rule:

I. *Dead ends are more easily forgotten, so to keep the story simple, these will not be narrated.*

5.1.2 One Unified Story to Tell

The Virtual Storyteller's fabula model is based on the causal chain network proposed by Trabasso et al. (1982). When, in a causal network, event A is **sufficient** for event B, it means that if A happens, B can also happen. When A is **necessary** for B, it means that if A does not happen, B will also not happen. These two conditions can give information about the relationship between two causally related events and their importance in the story. Given this and the general idea about causal networks, we can see that if elements are not causally related, they do not ('narratively') belong together. This means that when two separate fabula graphs exist, they are two separate stories. In other words: one story is one causally related network, and so I state:

II. One connected graph in the fabula is one story.

This also agrees with the fabula layer, plot layer and presentation layer distinction made by Swartjes & Theune (2006); from the fabula one consistent and coherent plot is selected.

5.1.3 Which Story to Tell

As multiple graphs can occur in a certain fabula file, the question arises which graph to choose (see assumption II). At first sight, there are three options:

- Tell all available stories
- Pick one (or multiple) story randomly
- Pick one (or multiple) story based on reasons

The last option gives room for multiple implementations:

- Reasons are implicit; the user chooses a story
- Reasons are explicit; The system has an algorithm which chooses the story based on its properties
- Middle-way: the system proposes multiple stories based on some algorithm(s) and the user chooses between those.

For now I chose the second implementation, this means that the system picks one story based on certain rules. The rule I will use in the Comics system is based on the length of the path from a goal to its outcome:

III. The graph containing the longest path from a goal to its outcome, is the most useful story (for evaluation purposes) to tell.

Though that story could be overly boring to the reader, it is interesting because it will be the longest story and so will produce the longest comics, which is interesting for evaluation.

Trabasso and van den Broek (1985) state that the ending of a story is determined by what happens to the protagonist's goal(s). If the goal is attained, the causal chain ends with statements that indicate that the goal has successfully been achieved. If the attempts fail, the chain ends with the direct consequences of the failure (Trabasso and van den Broek, 1985, p. 617-618). Normally, the causal chain consists of the longest chain of events through the story (Trabasso and van den Broek, 1985, p. 618).

Given this and for sake of simplification, I conclude:

IV. The goal with the longest path to its outcome is the goal that is central in that story.

V. The introduction is everything that is necessary and sufficient for this goal, i.e. the preconditions that enabled this goal.

This (assumption IV) means that the system does not look for a main protagonist, but only for a simple, unified story consisting of the longest causal chain available. Any other goals that are part of the story are thus part of the introduction, or a sub goal of the main goal. The introduction (assumption V) is necessary to justify why the main goal exists, and consists of all events that are part of the causal chain of the selected story that led to the main goal.

5.1.4 Perceptions and Beliefs

When an action or an event occurs in fabula, this is perceived by the characters and therefore they believe that it happened. This system of perceiving and believing gives each character a knowledge base to act from. But these paired 'perception-belief's are not interesting to be told, as these two events of perceiving and believing are trivial and best left implicit for the reader to infer; e.g. Jack eats an apple, he sees that he's eating the apple, therefore he believes that he's eating the apple. The reader can easily infer that Jack is aware of the fact that he's eating the apple. And so I state the following assumption:

> VI. *Perceptions and Beliefs that are a direct acknowledgement of an Action or an Event are deleted from the graph.*

This means that perceptions and beliefs that are not a direct acknowledgement are not deleted. In other words, when someone is drinking (action), then when another character sees and believes he's drinking (perception and belief), and therefore believes he's drunk (another belief), the latter belief is not deleted. To better understand the difference between what is a direct acknowledgement and what is not in this context, I will give an example: when a character throws a stone through a window, this will initiate two perceptions: (1) a character seeing that the characters throws a stone through the window, and (2) a character seeing that the window is broken. The first is a direct acknowledgement, the second is not a direct acknowledgement of the action.

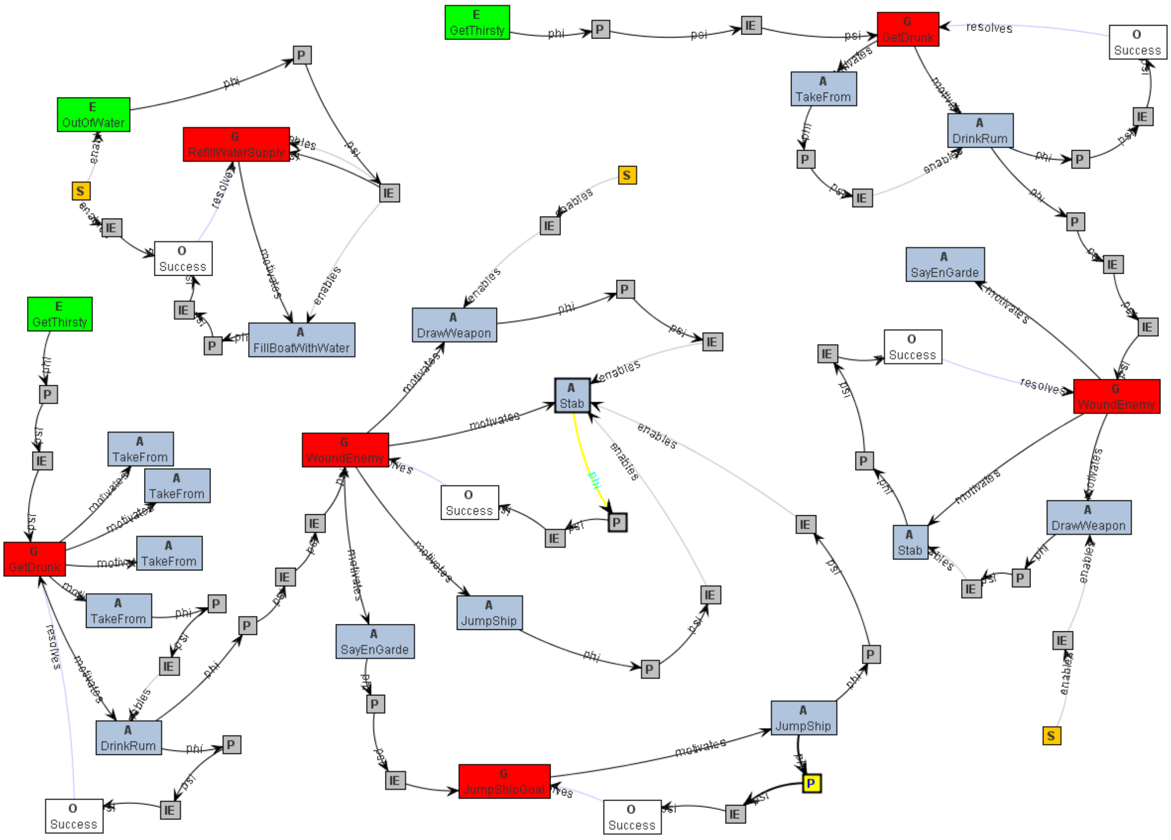


Figure 41 Top left: story nr. 3, top right: story nr. 1, and bottom left: story nr. 2.

5.1.5 Examples

To illustrate the application of these assumptions, I will show a number of stories, given the contents of a fabula file as shown in Figure 41. Below are three free translations of each graph to the English language. The last story is a free translation when all three graphs are told in one story, taking into account the chronological ordering (not visible in the figure).

Story nr. 1: Anne Bonney drinks Rum

Anne Bonney got thirsty and so she took a rum bottle from the deck and drank it. When Billy Bones saw this, he hated her, because he hated drunkards. Therefore he wanted to wound her. He said: “En garde!” and drew his weapon. Then he stabbed her.

Story nr. 2: Billy Bones drinks Rum

Billy Bones got thirsty and so he tried to get a rum bottle from the deck. This failed three times, but the fourth time he succeeded and he drank the rum. When Anne Bonney saw this, she hated him, because she hated drunkards. Therefore she wanted to wound him. She said: “En garde!”, but Billy Bones jumped off the ship into the sea. After Anne Bonney drew her weapon, she jumped into the sea too and stabbed Billy Bones.

Story nr. 3: Anne Bonney and the Water supply

Because the water supply was empty, Anne Bonney wanted to refill it. Therefore she filled the water supply with water from a pond.

All three stories

Anne Bonney and Billy Bones got thirsty. Anne Bonney took a rum bottle from the deck and drank it. After this she saw that the water supply was empty, so she refilled it with water from a pond. In the mean time Billy Bones saw that Anne Bonney was drinking rum and, because he hated drunkards, he hated her. That’s why he wanted to wound her. He said: “En garde!”, drew his weapon and stabbed Anne Bonney. When Billy Bones took a rum bottle after three failed attempts, he drank it. Anne Bonney saw this and so she hated him, because she hated drunkards. So she wanted to wound him. She said: “En garde!”, but Billy Bones jumped off the deck into the sea. After Anne Bonney drew her weapon, she jumped into the sea too and stabbed Billy Bones.

Conclusion

In this case each separate connected graph tells a unified story, which all read well. The combined story, on the other hand, does not read well (ignoring the inconsistencies of people drinking rum, while they hate drunkards). The story with longest goal to outcome (WoundEnemy in story nr. 2) is also the longest story.

5.2 Implementation

In this section I will first give a global overview of the system. After this I will discuss how to interpret fabula and what 3rd party software I use. Then I will discuss the various parts of the system that are used in translating fabula to ComicsDL.

5.2.1 Global Overview

The fabula is input to the system and before a graphical comic is created, a representation of it in a XML based format is produced. This XML based format is designed especially for this system and is called: ComicsDL (Comics Description Language, see section 4.4.3).

The process of translating the fabula to ComicsML is a three step process (see Figure 42). First it passes through a filter, which selects what part of the fabula to tell (see section5.2.3). The resulting graph is then passed on to a combiner that combines multiple fabula elements into groups, representing panels. Finally these groups (panels) are converted to ComicsDL.

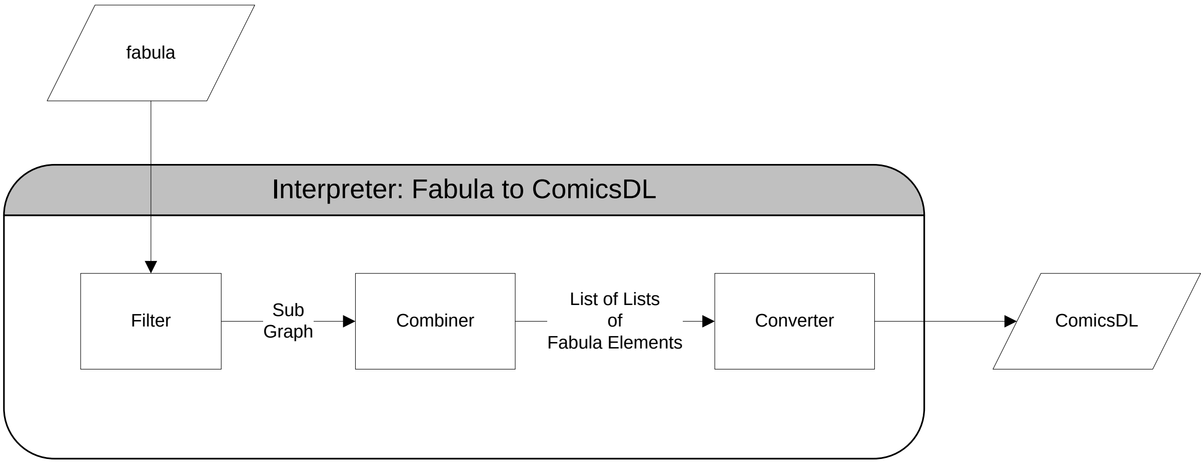
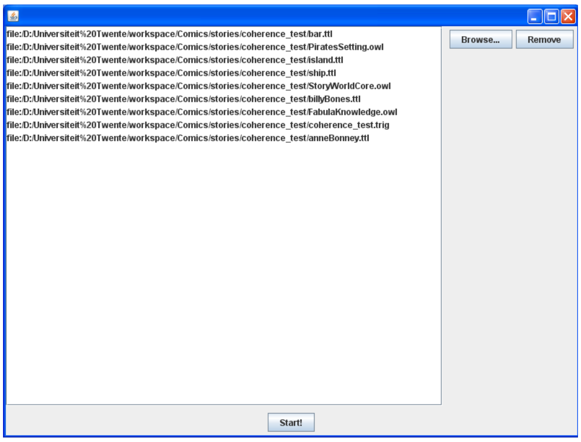

fabula
Interpreter: Fabula to ComicsDL
Filter
Sub Graph
Combiner
List of Lists of Fabula Elements
Converter
ComicsDL

Figure 42 *The three step process of translating fabula to ComicsDL*

F2CInterface (Figure 43) is a graphical interface to make it easier for the author to use the system. When the user has chosen the necessary files and presses start, the files are interpreted and passed on as a single knowledge base (containing the fabula and other necessary information) to the Filter. The Filter filters the fabula and throws away uninteresting elements based on the rules described in section 5.1. Its output is a graph which is a simpler representation of the original fabula. This graph is the input for the combiner which creates a list of lists by combining fabula elements based on rules I will describe below. Each list is in fact an early representation of a comic panel. These lists are then converted to ComicsDL panels by the Converter, which ultimately result in a ComicsDL document.

Filter, Combiner and Converter are Java interfaces, which can be implemented by anyone familiar with Java. Adding or changing functionality is therefore an easier job, and as a result increases extendibility (see evaluation criterion EC I).



```
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/bar.ttl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/PiratesSetting.owl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/island.ttl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/ship.ttl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/StoryWorldCore.owl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/billyBones.ttl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/FabulaKnowledge.owl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/coherence_test.trig
file:/D:/Universiteit%20Twente/workspace/Comics/stories/coherence_test/anneBonney.ttl
```

Figure 43 *F2CInterface with some example files filled in*

5.2.2 Jena and Named Graphs

Fabula is expressed in RDFS and OWL and thus I chose to use Jena; a third party framework which can interpret fabula. Jena[1] is a Java framework for building Semantic Web applications. It can be used to work with, and reason about RDF, RDFS, OWL and SPARQL.

RDF (Resource Description Framework) is a standard for describing relationships between things, where the relationship and the two things themselves can be named. RDFS (RDF Schema) is a semantic extension of RDF. OWL (Web Ontology Language) is a language which makes use of the

[1] See http://jena.sourceforge.net/

power of XML, RDF and RDFS and is even more powerful in expressing meaning and semantics. SPARQL (SPARQL Protocol and RDF Query Language) is a query language for accessing RDF.

Everything in RDF, RDFS and OWL is expressed in triples of the form `Subject Predicate Object`, e.g. `linda has key`. `Subject` and `Object` are both things that are linked through the relationship `Predicate`. This linking structure forms a directed, labeled graph and that is how fabula is represented.

Jena provides a way to read, write, interpret and infer about information in these various expression languages. This way the system is able to interpret the fabula and reason about causal relationships between the fabula elements.

5.2.3 Filter

The Filter is responsible for selecting the part of the fabula that will be used as contents of the resulting comic. In section 5.1 I have stated several rules and assumptions which I will restate here:

I. *Dead ends are more easily forgotten, so to keep the story simple, these will not be narrated.*
II. *One connected graph in the fabula is one story.*
III. *The graph containing the longest path from a goal to its outcome, is the most useful story (for evaluation purposes) to tell.*
IV. *The goal with the longest path to its outcome is the goal that is central in that story*
V. *The introduction is everything that is necessary and sufficient for this goal*
VI. *Perceptions and Beliefs that are a direct acknowledgement of an Action or an Event are deleted from the graph.*

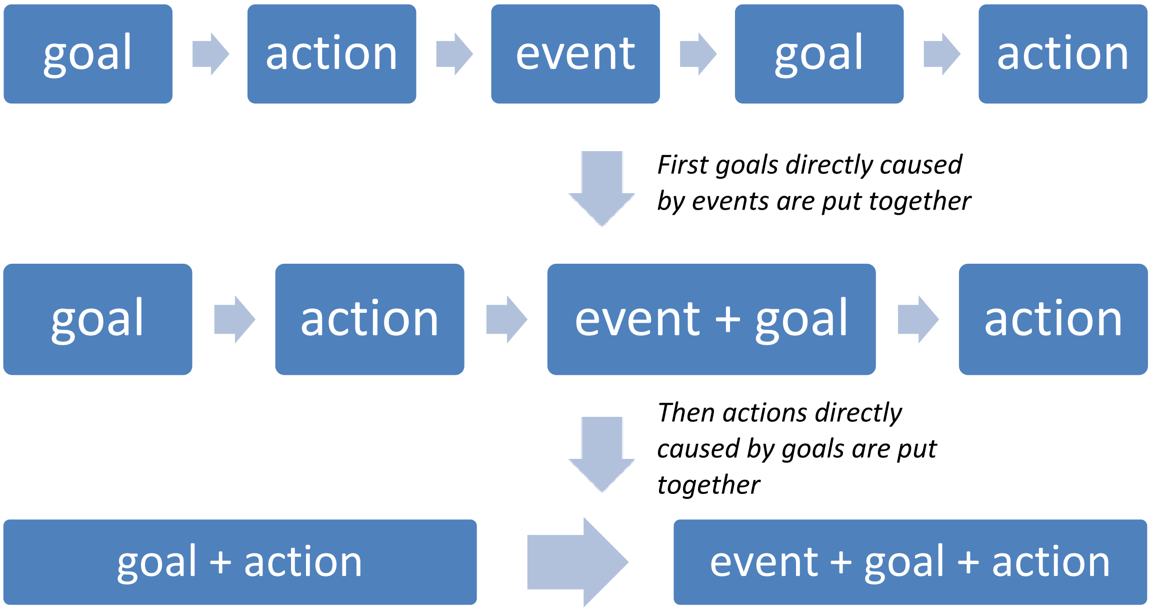
In my implementation of Filter, these rules and assumptions are realized.

First the longest goal (assumptions III, IV) is determined by calculating the longest path for each goal to its outcome. Then the intro (assumption V) is determined by taking all fabula elements that led to the initiating of the longest goal. After this all unnecessary fabula elements are filtered by deleting the perceptions and beliefs that are a direct acknowledgement of an action or event (assumption VI). Because, once the longest goal is determined, the graph is traversed using its relationships, the end result is a connected graph which represents a single story in the fabula (assumption II).

I haven't implemented the filtering of dead ends (I), as the fabula graphs that were at my disposal didn't have these. But this could be done by only adding the fabula elements that ultimately lead to the outcome of the main goal.

5.2.4 Combiner

The Combiner is responsible for combining various fabula elements into groups called panels. These panels are the panels that together form the resulting comic. In my implementation of the combiner, only settings, goals, actions and events are combined into panels. Because only actions are displayed through the contents of the panel, each panel should contain an action. Even though I made a distinction between internal and external events (see section 4.4.2), I chose to only narrate the events, and thus handling all events as if they were internal events. This ultimately leads to the result that all actions are combined with all settings, goals and events that led to that action since the last action. This means that subsequent actions are not combined with other elements, if they are not preceded by one. To give an example:


goal
action
event
goal
action
First goals directly caused by events are put together
goal
action
event + goal
action
Then actions directly caused by goals are put together
goal + action
event + goal + action

Secondly it determines and sets the locations of each character for each panel, because the locations have to be combined with the contents of the panels. This is done by looking at the beliefs of the characters where they are and where they aren't anymore. When a character goes from some place to another, let's say from the living room to the kitchen, it creates two beliefs:

> *I am not in the living room, I am in the kitchen*

These two beliefs are created at the same time. When we scan the fabula for all these beliefs, we can determine where all the characters have been and are at each moment.

5.2.5 Converter

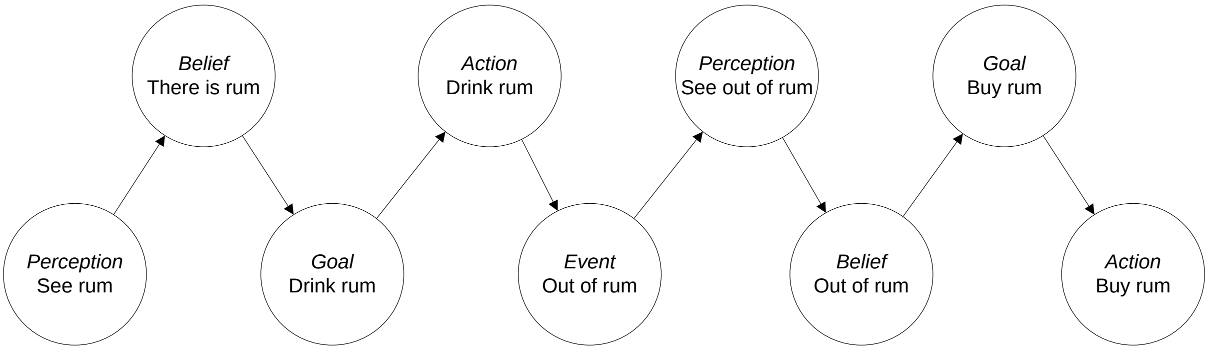
The final step in the process of translating the fabula to ComicsDL is performed by a Converter. A Converter takes a list of panels, after which the result is saved to a file or returned as an `org.w3c.dom.Document`, which is part of the java implementation of the Document Object Model (DOM) Level 3 Core[1].

It creates a ComicsDL document by iterating through the list of panels, and creating the appropriate ComicsDL description for the setting, action, event or goal it has to convert; settings, events and goals are narrated via a narrator box and actions are described through the ComicsDL `action` node. The text for the narrator box is also generated here, and is done through use of canned text. Goals are narrated as "<character> wanted to <goal name>" and events as "<character> suddenly <event name>". The system is not yet capable to narrate events without characters. Settings need to be handled a bit different, as a setting in the fabula contains a sub graph which contains a triple describing some relationship. For now only the 'has' relationship is narrated; e.g. Linda has ice cream. The sub graph is either a TruthGraph, meaning its contents are true, or a FalsehoodGraph, meaning that the negation of its contents is true. Therefore when the graph is not a TruthGraph, the phrase "doesn't" is used in the text. The resulting text is thus "It so happened that <character> has / doesn't have <instrument>". The background is determined by the location that is set for each panel by the Combiner and the duration of a panel is determined by the length of the fabula action contained in that panel.

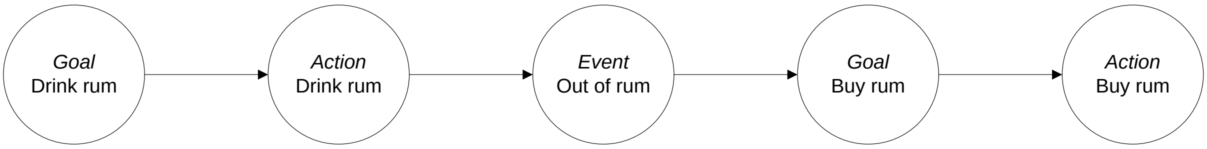
[1] See http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/

5.2.6 Example

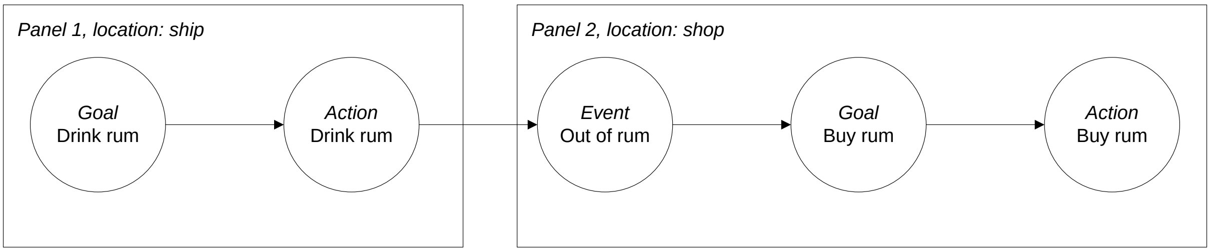
To illustrate how a fabula graph is converted to a ComicsDL document via the described pipeline of Filter, Combiner and Converter, I will give an example of a simplified fabula graph, the resulting ComicsDL document and the intermediate steps. First the fabula graph:



The main character, Billy Bones, sees rum, therefore believes there is rum, and thus takes on a goal to drink rum. He then drinks the rum, which causes him to run out of rum. He sees this, and therefore believes it and so takes on the goal of buying rum, after which he buys the rum. After passing this fabula graph through the filter, the longest goal (Drink Rum) is selected and 'perception-belief' pairs are removed from the graph. The resulting graph would be:



The perceptions and beliefs are deleted from the graph. Now the various events have to be combined into panels:



Finally this list of panels have to be converted to a ComicsDL document:

```
<panel border="std" duration="1" importance="5">
    <background id="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oShip_1"/>
    <action value="http://www.owl-ontologies.com/FabulaKnowledge.owl#DrinkRum">
        <agens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#billyBones"/>
        <patiens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oRumBottle_1"/>
    </action>
    <narrative>
        <phrase>
            <text type="normal">
                billyBones wanted to DrinkRum...
            </text>
        </phrase>
    </narrative>
</panel>
<panel border="std" duration="1" importance="5">
    <background id="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oShip_1"/>
    <action value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#BuyRum">
        <agens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#billyBones"/>
        <patiens value="http://www.owl-ontologies.com/StoryWorldSettings/Pirates#oRumBottle_1"/>
    </action>
    <narrative>
        <phrase>
```

```
            <text type="normal">
                billyBones suddenly runOutOfRum. billyBones wanted to buy rum...
            </text>
        </phrase>
    </narrative>
</panel>
```

5.3 Summary

Through literature and reasoning I showed multiple options in filtering the right story from the fabula. By giving arguments I made decisions which, briefly summarized, result in that each filtered story is one connected graph, without dead ends, containing the longest path from goal to outcome (relative to the other graphs in the fabula). With this I addressed research question RQ II: What makes an interesting story and how do you determine a plot to present from a fabula graph? And RQ III: How does the plot, which consists of fabula events, translate to comics and how do these fabula events map onto comics elements? In the second part of this chapter I discussed the implementation of this part of the system, showing the pipeline to go from fabula to ComicsDL: Filter the story, Combine the various elements, Convert the result to ComicsDL. With this design I took into account evaluation criterion EC I: **Extendibility**: How can the system be extended so that specific steps in the comic creation process can be extended or replaced, without affecting other parts of the system?

Chapter 6 Design: ComicsDL to Comic

The result of the previous process is a ComicsDL document. This document will be converted to graphical content. To be able to do this, the various actions that can occur in a story should be defined. The graphical representation of characters, objects, backgrounds, and the like has to be created based on some knowledge about the (comic) world. This is the responsibility of the author. This chapter will describe the design of this part of the system. Research questions RQ IV and RQ V will be addressed here.

6.1 Introduction

6.1.1 Three Kinds of Actions

To distinguish between an action or event in the fabula, in ComicsDL and in the resulting Comic, they will respectively be addressed as fabula action, ComicsDL action and Comics action. To avoid using the phrase “action or event” all the time, I will use the term action alone, though I also mean event in these cases. To give a better understanding of the three types of action definitions, I’ll briefly explain what they are. A ***fabula action*** defines the participants of any action, the time when it initiated, the time when it ended and whether the action was successful or not. It does not provide the semantics of what the action is, but a person reading a fabula action would understand what action it is because of the name of the action. A ***ComicsDL action*** is action described in ComicsDL format. This too does not provide any semantics of the action, but, like the fabula action, it describes the participants of the action and the link to the fabula action it originated from. The ***Comics action*** describes the positions and poses of the various participating characters and objects in the action. It also contains the name of the fabula action this Comics action belongs to.

6.1.2 Two Tools

The author has access to two tools: the ActionMaker and the ComicsDL2Comics. The ActionMaker is used to define the Comics actions that can occur in a comic, using stick figures and simple object representations. After the user has defined all the Comics actions, it can save the Comics action library to a file, which is used by the ComicsDL2Comics tool to convert a ComicsDL document to a comic. In this ComicsDL2Comics tool the user selects all the files that are needed to convert the ComicsDL document: the Comics action library file, the fabula knowledge files, of course the ComicsDL document to be converted and the PDF output file. After pressing start, the comic is converted to the PDF and saved on the file system. These two tools are explained in more detail in sections 6.3 and 6.4.

6.2 Libraries

There are three libraries that are used by the system to convert a ComicsDL document to a comic: the Action-Event library, the Graphics library and the Fabula library. The ActioneEvent library contains all Comics actions which are defined in the ActionMaker. The Graphics library contains all the graphical elements (limbs, which together form a character, objects and backgrounds) that can be found in a comic. Each element in the library is accompanied by an image name, a size factor (in relation to the characters) and an anchor point, which acts as a reference point in the image. The size factor is used to calculate what the size of the image is in relation to the characters defined in the library. For example, if the image of a rum bottle is twice as big as it should be (in pixels), then the size factor would be 0.5, making it twice as small. The anchor point is used to calculate the position of the object given the location specified in some action defined in the ActionMaker. For example, if an action ‘hold sword’ shows a character holding a sword, the sword’s handle should be placed in the hand. The sword then should be annotated with the anchor point being at the sword’s handle.

When a certain presentation cannot be found in the library, the system finds a more global representation through graceful degradation (see evaluation criterion EC II). This is performed using the Fabula library which contains all the fabula elements that can be found in a fabula file. In this part of the system the Fabula library is only used for graceful degradation; to find the type of a certain object or a superclass of a type. If for example a graphical representation for a 'rumbottle_1' object is needed, and it's not found, then the system can find a 'rumbottle' because that is its type. In case that 'rumbottle' is also unavailable, then it could be that a 'bottle' is found, because that is its superclass. The presence of the Graphics library in the system increases generativity (which addresses EC IV), as now there is virtually no limit to the number of artistic styles that can be used by the Comics system.

6.3 ActionMaker - Creating the Action

As explained in the previous section, the fabula action and event, nor the ComicsDL action and event contain any semantics about the action or event it describes. For a graphical illustration of some action or event, knowledge is required about the poses and locations of the participating characters and objects. This knowledge has to be created by the author of the stories. For each action and event that can occur in a comic, the author has to define the poses and positions of the various roles, so the comics system can use this knowledge to create the contents of the panels. Because comics artists often use, among others, stick figures to sketch the primary idea of the contents of a panel (Le Roux, F. and van Muylwijck, 1993), it seemed a good idea to use stick figures as a general representation for characters. As a general representation for objects I chose to use a simple representation that depicts the object's location and orientation (later on I'll discuss its shortcomings). This way the author can define the actions and events, and the system can apply this information to all characters and objects. By using generic action definitions to ultimately generate specific actions (in which a specific character performs a specific action, using specific objects), evaluation criterion EC IV (generativity) is addressed, as this approach increases the number of comics that can be generated.

6.3.1 Third Party Software

To define the poses and positions of the characters and objects, I looked at the following programs to see if they would be able to deliver the proper interface to define the actions and events, and output them to some format for the system to properly interpret;

- Stickman 5.2.23[1]
- Tisfat[2]
- Pivot[3]

Though these programs can be used to create animations with stick figures and thus create poses for multiple stick figures and objects, they lack the possibility to output some frame (containing the locations and poses of stick figures and objects) to a proper format which can be interpreted by the Comics system.

So I chose to implement my own tool to define the actions and events using similar stick figures and simple representations of objects, which I will discuss in the next section.

6.3.2 The ActionMaker

The ActionMaker (Figure 44) is a tool I designed to create the various poses for actions. The part to define events hasn't been implemented. The author can use this tool to define the actions that can be found in a comic. The four roles in an action can be activated, though the agens is always part of

[1] See http://www.rosoftdownload.com/download/Windows/Stickman
[2] See http://tisfat.sektorz.net/
[3] See http://www.snapfiles.com/screenshots/stickfigure.htm

any action, because any action needs a character to perform it. The agens is always a character, as only characters can perform actions. The patiens and target can be either a character or an object. The instrument, when used, is always an object, as at creation time it seemed logical that an instrument could never be a character. A character is represented by a stick figure of which all joints can be manipulated. When the user drags a joint, its limb is rotated in the desired direction. Dragging the middle joint of the stick figure positions the stick figure. An object can be positioned by dragging the middle dot, and rotated by dragging the end of the 'stick'. The line at the bottom of the screen is the ground line. At the left an action can be selected to be defined (events are also available, though the application offers no functionality to define them). These actions are read from the various fabula knowledge files; i.e. they contain any action defined in the fabula, thus any action that can occur in the resulting comic. At the right of the screen the user can choose to select the agens, patiens, target or instrument. For a character there are options to set the initial emotions and hand gestures for certain actions, but these are not yet implemented. The flip button flips the character to face the other direction.

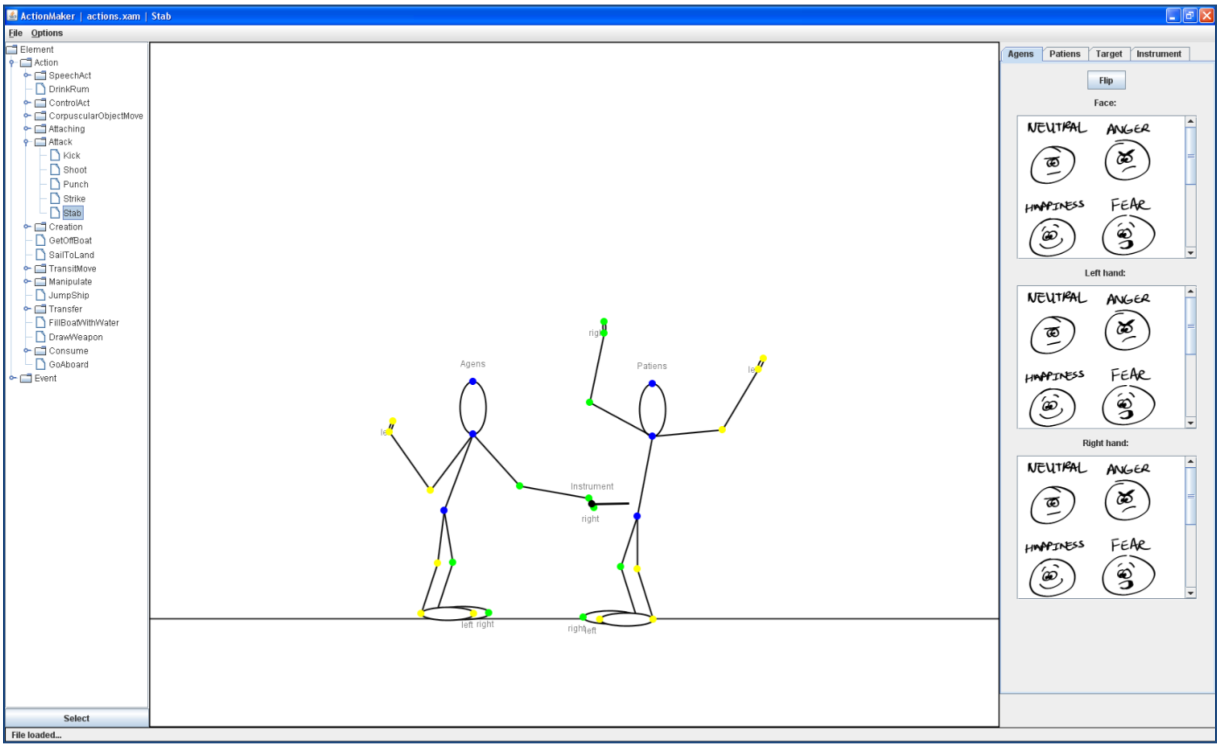


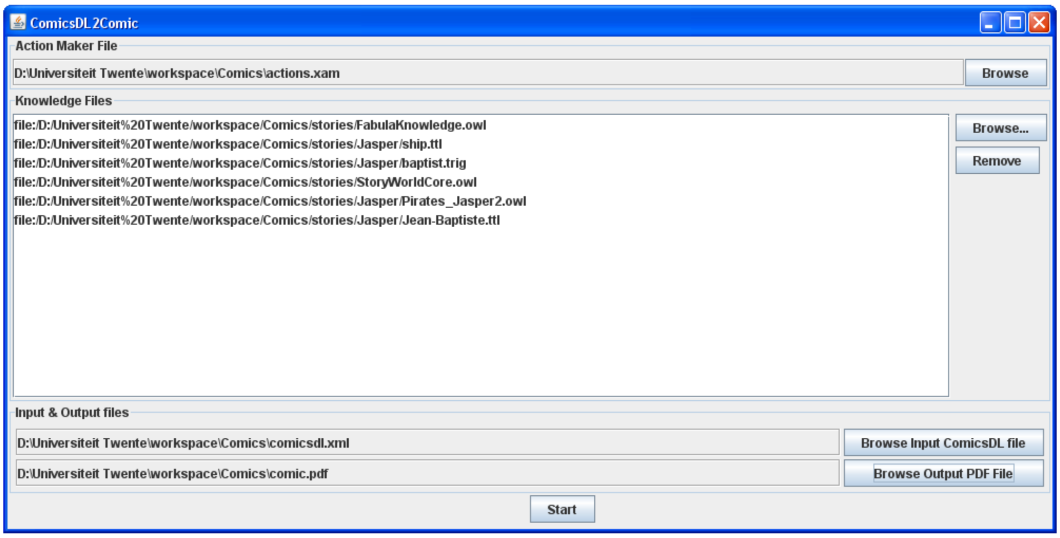
Figure 44 *An example screen of the ActionMaker*

When the user selects an action, creates the poses, he then has to 'set the action', which links the fabula action to the defined Comics action. Once the user is done creating all the actions, a file can be saved in which all actions are recorded. This file can later be opened to further edit the actions.

6.4 ComicsDL2Comic

6.4.1 Global Overview

ComicsDL2Comic (see Figure 45) is the tool that the author can use to translate the ComicsDL document to a graphical representation of that comic. The user selects the file which contains the actions defined in the ActionMaker, the fabula files which contain the story contents, the ComicsDL document to be translated and the PDF file to output the resulting comic to.



Action Maker File: D:\Universiteit Twente\workspace\Comics\actions.xam

Knowledge Files:

```
file:/D:/Universiteit%20Twente/workspace/Comics/stories/FabulaKnowledge.owl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/Jasper/ship.ttl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/Jasper/baptist.trig
file:/D:/Universiteit%20Twente/workspace/Comics/stories/StoryWorldCore.owl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/Jasper/Pirates_Jasper2.owl
file:/D:/Universiteit%20Twente/workspace/Comics/stories/Jasper/Jean-Baptiste.ttl
```

Input & Output files:

D:\Universiteit Twente\workspace\Comics\comicsdl.xml

D:\Universiteit Twente\workspace\Comics\comic.pdf

Figure 45 *The ComicsDL2Comic tool which the author can use to translate a ComicsDL document to a graphical representation of that comic*

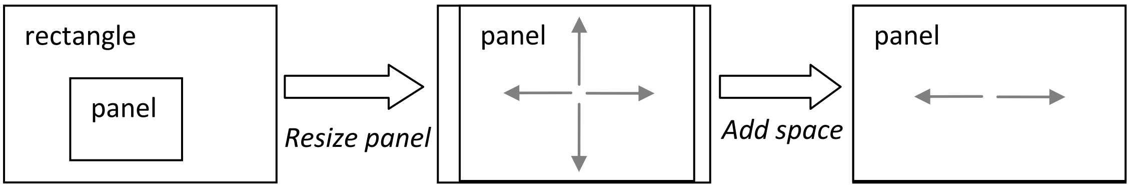
After ComicsDL2Comcis has read the input files it creates a list of ComicPanels which are Java objects containing the ComicsDL action, the background of the panel, and if present, narrator text. The ComicPanel Java objects also have a function which returns a graphical representation of the panel. The list of ComicPanels is then converted to a comic by placing the panels adjacent to each other, while outlining them to fill the entire page.

6.4.2 Laying Out the Panels

After the list of ComicPanels is created, the system creates lines of panels. Each next panel that can fit this line is added to that line, i.e. the current width of the (evenly heighted) panels in the line increased with the (minimum) width of the next panel does not exceed the width of the page, taking into account the margins and the spacing between the panels. When a line cannot be extended with more panels, each panel of that line is resized so that each line fills the complete width of the page. The resizing of the panels is done by the ComicsPanel objects themselves, which can be resized in width and in height. The widths are all evenly increased to fill the width of the page. The heights are all set on approximately ¼th of the page height, so a page can consist of 4 lines, has space between the panels and has margins at the top, bottom, left and right of each page. When all panels are added to the list of lines, the images are created, given their adjusted sizes, and 'painted' on the correct positions of the PDF document's pages.

6.4.3 Creating the Panels

As explained earlier, the ComicPanel object contains a ComicsDL action, a background and narrative text (if present). In the graphics creation process the drawing canvas is extended in size with each character and object added, so that its borders contain the graphical content without having unused space around the objects. Depending on the preferred size, extra space is added at the borders so that the panel is of the requested size. First the graphical representation of the action is created, in which all characters and objects are drawn. Then this resulting image of the action is drawn on the horizontal center of the background. After this the panels contents are scaled to fit the requested dimension, after which extra space will be added to the panel to completely fill the rectangle:



Because the backgrounds used in the system are much bigger than any panel[1], the resized panel will also be completely filled with background.

Finally the narrator box is drawn in the top left corner of the panel. The importance of each panel is also available, but is momentarily set at 5 (on a scale of 1 to 10). In a future stage this value could be used to, for example, resize the panel according to its importance; the higher its importance, the bigger the panel. This means that either the important panel should be resized to fit the whole line, or subsequent panels should be resized to the same height, or panels should be stacked, so that their combined height equals the height of the important panel. More layout variations can be added, but is left for future research.

When a panel is created, the various characters that participate in the action also need to be graphically generated. For this the actions defined in the ActionMaker and the graphical objects from the graphics library are combined. Limbs are recursively added from body to feet, hands and head using the anchor points and relative angles defined in both the action definition, as in the various limbs the character consists of. First the most 'behind' limbs are drawn, up to the most 'front' limbs, so that overlapping of the limbs is logical. The most 'behind' and most 'front' limbs of course depend on the orientation of the character. If he's oriented to the right, his left arm and leg are the most 'behind' and his right arm and leg the most 'front', and vice versa. With each limb added the drawing canvas is resized to be able to contain the image, without containing useless space. In Figure 46 an example is given of the action defined in the ActionMaker, and a result of the action after the explained creation process.

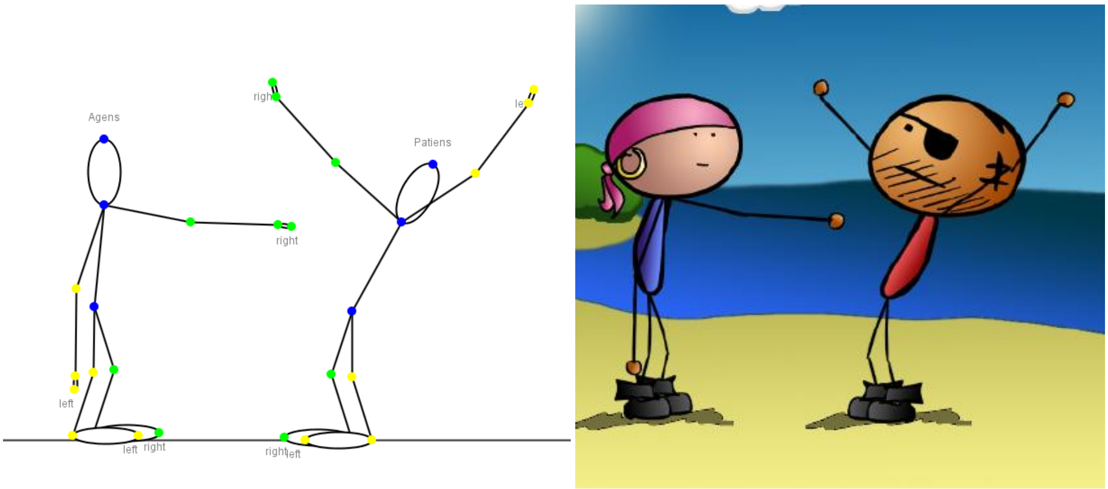


Figure 46 *Pose creation starts with the body and ends with the head, hands and feet. Left: the poses in ActionMaker, right: the poses created by the Comics system*

Before each character or object is drawn, its shadow is drawn on the ground line below the character or object. This shadow is a pre drawn image, and the shadow for characters differs from the shadow for objects, but is the same for all characters, as for all objects. To be able to correctly position the graphical representation of the action on the background, each background image is annotated with a ground line which matches the ground line defined in each action.

6.5 Design Issues

During design and after I made certain design decisions, some problems arose which I will discuss in the following sections.

[1] The chance a panel would be bigger than a background is extremely small given how the current actions are defined and the relative sizes of the characters and objects that are part of those actions.

6.5.1 Simple Character Representation

I chose a simple representation for the characters, which excluded shoulders and hips. When the first comics were generated, the resulting images were unappealing (see Figure 47), because the body was drawn ¾th (not completely from the side, nor completely from the front), but the arms and legs were positioned as if the character was seen completely from the side. I fixed this problem by creating characters with very small arms and legs (sticks), and very small shoulders and hips (see Figure 48).

A better solution would be to redefine the structure of the characters, and give them hips and shoulders. This way there is more room for the artist to create a list of characters where the user has more room to vary shoulder size, hip size, arm and leg sizes, and therefore body size and more. One important note here is that within a certain graphic library, the respective limbs of each character will be of the same size, e.g. all upper arms are of equal lengths, because when actions are defined in the ActionMaker, only the angles of the limbs are recorded. If certain limbs differ in size from the sizes used in the action maker, then the position of, for example, the hand could differ greatly from what was intended. To explain this I will give an example: a drinking action is defined in the ActionMaker with the character having an upper arm of some length, while the character performing that action has an upper arm with (much) bigger length. The result would be that the position of the hand in relation to the head would be completely different, and the action would not make sense any more. By using inverse kinematics[1] this restraint could be lifted.

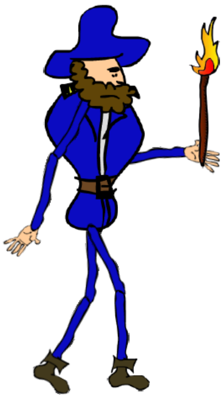


Figure 47 *First results showed that without hips and shoulders you have to be careful with how you draw the limbs*

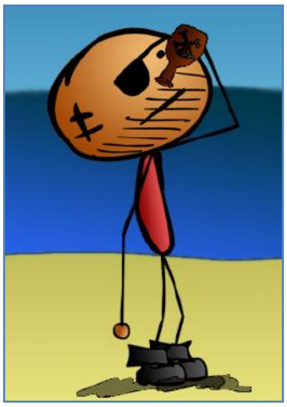


Figure 48 *The solution: small arms, legs, shoulders and hips*

6.5.2 Semantic Relationships between Panels

The consequences of what happens are not used in subsequent panels. This means that if, for example, some character picks up a torch, then, if in the next panel that torch is not part of the action, the torch will not be drawn. Another example is that if a character kicks another character, and the author chose to represent this by letting the kicked character fall, this will not influence the pose of that character in the subsequent panels, i.e. whether the character is lying in the next panel or not has totally no semantic relationship with what happened in the previous panel.

In Figure 49 on page 60 Bonney is kicked in panel 2 and lying down in panel 3. This is only because I defined the TakeFrom action with the patiens lying down. This is thus in fact a hack to make the comic look better. If this action is used in another context, it would seem weird that the patiens is

[1] Inverse Kinematics is the opposite of Forward Kinematics. In Forward Kinematics the position of the hand is the result of the angles of the joints, while in reverse kinematics the angles of the joints are the result of the position of the hand.

lying down. This problem shows the trade-off between generic action definitions and the specific graphical representation of characters performing the action (see evaluation criterion EC V).

To solve this problem, it should be subdivided in smaller problems. One is the presence of objects from one panel to the next. By recording what objects are in which characters' possession, the system is able to correctly handle this information. Extra information is needed on how to handle each object; e.g. a sword could be drawn hanging on the waist, a wallet would not have to be drawn, as it could be inside some pocket and a torch would have to be held by the character. The last one brings forth new problems, as this influences what poses the character can have (only one hand is available) as well as the poses themselves (the hand must hold the torch at a safe distance). Another sub problem is the consequences of what happens to characters in the previous panels. If for example a character is kicked down, then, if possible, the character should still be lying down in the next panel. Or a panel should be inserted showing that the character gets up again. Also the positions of the various characters should be recorded. If character A is standing left in one panel, he should still be standing left in the next panel. At the moment a character's position is only determined by the positions of the various roles in the action defined in the ActionMaker.

6.5.3 Missing Action Definitions

Currently the system is not able to handle the case when a definition of an action is missing. Best would be to find a more global definition through graceful degradation in the fabula knowledge base. For example when the `OpenDoor` action definition is missing, the system can also look for definitions of the super classes of OpenDoor: `Open`, `Manipulate` and finally `Action`. If there is an action definition for `Action`, then the system will be able to display each action. Though this action would give no knowledge about what action is performed, all participating characters and objects will be shown, and through the narrator box an explanation of what is happening could be given.

6.5.4 Mismatch ComcsDL Action and Comics Action

If in the ActionMaker the Comics action is defined to have 3 roles, though in the ComicsDL document some action only contains two roles, then the resulting image will only show the action with those participating roles that are found in the ComicsDL action. For example, the '`stab`' action is defined to have an agens (the stabber) and patiens (the one stabbed) and an instrument (the knife). If the Comics system processes a ComicsDL action which does not specify the patiens, the resulting image would show the agens holding a knife without someone to stab (see evaluation criterion EC II).

6.5.5 Anchor Points and Complex Objects

Anchor points are the points in objects that determine their positions and orientations. The graphics library describes the position of the anchor point in the image, and the ActionMaker makes use of the anchor point to determine the place and orientation of an object. In the Comics system, objects have only one anchor point, but for certain objects more anchor points would have made more sense. If two characters were having a rope pulling contest, then the rope should have two anchor points, one for the first character, and one for the second. A walking stick is another example of an object that should have two anchor points; one for the character, where he/she holds the stick, and one where it touches the ground.

There is, however, a limitation to the variation of size within any class of objects. If, for example, the user defines an action called '`leanOnStick`', which, as the name suggests, is an action where a character leans on a stick, then each instance (graphical representation) of that stick should have the same distance between the anchor points. This is because during definition of the action the user assumes that the distance between the position where to hold the stick, and the position where it touches the ground is the same as it is in the ActionMaker.

Inverse kinematics could be used to overcome this limitation however. In the ActionMaker the user defines which anchor points are linked (with certain extra parameters), and during the generation of the graphical representation of the action all anchor points are repositioned, given the constraints, so that the resulting action makes sense without getting unnatural. In the case of the ‘`leanOnStick`’ action this would mean that each instance of the stick could have any size (given certain maximas, as a character leaning on a stick of 1 kilometer in length would never look natural).

In both cases (with and without the limitation) objects could have any number of anchor points, this would given the author the possibility to create complex objects: objects that do not consist of only one part, but multiple parts. For example the canon: a canon can turn around an axis of a casing that is supporting the canon.

6.5.6 Missing Visual Elements

Symbolia, sound effects, motion lines and other additional visual elements (such as fire coming out of the nozzle when a canon is fired) are important missing factors in the current state of the system. With the definition of each action or event, these elements should be added to amplify those elements of that action or event. By adding motion lines not only will the direction of movement be more clear to the reader, but even the effect of motion will be more apparent than without. Symbolia, as shown in 2.2.1 **Graphical terms**, can show the current state of a character, the presence of stench, smoke from a cigarette, etc. Sound effects add another dimension to comics, as the sudden explosion of a canon that is launched can almost be heard by the reader. Finally other additional elements, like fire coming out of the nozzle of a canon, complete the visual presentation of a certain action or event.

6.5.7 Emotions and Facial Expressions

The ActionMaker has buttons for adding emotions/facial expressions to the actions for the various characters, but this is not yet implemented. Another option is choosing hand states (closed, open, pointing, etc), but this is also not yet implemented. When this additional functionality would be added, then through the facial expressions and hand states, each action could be more carefully defined. Nevertheless the system should give precedence to emotions (and such) coming from the fabula itself.

6.5.8 Shots

In the current version of the system there is no algorithm for choosing between various shots; long shot, medium shot, close-up, etc. It always shows all graphical information it has. However, would such a feature be implemented, it would increase how appealing the resulting comics are, as with shots tension can be build, or the focus could be drawn to a specific part, or a total shot could introduce the reader to a new scenery, etc.

6.6 Summary

With these two tools, the ActionMaker and the ComicsDL2Comics tool, the author can generate a comic given a ComicsDL document. First the user creates the poses for all possible actions and events in the ActionMaker. Then through use of the ComicsDL2Comic tool the user loads all necessary files to generate the comic from the ComicsDL document. Two research questions have been addressed. Firstly research question RQ V: How do generic fabula actions and events classes translate to specific graphical comics representations in which e.g. some specific character performs some specific action? Secondly research question RQ IV: What are the various ways to create comics graphics and what is the best way for the Comics system? Finally evaluation criterion EC IV has been addressed: **Generativity**: How generative is the system? How many varying comics can the system generate?

Below an example is given of a ComicsDL document and the resulting comic (Figure 49, the URL’s are shortened for readability).

```
<comic>
    <scene>
        <panel border="std" duration="1" importance="5">
            <background id="http://www.owl-ontologies.com/Pirates#oForest_1"/>
            <action value="http://www.owl-ontologies.com /Pirates#Chase">
                <agens value="http://www.owl-ontologies.com/Pirates#billyBones"/>
                <patiens value="http://www.owl-ontologies.com/Pirates#anne_bonney"/>
            </action>
            <narrative>
                <phrase>
                    <text type="normal">
                      It so happened that anne_bonney has oRumBottle_1. billyBones wanted to
                      WantToDrink. billyBones wanted to TakeRum...
                  </text>
                </phrase>
            </narrative>
        </panel>
        <panel border="std" duration="1" importance="5">
            <background id="http://www.owl-ontologies.com/Pirates#oBeach_1"/>
            <action value="http://www.owl-ontologies.com/Fabula#Kick">
                <agens value="http://www.owl-ontologies.com/Pirates#billyBones"/>
                <patiens value="http://www.owl-ontologies.com/Pirates#anne_bonney"/>
            </action>
        </panel>
        <panel border="std" duration="1" importance="5">
            <background id="http://www.owl-ontologies.com/Pirates#oBeach_1"/>
            <action value="http://www.owl-ontologies.com/Fabula#TakeFrom">
                <agens value="http://www.owl-ontologies.com/Pirates#billyBones"/>
                <patiens value="http://www.owl-ontologies.com/Pirates#anne_bonney"/>
                <target value="http://www.owl-ontologies.com/Pirates#oRumBottle_1"/>
            </action>
        </panel>
        <panel border="std" duration="1" importance="5">
            <background id="http://www.owl-ontologies.com/Pirates#oBeach_1"/>
            <action value="http://www.owl-ontologies.com/Pirates#DrinkRum">
                <agens value="http://www.owl-ontologies.com/Pirates#billyBones"/>
                <patiens value="http://www.owl-ontologies.com/Pirates#oRumBottle_1"/>
            </action>
        </panel>
    </scene>
</comic>
```

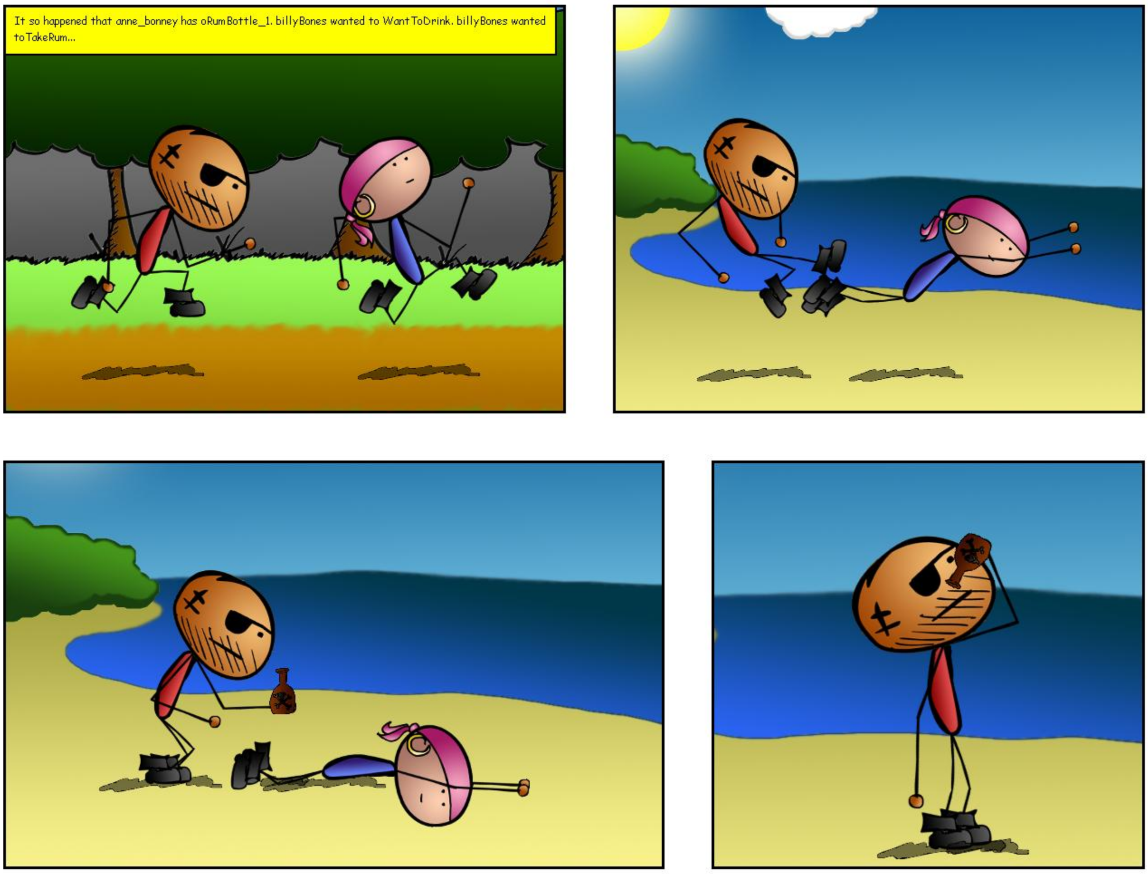


Figure 49 *The generated comic given the above ComicsDL document*

Chapter 7 Evaluation

In this chapter I will discuss the method and results of an experiment in which 10 participants read two comics, and re-tell what they read. These results are then discussed from which conclusions are drawn. Evaluation criterion EC III will be addressed here.

7.1 Research Goal

The goal of this empiric research is to test the system. Therefore I will not focus on the quality of the content of the story, as this is generated by the Virtual storyteller, but on the quality of the comic itself. I distinguish two points of focus available to research: graphic quality and content transfer quality. Because the graphic quality of the comic depends mainly on the ability of the *artist* who creates the graphics library, this will not be the point of focus in this evaluation. In *content transfer quality* I distinguish three points of focus: *content ordering* (clustering events that belong together given certain constraints, like time and/or place), *causal chain transfer* (transferring causal relations between events to the user) and *event content transfer* (transfer the content of each event to the user). Content ordering is important when stories contain parallel story lines. In general, when two sequences of events take place at the same time, the author must choose which chain of events to tell first, and if necessary, how to interleave them. Content ordering in the current state of the system is not interesting, as the current stories are all short and quite linear without parallel story lines. Also the transferring of the content of the events in a story depends mainly on the ability of the *creator* of the actions. What's left is transferring causal relations to the reader of a comic. The system's ability to transfer the causal relations to the reader implies its ability to convey the content of the story, which is ultimately its goal. I will elaborate on this; The fabula, the input of the system, is the causal network of the story. This causal network should be transferred to the mind of the reader of the comic. Such a network consists of events and their causal relations. If the reader is capable of constructing a causal network of the story after reading a comic, containing the same causal relationships as the original fabula of which the comic is created, I think we can conclude that the comic correctly conveys the content of the story to the reader.

I will conduct an experiment which resembles the experiment performed by Trabasso and Nickels (1992) in which children and adults read a picture story once, after which they were asked to tell the story. Ten (adult) participants read two comics, after which they re-tell the story by typing it into a document. The retold stories will be compared to the original, intended stories to make conclusions about the performance of the system. Furthermore, remarks made by the participants are also looked at.

I will also discuss another research method which can be used in future studies to further investigate how well the system depicts stories. It uses a psychological model called QUEST, which can determine the 'goodness' of a certain answer to a certain question, given a QUEST model of the story, which resembles a causal network.

7.2 Method

Ten participants each read two comic stories, which were generated by the system, and then retold the story they read by typing it into a document on a computer. The participants (5 males and 5 females) were all adults, ranging in age from 22 to 57. Participants were told to read one comic thoroughly (which took less than 3 minutes), after which they put away the comic and retold the story they read by typing it into a document on a computer. After the participants read the story, they were not allowed to see the comic again. After they were done with the first comic, they were asked to repeat the process for the second comic. Every participant was instructed and observed by me to make sure no extra variables were introduced into the experiment (as for example some participants reading the comic while retelling it, which has nothing to do with recall). To make sure no sequence effect would interfere with the research, half of the participants read story 1 first, and the other half read story 2 first.

Before I will discuss the results, I will state the results in the following sections. The stories were all in Dutch, so I freely translated them. At the end of each section I will summarize the comments that were given about the comic.

7.3 Comic 1: The Chase

This story was not generated by the Virtual Storyteller, but created by me and converted to a fabula graph, which then was the input to the Comics system. I could also have created this story using the Storyteller, but because my goal is to test the Comics system, so to safe time I skipped this step and created the resulting fabula file immediately. The fact that the story was created by me was not told to the participants.

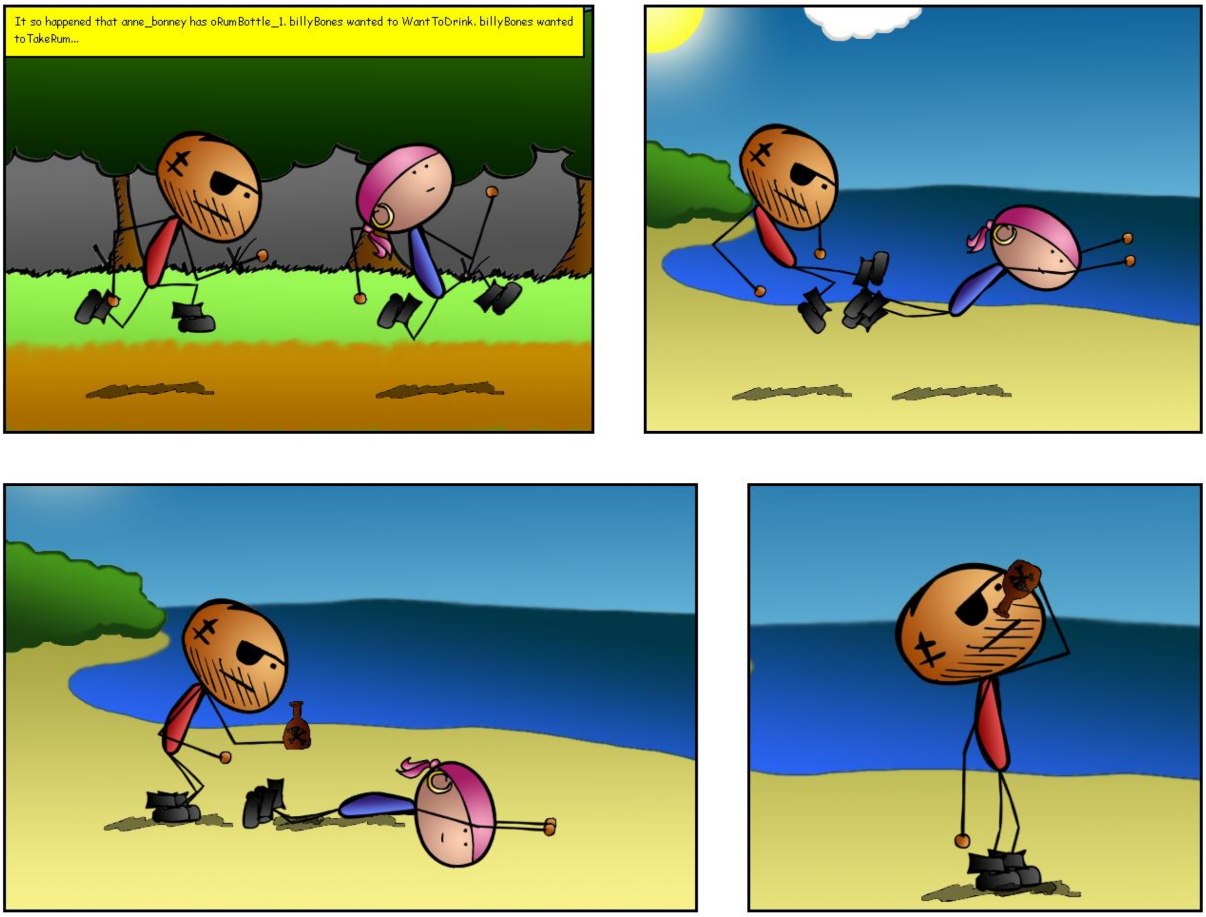

It so happened that anne_bonney has oRumBottle_1. billyBones wanted to WantToDrink. billyBones wanted toTakeRum...

Figure 50 *Comic 1: The Chase. A story thought up by myself, put into a fabula graph and translated by Comics.*

I encourage the reader to interpret the story portrayed Figure 50 and take a moment to re-tell the story as you read it. This is the story if it would be transformed to an English text:

> *Billy Bones suddenly got thirsty, so he wanted to drink rum. Anne Bonney had rum, so Billy wanted to take the rum from Anne. Therefore he chased Anne, kicked her, took the rum from her and drank it.*

The stories as interpreted by the participants are given in the next section.

7.3.1 Results

In Table 2 the stories as they were retold by the participants are given. I translated them to English (preserving original errors in syntax and grammar, e.g. Anne-Bonny in interpretation 2 and Anne baddy in interpretation 3).

Story Results	
1	Anne-bonney has a bottle of rum. Billybones wants to drink rum. Billybones chases anne-bonney. On the beach billybones tackles anne-bonney, which causes her to fall. Billybones takes the rum off of her and takes a sip.
2	Anne-Bonny chases Billy-Bones in the forest, because Anne-Bonny has a bottle of rum and Billy Bones wants to drink that rum. He tackles AB on the beach. When AB lies on the beach, he takes the bottle of rum and drinks from the bottle.
3	Anne baddy has a bottle of rum, and Billy Bones too fancies the rum! But anne bonney doesn't feel like sharing, and runs away, but got tackled on the beach by billy. Then billy bones has the rum and triumphs.
4	The one pirate wants to get hold of the rum of the other. That's why he chases him, lets him trip and takes the bottle off of him.
5	A person chases another. The first one fancies some rum. Close to arrival the foremost gets tackled. The last takes the rum and starts to drink.
6	Aunt has a bottle of rum, and Billy wants it, he chases it, then makes her trip and takes the bottle of rum and drinks it.
7	Anne _bonney runs away from billy bones. Billy bones chases her Billy tackles Anne, anne falls Billy has a bottle, of which he takes a sip
8	Anne_bones has Rum that Willy wants to have. Anne walks away and Willy tackles her. She lies on the beach, unconscious. Then he snatches her bottle of rum.
9	Anne has a bottle of rum. Billy wants to drink and would like some rum. He tackles Anne. Anne falls, afterwhich billy is drinking the rum :)
10	Jean-Baptista chases running another pirate and lets him trip on the beach after which he snatches a bottle with unidentified liquid and drinks it.

Table 2 *The interpretations of the first comic by all 10 particpants*

With interpretation number 6 the participant read aunt Bonney, instead of anne Bonney. After discussing this with the participant, it appeared this had to do with the small size of the font, which made it hard to read. And as you might notice, participant number 10 first read comic 2, and mistakenly thought that Billy Bones was Jean-Baptiste.

The following remarks were made by the participants after they read the comic:

- Participant 4: The piece of text in the first panel made no sense
- Participant 9: Nice vivid colors and Billy's appearance is appealing

The first comment may seem out of place, as the interpretations of the participants are pretty accurate, and so I think this has to do with how the text is written:

- Grammatical errors: *billyBones wanted to WantToDrink*
- Syntax errors: oRumBottle_1
- Absence of relations between the three facts: Anne Bonney has rum ***and*** Billy wanted to drink, ***so that's why*** Billy wanted to takes ***Anne's*** rum.

The reason for grammatical errors, syntax errors and absence of relations is because of how the text is generated, which is explained in 5.2.5.

7.3.2 Discussion

The first thing to notice is that how the story is retold by the 10 readers is pretty close to the original representation of the story. Some inferences are made, which is logical as is described earlier; logical inferences are made to increase the causality between the various elements in the story. I will discuss the various inferences, interpretations and missing facts in the following sections, roughly divided by the comics elements the comic consists of, which are discussed in section 2.2.

Backgrounds

While just one mentioned the forest (participant 2), 5 participants mentioned the beach. I think that the change of scenery triggered those participants to find this information important, i.e. critical to the story. Participant 5 mentioned 'close to arrival' which is an inference not further explained by the participant. Maybe this inference is made because in the background you can see the sea, and in the first panel the forest. This could imply Anne was running from the forest to something on or at the beach (e.g. a ship). The rest did not mention anything about locations.

Initial State and Goals

In the narrator box an initial state is mentioned: Anne has Rum. Participants 5, 7 and 10 are the only ones that do not mention this initial state. Next to the initial state two goals are mentioned: '`WantToDrink`' and '`TakeRum`'. Two participants (1 and 2) combine these two goals to one goal: 'drink rum'. Two other participants (3 and 5) state a goal which is compatible with both goals: 'fancy some rum'; fancy to have it or fancy to drink it. Participants 4, 6 and 8 mention that Billy wanted to have rum, which is the effect of the goal '`TakeRum`'. Participants 7 and 10 do not mention any goal and finally participant 9 mentions both goals.

Action in panel 1 - Chase

Seven participants correctly state the fact that Billy chases Anne (participant 2 apparently mixed the names of the characters). Two (3 and 7) mentioned that Anne was running away, of which one (7) also mentioned Billy chasing Anne; i.e. stating the fact and its causal effect (chasing involves running). One participant (8) mentioned Anne *walking* away, which could be a misinterpretation of the action portrayed in the panel. Maybe the lack of motion lines is the cause of this. Another possible explanation is that the participant intended running, but wrote walking. Participant 9 did not mention this action at all. This is maybe because the participant left it out because he found it trivial to mention it, or he forgot it.

Action in panel 2 - Kick

In panel 2 Billy kicks Anne, but this is perceived as Billy tackling Anne by all 10 participants. This is due to the way the action is defined, which obviously (as all participants interpreted this action the same way) resembles tackling, and not kicking.

Action in panel 3 - Take

Seven participants identify the action as Billy *taking* the rum. Two (3 and 7) only mention Billy *having* the rum, which is a state caused by Billy taking the rum, and one (9) doesn't mention this action at all. When Billy takes the rum from Anne, Anne is lying down in panel 3. This is not a literal translation of the fabula, but an interpretation by me and as such I defined the action '`TakeRum`' with the agens

(Anne) lying on the ground. Only participants 2 and 8 mention Anne lying on the beach explicitly, and one (participant 8) even makes an inference that Anne is unconscious. I think the fact that the other 8 participants did not mention this is because the fact that Anne is tackled by Billy obviously implies that this causes her to lie on the beach, which makes the fact trivial to mention.

Action in panel 4 - Drink

Seven participants mention Billy drinking the rum. Participants 3, 4 and 8 do not mention Billy drinking; it is interesting to point out that 4 and 8 mentioned the goal for Billy to have rum, but not to drink rum and 3 mentioned to fancy rum, which could imply fancy to have rum. After panel 3 this goal has been achieved, and further actions are therefore dead ends and not necessary to remember (see section 5.1.1).

Rum

Eight participants identify the bottle to contain rum, which is the correct inference as one of the goals of Billy was to 'TakeRum'. Participant 7 does not mention the contents of the bottle at all, but also did not mention any goal (which is the only indicator of what the bottle could contain). Finally participant 10 mentions the bottle to contain unidentified liquid, which could be a correct assumption, as the link between the goal and the fact that Billy takes the bottle and drinks from it do not necessarily have to be linked; i.e. Billy wanted rum, but took and drank something else. On the other hand, the participant did not mention any goal, which could mean that he did not read the narrator box. It is interesting to point out that both participants 7 and 10 did not mention any goals, and did not interpret the bottle to contain rum.

Characters

One participant (4) interpreted both characters as being male pirates, and two participants (4 and 8) did not mention their names, even though these participants (probably) did recognize Billy's goal to 'TakeRum' in the narrator box, where both names were mentioned. The fact that Anne is also perceived as being a man by this participant is due to the graphical representation of Anne, who has only a pink bandana, an earring, a lack of facial hair growth and a female name for her to be able to be identified as being a woman. Participant 5 also does not mention any name and even more has successfully omitted any word which would identify a character as being either male or female.

Though there probably are a few more things to notice, we can see by the retold stories and the discussed results that overall the comic depicts the story to the reader in a rather clear way. The inferences made by the readers are not illogical.

7.4 Comic 2: The Canon Launch

This story was generated by the Virtual Storyteller, which then was the input to the Comics system.

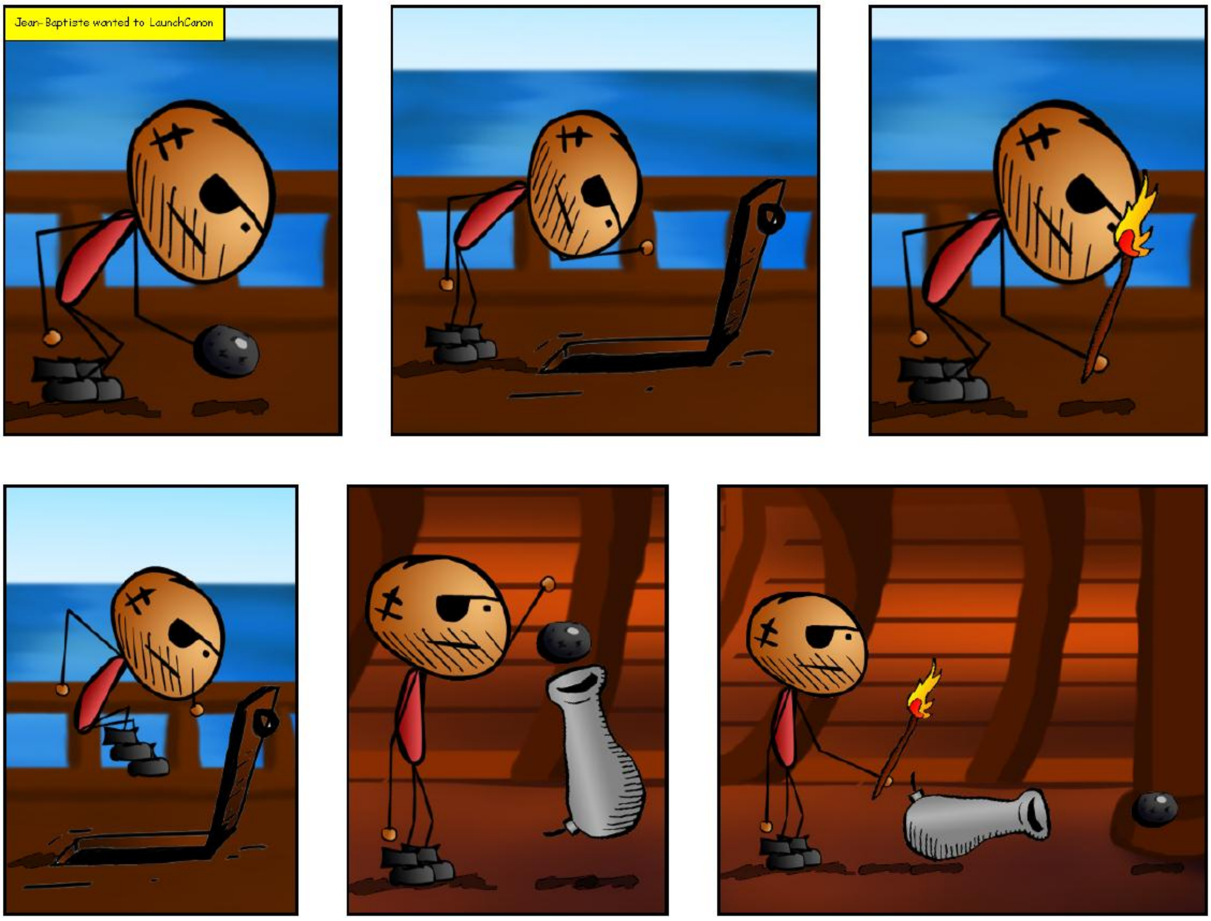


Figure 51 *Comic 2: The Canon Launch. A story generated by the Virtual Storyteller.*

As with the previous comic I encourage the reader to read the comic given in Figure 51 and re-tell it to get an idea about how you interpret it. This is the original story:

> *Jean-Baptiste wanted to launch a canon, so took a canon ball, opened a hatch, took a torch and went to the hold, through the hatch. Inside the hold he loaded the canon and fired it.*

In the next section the interpretations of the participants are given.

7.4.1 Results

In Table 3 the stories as they were retold by the participants are given. I translated them to English and tried to keep syntax errors and grammatical errors intact where I could.

	Story Results
1	*Jeab-baptiste wanted to fire a canon. He picks up a canon ball and opens the hatch to see if the canon is there and if there are matches. Downstairs there are no matches, so he takes a match. He jumps through the hatch to below deck. There he loads the canon ball into the canon. With the match in his hand he fires the canon.*
2	*Jean-Baptiste has a canon ball in his hand and wants to fire the canon. He opens a hatch in the deck and jumps through the hatch. He holds the canon ball above the canon and holds his match near the canon. The canon ball floats in front of the barrel of the canon.*

3	*JB takes a canon ball. He opens a hatch to go below deck. He takes fire to launch the canon. He jumps down the hatch to below deck, puts the canon ball into the canon, and fires the canon. Boom.*
4	*Jean-Baptiste wants to fire the canon. He takes a canon ball, throws it into the hold, takes a burning match and jumps down into the hold himself. There he puts the canon ball into the canon and fires it.*
5	*Jean Baptiste wants to load a canon, has a canon ball. Fires a match. After that the match apparently is extinguished, without the match he jumps into the hold. There is the canon the canon ball is loaded and he fires a match again.*
6	*Jean Baptiste looks into the hold, sees a canon, goes downstairs, puts the canon ball into the canon, lights it and fires.*
7	*Jean- Baptiste sees a canon ball on the ground.* *He opens a hatch and sees a canon.* *Billy takes some fire and jumps down the hatch.* *He puts the canon ball into the canon and lights it……….BOOOMMM*
8	*Jean Baptiste wants to launch a canon from a ship. He opens the hatch and lights a torch. Then he goes below deck and puts the canon ball into the canon… which he then lights (not visible).*
9	*Jean baptiste wanted to launch a canon. Opens a hatch, lights a match???(sort of), jumps in the hatch. Puts a canon ball into the canon and while he wants to fire it he sees that it already went off.*
10	*Jean-Baptista wanted to launchcanon: takes a canon ball and fire with him to below decks to the canon. Loads the canon and lights the fuse, after which he fires the canon.*

Table 3 *The interpretations of the second comic by all 10 participants*

After the participants read the second story, they were again asked for comments on the comic:

- Participant 1: The fuse is not lit in panel number 6, this leads to a different interpretation: firing of the canon failed, the canon tipped over and the canon ball fell out.
- Participant 5: The sequence of the panels is questionable. The panel in which Jean-Baptiste jumps through the hatch without a match is weird.
- Participant 6: To fire the canon inside the ship is pretty dumb
- Participant 8: You can't see that he fires the canon at the end. Nevertheless his intention to do so is clear.
- Participant 9: It seems as if the panels are in the wrong order. He opens the hatch, other panel and then he jumps in. In the panel in between the hatch is missing.

7.4.2 Discussion

With the exception of a few misinterpretations, the stories as retold by the participants are close to the original representation of the story. As in section 7.3.2 I will discuss the various inferences, interpretations and missing facts in the following sections.

Initial Goal

Six participants name the initial goal of Jean-Baptiste to fire, or launch, the canon. One participant (5) stated that the goal was to 'load' the canon, which is just a sub goal of the main goal. The other three participants did not mention any goals of Jean-Baptiste.

Sequence of Panels

Two participants comment on the sequence of panels. Participant 5 questions panels 3 to 6, and participant 9 questions panels 2, 3 and 4. They question the correctness of the sequences. In some interpretations, such a sequence leads to an (faulty) inference, e.g. that Jean-Baptiste is looking for something or sees something (participant 1, 6 and 7), or that he throws the canon ball into the hold (participant 4), or that the match is lit, extinguished and another match is lit again.

Torch

Another, rather funny, interpretation is about the torch, which is seen as a match 5 out of 10 times. Only one explicitly describes the torch as a torch. This problem arises because the torch is not drawn stereotypical enough.

Action in panel 1 – Take Canon Ball

Four participants recognize and mention this action as Jean-Baptiste taking or picking up the canon ball. Three participants do not mention this action at all, and two participants (2 and 5) say that Jean-Baptiste has a canon ball, which could indicate that these participants did not recognize this action as picking up or taking. This could be due to the missing motion lines; in one way it seems as if he's just holding the ball, while crouching a little bit. Then the last participant (7) says that Jean-Baptiste sees the canon ball on the ground. It could be that this action is unimportant enough for three participants not to mention it, and for this one participant to incorrectly recall it.

Action in panel 2 – Open Hatch

Six participants recognize and mention this action as Jean-Baptiste opening the hatch. However three participants make an extra inference to justify the action; participant 1 says that the hatch is opened to see if there are matches below deck, participant 3 says that the hatch is opened so he can go below deck (the correct inference), and participant 7 says that the hatch is opened which caused Jean-Baptiste to see the canon. Four participants do not mention this action at all. This is probably because going through the hatch implicates that the hatch had to be opened first, therefore this action can be left implicit.

Action in panel 3 – Take Torch

As said, the torch is mainly recognized as a match. Five participants correctly recognize and mention this action as Jean-Baptiste *taking* a match or fire. Three participants say that Jean-Baptiste lights or fires a match (participant 8 recognizes it as a torch) and two participants (2 and 6) do not mention this action at all.

Action in panel 4 – Go to Hold

All participants mention this action, which probably means this action is important to the story. Seven participants say that Jean-Baptiste *jumps* through the hatch, which is rather logical, given the graphical representation of the action. Two participants (6 and 8) say that Jean-Baptiste goes downstairs or below deck, which is the correct interpretation of the action. Participant 10 says that Jean-Baptiste takes a canon ball and fire with him to below deck. This means that the action is seen as taking some objects to some place, which could be a summarization of actions 1, 3 and 4, or a way to show what is the purpose of this action: to bring the objects to below deck. Participant 2 says that the match has apparently been extinguished. This inference has been made as the match is visually not present, and, for this participant, an extinguished match is the most reasonable explanation for this.

Action in panel 5 – Load Canon

Nine participants correctly recognize and mention this action as Jean-Baptiste loading the canon, or putting the canon ball into the canon. One participant (2) says that Jean-Baptiste holds the canon ball above the canon, which is a precise interpretation of the graphical representation of this action; here again, there are no motion lines, which could have indicated the ball falling into the nozzle of the canon. Nevertheless, even without motion lines, the action is correctly perceived by most participants. It also interesting that this action is mentioned by all participants, which could mean that this is an important action to the story.

Action in panel 6 – Fire Canon

Seven participants say that the canon is fired. One participant (nr. 9) says that the canon already went off. This could be because the ball has already left the nozzle, but the action of firing the canon has not been identified. This could be because the visual elements for this interpretation are missing, e.g. sizzling fuse. Another participant (nr. 2) says that the canon ball floats in front of the canon and that Jean-Baptiste is holding his match near the canon. This is probably because of visual elements like motion lines, a lit fuse and fire coming out of the nozzle of the canon are missing. Participant 5 does not even match the action of firing the canon, but does mention the action of firing the match again. This same participant says that the match is extinguished after panel 3, and so the fact that in panel 6 the match is visible again must indicate that the match has been lit again.

7.5 Conclusion

Interpretation errors fall in the following categories:

7.5.1 Missing graphical elements

At the moment the Comics system is not able to complete the graphical elements to convey a certain event to the user. Like the fuse that is not lit, no fire coming from the nozzle, the torch that is missing in panel 4 and 5 of comic 2, or motion lines and other comic elements. Expanding the system with the possibility of adding these graphical elements to events and actions would improve the interpretation of the comics.

7.5.2 Badly drawn graphical elements

The artist that creates the graphical elements in the graphics library is responsible for the fact that the graphical elements are perceived as intended. The fact that the torch is perceived as a match is thus the fault of the artist. Better drawn graphical representations would solve this problem.

7.5.3 Semantics

Some problems arise because the computer is not yet able to reason about semantics. When an action of firing the canon in a ship occurs in a story, human beings are able to interpret it as a canon, located in the hold, sticking out of the ship. The Comics system cannot and places the canon on a background of a picture depicting a hold. This is then interpreted by the readers as dumb, because firing the canon in the hold will of course destroy the ship.

A different problem with semantics is that a torch is lit, and then taken downstairs to fire the canon. But the torch is not seen until it is needed again in an action. This is an easier to fix problem than the first mentioned, as the system can track which objects are in possession of a character, and then use that information, if needed, to show these objects in other panels too.

Some sequences of actions and events are illogical and this is either because the Virtual Storyteller incorrectly generated the story, or the events were wrongly translated by Comics. Panels 2 and 3 of comic 2 are examples of this problem, as it would be far more logical to open the hatch just before you go down, and not in a seemingly random order to only satisfy all preconditions to firing the canon. It is interesting to see, though, that certain inferences are made to justify this sequence of events, as for example in interpretation nr. 1, where the reader thinks he looks if there are any matches in the hold.

These semantic problems probably need a lot more of research, so that the computer is able to reason about it. I will discuss my ideas about this in Chapter 8.

7.5.4 Wrong Inferences

Though this research is not about the way human beings reason and infer information to make things more logical, it is interesting to mention the fact that in certain places people make inferences, as discussed in section 5.1:

- The match that is extinguished (nr. 5 of comic 2), because it is not visible in the following panels.
- Jean-Baptiste that looks into the hold to see if there are matches (comic 2), because he takes the torch after opening the hatch and also because of the way the action is drawn, which is of course easy to interpret as a look-action, instead of an open-action, as it was intended.
- The canon ball that floats in front of the canon (comic 2).
- Anne Bonney being unconscious in comic 1 (interpretation nr. 8).

7.6 QUEST

For future research, another experiment could also be performed to test the performance of the Comics system, one that uses the QUEST model. Such an experiment would yield more precise results through use of ratings which give a numerical indication of how well the stories are conveyed to the readers.

Graesser, Lang and Roberts (1991) have developed a psychological model, called QUEST, which helps in determining the goodness-of-answer (GOA) of an answer to a question. The model can determine the GOA for the following question categories: why, how, enable, when and consequence. How well the causal model of a story is transferred to the reader's mind can be determined using the GOA on questions in these question categories asked to the reader. The GOA for each answer to a question is a general indicator of how good that answer is given the QUEST model. The answers of the reader are compared to these answers, which gives a score of how well the causal model has been transferred to the reader's mind. To determine the GOA of an answer, a three step plan is performed: (1) arc search procedure (associated with the question category), (2) structural distance and (3) constraint satisfaction. These steps are performed on a conceptual graph representing some information source, in this case a story. The conceptual graph is directed and can contain State, Event, Goal and Style nodes. These are then connected by Consequence (C), Implies (Im), Manner (M), Initiate (I), Reason (R) and Outcome (O) directed arcs (there are more arc categories, but these are not identified in Graesser, Lang and Roberts, 1991). The arc search procedure (1) determines which events are possible answers to a question about some event, by walking through the graph, following specific arcs determined by the question category. The structural distance (2) says that events closer to the questioned event are better answers. Constraint satisfaction (3) is used to evaluate whether the conceptual properties of the answer node are correct and compatible with the queried node.

When a QUEST model is created from a fabula model, there are two approaches: create rules to translate the events and causal relations from fabula to QUEST, or create the QUEST model by hand. In the latter case experimenter bias must be controlled for. Riedl and Young (2005) also used this model to test character believability in which they applied their enhancement to a story generation system and tested its performance by creating a control group and a test group. The control group read the story without the enhancement, while the test group with the enhancement. The answers to the questions were compared to see whether the enhancement indeed improved character believability. This same concept can be used for a future experiment where the current state of the Comics system is compared to an enhanced state of the system.

7.7 Summary

In a research in which 10 participants each read 2 comics, after which they retold the story they read, we can see that the comics were pretty well interpreted by the participants. From this we can conclude that the causal chain transfer, for those two comics, was quite successful, which is an indicator of the performance of the Comics system. Also the event content transfer must have been successful, as the readers understood the individual actions that were performed. We also see why, on certain points, the interpretations go wrong; because of the missing graphical elements, badly drawn graphics, lack of reasoning about semantics by the system and lastly the somewhat illogical story sequence generated by the Virtual Storyteller. These aspects should be addressed first when continuing research in generating Comics. Finally I gave a short explanation about the QUEST model which could be useful in future research. Evaluation criterion EC III has been addressed in this chapter; **Presentation Expressiveness**: Does the system create comics that are a good representation of the story content they need to convey to the reader? I.e. is the fabula correctly translated?

Chapter 8 Conclusions & Recommendations

At the beginning of this thesis I gave a list of research questions I set out to answer. Complete answers have not been found, and new questions were raised, but both the answers I did find and the newly raised questions will be discussed in this chapter. Secondly I will address the evaluation criteria I also stated in the first chapter. Finally I will give recommendations for future work.

8.1 Research Questions

In Chapter 1 I gave a list of research questions, which have been addressed throughout this thesis. In this section I will shortly discuss these research questions and what new questions they pose.

RQ I What are the steps that need to be taken to generate comics?

In Chapter 3 I discussed the literature in the field of comics generation and extracted three steps to comics creation:

1. Content creation
2. Translation of content to a (formal) comic description
3. Graphics creation

In the literature I discussed in Chapter 3 we saw that the described comic generation systems depend on some content that already exists; Comic Chat (Kurlander et al., 1996) depends on chat sessions, Comics2D (Alves et al., 2008) depends on content generated by FearNot! (Aylett et al., 2007), Comic Diary (Song et al., 2006) uses collected data of someone's behaviour, emotions, surroundings etc. as contents for its comics, and the system designed by Shamir et al. (2006) uses data from a computer game session log to create comics. What remains are steps 2 and 3. Though in one case these are two very distinct steps, such as with Comics2D (Alves et al., 2008), Carrie the Cartoonist (te Brinke et al., 2008), other systems do not have such a sharp distinction between these steps, such as the system presented by Shamir et al. (2006), which creates comics based on game sessions.

I think that whether there is a sharp distinction between steps 2 and 3 depends on the complexity of generating the comic given the content. E.g. in case of Comic Chat (Kurlander et al., 1996) the complexity does not lie in converting the content to a comic, but in generating the graphical components the comic consists of. Converting the content to a comic is a relatively simple process in which the chat messages are analyzed to determine emotions, gestures and who is addressed, but creation of the graphics themselves is rather complex, where the number of characters and chat messages contained by each panel is determined, how characters are positioned and where they are looking to, and how the balloons should be created and laid out so that reading order is preserved and important elements (such as characters) are not obstructed. As shown, steps 2 and 3 are distinguishable in the comic generation method of Comic Chat, but we also see that there is overlap. E.g. how many chat messages are contained by each panel could be a decision taken in step 2, but on the other hand, the amount of space each balloon takes in a panel could be the decisive factor in how many balloons fit in each panel, which still falls under step 2, but would happen at graphic creation time and thus overlap.

Like professional comics artists (almost) always first sketch, before they start on the final product, maybe a computer system too should first sketch before starting on the final product. This sketching could refer to the overlap discussed above, where a panel is graphically rendered, and depending on the intermediate results, feedback is given to recompose the panel(s), after which the panel is re-rendered, and so on. Future research should be put in this area to see if other steps can be distinguished, and how these steps can interact and run in parallel.

RQ II What makes an interesting story and how do you determine a plot to present from a fabula graph?

In Chapter 5 I stated several guidelines given literature about causal network models describing story content. These guidelines were:

I. *Dead ends are more easily forgotten, so to keep the story simple, these will not be narrated.*

II. *One connected graph in the fabula is one story.*

III. *The graph containing the longest path from a goal to its outcome, is the most useful story (for evaluation purposes) to tell.*

IV. *The goal with the longest path to its outcome is the goal that is central in that story*

V. *The introduction is everything that is necessary and sufficient for this goal*

VI. *Perceptions and Beliefs that are a direct acknowledgement of an Action or an Event are deleted from the graph.*

In Chapter 5 I showed, through use of an example, how these guidelines could be applied on a fabula file, and in section 7.4 I presented a comic of which the contents were selected using these guidelines, and showed that it presented a single, cohesive story. Both these examples already show that these guidelines are helpful in selecting a story from fabula. Nevertheless an evaluation experiment should be conducted to thoroughly test these guidelines.

RQ III How does the plot, which consists of fabula events, translate to comics and how do these fabula events map onto comics elements?

We saw that actions and external events (events that can be shown through graphics) can be mapped one-on-one to comic panels. All fabula elements, such as settings, goals, etc. can be combined with these actions and events, as an introduction to the panel, narrated through the narrator box. Future work can be spent on what other mappings are available, e.g. a fight, though one fabula event, can be mapped on e.g. 4 panels in an action comic, to entertain the reader. Maybe multiple actions and/or events can be mapped into one panel, e.g. a character opening a door and putting his coat on a rack, in one panel.

RQ IV What are the various ways to create comics graphics and what is the best way for the Comics system?

Depending on the content one approach is better than the other. In case of the Comics system, the Composition method was the only choice, as the Virtual Storyteller has no graphical interface from which screenshots can be taken. In the Comics system the central key in generating the comic are the action definitions that are created in the ActionMaker (see Chapter 6). Using stick figures and simple object representations, the author is able to create generic action definitions, which can then be used for specific actions performed by the characters in some story. Nevertheless problems arise such as missing graphical elements (also discussed under evaluation criterion EC III), e.g. fire coming out of a nozzle when a canon is fired. This problem could be addressed by giving the author options to add props, or additional graphical information when defining actions. Another problem is generic vs. specific, which I will discuss under evaluation criterion EC V.

RQ V How do generic fabula actions and events classes translate to specific graphical comics representations in which e.g. some specific character performs some specific action?

Through use of the ActionMaker (described in Chapter 6) global actions are defined, by making use of stick figures and simple object representations. These definitions are later on used to create the action poses of the various characters. This way each character can perform any part of any action. Other events such as settings, internal elements, etc. are narrated through a narrator box. In this case too, the trade-off between generic and specific poses a problem, as generic representations do not always correctly map on a specific event. This problem is discussed under evaluation criterion EC V.

8.2 Evaluation Criteria

In Chapter 1 I also gave a list of evaluation criteria which all have been addressed throughout this thesis. I will restate and discuss them here.

EC I **Extendibility**: How can the system be extended so that specific steps in the comic creation process can be extended or replaced, without affecting other parts of the system?

In Chapter 5 the translation step from fabula to ComicsDL is discussed, and there I've shown the modularity in the design of that part of the system. All three steps (filter, combiner, converter) are replaceable and/or extendable. One could even introduce additional steps in the translation process. As the action definition is the key component in generating the graphics, it means that changing the way characters or objects are represented would mean modifying the complete graphical generation part of the system, which makes it more difficult to maintain and/or to extend. Whether this could be avoided is doubtful. Nevertheless modularity has been added to this part of the system by creating a method for each comics element that is added to a panel. When additional elements, e.g. balloons, sound effects or symbolia are implemented, only the necessary methods need to be added and implemented.

EC II **Flexibility**: How flexible is the Comics system? In particular: how is missing information handled?

If the Comics system processes an action in which e.g. the patiens is not specified in the fabula graph, and thus not in the ComicsDL document, the resulting image would show the action without the patiens. To give an example: if the action is '`stab`', but the patiens is not specified in the fabula, then the resulting image would show a character stabbing in the air. If the patiens is specified in the fabula, but not in the action definition, then the resulting image would also show the action without the patiens. When a graphical representation of an object or character is not found in the library, then through graceful degradation another graphical representation is found; if e.g. '`rumbottle_1`' is not found, it looks for its type: '`rumbottle`'. If this object is also not present in the graphics library, the super class of '`rumbottle`', '`bottle`', is requested, and so on, until an object is found.

EC III **Presentation Expressiveness**: Does the system create comics that are a good representation of the story content they need to convey to the reader? I.e. is the fabula correctly translated?

In Chapter 7 I described an experiment I performed in which 10 participants read two comics generated by the Comics system. Through this experiment I showed how well the story content is conveyed to the reader through the generated comic. I also distinguished several shortcomings of the Comics system. The most important shortcomings of the Comics system are:

- *Missing graphical elements* – some actions need extra visual information to convey what is happening to the reader correctly. Important elements include symbolia, motion lines, sound effects as well as action-specific graphical elements; e.g. an action to fire the canon can be described as ***John*** uses a ***torch*** to fire a ***canon ball*** from a ***canon***. The various parts of the action are bold, and so what misses is the fuse that is lit, the fire that comes from the nozzle, i.e. everything that completes the visualization of that particular action. Without these elements, there is more room for differing interpretations.
- *Semantics* – because the Comics system is not able to reason about semantics, illogical things can occur, e.g. objects that a character picks up in the one panel, and then are not visible in the following panel. This problem about semantic reasoning can be subdivided into several subcategories, and should be addressed individually. In section 8.3.3 I will discuss two main categories I have distinguished: between panels semantics and in panel semantics.

One extra point to mention is not a shortcoming of the Comics system itself, but of the use of the system: *Badly drawn graphical elements* – how well the graphical objects are drawn is obviously very important to how the graphical objects are interpreted. For example in one comic, generated by Comics, a torch is consistently perceived as a match.

EC IV **Generativity**: How generative is the system? How many varying comics can the system generate?

In the world of comics there are, as with any other story telling medium, a multitude of genres, artistic forms, text/graphic combinations (see Chapter 2, page 14, section '***Text***'), etc. The Comics system does support multiple artistic forms through use of a graphics library. When speech balloons would be implemented, funny comics, or comics involving speech in general, can also be generated. Action comics can also be generated, though to keep them interesting, I would recommend creating a lot of different action definitions. Another factor of generativity is added by the way actions are defined. Because the author can define generic actions, each action instance can vary with each character, object and background variation.

Though there is no interaction between the background and the action/event itself, and the representation of the actions/events is always 2D and without variety in perspective (all characters and objects are placed in relation to a single baseline, this way e.g. bird or frog perspectives cannot be achieved), therefore the number of varying comic styles that can be generated by the Comics system is limited. As will be discussed in section 8.3.1 either perspective can be faked by relatively simple operations on the action definitions, else the way the graphics are generated should be revised.

Another feature that could be added in the future is the way the parts of the characters and objects are created. If, as discussed in section 3.3.2, the various parts are created on the fly, then depending on various conditions the graphical representation of each graphical element could differ. E.g. the face of a character could change depending on emotion, whether he is talking, or wearing an eye-patch and a hat. It could even contain scars from battle. These variations can apply to all parts a character consists of, and for all objects.

EC V **Generic vs. Specific Trade-off**: How does the Comics system handle the trade-off between generic and specific?

Through action definitions made in the ActionMaker, generic actions can be defined, after they can be applied in a specific manner, in which specific characters perform some action.

Though there is a trade-off, as shown in the third panel of the first generated comic (see section 7.3); Anne Bonney is lying down when Billy Bones takes the bottle off of her. The action, however, is called 'TakeRum', which would not imply that the character the rum is taken from would be lying down. The fact that Anne Bonney is lying down is in fact a hack to make this graphical representation in context of the complete comic more logical. One solution to this problem is also addressed under evaluation criterion EC III; if the system would be able to reason about semantics, then the consequences of some action in one panel could be used in the next, which would decrease this trade-off. In future work extra thought should be put in how this trade-off should be addressed.

8.3 Recommendations for Future Research

Throughout this thesis I already discussed shortcomings and gave recommendations for future research. I will summarize the future research recommendations I gave, and give some additional recommendations, which addresses some possible problems I encountered while working on this project that are interesting for future research. These recommendations can be subdivided in the following categories:

- Comics Layout Composition
- Action Definitions
- Intelligent Behaviour

These three categories of recommendations explained and discussed in the subsections below.

8.3.1 Comics Layout Composition

These recommendations address the composition of comics, i.e. the layout of panels, camera shots (e.g. long shot, medium shot, close-up, etc.) and other ways of presenting comics.

Size of Panels

In section 6.4.3 I discussed how the importance of panels can be used to determine its size. As seen in various comics, introduction panels, important panels, or panels that convey some scenery to the reader that the succeeding panels take place in, are often bigger in size than other panels. Research in the relation between the size of panels and various other properties (like importance, introduction, etc.) will help to determine relative sizes between panels.

Layout of Panels

The standard layout as often seen in comics is as how it is implemented in the Comics system. But there are more layout options (discussed in section 6.4.3), as e.g. seen in action comics, or eastern comics, where panels have diagonal intersections, or where one bigger panel is followed by two stacked panels (often accompanied by arrows to indicate reading order). The layout of panels is an interesting subject which could be added to increase generative power of the Comics system.

Shots

Shots such as long shot, medium shot and close-up are not yet possible to be applied to panels. The simplest solution is only choosing for different cuts; close-up of the face, or a medium shot including the body, or a total shot, showing e.g. the ship too, in case of a pirate comic involving a ship. When perspective comes into play, it would require taking a different graphics generation approach than the current, as all characters and objects are drawn from the side, and placed uni-dimensionally along a baseline. Maybe reasonably simple translations to the action definitions can be applied to fake different perspectives, like e.g. when a square is distorted so that the lower line and height are shortened, it creates the idea of 3D (see Figure 52).

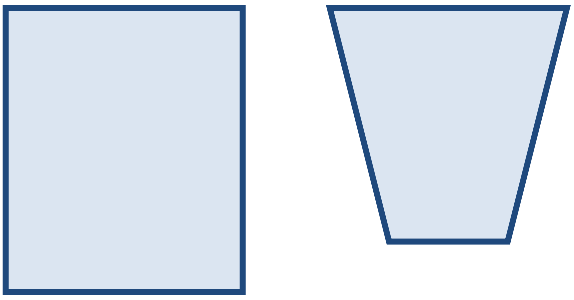


Figure 52 *Drawing the rectangle concave makes it seem as if we see the square from above*

Nevertheless reasoning about when to apply which shot is necessary to correctly handle these kinds of variations.

Parallel Storytelling

In one fabula multiple stories can take place that interact in which one plot has parallel story lines in which e.g. two characters each go their way, to later meet each other again. Scott McCloud in one of his latest books talks about bringing the comic to the computer and extending its current form. Infinite Canvas is a possibility, where the canvas is theoretically infinite and can extend to all sides. Through use of buttons or dragging, the user can choose how to continue to read the comic, and each choice leads to one of many story lines to follow. E.g. to the right character A continues his journey, and to the bottom character B. So by extending the result of the Comics system to this idea, we could show the whole story in one comic, without disturbing the main causal chain, which is important for a good understanding of the reader. But the reader can determine which causal chain he/she wants to follow. Other options of use of the Infinite Canvas should also be explored.

8.3.2 Action Definitions

As the definitions of actions play a central role in the Comics system, a lot can be improved in this area to consequently improve the overall result of the generated comics.

Additional Graphical Information

As discussed in section 8.1 as well as in section 6.5.6 there are a lot of extra additional graphical elements that could be added to the comics to make them more appealing, as well as more able to convey to the reader what is happening. Graphical information to add are things such as speech balloons, motion lines, symbolia (e.g. sweat drops or lines that depict shock or dizziness), sound effects, as well as props and visual information needed to properly display an action, like for example fire coming from a nozzle when a canon is fired.

Failing Actions

In the current state of the Comics system, actions that failed cannot be shown, nor goals that have not been achieved (see section 4.4.2). The most obvious solution for the system to be able to show failing actions, is to create two definitions for each action; one for when the action is successful and one for when the action fails. E.g. a character that is successful in opening a door, depicted by the character holding an open door (maybe accompanied by a ‘creak’ sound effect), and a character that is unsuccessful in opening a door by showing the character holding a closed door, accompanied with a ‘click’ sound, which signifies the door is shut. When a goal is not achieved, this probably means that not all preconditions for this goal are met. This could be narrated through a narrator box, and could be accompanied by some panel showing e.g. the disappointment of the character(s) involved.

Background Involvement

In the current state of the Comics system, backgrounds have no semantic relationship with the actions that are performed. They are only scenery in which the action occurs. In section 7.4, which presented the second generated comic with the canon firing in the hold, we saw that the lack of background involvement can lead to illogical graphical representations. If somehow the background can be involved into the actions and events, this problem could be solved.

Advanced Character Representations

As discussed in section 6.5.1 the characters are represented by simple stick figures that do not have distinct anchor points for shoulders and hips. I discovered that this lack greatly limited the way characters could be represented. By adding shoulders and hips to the stick figures, more variation could be brought in the appearance of characters. To go a step further, one could create a class of stick figures and even give the author the possibility of designing his own wire models so that even animals or other creatures could be represented through action definitions. This way the author can also define various classes of humanoids, and make distinctions between children, tall adults, fat adults, short adults, one armed bandits, etc.

Complex Objects

The representation for an object currently only consists of a single anchor point with orientation of the object. This limits the number of objects that can correctly be represented (see section 6.5.5), e.g. a canon that can turn around an axis of a casing that is supporting the canon could not be represented by the current object representation manner. By adding, as with advanced character representations, a way for the author to define his own object wire models, complex objects can be better represented in actions.

Number of Roles in Each Action

In the current state of ComicsDL, and thus the resulting comic, there cannot be more than 4 participating characters or objects in any action or event. This is not a real problem, as the system is intended to translate fabula to comics, and fabula shares this same shortcoming. One example of this shortcoming is that a captain cannot address a crew bigger than one person, or a priest cannot preach to a congregation which contains more than one follower. A possible solution to adding more characters to any action or event is to group them, and make them act together as one role. If, for example, there is an action where a priest speaks to his congregation, then the priest could fulfil one role, while the congregation fulfils the other. This feature should be looked at more closely so that an appropriate solution can be found.

Emotions and Other Physical Expressions

Though the option for adding facial expressions and hand gestures to actions in the ActionMaker has already been added, this feature has not been implemented. Further research should be done in how emotions and other physical expressions should be handled. These can, as said, of course be added when defining actions, but how do you choose which emotion or physical expression to choose if the fabula would also have these available, and how do you handle previous emotions (panel to panel semantics, see section 8.3.3). I would choose to give precedence to those given by the fabula, but other options should be investigated. Then there is the question what to do when emotions and/or physical expressions are missing in the action definition and fabula altogether; semantic reasoning could be applied to solve this problem, by analyzing what has happened, what is happening now, who is involved, etc.

8.3.3 Intelligent Behaviour

Intelligent behaviour in the Comics system addresses a way for the system to reason about semantics, handling missing information and using ways to increase the way generic definitions still fit when applied specifically.

Semantic Reasoning

As pointed out multiple times, if the system would be able to reason about semantics, this would greatly improve the quality of the resulting comics by making them more consistent in the information provided in and over the panels the comics consist of (see e.g. sections 6.5.2, 7.5.3 and evaluation criteria EC III and EC V). I will restate the various categories where semantic reasoning could be applied:

- *Between Panels Semantics*
    - *Object Consistency* – Objects that are introduced in one panel should also be visible in the next, or handled appropriately, e.g. to handle a wallet which is picked up from the streets, an extra panel could be introduced in which the character put the wallet into his pocket.
    - *Relative Character Locations* – Relative locations of characters (and objects) in panels should be consistent across all panels. This problem is also addressed in Comic Chat (Kurlander et al., 1996), where the authors say that maintaining the consistency of relative positions from panel to panel is the least important concern, as some comic artists do not think this is important at all.
    - *Action Consequences* – Overall consequences of actions should be visible in the next panel, as e.g. a character that has fallen should be lying down in the next panel too, or a panel should be inserted in which the character gets up again.
- *In Panel Semantics*
    - *Object Handling* – Objects that need to be visible in a panel have to be handled appropriately, e.g. a torch should be held in the hand, while a sword can be hanging from the waist and a hat can be worn on the head. This introduces a whole new set of problems, as poses need to be adjusted when e.g. an additional object needs to be held, and what to do when a hand is not free in some action?
    - *Background involvement* – certain actions require the background to be involved, as e.g. opening a door in a room, or even more complex, firing a canon from within the deck, as reasoning about the fact that a special opening for the canon is required is not trivial.
    - *Emotion Handling* – When emotions are added to the Comics system, then there should also be put effort in how to handle emotions in various cases: what gets precedence, emotions defined in the fabula, or emotions defined in the action definition? And how do emotions in one panel influence emotions in the next? What to do when emotions are missing in both the fabula and the action definition?

There are probably even more categories of semantic reasoning than mentioned here, but that is left for future researchers.

Missing Action Definition Handling

This is an easier problem to solve, which can be solved e.g. by introducing graceful degradation to select a super class of the missing action. The fact that this is not yet implemented is just a lack of time.

Inverse Kinematics

Using inverse kinematics would help solve problems in actions when characters and/or objects differ in (relative) sizes. If e.g. one character has longer arms than the other, the same action definition would not apply for both of them (see section 6.5.1). Or when a character is leaning on a stick, it would only make sense when the stick is of the same size as when the action was defined (see section 6.5.5). While in section 8.3.2 I discussed advanced character representation and complex objects to solve such a problem, with inverse kinematics this would be unnecessary, as the system itself would be able to apply one action definition to multiple specific comics actions, regardless of the size differences. Thus, with adding inverse kinematics, the trade-off between generic and specific (see EC V) would be reduced.

8.4 Conclusion

Future research and work on the Comics system should improve the quality and appealingness of the generated comics. It could even influence the way a story is told, e.g. by choosing the right shots. My advice would be to start adding additional visual elements, such as symbolia and motion lines. These would really make the generated comics come alive, as at the moment the panels are real static, as if they were photographs. Next step would be to extend the way characters and objects are defined (complex stick figures and object representations, more anchor points), and then the ability to add emotions and facial expressions to actions. After this effort semantic reasoning should be put in, so that coherence between and in the panels is increased.

Through research in the field of comics we see how comics is a unique art form with a long history and interesting characteristics. Nowadays computer science is trying to use comics for various goals to convey some content to its public. In this context the Comics system is designed to generate a comic based on the content produced by the Virtual Storyteller. The current version of the Comics system is able to generate comics that are a fairly good representation of the story content from which they were generated, and are quite nice to look at. More research and work would increase the quality of the resulting comics. Together with the Virtual Storyteller, I hope that someday the resulting comics will vary from funny comics to action comics, varying in artistic styles, conveying all kinds of stories to its readers.

Chapter 9 References

Alves, T.: *Comics2D: Automatic Generation of Comics*. MSc Thesis, Technical University of Lisbon. 2007.

Alves, T., McMichael, A., Simoes, A., Vala, M., Paiva, A., Aylett, R. *Comics2D: Describing and Creating Comics from Story-Based Applications with Autonomous Characters*. Proceedings, CASA2007. 2007.

Alves, T., Simões, A., Figueiredo, R., Vala, M., Paiva, A., Aylett, R. *So tell me what happened: turning agent-based interactive drama into comics*. Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems, Volume 3, p. 1269-1272. 2008.

Aylett, R., Vala, M., Sequeira, P., Paiva, A. *FearNot! – An Emergent Narrative Approach to Virtual Dramas for Anti-bullying Education*. Virtual Storytelling. Using Virtual Reality Technologies for Storytelling. Volume 4871/2007. p. 202-205. 2007.

Te Brinke, S., Franssen, M., van der Hoek, H., Wanders, B. *Carrie the Cartoonist*. Design Project document, University of Twente. 2008.

Cavazza, M., Charles, F., and Mead, S.J. *Generation of Humorous Situations in Cartoons through Plan-based Formalisations.* CHI-2003 Workshop: Humor Modeling in the Interface. 2003.

Comic Book Markup Language (CBML). www.cbml.org (in 01-2008)

Comics Markup Language (ComicsML). http://jmac.org/projects/comics_ml/ (in 01-2008)

Erol, K., Hendler, J., Nau, D. S. *Semantics for Hierarchical Task-Network Planning*. 1994.

Graesser, A. C. Lang, K. L. and Roberts, R. M. *Question answering in the context of stories*. Journal of Experimental Psychology: General, 120(3):254–277, 1991. Illustrating the Czar story

Hurne, I., Loker, A., Pouw, I., Vlieg, E., Volgers, R. *Annie the Animator – Animation of a Simple Plot.* Design project document, University of Twente. 2007.

Kurlander, D., Skelly, T., Salesin, D. *Comic Chat*. Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, p. 225-236. 1996.

Lekavý, M., Návrat, P. *Expressivity of STRIPS-like and HTN-like Planning.* In Proceedings of Agent and Multi-Agent Systems: Technologies and Applications, First KES International Symposium (KES-AMSTA). Pages 121–130. Wroclaw, Poland. 2007.

Liu, J., Chen, Y., Gao, W., Fu, R., Zhou, R. *Creative Cartoon Face Synthesis System for Mobile Entertainment*. Advances in Mulitmedia Information Processing - PCM 2005, p. 1027-1038. Springer, Berlin / Heidelberg. 2005.

McCloud, S. *Understanding Comics: The Invisible Art*. Kitchen Sink Press. Northampton, MA. 1993.

Riedl, M. O., Young, R. M. *An Objective Character Believability Evaluation Procedure for Multi-agent Story Generation Systems.* IVA 2005, LNAI 3661, p. 278-291. Springer-Verlag Berlin Heidelberg. 2005.

Le Roux, F., van Muylwijck, J. *Strip & Cartoontekenen.* ISBN 90-6533-305-3. 1993.

Ruttkay, Z., Noot, H. *Animated chartoon faces*. NPAR 2000: Symposium on Non-Photorealistic Animation and Rendering. p. 91–100. 2000.

Shamir, A., Rubinstein, M., Levinboim, T. *Generating Comics from 3D Interactive Computer Graphics*. IEEE Computer Graphics and Applications, Volume 26, Number 3. pp. 53-61. 2006.

Slabbers, N.: *Narration for Virtual Storytelling*. MSc Thesis, University of Twente. 2006.

Song, I.-J., Jung, M.-C., Cho, S.-B. *Automatic generation of funny cartoons diary for everyday mobile life*. Sattar, A., Kang, B.-H. (eds.) AI 2006. LNCS (LNAI), vol. 4304, p. 443–452. Springer, Heidelberg. 2006.

Swartjes, I.: *The Plot Thickens: bringing structure and meaning into automated story generation*. MSc Thesis, University of Twente. 2006.

Swartjes, I.: *Whose Story Is It Anyway? How improv informs agency and authorship of emergent narrative*. Dissertation, University of Twente. ISBN 978-90-365-3004-0. 2010.

Swartjes, I. Theune, M. *A Fabula Model for Emergent Narrative*. University of Twente. 2006.

Trabasso, T., Broek, vande n, P. *Causal Thinking and the Representation of Narrative Events*. The University of Chicago. 1985.

Trabasso, T., Nickels, M. *The Development of Goal Plans of Action in the Narration of a Picture Story.* The University of Chicago. 1992.

Trabasso, T., Secco, T., Broek, van den, P. *Causal Cohesion and Story Coherence*. The University of Chicago. 1982

Wikipedia. *The Lexicon of Comicana*. http://en.wikipedia.org/wiki/The_Lexicon_of_Comicana. May 2009.

Chapter 10 Figures

This is a list of figures that were used in this thesis that were neither taken from the cited sources nor created by me.

Figure 1
from www.bdquebec.qc.ca/historique/chap01.htm
last checked: 06-2010

Figure 3
accel6.mettre-put-idata.over-blog.com/0/52/72/82//Magritte-ceci-nest-pas-une-pipe.jpg
last checked: 06-2010

Figure 11 (middle picture)
from "*Comics & Sequential Art*" by Will Eisner
1985

Figure 12
from en.wikipedia.org/wiki/Speech_balloon
last checked: 06-2010

Figure 14
from brian-romero.blogspot.com/2007_10_01_archive.html
last checked: 06-2010