Mapping Invention in Writing:
Digital Infrastructure and the Role of the Genetic Editor

Dissertation for the degree of doctor in Literature at the University of Antwerp to be defended by

Elli BLEEKER

Promotor: Prof. Dr. Dirk Van Hulle
Antwerpen, July 5 2017
Mapping Invention in Writing:
Digital Infrastructure and the Role of the Genetic Editor

Elli Bleeker
Een Studie naar het Creatieve Schrijfproces:
Digitale Infrastructuur en de Rol van de Genetische Editeur

Elli Bleeker
A caveman painted on a cave.
   It was a bison, was a fave
The other cave people would rave.
   They didn’t ask ‘Why?’
   ‘Why paint a bison if it’s dead?’
   ‘When did you choose the colour red?’
   ‘What was the process in your head?’
He told their story.
   What came before he didn’t show.
   We’re not supposed to know.

Homer’s Odyssey was swell,
   A bunch of guys that went through hell.
He told the tale, but didn’t tell the audience why.
   He didn’t say, ‘here’s what it means,’
   ‘And here’s a few deleted scenes.’
   ‘Charybdis tested well with teens,’
He’s not the story.
   He’s just a door we open if our lives need lifting.
But now we pick, pick, pick, pick, pick it apart.
Open it up to find the tick, tick, tick of a heart.

(‘Heart Broken’ by Josh Whedon)
The research leading to this PhD dissertation has received funding from the People Programme (Marie Skłodowska-Curie) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement nº 317436.
## Introduction

### Chapter 1 - Disciplines and their Methodologies

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1. Genetic criticism</td>
<td>16</td>
</tr>
<tr>
<td>1.1.1. Objectives</td>
<td>19</td>
</tr>
<tr>
<td>1.1.2. Methodologies</td>
<td>25</td>
</tr>
<tr>
<td>1.2. Computational Methods for Humanities Scholarship</td>
<td>29</td>
</tr>
<tr>
<td>1.2.1. 'Computer-assisted' or 'Digital'? Different Ways of Editing</td>
<td>30</td>
</tr>
<tr>
<td>1.2.2. Computational Methods for Editing</td>
<td>34</td>
</tr>
<tr>
<td>1.2.3. Challenges</td>
<td>43</td>
</tr>
<tr>
<td>1.2.4. Building Bricks</td>
<td>50</td>
</tr>
<tr>
<td>1.2.5. Reflection</td>
<td>57</td>
</tr>
<tr>
<td>1.3. Digital Genetic Editing</td>
<td>59</td>
</tr>
<tr>
<td>1.3.1. Five Tasks of Digital Philology</td>
<td>60</td>
</tr>
<tr>
<td>1.3.2. Five Components of a Genetic Edition</td>
<td>62</td>
</tr>
<tr>
<td>1.3.3. Five Aspects of Textual Genesis</td>
<td>64</td>
</tr>
<tr>
<td>1.4. Concluding Remarks</td>
<td>69</td>
</tr>
</tbody>
</table>

### Chapter 2 - Transcription and Collation

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1. Preamble</td>
<td>74</td>
</tr>
<tr>
<td>2.1.1. Introduction</td>
<td>74</td>
</tr>
<tr>
<td>2.1.2. Definitions: A Closer Look</td>
<td>79</td>
</tr>
<tr>
<td>2.2. Theories and Practices</td>
<td>90</td>
</tr>
<tr>
<td>2.2.1. Automating Collation: Tools, Past and Present</td>
<td>90</td>
</tr>
<tr>
<td>2.2.2. Formalizing Collation: The Gothenburg Model</td>
<td>93</td>
</tr>
<tr>
<td>2.2.3. Plain-Text Collation</td>
<td>100</td>
</tr>
<tr>
<td>2.2.4. Transforming Markup to Plain Text</td>
<td>105</td>
</tr>
<tr>
<td>2.2.5. Pre- and postprocessing: Passing Along Markup</td>
<td>114</td>
</tr>
<tr>
<td>2.2.6. Reflection</td>
<td>128</td>
</tr>
<tr>
<td>2.3. Theories and Practices Applied</td>
<td>130</td>
</tr>
<tr>
<td>Native XML Collation</td>
<td>134</td>
</tr>
<tr>
<td>2.3.1. Native XML Collation: Theory</td>
<td>136</td>
</tr>
<tr>
<td>2.3.2. Native XML Collation: Practice</td>
<td>140</td>
</tr>
<tr>
<td>2.3.3. Reflection</td>
<td>146</td>
</tr>
<tr>
<td>2.4. Concluding Remarks</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 3 - Visualizing Textual Variance

3.1. Preamble

3.2. Theories and Practices
  3.2.1. Visualizing Transcriptions
  3.2.2. Visualization of Collation Output
    3.2.2.1. Alignment Tables
    3.2.2.2. Synoptic Viewers
    3.2.2.3. Parallel segmentation
    3.2.2.4. Variant Graph

3.3. Theories and Practices Applied:

3.3.1. B917/H2: The Archival Boxes of Sheherazade
3.3.2. Challenges to Representing Textual Genesis
3.3.3. Making the BDE: Objectives and Approach
3.3.4. Presentation of Editorial Findings
3.3.5. User Testing, or: Did You Get My Message?
3.3.6. Reflection

3.4. Concluding Remarks

Chapter 4 - Organizing Documents and Versions

4.1. Preamble

4.2. Theories and Practices
  4.2.1. Documentary and Textual Classifications
  4.2.2. A Network of Text

4.3. Theories and Practices Applied:

Two-Step Alignment

4.4. Concluding remarks

Conclusion

Acknowledgements

Abstract / Samenvatting

Bibliography
Introduction

‘Book historians and editors gaze into the past for their evidence, but they aim at informing or changing thinking in the present and the future.’
(Engert 2005, 98)

‘One great value of modern work on electronic archives lies in its process, in what the work teaches the worker, rather than in the accomplishment of the product.’
(Shillingsburg 2009b, 21)

When musician and producer Brian Eno invites bands into his studio to record an album, he sometimes makes them swap instruments. The lead singer is given a bass guitar, while the keyboard player suddenly finds himself behind the drums. Their unfamiliarity and awkwardness with the instruments does not instantly result in better music – on the contrary – but it produces a different sound and, most importantly, gets the musicians out of their comfort zone. It’s a well-known trick of improvisation. Deliberately introducing disruptive elements into the process of creation can trigger ‘the unknown, the unpredictable, the innovative and occasionally the outright brilliant’ (Tidman 2016). In similar ways, digital scholarly editing has benefited from implementing tools and instruments previously unfamiliar to the discipline. Doing manuscript research and making scholarly editions with digital technologies has compelled editors to reconsider notions and methodologies they have long been familiar with. As a consequence, ‘all the main a priori assumptions about scholarly editions come into question’ (Gabler 2010, 43).

This process of reconsideration is ongoing, for the developments in computational technology continue to influence and advance scholarly editing on both theoretical and practical levels.

The present study examines how these ever-changing conditions affect the function of scholarly editors and the framework in which they operate. I focus on editors that follow a genetic orientation to text, meaning that they investigate the development of text and study the creative processes that propagate literary invention. For centuries, people have been intrigued by how writers write and how literary works originate. This fascination took flight in the 19th century when, in line with the idea of the Romantic genius, authors and poets began to preserve their draft manuscripts and notebooks. In following decades, such drafts were collected as iconic art pieces or served as secondary material to help editors reconstruct the author’s intention and establish a
critical text. But with the advent of genetic criticism (critique génétique) as a discipline in the 1970s, draft manuscripts came to be legitimate objects of study as testimonies of the creative writing process.

Since genetic criticism is not the only editorial school that examines draft texts and literary writing processes, the genetic orientation to text serves as an inclusive container for ‘the network of editorial theories’ that have this specific focus (Dillen 2015, 15). The functionalities of digital technology provide numerous tools that support manuscript research, from high-definition digital facsimiles to automatic collation. Correspondingly, the past decades have seen an increase in digital genetic editions that provide a research environment to study draft manuscripts and the writing process. Taken together, these editions confirm the thesis that manuscript research is not solely at the service of edition-making, but that the edition can also serve as a tool to do manuscript research (Van Hulle [Variants 12, forthcoming]). In this dual nature lies the possibility of a rapprochement between scholarly editing or textual scholarship and genetic criticism, a development that Wout Dillen further explores in his PhD thesis ‘Digital Scholarly Editing for the Genetic Orientation’ (2015). In addition to supporting established editorial methodology, digital technologies can also be employed to further advance manuscript research. To name but a few examples, text can be structured as a non-hierarchical network of interconnected fragments, like it has been done in HyperNietzsche (chapter 4.2); the dynamic nature of the writing process can be both visualized and narrated, as can be seen in fluid text editions (Bryant 2002; chapter 3.2); or a collation tool can be customized to deal with deletions and additions (chapter 2.4). As such, the tools we use have the potential to become research instruments proper, expanding the boundaries of manuscript research and scholarly editing.

Scholarly editors have always operated between author, text, and reader, but as they encode text, customize software, or design a digital infrastructure – in short, as they enrich established methodology with digital technologies and computational methods – editors become more central in the edition. Indeed, Hans Walter Gabler notes, ‘not the author and the text, but the editor [is made] pivotal to an edition’ (Gabler 2010, 45). The specific manner in which digital technology is implemented becomes increasingly significant. Text lives on different carriers and in different formats, each influencing how it is perceived and used. As Alan Galey et al. put it, the identity of present-day editors is ‘both new and old, going beyond traditional hermeneutic, explanatory, and editorial
modes to influence the design of new digital reading technologies’ (2012, 21). In other words, the editorial involvement extends over the full lifecycle of text: digital editing connects book history and manuscript research with markup theories, text encoding, programming, prototyping, and interface design.

This study can therefore be located at the crossroad between several fields and disciplines. On the one hand, I engage with the methodologies of textual scholarship, digital editing, and genetic criticism, thus adhering to the unifying idea of the genetic orientation to text. I follow this orientation primarily for its focus on editing manuscripts and its methodology to capture the writing process. On the other hand, this study is guided by the theory and practice of modelling for textual research as proposed among others by Willard McCarty (2005), Elena Pierazzo (2015), and Dirk Van Hulle ([forthcoming]). It is the central argument of my thesis that the use of digital technology for scholarly editing and the genetic orientation enhance our ‘textual awareness’ (cf. Van Hulle 2004) and alter our thinking about text. As more processes are automated and new methods spring from using digital technologies, we have more and better instruments to map the writing process. But we need to pay equal attention to the thoughts that go into making these instruments: how do we automate the processes? And how does that affect our understanding of scholarly editing, of digital editions, and of text?

Van Hulle identifies at least two forms of textual awareness: the modernist writer’s perception of text as a process, and the genetic scholar’s apprehension of the ‘enormous amount of potential energy in many modernist texts’ (2004, 9; 158). I set out to investigate how the digital edition can deepen our insight into text, and what this means for the role of the scholarly editor. The materials of Sheherazade of Literatuur als Losprijs (1932; henceforth Sheherazade), a collection of stories written by Flemish author Raymond Brulez, serve as a case study. At the moment of writing, the Centre for Manuscript Genetics (CMG) of the University of Antwerp is making a digital edition of this work. Most of the textual material has already been transcribed and encoded in TEI/XML. Examples from Sheherazade’s manuscripts are therefore scattered throughout this study. With its complex draft manuscripts and revision history, Brulez’ Sheherazade can serve as an illustration of how genetic editors operate within a stimulating and compelling framework for digital research methods.

I aim to contribute to a further rapprochement between scholarly editing and genetic criticism by showing how critical reflection on digital technologies is as important as implementing them. Each chapter of this study
comprises of illustrative examples of ongoing research in the field of scholarly editing as well as the practical application of certain theories discussed. This way, the study demonstrates the advantage of synthesizing textual genetic research and computational methods. Throughout my research, I look for the benefits of playfulness and serendipity that come with using methods from different disciplines, much alike musicians playing on unfamiliar instruments.

The overall structure of my thesis takes the form of four themed chapters that each address one of the following research questions:

1. Which components of a digital edition have significant potential to be research instruments for textual genetic research?
2. Can information about text be encoded and processed in a way that advances a study of its genesis?
3. In what ways can textual variance be visualized so that the visualization supports and enhances our understanding of the textual genetics?
4. Seeing that textual material is classified in various ways, what systems of organization and classification specifically support genetic research?

The first research question establishes a methodological framework in which the ensuing research questions are studied. It explores how the methodologies of the genetic orientation and of computer science can be combined in order to advance the research into textual genetics. After treating the notion of the genetic orientation to text in further detail, I discuss the application of computational methods in humanities research, more specifically in textual scholarship. A focal point is the distinction between computer-assisted philology and digital philology as identified in contributions by Tara Andrews (2013) and Pierazzo (2015). The chapter concludes by describing a number of modular components of a digital edition that are fundamental to textual genetic research. These components and their (theoretical) implementation in a digital edition are at the heart of the remaining three chapters of the thesis, which focus, respectively, on transcription and collation (Chapter 2); visualization (Chapter 3); and systems of classification (Chapter 4).

Chapter 2 examines the challenging task of modelling the multidimensionality of the manuscript page in a digital transcription. Genetic editors use a variety of tools to capture and structure this aspect, the most notable being the model offered by the Text Encoding Initiative (TEI). In a TEI/XML transcription the textual genesis can be encoded by tagging added,
deleted, and revised text. Furthermore, information about the text’s documentary and temporal features can be added. This results in complex text encodings with different writing stages or ‘layers’ of the text. It is generally accepted that the editor plays a decisive role in encoding the text: making a transcription is a critical act (see Robinson and Solopova 1993, 21; Pierazzo 2015, 100-1). If the layered transcription is subsequently processed by a collation tool, it needs to be reduced to a more simple, linear model. This chapter explores the possibility of so-called ‘XML-aware collation’ that allows preserving the TEI/XML-elements and collating them together with the text. I argue that this method may relieve the editor from the responsibility of selecting one layer and linearizing the TEI/XML-encoded text and thereby significantly reduces the loss of encoded information. The advantages and disadvantages of this approach are assessed through comparative analysis of current digital collation methods as well as a small-scale experiment focused on automated collation with the revised manuscripts of Brulez.

Chapter 3 looks into ways of using a digital visualization as a research instrument. It concentrates on the editor’s mediating position between the reader and the textual material – that is, the ways in which the editor recounts the story of the text’s development. More than other genetic operations that focus on recording and classifying the material and collating the text’s witnesses, visualization is oriented towards the readers/user of the edition. In addition to this communicative act aimed at an external recipient, visualization is also an intellectual act that has the potential to shape the editorial thought process. Therefore, I will also look at how visualization operates as a cognitive exercise, an exercise that requires the editor to answer questions like ‘what argument does the edition make about the textual genesis?’ and ‘how can this argument best be presented to an end-user?’ For reasons of consistency, I explore these and related questions for the topics of transcription and collation. In the last part of the chapter (3.3), I elaborate upon the role of the editor as a mediator between reader/user and text by further exploring the method of the revision narrative. There are numerous ways to explain textual revision, the most famous of which is probably John Bryant’s ‘fluid text’ approach, which he applies to Herman Melville’s Typee (Bryant 2002; 2009). I present a different approach that is oriented towards a larger and more general audience. In collaboration with a web design company and Antwerp’s literary museum, the CMG created a digital museum exhibition that showcases the genesis of Sheherazade. This virtual exhibition combines two notable functions of visualization: the act of modelling
as a cognitive exercise for the editor, and the revision narrative as a way of communicating the editor’s knowledge to the readers/users.

The fourth chapter examines the last component identified in Chapter 1, namely the ‘systems of classification’. Systems of classification entail the various ways in which the material in the genetic dossier can be organized. Of course, classifying documents and texts is an important task in any type of scholarly editing, but in the genetic orientation it plays a key role. Ideally, this facilitates a more thorough investigation of the relationships between document - text - version, allowing the same collection of material to be studied from different angles and perspectives. I examine the various ways in which the documents can be classified and, furthermore, discuss the technological aspects of providing an infrastructure for systems of classification, from knowledge sites to modelling text as a network. This includes the Text as a Graph (TAG) data model that is currently being developed at the Huygens Institute for the History of the Netherlands. TAG will allow editors to store information about text in the form of annotations that can be interconnected. Since the graph structure by definition imposes no hierarchy on the data it contains, the model proposes a solution to the overlapping hierarchy issue that often arises when aspects of text and document are structured in XML. The final part of the chapter (4.3) revisits the experiment with XML-aware collation discussed in chapter 2, and explores the possibility of a ‘two-step collation’, where the results of a textual collation are refined based on information about the document.

Before we continue, it is necessary to clarify several definitions, as words like ‘text’, ‘document’, ‘version’ or ‘work’ can be understood in different ways. This reminds us of the words of Peter Shillingsburg, who once lamented ‘the waste of breath in so many disputes when the same words are used to mean different things’ (2013a). For work, text, and document, I follow the broad, uncontroversial definitions proposed by Pierazzo. Documents are ‘physical objects that contain some sort of inscribed information’. In the realm of textual scholarship, this means a document is ‘a physical object with some text on it’ (2015, 40, emphasis mine). Pierazzo defines Text as immaterial and interpretative, as the meaning(s) readers derive from the inscriptions on a document (2015, 43). We are also mindful of the definition of Dillen, who describes text as ‘a character sequence’ of letters, spaces, and punctuation, including symbols and musical notation (Dillen 2015, 110). A Work represents ‘the quintessence of immateriality [that] can only exist in posse, as an aspiration’ (Pierazzo 2015, 60). In other words, it is implied by the sum of all the
documents and texts derived from the documents. It is a ‘mental construct’ (Shillingsburg 2010) that is represented not only by the variant texts but also by the tension between them.

Another term highly susceptible to interpretation is Version. Like work, a version has no material substance but is implied by text that represents the author’s intention at a certain moment in time. Clearly, such definition is open to debate, prompting questions such as: ‘does adding a word to a text create a new version?’ In Chapter 2 this notion is discussed in more detail, as that chapter deals with collation, including the concepts of ‘word versions’ and ‘sentence versions’ (Van Hulle 2005). The terms Encoding and Transcription, finally, are often used interchangeably, because in present-day editorial practice the text of a document is usually transcribed in the act or process of encoding. Nevertheless, not all transcriptions are machine-readable or even digital in nature. It is useful, therefore, to distinguish several forms of transcription: between analogue and digital we can further classify linear, diplomatic, or ultra-diplomatic. Although a digital transcription could technically also be a .txt file or a MS Word document, here a digital transcription is taken to be synonymous with a marked up text. That implies text is translated to a machine-readable format and marked up, for instance with TEI tags. In all cases, it involves an ‘act of translation’ between different sign systems and, as said above, it always constitutes a critical act of editorial interpretation.

This study emphasizes the editor’s foundational role and continuous presence in manuscript research, text modelling, visualizing variance, and narrating revision. Hers is, above all, a desired presence, for editorial knowledge is essential if we want to successfully store, process, and analyse text. Our research questions have always been informed by curiosity and a desire to plumb the unknown, but at the same time they are shaped by the existing body of knowledge and the prevailing set of scholarly beliefs. My research aligns with the idea of digital philology and ‘thinking through making’. By combining methods from different disciplines and in particular computer science, I aim to show how applying computational technologies to textual scholarship reveals infinitely more about the ways in which text come to be made and mediated, adding immensely to our textual awareness. Following what Edward Vanhoutte called ‘an inclusive approach towards tradition and innovation’, this study argues for the benefits of an interaction and integration of other, unknown modes of investigation.
Chapter 1 - Disciplines and their Methodologies

1.1. Genetic criticism

‘To analyze work habits, we must be able to “go backstage”, to enter into “the workshop, the laboratory” of the writer. Next, to be able to interpret the meaning of what we discover there, we must also be equipped with sufficient theoretical intelligence about these “internal mechanisms”.

(Hay 2004, 19)

Introduction and Definitions

This chapter begins by laying out the theoretical dimensions of research into the process of writing. It goes on to outline the subtle difference between genetic editing and genetic criticism, and explains how taking the perspective of the editor’s orientation to the text may bridge this difference. The introduction already defined my use of the terms ‘text’, ‘work’, and ‘document’, but as this study focuses on the genetic orientation, it is necessary to further define ‘manuscript’ as well. Unless otherwise indicated, in this study the term refers to modern manuscripts. This type of material dates from a period¹ when writers increasingly used pen and paper to structure their thoughts (‘thinking on paper’): modern manuscripts reflect the workings of a writer’s mind (cf. Callu 1993, 65; Van Hulle 2014b, 9; idem 2016, 225). In what follows, modern manuscripts are generally understood as ‘autograph or holograph draft manuscripts’ that are often of private nature (Pierazzo 2009, 179). Another important characteristic of modern manuscripts, according to Vanhoutte, is that they may be divided into temporal units of writing that do not follow the typical chronological order of reading a page – where we read, typically, from the upper left corner to the lower right, filling a page, by contrast, involves distributing text in non-linear and seemingly illogical order (Vanhoutte 2005, online). This last feature poses some specific challenges both for analysis and text encoding, which I discuss in more detail in Chapter 2 (‘Transcription and Collation’).

It is also necessary to further define what we currently understand by scholarly editor. Contrary to copy editor or editor for a press, it is a scholarly

¹ A comprehensive definition of ‘modernism’ and ‘modernity’ is given by Dirk Van Hulle, who describes modernity first ‘as the narrative of the Enlightenment and its aftermath’, and secondly as ‘the multiplicity, of velocity and volatility of modern, urban life’. The period of modernity comprises about 150 years and begins approximately when literary works regarding the modern, urban life are published (Van Hulle 2014b, 8). According to Jean-Michel Rabaté, it reaches its peak in the second and third decades of the twentieth century (Rabaté 2013, 2).
position and therefore implies a critical approach to text. Scholarly editing is part of the field of textual scholarship and as a consequence, the terms 'scholarly editor' and 'textual scholar' are sometimes used interchangeably. Textual scholarship has been regarded as a 'pre-scientific auxiliary discipline' (Lernout 2013, 77): the (invisible) activity of preparing texts at the service of literary scholars. Similarly, Sir Walter Wilson Greg once famously stated that bibliography (a major task of textual scholarship) essentially figures as 'the handmaid of literature' (Greg 1913, 40). In short, the reputation of textual scholarship and those practicing it was, for a long time, rather poor. At the same time, however, textual scholarship is intrinsically linked with almost every discipline in the humanities and human sciences. Our Western society is built around the ‘Order of the Book’ (Van der Weel 2011, 2) and despite the increasing presence of digital technologies in our society, we still live in it. Neil Fraistat and Julia Flanders conclude that the field is highly important for ‘current literary studies of all kinds’ and that it can ‘explain in specific, material ways the constructedness of all texts’ (2013, 2; idem, 8). Its influence can be felt from the canon to the coffee table.

The knowledge gained through textual scholarship is applied to the scholarly editing of texts. There is not just one accurate way of scholarly editing. Traditionally, the practices of scholarly editing could be classified according to periodical and geographical terms, leading to an Anglo-American, a German, and a French tradition.2 Current scholarly editing methods incline towards an eclectic or synthesized methodology rather than a strict traditional paradigm. Indeed, we can observe a ‘coming together of forms of literary scholarship’ resulting in editorial methods that are ‘transnational and transcultural’ (Eggert 2009, 203; McGann 2013, 275). This is undoubtedly engendered – at least partially – by the globalizing effects of the digital turn. Currently, choosing and applying the most optimal approach for a scholarly edition depends on numerous factors such as the nature of the text being edited – its author,

---

2 The latter two are sometimes taken together under the term ‘Continental’ and complemented with Italian, Scandinavian, Dutch, and Flemish approaches (Lernout 2013, 76-77). These traditional editorial schools are the logical consequence of the criteria imposed by the textual material of national authors, such as Goethe and Schiller in Germany, and Shakespeare in England (see Lernout 2013, 76-77; Van Hulle 2014b, 9). At the same time, the divergences are often influenced by complex political, national and academic agendas, all of which resulted in 'identifiable national schools of critical method' (Warren 2013, 120). A number of authors provide elaborate studies into the editorial traditions. See for instance Greetham 2004; Van Hulle 2004a; Vanhoutte 2007, 157; Warren 2013, 120; Lernout 2013.
language – and audience (Vanhoutte 2007, 157). An equally important factor, which can affect her decision-making, is the interest or orientation of the scholarly editor. It has been mentioned earlier that the objectives of scholarly editing, textual scholarship, and genetic criticism do not quite coincide. As others have suggested, this imbalance can be solved by combining the objectives of genetic criticism and genetic editing in the genetic orientation to text (Van Hulle 2004a; 2009; Dillen 2015).

This genetic orientation also informs our study of the genesis of the collection of stories Sheherazade. Although Brulez cannot be considered a modernist writer in the strictest sense, mainly because he was averse to any ideology, Sheherazade offers an almost ideal case study for digital editing with a genetic orientation. The work was written between 1929 and 1931 and Brulez has diligently preserved the materials related to the collection’s creation: notes, writing plans, manuscripts, typescripts, and page proofs. What is more, the story’s principal themes are storytelling and the perpetual struggle of literary creation. Despite a considerable literary production, Brulez himself did not quite enjoy being a writer. So, when attending a staging of the suite Sheherazade in 1929, he recognized in the protagonist a patron saint for struggling storytellers. Sheherazade’s stories constitute her ransom to the Sultan, hence the subtitle that Brulez gave to his work: Literatuur als losprijs (‘Literature as ransom’). Its structure resembles that of Arabian Nights, in which the narrative about Sheherazade and Sultan Shiriar provides a frame for the stories she tells him.

The first publication of Sheherazade as a story collection was in 1932 with publishing company Steenlandt (Kortrijk); it was published again in 1946 at De Nederlandsche Boekhandel (Antwerp). Although Brulez never enjoyed great literary fame during his lifetime, he was generally appreciated as a writer and, more general, for his work in the cultural circuit in Belgium. With the recent

---

3 In a letter to his friend Raymon Herreman, Brulez stated: ‘I don’t write “easily”... Writing is difficult for me (that is, in Flemish. I write more fluent in French). You will ask me: then why do you write? Well, my friend: because once I get a little story into my head I need to get rid of that obsession. This might be why my handwriting is so sloppy. I do not enjoy writing: nor do I enjoy the “urge to create” or the reception of my work. You will ask me: why do you publish? Because I cannot mess with it anymore, once it’s printed in black and white and also (I’m happy to confess) from a certain vanity: to prove my loved ones that I am capable of doing something.’ dated November 2, 1932 (AMVC Letterenhuis B917/B; own translation).

4 Some of the stories have been published as separate items in journals, like ‘De projectielantaarn Aladin’ in Dietsche Warande en Belfort (1931) and the frame story ‘Wat is liefde zonder verleiding’ in Forum (1932).
publication of this biography ‘Gelukkig en vol droefenis’ (2015) by Joris van Parys, *Sheherazade* has been reissued at Houtekiet. This thesis however will concentrate on the genetic dossier of the frame story ‘Wat is liefde zonder verleiding’. The (TEI/XML transcriptions of the) materials will serve as illustration of editing for the genetic orientation, or will be used as a model for the experiments with automated collation (chapters 2.3 and 4.3).

1.1.1. Objectives

> ‘Working manuscripts contain vital clues to how authors worked and writings evolved – clues that the production processes of print regularly erase. In reading manuscripts we get closer to certain aspects or signs of creativity.’
> (Pierazzo qtd. in Deegan and Sutherland 2009, 8)

Everyone who has ever stared at that infamous blank page, trying to find the ‘right’ words, knows that writing is not always easy. At times, language can be a catalyst in communicating our feelings or state of mind. But just as easily, the words we have at our disposal do not suffice to capture or even approximate our thoughts. Through rereading and revising what we have written, we try to come as close to our thinking as possible, to suddenly find that we disagree with our previous selves. At the same time, these thoughts and ideas might have changed in the meantime. As a result, nearly every literary work is the product of a struggle, a slowly growing awareness of semi-shaped ideas. The material evidence of this process (i.e., documents) forms the basis of genetic criticism’s study of this struggle.

The concept of textual genesis is not easy to grasp, but speaks to the imagination, as is illustrated by poetic descriptions such as ‘a continuum of overlapping and interpenetrating actions flowing seamlessly from one to the next text’, ‘continuous revision’ (Bryant 2002, 73), ‘a movement’, ‘textual invention’ (Deppman et al. 2004, 11), ‘the science of written invention’ (Ferrer, qtd. in Van Hulle 2014b, 9), and ‘a productive confrontation’ between the writer’s thoughts and the external forces working upon the text (Hay 2004, 24). Jed Deppman et al. stress that genetic criticism pursues an immaterial object (i.e., the text and its development) based on material traces on the document.

---

5 As stated in the introduction, work on the digital edition of *Sheherazade* is currently ongoing at the CMG. For a comprehensive reconstruction of the genetic dossier, see Fierens 2015.
6 Michel de Montaigne was in fact fascinated by this very process: instead of erasing ideas from his *Essais* he took to the margins and entered into a discussion with his past self.
The feeling one gets from such descriptions is one of chasing an unknown creature, of which there is no accurate description except what we can infer from the traces it has left behind. John Bryant for instance, in analogy with his digital scholarly edition of Herman Melville’s work, compared it to chasing a white whale. A mainstay for genetic researchers is therefore a critical methodology. Within this methodology, it is important to distinguish between, on the one hand, the study of writing traces and, on the other hand, the critical interpretation of the trail they left behind.

**Writing Traces**

The interest in authors’ manuscripts surfaced in the Romantic period. This period is a first incentive towards what Foucault describes as the point where ‘we began to recount the lives of authors rather than heroes’ (Foucault 2006, 281). Geert Lernout lists several phenomena that contributed to the appeal of draft manuscripts and tattered notebooks. Industrialization for instance caused a desire for originality as well as a nostalgic feeling about the past. At the same time, the study of literary manuscripts contributed to the establishment of a national literature and hence culture (Lernout 2004, 290; 294). The growing public interest in literary practices led to an increased awareness among authors about the importance of their personal archives and draft manuscripts. Naturally, such consciousness was especially present if the author enjoyed some fame during his or her lifetime. It is not a coincidence that the writings of authors such as Victor Hugo, Gustave Flaubert and Johann Wolfgang von Goethe formed the subject of the early genetic studies: the fame of these authors and the persistent belief in the idea of a solitary genius contributed to the general public’s interest in autograph manuscripts. It also takes a certain level of self-confidence to believe that later generations might be interested in your literary experiments, preliminary writings and first drafts. In some cases, this resulted in authors actively ‘hiding’ specific textual material they did not like to see as part of their literary heritage (e.g. Van Hulle 2004a, 8-9).

When paper became increasingly affordable and widely available, authors started to plan and outline the structure of their work, practicing and sketching and drafting on paper. Considering the number of late nineteenth and early twentieth-century authors who did this, the period has been described as the ‘Golden Age of the contemporary manuscript’ (Callu 1993, 65; see also
Van Hulle 2014b, 4). The manuscripts of modernist authors have therefore long formed the heart of textual genetic research. Evidently, modernist authors were not alone in using pen and paper to think or preserving their working manuscripts. But in contrast to the limited number of classical and medieval manuscripts, researchers have access to large quantities – an ‘abundance’ even (Van Hulle 2004a, 8) – of modern manuscripts: many national libraries hold ‘the complete manuscripts’ of contemporary authors. Another reason why modernist authors present an interesting subject of research is their interest in their own writing process. The modernist preoccupation with the workings of the mind, the method of ‘thinking on paper’, and the subsequent preservation of writing traces (diaries, correspondence, notebooks, working drafts, etc.), ensure that modern manuscripts provide rather ideal material for detailed investigations into the dynamics of the process of writing.

A focus on textual material and the ‘unique features of every single writing process’ (Van Hulle 2004a, 6) substantiates the concept of the author as creator in the sense of an artisan or crafts person. Considered as a ‘divinely endowed individual’ (Kapoor 1995, 32) in the Romantic period and pronounced dead in later times (most famously by Roland Barthes in 1968), the topic of authorship has always been a hot potato in literary theory. In present times, Barthes’ theory has sufficiently been evaluated, developed and expanded upon and we may conclude that the author can never be entirely omitted from a critical analysis of his work.

This implies that the author should be considered as an important agent in the creative development of a literary work. At the same time, if we consider a literary work as ‘multiple texts in multiple dialogues’ (Bushell 2009, 6), we come to realise that although the documentary traces might be left by the author’s hand, there are other external factors that also influence the development of the work. These factors vary from interventions by individuals

---

7 With regard to western manuscripts, in fact, this golden age has been defined as the period between 1750 and 1950 (Grésillon 2001, 11), a time when paper was inexpensive enough to be used for drafting and sketching, and when authors preserved these drafts out of an interest in the creative processes.

8 If the author is not only an impersonal instrument of divine creativity, it implies that knowledge of the writer’s biography (his/her personal life, work methods, intellectual environment, etc.) might facilitate the understanding of the literary work. This idea has been criticised in later times, especially according to the belief that the author has no immediate influence on the literary work (Barthes 1968). In the same vein, the theory of ‘intentional fallacy’ nuances the idea that one can actually use personal information about the author to a better interpretation of a work (Wimsatt and Beardsley 1954, 3-4).
(publisher, friend, censor, typesetter) to sources of inspiration more difficult to distinguish, like other books. Writing is invariably influenced by social and cultural events such as (self-)censorship, publishing and reading practices, translations and adaptations. If we understand that ‘a literary text is a field of communicative exchange’ (McGann 1991, 62), it follows that a close examination of the writing process and the development of a literary work may give us not only insight into the mind of the author, but also into the way the author was affected by these phenomena. In fact, several scholars of the Centre for Manuscript Genetics at the University of Antwerp currently carry out this type of research. Their work is largely inspired by the monograph Modern Manuscripts (2014), in which Van Hulle brings together the fields of cognitive narratology and genetic criticism.

**Critical Interpretation**

Draft manuscripts, therefore, provide us with some insight into the work methods of authors and give clues to the origins of literary works. They also help us to understand certain aspects or signs of creativity’ (Deegan and Sutherland 2009, 8). The critical interpretation of these signs gives us a better understanding of the work. By publishing the results, the genetic critic assumes the role of a scholarly editor. As such, textual genetic research takes place on a sliding scale of manuscript analysis, critical interpretation, and scholarly editing. Since these methods to a certain extent influence each other, the research entails an iterative approach of continuous refinement of findings. Present-day scholarly editing synthesizes existing traditions and experimental methods, generally guided by (1) the nature of the work; (2) the orientation of the editor; and (3) the medium of publication.

This results in an interesting state of affairs. On the one hand, most ‘innovative’ editorial methods still have their roots in traditional philology and editorial theory. On the other hand, a number of textual scholars emphasize that the use of digital technologies in textual research calls for or leads to a reconceptualization of traditional methodology (e.g., Gabler 2010, 43; Galey 2010, 21; Shillingsburg 2013a). In ‘Orientations to Text, Revisited’, Peter

---

Shillingsburg and Van Hulle reconceptualize scholarly editing methods by focusing on the perspective of the editor. They distinguish five different editorial orientations to text, each of which reflects the main interests and objectives of an editor. An editor’s orientation determines the way that textual analysis is carried out; which factors (e.g., variants, material, form, history) are considered relevant; and which are estimated to be of secondary importance. Orientations are not explicitly linked to any traditional editorial school, although the methodology they imply evidently echoes the editorial-theory movements of the past decades. From among the five orientations defined by Shillingsburg and Van Hulle, the ‘genetic orientation’ is most relevant for this study.

The genetic orientation to text is mainly inspired by the French school of genetic criticism (critique génétique). Central to the discipline of genetic criticism is the study of the creative writing process, based on the draft manuscripts of a writer. There are a number of classic studies on genetic criticism (e.g., Debray-Genette 1979; Grésillon 1994; De Biasi 2000; Hay 2002, Deppman et al. 2004), which are concisely synthesized by Van Hulle in Textual Awareness (2004a). One important theme to emerge from these studies is the difference between genetic criticism and genetic editing. Genetic criticism uses the edition (and the making of an edition) as ‘a tool to facilitate manuscript analysis’ (Shillingsburg and Van Hulle 2015, 37). The outcome of that analysis may or may not result in a published edition. On the other hand, the main objective of genetic editing is the publication of an edition. It does examine the genetic dossier, but this manuscript analysis is considered ‘as a means to an end’ (37): as a tool to create and publish an edition. The line between genetic criticism and genetic editing is thin and often blurred. Although both approaches have different objectives, they are both oriented towards the writing process and they share the same methods. Genetic criticism is a form of textual research that may or may not result in the making of an edition; genetic editing on the other hand considers the making of an edition as the goal of genetic research.11

10 The ‘genetic dossier’ or dossier génétique is formed by the complete documentation relating to the genesis of the literary work, e.g., preparatory documents like draft manuscripts, notes, and writing plans, but also corrected page proofs or correspondence related to the work.
11 The ERC-funded project ‘CUTS’ (University of Antwerp) is set out to examine exactly how the creation of a genetic edition can be an instrument for genetic research. See http://bit.ly/2djG62V for the project description (last accessed September 23, 2016).
The concept of ‘orientations to text’ as proposed by Shillingsburg and Van Hulle may form a constructive strategy to overcome this division. They define the ‘genetic orientation’ to text as an editorial methodology that focuses on the trajectory or trajectories of creative development, including sources of inspiration. It does not have to, but may involve a form of scholarly editing. If it does, it implies an editorial strategy that displays creativity in motion rather than settling on a final version as the main object of editing (Shillingsburg and Van Hulle 2015, 37). As Dillen points out, using a category such as ‘genetic orientation’ allows for the inclusion of other editorial schools that study writing processes and draft manuscripts (Dillen 2015, 15). One of the main objectives of the genetic orientation is to produce ‘a narrative of genetic development’ (Shillingsburg and Van Hulle 2015, 37), so it focuses both on the ways to build the genetic narrative (editing) and on the narrative itself (criticism). Incidentally, taking process and product as two complementary forces does not belong exclusively to the genetic orientation only. Chapter 1.2 (‘Computational Methods’) describes for instance a similar relationship regarding the use of computational methods in digital editing. In fact, this reciprocity runs as an undercurrent throughout this thesis. I take scholarly editing as a main topic, but focus especially on the process of doing (manuscript) research. Taking inspiration from Shillingsburg and Van Hulle, I circumvent the distinction between scholarly editing and computational research, and focus instead on the production methods of genetic narratives and the presentation of the research findings. As such, I aim for a balanced exploration of genetic criticism as well as digital scholarly editing. This approach allows me to touch upon the renewed position of the digital editor who increasingly uses the computer to achieve the objectives of both fields. In Chapter 2.3, for instance, I examine how to automate a collation that takes into account revisions in a manuscript, and in Chapter 3.3 I discuss the endeavour of developing an interface in which the ‘story’ of *Sheherazade’s* genesis is explained in a visually attractive way.
1.1.2. Methodologies

'Knowing how something originated often is the best clue to how it works.'
(Deacon 1997, 23).

In view of the considerable amount of research on genetic criticism and genetic editing,\(^\text{12}\) the following paragraphs are limited to clarifying some of the main concepts relevant to my thesis. In my thesis I will use the term ‘genetic orientation’ to refer to the combined efforts of textual genetic research and scholarly editing. To take a closer look at what the methodology entails, I will follow the typical workflow of a genetically oriented editor. This exercise will largely be based on Pierre-Marc De Biasi’s comprehensive description of the successive operations of a genetic editor (2004, 43-60). Other methodological descriptions can be found in Almuth Grésillon’s famous *Eléments de critique génétique* (1994) and Paolo D’Iorio’s ‘constituents’ of genetic criticism (2010).

De Biasi distinguishes roughly two stages in the practice of genetic editing: (1) assembling the documents and preparing them for reading and analysis; (2) reconstructing the genesis (De Biasi 2004, 42). The first stage, called ‘manuscript analysis’, usually constitutes four successive operations: (a) the assembly or collection of documents and other types of material considered as related to the genesis of the text; (b) the classification of each document in that dossier; (c) the organization of the complete corpus of documents; (d) the deciphering and transcription of the textual material. Although at this stage little analysis takes place, these tasks cannot be carried out mindlessly: the classification, organization, and transcription of manuscript material all entail considerable scholarly interpretation. The second stage then concerns the reconstruction of the genesis. Through a critical evaluation of the assembled material, the genetic researcher ‘transforms’ the corpus of collected extant manuscripts into a genetic dossier. The separation of these two stages is partly artificial: a genetic editor may (consciously or otherwise) already anticipate the reconstruction of the text’s genesis during the first phase of classification and organization of the documents, and conversely the organization of documents may be fine-tuned or adjusted during the second editing stage.

\(^{12}\) Apart from the aforementioned literature, Paolo D’Iorio (2010, 49) suggests some further required reading specifically on genetic editing: Lebrave 1994; Zeller and Martens (eds.) 1998; De Biasi 2000; Stussi 1994.
Clearly, then, the term ‘genetic dossier’ is slightly ambiguous. Almuth Grésillon defines the genetic dossier as the ‘ensemble de tous les témoins génétiques écrits conservés d’une œuvre ou d’un projet d’écriture, et classés en fonction de leur chronologie des étapes successives (Grésillon 1994, 242). In this definition – and as Grésillon also suggests – ‘genetic dossier’ would be synonymous for ‘avant-texte’. Conversely, De Biasi differentiates between these two concepts when he calls the dossier containing all of the genetic documents of the work the genetic dossier, and an organized system of these documents the ‘avant-texte’. Regardless of this distinction, Pierazzo points out that the term ‘genetic dossier’ is better suited than ‘avant-texte’. If we take ‘avant-texte’ literally, she argues, it implies a form of pre-publication, whereas genetic criticism concentrates on the writing process and not the final product or publication (2015, 14). For reasons of clarity, therefore, this thesis uses the term ‘avant-texte’ solely to refer to the organized system of documents that represent the text’s genesis as perceived by the genetic researcher.13 The ‘avant-texte’ is constituted in the second phase of research, after a first round of manuscript analysis.

If we now place De Biasi’s successive stages of genetic research in the light of the distinction made by Shillingsburg and Van Hulle between editing and criticism, a more precise view of the workflow of a genetically oriented editor emerges. At first glance, De Biasi’s two phases correspond to the distinction between genetic editing (phase one) and genetic criticism (phase two). However, he concludes that ‘the essential objective of textual genetics is to produce genetic editions’ (2004, 61) and that these editions subsequently allow for new research into the work’s genesis. Both phases described by De Biasi thus appear to fall under genetic editing. Accordingly, De Biasi’s study underlines that genetic research is characterized by a ‘back-and-forth play’ between editing and criticism (2004, 44), similar to the difference between product and process.

With this in mind, we can distinguish two forms of genetic research. First, the scholar examines the full body of manuscripts and carries out a series

13 Admittedly, the variety of definitions and the subtle differences between them can make one’s head swim. An extensive yet clear overview of definitions and terminology is provided by the Lexicon of Scholarly Editing (LSE): a multilingual lexicon that gathers existing definitions of scholarly editing concepts. It is initiated by Van Hulle and the European Society for Textual Scholarship (Ests) in 2012, developed and maintained by Dillen, and hosted by the University of Antwerp. At the moment of writing, the LSE comprised of 418 entries and 871 definitions in inter alia English, French, German, and Italian. It has 12 registered contributors, of which I am proud to be one. See http://uahost.uantwerpen.be/lse/ (last accessed May 23, 2017).
of genetic research operations, like classifying or transcribing the documents, which normally also leads to a publication (cf. the two phases by De Biasi) in the form of a genetic edition. This publication may subsequently inspire new forms of (genetic) research that build upon the existing research, although that research generally does not result in the publication of yet another edition. Rather, it supports literary analysis or, as the Beckett Digital Manuscript Project (BDMP) \textsuperscript{14} states, it ‘opens the manuscripts’ hermeneutical potential’ (2016, online). Despite this fluidity between the phases of genetic research, it is useful to maintain an artificial distinction. That is, the description of the operations (e.g., transcription, analysis, classification) allows us to assess the work of the genetic researcher in the light of the use of the computer. Foregrounding specific editorial operations, like ‘transcription’ or ‘collation’, permits us to develop specialized tools. At the same time, the development of such tools requires us to devise, step-by-step, what exactly editors do when transcribing a text, and how a computer can facilitate that work. Each of these operations brings about a number of specific questions and challenges.

To better understand and study the various textual changes that can be perceived on autograph manuscripts, they are classified into the categories of \textit{exogenesis} and \textit{endogenesis} (Debray-Genette 1979, 28) and \textit{epigenesis} (Van Hulle 2014b, 14). The first, exogenesis, is concisely described by De Biasi as ‘any writing process devoted to research, selection, and incorporation, focused on information stemming from a source exterior to the writing’ (1996, 43-4). This concept is often illustrated by a writer who, inspired by a book, annotates in the margin some words that recur in a later version of his own text (cf. Van Hulle: ‘very often the genesis of a book starts in the margins of another book’, 2014b, 8). Consequently, the personal library of an author forms a valuable source of information for a genetic researcher who is particularly interested in exogenesis. Endogenesis is, again in the words of De Biasi, ‘the process by which the writer conceives of, elaborates, and transfigures pre-textual material, without recourse to outside documents or information’ (1996, 42-43). This typically begins at the first moment of inscription and ends with the author’s decision that the text is ready for (first) publication described as the \textit{bon à tirer} moment (Grésillon 1994, 241; De Biasi 1996, 37).

\textsuperscript{14} The BDMP project, carried out by my colleagues at the CMG, produces a bilingual digital genetic edition of Samuel Beckett’s works. Due to copyright, the BDMP can offer only restricted access, but a demo is available on \url{http://www.beckettarchive.org/}, under ‘Free Features’ (last accessed October 25, 2016).
According to Van Hulle, exogenesis and endogenesis can be seen as ‘two movements in the processing of extratextual material’ (Van Hulle 2004a, 7): the author first gathers extratextual material and subsequently assimilates it into his or her own texts. Van Hulle proposes a third process, epigenesis, to describe the continuation of the genesis after the text has been published; after the moment where the author first decided the text was ‘ready’ to be published. This would then be the après-texte (Van Hulle 2014b, 14). Epigenesis takes into account the various social and cultural factors that influence the author, such as a publisher or a censor, and may lead to additional revisions in later publications. Good examples of epigenesis are Samuel Beckett’s reworking of a play after it had been staged, and the ‘evolution’ of Charles Darwin’s Origin of Species that was updated with each of its six publications (Van Hulle 2014, 99-102).

Taken together, these three stages represent the writing process on a detailed level. While exogenesis and endogenesis describe a mode of writing, epigenesis is defined in relation to the bon à tirer moment. Epigenesis takes place in the post-publication phase and would therefore fall in the domain of textual scholarship and not genetic criticism, as Van Hulle points out (2015, 50). In practice, however, rewriting consists again of the combination of exogenesis and endogenesis. It is therefore important to emphasize that the distinction between them is also partly artificial, and that they do not necessarily take place in a chronological order. Again, there is a continuous interaction, a back-and-forth movement, of which the documents of the genetic dossier bear the traces. As digital editors, we aim to capture the nature and origin of these traces as best as possible: the more detailed our model of a specific writing process, the better we can examine how they inform and shape each other.
1.2. Computational Methods for Humanities Scholarship

‘The fundamental question of computer science is simply “What can be computed?”’
(Zelle 2012, 2)

‘(D)igital technologies constitute the greatest opportunity we have had to redraw the map of knowledge since the times of Erasmus and Leon Battista Alberti.’
(Usher and Fiormonte 2001, xi0)

Introduction and Definitions

It is no light claim to say that digital technologies offer the opportunity to redraw the map of knowledge, but Usher and Fiormonte are certainly not the only ones to make it. When the authors published their volume in 2001, the use of computational technologies in humanities research was first taking flight. The practice was then referred to as ‘humanities computing’ and it was expected to bring great methodological changes, in fact, to ‘upgrade’ the entire discipline of the humanities.15 Indeed, in the following decade the number of researchers experimenting with digital technologies grew exponentially. We now have a wide variety of terms to describe this type of research, ranging from ‘e-research’ to ‘data-driven science’.16 Currently, the most widespread designation is ‘digital humanities’, although a growing number of scholars have voiced their discomfort with this term.17 The proverbial big tent of the digital humanities may have become too big and too vague for researchers to identify with. Dino Buzetti proposes to (re)focus on the methodological consequences and ‘theoretical import’ of computing and has reintroduced the term ‘humanities computing’.18

Finally, the words ‘data’, ‘information’ and ‘knowledge’, which are sometimes used interchangeably, have to be distinctly defined. In the context of philology, a comprehensive set of definitions is offered by Van Peursen et al (2010). Data are defined as ‘the sequences of graphemes that constitute a text’. This data transforms into information once it is ‘processed to be useful, and

15 Cf. Rens Bod’s definition of the humanities 2.0 – the use of pattern recognition tools on humanistic material – and the humanities 3.0 – the synthesis of computational and humanistic/hermeneutic methods (Bod 2013).
16 See for instance Wyatt (2016, 519) for an overview of the terms used.
17 See for instance the discussion threads on the Humanist Discussion Group 29.899 to 29.910.
18 In a response to the Humanist Discussion Group 29.906 on 29 April 2016.
when meaningful connections between data are established’ (21). In other words, information can provide answers to ‘who’, ‘what’, ‘where’, and ‘when’ questions, whereas knowledge provides answers to the ‘how’ and ‘why’ questions.

Terminology aside, did humanities scholars really manage to redraw the map of knowledge as envisioned by Fiormonte and Usher, or is this reconceptualization still to come? And how exactly do digital technologies provide the opportunity to do so? If computing is what Willard McCarty called ‘the mediation of thought by the machine’ (qtd. in Meister 2002), the way we examine and think about text will take on new and different levels of abstraction. It comes as no surprise, then, that the debates about digital methods in textual scholarship and in the humanities as a whole have been dominating the field for a while. The present section explores this question from the perspective of editing for the genetic orientation and tries to establish whether the use of computational methods changes the way research is carried out. The main focus is on methodological and epistemological concerns. In the first part, I define two different uses for the computer in textual scholarship. In the second part, I discuss the most important method in computing, modelling, and goes on to discuss the consequences of implementing that method in textual scholarship.

1.2.1. ‘Computer-assisted’ or ‘Digital’? Different Ways of Editing

It is generally acknowledged that the merging of different methodologies, in this case those of textual scholarship and of computer science, can yield unexplored research territories. There are two different approaches to the use of the computer: on the one hand, using the computer as a facilitator for humanities research, and on the other hand applying computational methods to humanist material. This distinction is a subtle yet fundamental one. The former approach does make use of the affordances of the computer – e.g., its calculative and mnemonic powers – but only to carry out traditional forms of textual research. Examples are transcribing text in an XML editor like oXygen; or using a digital collation tool to compare texts. This method is perfectly valid by itself. The use of such technologies already compels a scholar to clarify her intuitive thought processes and it could thus lead to innovative results. However, the second approach takes a step further towards experimental research and intends to use
computational methods to create original digital objects that could indeed advance humanities scholarship. The aim, then, is to produce ‘virtual knowledge’, a form of knowledge that Sally Wyatt et al. define as:

creativity, potential, and dynamism in combination with actual practices and understandings. It also emphasizes the ongoing dynamics of change, both in the form and content of knowledge and in the craft of generating new knowledge (Wyatt et al. 2013, 11-12).

The distinction can be extended to the field of scholarly editing. There is the traditional form of editing that is supported by digital technologies and described as ‘computer-assisted philology’ (Pierazzo 2015, 113). The main characteristic of this approach is a division of intellectual labour between the editor and a technical assistant (that is either a human being charged with text encoding, or the computer itself). Simply put, in this scenario the assistant is merely a support or facilitator, whereas the editor continues to perform the critical or intellectual work. It is also implicitly taken for granted that the assistant does not contribute significantly to generating novel forms of (virtual) knowledge. Pierazzo outlines three main criticisms of computer-assisted philology (2015, 111-113). The use of computational technologies without understanding how they function, without knowing the assumptions and biases on which they are build, has serious consequences for the results produced with these techniques. Furthermore, she points out that every tool reconceptualises the world, which means that the use of the computer – even for basic editorial tasks – should be considered as intrinsically different from print editing. Finally, computational tools that are developed for a small group of scholars are often unsustainable and, in most cases, cannot be used in other projects.

Applying computational methods to humanist material, by contrast, is central to the approach of ‘digital philology’ (Andrews 2013; Pierazzo 2015). Pierazzo defines it as a type of ‘experimental textual scholarship, able to engage with sophisticated computational models searching for new solutions’ (2015, 205). Here, the computer plays a prominent role in both the process and the outcome of the textual research: its presence is acknowledged, embraced, and, indeed, essential to research. In fact, its application forms a separate topic of study in its own right. Over the course of this chapter, it will become clear that the boundaries between computer-assisted philology and digital philology are porous. Both editorial approaches can be seen as complementary and may even be applied in the same editing project. In addition, editors who start off as computer-assisted philologists may progress to become digital philologists.
Nevertheless this distinction, although very simple, provides a useful frame for a discussion about the different methodological consequences of computing in textual scholarship.

Computer-assisted philology and digital philology differ mainly with regard to their respective research objectives and (envisioned) output. The principal objective of computer-assisted philology is to produce a digital edition or publish a digital text. In contrast, digital philology focuses on experimenting with computational methods to examine a text in novel ways. The computer plays a crucial role in the process of research, and not just a supportive one. This has an obvious effect on the output, because experimentation does not necessarily result in successful or presentable products and, in case of ‘failure’, it may not lead to any tangible output at all. Therefore the value of this type of research can best be measured by its methodological and epistemological implications. This is not to say digital philology implies a complete freedom to ‘fool around’ and experiment away without regard to the consequences: in most cases, scholars continue to study texts and maintain the wish to represent their findings, often together with a digital version of the text. Digital philology could still result in a digital edition, but that may be ‘just one of the outcomes of the research, [as] the embodiment of a new editorial theory or model, and not its most important outcome’ (Pierazzo 2015, 203). Instead, the emphasis is on examination with computational methods and novel ways of conducting digital research instead of on the publication of text. An additional result of digital philology is therefore that it questions or improves existing editorial methods. Digital philology is willing to embrace the consequences of digital technologies, to consider new perspectives and to question or improve existing methods.

In general, editors select a certain method according to the desired outcome, but this is inevitably influenced by the expectations and models of grant suppliers or funding bodies. Because it translates reasonably well to recognizable scholarly products, the typical outcome of computer-assisted philology (i.e., a digital publication of the edited text) is usually more accepted by advisory committees on retention, promotion, and tenure. In that respect, computer-assisted philology is similar to traditional ‘analogue’ editing, which also focuses on delivering a final product with a (newly established) edited text. However, even when the computer is only used as research assistant and the digital edition follows the conventional forms of philology, the machine still affects the way research is carried out. This becomes clear if we take a closer look at text encoding. As said earlier, it is generally acknowledged that this is far
from mechanical and objective, but rather an intellectual activity that compels the editor to translate her analysis into a computer-readable format. Text encoding is not simply copying text off a document; it requires an active engagement of the editor on a practical (technical) level and a theoretical (abstract) level. For editors not used to expressing their views in such explicit terms as those required by a computer system, this abstraction can be problematic. The same applies to the idea that each text encoding system, whether an existing standard or custom-made, articulates a specific view on the text. Finally, a text encoding system can only work when it is as formalized as possible and thus disregards the idiosyncrasies of the text in question. In short, the use of computers affects both the methodologies and the results of textual scholarship.

We can identify a number of reasons, some of them more problematic than others, why scholarly editors would opt for computer-assisted philology over digital philology. Editors might consciously decide to follow the more traditional approach of computer-assisted editing because their main goal is the publication of a digital text; or they may do so out of an intrinsic reluctance to adopt computational technologies. In his study of scholarly practices, Wolfgang Kaltenbrunner observes that many traditional scholars still consider text encoding as 'data work', as a 'subordinate' task that they prefer to delegate (2015, 86). Other factors at play here are the lack of consensus about digital editing methods, and the tendency to prefer customized models to established standards and formal models (Andrews 2013). Finally, there is the obvious difficulty of getting the computational work recognized by the academic community and funding bodies (Wyatt 2016, 521; Pierazzo 2015, 203-5). These obstacles are discussed at greater detail in the following section.
1.2.2. Computational Methods for Editing

*Putasne brevi immittere vasculo mare totum?*
*Do you think you can put the whole sea in a little vase?*
(Eusebius Sophronius Hieronymus, ca. 1485; qtd. in Galey 2004, 97)

**The Principles of Modelling**

Despite their different objectives, computer-assisted philology and digital philology both make use of modelling. Within digital scholarly editing, and digital humanities research at large, the concept of modelling is one of the most widely adopted methods from the computer sciences. Simply put, modelling is a way to represent and test a theory. It includes the making of abstractions and as such the concept is not unfamiliar in the humanities, although the use of the term itself is relatively new. Moreover, as we will see in this section, the fact that this activity is now understood within a computational framework has additional consequences for the practice of modelling. ‘Modelling’ is thus an ambiguous term - or at least used in an ambiguous manner - it can mean anything from the realization of a concept to the broader meaning of the concept itself (cf. Ciula and Eide 2014, 36). This section does not aim to encompass the entire discussion of models and modelling, but concentrates on the usefulness of the methodology of modelling in relation to textual scholarship and editing.

The reason for the use of models in digital editing is simple: computers require them to function. Any form of editing that uses the computer therefore makes use of models. If modelling is consciously used as a technique (cf. digital philology), it can function as a research method. Paul Fishwick describes different types of models and demonstrates for each of them how modelling helps scholars to conceive and understand an object (or any other phenomenon). He lists three methods: (1) building actual, real-life models based on theoretical designs; (2) conceptualizing models; (3) writing computer programs. Modelling is most useful when taken as an act, as a process, and this process can be divided into two types: ‘modelling of’ and ‘modelling for’ (McCarty 2004, 255). This distinction is a good illustration of the abovementioned ambiguity between conceptual models and objects to be modelled. In short, ‘modelling of’ refers to ‘the theory underlying the modelling process’ (Ciula and Eide 2014, 37). It is ‘a descriptive simplification of a
complex object’ (Pierazzo 2015, 39) ‘Modelling-for’ is the actual implementation of this theory into something new. In other words, a theory of scholarly editing (model of) can function as a model for a specific scholarly edition. The two types of modelling are not exclusive but rather complementary actions: usually a researcher will go back and forth between the implementation of a model and the adjustment of that model. This emphasizes the dynamic nature of modelling: it is a continuous process of investigation, refinement, and application.

Before we go on to discuss modelling in relation to editing and the (epistemological) consequences of this practice for textual scholarship, it is important to keep a number of things in mind. First, models are always representations and simplifications of the original. The act of modelling has accordingly been described as ‘finding simplicity in complexity’ (Wainwright and Mulligan 2004). This simplification serves a crucial purpose as the more complex the model, the less useful it becomes. By emphasizing a certain component of the object to be modelled, many of its other features fall away. Modelling is a dynamic process of checking whether the model fulfils its purpose and refining it where needed: complex and detailed models are more difficult to test. Once the first simple model succeeds (i.e., meets the expectations stated in advance), it can be developed and scaled up, for example by including more properties of the original object. Modelling is, finally, an iterative heuristic process of manipulating representations.

We can make a second important distinction between implicit and explicit models. In his article ‘Why Model?’, Joshua Epstein explains that with implicit models ‘the assumptions are hidden, their internal consistency is untested, their logical consequences are unknown, and their relation to data is unknown’ (2008, online, 1.3). Implicit models would thus not make a good basis for computing, which requires explicitness and consistency. This echoes one of the points of criticism on computer-assisted philology: making use of computational tools without being aware of their inner functionalities. In explicit models, on the other hand, the underlying assumptions are formulated in detail, which ensures that the results can be replicated (and thus tested) and the model can be reused on other datasets. One of the consequences of explicit formulations is that ‘models make their creators re-think and question their theories’ (Ciula and Eide 2014, 36). A second consequence is that it becomes easier to communicate and work with other disciplines. For the digital
humanities, a field that is continuously in flux and prides itself on multidisciplinary collaboration, this is useful indeed.

It is worth noting how Matthew Kirschenbaum describes the value of a computational model. He stresses that researchers must learn to consider the models as ‘rhetorical and ideological spaces’ and examine them in the same way that they study literary works or film where the rhetoric and ideologies of the maker are included in the analysis. His point is confirmed by Paul Fishwick who emphasizes that models are interpretations too: they are after all based on the modeller’s knowledge and conception of an object (2016, online). Essentially, modelling can be considered as a form of translation: it aims to deconstruct an object and translate it into software. The subject of this translation is not so much the original object; it’s the scholar’s preconception of that object together with his or her research objectives. The scholarly purpose and the research interest of the creators (cf. the orientations of the editor) shape the model of the text or the edition, which in turn influences its user’s reading of the text. Moreover, since many disciplines use the technique of modelling, a model is particularly context-dependent. As such, the process and outcome of modelling also sheds light upon our own preconceptions and assumptions. An understanding of how a model is created (and by whom) heightens a critical awareness of the ways in which they function and can be used.

**Modelling in Textual Scholarship**

A clear way to illustrate how models may function for textual scholarship is by linking them to editorial practice. Øyvind Eide and Arianna Ciula (2014) put forward the TEI Guidelines and the CIDOC CRM as two functional models for cultural heritage research. They show that even in this relatively small sector, the use of modelling is widespread and diverse. Furthermore, Pierazzo’s *Digital Scholarly Editing* (2015) describes a large number of digital practices and as such paints a comprehensive picture of the different ways in which the field currently makes use of models (see also Bleeker [Variants 12, forthcoming]). Van Hulle provides another useful example by drawing parallels between the approach of the editors of the BDMP and McCarty’s distinction between ‘modelling for’/‘modelling of’ ((forthcoming)). Taken together, these studies outline two critical conditions of computational modelling: (1) consistency and explicitness, and (2) selection.
First, every computational model needs to be controllable and manipulable by the computer. This necessitates that they are constructed with the goal of ‘complete explicitness and absolute consistency’ (McCarty 2004, 258), as Epstein’s definition of explicit models suggests. The challenge is that the knowledge of humanities’ researchers is not that easily expressed in explicit principles. Consider for instance the expertise of an editor according to Peter Robinson:

Scholarly editors ... must know many, many things. They must know how texts are constructed, both as document and work instance; they must know how they were transmitted, altered, transformed; they must know who the readers are, and how to communicate the texts and what they know of them using all the possibilities of the digital medium (2013, online).

It would be practically impossible, then, to computationally model an editor’s understanding of how Marcel Proust wrote A la recherche du temps perdu. In his reconstruction of this particular writing process, Van Hulle makes use of inter alia the correspondence between Proust and his publisher(s); a number of analyses and interpretations by other Proust scholars; and suggestions based on educated guesses (2004a). This critical analysis is a crucial part of the methodology of genetic criticism (as made clear in chapter 1.1. ‘Genetic Criticism’), but it is difficult to express in explicit rules. Conversely, as the digital prototype made by Julie André and Pierazzo demonstrates (2012, online; André and Pierazzo 2013), it is much easier to model a smaller or specific phenomenon, such as the writing sequence of two pages of Proust’s notebook 46. McCarty concludes that the ‘computational form, which accepts only that which can be told explicitly and precisely, is thus radically inadequate for representing the full range of knowledge’ (2004, 256). What, then, determines our grounds for selection?

Just as the only complete map of the world would be the size of the world itself (cf. Rosenbluth and Wiener, qtd. in Pierazzo 2015, 38), the only ‘complete’ edition of a literary work would be the original work itself. For textual editing, then, Pierazzo states that 'the edition represents a selection of the infinite features of the work, organized according to [a] particular point of view, the one of the editor’ (2015, 38). This leads to an interesting situation if we consider the explicit and consistent formulation of rules. On the one hand, the selection of features is based on an editorial orientation; on the other hand these selected features need to be described in the most formal and
comprehensive manner possible, leaving little room for implicitness or subsequent interpretation.

Both computer-assisted philology and digital philology make use of models, but they do so in different ways. Computer-assisted philology uses the text encoding systems mainly as a modelling tool to produce a new edited text. In that sense, it is closely related to modelling for, whose principal objective remains the creation of a new object.\footnote{Although Ciula and Eide assert that it might also 'generate new knowledge and can very well be part of research projects' (2014, 36).} Digital philology can then be classified as modelling of, whose general aim is to question existing theories through investigation. But, once more, in reality the distinction is less clear-cut. When Ciula and Eide describe how the TEI Guidelines can be used as a model for an edition, they acknowledge that at the same time the TEI proper constitutes a standard and, as such, is also a model of a text (2014, 37). To be more precise, the TEI Guidelines propose a model of a transcription (Robinson 2013a). Additionally, we can say that the continuous efforts of the TEI community to improve the TEI Guidelines are part of the iterative, self-reflexive process that McCarty considers part of modelling of. This blurring of lines is also evident in Alan Galey’s ideal of a digital editor. On the one hand, Galey defines a digital editor as a researcher that undertakes to create and publish an online edition (cf. computer-assisted philology and modelling for). On the other hand, he gives an exhaustive list of technical skills that digital editors could use, arguing that

[w]hat digital textual scholars need to know, then, may be learned by reckoning with our unknown knowns concerning knowledge work, and by rediscovering what we already know about our own mechanic exercises (2012, 97).

Galey goes on to make the case for opening the ‘black box’ of the computer, and for understanding the dynamics of programs and algorithms in order to make informed judgments (cf. digital philology and modelling of). He refers to Kirschenbaum who sees it as essential that researchers acquire ‘an appreciation of how complex ideas can be imagined and expressed as a set of formal procedures—rules, models, algorithms—in the virtual space of a computer’ (qtd. in Galey 2012, 94). In short, there exists a dynamic and continuous exchange between modelling for and modelling of, similar to the relationship between computer-assisted philology and digital philology.
Epistemological and Methodological Implications

The observations above may help us understand the consequences of using computational models and modelling in textual scholarship. The situation that arises is summarized by Van Peursen’s question:

how do computation, calculation and sorting of data relate to traditional values of scholarship that seem to escape calculation, such as subjectivity, the master’s eye, and the intuition of the experienced scholar? (2010, 4)

The continuous search for an approach that could answer this question, an approach that allows scholars ‘to capture the messy complexity of the natural world and to express it algorithmically’, has been described as the holy grail of computer science (Davidson 2006, 66). As with all daunting tasks, it is best to start small. In the case of digital editing, this means selecting a small or specific phenomenon to model and, if it functions, expanding upon that. One tested approach to modelling digital editions is put forward by Pierazzo, who writes that it is best to start from the end product, think about what the final product should look like, and work backwards. This gives textual scholars the possibility to concentrate on the research questions they want to answer, and to ‘think out of the box’ or at least out of the page paradigm (2015, 105). The challenges are, then, (1) to select a coherent set of data that represents the complexities of the original object. Or to be more precise: a data set representing the complexities that one wants to examine. Furthermore, (2) to formulate the questions and objectives the model should provide an answer for. As Pierazzo writes, ‘[w]e must model, we must select, and we must establish limits to the selection process, and these limits represent the boundaries within which the hermeneutic process can develop’ (2015, 38-9). A third difficulty (3) is to make sure that the model meets the standards of the community. In the editing community, where standards may function as models and vice versa, it is an ongoing challenge to define a fixed set of standards.

Modelling, then, means finding a balance between on the one hand solving ‘the inherent complexity of modelling the subject matter’ (Middell 2013), and on the other hand ensuring that the model meets generally accepted standards.

20 In the article ‘Measure for Measure: Exploring the Mysteries of Conducting’, an interview with Teresa Nakra by Justin Davidson in the New Yorker magazine (21 August 2006), 60-69. Needless to say, the concept of the natural world is also subject to personal perspective.
standards (in short: that it is useful for and useable by others). As we have seen, establishing formal rules may have the positive consequence of gaining a different and often deeper understanding of the material, but it can be a veritable challenge to translate existing scholarship into formal and explicit principles. Moreover, this formalization takes place on two levels: first, on the detailed level of a particular text; secondly, on the more abstract level of the literary work and its place within the field of research and the wider academic community. The former is central to the next chapter and is illustrated by a discussion of the rules that can be derived from an XML encoded text in order to process it computationally and with a meaningful result (Chapter 2, ‘Transcription and Collation’). The following section focuses on the latter and discusses what is at stake when we try to translate the standards of the text editing community into a formal set of rules for digital editing. It discusses two aspects that are especially difficult to formalize and yet essential for editing: subjective expertise and instinctive decision-making.

**Formalization: Subjective Expertise**

Scholarly editions are quite idiosyncratic as a result of the material, the creators, as well as the intended community. Consider for instance the variety of digital products that make up the Catalogue of Digital Editions (Franzini 2016)\(^{21}\) and the catalogue of Digital Scholarly Editions (Sahle 2013 [last update 2016]),\(^{22}\) and how each edition focuses on a slightly different aspect of the historical sources it aims to represent. The text editing community can be subdivided into a number of fields or disciplines, e.g., palaeography, linguistics or codicology, each with their own set of standards. As a common denominator, these subfields share the norms and values of editorial work in general. In the twentieth century, these values were summarized by classical scholar Gilbert Murray, who argued that an editor was supposed to be ‘of the highest moral character, religion, and social grace’.\(^{23}\) These ‘virtuous qualities’ were supposedly needed for the handling of a text and the building of an apparatus for laymen. One may ask whether it is at all possible to explicitly formalize these virtues, and more importantly whether

---

\(^{21}\) The data is available at [https://github.com/gfranzini/digEds_cat](https://github.com/gfranzini/digEds_cat) (last accessed July 1, 2016)


\(^{23}\) This citation came up in the Humanist Discussion Group on the virtues of the scholarly editor. Murray’s definition is first provided by Charles Archer (1936, 36-7), and is also cited by Duncan Wilson (1987, 114).
the current digital practice might require a different set of values than Murray’s. The idiosyncrasy of scholarly editions hinders the abstraction of explicit rules and consequently hardens the construction of generic models. How can we translate qualities such as subjective expertise and critical analysis into computational models? Pierazzo attempts to conceptualize the principles of digital editing by describing a set of digital editorial models that are decidedly different from traditional editing because they are born-digital: phylogenetic editions, social editions, crowd-sourced editions, and paradigmatic editions (2015, 17-32). Each of these types is an editorial model that can vary greatly depending on its creators but, in the best cases, the assumptions behind the model are clearly formulated, and can be studied and questioned. As it remains difficult to apply these digital editorial models to different textual material, they may be located on a sliding scale of Epstein’s implicit and explicit models. Pierazzo names a number of ‘dimensions’ that a reader (i.e., an editor) can distinguish on a document, such as the linguistic dimension and the literary dimension (2015, 41-2). She underlines that the choice for a particular dimension is an individual one; we may add that it is also – partly – an unconscious or instinctive choice, affected, for instance, by the cultural environment of the researcher.

Taken together, the editorial orientations to the text, the dimensions of the document, and the variety of digital editorial models tell us something about the values of digital editing, which turn out to differ little from the more traditional principles of analogue editing. First, the concept of scholarly expertise remains, ideally, a balanced combination of trained knowledge and interpretation leading to a reliable judgment of what is on the page. Next, the interpretative aspect of the digital edition needs to be clearly communicated. It is, of course, already partly implied by the fact that the edition is a scholarly work, meaning that it usually comes with an editorial rationale or an introductory note to the text. Finally, most digital editions include facsimiles of the manuscripts. The presence of the ‘original’ artefact next to transcriptions

---

24 A similar point came up recently in the Humanist Discussion Group (30.51).

25 Elsewhere, I explored in more detail the role of objectivity and interpretation in editing (Bleeker 2015a). Over the past decade the textual editing community has gradually come to realise that the proverbial large digital silos of (textual) data are not necessarily preferable over a digital text that is clearly edited by scholars, and currently most scholars agree that editorial interpretation in some form is useful and even desirable.

26 A digital facsimile is evidently not the original artefact proper: the relationship between the original artefact and a digital copy is certainly more complex. However, here it is taken as the
and editorial annotations also nuances the role of the editor as solely a supplier of text. A substantial part of communicating the ‘scholarly’ aspect of digital editions may be achieved through the presentation of the text and facsimiles and, more broadly, the interface of the digital edition. This issue will be more elaborately discussed in chapter 3.2.2 (‘Visualization of Collation’).

**Formalization: Instinctive Decisions**

It is hard to grasp the critical yet often subconscious process of decision-making that underlies the creation of a digital edition. Whether transcribing the text, building an infrastructure, or designing an interface, editors and researchers make decisions based on their knowledge and expertise. They select text(s) on which the edition is based, they process the textual variants and decide upon the representation of these variants and versions, they aim for a specific type of edition and audience, and finally they contrive ways to reach that audience. While these aspects, as we learned, are variable and thus unstable, the act of modelling nevertheless requires that they be clearly defined. The main obstacle here is that in practice, this formalization does not happen at the outset of the project, but often in retrospect.

An evident and simple example is that the way a text is encoded in TEI/XML has an impact on how it can be displayed, organized, and searched. Even when the editor follows a predefined set of encoding guidelines, she still makes tiny and subconscious decisions that could later influence the modelling process. For instance, semantically encoding a text could require researchers to explicate ambiguous notions that may be only understandable within a specific context, or require multiple tags. Another example, more extensively discussed in Chapter 2, is that its TEI/XML encoding influences how a file is processed. If the file is processed by a so-called ‘XML tree traverser’, for instance, the traversal program needs to know which actions need to be taken for each text character or XML-element it encounters. Even the order in which attributes are placed can be significant, so `<del type="instant" rend="pencil">` might be treated differently by the program than `<del rend="pencil" type="instant">`. Most editors are not aware of these subtleties when they are encoding and ordering their attributes. In the words of Ciula, modelling

simple fact that a digital facsimile of a document represents the primary material, thus taking the concept of ‘text’ out of the abstract realm and – if the reader so desires – making it possible to ‘check’ the work of the editor.
makes explicit what is ‘otherwise presupposed – if not buried in practices’ (2012, 100). Through modelling, editors begin to contemplate what they might not have questioned otherwise: their assumptions, their practice, and the existing standards of the field.

1.2.3. Challenges

Standards versus Flexibility

Textual scholars and editors are typically averse to the kind of standardization that is required to build formal models. For one, the subject matter (a literary text) is generally thought to be more complex than a set of general rules can convey, and the establishment of generic steps is also hindered by the wide variety of perspectives from which the subject matter can be studied. The plurality and idiosyncrasy of text editions is reflected in the TEI Guidelines that are renowned – or notorious – for their flexibility and customizability. Naturally, there is much to be said in favour of a customized text encoding that captures as much as possible the characteristics of each individual artefact. Yet an increasing number of voices plead for more standardization in text editing (e.g., Buzetti 2002; Andrews 2013; Robinson 2013a; Pierazzo 2015). As they facilitate the modelling process, standardization and formalization allow for the creation of interoperable software and data exchange. Although principles of interoperability and data exchange also underlie the TEI Guidelines, the design of the TEI is always in response to the needs of the editing community. As pointed out by Peter Robinson, a chapter of the TEI Guidelines may be based on the standards of a relatively small sub-discipline, whose editorial orientation subsequently affects the encoding guidelines proper (2013, online).

It seems that this community is currently suspended in a limbo, balancing between the desire to customize the TEI Guidelines to harmonize perfectly every edition project, and the need for interoperability, computational text analysis, and tool development. Editors need a certain freedom of interpretation and customizability; computers need standards and formal models. Andrews (2013) believes that the main problem is the mentality of editors, yet the choice between customization and standardization also depends on the goal of the edition project. The more flexible the take on the guidelines, the easier it is to convey the idiosyncrasies of the text in question; the more standardized the approach, the easier it is to experiment and develop applicable
software. The risk here is that editors stick to the digital imitation of traditional practices (cf. computer-assisted philology), instead of turning to methodological innovation (cf. digital philology).

The TEI Guidelines can function simultaneously as a *model for* an edition and a *model of* an edition. Although the TEI provides a formal semantic syntax that should enhance the cross-referencing and even reuse of editions, the way the TEI Guidelines practically are organized and employed does not provide a suitable method for the envisioned exchange. In all fairness, the TEI Consortium has always stressed that they provide mere guidelines and not conclusive rules. Nevertheless, they are generally considered as the ‘*de facto* standard for text encoding’ (Andrews 2013). According to Dino Buzzetti, the reason why TEI encoding and formal models make a problematic marriage is that the formal characteristics of the TEI Guidelines are often not compatible with the structural properties of the text (2002, 63), or at least what the editors wish to represent. Buzzetti concludes that ‘the formalization of the critical and interpretative procedures, and the specification of the operative model, remain essential for the production of digital editions’ (2002, 84). In this respect it is interesting to look at the approach of the *Roland HT*-edition. Its creator Vika Zafrin decided not to follow the TEI Guidelines, but to create her own set of XML tags. With regard to that decision, she writes:

> Separating the encoding process from the TEI has allowed my primary sources to dictate their own semantic structure, without the necessity to fit into a pre-existing framework developed without these materials in mind (2007, 66).

Since the TEI Consortium theoretically posits that ‘any concept a humanities researcher can imagine should be encodable using the modular tagsets it makes available’ (66), Zafrin plans to translate her encoded text into a TEI-compliant format. If this fails, she suggests that her tagset can be incorporated into the Guidelines for the encoding of texts similar to hers. This is certainly a defensible way to work around the imposed TEI format, and a fitting illustration of the flexible way the Guidelines can be used. It demonstrates the potential grassroots workings of the TEI community. At the same time, it is important to acknowledge that Zafrin’s approach requires a level of technical skills that is not available to everyone. Moreover, the TEI council may decide against the incorporation of her tagset into the Guidelines. In short, as long as it is customized for each individual editing project, the TEI cannot function as a standard for digital philology.
One of the purposes of following a standard is that it sets a bar and accordingly facilitates the evaluation of research output. The lack of standards for digital editions, indeed the lack of a proper definition, has led to some puzzlement with regard to evaluating them. It is generally acknowledged that scholarly work carried out in a context of changing digital technologies should be evaluated taking these changes into account, but the large variety of projects and different levels of collaborative work make it difficult to set any bar. What should be the outcome of digital editing practices? If we also accept prototypes and models besides text editions, how can we evaluate them? A valuable attempt to evaluate digital editing practices comes in the form of RIDE, the digital review journal of the German Institut für Dokumentologie und Editorik (IDE) that provides a checklist for reviewing digital editions. The RIDE checklist is a helpful instrument to create a published digital edition and assess its value, but if the output is a digital model rather than a complete scholarly edition, the evaluation process becomes more complex.

Models as Research Output: Assessment

If we know not to judge a book by its cover, can we judge a digital edition by its interface? Zafrin writes that in order to understand the results presented using new technologies, it is necessary ‘that the recipient knows something of these technologies, their benefits and limitations in a humanistic context’ (2007, 64). Zafrin provides short explanations of the technologies she used (i.e., XML and XSLT) in the methodology section of her dissertation. Dillen follows the same approach in his thesis (2015) where he briefly describes the concepts of XML encoding and other technical details of the digital edition he discusses. However, for all his laudable care, an increasing number of scholars suggest that it is not enough to explain digital technologies in a theoretical framework; instead textual scholars should be actively engaging with the underlying code. This suggests that, in addition to a chapter on XML, projects should also make available their XML files and code for others to assess and possible reuse. What does that mean for our traditional (print) research output?

28 For RIDE, see http://ride.id-e.de/ (last accessed June 6, 2016).
Kaltenbrunner confirms that ‘the creative uptake and adaptation of technological possibilities’ leads to more productive forms of digital scholarship (2015, 189). It is only by experimenting with the ‘algorithmic approaches’ of digital technologies, he writes, that scholars may gain the skills necessary to actually profit from them (2015, 183). This means, in other words, looking under the hood of computer programs and understanding both their functionalities and their limitations. Peter Stokes makes a similar argument in response to a discussion of the online Humanist Discussion Group, asking ‘if we don’t understand the box and what the results mean, then [how] can we move from those results to an understanding of the world that they purport to understand?’

In the same discussion thread, Tim Smithers outlines four conditions to be met for a successful integration of computational methods in humanities research. Taken together, Smithers’ conditions give a clear idea of how models can be assessed in digital editing. According to Smithers, editors using a specific tool would have to know and understand (1) all conditions in which the tool is intended to be used; (2) the theory of the structure and its application; (3) how to verify that the tool is designed in compliance with this theory, for all the intended conditions of use; (4) how to confirm that the tool has been well built in a way that fully satisfies the design. Note the repeated emphasis here on ‘know and understand’. Smithers further distinguishes between two types of users of tools. On the one hand, the 'skilled user' that meets all the conditions set above and has a thorough understanding of the principles underlying the tool; on the other hand the user that he describes as ‘feeders of automatic processing machines’. Formulated as such, it is clear which type has his preference. However, it is sometimes unclear what kind of user produces the publications and reports about models and editions.

**Models as Research Output: Acceptance**

Throughout this chapter, we have seen indications that another way of thinking is required: editors are urged to change their mentality, and be open to alternative methods and digital models as research output. However, there is

---

29 In Humanist Discussion Group 30.118, on June 24, 2016.
30 In Humanist Discussion Group 30.118, on June 24, 2016.
31 This also regards the reluctance of funding bodies and grant agencies to support this type of research. However, the workings of academic institutions and universities are infinitely more
also a kind of scepticism among some humanities scholars towards digital philology. As Doug Reside notes, ‘work that is largely based on technical methodologies may seem unfamiliar and perhaps threatening to traditional humanities scholars’ (Reside 2011, online). An explanation for this hostility is offered by Van Peursen, who attributes the scepticism to the fact that they continue to see a gap between quantitative and qualitative methods. Like McCarty, Van Peursen stresses that ‘the interaction between human researchers and computers, and between calculation and interpretation, is far more complex’ (2010, 4), and he postulates that this complex interaction can only become clear through examples from practice.

It is better, therefore, to consider in particular the voices coming from within the humanities community. Willard McCarty demonstrates an awareness of the situation when he concludes that the constructive consequences of modelling are often annihilated because ‘we lack a disciplined way to talk about [it]’ (2004, online). He notes that, even where modelling is successfully employed leading to ‘forthcoming’ results, researchers have not succeeded in making the intellectual case for this method. What the community needs, he states, is ‘an intellection of praxis’. Researchers need to fully grasp and understand modelling practices, in order to discuss them in constructive ways instead of dismissing them as ‘just a tool’ or ‘not a discipline’ (2004, online). In this light we can read the remarks of Vanhoutte, who advised young scholars to take a data set that is small or at least clearly delimited, experiment with it, document these experiments and report the findings in both Dutch and English, not ignoring or underestimating the technical audience. This way, textual scholars can enlarge their audience: if they can effectively convey the computational method used on a certain dataset, it should hardly matter whether this dataset is derived from a literary text in Dutch, English or Sanskrit. It may be an unfamiliar task to reach this double audience but, if successful, the boundaries of the disciplines are expanded or even blurred.

A first step in this process can be to avoid talking about computational methods in terms of witchcraft and wizardry. David Hoover, for instance, has complex and cannot be reduced to some remarks about a necessary shift of mindset and the conclusion that we should all learn to code. Although those are valid points, they may be taken the wrong way if made without taking into account the complexities of institutional infrastructure and established practice.

32 In private communication, at the DHBenelux 2015 conference that took place in Antwerp, Belgium.
been said to point out that participants of a digital humanities course would not be doing magical things with computers, but rather, using computers to assist their understanding of the magic that is literature. Wendell Piez, too, writes that

"a big part of the problem is magical thinking on all sides. People persist in thinking the machine is magical and treating it as a kind of fetish, intelligible only to initiates, a site of special power to be approached – or guarded – with enforced reverence and piety."

A second structural step is to make sure that the terminology employed is clear for all parties involved. The words ‘text’, ‘variant’ and ‘edition’ have multiple meanings and, not incidentally, most contemporary contributions in the field of digital humanities open with a definition of these terms (e.g., Eggert 2005; Caton 2013; Pierazzo 2015; Shillingsburg and Van Hulle 2015). The same goes for the terms data, information, and knowledge. With magical thinking and definitional ambiguity out of the way, we may be able to take serious steps with regard to interdisciplinary collaboration, the development of standards, and the acceptance of alternative (i.e., digital) research output.

Finally, and most directly related to the issue of acceptance of computational methods, there is the concept of authority. Adriaan Van Der Weel convincingly argues that our society is still dominated by the ‘Order of the Book’ (2010, 262), including the existing gatekeepers of the channels of publication. In this realm, we know which (scholarly) publications to trust. Digital models and computational methods are more difficult to verify and accredit through the traditional channels of humanities publication. As Reside notes, a ‘scholar-programmer in a traditional humanities department may find it challenging to communicate the value of her work to her colleagues’ (Reside 2011, online). When it comes to computational models or software programs, there are, as of yet, no guidelines or checklists, so hiring and tenure committees are at loss as to how to evaluate such output. So how can the digital humanities community convince the relevant stakeholders that models and programs and

---

33 James O’Sullivan, in the Humanist Discussion Group 29.449.
34 In the Humanist Discussion Group 29.452.
35 Van Peursen points out that even in the Order of the Book there are numerous alternative ways of publishing outside the control of the established gatekeepers (2010, 22-3). An example is the increasing number of scholars using a blog to publish (intermediate) research, and sharing their slides and conference talks online. Of course, this works better for prominent researchers whose name suffices to attract readers and whose track record ensures that even their blog posts are read with attention.
digital editions are scholarship and need to be valued accordingly? Fishwick states that in the computer science ‘the act of writing software (sans argument) would not be considered scholarship’ either, but he goes on to argue that ‘significant software implementations are considered highly significant’, and even expected in a computer science publication.\textsuperscript{36} Could a similar format, where the importance of the software (implementation) is framed in a clear argumentation, work for digital philology? Would it be accepted to include lines of code in the publications – or to make that code available for consultation online – and if so what is the best way to explain this code? Fishwick writes that we can read a machine ‘by examining its design, which generally means “reading” and understanding diagrams of increasing complexity’.\textsuperscript{37} At the same time, software can only be understood in action, by seeing it work. Otherwise it remains a rather abstract discussion, diagrams notwithstanding. It is interesting to consider in this respect the ongoing discussion about the acceptance of alternative PhD dissertations in the humanities that include software or other types of digital publications.\textsuperscript{38}

The question remains, can we initiate a balance between the long-established values of textual scholarship, and the required explicitness of computational methods, in order to productively automate the mental tasks of scholars? Under some conditions, the preliminary answer to that question would of course be positive. Based on the discussion above, these conditions may be grouped under four headings: experimentation, collaboration, reuse and openness. The next section discusses one approach towards digital philology that incorporates these conditions. It is tentatively called ‘the bricks approach’ and has been suggested in different formats by a number of scholars.

\textsuperscript{36} In the discussion Humanist 29.425, on 27 October 2015.
\textsuperscript{37} In the discussion Humanist 29.425, on 27 October 2015.
\textsuperscript{38} Examples are the work of Vika Zafrin (2007), or the dissertation ‘How can you love a work if you don’t know it?’ Critical Code and Design Towards Participatory Digital Editions by Amanda Visconti (2015).
1.2.4. Building Bricks

Advantages

In chapter 1.2.1, digital philology is defined as an experimental type of editing that uses computational methods like modelling for textual research. Its main focus is not to publish a digital edition per se, but to examine text by experimenting with new techniques. The section concluded that computer-assisted philology and digital philology can complement each other; together creating a framework of digital editing that is both productive and oriented towards innovation. Several scholars argue in favour of this format, although they have different viewpoints about its realization. Where Andrews is in favour of training textual scholars and editors to acquire programming skills and carry out their own text processing, Pierazzo considers this implementation of digital philology valuable but radical. She adds that it may be unfeasible considering the high level of computational skills it requires (2015, 116). She proposes to solve this problem by stopping to expect editors to control the complete digital workflow for their editions, and to divide that workflow into specific smaller subtasks or components instead. This is – tentatively – called the 'bricks approach', where 'bricks' refer to small tools that can be configured, combined and customized to fit into a specific editorial work (2015, 116). We see similarities to Hans Zeller’s Baukasten model, which deconstructs the scholarly edition into different modular components ('Bausteine') with their own concepts and typologies (1985, 321).

Pierazzo emphasizes that this approach is in fact realistic, even if we consider the technical and institutional limitations described in this chapter, as long as it is counter-balanced by the production of digital editions as envisioned by computer-assisted philology. As an illustration she offers the Interedition project, which ended a couple of years ago (2012) but continues to serve as an example and inspiration. In short, Interedition aimed to promote the interoperability and sustainability of both methodology and tools for digital editing. The project concentrated on facilitating contact between researchers and developers and communication between scholars.  

39 This translated into the development of open source software components (i.e., tools) of which

---

StemmaWeb and the collation program CollateX are the best-known examples. The reason why Interedition remains one of the most representative implementations of the principles of digital philology is that it offers a practical response to many of the challenges outlined above: from the reluctance of funding bodies to support innovative research, to the need for customization and flexibility, the acceptance of modular components as valid research output, and finally the collaboration between humanities and computer science.

The question is, with an editorial practice so divided and idiosyncratic, can textual scholars define general tools that suit the needs of the broader community of scholarly editors? Pierazzo thinks it is ‘relatively easier’ than developing a full-blown digital infrastructure, and she proposes ‘to highlight “microtasks” that are undertaken in most editorial enterprises and build tools that support them’ (2015, 117). It remains up to the editor/user to select useful tools and customize them to fit the need of the edition. Accordingly, the bricks approach also leans heavily upon the idea of technically savvy editors, or at least assumes that those working on the edition are able to customize and implement tools into a digital infrastructure. In the long run this idea can be realised through training of young scholars and early-career researchers. In the short run, the keyword is evidently ‘collaboration’. Chapter 2.3, chapter 3.3, and chapter 4.3 exemplify different forms of such collaborative scholarship. In the same vein, Ciula stresses the importance of communication and mediation:

One has to know how much one can about the human and mechanic principles of technology but mainly to be able to talk with, interact meaningfully with and understand those who master them; we don’t all make chairs. Some imagine them and draw them. It’s not a hierarchy, it’s a design team.  

The practical question is how much we need to know of each other’s fields of expertise in order to constructively work together. Ciula’s comments imply that humanities scholars need to know ‘enough’ of computational methods to effectively collaborate or to critically assess digital products, but are not required to fully develop their coding skills. We are reminded of the comforting words humanities students often hear when discussing future job prospects: they might not be able to ever use their knowledge of 12th century paintings, but their training in critical thinking will be universally appreciated. Accordingly, Smithers does not expect every humanities researcher to have such technical

---

40 In the Humanist Discussion Group 29.537 (October 30, 2015).
skills or to fully understand the tools being used, but he stresses that at least one person in the collaboration must, in fact, have both these skills and knowledge.  

Robinson takes the discussion of the collaboration between textual scholars and IT one step further in his rather controversial blog post ‘Why Digital Humanists Should Get Out of Textual Scholarship’ (2013, online). First, Robinson distinguishes between textual scholars and digital humanists. Although his definition of the latter is not entirely clear, they can be conceived as specialists working at a digital humanities department or Digital Humanities center. They may have worked on digital editions, but are not trained editors. Where Arianna Ciula envisions an editorial team, Robinson concludes that the combination of textual scholars and software engineers can only be productive if they work directly together, that is, without the intervention of digital humanities specialists. Most probably, he envisions a creation process where textual scholars are actively involved in every step and are challenged to explicitly formulate their needs and wishes. This process is too valuable to be delegated to external digital humanities specialists.

Dismissing the traditional workflow of computer-assisted philology (‘one scholar, one project, one digital humanist’ or ‘1S/1P/1DH’), Robinson also suggests a variant of the bricks approach where digital editions are made by a large and varied group of scholars, making use of online, open source tools. Editors would work with the tools directly and the editions would furthermore be published under a creative commons license, ensuring that the data is not ‘locked away’ but available for future use by others. He foresees ‘a burgeoning community of developers, hackers/scholars, interface creators; digital explorers of every kind’ (2013, online). A similar point of view can be found within the concept of ‘digital creativity’ given by Van Peursen et al., who expect the digital medium to not only provide new ways to store data and to represent knowledge, but also to encourage collaborations and to ‘support new creative processes’ (2010, 4). Robinson’s digital editions also echo the products of virtual knowledge described by Sally Wyatt in chapter 1.2, that is, a type of knowledge that inspires creative reuse and sustains, improves or develops tools and implementations.

Standardization and formal models are paramount for the bricks approach, which is based upon the concepts of interoperability, data exchange.

---

41 In the Humanist Discussion Group 30.118 (June 24, 2015).
and reuse. As mentioned above, this requires a departure from the flexible interpretation of the TEI Guidelines, and from the conviction that each edition has to be built from scratch with specifically developed tools. Instead, editors need to think about the edition in more abstract ways. They need to give more consideration to the edition’s functional and structural needs if they want them to correspond with their orientation towards the text, their research objectives, and their (envisioned) audience. This corresponds with what Malte Rehbein calls ‘the transition from classical to digital thinking’ (Rehbein, qtd. in Pierazzo 2015, 105) and comes down to the iterative technique of modelling and the formalization of existing tendencies discussed above. What the bricks approach subsequently asks of scholars is to focus on the separate components of an edition rather than on the complete, finished product. Furthermore, it requires the editing community to reach a level of consensus about the constituents of a digital edition. A suggestion comes from Andrews, who defines a number of microtasks of digital editing: transcription, collation, analysis, edition, and publication (2013). If we take a look at her list, it becomes clear that they are at least in name not different from the components of traditional print editions. Indeed, Van Peursen et al. acknowledge that, computationally imitating or mimicking established practices constitutes a first, initial step towards true innovation (2010, 14-15). Modelling, then, is essentially a new way of dealing with existing problems (McCarty 2001, 7). In (experimenting with) the realization and implementation of the modular components, true innovation can take place.

Theoretically, the bricks approach would also facilitate building suitable digital technologies and thus address the condition of reuse. Every now and then, the dissatisfaction with the TEI Guidelines and, on a broader level, the use of XML for transcriptions of text, culminates in frantic discussion about the best tool for the job. Until now, however, there has been no suitable alternative for XML. As computer scientist Fabio Vitali demonstrates, we have a variety of technologies to represent text (e.g., RDF, SQL, HTML5, XML) and although the choice often depends on the research objective, XML is by far the most extensive and elaborate format, despite its limitations. In a similar vein, James Cummings argues that a discussion about formats and the ‘right’ tool is futile. In capital letters that hint at his frustration about such disputes, he writes that it is rather ‘about granularity of information and appropriate technologies for

42 In a talk during the workshop ‘Editing Beyond XML’, DiXiT 2 convention (Cologne 2016).
appropriate uses’ (Cummings 2015, online). Breaking apart the digital edition into smaller components makes it easier to decide which digital technologies are most suitable to use.

In conclusion, it is clear that the bricks approach could lead to a more effective division of labour (cf. the condition of collaboration). Editors that do not have the technological skills to actually create ‘bricks’ of software can think about it on an abstract and conceptual level. They can go as far as to describe in detail how the tool or component (would) function and how it can be implemented. Close contact with the engineer who develops the piece of software and, by extension, with the community of external scholars and engineers who might use it, ensures a productive feedback cycle where underlying assumptions are continuously questioned. This also implies a step away from the conventional ‘maker/consumer’ (or ‘developer/user’) division: editors are not merely clients with a checklist of needs; software engineers are not merely uninterested executers of that list. Ideally, the bricks approach engages all parties involved and combines the strengths of both disciplines: the interpretive, critical judgment of the humanities with the formal and explicit logic of computer science.

**Obstacles**

Although the studies presented thus far provide convincing arguments in favour of the bricks approach, there are a number of obstacles that prevent the more this method’s more widespread adoption. These obstacles are similar to the challenges for digital philology outlined above. A first obstacle is the aforementioned conservative attitude of funding bodies and supporting institutions, as well as individual scholars. One of the consequences is a current deficiency of open source digital tools with which editors can experiment (cf. Pierazzo 2015, 205). A second challenge is the evaluation and assessment of computational methods described above. In this respect, Usher and Fiormonte note that ‘as applications move towards the interpretative, the challenge to both systems and critics becomes even greater’ (Usher and Fiormonte, 2001, x). It is difficult to assess the effectiveness of tools for the general community if they are designed for a highly specific purpose, e.g., from a certain editorial orientation or with a particular text in mind. Conversely, Boot and Joris van Zundert state that the publication of tools on the web may actually ensure their efficiency and reliability (2011, 151-52). Following their argumentation, we can say that the
open source aspect of the bricks approach provides a system of extensive peer-review, complemented with (variations on) online resources currently used in software development, such as GitHub and Stack Overflow.\(^{43}\) A third obstacle regards the textbook multidisciplinary collaboration sketched in the previous paragraphs. This is difficult to realise in practice, mainly due to the (lack of) technological knowledge and skills of textual scholars, and the environments in which such collaboration may take place. Consequently, the technical training of humanities scholars constitutes one of the most debated issues in the editing community and the humanities at large.

The current training programs of scholarly editors are largely oriented towards XML and related ‘X-technologies (e.g., XSLT, XQUERY, XPath) that are useful for the transcription and (re-)presentation of texts. The community is divided about whether or not this training is already too technical, whether the editor should rather focus on the editing of text (cf. Pierazzo 2015, 113). The general critique on the concept of digital philology and the bricks approach is that it is unfeasible or unrealistic to expect editors to command a variety of technical skills on top of their editorial expertise, or to assume that they receive infrastructural or financial support. For instance, when Robinson insists that digital editors must fully understand each step in the creation process (Robinson 2013b), it could also be read as a support for the claim that all editors should learn how to code. He argues against the involvement of digital humanities specialists in editing projects, and implies the necessity of giving textual scholars full control over the digital workflow, but if we remember that he himself has been a digital humanities specialist for over twenty years, his argument rather loses weight. Robinson’s standpoint is understandable for someone who can do most of the coding himself, but such a long period of training is not available to everyone. The same goes for the Interedition project, of which most participants were either self-taught programmers or had a background in computer science or software engineering. The tools CollateX and Stemmwaweb are currently used by a number of different editing projects, but their original developers – Haentjens Dekker and Tara Andrews respectively – are in practice also the only ones responsible for maintenance, sustainability, and (user) support. Sufficient and detailed documentation is therefore only a

\(^{43}\) GitHub is an online repository hosting service used mainly by (software) developers to collectively build projects and work together on open source technologies. See http://github.com. Stack Overflow is an active online forum that currently hosts almost 5 million programmers; see http://stackoverflow.com.
first important feature if a brick/tool is to be used by a large and diverse audience – followed closely by institutional support (time and resources) and professional recognition.

In discussing the question of what textual scholars and editors need to know, Galey writes that ‘any answer depends upon complex relationships between labour, epistemology, and technology, which extend beyond any primarily technical discussion’ (Galey 2010, 83). English professor and digital humanities specialist Merisa Parham concurs; saying that the most important skill is to be able to reflect on ‘media and medium, software and hardware, and where they stand in relation to the production of meaning and also how the technologies themselves produce meaning on the level of methodology and instrumentality’ (Dinsman 2016). Similarly, Robinson writes that textual scholars have to ‘absorb’ the effects of digital technologies while remaining completely aware of the consequences for ‘what we do, how we do it, and who we are’ (2013, online). A more technical point of view is taken by Doug Reside who envisions an ‘analogous hybrid’ type of editor: a scholar-programmer with the mindset and credentials of a scholar and a practical skillset of a computer programmer (Reside 2011, online). This scholar-programmer should have the ability to ‘think algorithmically’, that is, in pseudo-code, about humanities subjects. Reside emphasizes that he is not describing a computer scientist, but rather a ‘scholar for whom algorithms are an important mode of inquiry and communication for work in his or her chosen field’ (Reside 2011, online). The scholar-programmer has a way of thinking that ideally allows for a better understanding and assessing of the digital models behind digital editions. Such creatures are currently quite rare; they need to be formed through education, training, or knowledge transfer and supported within a suitable environment. Furthermore, it is essential to realise there are more widespread systematic complications at play. For instance, humanities programs are often at loss when it comes to evaluating non-conventional scholarly output such as a tool. So even if we find consensus on the ideal ‘scholar-programmer’ and the training of (future) editors – which is unlikely – there remain a number of battles to be won.

In conclusion, there is currently no concrete or definitive answer that goes further than having the ability to reflect upon the digital transformations of text on an abstract and conceptual level; preferably combined with a basic understanding of code. The bricks approach intends to circumvent the challenges outlined above by reducing the requirements of technical literacy to
a reasonable level' (Pierazzo 2015, 117) and provides a sustainable environment by making the tools open source. The latter ensures that the tools are readily available for established scholars as well as young scholars from both inside and outside an institutional framework. When successful, this creates a climate where scholars can ‘experiment with the new medium until its limits and possibilities are uncovered and explored to the end’ (Pierazzo 2015, 204). If there is an end, that is.

### 1.2.5. Reflection

In the sections above, I aimed to show how using computers as research instruments entails a fundamentally different approach to humanities research and scholarly editing, since the computer requires formal models to function. Traditionally, humanities methods leave room for subjectivity, that is, for interpretation and scholarly judgment, whereas in the computational methods the reasoning behind each decision needs to be made explicit. The positive consequence of using computational methods is that humanities scholars are compelled to confront and reflect on their own critical assumptions and presuppositions. Building models is an iterative learning process in which, ideally, a hypothesis is constantly tested and refined until proven. Moreover, once the editing community agrees about unambiguous definitions of (the components of) a scholarly edition, the development of interoperable software may take off.

The downsides are, first, that some concepts of humanities research are difficult to translate into exact wordings or may even benefit from a deliberate ambiguity. The status quo is neatly summarized by Van Peursen, who writes that ‘the reflexive concepts that are central to interpretation – individualism, subjectivity – are affected by the anonymised, normative assumptions implied by formal categorisations of text as digital data’ (2010, 13). Editors and textual scholars often waver between their natural inclination to document every idiosyncratic detail of the text, and contributing to the establishment of formal standards needed for tool development and data exchange. Secondly, it is difficult to assess the products generated by computational methods (e.g., models, software) in a structured and consistent manner. A final point of discussion is the technical skillset of the editors. It is argued that in order to use computational methods to their full advantage, editors need a thorough grasp of the workings of tools and computer-programs. This gives rise to a number of
confronting questions about the training of editors, collaboration with IT, and a supportive framework. It also points to the changes that happen – or need to happen – within academic institutions and the academic community at large regarding the evaluation and recognition of tools as scholarly products. The work carried out by a number of PhD candidates is in that respect promising, but we still have a long way to go.44

A proposed solution is described as the ‘bricks approach’, which combines the experimental perspective of digital philology with a dividing up of the editorial workflow in small, modular components (i.e., microtasks or ‘bricks’). By focusing on the development of computational tools to support and advance those microtasks within an open source environment, the bricks approach may provide a workable solution to the most pressing issues. A first definition of the microtasks of editing is proposed by Andrews (2013) and confirmed by Pierazzo (2015, 112). Although tasks like ‘transcription’ and ‘analysis’ suggest a conventional or analogue method, they are understood within the context of digital philology, meaning that the use of the computer makes a significant difference. It is especially interesting to consider these tasks in the light of editing from the genetic orientation: revised manuscripts present typical challenges for transcription, automated collation, as well as representation. Genetic editors have become familiar with questions like ‘how do we encode successive layers of revision while respecting the temporal aspect they represent?’ and ‘how do we explain our analysis of the textual genesis to readers?’, but so far no clear-cut solutions exist. In the following section, I will explore these tasks further and discuss how they may be implemented for genetic editing.

44 I already mentioned the work of Zafrin and Visconti; other examples are the experiments with automated collation algorithms by Elisa Nury (King’s College London [forthcoming]) and the digital visualization developed by Dillen (University of Antwerp, 2015).
1.3. Digital Genetic Editing

‘Manuscript editing, that is, the editing of manuscripts from private transmission such as drafts, diaries or letters, belongs exclusively in the digital medium, as it can only there be exercised comprehensively.’

(Gabler 2010, 50)

Introduction and Definitions

In the previous section it became clear that, faced with experimentation, innovation, and multidisciplinary collaboration, a consistent terminology and mutual agreement on basic concepts are paramount. We also saw that in the fields of textual scholarship and scholarly editing this consensus on terminology is still lacking. There are several synonyms for the concept of a digital scholarly edition, but as Vanhoutte notes, ‘the use of “edition” in these labels presuppose a conventional understanding of that word, and is based on the implicit assumption that there is a general agreement on what an edition is’ (2004, 1). Despite a number of elaborate definitions of a digital edition – and many more attempts to define them – we do no have a fixed definition nor ontology of digital editing.45 Pierazzo, whose Digital Scholarly Editing constitutes an attempt to define digital scholarly editions, proposes in the end the following description: ‘an interpretative representation of historical documents that encompass a combination of primary sources surrogates, edited text(s) and tools to exploit them’ (2015, 200). While recognizing that the terminological issues have not yet been solved, this study will use Pierazzo’s definition as it is sufficiently broad but does not oversimplify what is at stake.

A digital genetic edition is here understood as a digital publication that consists of multiple genetic dossiers of a creative work or works. It contains one or several tools that enable further exploration of the content and thus further study of the genesis of the work. Accordingly, the digital edition proper functions not only as a product of genetic editing, but also as a research instrument for genetic criticism. This corresponds fittingly with the aims of genetic criticism: it is a type of research that concentrates on the process, not on the final product.

45 That the field of digital editing continues to develop is also confirmed by the authors of the Introduction to Advances in Digital Scholarly Editing, a collection of papers presented at the three DiXiT conferences (see Boot et al. [forthcoming]; the introduction is written by Boot, Dillen, Franz Fischer, and Van Hulle).
The previous section concluded with a number of editorial tasks that change significantly if they are approached from a computational perspective, when the computer is used not as an assistant that facilitates conventional methods but as a proper research instrument. Consequently, a clear and consensual description of these tasks may function as a basis for the development of tools or ‘bricks’ that can be implemented into a digital edition. In her rationale for digital philology (2013), Andrews outlines five editorial tasks for which she embraces the use of the digital medium. This present section goes on to evaluate these tasks and juxtaposes them with the components of a genetic edition as described by Paolo D’Iorio (2010) as well as the five levels of genesis as distinguished by Van Hulle (forthcoming). As we shall see with the digital editing of Sheherazade, the three theories provide a useful perspective to explore what components of a digital edition may in fact be used as research tools for digital genetic editing.

1.3.1. Five Tasks of Digital Philology

Tara Andrews argues that taking ‘advantage of the complementary strengths of man and machine’ may lead to superior results (2013). To better understand how these strengths can be combined, Andrews classifies five successive steps in the editorial workflow that are transformed when computing is introduced into the equation. Accordingly, the tasks she distinguishes are part of a conventional editorial workflow: they differ only in the applied methodology.

The first step of many editorial workflows – although incidentally not the first step of genetic editing – is transcription. As chapter 1.1 and chapter 1.2 have already argued, transcription involves a significant amount of editorial interpretation. Andrews recognizes this, stressing that transcribing is a useful act, ‘if a painstaking one’. Indeed it requires a close examination of the text and – in some cases – the document, giving the editor a detailed and indispensable knowledge of the material. Consequently, she suggests few alterations to the existing approach here, mentioning only future possibilities of OCR for manuscripts and ‘more user-friendly transcription tools’ but not going into detail about how these could work. Collation is classified as the second step and here the use of the computer can in fact make a difference. Andrews mentions three tools – Juxta Commons, CollateX, and nCritic – that can assist the editor with the collation. The editor feeds them the transcriptions ‘secure in the knowledge’ that even highly detailed diplomatic transcriptions will be handled
in a relatively short time (2013). She concludes that the editor only needs ‘very few hours’ of verifying the collation output instead of the serious amount of time they spent on manual collation. Although the tools indeed facilitate the collation process, especially time-wise, the reality of automated collation is actually more nuanced.

In Chapter 2 (‘Transcription and Collation’) it will become clear that the idea of accurate collation is largely dependent on implicit assumptions and may be subject to change. In other words, editors may be even more secure in the knowledge that their collation is carried out correctly if they are able to influence the process — or at least if they are aware of it. Andrews already hints at this when she writes that the tools should (1) minimize or at least postpone the normalization of the input data and (2) not require of the editor to assess every detail.

The third step is formed by analysis, an activity that in Andrews’ workflow is carried out after the transcription and collation. Here, analysis means the application of computational methods to establish authorship attribution, or to perform phylogenetic or stemmatic analysis. Andrews considers computational methods especially relevant for this type of activity. She also gives an assisting role to the computer in the fourth step: the establishment of an edited text and the ensuing (critical) edition. Here, the use of the computer ensures consistency and, again, a speedy way of transforming the XML-transcriptions and collation-output into a publishable format. She points out that, although it should be a ‘relatively straightforward task’, there is as of yet no public tool available that can assist the editor with this step. For her digital edition, Andrews wrote a script herself that remains available online. Keeping in mind the discussion in chapter 1.2.3 about the acceptance and evaluation of computational methods, it is interesting to note that she has also published an article on the implementation of that script.46

The fifth and last step, publication of the edition, is largely left open. Bound by the institutional guidelines at the time, Andrews published her edition in a PDF format. She recognizes that beyond some essential features there is no clear idea about the look, the content, and the organization of a digital edition, which ensures that most digital editions linger in the Order of the Book and are published in a corresponding format. However, she points out

---

46 In her article, Andrews provides a link to the specific publication on the script she wrote, but unfortunately the link is not functional anymore.
that the driving force behind change comes not only from the community of
textual scholars and editors, nor only from institutions and grant suppliers:
since the production is influenced by demand, the audience of users too can
influence the future shape of digital editions.

Andrews concludes with a discussion of how the use of computational
methods can influence stemmatic analysis in particular. To name but one
example: the capacity of the digital medium to store and process large amounts
of data ensures that the stemmatologists do not have to limit themselves to
‘relevant’ variants and instead can take all available textual variants, which in
turn could result in unexpected discoveries (2013). Consequently, Andrews
writes, if editors adopt computational methods and embrace the ‘power of raw
computation’, thus surpassing most practical limitations, they may reinvigorate
the traditional methods to the extent that they will tread on new ground. As
mentioned above, this can be seen as a radicalization of digital philology (see
chapter 1.2; Pierazzo 2016, 116), but at the same time it is the kind of mindset
needed to grasp the reconceptualization of the editorial workflow. Finally,
Andrews clearly makes the case for a focus on production methods, although,
again, this does not exclude a publication mode.

1.3.2. Five Components of a Genetic Edition

In ‘Qu’est-ce qu’une édition génétique numérique?’ (2010), D’Iorio strongly
argues for a well-designed architecture of a digital edition. In the case of a digital
genetic edition, which can consist of several genetic dossiers, the organization of
the edition is crucial to ensure that its users have all the instruments to further
explore or examine the content of the dossier(s).47 These instruments can be
divided into five components that every digital genetic edition is supposed to
have:

1. One or several genetic dossiers
2. Digital facsimiles systematically organized in a catalogue
3. Transcriptions of—ideally—the complete genetic dossier
4. One or several systems of classification
5. Representation of genetic processes

47 D’Iorio clearly uses the term ‘genetic dossier’ in the sense of an ‘avant-texte’. Following the
definition as stated in chapter 1.1.2, we can rephrase and say that a genetic edition can contain
one genetic dossier organized in several ‘avant-texte’s.
The following paragraphs discuss each component in further detail and specifically from the perspective of using computational methods. Not every component is perfectly suitable for computational methods, writes D'Iorio (2010, 50n4). He believes that the computer can play an important role for digital genetic editing, but — contrary to Andrews’ beliefs — mainly in the publication phase. He recognizes the merit of the digital medium for storage and manipulation purposes (in line with the idea of the computer as research assistant), but he does not mention the possible application of computational methods to the first component (establishing a genetic dossier) or the third component (transcribing). It may be due to the relatively short length of his contribution, but D'Iorio does not allude to the possible advantages of modelling a manuscript by means of TEI/XML. The transcription serves in the first place to make sometimes illegible manuscripts readable, ensuring a ‘more comfortable’ access to the text (2010, 51), and could therefore theoretically be withheld from the genetic edition (2010, 50). His main issue with a linear transcription seems to be just that: its linearity cannot reflect the complexity of the manuscript page. As mentioned in chapter 1.1, this is a recurring topic in the field of genetic criticism. The general consensus is to acknowledge its limitations and embrace its advantages. Here, the value of markup and edited texts is mentioned mainly in terms of accuracy, speed and the possibility to generate a critical text. The same goes for the added value of collation tools, of which D'Iorio especially appreciates that they can handle the various texts that make up the genetic dossier and produce a full apparatus (51).

A draft manuscript often consists of successive layers of writing. Both D'Iorio and Cesare Segre talk about textual layers or ‘des couches successives’ (1995, 37) Similarly, Gabler stresses that these layers are an editorial construct. Gabler defines textual levels as the variance that occurs across document borders, i.e., the development of the text from witness to witness. Textual levels, he asserts, are not an editorial interpretation: they constitute themselves in the transitioning from document witness to document witness. It follows that this type of textual variation can only come to light through collation of the witnesses (Gabler and Schaüble 2016). Throughout this thesis I follow the distinction as proposed by D'Iorio and Gabler. Textual layers are the layers of revision on a document page (intradocumentary variation) and established by the scholarly editor; textual levels occur across document borders and can be established through collation (interdocumentary variation). They pose an additional challenge for the handling of variants. At which point do a deletion
and an addition actually constitute a new version? D'Iorio points out that the collation of textual variants is complicated by the cancelled and added text within a witness, but he concludes that the constitution of a version is primarily a decision the editor takes during the genetic analysis and the encoding of the texts.

In the end, D'Iorio sees most room for computer-involvement in the fourth and the fifth component. The fourth component corresponds with what he calls ‘systems of classification’ (2010, 52-3). If we assume for a moment that the edition has a suitable architecture, the digital technology then allows its users to arrange and organize the contents in various ways, for instance chronological, according to document type (e.g., notes, letters, published texts, etc.), or genetic, where each textual segment is presented with its genetic development. This brings us to the fifth and final component: the representation of genetic paths or, in D'Iorio’s words, ‘parcours génétiques’ (2010, 53). He sees this as the most typical constituent of a digital genetic edition and a first point of entry into the study of the creative processes. It is implied that these parcours or ‘genetic pathways’ are based on careful genetic research and scholarly judgment, meaning that they contain a certain amount of editorial interpretation. The genetic pathways may be represented in various ways. D'Iorio mentions figures and diagrams; chapter 3.4 presents a more experimental attempt, namely a virtual museum application. D'Iorio acknowledges that encoding a text in TEI/XML allows for a more in-depth analysis regarding additions and deletions. However, he asserts, all these different ways of representing the genesis (be it through parcours, diagrams, or collation) would still only give an approximation of how a (segment of) a text came to be. Therefore, the merit of a more elaborate explanation provided in the form of a narrative by the editor, should not be underestimated. And finally, the edition needs to make room for its users to interpret the textual genesis themselves.

1.3.3. Five Aspects of Textual Genesis

The fundamental idea of using a digital genetic edition as a research instrument for genetic criticism and, consequently, to define the activity of digital editing as a way to perform manuscript research comes from Van Hulle. In a forthcoming article ‘Modelling a Digital Scholarly Edition for Genetic Criticism: A Digital Rapprochement’, he distinguishes five aspects – five appears to be the magic
number of textual variance and illustrates how each aspect can be included and studied in a digital edition. The aspects are as follows:

1. Exogenesis
2. Endogenesis
3. Epigenesis
4. Microgenesis
5. Macrogenesis

These can be grouped in two categories: on the one hand, ‘exogenesis’, ‘endogenesis’ and ‘epigenesis’; on the other hand ‘microgenesis’ and ‘macrogenesis. The distinction between these aspects is artificial and primarily made to facilitate analysis. In reality, the five aspects overlap and interact (see figure 1) so we could, for instance, study the exogenesis of a text both from a microgenetic and a macrogenetic point of view.

Furthermore, it is important to emphasize the subtle yet essential difference between Van Hulle’s aspects, D’Iorio’s five components of a digital genetic edition, and Andrews’ five stages of digital philology. The latter two propose respectively elements of a digital edition or of the workflow of digital editing – be it genetic (D’Iorio) or critical (Andrews) – and discuss the role of the digital medium for each component/stage. Van Hulle on the other hand distinguishes two levels of genesis, and explores subsequently how a digital edition may support the study of each of these levels. Hence, his classification of variance originates in a specific practice (i.e., editing modern manuscripts by Samuel Beckett) rather than an overarching theory. Moreover, the five aspects of variance are intended to cover the complete spectrum of textual variance, and are thus not identical to the successive layers of text that can be distinguished on a revised manuscript (which would fall under endogenesis). Nevertheless, there is a certain amount of overlap between the contributions of Andrews, D’Iorio and Van Hulle, and collectively they give an exhaustive view on digital genetic editing. By synthesizing the three theories, we can come to an comprehensive definition of the components in a digital genetic edition that provide opportunities to use computational methods to explore the genesis of a work.

Van Hulle adopts the approach of modelling as discussed in chapter 1.2. He illustrates the application of this approach with a simplified model of these five aspects of variance (figure 1), taking examples from the manuscript of
L’Innommable/The Unnamable. The model is also implemented in the digital architecture of the BDMP.  

The three levels of exogenesis, endogenesis, and epigenesis have already been defined in chapter 1.1. (‘Genetic Criticism’) but can be summarized as:

- the processing of a particular exogenetic source text;
- the revision history of one specific textual instance across endogenetic and/or epigenetic versions;
- the ‘réécritures’ or revisions within one single version’ (Van Hulle [forthcoming]).

They can be studied from a microgenetic and a macrogenetic perspective. The concept of macrogenesis is already implied in its name: it concerns the genesis of an entire literary work across all its versions. Van Hulle intends it to incorporate both a document-oriented and a text-oriented perspective: it includes for instance the materiality of a manuscript as well as the development of the narrative. He points out that digital collation tends to be “micro-focused” and indeed collation traditionally examines variance on the level of words or even punctuation. As noted earlier, however, collation could also be helpful in exploring how a text develops across documents (cf. Gabler’s textual levels). By presenting also a visualization of the macrogenesis, Van Hulle aims to help users ‘find their way into the labyrinth of manuscripts’, which they can then further examine on a more detailed level. Van Hulle’s proposed model offers a valuable method in that it aims to offer ways to examine the genesis of the text from multiple perspectives.

---

With regard to providing access to the genetic edition and facilitating such examination, Van Hulle proposes two strategies similar to D’Iorio. A first entry point is the indication of revision sites (cf. D’Iorio’s fifth component: the parcours or genetic pathways). Van Hulle, too, considers the representation of genetic paths a suitable way to enter a genetic edition. As mentioned above, D’Iorio considers the digital medium ideal to represent the creative invention through a visualization of the genetic pathways, but he concludes that the editor is invaluable for her explanation of variance in editorial annotation. Similarly, Van Hulle suggests the use of a ‘revision narratives’ as first proposed by John Bryant (2002) to highlight and explain the three aspects of exo-, endo- and epigenesis. He develops this idea by describing a dynamic visualization that displays either the order of the sentences as they appear on the page (document-oriented) or the order of the sentences as they appear in the narrative sequence of the text (text-oriented). This visualization reflects D’Iorio’s system of classification (i.e., his fourth component).

A second entry point into the text’s genesis is achieved by providing access to the texts and the facsimiles (cf. D’Iorio’s second component: digital facsimiles) and offering a collation tool (cf. the third component: transcriptions). The tool in question, CollateX, can assist users of a digital edition in discovering the interesting instances of variance. Chapter 2.2 (‘Transcription and Collation’) discusses in detail how this specific tool is applied to the BDMP. For now, it is interesting to note that in this case CollateX is used primarily as a tool for users of the digital edition: the program is incorporated into the BDMP, presented in a specifically designed Graphic User Interface (GUI), and the underlying code is not visible. Keeping in mind the two-sided medal of digital philology, (i.e., a focus either on experimenting or on creating and publishing an edition), the specific implementation of CollateX in the BDMP can be classified under the latter.

Accordingly, the main reason for the visualization is to present the user with an idea of how the text developed, and to demonstrate the creative process that produced a specific textual segment. It includes for instance also the sentences that did not make it into the published text, so-called ‘cul-de-sac’ sentences (Van Hulle 2008, 283; Van Hulle [forthcoming]). As such, the visualization remains an alternative ‘access point’: just like the collation tool, it

49 This feature is developed in collaboration with Dillen and Neyt, and more extensively discussed in chapter 3.2 (see also Dillen 2015, 249).
is an instrument to engage with the text's genesis as it is established by the editor. The collation tool and the visualization are first and foremost aimed at the user of the edition, not at a scholar who wishes to experiment with the material in other ways than provided. Still, being part of the BDMP, both instruments have a scholarly quality and there is little risk that editors would dismiss them as ‘not scholarly enough’. The fact that both are used within the environment of the BDMP also implies that they can be gathered under the publication-aspect of digital philology. Incidentally, this does not imply that the editors of the BDMP are not keen on conducting innovative experiments: on the contrary (see for instance Haentjens Dekker et al. 2014). It simply demonstrates that such experimentation does not take place within the environment of a digital edition, but rather in its periphery.

---

50 In her discussion of non-scholarly editions, such as those published by TouchPress, that attract a significantly larger audience, Pierazzo asks: ‘could a critical apparatus be made more engaging if offered in an interactive way in an environment that has some sort of ludic aura around it, or will editors think that it does not look scholarly enough?’ (2015, 155) Here, the fact that the collation tool and the visualization are integrated in a scholarly edition and use scholarly produced data, probably reduces that risk to almost zero. It may be, therefore, an interesting direction for further research. This is further explored in Chapter 3.
1.4. Concluding Remarks

‘There has thus been a new acknowledgement that textual criticism is not merely a dry, mechanical, tedious investigation of physical fact but, like all other intellectual activities, operates under various theoretical persuasions, which may change from time to time.’

(Greetham 1992, 342)

This chapter set out to define what specific tasks of genetic editing could be complemented by the use of computational methods as described in chapter 1.2. In particular, my objective was to find how these different methodologies might be used to the advantage of the prevailing research activities of scholarly editing. To this end I discussed three articles that identify, respectively, five steps in the editorial workflow (Andrews), five components of a digital genetic edition (D’Iorio), and five aspects of textual variance (Van Hulle). The results of this discussion confirm the thin line between computer-assisted philology and digital philology. Moreover, they follow the general trend of not making any assertions about what exactly a digital edition should look like, and propose instead a set of different modular components. Both Andrews and D’Iorio mention the possibility of generating a (critical) edition based on the XML transcriptions; D’Iorio asserting that although the main goal of genetic criticism is not to produce an edition, it would be a shame not to use the transcriptions if you already have them. Here, the digital medium is mostly used for reasons of convenience.

Furthermore, all three authors make suggestions as to how the digital medium may be employed to further advance textual research. They acknowledge the usefulness of (TEI) markup for collation and mention the use of a collation tool for the purpose. However, despite an emphasis on editors understanding the code of the tools they use, Andrews does not go into further detail about whether such tools contribute more than speed and accuracy. These are also the only two advantages identified by D’Iorio, who subsequently mentions the difficulty of distinguishing versions from variance with—often heavily revised—modern manuscripts. Van Hulle on the other hand does include a more detailed report of the implementation of a collation tool. He demonstrates how a (customized) version of CollateX can also display the deletions and additions that are so important for genetic criticism. As such, he simultaneously offers a possible solution to the problem of distinguishing versions and variance. Since it is not always clear where a revision campaign ends, especially if the revisions are made in the same ink, a clear visualization
that displays the various layers is important in order to maintain and respect the sometimes-subtle differences between them. Moreover, D’Iorio and Van Hulle recommend a supplementary editorial annotation (e.g., a note or a revision narrative) to explain the more compelling or complicated cases of textual variance.

Another important issue for digital genetic editions is also mentioned by both D’Iorio and Van Hulle and regards the indication of ‘genetic pathways’ through the text. These pathways, or parcours as D’Iorio calls them, form an effective way to introduce users to the content of the edition. The pathways are constructed by the editors, based on their knowledge of the material, and are primarily intended as illustrations of the writing process. It is therefore important that they are understood in the context of editorial interpretation. Furthermore, we can make a clear distinction between on the one hand a genetic pathway of a textual segment, say, a draft note to a galley proof, and on the other hand a ‘revision site’. The latter relates rather to a place in the text where the writer made one or several revisions and can be as detailed as word- or character level. Here, it is relevant to consider Van Hulle’s macrogenesis and microgenesis, and D’Iorio’s system of classifications. Both authors argue that, ideally, a digital genetic edition allows the various perspectives of the genesis of a text to coexist (e.g., chronological, material, genetic).

Returning to the question posed at the beginning of this chapter, it is now possible to identify the components of a genetic edition where the use of the computer affects both the methodology and the output. In three particular cases, the computational aspects of a digital edition may serve as research instruments in the sense of digital philology.

**Transcription and Collation**

The first component can be classified as ‘transcription and collation’. In genetic editing, the task of transcribing brings forward the challenge of distinguishing between versions and variance, between revision campaigns and genetic pathways. The TEI Guidelines provide extensive models to transcribe genetic manuscripts in great detail, and as a consequence, the TEI/XML transcriptions are often particularly complex. Collation is a highly useful method to examine endogenesis in particular and, by extension, microgenesis in general. At the same time, the fact that this task can now be carried out almost entirely automatically yields new ways to examine texts. A possible direction is for
instance the collation of texts with their XML-markup, in reaction to the importance that genetic editing attaches to the different layers of text.

**Visualization of Transcription and Collation**

The importance of visualization is one of the more significant findings to emerge from the literature. If we think about the importance of transcription, if we consider it as a fundamental act and as a critical step in analyzing the text’s genesis, the need becomes clear to have a useful visualization of a transcription. Such a visualization would not only offer an adequate and understandable representation of this analysis, but also constitute a valuable instrument in communicating the editor’s interpretation. Similarly, if editors use collation software to examine the different levels of genesis, it follows that the collation output is visualized accordingly. The second component therefore regards the visualization of the transcription and the collation output. Whereas the concept of visualization and the related issue of interface have often been discarded as an afterthought to digital scholarly editing, or even as a matter for web designers and not for editors, both D’Iorio and Van Hulle stress that a visual representation is invaluable for a better understanding of the creative writing process. This is relevant for the genetic pathways across different versions, but also the additions and deletions within a text version that are processed by the collation tool.

At the same time, to establish a scientifically sound visualization that is both informative and exhaustive can be quite challenging. As a result, the process of creating a helpful visualization (i.e., modelling) could be considered just as valuable a research instrument for the editor as the visualization proper (the model). A confirmation of this way of thinking is given by Fishwick when he writes that the ‘user interface [is] a human interactive front end to the visually less observational digital hardware’ (2014, 209). Even though user interfaces generally follow our Order of the Book and mimic the analogue world (also described as ‘skeuomorphism’; see e.g., Pierazzo 2015, 159), they remain highly useful, natural instruments for human beings to shape their thoughts and

---

51 Pierazzo notes that interface design is ‘a niche field in Digital Humanities’ and even ‘something alien to editorial concerns’ (2015, 158). Recently, things seems to have changed, with a number of seminal publications on the subject of visualization of digital research in cultural heritage (Rosselli del Turco 2011; Ruecker et al. 2011) and a DiXiT symposium around the topics of scholarly editions and interface design (‘Digital Scholarly Editions as Interfaces; 23-24 September 2016, Centre for Information Modelling, Graz).
inform their cognition. Following this perspective, visualization is much more than a final glossy layer over a digital infrastructure; it constitutes a visual means to support and model thought processes of both editor and end user.

**Classification: Organizing Documents and Versions**

The third component encompasses the different systems of classification and how they are integrated in a digital infrastructure. D'Iorio even considers it one of the most significant aspects of digital editing: it is ‘grâce à la souplesse du support’, he writes, that a digital genetic edition allows users to arrange the content according to their needs and interests (2010, 53). Although this topic is the least explicitly discussed in the contributions by Andrews, D'Iorio and Van Hulle, it is inevitably implied in their remarks about the coexisting systems of classification, about the various ways to access the genetic edition, and about the possible integration of tools like CollateX into an edition. In order to achieve this, a suitable infrastructure is just as important as the informative user interface discussed above. What is more, in designing and creating the infrastructure, the editor is again obliged to think and reconsider the ways in which the edition’s content is interconnected and related. The choices made here can influence the future use of the edition, but also reflect the ways in which the material is imagined. For example, the non-hierarchical and intertextual nature of hypertext often resonates with the notion that editors have of a (corpus of) texts. Conversely, the implementation of a hypertextual infrastructure influences how that corpus is perceived and examined. Where XML transcription and automatic collation deal with the modelling and examination of the text on a highly detailed, microgenetic level, the infrastructure also takes into account the macrogenetic level of the text with the systematic classifications and structuring of the genetic dossier.

The following chapters examine in what ways each of the three components distinguished above may contribute to the use of the digital edition as a research instrument for genetic criticism. Each component comprises one or several editorial tasks that, keeping in mind the concept behind the bricks approach, may serve as a focus for further tool development. In order for this approach to be successful, it is essential that editors have a clear understanding of every step in the process, also where it concerns the implementation of computational methods. They do not have to be fully developed programmers, but scholarly editors do require a critical awareness of the use of code and
computing (chapter 1.2). It will become clear that there are many overlaps between the topics of transcription and collation, visualization, and infrastructure. The distinction between them is artificial and chiefly maintained for reasons of clarity. It is by focusing on these individual aspects that we may grasp the more general picture of how the use of computer-based tools and methods can work to the advantage of genetic criticism.

Finally, in view of what is discussed, we may expand the definition of the digital genetic edition given above (see ‘Introduction and Definitions’). A digital genetic edition provides a means (in the form of both material and tools) to further examine the genesis of a work. The fact that these materials (e.g., transcriptions, facsimiles, edited texts) are ‘published’ with the launch of an edition is more a fortunate consequence than the main goal. As a result, a digital genetic edition can prolong the (after-)life of a work by enabling scholars to digitally examine the process of creation and invention in new and original ways. This is achieved by using the computer as a research instrument (contrary to research assistant) and by engaging in different working methods (as discussed in chapter 1.2.).

---

52 Evidently, the whole concept of ‘publication’ in the digital paradigm is in flux, since content can be continuously updated and interfaces redesigned.
Chapter 2 - Transcription and Collation

Writing systems, print technology, and now digital encoding: each licenses a set of markup conventions and procedures (algorithms) that facilitate the self-reflexive operations of human communicative action.

(Buzzetti and McGann 2006, 67)

2.1. Preamble

Introduction

‘Laborious’, ‘error-prone’, ‘painstaking’ and ‘time-consuming’: collation, a core activity of textual scholarship, is generally not described in positive terms. Collation, at its most basic level, means the comparison of two or more texts (literally: ‘placing side by side’). Over the centuries, textual scholars have collated for three reasons: (1) to track the transmission history of text; (2) to come as close as possible to the ‘original’ text; (2) to establish a critical or final text. Collation is a repetitive task that nevertheless requires a high amount of concentration and attention to detail. Not incidentally, collation is one of the jobs that quickly became automated when editors started working with digital technologies, resulting in a number of automated collation programs. These programs are able to carry out the job at high speed and very accurately, for a large number of witnesses. As a result, they take away the burden of the mechanical work of comparing texts and mapping variants. However, when the material to be collated consists of digital texts encoded in a variety of ways, the concept of ‘accuracy’ is ambiguous. Evidently, the collation output needs to be correct but, as this chapter will explore in detail, it is more difficult to establish what exactly constitutes a correct collation result.

Traditionally, collation has been considered an auxiliary task: a necessity to map textual variation and gain a better understanding of textual transmission. In that sense, it is an elementary activity of preparing the material to ensure good scholarship, not necessarily a part of the hermeneutical process of textual criticism. Hence, the collation output ‘just’ needed to be reliable and as accurate as possible. At the moment of writing, there are roughly two main schools of thought in textual scholarship. On the one hand, there are the scholars who are inclined to consider collation from this traditional perspective (i.e., texts are collated by a tool and the editor interprets the results). For them,
collation remains largely outside the hermeneutics of textual criticism. This is epitomized by the idea of the computer as a research assistant charged with the ‘tedious’ tasks of editing. On the other hand, a growing number of scholars prefer to engage directly with a collation algorithm and adjust it to their research interests. Here, the collation process is incorporated into the main workflow of creating an edition.

In both cases, however, the collation itself is still mainly performed on plain text without markup or, to be more precise, so-called ‘string-only input’. The meticulously (TEI/XML) encoded transcriptions of manuscript material are first transformed into strings of characters, i.e., stripped of their markup, before they can be collated.\(^{53}\) As we will see in chapter 2.2, some approaches make it possible to ‘pass on’ designated markup through the collation pipeline together with the character strings. Those markup elements can be visualized again in the output. Although it is technically possible to influence the alignment\(^{54}\) based on the preserved markup elements, this is seldom done in practice. In other words, the input and output of a collation program tends to be plain text.

This situation has implicit yet important consequences. First, the significant amount of scholarly knowledge that editors put into the average encoded text is ignored. Secondly, not many scholars are aware of the specific approach taken by their automatic ‘assistant’. That is, collation software often functions as a black box: users ignore that the text is subject to a series of edit operations, or how these edit operations can be influenced. As long as the collation output is correct, many users would not care about the insides of the black box. The problem is that the definition of ‘correct’ is no longer clear-cut. The simplification and transformation of the encoded texts cannot be underestimated: the text constantly changes as it is processed by a collation tool. Each operation involves a certain amount of editorial interpretation. For example, preparing the file for collation means that certain — but not all — elements of markup are taken away. So where text encoding by itself means adding markup (i.e., interpretation) to a transcription, the same goes for extracting markup: it is an interpretative act that creates a new text.

A considerable set of technical skills is needed to prepare (‘preprocess’) the material and inform the collation software about which markup can be

---

\(^{53}\) Incidentally, we should not forget that the plain text of a transcription is also an editorial construct, as Pierazzo (and others) repeatedly point out: ‘even when looking at a clean reading text, we must not forget its essentially interpretive nature’ (2016, 319).

\(^{54}\) The next section (‘Definitions: A Closer Look’) gives a detailed definition of alignment.
included. Not every scholarly editor has these skills. In the case of the collation software CollateX, for instance, a basic knowledge of JSON is necessary to create the CollateX-specific JSON input format where the selected markup can be added to the text. The input TEI is transformed into JSON, processed by CollateX, and the output is again transformed in the selected output format, say TEI/XML. Additionally, the output can be postprocessed in order to create better collation results. With each step, it becomes more difficult to keep track of the exact state of the text: layers of markup are added or removed, and the text as well as the markup is transformed several times. Manual (or actually, ‘eye’) collation might reveal such issues in collation results, but it becomes increasingly impractical as the number of witnesses climb or ever more heavily marked-up texts are processed. On a more theoretical level, the editor has to decide what markup to preserve in the first place. Would that be the carefully documented deletions and additions in the text of a modern manuscript? The line breaks in a poem? What about a shift in writing material or writing hand, or the editorial annotations? Each of these could have a significant value, depending on the kind of text, the material, and the type of edition.

As a consequence, it is difficult to write a generic script for the transformation process from TEI to JSON and thus more likely that each editing project requires a tailored transformation process to deal with the project-specific use of TEI elements. Which brings us back to the technical skillset of the editor. In short, automated collation is far from a neutral process. The question is whether editors and users are aware of this situation. It is easy to see that the scholarly editor plays a decisive role in the pre-processing of the encoded texts before collation, but the transformations that the text undergoes unnoticed while being collated are near to invisible.

The objectives of this chapter are twofold. The first is to explore the interpretative acts performed on the text in the collation process. How does the text transforms and to what extent can editors influence these transformations? This necessitates a distinction between the collation algorithms that perform operations on linear sequences of bits, and the preparatory process of transforming the documents into these linear sequences of bits. Where the former can be considered an objective process, the latter, this chapter argues, sure cannot. The second objective relates to the editorial awareness of the

---

55 CollateX is a software program for collating textual sources, with Ronald Haentjens Dekker as main developer. See for more information the project page of CollateX at the Huygens institute (https://www.huygens.knaw.nl/collate-x-an-automatic-text-alignment-tool/).
collation algorithms and how the algorithms may be influenced. This would result in a broader understanding of the text and, consequently, a fuller command of the editorial workflow. In the final part of this chapter, the case is made for a so-called ‘XML-aware’ collation approach that allows the editor to retain markup during the process of automated collation. This approach is of interest for a broader range of editorial orientations but, as I explore in chapter 2.3, especially valuable for the genetic orientation in which the layers of revision are distinguished with markup.

The chapter is organized as follows: chapter 2.2. (“Theories and Practices”) discusses the current theories regarding transcription and computer-supported collation. Where most collation software accepts TEI files as input, they operate on the level of the text (that is, they collate the strings of text characters and ignore the markup tags). However, it is often possible to make the software aware of (some of) the TEI elements and pass these through the collation pipeline. I analyse a number of practices that make use of this possibility. It is especially advantageous in the case of a genetically encoded file, since the markup elements also contain text characters (such as the <del> or <add> elements) as well as important information regarding the editor’s analysis of the text’s genesis. There is much to be said for the inclusion of markup into the collation process, but at the same time this approach poses some additional problems. It involves a considerable amount of coding, since it requires software programs to be adapted (‘tailored’) to the markup schema of the text in question. I distinguish three possible approaches towards automated collation that allow editors to pass along designated markup elements and, as such, assist the user to visualize the text’s genesis in the collation output.

These approaches are illustrated with current practices of automated collation and placed in the aforementioned spectrum of opinion on collation. Each method under discussion either enables its users to include markup (to a certain extent), or has found a way to work around the problem. Nevertheless, the current approaches still have some downsides that are discussed in the final section of the chapter. The most common drawback of these practices is that the input file needs to be preprocessed before it can be collated with markup. The editor designates the kind of markup he or she wishes to preserve, and then transforms the input file in such a way that the collation algorithm does not ignore the specified tags. Some approaches allow the application of ‘postprocessing’ on the collation output. If it turns out, for instance, that the witnesses are not optimally aligned, the editor can analyse and manipulate the
output, and ‘re-feed’ the changed text into the collation program. This implies first of all that the editor needs to be technically savvy, and secondly that preprocessing is not free of interpretation.

Chapter 2.3. (‘Native XML Collation’), makes the argument for collation software that does allow users to preserve all markup elements and, moreover, uses the information encoded in tags, attributes and attribute values to better collate the input texts. This section discusses the advantages of XML-aware collation - that the text does not need to be stripped of its markup. In this model, the TEI-encoded texts do not require preprocessing: both the plain text characters and the TEI tags are collated. It is a particularly good solution for a manuscript transcription with several layers or writing stages: since all layers are taken into account, little to no information is lost in the collation process. In addition, the user retains significantly more information about the writing process than with regular collation software. The practical application of Native XML Collation results in a number of foreseeable challenges, for instance the general difficulty of explaining human values to a machine, or the visualization of the collation output. Such challenges only underline the necessity of further research in this area, and the final paragraphs of the section report on the preliminary results of an experiment to develop an XML-aware collation algorithm (section 2.4.2. ‘Experimenting with Native XML Collation’).

In the previous chapter, transcription and collation have been defined as modular components (‘bricks’) that benefit significantly from digital technology. To be sure, using automated collation software to compare TEI/XML transcriptions is a good example of modelling. The experiment with Native XML Collation, in which I use computational methods to influence the process of automated collation, is an example of digital philology. As such, the chapter addresses a second research question: can information about text be encoded and processed in a way that advances a study of its genesis? Two themes run as undercurrents to this chapter. First, the responsibility of the editor in relation to the use of computational technologies for text comparison and, by extension, the question of how to translate textual scholarship to code. This invites the evident questions of whether this code is scholarship in its own right and if so, how it is to be assessed it if only a small group of scholars can read it. The second theme relates to the training in computational techniques of a new generation of textual scholars. As such, the chapter also touches upon the broader topic of knowledge transfer between information scientists and textual scholars. As I provide a step-by-step account of the experiment as well as a
discussion of the results, I demonstrate how textual scholars can benefit from having basic computational skills, or at least from having a critical understanding of how to implement them in their research.

**Definitions: A Closer Look**

Although this chapter does not aim to present a full discussion on markup nor on text collation practices, a brief discussion of a number of key concepts is useful. First, considering that text collation is one of the main points of focus of the present chapter, it is probably good to stress the definition of text used in this thesis (see ‘Introduction’). To recapitulate, text is understood as a sequence of characters, spaces and punctuation. Text forms a kind of ‘trinity’ together with the document (as a material carrier of a textual record) and the work (as represented and interpreted by the sum of variant texts). Scholarly editors decide which aspects of this trinity are relevant for their editorial work (conform to their orientation) and need to be included in the transcription. It follows that a transcription holds not only text, but also information about the text, about the document and – possibly – also about the work. The editors subsequently determine what information in the transcription is going to be processed and collated.

Another consideration is the difference between markup and text. To a human reader, the characters ‘body’ in the XML tag `<body>` are not the same as the word ‘body’ in the text content of an XML element: they differ on a semantic level and a functional level. A computer program that has not been informed about the difference (`<body>body</body>`), will consider these characters as identical. Since it is possible to collate TEI/XML transcriptions with a combination of markup elements and text characters, it is important to be clear about this distinction. For reasons of clarity, therefore, I will use the term ‘text content’ (in the context of an XML file) to refer specifically to the text characters that are contained within XML elements.

This brings us to the term transcription. After all, what we generally collate are the transcriptions of documents or, to be more precise, the information these transcriptions contain. A logical question would therefore be how these transcriptions come to be and which transcription rules were followed? Most genetic editors recognize that the act of transcription is a crucial step in the examination of a text’s genesis, but this recognition does not always apply to other orientations of scholarly editing. Vanhoutte, for example,
describes how manuals of critical editing skip from collecting witnesses to collating witnesses, ‘silently folding’ transcription into the second step. He describes transcription as ‘non-critical editing’ and the ‘epistemic foundation for all further textual criticism’, stating that it should follow directly after the collection of witnesses (Vanhoutte 2007, online). In the same contribution, he defines transcription as, indeed, vital to genetic criticism, because of its focus on non-critical representations of the document. Note here the phrase ‘non-critical representation’; elsewhere I have already discussed the problematic idea that it is possible to produce an objective, non-critical transcription of a text (Bleeker 2015a). Crucially, this applies to both analogue and digital transcriptions. TEI/XML transcriptions present some original complications that will come up in the course of this chapter. For now, we can define a transcription of a text as a critical, editorial construct. It is created with one or several goals in mind (such as preservation, presentation, or legibility); and it encompasses a variety of information about the text, the writing sequence, the document material, and contextual information among others. In other words, it is a rich data structure that can be used for multiple purposes.

Subsequently, we need to define collation. Computationally speaking, collation is a form of data comparison. These data can be anything as small as single digits or characters. In the context of textual scholarship, the data to be collated are textual versions of the same work. We refer to these versions as witnesses. One work can have multiple witnesses, where each witness is usually one version of that work (e.g., manuscript A, manuscript B, typescript C, typescript D, and so on). Version, then, may be defined as a materialization or instantiation of a work that differs from another materialization (cf. Plachta 1997, 136). When applied to the practice of genetic editing, the question of what is a version quickly becomes more complex. In the case of a modern autograph manuscript, for instance, manuscript B might be heavily revised by the author and contain a large number of cancellations and insertions, inviting the question: at what point do a set of textual variants constitute a new version? It has been argued that each cancellation and insertion constitutes a new version of that text (cf. Zeller 1975), a theory that informs editorial practice to this day. Others have suggested that smaller revisions are better described as revision campaigns or stages in the writing process, and that giving equal treatment to versions and variation ‘impedes rather than furthers the study of revisions’ (Bryant 2002, 75). Since genetic critics understand the text as a continuous fluid movement, invariant text by its very nature does not exist: ‘comment parler de
“variantes” en l’absence de tout invariant?’ asks Pierre-Marc De Biasi (2000, 20). Some propose to use the words ‘rewrite’ or ‘genetic operation’ instead of ‘version’. Pierazzo, for example, says that the traditional concept of variant ‘is to be refused in a genetic critical framework’, because it implies the existence of such a thing as ‘one text with alternative formulations’ (2015, 185). She suggests using the term ‘substitutions’, which does make sense if you consider that one (range of) character(s) has been deleted or omitted and substituted by one or several others. The editors of the BDMP take a pragmatic stance: if we want to compare rewrites against each other, they argue, we need at least to designate temporary invariants (Neyt and Van Hulle 2016). They make a good case for this system – which they call ‘relative calibration’ – as the dynamics of the writing process indeed become clear by studying the textual instances that vary from one version to another.

An overview of the debate on versions over the years, from considering each variant separately to a statistical system for establishing the ‘best’ reading and an algebraic method to calculate relationships between witnesses, can be found in Greetham’s discussion in ‘Textual Scholarship: an introduction’ (1992, 323). Although the technologies for collating texts have evolved since then, the concept of version is still undecided. For now, the discourse is perhaps best summarized in the words of Roger Lüdecke, who states that versions are textual unities that show linguistic variation, adding that they are of ‘interpretative importance’ (Lüdecke 2013). This is supported by Shillingsburg, who argues that a version is an editorial construct (1996, 44). The purview and range of a version is thus one of the decisions a scholarly editor needs to make at the outset of her investigations.

The need to fix a version is understandable, observes Bryant, because ‘there is an undeniable comfort in seeking out a unitary object to edit or analyse rather than a hydra’ (2002, 94). However, it is exactly the understanding of these dynamics, the revisions piled on top of revisions, that helps us to study the dynamic writing process. Indeed, as Dillen asserts, ‘[t]he interpretation of … the distinction between different versions and different writing stages, is one of the most important tasks of genetic criticism’ (2015, 85). Looking at the matter practically, we need a working definition of version because the act of collation presupposes the commensurability of the entities being compared. As the saying

---

56 The French term ‘réécriture’ is more commonly used in genetic criticism, see e.g. Grésillon 1994, 245.
goes, we cannot compare apples and oranges. Following Van Hulle’s theory that textual genesis has five different aspects (chapter 1.3.3), the definition of version would vary according to the aspect of genesis the editor is concerned with. Accordingly, Van Hulle understands a version as a single unit of textual change, implying that the actual size of the version is flexible and can be as detailed as phrase version or even word versions.\(^{57}\) Note that the definition does not mean that every single unit of textual change automatically constitutes a new version of the entire text. That is, ideally the invariant text would not need to be reproduced each time a small revision occurs.

In theory, the editor has ample ways to discriminate between aspects of genesis (and thus between different kinds of textual versions) by combining the tagsets of the TEI Guidelines and the Encoding Model for Genetic Editing developed by the TEI Workgroup on Genetic Editions (henceforth WG-GE). Nevertheless, combining these tagsets leads to a number of practical obstacles that necessarily affect the collation: as the Text Encoding Initiative, the TEI focuses on the text on a document, but the WG-GE is also concerned with the materiality of the document. Although a TEI transcription can theoretically contain both perspectives, it obstructs the processing of such a file. The transcription practice is consequently highly significant. The transcriptions are the foundation for collation: in line with the axiom ‘garbage in, garbage out’, a collation output is only as good as the underlying transcriptions. We will see that transcription presents some particular challenges for genetic editing. In order to fully understand these challenges, I will now expand upon the concepts of markup, XML, TEI, and overlapping hierarchies. Others before me have discussed these matters more extensively (see for instance the TEI Guidelines, chapter v; Pierazzo 2015, 61-64; Dillen 2015, 51-79). I will therefore comment upon them only where it is relevant for the present study, which means I focus on markup tags and XML’s tree structure.

I start with the term central to this chapter: markup. We can distinguish a variety of markup types (cf. Coombs et al. 1987), the most relevant one being descriptive markup.\(^{58}\) One of the principal features of descriptive markup is the

\(^{57}\) In ‘The Inclusion of Paralipomena in Genetic Editions’ (2005), Van Hulle expands upon the German editorial term Textfassung (a version of a text) and argues that in order to correctly edit Joyce’s Finnegans Wake in a way that includes Joyce’s creative process, one also needs Wordfassungen to account for smaller versions of text.

\(^{58}\) In the words of Dillen: ‘Descriptive markup, then, does not tell the computer what to do, it describes what the text is (such as: ‘this section of the text is a paragraph’) ... To tell the computer which sections of the text are which, descriptive markup languages use ‘text graphs’ to
separation of form and content: on one hand the material document, on
the other hand the implied and abstract text. Accordingly, a digital editor has
to decide whether to mark up the document or its textual content, something
that is easier said than done because the two often overlap. A simple example is
a paragraph that runs over two pages of a document and may need to be
encoded like this:

```xml
<div page="1">
  <p xml:id="para-part1">
    <!-- part one of sentence -->
  </p>
</div>

<div page="2">
  <p xml:id="para-part2">
    <!-- part two of the sentence -->
  </p>
</div>
```

The distinction between textual content and material document constitutes one
of the fundamental issues of digital editing and, consequently, is a central
concern of this thesis. Some critics consider the ambiguous nature of electronic
text markup to be problematic, and propose transcription methods that clearly
distinguish between the documentary features and the textual content of a
text, but as I already noted, any type of transcription and markup is de facto an
interpretative and critical act.

Markup is most effectively used in combination with standards and
conventions (e.g., we generally agree that a period marks the end of a sentence; a
new sentence starts with a capital letter). An XML element can contain textual
content and consists of an opening and a closing tag, typically marked by
angular brackets, e.g., `<title>Sheherazade</title>`. One of the inherent

mark the beginning and ending of each section’ (2015, 53). It follows that descriptive markup is
always accompanied by a set of rules that explain the use of text graphs, i.e., a markup scheme.

59 Consider for example the markup of the following sentence: `<s>This is a <add
type="above" rend="grey pencil">simple</add> example</s>`. The values of the
@type and @rend attributes refer to documentary aspects, whereas the elements `<s>` and the
`<add>` are rather associated with the interpretation of the text (‘this is a sentence, this is an
addition’).

60 Buzzetti describes how markup is already held by the text, e.g., in the form of punctuation,
and added to the text in the form of an editor's interpretation (2002, 63). These different types
are not distinguished from one another since they are contained in the same format of
descriptive markup.
characteristics of an XML document is its tree-structure: it starts with a root element that ‘branches’ out to multiple ‘child’ elements. Trees have a hierarchical order, which means that all elements be properly ‘nested’ – a structure often compared to Russian Matryoshka dolls (e.g., Pierazzo 2015, 61; Dillen 2015, 61). Elements that are not properly nested – and thus overlap – produce a document that does not conform to the XML grammar and syntax, and cannot be processed. Such overlapping elements may be as small as a line break in a stanza, but they represent the more substantial problem. In short, text encoders using XML-based TEI have to choose between taking the structure of the source document as leading (i.e., following the structure of documentary elements such as page breaks) or taking the structure of the text as leading (i.e., following the arrangement of chapters, paragraphs, sentences, etc.). It is not always clear how to separate text and document: a source document can contain multiple versions of one text, for instance when a writer recommences working on a poem on the same manuscript page that contains an earlier, completed version of that poem. Or, conversely, a single manuscript page can contain versions of multiple works because a writer wanted to economize on paper (cf. Mathijsen 2003, 44). Because it imposes a strict hierarchical structure, XML is not designed to represent complex texts or texts in progress that do not adhere to such an internal hierarchy.

At the same time, XML is the predominant technology in the digital editing community. It provides its users with the largest variety of means to express information about a text, largely independent of the device or system used. Moreover, it constitutes the basis of the encoding scheme of the TEI. To conclude, it is important to know that although the terms XML, markup, and TEI are sometimes used interchangeably, they are not the same thing. When

---

61 All XML elements need to be properly nested, meaning that you always need to close the last text graph you opened before closing a previous one. If not, the elements overlap. The example below shows improperly nested XML, because the parent <text> element is closed before the <sentence> element:

<text><sentence>I like chasing moths by moonlight.</sentence></text>

62 Contrary to what was stated by Steven DeRose et al. in ‘What is Text, Really’ (1990), text is not inherently hierarchically structured. Indeed, in a later contribution Elli Mylona and Allan Renear contest their earlier statements (1996). For other contributions that are critical of XML, see for instance Vetter and McDonald 2003; Schmidt 2006; Schmidt 2009.

63 In principle however, the TEI does not depend on a certain technology. For instance, the Guidelines were originally based on SGML before XML became predominant. Were the textual editing community to adopt a new technology, it follows that the TEI Guidelines would adapt again.
talking about a **TEI-encoded text** or a **TEI/XML transcription** in this thesis, I refer to an XML document that is marked up according to the TEI Guidelines and would validate against a TEI encoding schema.

Since the XML data model only supports a single hierarchy, XML rules out the possibility of any **overlapping hierarchies**. Overlapping hierarchies are not *per se* negative in and by themselves. A contribution from the field of computer science points out that ‘overlap is needed when multiple independent items refer to the same segment ... we need overlap whenever multiple markup elements needs [sic] to be applied over the same content’ (Di Iorio et al., 2011). Overlap became a problem, the authors explain, when XML was adopted as the standard for textual markup:

> XML grammatically imposes and requires a strict hierarchy of containment generating a single mathematical tree of the document where no overlap is allowed. This requirement has been turned into an intrinsic characteristic of the documents XML was meant to represent, rather than a syntactical and conceptual constraint into which these documents need to fit (idem).

Overlapping hierarchies in TEI/XML transcriptions – or avoiding them – is widely considered as one of the principal challenges for text encoding. As a consequence, those working with TEI/XML files have tried to find ways to cope with these constraints or to find ways to work around them. Familiar ‘solutions’ used in the TEI community are milestones, segmentation or standoff markup. Although these solutions may work locally, researchers struggle to find a more general way out of the complex problem. What is more, these encoder solutions are often a hindrance when the TEI/XML files need to be subsequently processed and transformed.

Finally, I would like to reflect on **genetic markup**. Considering that any type of markup can result in overlapping hierarchies, even a relatively simple TEI encoding of the lines of a poem, the additional encoding of (authorial) revisions significantly increases the possible overlapping hierarchies within that file. Additionally, genetic editors aim to encode both features of the text and of the document, because both can be important for a genetic analysis. The WG-GE recognizes that the document plays an essential role in textual genetics and consequently proposes an encoding scheme for documentary features. These

---

64 Consider for instance the `<del>` or `<add>` elements, which are often nested within a wrapper element such as `<subst>` or `<seg>`. 
efforts demonstrate an increasing awareness within the editing community at large to consider the literary work within a document-oriented paradigm. It is however technically difficult to combine a text-oriented encoding scheme with a document-oriented encoding scheme: the differences between both tagsets often result in overlap and violations of the rules of XML well-formedness.

Another difficulty is the fact that transcription, whether analogue or digital, entails the linearization or ‘flattening’ of the text (Ferrer 2014, 58). Words and sentences may tumble over one another on the source document itself, but in the transcription they are put in linear order, from left to right and from top to bottom (for western scripts at least). Modern manuscripts, by contrast, which bear witness to the authorial struggle of writing and are so valuable for textual genetics, are multilayered. Words are crossed out; an alternative is placed above the line or in the margin of the text, crossed out again, and replaced again ad infinitum in the continuous dynamic of thinking, writing, and rewriting (see figure 2).

![Figure 2. Selection of MS B917-H2a-2 of Sheherazade (Brulez, ca. 1927, collection AMVC Letterenhuis B917/H/2a).](image)

Such modern manuscripts are what Mathijsen calls ‘complex documentary sources’ and contain different ‘layers’ that represent the successive stages in which the author worked on the text (2003, 47). She contrasts them with ‘linear documentary sources’ that contain a single version of a work with only ‘direct’ or immediate revisions. Complex documentary sources could also consist of a combination of manuscript and print with (handwritten) authorial revisions. The manuscript fragment of Brulez’ Sheherazade reproduced above, for example,

---

65 An exception is the ultra-diplomatic transcription, which aims to mimic the exact location of the words on the source document, although even here a certain linearization is unavoidable.
is written on the reverse of a sheet of typescript. The TEI suggests various tags to encode this kind of material, with attributes to indicate the place of the addition, the writing material, the writing hand, etc. However, any discussion of transcription and encoding of manuscript sources necessitates at least three critical caveats.

The first is the lack of critical consensus about transcription and encoding. Vanhoutte writes that ‘a coherent system or methodology for the transcription of modern material still has to be developed and tested, and that an ontology of the text must be agreed on’ (2007). Although he made this point some years ago, it is still valid. The question is whether the editing community will ever agree upon one coherent system for the transcription of modern manuscripts, or whether they prefer the flexibility offered by the TEI. A second issue is that encoding and transcribing these complex sources requires that they be linearized. A transcriber normally places a deletion before an addition in a transcription (regardless of the actual positioning of these elements), simply because – in the western world – we read a transcription from left to right and we usually assume that a word is first deleted and then replaced. Since the goal of a transcription is in the first place that a text is rendered more readable, the linearization or ‘flattening’ of the dimensionality of the manuscript page is desirable but not without cost. A third and final point is that the actual layout of the text on the page can be rendered with the aid of XSLT transformations and illustrated by a digital facsimile image. Chapter 3 discusses the matter of visualization in more detail, but it is important to realise that the software and tools used in digital editions process the TEI transcriptions of the text. In other words, the output is always an approximation of a complex object, always an act of interpretation on the part of editors and encoders.

This brings us to what I see as the most distinctive characteristic of genetic markup: the editorial analysis that is encoded in the genetic markup. Naturally, any encoding (or transcription) is the result of a critical act, but in the case of genetic encoding the issue deserves special mention. As the WG-GE points out, ‘there is an obvious difference between the interpretation that some trace of ink is indeed a specific letter and the assumption that a change in one line of a manuscript must have been made at the same time as a change in another line because their effects are textually related’ (WG-GE 2010, online). Genetic editors usually encode both types of interpretation in one tag, although we can imagine cases where it is useful to discriminate between them. Consequently, the WG-GE follows Hans Zeller’s famous separation of Befund
(record) und Deutung (interpretation) und proposes to talk about different ‘layers’ of interpretation: on the one hand, the ‘codicological or documentary point of view’ (i.e., what is on the document page), on the other hand the ‘process of writing and re-writing of the text’ (i.e., the interpretation of the relationship between different facts). This is similar to the encoding model that Barbara Bordalejo developed for the Commedia-project (2013). Bordalejo’s model distinguishes between the text as represented on the (source) document and the text as perceived and constructed in the reader’s mind. Although the model was designed for early modern texts, this similarity only demonstrates how the text editing community is in fact united in their endeavour to find the best possible encoding for complex texts. In practice, a separate encoding of the editor’s documentary findings and her textual interpretation (i.e., two layers of interpretation) may also lead to overlapping hierarchies in the XML file. Not surprisingly, the proposed encoding strategies to encode these layers of interpretation resemble those dealing with overlapping hierarchies in XML (Dillen 2015, 91). In conclusion, genetic markup considers the text from two nearly incompatible yet complementary paradigms. The documentary perspective, focused on the material aspects of a text, contributes to the textual perspective; together they form the editorial analysis of the genesis. It follows that it is complex but fundamental to take into consideration the genetic markup during the collation of transcriptions.

Let’s therefore take a closer look at automated collation software used in textual scholarship. In general, the software consists of a set of algorithms that detects differences between sequences of tokens. Tokens are the smallest units of comparison and in theory they can be any meaningful unit. In the practice of text collation, tokens are usually words (delimited by whitespace) and some punctuation, but CollateX for instance allows users to define the range and scope of a token. Hence, in the experiment discussed in chapter 2.3, XML tags are treated as separate tokens. The act of tokenization, then, which is usually also carried out by the software program, involves splitting up a text in a string of tokens that is by definition linear. Note that the input text had already been linearized once in the transcription stage described above.

Alignment implies comparing the tokens of each witness and connecting the matching tokens. In principle, any token in any witness can be aligned with any token in another witness, but of course, there is an ‘optimal’ alignment. As we will see, this optimal alignment is not a completely objective concept, but it depends partly on the editor’s ideal of ‘optimal’ and the expected
collation output. Aligning tokens presents a number of specific challenges, such as the definition of ‘match’ or the fact that tokens may not only be aligned based on their similarities but also on their relative position in a witness (see chapter 2.2.2 ‘Formalizing Collation’). An interesting question is whether users of collation software should be able to influence the alignment, and how. If so, it follows that the user also needs to define what exactly constitutes a match. When are two or more tokens considered alike? By default, a collation program has a certain conception of a match, but here too is room for customization. Should spelling variation, for example, be normalized or, in the case of the collation of XML files, should the same XML tags be a match when they contain different attributes? For these and other instances, the more sophisticated collation software allows for ‘fuzzy’ or ‘near’ matches and includes ambiguities in a witness’ text. Central to this process is the concept of a base text. In scholarly editing, the ‘base text’ has a double meaning and can refer to either the copy-text (i.e., the textual witness selected by the editor to constitute the basis of the edition), or the text against which other witnesses are compared. The difference is subtle yet important: in the second case, the base text does not necessarily constitute the text of the edition or the reading text; it has merely a temporal function within the framework of collation. Unless otherwise specified, I follow that definition and take ‘base text’ as the text against which other witnesses are compared.

Finally, the notion of textual layers can be confusing if not properly explained, especially in relation to the layers of interpretation mentioned above and the layers of genesis discussed in the previous chapter. Here, the term ‘textual layers’ refers to the different writing stages that can be discerned in a manuscript text. Since genetic critics aim to model the multidimensionality of the manuscript page and the genesis of the text by encoding the added, deleted and corrected text, the distinction of different stages is relevant for this type of research. These different stages in the writing process constitute ‘layers’ within the text on the document. For modern manuscripts with many corrections the TEI encoding can easily become quite dense and complex: documents containing three or more layers of text are not rare. Editors who need to extract one of these textual layers for automated collation face the task of selecting one layer (usually the top layer). They thus create a text that never really existed on the page, and that arguably has no more or less right to be selected than any other layer. However, if they do not intervene, the layer would not be included in the collation.
2.2. Theories and Practices

Translating texts into and through a computational environment produces both new and unsuspected ways of imagining textual groupings as well as ones that are mind-numbingly familiar - strange precisely because they are so mundane.'

(Piper 2015, 6)

Introduction

This section looks at theoretical approaches to including markup in the process of collation. Over the years, a variety of collation software has been developed. This section starts therefore with a brief outline of the main tools and technologies (2.2.1). I then go on to sketch the principles of automated collation following the Gothenburg model of a collation workflow (2.2.2). Finally, I consider a number of opinions regarding the collation of texts with (genetic) markup, ranging from simple forms of plain-text collation to more complex forms of ‘markup-aware’ collation (2.2.3 and 2.2.4). The diversity of these approaches is indicative of the field’s need for advanced collation software as well as the - recurring - fact that the ‘best’ method depends on the editorial orientation, the project’s objectives, and the type of textual material. Since most approaches require the active engagement of an editor, the issue of learning how to code is implicitly touched upon.

2.2.1. Automating Collation: Tools, Past and Present

Over the past few decades, a wide range of automated collation tools have been developed. Corresponding to the various conceptions of scholarly editing, the tools serve different purposes and goals. Although some of them are no longer in use, a short overview of these historical approaches to automated collation can be important for a future discussion.66 It illustrates where we come from and thus what we actually talk about when we talk about automated text collation. One of the earliest machine-assisted collation programs is TUSTEP, a set of tools developed by the University of Tübingen (the acronym stands for the

---

66 This overview is partly based on a Jupyter Notebook, created by Elena Spadini and myself for the DiXiT workshop ‘Code and Collation’. The workshop was organized together with Ronald Haentjens Dekker and David Birnbaum at the Huygens Institute for the History of the Netherlands in November 2016. The workshop material is available as Jupyter Notebook, see http://nbviewer.jupyter.org/github/DiXiT-eu/collatex-tutorial/blob/master/intro.ipynb (last accessed April 10, 2017).
TUebingen System for TExt Processing programs). Following the German editorial school, TUSTEP is primarily geared towards scholarly editors who are producing critical editions, but it also allows for manual intervention so that the editor has near to ‘perfect control over every step of the [editorial] work’ (Ott 2000, 99). Such intervention is facilitated by TUSTEP’s modular approach: the program discerns a number of discrete stages in the editorial workflow at each stage and for each stage it offers certain functionalities that support the editor’s work. For instance, the program distinguishes between ‘compare’ and ‘collate’. The first functionality compares two text witnesses as plain-text strings and stores the variation between them in a separate file. The second functionality takes that file and collates it with a base text (‘basic text’, usually witness A). ‘Collate’ then produces two files: a synoptic list of the text and its variants for ‘visual inspection’, and a file ‘for further processing’ (Ott 98). TUSTEP has been used to create most famously the critical and synoptic edition of James Joyce’s Ulysses.67 It is still in development68 and it currently also offers an XML-oriented frontend (TXSTEP) as a prototype.69

A second automated collation tool with a notable history is COLLATE, developed in the late 1980s by Peter Robinson, specifically for the collation of medieval manuscripts (specifically the Old Norse text Solarljóð and the Prologue to the Old English text Wife of Bath). Although COLLATE is no longer in use, it is of historical importance as one of the first tools that address the tasks of alignment, collation, and analysis.70 Throughout his research, Robinson has shown a keen interest in the workings of the computer and in how its functionalities may help editors with their research. For instance, COLLATE is developed according to the idea that giving computer programs an understanding of markup would lead to a better and more detailed collation of textual witnesses. Moreover, it divides that collation process into separate stages, allowing the editor to adjust the collation at several occasions. As we will see, this is a principle that most collation tools have in common. The software is used by the Canterbury Tales project, the Commedia project and the Perzival project among others. In 2008, Robinson decided to transfer the further

67 In his review of this edition, Geert Lernout outlines the technical proceedings of the editors of Ulysses (1986, 653).
68 Gabler’s experiment with diachronic markup in some texts of Joyce and Virginia Woolf is discussed in chapter 3.2.
69 See http://www.txstep.de/ (last accessed April 11, 2017).
70 See Robinson’s introduction to the Textual Communities project in which he recounts the origins and development of Collate (Robinson 2014).
development of COLLATE to a group of scholars involved in the Interedition project, which resulted in the collation tool CollateX. For a detailed timeline of this handover and the considerations that went into it, see Robinson 2014.

Similar to TUSTEP and TXSTEP, the software Juxta Commons supports the collation, comparison, and visualization of textual variation. It comes as either a desktop application or a web service and takes either plain-text files or XML-files as input. In the same vein, the ‘Multi-Version Document’ (MVD) technology of Desmond Schmidt takes care of several stages of the collation workflow (including a visualization of the result, although a suite of tools is required to further process the MVD output file). Schmidt originally created the MVD to process textual variation but has now taken on the challenge of overlapping hierarchies. The non-linear structure of text as it exists in multiple versions is recorded in one single entity, the Multi-Version Document.

Finally, as we will be discussing XML collation, it is relevant to take a quick look at the tools offered by the program oXygen, an XML editor used by many digital scholarly editors. The ‘Compare’ functionality of oXygen uses diff algorithms to enable the side-by-side comparison of two XML files. Diff algorithms usually operate on a line level granularity, although oXygen also offers a word level and character level comparison. ‘Compare’ calculates and displays the difference between two versions of an XML file. It is especially useful when you want to compare structured XML files, but not very well-suited to deal with complex TEI/XML transcriptions of draft manuscripts. The TEI Comparator, developed and maintained by Arno Mittelbach and James Cummings, seems to extend the Compare functionality of oXygen. It is also a text collation tool that allows for the comparison of two XML files, but uses a fuzzy-text comparison engine to compare units the size of paragraphs. As the short documentation indicates, it was initially developed to compare two TEI files but should technically accept as input other XML files as well. The use of ‘fuzzy’ matching implies that here, too, the editor/user plays a role in the collation process by manually correcting the detection of matches or accepting suggested matches.

In the sections 2.2.4 and 2.2.5, I examine the practical implementation of three collation tools, namely Juxta Commons, CollateX, and the MVD. Each

---

71 See http://juxtacommons.org/guide#advanced (last accessed September 22, 2016).
of these tools can be taken as an illustration of a different approach to automated collation: Juxta Commons provides editors/users with a complete ‘collation package’ including a GUI and a variety of visualizations, while the use of CollateX in this thesis illustrates how editors/users can become directly involved in the process of collation. The MVD, then, represents a departure from the technology of XML for text encoding and processing. I pay special attention to the way these tools handle markup elements, and I ask to what extent they successfully deal with manuscript transcriptions containing several layers of text. To be clear, I do not intend to give a comprehensive overview of every collation program available; others have provided such surveys elsewhere, e.g., Elisa Nury in her ongoing PhD research Digital Editions, Automatic Collation: From Theory to Practice (King’s College London).73 Nor do I intend to present a full and exhaustive discussion of automated collation practices. Instead, the collation software is selected because it allows – to a certain extent and in different formats – the inclusion of XML markup. The approaches presented here are portrayed as a spectrum in which not one method is preferred over the other. There is not single correct method to handle markup during the automated collation process; the approach depends largely on the objective(s) of the collation, the type of text, the orientation of the editor, and the final goal of the editing project. In some cases, the collation takes place within the environment of a digital edition and the amount of user involvement varies greatly. But, before we go on let’s take a closer look at the principles of automated collation. A detailed understanding of this process helps us to examine the ways in which the act of collation may contribute to genetic research.

2.2.2. Formalizing Collation: The Gothenburg Model

Let’s imagine a genetic editor who has carefully compiled the ‘avant-texte’ of a work, and encoded the documents according to a certain TEI based encoding scheme. In doing so, the editor has – implicitly or explicitly – established writing stages and revision campaigns, making sure to discriminate between different textual layers. All this information is stored in a well-formed XML transcription that validates against either the TEI as a whole or a customized scheme that is

73 See also the whitepaper of the Modernist Versions Project on collation tools (Hucalak and Richardson 2013).
derived from the TEI. In order to create a well-formed file and avoid overlapping hierarchies in the XML, the editor may make use of workaround methods such as milestones or empty elements.

Whether the editor wishes to collate plain text or XML files, the consecutive steps taken by collation software programs are largely the same. Most collation software follows the workflow of collation described in the Gothenburg model (GM). From 24 to 27 May, 2009, the developers of two collation software programs, CollateX and Juxta Commons, met in Gothenburg (Sweden) to discuss a more efficient collaboration between projects. They realised the need for a common understanding of the workflow of mechanical collation and an open source approach. I already mentioned in the previous chapter how the GM presents a good case of a separation of concerns into self-contained units or tasks that can be solved individually (in this case: what separate tasks constitute automated text collation?). To be clear, the model does not suggest any particular methods to solve these tasks; this is done by the software programs based on the model (see section 2.2.3 and 2.2.4). The idea of the GM was, of course, to try and automate the tasks as much as possible, but in some steps the editor’s influence or feedback is needed.

In the GM, the workflow of a collation program is conceptualized as a pipeline with five phases or consecutive steps:

1. **Tokenization**: the individual witnesses are tokenized into a stream (i.e., a linear sequence) of tokens. For most programs, tokens can be anything (text, markup elements, etc.) as long as they are properly defined at the outset.

2. **Normalization/regularization**: this is optional, but comes in handy in case of spelling variation, translated text, or non-unicode characters you do not want to be flagged as variants. The divergent tokens are given a normalized property (optional).

3. **Alignment of token streams**: in the alignment phase the token streams are compared and aligned based on certain criteria, for instance the largest set of matching tokens. Alignment is the core of many collation programs. It also provides the most interesting challenges for the humanities, because the answers to 'what is a match' and 'what is a good alignment' tend to be project and editor specific.

4. **Analysis** (possibly manual adjustment) of outcome: following the intricacies of text and the varying judgments of a 'good' alignment, it may happen that

---

74 See [https://collatex.net/doc](https://collatex.net/doc) (last accessed September 8, 2016).
the collation output is not satisfactory. Transposition, for instance, presents challenges for the alignment algorithm, because they usually depend on a user’s own interpretation. What is more, they depend on graphical elements on the document that do not easily lend themselves to transcription. Some collation programs offer their users the possibility to adjust the alignment and re-collate the tokens; others require an adjustment of the input. The connection between analysis and alignment can be described as a ‘feedback cycle’.

5. **Visualization**: the collation outcome can be visualized in different ways, ranging from variant graphs to alignment tables. Each of these tasks is performed by a separate algorithm. A collation program, then, is a combination of algorithms that work together to ‘solve’ each of the five steps.

Without diving deep into computer science, it is helpful here to briefly look at the third step, alignment, from a computational perspective. As said earlier, this is a critical operation within the collation process, because the alignment of tokens depends on the concept of ‘match’. Alignment is all about making decisions, the most important being the decision on what constitutes a match: under what condition may two tokens be considered equivalent? Clearly, the definition of match varies greatly in editorial practice. For now, we assume that our editor has provided some concept of a match, or uses the default ‘match’ determined by the collation program.

The first step in the collation pipeline is to tokenize the input text. Although a token can be any length, in the case of text collation it makes sense to use a text character or a word as one token. CollateX, for example, tokenizes on the word level and separates the tokens based on whitespace. Subsequently, tokens can be normalized or regularized. Note that ‘normalization’ in the context of automated collation is not equivalent to normalization that happens in transcription. For example, editors can transcribe orthographic variation because they consider it important to be preserved in both the transcription and the collation output. However, in the collation process itself, they may want to normalize orthographic variation because they do not want it to influence the alignment. In that case they need to normalize their tokens before inputting them in the collation software.

When given two witnesses (in the form of token streams), the alignment algorithm first tries to find large blocks of matching tokens. At that point, the location and order in which they appear in the text is not important. The
algorithm takes these blocks of matching tokens and aligns them as optimally as possible. At this point, the (relative) position of the tokens does matter. A simple example can illustrate this:

Witness A: the white moon and the black sky
Witness B: the silver moon and the dark sky

The word token ‘the’ occurs twice in both witnesses. If the relative position of the tokens is not taken into account, the token ‘the’ in witness A is a potential match with both the first and the second token ‘the’ in witness B. By looking at the surrounding word tokens as well, the algorithm has more chance at aligning the word tokens correctly.

There exist a number of algorithms that take care of alignment, the most common being the Needleman-Wunsch algorithm and the Dekker algorithm (both named after their respective developers). These two algorithms follow the principle of ‘progressive alignment’, which means that they do not compare all witnesses at the same time. Instead, they first compare two witnesses, store the result of that comparison in a so-called variant graph\(^\text{75}\) and then progressively compare another witness against that graph, at every turn merging the result of that comparison into the graph until all witnesses are merged. This progressive alignment method reflects the idea of ‘dynamic programming’, which takes a complicated problem and breaks it down in smaller sub-problems. In this case, the complicated task of multi-witness alignment is broken down into a repetition of the relatively easier task of two-witness alignment. A downside of progressive alignment is that, apparently, the order in which the witnesses are compared influences the final result. That is, the final alignment of three witnesses A, B, and C may differ if witness C is compared against the variant graph of witness A and B, or if witness B is compared against the variant graph of witness A and C.\(^\text{76}\)

\(^{75}\)A variant graph is a suitable data model to structure textual variation. It is a comprehensive data structure that can store a significant amount of information about the alignment of witnesses. For a clear explanation of the variant graph, see the documentation of the Ecdosis project see http://ecdosis.net/main/node/10 or the documentation of CollateX, section 2 on https://collatex.net/doc/ (both websites last accessed October 27th, 2016). I will discuss the variant graph format later (in chapter 2.3, chapter 3.2, and chapter 4.2).

\(^{76}\)For a brief but clear explanation of the pros and cons of progressive alignment, see the online documentation of the collation program CollateX on https://collatex.net/docs (last accessed September 8, 2016) and Spencer et al. 2004.
A simple representation of alignment of two witnesses is an ‘alignment table’. The alignment table is derived from the variant graph and thus contains less information, but it has the advantage of clarity and, moreover, it is a familiar format for most textual scholars. I will illustrate the concept using a sentence from the folio typescript of *Sheherazade*, and I collate it with its equivalent in the quarto typescript of the story. For reasons of space, I use only the first part of the sentence:

**Witness A:** En als je geen verhalen kent, dan
**Witness B:** En zoo je geen vertellingen kent, wel dan

Figure 3 (next page) shows the alignment table, its columns filled with the tokens of each witness (one token per cell). Witness A occupies the left column and witness B the right column; matching tokens are aligned in a row; an empty cell indicates that the token has no match.

![Alignment Table](image)

Figure 3. The vertical alignment table of two witnesses; with non-matching tokens indicated in red.
It is clear that the word tokens ‘als’ and ‘zoo’ are flagged as variant. The same applies to ‘verhalen’ and ‘vertellingen’. The punctuation is also taken as a separate token. The word ‘wel’ in witness B has no equivalent, so in order to correctly align the subsequent word tokens ‘dan’ the algorithm inserts an empty cell. Depending on one’s perspective (remember, there is no base text here), ‘wel’ is an addition or an omission. This alignment table visualization allows a scholarly editor to see at first glance the variation between tokens that are at the same relative position.

As mentioned above, the concept of match is pivotal in any form of collation. The editor can decide to either let the program find only exact matches or look for approximate matches (‘near matching’). The latter are based on the Levenshtein distance (again named after its creator) that calculates the number of edit operations needed to match two tokens. It takes only one edit operation to go from the word token ‘eigen’ to the word token ‘eigene’: the addition of the letter ‘e’. On the other hand, it takes as many as ten edit operations to go from ‘verhalen’ to ‘vertellingen’ (the omission of the letters ‘h’, ‘a’ and ‘l’; the addition of the letters ‘t’, ‘e’, ‘l’, ‘t’, ‘i’, ‘n’, ‘g’). Needless to say, this is an artificial metric used to compute text comparison: it has nothing to do with the way authors revise their work. The Levenshtein distance of the first set of non-matching tokens is 1; of the second set the distance is 10. If the editor expects a certain amount of variation between tokens that she does not want to be flagged as variant (for instance the German Eszett versus a double ‘s’), she can indicate that she allows an edit distance of 2 or 3. Conversely, of course, if she is interested in precisely such features as orthographical changes in the German language, she may want the collation software to detect the variation and allow less or zero edit distance.

The Editor and the Algorithms

Reading between the lines, we can see that the editor plays a significant role throughout this process of automated collation. Even when she does not interact with the code of the collation program directly, she still decides what a match is and, accordingly, which edit distance is allowed, whether orthographic variation matters, which textual layer to select, and so on. It should be clear, therefore, that all decisions made are project-specific and in that sense subjective. As a consequence, it is to be expected that the algorithm will not
function perfectly, that is, yield the most optimal results in the eye of the editor, right away. This brings us to step 4 in the GM: analysis. Here, the editor plays an even larger role.

In general, the analysis phase is where the editor and the collation program interact. Developers of collation software know that the whimsical contingencies of text cannot all be computed. For example, even an advanced collation program will not always succeed in figuring out the difference between a deletion and a transposition. The latter is a notorious challenge for collation software, which basically has to interpret the relationship between a cancellation and an addition. Even if all the information required to make an assumption about such operations as transpositions were encoded in the transcriptions it remains possible that the results yielded are not satisfactory to the editor. And, evidently, this is not unique for transpositions. As De Biasi writes, each genetic process has ‘its own rhythm’ and ‘unique itinerary’ (1996, 51). In short, as the creators of the GM confirm, textual editing is in the first place a heuristic process. For this reason, most collation software allows editors to give feedback upon the alignment outcome in order to adjust and improve the workings of the algorithm. Some examples of this will be discussed in chapter 2.2. (‘Theories and Practices’). It follows that at this point of analysis and feedback, the collation becomes increasingly project-specific. Because of this kind of editorial involvement in the collation process we sometimes talk about ‘semi-automated collation’, implying that fully automated collation – collation without any input of the editor whatsoever – is not feasible.

In the GM, the last phase of automated collation concerns the visualization of the final collation results. This is ‘a complex problem on its own’. At the same time, visualizing textual variance is one of the most discussed potential features of a digital scholarly edition. Genetic researchers frequently mention the hypertextual properties of the digital medium and the affordances of technology in this respect. Accordingly, I will discuss the visualization of collation results at length in Chapter 3 (‘Visualization of Textual Variance’). For now, I examine the theory and practice of automated collation in more detail. I begin by looking at the principles of plain text collation (2.2.3), and then go on to review several editorial practices that take (XML) markup elements into account during the process of collation. The practices can be

---

77 See section 1.5 of the documentation of CollateX on https://collatex.net/doc (last accessed September 9, 2016).
roughly divided into two categories: either the markup elements are transformed into plain text (2.2.4), or the markup elements are passed along the collation pipeline (2.2.5). Each approach is illustrated with examples from edition projects.

2.2.3. Plain-Text Collation

Plain text collation, as the name suggests, takes place on the level of the textual content. This means that a TEI/XML transcription is ‘stripped’ of its markup tags until only the plain text characters remain. This sounds simpler than it is in practice. Tags cannot be removed easily because they contain text as well. Consequently, when collating plain texts, the genetic editor selects per witness one textual layer – usually the top layer – that is collated against the same layer in other witnesses. Take for example the (linearized) TEI transcription of a sentence from the typescript of Sheherazade in which Brulez has made some simple revisions:

```
En <del type="crossedOut">zoo</del><add place="infralinear">als</add> je geen verhalen kent, zoek dan maar in je eigen<del type="crossedOut">e</del> droomen
(Sheherazade, TSfol-B917-H2bis-6)
```

If we look at the `<del>` and the `<add>` elements, we see that the word ‘zoo’ is replaced by the word ‘als’, and the letter ‘e’ is deleted. Here, the first (base or ground) layer includes the text in the `<del>` tags; the top layer refers to the last revisions made and thus ignores the cancelled text:

```
Base layer: En zoo je geen verhalen kent, zoek dan maar in je eigene droomen
Top layer: En als je geen verhalen kent, zoek dan maar in je eigen droomen
```

This is a type of textual variation that takes place in the text of a single document and creates two layers of text. Although the genetic value of this type
of variation is undeniable, it is commonly ignored when it comes to plain text collation.

As said before, the traditional objective of collation is to determine textual relationships, to establish textual transmission, and to map textual variation. Collation can be performed with or without a base text (as we saw above with the example of a sentence from the Sheherazade typescripts) and may or may not result in the construction of a critical apparatus and the establishment of a critical text. Since mapping variation is a goal an sich for genetic editors, they do not necessarily aim to establish a ‘final’ critically edited text. Rather, they aim to confirm what Raymonde Debray-Genette calls ‘the foolishness of trying to conclude’ (in: Genetic Criticism, eds. Deppman and Ferrer 2004, 93), emphasizing that there is no such thing as a ‘final’ text and that text is in continuous motion. From this concept of textual fluidity, it follows that most genetic editors prefer not to collate with a base text. Selecting a base text means giving preference to one draft text over another by labeling one witness as ‘more complete’ than the other. For an editor working with a large set of manuscript draft materials, objective criteria for selecting a base text are elusive. From the point of view of genetic criticism, the rationale against a base text is therefore similar to that of using ‘rewrite’ and ‘stages of revision’ instead of ‘version’: it is a way of freezing the ongoing, ever-evolving process of writing. Accordingly, most collation software programs do not require a base text. In a method called ‘baseless collation’, all witnesses are collated against each other, none is preferred over the other, and the collation program is, in that sense, less biased.

As I outlined in chapter 1.1, the requirements of textual genetic research come down to a consideration of the multidimensionality of the document page and a flexible representation of the relationships between documents of the genetic dossier. Pierazzo (2007; 2009) and Vanhoutte (2000), among others,

78 De Biasi writes for instance that with suitable marked-up text and with a powerful collation program, ‘il est possible d’en tirer, au-delà de la visualisation des simples modifications textuelles, des leçons plus générales sur les opérations génétiques comme l’ajout, le déplacement, etc.’ (De Biasi 2010, 53).

79 The intricacies of the composition process are described by De Biasi as ‘the activity of quite distinct writing movements belonging, despite their interdependence, to a complex logical scenario, full of indirection and diversification’ (1996, 30). Although De Biasi divides the writing process in distinct phases, its dynamics are still difficult to entangle and overlap is inevitable.

80 That is, the order in which the texts are collated against each other could make a difference, as explained in the previous section (2.2.2) on the topic of progressive alignment.
make the case for adding a fourth dimension to textuality: temporal perspective. In short, the collation program would pay attention to Van Hulle’s five aspects of genesis: exo-, endo-, epi-, micro- and macrogenesis. The question remains whether a collation software can (or can be made to) take all of these dimensions and aspects into account and, more specifically, do so simultaneously? Each level of genesis has a different perspective on ‘text’. Depending on where we stand on the vertical and horizontal axes of variation; whether we consider the paradigmatic revision sites or the longer syntactical or chronological sequences of text, the microgenesis or the macrogenesis: each time we have a different definition of text. Is it realistic to expect a computer program to automatically recognize the most relevant perspective? Aside from the potential for collation, such multi-dimensionality also poses considerable challenges for visualization (see chapter 3.2.2).

At first sight, the principle of plain text collation seems to collide with the idea of the multilayered manuscript, because it requires a complex manuscript to be reduced to a more simple, linear model of a text. But the first round of manuscript linearization occurs earlier, at the moment of transcription and markup, when the editor flattens ‘les segments et lexies que le manuscrit distribue de manière étagée’ (De Biasi 2000, 64-4) by placing them in a chronological, linear sequence. When considered in a regular XML editor such as oXygen or jEdit, a TEI encoded text also appears to be a linear sequence:

```xml
<text>
  <s>
    I like <del>eating</del><add>chasing</add> moths by moonlight.</s>
</text>
```

In fact, as figure 4 makes plain, it is a tree:

![Figure 4. A tree representation of the XML encoded sentence above.](image)
Transforming this tree into plain text means removing the markup tags, which results in an illogical sentence, so the editor is forced to select one textual layer, in this case either the ‘eating’ or the ‘chasing’ moths (with a semantic preference for the latter). The transformation also means that all other types of markup such as paragraphs and line breaks are removed as well. Although such TEI elements may seem less ‘significant’ than textual content, they are an integral component of the text’s organization and meaning - from line breaks in poetry to paragraphs in prose. Daniel Ferrer and Jean-Michel Rabaté, for example, demonstrate how paragraphs and paragraph breaks can also be meaningful in demonstrating the growth of a literary work (2004, 135; 142). Editors and developers try to work around the limitations of plain text collation, both theoretically and technically, finding ways to maintain the genetic aspects of texts during the collation and in the collation output. An objective of these studies that especially valuable for genetic editing, is the preservation of all XML tags that represent the editor’s interpretation of the text’s genesis when the different witnesses are collated.

This brings us to an important characteristic of automated collation that illustrates why it is sometimes called ‘semi-automated’: preprocessing. The editorial presence in the five steps of the collation workflow has already been mentioned. Although the respective algorithms take over a large part of the work, it is still the editor who divides the text into segments and extracts either a base layer or a top layer. It is important to realise that the text undergoes multiple transformations in the process of being transcribed and collated. With each step in the workflow, with each transformation the text undergoes, information is added or taken away. To be sure, this is not necessarily a bad thing: it is generally accepted and appreciated that the editor encodes her critical interpretation of a text in the TEI/XML transcription (Bleeker 2015a, 117; Pierazzo 2015, 99). As long as we are aware that the preprocessing, such as extracting markup from a transcription and other edit operations performed by the collation software, are also a form of interpretation.

Within the approaches currently adopted in textual research, we can roughly distinguish two different belief systems or perspectives on using automated collation software. Evidently, both want to obtain the most optimal collation results. Within the first perspective, editors believe that an optimal result is unattainable with software alone, so they opt for a manual correction of the output. The act of collation is thus considered from a traditional perspective where the software is used as a research assistant that produces a certain output
that the textual scholar interprets and corrects. We are reminded of Greetham’s remark:

Removing a large part of the drudgery from traditional textual scholarship has served to highlight the special role of critical intervention in the most significant moments in the production of a text. The machines can provide the data, and can even present us with the options possible, but editing is ultimately, like every other aspect of textual scholarship covered in this book, finally a critical activity, and this is as true in collation as it is in selecting variants... (Greetham 1992, 370-371).

Collation is not considered as a hermeneutical process in itself; the software program is rather a provider of data. The exact steps it takes during the collation (e.g., tokenization, alignment) are not clear to the user. Conversely, the second perspective on automated collation has editors and philologists customizing the data processing in order to obtain an optimal output. They engage directly with the collation, look ‘under the hood’ of the software, and adjust its algorithms to achieve a more effective collation and a most optimal result. As such, the act of collation is incorporated in the workflow of editing as well as textual research. In some cases, the collation software is even incorporated in a digital edition proper (cf. the implementation of CollateX in the BDMP). The two perspectives reflect the distinction between computer-assisted philology and digital philology (cf. chapter 1.2.1). In the terminology of the bricks method (cf. chapter 1.2.4.), the collation software would be one of the bricks that is configured and customized to fit into the infrastructure of a digital edition.

Within the first perspective, the main method to preserve – to a certain extent – the writing layers of a text is by taking each textual layer as a separate version of the text. The second perspective on collation, where editors engage more directly with the collation program in order to adjust it to their research interests, implies that most approaches are catered to a specific use-case. As a consequence, there is a variety of approaches, each informed by the material being collated, by certain editorial theories and by the editorial orientation to the text. Discussing each existing approach individually would not be sensible for reasons of time and clarity. Moreover, the approaches tend to be project-

81 By integrating CollateX into the digital edition, the BDMP offers their user/readers the chance to collate the versions of Beckett’s texts themselves. As such, the collation tool becomes an instrument to organize textual versions. In chapter 4.3 I discuss this approach and its implications more extensively.
specific and, as they continuously evolve, any such analysis would soon be outdated. For the sake of discussion, we can classify them as follows: (1) the genetic TEI/XML elements are transformed into plain text and collated as such; (2) certain markup elements are passed along through the pipeline of the collation program. Additionally, in some cases, the editor has the possibility to review the collation results and give feedback, for instance on the alignment of tokens, and feed the improved output again into the collation program. This way, automated collation becomes an iterative process that is continuously refined.

The following two sections (2.2.4 and 2.2.5) discuss the theoretical aspects of both plain-text collation and markup-aware collation, illustrated with an example of how they are put into practice in the framework of a digital edition.\textsuperscript{82} No matter which collation method is being used, though, we will see that there is always a form of preprocessing involved when the editor prepares the transcription files for collation. The different viewpoints on collation function primarily as illustrations of the argument outlined above: how collation can constitute a research instrument for genetic analysis by interacting directly with software and adjusting the algorithms to specific research interests. Each method is examined with the following points in mind – although they may not be addressed explicitly – (1) the way markup is handled; (2) the amount of preprocessing necessary to prepare the collation input; and finally (3) the involvement of the editor/user. Taken together, the methods present a variety of ways to deal with the theoretical and technological issues of automated collation discussed in the previous sections.

2.2.4. Transforming Markup to Plain Text

The inclusion of XML markup elements in the collation process can be advantageous for (genetic) research purposes. At the same time, it poses unique challenges at the technical level because the software needs to take into account additional layers of text and information and, most importantly, it needs to distinguish the XML markup from text content. One solution is to transform the XML elements to a format that does not complicate the collation process, such as plain text. In the aforementioned MVD approach, markup is stored...

\textsuperscript{82} A disclaimer is at place here: this discussion is primarily intended to be illustrative and therefore does not give a comprehensive overview of all digital editions that may follow the approach in question.
elsewhere (‘standoff’) and the different layers of plain text are collated against each other. That this technology is not entirely suitable for draft manuscript is illustrated with an example from the electronic edition of the story Achter de Schermen of Willem Elsschot, made by Peter de Bruijn, Vincent Neyt, and Dirk Van Hulle.

**The Multi-Version Document (MVD)**

With the MVD, Schmidt et al. propose an answer to the problems of overlapping hierarchies in the encoding of textual variation by avoiding XML-encoded transcriptions altogether (e.g., Schmidt 2009a; 2009b; Schmidt and Fiormonte 2010). The format is designed specifically to cater for different types of textual variation, as explained by its main developer Schmidt (2015, online). The two main characteristics of this approach are, first, that markup is separated from the text content and, second, that text content is broken down into fragments or ranges (called ‘standoff properties’). These are linked to a set of witnesses in which the specific fragment occurs. This pair of two values (textual fragment - set of versions indicated by their sigla) is stored in an MVD. The structure of an MVD is thus similar to that of a variant graph. It is a collection of nodes and arcs, in which common fragments of text are merged together and only the variant text is made explicit (see figure 5). One reads a variant graph from left to right, following the arrows.

![Variant Graph](image)

Figure 5. Example of a variant graph with four witnesses A, B, C, and D (Schmidt 2015, online).

---

83 The use of standoff properties is a way to extract markup from the text and store it elsewhere, linked to the range of text to which they apply. Contrary to standoff markup, the structure of standoff properties does allow overlap between properties and, moreover, the combination of sets of markup. See also Schmidt 2016b, 64.
Schmidt and Colomb (2009) provide more detailed information about the MVD and the technology of standoff properties. They first describe how the model of a variant graph captures the content as well as the variation of a text. Then, they demonstrate that the problem of overlapping hierarchies is caused by variation in the metadata (i.e., markup), and conclude that finding a solution to the problem of structuring and representing textual variation would also imply a solution to the overlapping hierarchies problem. Where figure 5 presents a variant graph of variation between witnesses, it is also possible to construct a variant graph of variation within a single witness. The model of the variant graph that they propose thus represents edit operations that are otherwise represented with markup – such as cancelled or inserted text, alternatives, or transpositions (see figure 6):

The resulting graphs are accordingly complex and more difficult to read, but they can however be into an MVD: a simpler output format that allows inter alia for a side-by-side text comparison of two versions (2009, 508). Briefly put, an MVD brings together a number of separate textual witnesses of a work. All text in common between witnesses is recorded only once, and all the differences are stored as separate files.

It is clear that the authors are aware of both the relevance and the challenges of collating texts with multiple layers. Their decision not to work with XML allows them to circumvent the challenge of dealing with overlapping hierarchies. In order to account for the multiple layers of revision in a draft manuscript, the layers are separated and treated as individual witnesses during the process of collation, and merged again in the collation output. This
significant point has several implications. First, however, we take a brief look at the technology behind the collation of the variant graphs. The collation program ‘Compare’ is used to process a MVD. ‘Compare’ does not recognize TEI/XML elements and consequently does not differentiate between text and markup. It is not quite correct to say that the collation program ignores the markup: it is not filtered out before collation, because the markup was never there in the first place. As a consequence, the process of collation remains the same as for regular plain text collation. Additionally, the collation program collates on text character-level, not on word-level. The reason for this, the editors explain, is that ‘word-level differences would obscure the comparison’. In other words, even when only one text character is different, the visualization would expand this to a whole word, ‘forcing the user to expend effort to discover the true difference’ (see Ecdosis 2016, section ‘Compare’). Overall, the advantage of this approach is that it avoids the problematic issue of overlapping hierarchies, simply because there is no longer any question of an XML hierarchy: the witnesses are linear strings of characters. The variant graph of Schmidt and Colomb does not recognize markup as such: it considers a TEI tag as plain text characters preceded and followed by angle brackets. The downside is that it is, so far, not possible to transform the plain text into TEI elements again after the collation.

Theoretically, it is possible to derive a pair-wise representation of the alignment of two witnesses from a variant graph. This, in turn, can be transformed into a traditional critical apparatus or into ‘alternative visual representations of variation such as a pop-up text’ (Schmidt and Colomb 2009, 509). In practice, however, the question remains how useful this is for genetic textual research. A TEI/XML transcription of a draft manuscript holds more than just revision layers; it contains the genetic analysis of the editor with valuable information about the variation (hand, place, possible writing sequence, etc.). This information is needed to generate an informative collation output (be it an alignment table, parallel segmentation or something else) that can be used to advance genetic textual research. It can be argued, furthermore, that the choice to create and collate pseudo-witnesses does not produce the proper results and may even be considered arbitrary or subjective. Nevertheless, by focusing on the layers of writing in manuscripts, Schmidt and Colomb demonstrate their awareness and willingness to tackle the obstacles of automated collation with modern manuscripts.
The technology behind MVD allows the editor to transform the plain text files together with the relevant standoff properties into HTML for visualization purposes. Theoretically, editors following this approach can store an infinite number of writing layers within one MVD by creating individual versions based on cancelled, added, and transposed (units of) text. This method seems to ask for an engagement with the code on a detailed level, requiring editors who either are reasonably skilled in terms of programming, or have a close collaboration with IT specialists. In order to meet these needs, the Ecdosis project is set up to provide an online platform and interface where editors may experiment with the MVD format and the collation tool Compare among others.  

One project making use of the MVD technology and the Ecdosis platform is ‘The Charles Harpur Critical Archive’ (CHCA), intended to be both a digital scholarly edition and a digital archive of the works of Australian poet Charles Harpur. The editors of Harpur’s texts distinguish versions of a work. For example, witness H001a and H001b would be derived from the witness H001, and within each witness a number of layers are identified (for instance, H001a-add and H001a-del). An editorial note, similar to a revision narrative, accompanies each witness, outlining the author’s method of revision and explaining how the versions are related. With regard to the collation, the editors assert that ‘to reach authoritative conclusions about the textual circumstances of each work the editorial phase requires a collation of variant versions in a far more detailed, character-by-character form than the less granulated kind needed for the archival phase’ (CHCA, ‘Editorial Policy’, online). In practice, this means that two layers of a witness (let’s say H004b and H001a-add) are collated and the outcome is represented side-by-side with the variant text characters in a different colour.

In a recent contribution, Schmidt elaborates upon the difficulty of representing layers of authorial revision in the text of a document (2016a, online). He first proposes that writing layers be distinguished on the basis of single instances of textual variation (e.g., a deleted text character) and then to use this layer as a temporary witness for collation. By comparing these

---

84 As successor of the AustESE project, the Ecdosis project aims to provide a ‘workbench’ and a set of tools for the digitization of cultural heritage tools (see http://ecdosis.net/main/node/1; last accessed September 20, 2016)

temporary layers with one another using collation, it becomes clear what text was cancelled and what text was added. The ‘exact position of the changes on the page’, normally recorded in the XML markup with for instance the TEI attribute @place, can either be added in a layer of standoff properties or inferred from the facsimile if the latter is represented alongside the transcription (2016a, online). In that respect, their practice seems to be informed by Zeller’s suggestion that ‘in the most extreme case’ one single authorial variant can make up a separate version (1975, 237). Schmidt recognizes that this practice of layering requires the duplication of invariant text, and that in cases where textual variation is limited to one or two characters, the result could be a vast amount of almost entirely similar versions. However, he counters that the data format of the variant graph that merges all invariant text and only shows the variation between witnesses; not the invariant text. Furthermore, he adds, the attribution of layers is an editorial act, ‘a matter of interpretation’: the editor decides on the number of layers she wants to identify (2016a, online).

Schmidt is obviously right in saying that the identification of layers of writing is an interpretative editorial act, and indeed editors must be pragmatic, to a certain extent. Nevertheless, grouping together the textual content of all <add> elements in a transcription and labeling the resulting text as one layer (e.g., witness H001a-add) belongs to the category of speculation. With that decision, the editor implies that all additions were made at the same time. In the case of draft manuscripts, in particular, such reasoning is too rigid and restrictive. Moreover, creating temporary witnesses based on textual variance presents a number of additional obstacles for specific textual instances such as instant corrections and open variants.

In conclusion, the collation method of the MVD, with the combined use of a variant graph and standoff properties, provides an interesting and valuable perspective on the issue of overlapping hierarchies in markup. It answers the needs of scholarly editors who wish to distinguish multiple layers of writing within one witness and include these layers in their collation. Since the collation itself is performed on the level of the plain text, we cannot define it as ‘markup aware collation’. The markup is lost when the TEI/XML file is transformed to plain text, but any relevant information regarding the documentary aspects can still be encoded as standoff properties or inferred from a digital facsimile representation. In that sense, the scholarly editor still plays a significant role in this collation method, but rather on a hermeneutical level as she discerns the writing layers and creates the resulting temporary witnesses. An
example from Achter De Schermen (ADS), an electronic edition of a short work by the Flemish writer Willem Elsschot and created by Van Hulle, De Bruijn, and Neyt, illustrates some of the pitfalls of this approach.\footnote{The electronic edition is published on CD-ROM in 2007.}

\textit{Achter de Schermen (ADS)}

ADS presents a good illustration of further complications of discerning layers of writing based on instances of textual variation. Although the collation method of the ADS edition is not relevant for this discussion because it ignores all (genetic) markup by comparing only the top layer of each witness in plain text format, I will use the ADS edition here to show how the approach discussed above becomes problematic when applied to a draft manuscript with several layers of revision.

Elsschot wrote Achter de Schermen as an analysis of the writing process of another of his works, Opdracht, so it is fitting that the editors of ADS carried out a similar analytical exercise.\footnote{Incidentally, in doing so the editors conclude that it is quite possible Elsschot took some liberties reconstructing his own writing process of Opdracht, at certain points making his revision process intentionally more rigorous (ADS, online).} Accordingly, the ADS edition presents an admirable attempt to disentangle the different chronological stages of Elsschot’s writing process. Instead of attributing a new layer to each instance of textual variation, the editors analyse the writing process sentence per sentence, placing the subsequent stages in a synoptic overview. For example, the editors distinguish three different phases (A, B, and C) of revision in the first sentence of the manuscript (AdsM1):

\begin{quote}
Step A: ‘Mijn dochter is getrouwd en heeft een kind gekregen.’
Step B: ‘Mijn dochter is getrouwd en heeft een kind gekregen.’
Step C: ‘Mijn dochter is getrouwd en heeft ons verlaten.’
\end{quote}

In the TEI transcription of the story, the editors numbered each <add> and <del> element according to the chronological order in which they reason the revisions took place. The TEI source code of the sentence above looks as follows (simplified for reasons of space):

\begin{verbatim}
87 Incidentally, in doing so the editors conclude that it is quite possible Elsschot took some liberties reconstructing his own writing process of Opdracht, at certain points making his revision process intentionally more rigorous (ADS, online).
Mijn dochter is getrouwd en heeft een kind gekregen.
heeft ons verlaten.

Here, the attribute value of the @ana attribute indicates the order of the revision. As Dillen points out, this ordering, which is already and editorial hypothesis, could quickly become purely speculative, especially for larger corpora in which it is nearly impossible to deduce the order of revisions (2015, 90).

Applying the MVD method to this material (i.e., creating hypothetical and temporary witnesses) works for the simpler transcriptions. To be sure, the example above contains a ground layer (step A) and a top layer (step C) of a sentence that could be collated against each other. Step A would temporarly form witness A; step C would become witness C; the invariant text would be merged and the variant text is the textual content of the <del> and <add> elements. Figure 7 reproduces a variant graph of this sentence:

![Figure 7. Variant graph with the invariant text in the nodes and the witness sigla on the edges.](image)

The editors of the ADS took the same approach, offering a user the chance to collate the ground layer with the top layer of the typescript transcription (AdsM2). Because a typescript usually has a clear ground layer, it is not too difficult to create a temporary witness (here M2 ground). All the same, the top layer contains multiple revisions in different colours of pencil and ink – the editors distinguish step B to step F. For collation purposes, these revisions have all been combined into one top layer (here M2 top). It becomes harder when the text presents textual variance in the form of an immediate deletion or an open variant. Figure 8, sentence 005 from MS AdsM1, presents such a complex example:
The top layer of the sentence reads ‘Een hoos heeft ons uiteengejaagd: dat meisje de baan van het moederschap op en ons beiden weer naar denouden haard toe, waar dat smeulend leven moet voortgezet.’, but the development of the sentence consists of 12 steps (A to J) in total. For reasons of space, five steps are given below but they suffice as illustration. Bold indicates text that was added in the course of revision:

Step A: ‘Een windvlaag heeft ons uiteengewaaid: zij naar hare m’
Step B: ‘Een windvlaag heeft ons uiteengewaaid: zij naar hare m’
Step C: ‘Een windhoos heeft ons uiteengejaagd: dat meisje naar hare bestemming’
Step D: ‘Een windhoos heeft ons uiteengejaagd: dat meisje naar hare bestemming’
Step E: ‘Een hoos heeft ons uiteengejaagd: dat meisje naar hare aanstaande moederschap en ons beiden weer naar denouden haard toe, waarin het vuur met moeite dreigt uit te gaan’

Following the theory that every significant authorial revision constitutes a new witness of the text, sentence 005 can be broken down to as many witnesses as there are steps, say witness 005-A to witness 005-J. It quickly becomes clear that it is not possible to determine a ground layer of this sentence, because Elsschot did not finish writing the sentence when he started revising. He made these revisions *currente calamo*: step A ends after the words ‘zij naar hare m’. Were the editors to follow the MVD method, it would be nearly impossible to determine where a witness would start and where it would end. Would witness 005-A comprise only nine words and the text character ‘m’? How then to display the edit operation of deletion and addition in the collation output? With more complex sentences this splitting up in successive steps becomes infeasible.
Consider for example the manuscript reproduced in figure 8: how can one say with certainty in which order the revisions took place?

To conclude, the MVD method presented by Schmidt et al. and implemented in the CHCA would perhaps work for specific material such as print proofs with sporadic authorial insertions, where it is easier to make the distinction between a ground layer and a top layer. However, the method of creating a hypothetical witness text that never existed (a ‘pseudo-witness’) for the sake of collation proves inadequate when working with draft manuscripts that contain revisions made **currente calamo** or open variants. Theoretically, the temporary witnesses do not have to be published and could only be used for the purposes of collation, but, as the WG-GE argues, it is debatable whether textual variation within a single witness can be considered equivalent to variation between witnesses.\(^\text{88}\) We may also ask to what extent the collation result is still suitable for further research and analysis. It does show the location of textual variance, but does not provide much detail. It would thus be difficult to distinguish layers and levels of revision, because each intradocumentary variant is considered as identical to a variant across manuscripts.

### 2.2.5. Pre- and postprocessing: Passing Along Markup

It is also possible to ‘pass along’ a number of specified TEI elements together with the text tokens through the collation pipeline. As the editor needs to prepare the transcription files, this method is part of the preprocessing stage. The collation itself still takes place on text characters only. We can roughly distinguish two methods to do this:

1. The markup that is passed through is primarily used for visualization purposes. This approach underlines the relevance of visualizing the collation output: it is usually in this format that the end-users encounter the collation result, using it for analysis and further examination.
2. Designated TEI elements are allowed to influence the alignment. This method enables editors to designate certain markup tags that influence the alignment of the tokens. Think for instance of a word that is printed in bold in witness A but not in witness B. The technology behind this

---

method is noteworthy because it requires the editor to engage with the
collation program on the level of the code.

Examples of the first method are Juxta Commons, the implementation of
CollateX in the BDMP, and finally the Collation Editor of the Greek New
Testament project. The second method is illustrated by the practice of Slavic
scholar David Birnbaum, who applies it to an edition of Orthodox Slavic
manuscripts. Since the next chapter describes visualization of the collation
output in detail, the following paragraphs concentrate on the technology
underlying each method.

**Juxta Commons**

Juxta Commons accepts as input complete TEI transcriptions, but the tool
filters out the TEI markup before it actually proceeds with the collation process.
As many of the TEI tags are saved as standoff annotations to the text tokens,
they can be visualized in the so-called ‘Heat Map View’ output. Editors who
wish to collate their TEI transcriptions need to prepare them for collation (a
pre-processing step) and decide which textual content of TEI elements need to
be preserved in the collation output. Furthermore, Juxta Commons allows users
to derive multiple witnesses from one TEI source file. In this respect it
resembles the method of the MVD discussed above. After inputting a TEI
transcription of a text with deletions and additions, one can create one witness
without the textual content of the `<add>` elements (i.e., the ground layer) and
one witness with the textual content of the `<add>` elements (i.e., the top layer).
When collated against each other in Juxta Commons, the collation output
shows the textual variation of deletions and additions as differences in red and
green respectively. This visualization may be useful for analysis, but the collation
output cannot be used for any purpose other than visualization. Thus far, it is
not possible export the output to, say, a TEI parallel segmentation. The
visualization is based on the temporary witnesses that no longer contain `<de1>`
or `<add>` elements and thus cannot be transformed back into the original
source file without losing these TEI tags. Deleted and added words are
outputted as plain text characters, producing a witness with text found on no
document page.

Finally, it should be clear that it is the editor who decides which TEI
elements are relevant to the collation; it is the editor who composes the textual
layers. As with the MVD method, Juxta Commons’s technology would not be able to account for heavily revised draft manuscripts: since it is not possible to filter on TEI attributes, editors cannot distinguish between instant corrections (<del> elements with the attribute @type="instant") or later deletions. TEI elements, TEI attributes and their textual content contain the editorial genetic analysis and are therefore vital for any further examination of the microgenesis and macrogenesis. For this kind of research, Juxta Commons does not provide the necessary technology.

There is another, more technically advanced method to pass along genetic markup through the collation tool: a method that allows an interaction with the underlying code of CollateX by transforming certain TEI elements into properties to the text tokens. In this method, designated markup can either be passed through unnoticed or actually influence the alignment of the tokens.

**The ‘Birnbaum Method’**

This approach is discussed in the tutorial ‘XML-to-JSON in Python 3’ that was written by Birnbaum for a one-day collation workshop preceding the DH 2014 conference in Sydney. Although the workshop was taught by several instructors, for reasons of brevity I will refer to the approach as ‘the Birnbaum method’. In short, the method requires the conversion of an XML-encoded text into a JSON format. The choice to convert into a JSON format is straightforward, explains Birnbaum, because at the time of the workshop it was the only format in which the editor could adjust the tokenization and normalization operations of CollateX. These customizations are necessary because, by default, CollateX collates only on a string level (i.e., plain text) and ignores XML markup. Birnbaum’s approach concentrates on preserving a selection of XML elements during the process of collation, so that they are still present in the output and can subsequently be used for visualization purposes. In other words, the XML markup is passed through the collation pipeline but does not influence the collation proper.

Birnbaum’s own research concentrates on Orthodox Slavic manuscripts that pose complex collation challenges of their own – for instance, the normalization of the Cyrillic script in which various manuscripts are written. An

---

89 For the workshop’s website and material, see [http://collatex.obdurodon.org/](http://collatex.obdurodon.org/) (last accessed September 22, 2016).
example of these texts can be found in the online critical edition of the Primary Chronicles, or Povest ’vremennyx let, which uses semi-automated collation primarily to group sets of lines across multiple witnesses. Birnbaum calls this ‘leveraging’ the XML-markup: making use of existing line divisions in order to simplify the collation and to reduce the work of CollateX.

The online tutorial makes use of simpler example texts from the electronic edition of Partonopeus de Blois that offers diplomatic TEI/XML transcriptions of the Old French and Middle French manuscripts. Here, too, markup can be used to create sets of witnesses based on the unique xml:id attribute of the <l> element for lines. Birnbaum treats this as a separate and optional step; to be sure, the main focus of the approach is on preserving the designated XML elements. The use of different types of text reinforces the point that the method is not project-dependent and can in fact be applied to any kind of TEI/XML transcription. As we will see below, the editors of the BDMP use it as well. The Birnbaum-method consists of three main steps: (1) the preprocessing phase in which the XML is prepared for the conversion into JSON; (2) the collation proper carried out by the existing algorithms of CollateX; and (3) the postprocessing phase in which the JSON output is again transformed into XML. Since JSON only recognizes plain text, the differentiation between markup and plain text takes place in the preprocessing phase and not during the collation itself. For this reason, the following discussion focuses on the first step. It will be quite an extensive discussion, too, because (a variation of) the Birnbaum method is used by other projects and it implicitly clarifies several operations of CollateX.

In the preprocessing phase, the input XML file undergoes a number of XSLT transformations to ensure that the next step, the tokenization, is performed correctly on both XML elements and text tokens. Normally, CollateX creates individual tokens based on whitespace characters. This works fine for text nodes because words are usually separated from each other by spaces. However, XML tags may include insignificant spaces that do not represent word boundaries, for instance between the name and the attribute in the following tag: <add place="above">. The regular tokenization algorithm of CollateX would in this case create two separate tokens (one for add and one for place) instead of one token (add). It is therefore important that, during the

---

91 For the complete archive, see http://ota.ox.ac.uk/desc/2499 (last accessed August 2, 2016).
preparation of the TEI/XML file, spaces within text nodes are treated differently from spaces within XML tags. With regard to the text nodes, Birnbaum inserts empty \(<w/>\) elements (i.e., milestones) in the place of a space and subsequently transforms those \(<w/>\) milestones into wrappers. This ensures that every word token is wrapped in a \(<w/>\) element. For instance, the following hypothetical sentence:

\[
\text{Nus clers ne vos <crease>poroit</crease> descrire}
\]

looks like this after the first round of transformation:

\[
\text{<w>Nus</w><w>clers</w><w>ne</w><w>vos</w><w>poroit</w><w>descrire</w>}
\]

Again, this ‘word-wrapping’ is only done for the text tokens; the approach is slightly different for the XML tags. First of all, the user needs to decide which elements he or she wants to retain. Although the selected XML elements do not influence the collation proper, they can be of use for the representation of the collation results. An example of this is the word token \textit{deus} in witness A and a token \texttt{de<abbrev>us</abbrev>} in witness B: the tokens are aligned because the collation algorithm ignores the markup and collates them on their string value, but it may be desirable to visualize in the output that the token in witness B is in fact an abbreviated form of the word ‘deus’. The editorial decision which elements to retain may differ per text, which is why it is best if the editor writes her own transformation script. This illustrates a point made in the previous chapter: the flexibility of TEI and the ensuing idiosyncratic XML transcriptions make it difficult to produce a general set of rules.

The selected XML markup comes in different degrees of complexity: (1) simple markup, such as the aforementioned demarcation of lines with \(<l>\) tags; (2) markup that is confined to a single word token such as \(<del>deus</del>\) or \texttt{de<del>us</del>}; and finally (3) markup that encompasses multiple word tokens such as the \(<crease>\) element in the line \(<l>Nus clers<crease>s ne vos poroit</crease></l>\). Especially the latter two pose some difficulties for the XML-to-JSOn conversion. Most of these are caused by the transformation from tree to string, i.e., from the hierarchical structure of the XML file to a linear sequence of characters that forms the JSON input. By default, the program treats the XML input file (which has a hierarchical element structure) as if it were a non-hierarchical stream of characters. The characters representing
markup are just part of the stream: they are text characters that coincidently have angle brackets. This means that the word tokens \textit{deus}, \textit{\texttt{<del>deus</del>}}, and \textit{de<del>us</del>} would not be considered a match by the alignment algorithm. The solution offered by the Birnbaum method is to normalize the word token, stripping away the internal markup. This normalized ‘shadow’ would be given to the JSON word token object as an ‘n’ property. The ‘t’ property of the JSON word token object would be the string value with markup. The JSON word token object of our hypothetical ‘deus’ example would then look as follows:

\begin{verbatim}
{ "t" : "\texttt{<del>deus</del>}", "n" : "deus" }
\end{verbatim}

By default, the program collates the value of the ‘t’ property unless the JSON word token object also has an ‘n’ property, then it first tries to identify correspondences based on the value of the ‘n’ property. If it does not detect an ‘n’ property it reverts back to the ‘t’ property. This ensures that, for example, the word tokens \textit{de<del>us</del>} and \textit{deus} are considered a match. CollateX outputs and renders only the ‘t’ property of the JSON word token object, which is incidentally still a string of characters at that point. In the third step, the postprocessing, this string can be converted back to XML. In this respect, too, the Birnbaum method is distinctly different from the MVD and Juxta Commons, which do allow XML input but only to collate it as plain text with standoff properties.

The third category, XML markup that spans multiple word tokens, presents the more complex issue of overlapping hierarchies that violate the rules of XML well-formedness. Consider for instance the hypothetical line \begin{verbatim}<l>Nus\texttt{cler<crease>s ne vos poroit</crease>}</l>
\end{verbatim}, where the \texttt{crease} element starts in one word token and ends in another. When the word tokens are wrapped in their individual \texttt{w} wrappers, we end up with a problem of overlapping XML hierarchies (highlighted):

\begin{verbatim}<l><w>Nus</w><w>cler<crease>s</w><w>ne</w><w>vos</w><w>poroit</w></crease></l>
\end{verbatim}

Birnbaum proposes a solution – or at least a way to work around the problem – by means of flattening the XML file, i.e., eliminating one hierarchy. Using a second XSLT transformation, the XML elements that span multiple word tokens (and as such form a risk for overlapping hierarchies) are converted to
empty milestone tags. These milestone elements are given an attribute with a value indicating whether they are a start tag or an end tag. The word token ‘clers’ of the example above would then look as follows:

\[
\text{<w>cler<crease type="start"/>s</w> } \text{ <-- some text -->}<\text{crease type="end"/>}
\]

Note that the only tags that are flattened are those that might span word boundaries; the tags that are confined to one word token do not pose a risk of overlapping hierarchies. The JSON word token object would look as follows:

\{
  "t" : "cler<crease type="start"/>s",
  "n" : "clers"
\}

In the next step, the postprocessing phase, the JSON output (containing plain text) is again transformed into XML and possibly into HTML by means of XSLT stylesheets. A part of the markup that is passed through can be rendered again to XML as well. However, Birnbaum points out that this is only possible for the second type of XML markup that is limited to one word token only. If we take a look again at the ‘clers’ example above, only the \<crease\> start tag is part of the word token clers; the end tag is associated with the word token poroit. The tags could therefore not be converted back to one XML element with a start tag and an end tag, or they would not conform to the XML laws about well-formedness and properly nested elements. In the method as described in the online tutorial, Birnbaum decided to keep the milestones with a start or an end attribute in the output, so that one has at least an indication of where they are located in the text. Elsewhere, he suggested that a variant of this method may be to flatten all XML markup (that is, not only the XML elements that span multiple word tokens), and leaving it flattened until the postprocessing phase, where the editor can decide which flattened tags to restore to proper XML elements. In both cases, however, this would only work for the XML elements that are confined to one word token only.

The Birnbaum method is one of the more advanced approaches to automated collation, in the sense that it requires a significant amount of work (i.e., coding) on the part of the editor. The input (and possibly the output) of CollateX has to be adjusted according to the characteristics of the text in question, which means informing the program how to handle certain XML elements that occur in the transcriptions, such as \<sic> and \<l>. The editor interacts with CollateX on a more direct level; it is an interaction that does not
take place within (the interface of) a digital edition but at the level of the code. Although the steps taken within the pre- and postprocessing phases require a basic understanding of JSON, Python and XSLT, the approach should be feasible for the average editor, especially when she takes advantage of the comprehensive documentation in the online tutorial. The high level of editorial involvement in the technical details of the collation process builds the idea that scholarly editors can profit from interacting with the code they use – on the one hand, by getting to know their XML-encoded text better, and on the other hand, by adjusting the CollateX program to cater specifically to their needs.

With regard to the role played by XML markup, it is clear that in the Birnbaum method only the text characters (i.e., the word tokens) are collated. The hierarchy in the XML file is flattened or, to put it in the words of Birnbaum, ‘the [collation] program doesn’t care about XML elements at all’. In a way, though, the Birnbaum method can be described as ‘markup-aware’ collation, but this awareness solely regards the preprocessing phase where spaces within a text node are distinguished from spaces within XML tags. This information is then used to tokenize the text nodes based on words separated by spaces. After the editor/user decides what XML elements are to be present in the collation output (and thus have to be preserved during the collation process), these make up the ‘t’ property of the JSON word token object. The normalization operation strips the word token of all markup, which forms the ‘n’ property. CollateX uses the ‘n’ properties to identify matching tokens, passing the ‘t’ properties through untouched. In other words, it matters not for the alignment algorithm whether, say, the word token deus is wrapped in a <del> element or an <add> element. Editors working from a genetic orientation who consider this information valuable may want to render that information in the collation output.

They also have another option. Birnbaum already hints at this when pointing out that the editor is not bounded to using the ‘t’ and the ‘n’ property only. He writes ‘other properties are for the convenience of the user; CollateX passes them through to the output, but does not use them directly’ (Birnbaum 2015, online). This is a valuable point: it means that editors can create arbitrary JSON properties and assign to those properties other information from the XML markup. A token can for example receive the additional property ‘type’ with the value ‘deletion’:

```json
{ "t" : "<del>deus</del>", "type" : "deletion" }
```
While Birnbaum is right that normally CollateX does not take the extra properties into account when matching and aligning tokens, recently a new matching function has been added to the CollateX library that does consider additional properties. Here, the new matching function overrides the default one. As a consequence, editors can decide whether tokens with different values for the property ‘type’ (e.g., ‘deletion’ or ‘addition’) are actually considered a match. The editors of the complete works of Dutch author Willem Frederik Hermans follow a similar approach. They consider the layout of the typescripts and print proofs significant for the textual development of the works. The project wants the tokens `<bold>title</bold>` and `</italic>title</italic>` to be flagged as variant, which can be done for instance by creating an extra property ‘print’ to indicate the way the textual content appears in print:

```
 [{ "t" : "<bold>title</bold>", "n" :  "title", "print" : "bold" }
 and
 [{ "t" : "title", "n" :  "title", "print" : "italic " }]
```

Although the ‘n’ properties of the tokens are a match, this is overruled by the presence of the ‘print’ property with variant values.

In conclusion, the method to pass markup through the collation pipeline is technologically advanced and at the same time highly useful for editors who wish to influence the alignment of the text tokens. The method provides a way to process significant markup elements with CollateX, while the textual content of the tokens is still leading and markup is treated as secondary. A final, more general point that arises from this approach is that we need to be conscious of our understanding of the data formats of XML and JSON. Although they are often represented as serialized strings in a linear sequence (think of a TEL/XML transcription in the oXygen editor), they have in fact a hierarchical tree structure. This means that they are processed differently than we may think intuitively. It is important to understand how the presentation of XML or JSON as a linear sequence influences the perception of the scholarly editor. Van Hulle, for example, refers to Ferrer’s description of a linear

---

92 The project ‘Willem Frederik Hermans, Complete Works’ is carried out at the Huygens ING, see [http://www.wfhermansvolledigewerken.nl/?lang=en](http://www.wfhermansvolledigewerken.nl/?lang=en) (last accessed October 24, 2016).
transcription as an aplatissement or flattening of the text. In a way, this is indeed the case: in order to make clear the various stages of writing, the deletions and additions are apparently placed in a linear order. However, since XML has a tree structure, these elements are in fact not ordered linearly but hierarchically. It is paramount to keep in mind the conceptual difference between formats when we start thinking about working with XML or JSON files, because their format affects the way the editor handles them and the ways in which they are processed by a computer.

The BDMP: Experimenting with the Collation of Manuscripts

Since 2009,91 the editors of the BDMP have integrated CollateX into their digital edition of Samuel Beckett’s draft manuscripts. As the current technological approach to automated collation in the BDMP is highly similar to that of Birnbaum, I will not treat it in detail here. Suffice to say that by passing along TEI elements such as <del> and <add> as JSON tokens through the pipeline of CollateX, the layers of writing per sentence can be visualized in the output. The following discussion focuses on the editorial considerations of the BDMP editors regarding the role of automated collation in their genetic research.

With regard to their orientation to text, the editors write ‘we have come to appreciate text in its essential fluidity and its forms as a process rather than a static object’ (Haentjens Dekker et al. 2014, 463). In other words, their approach is informed by the genetic orientation, something that is also apparent in their discussion of automated collation and their suggestion to use of the concept ‘temporary invariant’ to compare other variants against (see chapter 2.1.2). The unit of comparison (a version or witness) in this case is not a complete text, but the size of a sentence. Similar to the ADS edition, the BDMP offers a synoptic overview of the textual development per sentence. The textual variants of the sentences are presented ‘in their syntactical context’ in order to ‘visualize the dialectic’ of literary genesis (Van Hulle and Neyt 2016, online).

Considering that the textual material consists, in the main, of draft manuscripts and that two of the editors of the BDMP were also involved in the

---

91 The first results of the collaboration between the CMG and the Huygens ING on the topic of automated collation were presented at the workshop on ‘Current Issues in Digitally Supported Collation’ on 18 and 19 November 2009.
making of the ADS edition (i.e., Van Hulle and Neyt), it is not surprising that they expand upon the same issue discussed above: how to collate the multitude of textual layers that can be discerned in a sentence? Ideally, one would like to compare separate writing stages in one sentence against other stages in different sentences. A conceptual illustration of this possibility is given in the article ‘Computersupported collation of modern manuscripts: CollateX and the Beckett Digital Manuscript Project’ (Haentjens Dekker et al. 2014). Their exploration of ways to collate not only versions of text, but also writing stages within versions is especially insightful.

First, the authors outline the technological specifications of the automated collation program CollateX, which builds upon the concept of variant graph as proposed by Schmidt and Colomb (see above, figure 5). The variant graph is able to represent the possible edit operations in a text (e.g., deletion, addition, open variant, transposition). CollateX’s variant graph is conceptually similar to Schmidt’s but has the text in the nodes and the witness sigla on the edges, as already seen in figure 7 above. Regarding the use and interpretation of a variant graph, the authors note that it ‘ultimately describes variation between witnesses; it does not – nor does CollateX – interpret or infer the type or cause of variation’ (Haentjens Dekker et al., 460). With this remark, they emphasize that the interpretation of collation output, whether in the form of an alignment table or a variant graph, ultimately lies with the scholarly editor or the end-user.

In the remaining part of the article, the authors describe a theoretical method that would enable the collation of writing layers within one textual witness. They use the following example from Beckett’s Stirring Still manuscript (UoR MS 2934, 9v-10r) with revisions in Beckett’s hand:

and then again faint from far within

They encode this sentence in TEI/XML as follows:

```xml
<subst xml:id="subst1"><del xml:id="del1">faint</del>
   <add xml:id="add1"><subst xml:id="subst2">
      <del xml:id="del2">hoarse from long silence</del>
      <add xml:id="add2">faint</add></subst>
</add></subst> from far within
```

124
By grouping together the attribute values based on their number, the authors consider it possible to differentiate between writing stage 1 (S1) and writing stage 2 (S2) not only in the transcription, but also in the collation. They suggest introducing a new node in the variant graph introducing a writing stage. This outputs as the following variant graph:

![Variant graph showing two different stages (introduced by node S1 and node S2) in the writing process of a sentence from Beckett’s manuscript (2014, 467).](image)

The main advantage of this approach would be that other versions of the same sentence could subsequently be collated against this variant graph. In other words, it would be possible to take into account different writing layers within a sentence. Furthermore, this method allows the unit of comparison, originally set at sentence length, to decrease to an even smaller unit of textual change (cf. the definition of ‘version’ given in the introduction to this chapter).

At the time of the contribution’s publication (2014), this method existed only as a concept. The authors indicate their awareness that the collation input still needs to be prepared before it can be processed by CollateX and state that it would be ‘helpful’ if the collation input were directly derived from the TEI/XML file. Additionally, this would remove the need to preprocess the TEI/XML transcription. For this to happen, CollateX would need to be able to understand the difference between XML and text characters. Moreover, the TEI/XML transcription would need to contain not only the stages but also the substages (e.g., indicated by the attribute value subst1, subst2, etc.) of writing. Recently, this approach has been further explored, again as a collaborative effort between the CMG and the Huygens ING. In short, <del> and <add> elements that are part of the same revision are wrapped in a <subst> element to indicate they are part of the same writing stage or revision layer. When the collation software is made aware of TEI tags, it is able to recognize the <subst> element. By treating the <subst> element as a so-called OR operator that indicates the presence of ‘operands’ (in the form of <del> and <add> elements), the textual content of all operands can be taken into account. At the moment of writing
this research is still in an experimental stage. A number of preliminary findings were presented at the joint ESTS/DiXiT3 conference 2016.\footnote{Bleeker, Elli, Bram Buitendijk, Ronald Haentjens Dekker, Vincent Neyt and Dirk Van Hulle. 2016. ‘The Challenges of Automated Collation of Manuscripts’, paper presented by Vincent Neyt at the joint ESTS/DiXiT conference in October 2016 in Antwerp.}

Apart from being one of the most advanced experiments with collating textual layers, this approach also demonstrates the benefits of a close collaboration between computer scientists and scholarly editors. It confirms that the intricacies of draft manuscripts and the desire of textual genetics to record them can also be interesting from the perspective a computer science.

**The Collation Editor: Feedback Cycle**

Another conceptually interesting approach to automated collation is presented by the ‘Collation Editor’, a tool that is based on CollateX and that was developed by Catherine Smith of the University of Birmingham.\footnote{See the project’s GitHub page for a detailed account of the workings of the Collation Editor and instructions as for how to use it: \url{https://github.com/itsee-birmingham/collation_editor} (last accessed April 13, 2017).} It is used for a digital edition of the Greek New Testament (the *Editio Critica Maior* project, henceforth ECM). The textual material is not immediately relevant for genetic criticism in the sense that it concerns the transmission of medieval biblical manuscripts instead of manuscripts with authorial revisions, and the collation itself takes place on the plain text. Nevertheless, the project’s method of engaging with the collation software and customizing the initial output demonstrates a theoretically interesting perspective on the editor’s role in the pre- and postprocessing phases of collation.

The Collation Editor is integrated into an online collaborative work environment, conceived for the editing of the ECM. In a recent publication (2016), Smith and Hugh Houghton describe the workflow of collating TEL/XML transcriptions. Owing to the use of collation software, the focus of the ECM has shifted from producing a critical apparatus to the transcriptions that accordingly become more authoritative. That is, the collation result is not considered ‘primary evidence’ but merely an abstraction, and when editors judge the output to not be adequate, they can return to the transcriptions and collate them again (2016, 115). As such, Smith and Houghton write, ‘the collation is simply an abstraction, processed data rather than primary evidence’ (2016, 115). Though automated and carried out within the online work
environment, the process of collation is broken down in several stages that allow
the editor to intervene in the collation whenever they disagree with the initial
output. The editorial intervention into the operations of the collation software
is exactly what makes this approach to collation thought-provoking.

First, the TEI/XML transcriptions undergo a round of preprocessing
during which ‘separate readings’ (119) – i.e., the corrections in different hands –
are extracted. Presumably these separate readings are understood as temporary
witnesses in the fashion of the MVD, although Smith and Houghton do not
elaborate on this. The first collation is carried out with a number of set
parameters regarding the original spelling, the treatment of abbreviations, etc.
Editors/users nevertheless have the choice to influence these parameters too, so
that they can ‘tailor their apparatus to the purpose at hand’ (119). After the
witnesses are aligned, the output is presented in an interactive user interface.96
Here, the editor/user can ‘drag and drop’ in a new order tokens that she
considers misaligned. The rearranged tokens are then re-collated to produce a
new output.

Smith and Houghton outline the different forms of editorial input in
the process of collation, starting with the regularization of orthographic variance
or obvious spelling errors. Editors can furthermore overrule the length of units
of variance by ‘combining or splitting them in order to present them in the
most logical way’ to the editor (120). With the creation of this interactive user
interface around the Collation Editor, Smith ensures that ‘all manipulation is
left to the scholar’ without requiring of that scholar to interact directly with the
underlying code, thus lowering the threshold to participation for editors/users.
Being a computer scientist herself, Smith remarks upon the collaboration
between technical and academic staff, describing how it presented ‘a new set of
challenges’ and required ‘a great deal of flexibility on both sides’ (119-120).

In conclusion, the Collation Editor involves an iterative process aimed
at improving the collation alignment according to specific user needs. Smith’s
approach seems to address a remark in the documentation of CollateX
regarding the analysis step in the Gothenburg model (GM):

An additional (possibly manual) analysis of the alignment result
therefore may alleviate that deficiency by introducing the possibility of a
feedback cycle, in which users edit the alignment and feed their

96 Unfortunately neither Smith and Houghton’s publication nor the project’s documentation
have provided images of this interface.
knowledge back into the alignment process for another run delivering enhanced results (CollateX documentation, online).

When thinking of the steps of the GM, we may ask ourselves where this step would fit in. In the original GM the analysis phase primarily involves computational adjustments to improve the alignment and the treatment of transpositions, whereas the Collation Editor provides an interface for a round of human intervention. Because essentially the process is a feedback loop based on the editor’s preferences, the collation may be described as ‘semi-automated’.

2.2.6. Reflection

When collating TEI/XML transcriptions with multiple levels of variation, one is bound to run into problems with the overlapping structures of the document and the text. This subchapter has focused on several approaches that offer different ways to work around the problem of overlapping hierarchies. In each case, the involvement of the editor/user is paramount. The editor needs to decide which markup elements to select and pass through the collation pipeline and to formulate a clear research question and attempt to customize the algorithms in such a way that they produce the most optimal alignment. It therefore needs to be defined up front what the ‘most optimal alignment’ is. After an initial analysis of the collation output, the algorithm may be adjusted. As a consequence, the collation algorithms become better attuned to the preferences of the editor and the intricacies of the text. This illustrates an important aspect of how computing can help the scholarly editor to get to know her text even better. It reminds us, as Vanhoutte remarks, that transcription is a form of close reading that teaches the editor/encoder something new about the text (2007, online).

We can draw another parallel between the use of collation software for textual research and interfaces of digital editions. The interface makes certain parts of an edition accessible and helps readers/users to carry out research, whilst simultaneously obscuring other parts. Of course, writing or customizing software is also a form of translating a scholarly argument, an editorial orientation, or a research objective into a series of computer-readable steps. Correspondingly, the use of an existing software program (i.e., a package of someone else’s algorithms) implies that one should at least know how these algorithms operate on text, which functions they perform, how they are parsing and processing the input material, and how to interpret the outcome. In that
line of thought, algorithms may be considered as a representation of a scholarly argument. Consequently, the process of preparing a text for automated collation becomes highly informative: being aware of the operations that algorithms perform on the text and the results they produce provide a novel perspective on text. The result is a valuable form of textual awareness (cf. Van Hulle 2004a).

It is important to keep in mind that the approaches discussed in the present section deal primarily with the preservation of markup elements for reasons of visualization, i.e., for representing the writing layers in collation output as well. A certain amount of preprocessing is required to pass along the designated TEI/XML elements, but the collation software still operates on the level of the text. This seems natural when considering collation as a textual activity. However, the gradual inclusion of markup elements in the collation prompts the following questions: do we prefer to make the collation software ‘TEI-aware’ and only able to process a transcription that is encoded according to a certain kind of TEI schema? Or is it better to develop a collation tool that can process complete XML files (i.e., ‘Native XML Collation’), to cater for each type of input transcription? In the following section of this chapter, I explore the second option.
2.3. Theories and Practices Applied: 

Native XML Collation

'It is not enough to invoke origins and perspectives; it is necessary above all to confront actual problems in genetic studies.’

(Hay 2004, 25)

Introduction

This section discusses the proceedings of an experiment with markup and collation software, which can be seen in the light of digital philology and modelling. The objective of the experiment, carried out with Ronald Haentjens Dekker at the Huygens Institute for the History of the Netherlands, was to collate complete XML files against each other. There are two advantages to this method:

(1) It is no longer necessary to select a certain number of TEI/XML elements that are preserved during the collation, because all markup will be taken into account. That is, the tokenizer creates both text tokens and markup tokens. In other words, the editor has significantly less preprocessing to do and the text thus undergoes less acts of interpretation.

(2) The collation output still includes the complete editorial analysis and thus provides a better overview of the different levels of genesis. Not only does it represent the development of the text across witnesses but also captured is the variation within a single witness.

The program we wrote is called ‘Native XML Collation’, and can be found on GitHub under ‘xml_collation’. We set out to test the code on a relatively small data sample: a set of seven sentences from the genetic dossier of Brulez’ Sheherazade. The selected fragment contains additions and deletions as well as transpositions and a number of overlapping hierarchies (e.g., the sentences are spread over two paragraphs and two sheets in the quarto typescript). Cropped images of the typescripts are pictured below as references (see figures 10, 11a, and 11b). The small size of the data set allows us to manually check whether the collation algorithm has done what we want it to do. Moreover, we experiment

---

97 The term ‘program’ is defined simply as a ‘detailed, step-by-step set of instructions telling the computer exactly what to do’ (Zelle 2012, 2).

98 See https://github.com/bleekere/xml_collation (last accessed May 12, 2017).
with two witnesses only: a collation with \( n \) witnesses is inherently more complex and difficult to verify. For two witnesses, we can still visually check whether the algorithm takes every possible combination of alignment into account. If the algorithm works for two witnesses, the presumption is that it can later be extended for collation with \( n \) witnesses.

Figure 10. Page 1r of witness A, the folio typescript of the frame story (AMVC Letterenhuis B 917 / H 2 bis)

Figure 11a. Page 6r of witness B, the quarto typescript of the frame story (AMVC Letterenhuis B 917 / H 2 bis)

Figure 11b. Page 7r of the quarto typescript of the frame story (AMVC Letterenhuis B 917 / H 2 bis)
The workings of the algorithm are described on the basis of one sentence from the data set. Sentence number 21 offers a good example as both witnesses contain the same corrections in the author's hand: in the folio typescript (witness A) and the quarto typescript (witness B), Brulez crossed out the words ‘werven om’ and added ‘trachten naar’. Furthermore, he made a small addition in the punctuation of the folio typescript (two full stops) that are not in the quarto typescript. Finally, one sentence in the folio typescript forms two sentences in the quarto typescript. Below is the TEI/XML transcription of witness A and witness B.99

Witness A (folio typescript s=021)

```xml
<s n="B917-2bis-B5-ts-fol-Liefde,[021]"
Hoe zoet moet nochtans zijn dit
<lb/>
<del type="crossedOut" rend="deepblack ink" hand="#RB"
resp="#EB">werven om</del>
<add place="marginleft" rend="deepblack ink" hand="#RB"
resp="#EB">trachten naar</add>
een vrouw, de ongewisheid vóór de liefelijke toestemming!
<add place="inline" hand="#RB" rend="deepblack ink"
resp="#EB">..</add>
</s>
```

Witness B (quarto typescript s=021 and s=021|001)

```xml
<s n="B917-2bis-B5-tsq-Liefde,[021]"
Hoe zoet moet nochtans zijn dit
<del rend="deepblack ink" type="crossedOut" hand="#RB"
resp="#EB">werven om</del>
<add place="supralinear" rend="deepblack ink" hand="#RB"
resp="#EB">trachten naar</add>
een
<lb/>vrouw !
</s>
<s n="B917_2bis_B5_tsq_Liefde,[021|001]">Die dagen van
nerveuze verwachting vóór de
<lb/>liefelijke toestemming.
</s>
```

99 The encoding schema is largely based on the Encoding Guidelines of the BDMP, which can be found online on http://uahost.uantwerpen.be/bdmp/ (last accessed May 19, 2017).
First, we formulated the envisioned outcome of the XML collation. In line with the methodology of modelling, we found it helpful to think about the output of a regular text collation with CollateX and subsequently reflect on what we would like to see on top of that. A regular collation requires the selection on one ‘layer’ of the text: the ground layer or the top layer. In the following example, we have selected the ground layer, ignoring the <add> elements, which results in the following alignment table output:

<table>
<thead>
<tr>
<th>Witness A</th>
<th>Witness B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hoe zoet moet</td>
<td>Hoe zoet moet</td>
</tr>
<tr>
<td>nochtans zijn dit</td>
<td>nochtans zijn dit</td>
</tr>
<tr>
<td>werven om een vrouw</td>
<td>werven om een vrouw</td>
</tr>
<tr>
<td>+---------------------------+---------------------------+</td>
<td></td>
</tr>
<tr>
<td>, de ongewisheid</td>
<td>! Die dagen van</td>
</tr>
<tr>
<td></td>
<td>nerveuze verwachting</td>
</tr>
<tr>
<td>+---------------------------+---------------------------+</td>
<td></td>
</tr>
<tr>
<td>vóór de liefelijke</td>
<td>vóór de liefelijke</td>
</tr>
<tr>
<td>toestemming</td>
<td>toestemming</td>
</tr>
<tr>
<td>+---------------------------+---------------------------+</td>
<td></td>
</tr>
<tr>
<td>!</td>
<td></td>
</tr>
</tbody>
</table>

What this specific output lacks, is the information that the words ‘werven om’ are crossed out and corrected into ‘trachten naar’ in both witnesses, and that sentence 020 in witness A makes up two sentences in witness B. However, the XML/TEI transcription above contains more useful information. The attributes and the attribute values of the elements <add> and <del> hold information about the author’s hand and the writing material, the place of the additions, and the kind of deletion. There are line breaks and sentence numbers. What is more (but not yet visible in the snippet of the XML/TEI transcription above) the sentences are embedded in a <p> element, which is a child of a <div> element, which is a child of the <text> element, etc. Depending on the edition’s objectives, some of this information is of greater significance for the collation process. For a genetic edition, the notes with XML annotations about who transcribed what may be of less importance. The same goes for additional markup such as linguistic information (e.g., <pos type="noun"> and <pos type="verb">).
If you remove the XML markup of a genetically encoded text (i.e., elements such as `<del>` and `<add>`), the resulting text can become confusing. This happens either when you remove only the TEI/XML elements or the text inside them as well. If we look at our example sentence, it would read either:

Hoe zoet moet nochtans zijn dit werven om trachten naar een vrouw

or:

Hoe zoet moet nochtans zijn dit [...] een vrouw

Neither is ideal. We can call this type of markup ‘semantic’ or ‘text-inherent’, although perhaps ‘hermeneutic markup’ is better, because the elements and the text they contain are part of the text’s meaning and content. Nevertheless, it also holds for presentational markup: one would wish to see whether or not an underlined segment in witness A is also underlined in witness B. Although removing presentation markup results in a text that is still readable, the omitted material is also part of the text’s meaning. In the end, it is difficult to maintain a distinction between different types of markup and how best to treat them. In other words, a collation tool should not distinguish between types of markup but, rather, would process the entire TEI/XML transcription. Such an approach has several implications: we need to decide whether the document structure or the text structure is leading; whether attributes and their values also influence the alignment; and we need to create an informative visualization of the output. In the paragraphs that follow, I discuss the collation of XML trees from a conceptual and a practical perspective.

2.3.1. Native XML Collation: Theory

We start with a number of (more or less) givens. Genetic editors aim to model the multidimensionality of the manuscript page and the genesis of the text by encoding the added, deleted and corrected text. This method results in complex TEI/XML transcriptions in which the different stages in the writing process are encoded as layers of text. The editor needs to prepare the files for collation by either selecting one textual layer; passing along certain markup tags; or creating temporary witnesses based on revisions. As it is primarily a project-specific choice, there is no single correct method. Nevertheless, for a genetic editor even
the usual selection of a base text for collation purposes is a challenge: genetic editors would need to choose from the ground layer, the top layer, or some in-between layer. The ideal collation tool would therefore not require the editor to distinguish different types of markup, nor choose between the ground and the top layer of the text.

Finally, we also need to keep in mind that the linear order of XML tags in the transcription is the result of the editor’s interpretation of the manuscript. To be more precise: the difference between `<add><del>werven om</del></add>` and `<del><add>werven om</add></del>` is meaningful. In some cases, for instance when the author has manually corrected his typescript, the order in which the corrections are made is clear enough. Yet with complex manuscripts this order is not always evident and, thus, dependent on editorial interpretation. As discussed above, the scholarly editing community has generally accepted that the acts of transcribing and encoding are interpretative and that they linearize the complexity of a manuscript page. This linearization needs to be simplified further for collation tools, because they currently require as input strings of tokens. By using what we describe as ‘Native XML Collation’, Dekker and I hope to do away with the need for such simplified linearization. In short, this approach entails the collation of complete XML trees.

In the previous section, we saw that some collation tools allow for the preservation of a number of XML elements. These are given as a property to the token and can, if desired, influence the alignment of the tokens. These approaches work well and produce adequate results, as long as the editors are fine with doing some technical work in the preprocessing phase. Our concept of native XML collation implies that the complete TEI/XML files are collated, including every XML element and text character. This means that the editor does not have to choose which specific XML elements are given as properties to the tokens and which are ignored, nor does she have to preprocess the XML.

100 Collating modern manuscripts presents the same problem as with regard to their transcription: in order to make a modern manuscript readable, its topographic complexity is reduced to a textual model. In other words, the multidimensional image on the page is reduced to a linear text. This problem of linearization has been the topic of long debates in French ‘critique génétique’ (Grésillon 1994; Lebrave 1992; Van Hulle 1999, 2004a; Ferrer 2011). Dahlström notes that ‘obviously, linear sequentiality is a subjectively experienced textual phenomenon, in the sense that it is there for anyone who chooses to see it typographically, but so is retinal spatiality for anyone choosing to see the page of text as a graph’ (Dahlström 2002, n10).
files. Conversely, it still requires a considerable amount of editorial input. The steps taken are described in the paragraphs below.

2.3.2. Native XML Collation: Practice

‘Xml_collation’ is a collation program written in Python, and it has three main functions that (1) create text tokens and element tokens; (2) collate these tokens in a so-called ‘superwitness’; (3) convert the superwitness into an XML tree. The workings of each function are outlined below.

(1) **Tokenizer**
XML files have a tree structure, so in order to extract the tokens from the transcription file we need to walk over or ‘traverse’ the tree. This is done with a so-called ‘XML visitor’: a program that goes across the complete file and takes from it whatever component we specify. In this case, these components are both TEI/XML elements and (text) characters. There is an important difference with regular collation here: the XML elements encountered are considered as tokens in and of themselves; not as properties of the text tokens. Our XML tokens can have attributes with values – since so much information is recorded in the attributes and their values – but the text tokens are just strings of characters.

The Tokenizer first creates an empty list (called ‘tokens’); then parses the witnesses (i.e., the TEI/XML transcriptions). When it encounters a string of characters, it strips the string of any spaces, and adds it to the list ‘tokens’. If it encounters an XML element, it checks whether this is an opening tag or an end-tag and creates an ElementToken. This ElementToken is given the name of the element and, if it is an end tag, a ‘/’ as well. The Tokenizer thus transforms witness A and witness B into two linear sequences of tokens.

(2) **Collate Tokens**
The next step is to collate these sequences and collect the output in a list called ‘superwitness’. A superwitness is a separate pseudo-witness with the most optimal collation result. In that sense, the concept is comparable to the workings of TUSTEP (see chapter 2.2). First, the program builds an edit table\textsuperscript{101} and fills it with the tokens from witness A and witness B. Each cell in this table

\textsuperscript{101} For the concept of an edit table, see for instance the online documentation of CollateX (on http://collatex.net/doc/; last accessed April 12, 2017).
has a row and a column coordinate (x and y), and proposes a possible alignment of two tokens. These coordinates change in two cases: (1) from a cell with matching tokens to a cell with non-matching tokens; or (2) from a non-matching cell to a matching cell. The similar process is used when the tokens are grouped into segments: a new segment occurs with the change from match to non-match, and conversely from non-match to match. Evidently the program must find the largest number of matching tokens/segments, because that is the optimal alignment of the witnesses (see figure 12 below). From a conceptual perspective, this means that the program has a certain definition of ‘match’. This will be discussed in more detail later.

For now, we take the definition of a match as a given and examine the way in which the current program assesses the optimal alignment. The coordinates of the upper left cell of the table are 0,0. The program traverses the table diagonally (to make sure to take every possible alignment into account) and scores each cell based on its coordinates vis-à-vis the start position. This is called the ‘global score’. When the lower right cell is reached, the program knows the global score of each possible alignment. Then, the program starts in the lower right cell and tries to find the path with the lowest global score back to the upper left cell. This path with the lowest global score and thus the least edit operations, has the longest sequence of aligned cells and thus the most optimal alignment of the two witnesses. This is the path indicated in black in figure 12 below:

![Figure 12](image-url)
(3) Convert Superwitness in XML Tree

The sequence with the optimal alignment is what we call the ‘superwitness’. The superwitness thus is a list of the collated tokens of witness A and witness B. As mentioned above, the program knows of each individual token whether it is added, omitted, or a match, and whether it is a TextToken or an ElementToken. Each token in the superwitness is now given a so-called ‘wrapper’ that holds this information, called ‘ExtendedToken’. In short, we have a collation result that contains a considerable amount of information about the text of the input witnesses as well as the editor’s analysis. This information can be used to create a detailed collation result that is useful for genetic editing.

Now we need to process the superwitness, because in its current form it is too abstract to be of any use. We construct an XML tree based on the superwitness and thus contains both text nodes and element nodes. The function that constructs the XML tree assumes that every XML element is an element node, and that the text within the XML elements is a text node. First, an empty root node is set. Then the program unwraps the first ExtendedToken of the superwitness and looks if the token is a TextToken or an ElementToken (in case of an ElementToken: whether it is an opening tag or an end tag). When the extended token is a TextToken, a new text node is created and appended to the XML tree as a child to the latest node. This node has the content of the TextToken (i.e., a character string). For invariant text nodes no additional steps are taken. If the TextToken is an addition or an omission, the program introduces a new node in the tree with the attribute ‘\texttt{cx: added text}’ or ‘\texttt{cx: omitted text}’. The ‘\texttt{cx}’ here refers to a CollateX namespace that is created to differentiate between the tokens in the superwitness and the original witnesses. Then the content of the TextToken is appended as a child to this newly introduced element node.

If the token in question is an ElementToken and an opening tag, the program creates a new element node and appends it as a child to the latest node in the XML tree. The node is given the element name of the token. Again, for invariant element nodes no additional steps are taken. If the ElementToken is an addition or omission, the node gets an attribute as well (respectively \texttt{cx:addition} or \texttt{cx:omission}). Because witness B is collated against witness A, ‘addition’ refers to a token that occurs in witness B but not in witness A. For example, the element node \texttt{<del \texttt{CX:}addition> means that there is a \texttt{<del}>
element in witness B and that it does not occur on a similar position in witness A.

The XML-tree of the superwitness of the example sentence used above is as follows:

```xml
<root>
  <TEI>
    <body>
      <text>
        <div>
          <p>
            <s>
              Hoe zoet moet nochtans zijn dit
              <lb CX="omission"/>
              <del>werven om</del>
              <add>trachten naar</add>
              een
              <lb CX="addition"/>
              vrouw
              <CX="omitted text">, de ongewisheid</CX>
              <CX=added text>!</CX>
            </s>
          </p>
        </div>
      </text>
    </body>
  </TEI>
</root>
```

A number of things become clear when we examine this tree. To begin with, we can now see that in both witnesses the deletion of ‘werven om’ and the addition of ‘trachten naar’ occur: the element nodes and the text nodes do not have a ‘cx’ attribute. It takes more thorough examination to see that the text
tokens ‘, de ongewisheid’ are omitted from witness B, and that they are present in witness A. In other words, it is difficult to extract the information that this XML-tree holds about, for instance, the textual layers. Nevertheless, the tree does contain this information, which suggests that the difficulty is a matter of output visualization. (A suggestion for this visualization is discussed in more detail below.) Finally, the partition of the sentence is not rendered correctly. The program does detect the insertion of several element and text tokens (<s>!
Die dagen van nerveuze verwachting</s>), but the place where they are inserted is not correct:

```
... vrouw
<vrouw>
<omitted text>, de ongewisheid</omitted text>
<added text>!</added text>
</s>
<s CX="addition">
<Die dagen van nerveuze verwachting</Die dagen van nerveuze verwachting>
<vóór de
<liefelijke toestemming
<omitted text>!</omitted text>
<added text>..</added text>
</add>
<added text>.
</s>
```

The second <s> element is placed only after the closing tag </s> (highlighted), whereas in fact we expect it to be nested within the first <s> element. In general, the nesting of elements and the related issue of overlapping hierarchies is a complex problem and one that cannot be easily resolved. The problem primarily concerns the topics of alignment and visualization.

2.3.3. Reflection

Alignment

The Native XML Collation program assumes a definition of match and uses that to come to an alignment. This definition is significant: it is the basis for the division of tokens into segments, constructing the edit graph, and deducing the best possible superwitness. In theory, the editor/user can decide what influences
the process of alignment. In our case of Native XML collation, this could include (but is not limited to) the XML attributes and their values. As it is currently written, the xml_collation program ignores these attributes, which means that a `<del type="crossedOut">` is aligned with a `<del type="instant correction">`. Since the different values of these attributes contribute to the genetic analysis of the text, we might want to take them into account as well.

This leads to a bigger question: whereas the initial idea of Native XML Collation is to be independent of the type of edition (genetic, critical, etc.), the value that is accorded to specific elements and their attributes is project-specific after all. A major factor here is that the TEI is not a standard nor is it machine-oriented, so it is difficult — if not impossible — to extract generic values. It is therefore advisable to consider the TEI (and thus the TEI/XML transcriptions) ‘as produced by a close group of people that adhere to a very strict set of guidelines created with machine consumption in mind’.\textsuperscript{102} It remains to be seen how ‘close’ this group of people are in practice. Are we talking about the team of an edition project or, more generally, about genetic editors? Considering the freedom that the TEI gives editors to formulate their own attribute values, all the evidence points to the former.

\textit{Visualization}

The collation output is currently an XML tree that consists of text nodes and element nodes and, in the CollateX namespace, supplementary nodes that indicate variation. So how can we show the collation output in such a way that the result is meaningful for an end-user? At the moment, it is best to define this ‘user’ as a technical savvy scholarly editor who is sufficiently skilled to interpret the XML tree output, and possibly to transform it into a more conventional format. As discussed above, the additional value of Native XML Collation becomes apparent in the case of TEI/XML transcriptions that contain several layers of revision, because it releases the editor from the need to define and extract a single textual base layer. The XML tree can be converted into a variant graph or perhaps an alignment table (although the latter can display significantly less information; see chapter 3.2.2). On the one hand, a suitable GUI in which the user can indicate which elements, attributes, and attribute values are significant, could solve this issue. On the other hand, it might be a matter of

\textsuperscript{102} Gioele Banabucci, in an email to the author, dd. February 5th, 2016.
developing some stable yet general components so they can be incorporated in the infrastructure of an existing digital edition. This dichotomy is inherent to CollateX as well: CollateX offers an algorithm to produce a critical apparatus, but it is also a software component that can be integrated into a digital edition.\(^\text{103}\) We focused primarily on the first component: the algorithm. Moving forward, we can choose to further develop and refine the algorithm, or to shift our focus and create a veritable software component. Both are valid options, but disparate enough that we have to choose between them.

We have to keep in mind, though, that creating a GUI is a different challenge altogether. Moreover, if the program manages to produce a logical and correctly structured XML file, a number of end users would not even require an interface: the would be able to transform and visualize the XML output themselves. Accordingly, the solution to visualizing could be the parallel segmentation method.\(^\text{104}\) This method introduces sigla instead of ‘addition’ and ‘omission’. As said, witness B is collated against witness A, so ‘addition’ refers to a token that occurs in witness B but not in witness A. It follows that we can use the attribute ‘witA’ instead of ‘addition’, and ‘witB’ instead of ‘omission’. If no variation occurs, the sigla of both witnesses are given. Conform the TEI parallel segmentation method, the variant readings are placed within an \(<\text{app}>\). The \(<\text{app}>\) element and the appropriate \(<\text{rdg}>\) elements can be given an affix of the CX-namespace as well, to underline that they are not part of the original witness text. An example of such an XML tree would look as follows:

```
<root>
  <TEI CX="witA witB">
    <body CX="witA witB">
      <text CX="witA witB">
        <div CX="witA witB">
          <p CX="witA witB ">Hoe zoet moet nochtans zijn dit
            <lb CX="witA"/>
            <del CX="witA witB">werven om</del>
            <add CX="witA witB">trachten naar</add>
            een
            <lb CX="witA"/>
        </p>
      </text>
    </body>
  </TEI>
</root>
```

\(^\text{103}\) See also the project page of CollateX on https://www.huygens.knaw.nl/collate-x-an-automatic-text-alignment-tool/ (last accessed May 19, 2017).

\(^\text{104}\) I am very grateful to Vincent Neyt for his suggestions and insightful comments regarding this issue.
It is technically possible to transform this output, for instance with XSLT and Python, and to visualize it in the same way as a regular parallel segmentation encoding. An additional advantage is that the sigla could allow for a collation with n witnesses, because we no longer base the collation on ‘addition’ and ‘omission’. This, however, remains to be seen: collation with n witnesses presents complications on a wholly different level.

Incidentally, this approach does not solve the problem with the nested `<s>` element. It may be that the solution for that issue lies in the graph as a data model. Thus far, the concept of graphs has been little explored in textual studies. As Eide notes, ‘there has been an undercurrent of experiments and theoretical research into other types of solutions’ (2014). The most advanced research into graphs as data models for textual research is currently being undertaken by Haentjens Dekker and his team at the Huygens ING (working title ‘Text As Graph’ or TAG in short), but at the time of writing no official publication is yet available. A comprehensive analysis of the TAG model is beyond the scope of this chapter, but the concept of TAG could provide a solution to deal with the overlapping hierarchies in the collation output. The
main argument to use a text graph is that in XML the information has to be stored in a hierarchical context, while a graph model has no such hierarchy. I will return to this issue in more detail in chapter 4.3.

Furthermore, we can focus on improving the way the Native XML Collation deals with transpositions. A transposition can be understood as a deletion and addition and this is presently how the transpositions in our dataset are treated. In the following (simplified) witnesses of sentence 0024 in Sheherazade, witness B contains an intradocumentary transposition, but when the two witnesses are collated another transposition appears as an interdocumentary change.

Witness A:
<s n="B917-2bis-tsg,[0024]">Een ambtenaar-agronoom leerde hem juist <del type="instantCorrection">grx</del>agronoom leerde het verschil onderscheiden</del> <lb/>tarwe onderscheiden van rogge en haver

and

Witness B:
<s n="B917-2bis-tsfol,[0024]">Een ambtenaar-agronoom leerde hem juist <del type="crossedOut">het verschil</del><seg type="transposition" target="trans1-ts-fol-01r">onderscheiden</seg> van</del> <lb/>rogge, tarwe en haver <anchor xml:id="trans1-ts-fol-01r"/>

If we input these witnesses into the xml_collation program, it outputs the following, complicated XML tree:

...<s>
<lb CX="omission"/>
Een ambtenaar-agronoom leerde hem juist het verschil
<CX="added text">onderscheiden</CX>
</del>
<seg CX="omission">
<CX="omitted text">onderscheiden</CX>
</seg>
Even though the visualization of the outputted XML tree is not part of the experiment, it is valuable in itself to reflect upon the various ways in which the output can be represented. If we consider the problem with the nested `<s>` element, a number of questions arise: how do we visualize this difference in the structure of the text? Do we visualize the text graphs and their attributes? Is it possible to develop a – more or less – standard visualization of the XML tree such as a variant graph, or is a more project-specific methodology indeed unavoidable? The Native XML Collation demonstrates how digital modelling allows us to get to know the text as well as the TEI/XML transcription in great detail. It confronts us with novel yet important questions for text modelling. These and other issues are the topic of a second experiment with collation software, again under the supervision of Dekker (discussed further in chapter 4.3 ‘Two-Step Alignment’). The main focus of that research is on the two different perspectives on the desired output of Native XML Collation, one of which prioritizes the textual content; the other leads with the document structure.
2.4. Concluding Remarks

‘Whenever we ask what new technology can do for textual scholars, we must not lose sight of a deeper question: what is at stake in the work textual scholarship does, digitally and otherwise? What makes this work worth doing?’

(Galey 2010, 100)

This chapter first outlined the basic principles of automated collation and discussed three key approaches to semi-automated collation with genetically encoded text: plain text collation; markup to plain text conversion; and collation that retains certain markup tags. Each approach deals with the familiar problem of conflicting hierarchies – a fundamental issue for genetic text collation and, indeed, for processing digital texts in general. When making a TEI/XML transcription of manuscript material, the editor adds a hierarchy to the transcription, distinguishing between text and document. Generally, the textual hierarchy is leading, because the TEI is principally text-oriented. As the physical features of a document, such as the lay out, are important for textual genetic research, the WG-GE proposes guidelines to encode a manuscript from a document perspective as well. How this translates into editorial practice depends on the preferences and needs of a given project team. Editors position themselves somewhere on the continuum between text and document and stick to this choice in order to produce consistent TEI/XML transcriptions.

Other solutions may be found in the use of standoff markup or standoff properties; or in encoding everything twice (cf. the practice of the digital Faust edition who encoded their material text from a textual perspective as well as a document perspective; see Dillen 2015, 75 and chapter 4.2). It also implies, though, that collation software needs to be quite flexible in order to cater for the needs of different projects; to be able to process a variety of files with overlapping structures that cannot be represented simultaneously. The complication with collating texts with genetic markup is that, while the genetic orientation is interested in both text and document, the collation always takes place on a textual level: we collate text, or layers within that text. Therefore, I examined the practices that present an innovative approach towards collating with both text and markup elements, and that thus pave the road for further experiments with markup-aware collation.

Another important point that emerged in the chapter concerns the presence of the editor in the process of collation. Dekker et al. (2014) already mention how early collation software required editors to ‘manually calibrate’ the
input texts to fit the then quite basic collation algorithms, but we have advanced significantly since then. Nevertheless, the state-of-the-art collation software still requires the involvement of the editor. Regardless of the collation method, the input files need to be preprocessed to make sure that the software program handles them correctly. This preprocessing includes the selection of one textual layer, or the generation of multiple witnesses based on the internal variation in one witness. It could also imply that, in the method proposed by Schmidt et al., text and markup alike are converted to plain text and collated on a character level. In a technically more advanced approach, the preprocessing of a file amounts to transforming the file into JSON and attributing designated markup elements to the properties of the JSON tokens, so that they can be passed along with the text and visualized in the output. With more complex input text, chances are that the output is not immediately correct. Therefore, the option exists to analyse the collation output – namely the alignment – in the fourth step of the collation workflow (‘analysis’) and make some adjustments, for example by modifying the regularization or grouping together tokens from the same writing layer within a TEI/XML element. The tokens can then be collated again, hopefully resulting in a better alignment and as such contributing to a refinement of the genetic analysis. This strategy is called postprocessing and is applied in Smith’s Collation Editor. In any case, preprocessing text for collation purposes means removing layers of interpretation and analysis carefully added in the transcription phase. The collation process itself, and the potential influencing of the collation output in the postprocessing phase is – again – adding one or more layers of interpretation.

It follows that this approach to and use of collation software is project-specific. The collation input – the transcriptions – are affected by the editor’s positioning regarding text or document; and the implementation of the collation software consists of a number of technological workaround strategies that have a clear function within the edition in question, but may not necessarily be effective for other projects. In short, we cannot speak of a ‘right’ or a ‘wrong’ approach to automated collation because the practice is influenced by the editorial orientation and the idiosyncrasies of the text. Nevertheless, such

105 The CMG experimented with this approach in 2015, and is currently working together with developers at the Huygens ING to expand upon the preliminary findings. The intermittent results of this collaborative research were presented in the aforementioned paper at the joint ESTS/DiXiT conference 2016 (Bleeker et al. 2016).
individual practices tackle original challenges. Scholars are now compelled to think about the hierarchies present in the input files and question how they are processed, how their treatment affects the output, whether the hierarchies are preserved in the output, etc. This way, they are still highly relevant for general research objectives, especially if they build upon existing open-source collation software such as CollateX.

In the final part of this chapter, I explored the possibilities of Native XML Collation, that is, the collation of two entire XML files. It is possible that this approach brings us closer to a solution to the current issues of automated collation of TEI/XML files with multiple textual layers. Ideally, the editor/user no longer has to select one specific writing layer, the TEI/XML transcription does not have to be further linearized, and the multidimensionality of the manuscript page can be better preserved. Because – theoretically – all XML tags can remain in the witness, this approach also means that the additional information about the text’s genesis is also preserved in the collation output. In short, Native XML Collation offers a number of promising solutions for the automated collation of manuscript materials. It also offers a number of original challenges, inter alia an overlapping hierarchy in the collation output. More research is required to determine the adequacy and effectiveness of this approach. Accordingly, a second experiment concentrates on these preliminary findings (chapter 4.3). A last finding to emerge from the experiment is the value of visualization of the collation output of the Native XML Collation. Accordingly, it sets the stage for the next chapter.
Chapter 3 - Visualizing Textual Variance

3.1. Preamble

'Multiple layers are conveyed in unison / each retaining its distinct identity, while contributing to the whole ... Meaning is thus conveyed not only by what is depicted, but through structure: the size, shape, placement, and relationship of components.'

(Sousanis 2015, 65-6)

The power and effectiveness of visual communication have long been recognized by marketers and those working in the advertisement industry. Following the maxim that a picture is worth a thousand words, they use visual styles to reach a wider audience, to attract more buyers, or to better express an argument (e.g., why a certain brand of shampoo is better than another). The study of draft manuscripts teaches us that human beings frequently resort to visual aids to shape their thoughts and support reflection. Authors make a schematic overview to ponder over a plotline (figure 13); pedagogical handbooks use diagrams and drawings to make mathematical questions clear; PhD students sketch mind maps to organize their thoughts; digital humanists employ data visualization tools to discover patterns in 'big data', etc.

Figure 13. Note 3. Schematic overview of the plot of Sheherazade (AMVC Letterenhuis B917/H2a)
Visualization thus functions as a cognitive aid, as an instrument of research and study. Visualizing something is a way to manipulate and play with information; consequently it may generate different research questions and result in new findings or hypotheses. Examples from academia are plentiful: the study of authorship attribution by using scatter grams;\(^{106}\) the use of timelines in historical research; or the use of a network visualization tool to discover relations between correspondents.\(^{107}\)

As the general concept of visualization is large and all-encompassing, so it is necessary to keep this chapter’s discussion within certain boundaries. I will examine the process of digitally visualizing the electronic transcriptions and automated collation output. In particular, I will study in what ways a schematic presentation of information that can be abstracted and extracted from a digital file like a TEI/XML transcription. ‘Visualization’ can thus be considered an act of modelling as well as a model (as the result of the act). Bringing together the topics of transcription and collation discussed in Chapter 2, the present chapter focuses on the digital visualization of the textual variance as it is recorded in transcriptions or mapped through automated collation. The specific objective of the chapter is to find out how digital visualization can be used as a research instrument for textual genetic studies. I explore this question from the perspective of the scholarly editor who uses the technology for information modelling and for mediating between text and reader/user.

Digital visualization is sometimes regarded as an afterthought in humanities research or even, according to Martyn Jessop, considered with a certain suspicion (2008b, 47). Some consider it a mere technical undertaking, an irksome habit of some digital humanists who recently learned to work with a ‘flashy tool’\(^{108}\). However, as outlined above, visualization can be a cognitive process – or promote a cognitive process – and as such become an instrument

---

\(^{106}\) A ‘scatter gram’ or a ‘scatter plot’ is a graphical diagram used to show, for instance, the correlation of function words over a large period of time, helping to identify an author.

\(^{107}\) An interesting example is offered by Richard Hadden’s ongoing PhD research *Digital Textual Scholarship and Digitised Collections* (University of Maynooth), in which he examines the correspondence network as it is formed by the ‘Letters of 1916’, a crowd-sourced digital collection of letters written between November 1915 and October 1916. See [http://letters1916.maynoothuniversity.ie/](http://letters1916.maynoothuniversity.ie/) (last accessed May 20, 2017).

\(^{108}\) For some examples of particularly flashy data visualization see the Scientific Visualization Gallery of the Texas Advanced Computing Centre (TACC) on [https://www.tacc.utexas.edu/scientificvisualization-gallery](https://www.tacc.utexas.edu/scientificvisualization-gallery) (last accessed April 1, 2017). I can make no claim whatsoever about the underlying data and the usefulness of the visualizations.
for study. Elsewhere, Jessop argues that certain digital visualization tools have ‘the ability ... to allow visual perception to be used in the creation or discovery of new knowledge’ (Jessop 2008a, 282). He proposes to consider visualizing as a scholarly activity, with the consequence that the resulting digital visualization constitutes a scholarly product. This implies a subtle yet important contrast with a digital visualization of a scholarly product. In order to be considered a scholarly activity, asserts Jessop, the digital visualization should be the main rhetorical device or at least be equally important as the main rhetorical device. In contrast to an illustration that is merely supportive to the text, a digital visualization is the ‘prime carrier of information’ (2008a, 283). Following Jessop’s argument, this thesis assumes the position that digital visualization is, in fact, an essential part of the digital editing process. It is not the final step to be taken only after the ‘real’ editing is done, but it is a valuable form of information modelling. By highlighting a number of essential aspects of the digital visualization of textual variance, this introduction intends to provide a background and context for a discussion about current practices.

The concept of visualization has gained prominence in the scholarly editing community, with many scholars arguing that the digital medium provides novel ways to represent the dynamic development of a work (e.g., D’Iorio 2010, 49; Gabler 2010, 51; McGann 2013, 275). The medium’s capacity to store and provide (online) access to large amounts of texts gives way to new debates about how to analyse this data without becoming overwhelmed by it. To date, there has been little agreement on the design principles for visualizing a text’s genesis. Nevertheless, we can assume that an adequate visualization of a transcription shows the various aspects of genesis by allowing readers/users to switch between ‘textual viewpoints’, e.g., from microlevel to macrolevel and vice versa. Moreover, readers/users would have the possibility to see the connections and relationships between witnesses of text and perhaps establish new correspondences by themselves. In short, a digital visualization would reflect the nature of the work as a dynamic, evolving entity. Finally, we can take some inspiration from the principles of interface design for large digital collations, such as the rich prospect browsing of Stan Ruecker et al. (2011). The latter suggest that on the primary webpage the users need to see all information available to them, understand how they can call it or manipulate it, and be in full control of the display and manipulation tools (2011, 3-4). If we keep in mind the character of a genetic dossier, it seems desirable to have an interactive
visualization that allows readers/users to organize the documents on a structural level before accessing them.

**Distant Reading of Manuscripts**

Having defined these general conditions, it is striking how they correspond to the theory behind close reading and distant reading. The concept of distant reading, made famous by Franco Moretti, proposes to visualize texts as trees, graphs and maps and to use these visualizations for quantitative analysis such as text mining analytics (2005). His theories caused some commotion in the humanities community: some literary scholars did not consider ‘big data analysis’ as their core business or saw quantified research as a devaluation of their trade. However, a combination of close and distant reading methods does have interesting potential for literary scholars. A visual overview, for example, could alert readers/users to a potentially interesting text that can be further studied with close reading techniques. This prompts the following question: can a digital genetic edition profit in similar ways from theories on close reading and distant reading?

Clearly (the creation of) a scholarly edition provides ample means for close reading. I already mentioned that Vanhoutte considers transcribing ‘the closest kind of reading one can endeavour’ (chapter 2.2.6; Vanhoutte 2007, online). Similarly, Dillen observes that ‘transcribing a text with markup can be considered an extreme form of close reading’ (2015, 235). It appears that we can also draw some parallels between the concept of distant reading and genetic editing. In chapter 2.2.3, for example, I pointed out that the underlying data structure of a TEI/XML transcription is a tree, but if we open it in an oXygen or jEdit editor we see a serialized rendering of this tree, displaying the linear sequence of the characters and TEI elements. Only in the left panel, you can see how the transcription unfolds like a tree. Furthermore, I described that a textual witness is taken as a graph in the memory of the CollateX, and that the concept of a graph structure may well be suited to avert issues of overlapping hierarchies. In other words, the chapter advanced the idea to consider text as a tree and as a graph. One may object that Moretti’s idea of distant reading refers to something slightly different – more visual and less focused on the underlying data structures – but the idea remains the same: it is about different ways of

---

109 For an overview of some responses to Moretti’s theory, see Goodwin and Holbo 2011.
considering the concept of text, and how these different ways of looking and processing text can advance research because they contribute to our understanding of the literary phenomenon.

As mentioned above, much of the current literature on distant reading investigated and confirmed that combining close and distant reading techniques is promising.\textsuperscript{110} A good example is provided by Katherine Coles and Julie Lein, both literary researchers with a focus on poetry. They describe how the visualization that comes with distant reading helps us to see texts as large complex systems as well as on their most granular levels (Coles and Lein 2013, online). Their research builds upon what they describe fittingly as

an urgent desire in the literary community to embrace and explore the power of computation while at the same time prioritizing and protecting the relationship between literature and human readers (Coles and Lein). Working closely together with visualization specialists who use visualization to identify problems and isolate data, Coles and Lein come up with specific visualizations that emphasize a characteristic of poetry they consider important for further study: the aesthetics. This results in visualizations that do not substitute for close reading, but that prompt readers/users to continue exploring the texts by suggesting potentially interesting parts of the poem. After reaching the evident conclusion that the computer can perform research in ways not humanly possible, the authors reflect on two important points: first, that human-computer interaction proves a proper research instrument, and second, that working together with computer scientists shines new light on the fundamental idea of collaboration.

In conclusion, if we recognize the scholarly editing community’s predisposition to the concept of close reading, we can regard visualization as constituting a bridge between the methodology of both close and distant reading. A combination of these technologies provides a means to study text from different levels of granularity, and serves as a research instrument for textual genetic studies particular to the digital medium. As visualizing is an iterative process of continuous polishing and of refining a concept, it follows that creating a visualization is a modelling method and the resulting visualization is the model (see chapter 1.2). In this chapter, I examine the ways in which visualization performs as a communicative act as well as a cognitive act. Visualization may be one of the most direct means for a scholarly editor to

\textsuperscript{110} For a comprehensive overview see for instance Jänicke et al. 2016.
reach an audience, making it a model primarily oriented towards readers/users. Conversely, the making of a digital visualization is a form of information modelling that can be a useful cognitive exercise. Activities such as plotting an alignment table or constructing a genetic map can be challenging and eye-opening activities (and, incidentally, have been so for decades). In both cases, editors are forced to give physical form to an abstract idea of textual variation that exists at that point only in a TEI/XML transcription and the resulting collation. What features do they select to support their argument? If the visualization is not satisfactory, perhaps they can adjust the transcription or rerun the collation program? As such, visualization constitutes a proper step in the editorial workflow and, by extension, in the process of digital textual research.

**Readers/Users: Finding an Audience**

Before we go on, it is necessary to point to another limitation of this chapter. That is, I will explore the potential of the digital visualization. It is beyond the scope of my research to establish whether the digital visualization works to its full effect, that is, to hold a survey among the readers/users of the selected editions and inquire if the visualization assists them in exploring textual variance on multiple levels.111 As I concentrate on digital visualization, it is inevitable that the concept of the reader/user is discussed, but the main topic of this dissertation remains the scholarly editor. So when talking about the ‘communicative aspect’ of visualization, I refer first and foremost to the editor who plays the part of intermediary. Nevertheless, questions such as ‘are the readers/users capable of using the tools as intended?’ and ‘is this visualization helpful to explore the text’s development?’ are useful and may help scholarly editors develop more effective visualizations. A major obstacle to the study of the effects of visualization, however, is that the background and the research interests of the readers/users of a genetic edition are largely unknown. Does the audience primarily consist of other editors and textual genetic scholars, or is it possible to reach a more diverse audience?

Not too long ago, Charlotte Cailliau noted that the primary audience of a scholarly edition has not been sufficiently studied (Cailliau 2009, 275) and

111 Such work is for instance undertaken by the research group of Jakob Nielsen, although his studies concern webdesign and interfaces in general. With regard to digital scholarly editing, there is little published data on readers/users and their use of editions.
this appears to be the case still. Indeed, while scholarly editors do like to create an edition that meets prevailing user needs and offers new, unexpected advantages, they appear primarily interested in what they can do with their data.\textsuperscript{112} Whether this unilateral interest results in editors staying within the confines of their own discipline, in catering only for the needs of their own community, is a question I do not intend to answer here. Nevertheless, it is reasonable to assume that the main audience of a genetic edition consists of either scholarly editors or other members of the scholarly community who understand concepts like textual variance, authorial revision, and draft texts, and who are primarily interested in tools to further examine those phenomena.\textsuperscript{113} Evidently, this has consequences for the visualization of textual variance: we will see in chapter 3.2 how editors try and find a balance between a user-friendly interface and a comprehensive, scholarly presentation of variants.

Another challenge is the potential bias of visualization. Although its purpose is not to present all the material and tools of the edition, the visualization stems from an editorial argument about the material and supports further investigation in that direction. Visualizing is a means to present the information contained in the edition and direct the attention of a reader/user towards it. As a consequence, it may stand in the way of other information contained in the edition. Keeping in mind that user needs are not always apparent, visualizations that are aimed at one specific aspect could make an edition less useful for other audiences. As Robinson argues, another person’s interface can be your worst enemy (Robinson 2013b). In the case of \textit{Sheherazade}, the material consists of an assembly of draft manuscripts and typescripts with substantial authorial revisions. Since the genetic orientation offers a sound methodology to examine this type of material, the resulting visualization would be adapted accordingly to accommodate textual genetic research. Readers/users have to be able to further explore the documents and their transcriptions, and

\textsuperscript{112} Several studies have discussed the topic of the user of a digital edition (e.g., Dahlström 2006; Vanhoutte 2009; idem 2012; Sutherland 2009; Pierazzo 2015, 161). Often such studies make mention of the difficult equilibrium of catering for different needs (i.e., producing a reading-edition as well as an study-edition).

\textsuperscript{113} Of course, the material may unexpectedly appeal to other fields of research. A (fictional) example is a digital edition of Anne Frank’s writings that is predominantly historically oriented, but turns out to be a treasure throw for linguists interested in language use of bilingual children in the 1930s and 1940s. Since few digital editions implement user statistics (or provide that data), it is difficult to say by whom and how they are used, and whether they are used by the intended audience.
at the same time understand that what they see is an editorial argument about the constitution of the genetic dossier.

So how can digital visualization - the model as well as the modelling - promote research into textual genesis? For the genetic orientation to text two things are most important: first, that each state of the text is considered of equal importance; secondly, that the overall evolution and development of the text as a whole can be followed. Within this framework, a genetic edition facilitates the study of the ‘avant-texte’. Chapter 1 presented a number of digital components to examine the ‘avant-texte’ of a work: a transcription of the documents that constitute the ‘avant-texte’, a collation of the text on these documents, and a study of the various classifications of the genetic dossier as a whole. The first and second components, transcription and collation, both provide a way to record certain aspects of textual genesis. Transcription is especially suited to encode the textual variance on the page, for example with the TEI elements `<del>` and `<add>`, whereas (automated) collation offers a means to study the text’s historical development on different levels. The third component entails rearranging the genetic dossier according to one’s interests, following either a chronological, physical, or genetic classification. Since the third component constitutes the topic of Chapter 4 (‘Organizing Documents and Versions’), what we are looking for in this chapter is an effective way to visualize the transcription and the result of the automated collation. Following the research question of this thesis, we are looking specifically for digital visualization that affects both the methodology and the result of genetic editing. The scholarly value of this digital visualization can furthermore be established by looking at its additional ‘epistemological surplus’ compared to its printed counterpart, and whether it has an exclusive rhetorical value. Finally, we are looking for ways to represent the editor’s analysis of the genesis. That last point suggests that readers/users are presented with tools to study both the ‘genetic’ part and the ‘criticism’ part. Ideally a genetic edition would exhibit the dynamic interplay between the different aspects of genesis: endo-, exo-, epi-, micro- and macrogenesis (see chapter 1.3).

**Outline**

The next section (chapter 3.2 ‘Theories and Practices’) presents a synthesis and evaluation of some editing projects that currently deploy digital visualization to represent and study textual variation. This list is not exhaustive: if I were to
present a comprehensive discussion of editing projects that treat textual variation, the list would be virtually endless because each scholarly edition deals with textual variation in some way. I am primarily interested in those projects that focus on textual variation and understand text as a dynamic network, and I want to find out how this translates to the visualization of their transcriptions and the result of (automated) collation, and how they employ visualization to present that focus and their scholarly argument. Evidently not every project will overtly refer to the editorial ‘thought processes’ behind certain visual representations. The projects are therefore also selected because they are complemented by one or several publications (an editorial rationale in the edition itself or a scholarly contribution elsewhere) in which the editors explain their reasoning and approach, which enables me to reconstruct in some way how visualization constitutes a research instrument for the editors. This has the additional advantage that the interesting discrepancy between theoretical intentions and practical realizations becomes clear.

With regard to transcriptions, I distinguish three approaches to visualize variation. First, by displaying the revision layers in varying levels of granularity. Second, by complementing the transcription with a digital facsimile and consequently defining the role of the transcription as a tool to understand, navigate, and translate the document page. Third, by narrating the revision sites, a technique that suggests that visualization by itself does not always suffice to explain a text’s genesis and that allocates a prominent role to the editor as expert and narrator. The distinction is of course artificial, for most editorial projects combine two or more approaches, giving them a personal twist. Equally diverse is the practice of visualizing the output of automated collation. I already discussed the alignment table and a variant graph, but there exist more experimental visualizations as well (e.g., the ‘Diachronic Slider’ of Gabler and Schäuble). As expected, the presentation of collation results is highly influenced by the edition’s objectives and in turn it affects the future use of the results. In the last section (chapter 3.3 ‘Exhibiting Literary Genesis’), I elaborate on the role of the editor as a mediator between reader/user and text by further exploring the method of the revision narrative and by discussing how digital visualization can be used to reach a larger audience. The development of a digital museum exhibition of Sheherazade’s genesis is used as a practical illustration.
3.2. Theories and Practices

‘One thing a digital visualization can do is make a virtue of ambiguity in ways that print cannot, combining the elements of time and motion to represent variants in ways that challenge the idea that texts are fixed and immutable.’

(Galey, online)

‘[T]he scholarly edition is aimed at articulating the editor’s notions, perspectives, or theories of the texts [...] By doing so, the scholarly edition could invoke the literary debate by providing the materials and tools to explore new ways of understanding and studying the text.’

(Vanhoutte 2004)

Simply put, the present section is about turning code into image: visualizing an encoded text, or representing the collation output. After the arduous tasks of assembling the ‘avant-texte’, analyzing and transcribing the text on the documents, and collating the witnesses, the visualization seems just a matter of ‘presenting it nicely’. However, I argue that visualizing is not the last step: it can also help the editor with her genetic analysis and the construction of her argument. This comes in addition to facilitating the readers/users’ access to the edition.

Keeping in mind the labyrinthine relationships of the documents in the ‘avant-texte’, and the often hard-to-read draft manuscripts, it is easy to imagine that a genetic edition can benefit from visualization. The visual means of the print edition are limited when it comes to genetic editions. Diplomatic transcriptions are provided to help the reader decipher the text but they also intend to capture each particularity of the document in meticulous detail. The resulting transcripts are sometimes as difficult to read as the draft manuscripts they are derived from (Pierazzo 2014, online). Moreover, it is not feasible to reproduce the entire ‘avant-texte’ in a print edition and, apart from genetic plans and diagrams, there are little visual means to showcase the macrogenesis – let alone to represent the network of interdocumentary relationships.

The digital medium, of course, has been hailed as a way to take care of all that. Multiple transcription views, ranging from a reading text to a diplomatic transcription, can be generated from one and the same source file. Various collation output formats help to make the results of a collation more useful. A variety of interfaces can be employed to narrate the story of a work’s origins and development. Yet these new possibilities come with new challenges. In line with the versatility of the digital medium there is no such thing as 'the best' visualization. It is up to the scholarly editor to select or create her own
visualization that corresponds with her orientation to the text and to the argument she wishes to make about its genesis. Like transcription, visualization is a process of selection and rejection. So, like transcription, it is essentially (the making of) a model. The modelling process ensures that it is a useful research activity for a genetic editor who is, once more, pressed to clearly formulate her concept of text and its genesis. In addition, visualization can be employed to represent a text’s genesis in a way that is meaningful to a reader/user who is not familiar with the intricacies of the ‘avant-texte’. Instead of an afterthought of scholarly editing, visualization is, in fact, a complex and rewarding task.

Section 3.2.1 discusses the representations of the epistemic foundations of a scholarly edition: documentary transcriptions. I will describe different ways of handling the continuum between document and text, and of representing the multiple layers of writing that are present in a revised text. I will pay specific attention to the narrative of the genesis, as it has been established that the representation of a work’s genesis thrives by a certain form of narration (Dillen 2015, 92), and that the digital medium offers possibilities to visualize this editorial argument. The various visualizations of the collation output are discussed in the next section (3.2.2).

What we will see throughout the chapter, but in the discussion about automated collation especially, is how a continuous awareness of both the underlying code and the bias of visualization are paramount. Creating a visualization thus gives editors a tool to influence how an edition is used and to determine future research endeavours. They can highlight certain aspects of the text, but equally obscure others that they do not consider important or useful. I therefore take the function of the editor as a point of reference throughout the chapter, discussing both the cognitive aspects of visualizing (modelling) a digital transcription (model), and the transformations of the text from one format to another. Finally, I would like to stress that it is not my intention to juxtapose print and digital methods. Since the aims of editing for the genetic orientation have remained largely the same over the past decades, such a comparison seems out of order. Instead, I examine how the use of the computer can advance the established methodology of close reading, analysis, and criticism – and possibly introduce new (distant) perspectives on genesis.
3.2.1. Visualizing Transcriptions

Fundamentals of Technology

In Chapter 2, a digital transcription was defined as a rich data structure that is essential for a wide range of purposes. If we keep in mind that the goal of all transcriptions is to make a representation of a physical object that is easier to use (Huitfeldt and Sperberg-McQueen, qtd. in Pierazzo 2015), it follows that making a digital transcription is a way of modelling information on a document. Two other truisms are essential to this chapter as well: first, since transcriptions made from a genetic orientation to text store detailed information about textual variance, they fulfill both an archival and a display function. Secondly, the editor needs to make decisions about the output format as early as the transcription phase. This last observation suggests indeed that visualization is not the final step in editing.

The information in transcriptions is processed in different ways, by different tools, and for different objectives. With regard to visualization, attending to the various layers of revision that are often present in a draft manuscript seems to be a logical first step. Not incidentally, one of the most prominent features of digital manuscript editions is the visualization of the actual revisions on the document page: words that have been crossed out, corrected, transposed, or added. To this end, many genetic editions make use of documentary transcriptions in various degrees of rigor. However, making a transcription that contains both the documentary features as well as the textual elements requires some careful consideration.

The main issue is, again, the distinction between text and document and the relative impossibility to properly compute this. As we saw in Chapter 2, the structure of the markup language influences how the document is processed. While the one-parent tree structure of XML only allows for one dimension and a strict hierarchical ordering of the textual elements, the editor of draft manuscripts often deals with a multidimensional content of non-hierarchical elements. The chapter described how making a transcription in TEI/XML entails a flattening of this multidimensionality. The `<delSpan>` element provides another good example of such flattening. `<delSpan>` is a milestone element, used in combination with an `<anchor>` element to encode large
deleted segments of text without creating conflicts with nested elements.\(^{114}\) It does not actually add the extra dimension that is created when the passage was deleted; it only represents the place where the deletion starts and where it ends. This could be visualized as a crossed-out passage, but that is because the associated stylesheet would include a template that presents all text characters between the `<delSpan>` and the `<anchor>` as ‘strike-through’.

It is vital that editors who encode and process XML-based TEI transcriptions remain aware of the base structure of the transcription and how this structure affects future processing. A TEI/XML transcription is a serialized representation of a hierarchical structure of one tree containing diverse information. This is the format in which the text is presented to, and processed by, a computer. When that file is subsequently processed, XSLT and CSS-like technologies employ special stylesheets that render certain aspects of the encoded transcription in HTML or other formats that can, in turn, be processed and visualized by a web browser. In the stylesheet file is indicated which elements to render and how (as well as which elements to ignore). Thus the transcription is a rendering of the source document and the online visualization is in turn a rendering of the transcription (see figure 14). The visualization represents features of the transcription, not of the source document. In other words, the transcription you see in a web browser is at least twice removed from the source. Both renderings are critical, editorial constructs: they highlight certain features and obscure other features.

---

\(^{114}\) According to the TEI Guidelines description of the `<delSpan>` element, it ‘marks the beginning of a longer sequence of text deleted, marked as deleted, or otherwise signaled as superfluous or spurious by an author, scribe, annotator, or corrector’ (Burnard et al., 11.3.1.4).
Working with Digital Facsimiles

Because neither the transcription nor its visualization can substitute for the source document, a digital facsimile is often presented alongside the transcription. The combination of transcription and facsimile gives readers/users a fuller impression of the source document, because there are certain features that are not easily captured in an encoding, as Holbrook and Kline explain:

The character of the handwriting or typewriting can offer clues to the author's alertness or health. Careless penmanship in one recipient's copy of a letter and painstaking inscription of a letter addressed to another can indicate different degrees of formality between the author and the two correspondents. The nature of the paper, ink, or pencil or typewriter ribbon can provide important clues to the time and place of the preparation of an undated letter or journal entry. Many of these important factors in the source's documentary contents cannot be reproduced; they can only be described (Holbrook and Kline, third online edition, 144-5).

The transcription in this case is a map or a 'key' that can be used to navigate and understand the image (Pierazzo 2014, online). Furthermore, providing a digital facsimile alongside a transcription gives readers/users the possibility to review the scholarly work of the editor. Some authors feel that this presentation puts the editor 'on trial' or creates a competition between editor and reader (cf. Sutherland and Pierazzo 2011; Pierazzo 2014). However, it also has also an unmistakable value. It enhances the readers/users’ awareness of the processes of transcription and visual representation of the inscriptions on a document. This is even more the case for editions that also present the underlying encoding. When viewed next to a digital facsimile, a TEI/XML transcription gives an insight into how transcribing is a process of linearizing the layers of writing. As such, it is a visual means of promoting both an understanding of the editor’s choices and an increased awareness of the different ways of seeing a text.

In general, it is felt that that a transcription does not need to visually mimic the exact topographical structure of the document when it is presented

115 Some caution is needed here, because the digital facsimile is a reproduction too. Of course, this is not to say that the end-user should mistrust everything a digital edition offers: these 'critical constructs' are ideally the product of a scholarly thought process.
alongside a digital facsimile.\footnote{Apart from the fact that one cannot model every detail (‘the relationship of form to content is complex and sometimes beyond exhaustive modelling’; Galey 2010, 93), creating a gap between the transcription and the original document could promote a better analysis of the document itself (Pierazzo 2015, 98). Indeed, as the material features of a source document can be inferred from a digital facsimile, the main function of a transcription would be the provision of a legible text.} An exception to this rule comes in the form of digital editions that plot the transcription over the digital facsimile. Especially in the case of complex manuscripts, this method provides a tool for readers/users to decipher the source document, without having to continuously compare a digital facsimile with its adjacent digital transcription. Accordingly, we see this practice with digital editions that contain draft manuscripts, such as the BDMP, the Faust edition, or the Woolf Online project. Yet associating the facsimile and transcription for display purposes does not automatically mean that they are also connected on the level of the code. Some projects make use of static HTML pages to display the transcription-to-image visualization, like the Woolf Online project. In this case, a topographical transcription is provided that is plotted against the background of a digital facsimile of the notebooks of Virginia Woolf’s \textit{To the Lighthouse} (figure 15).

![Figure 15. Selection of a digital facsimile of Woolf's notebook on which a diplomatic transcription is mapped (Berg materials, notebook 1, fol. 13, SD p.6).](image)

The diplomatic transcription containing information on Woolf’s revisions can only be exported as plain text or static HTML. The latter contains dense code that is nearly impossible for humans to read and, more importantly, cannot be reused for other purposes (see figure 16).
Another telling example is the *Cœdmon’s Hymn* project. Initially, the editors of the project had set out to create a digital database of diplomatic transcriptions in TEI/XML, from which different views on the text could be generated automatically and according to the interests of the reader/user (O’Donnell 2010, 110-9). Unfortunately, the technological limitations of the time restricted the dynamic, on the fly conversion (from ‘archive level’ transcription to ‘display level’ HTML visualization) and the editors opted for preprogrammed HTML pages that each present a different view on the text. In total, the relatively short text of *Cœdmon’s Hymn* made for over 90 different individual HTML pages that each presented a different view. Every time readers/users would select a certain view, the related HTML page is loaded into their browser. One of the downsides, O’Donnell remarks, was that

even the [edition’s] designers commonly forgot that the underlying data on each page was not the same: during proof-reading, we were often surprised to discover that corrections made to one preprogrammed view did not automatically show up in other related versions (2010, 188).

In other words, since the HTML and the TEI/XML represent the same text conceptually, project members intuitively assumed they were connected on the level of the code. Having two separate files of one textual witness – the display transcription in HTML and the archival transcription in TEI/XML – turned out to be a significant disadvantage. These observations illustrate the TEI/XML transcription’s dual function of record and display and, moreover, the importance of having an awareness of the workings of the code.

**Visualizing to Advance Genetic Analysis**

When it comes to capturing the material and textual features of the source text in a TEI/XML transcription, most digital editing projects go for ‘a little bit of
both’, with a slight predisposition towards the text. Presenting users with the option of selecting different views on the text – usually giving them the choice between a reading text and a diplomatic transcription – is becoming common practice for most digital editions. Pierazzo describes them as ‘paradigmatic editions’ because they allow the switch between different paradigms of looking at text. The editions provide ‘alternative options for the same string of text in a nonlinear way’ (Pierazzo 2014, online). We already learned that a detailed diplomatic transcription of the text helps to study its endogenesis: it displays the layers of revision and their location on the document. But of course, the digital transcription holds more information. The editors of the Shelley Godwin Archive (S-GA) for instance decided to focus on one particular aspect of the draft notebooks of *Frankenstein*. Since Mary and Percy Shelley worked together on the novel (the former as author, the latter as her editor), the S-GA considered it valuable to display the composition of *Frankenstein* from the perspective of this co-authoring process. In addition to (six) different views on the text, it is possible to visualize only the writings in Mary’s hand or, conversely, only Percy’s revisions (see figure 17). The underlying TEI/XML transcription holds genetic information about the composition process and is therefore a valuable archival record, but the editors of the S-GA visualize only a specific part of it. Because the S-GA does not engage any further with the writings on the page – i.e., the editors do not provide a macroanalysis of the composition nor a genetic narrative – it cannot be truly considered a genetic edition (Neuber 2014, §22; Dillen 2015, 92). Still, it demonstrates clearly how a visualization of the encoding follows the interests of the editor and can be used to direct the gaze of a reader/user.

Accordingly, for the *Cædmon’s Hymn* project, being a paradigmatic edition meant preprogramming over 90 different visualizations of the text of the poem, ranging from different views of individual manuscripts through different types of critical editions.

As my DiXiT colleague Frederike Neuber remarks in her review of the S-GA, it is a shame that the editors did not make more use of the genetic encoding of the manuscripts. By focusing on the co-authoring process only, she asserts, the editors neglect other aspects of the composition process (Neuber 2014, §22).
To illustrate some of the observations above, let’s take a practical example using a sentence from *Wat is liefde zonder verleiding*, the main story of Brulez’ *Sheherazade*. A selection from the digital facsimile (figure 18) shows that it is quite a complex autograph. At the beginning of sentence 0019, the word ‘Inderdaad’ is inserted; a word is crossed out after the word ‘Liefde’, a new word is added above it and crossed out as well. Finally some letters are retraced in order to clarify them.

The TEI/XML transcription in figure 19 shows the information we encoded of this sentence:
We could call this a semi-diplomatic transcription, because the place of the insertion 'Inderdaad' is indicated by approximation (with the attribute @place="supralinear"), and not all the spots and lines on the page are captured. We furthermore refrained from indicating the writing sequence, i.e., the exact order in which the words are written and the revisions are made, because we considered it too much conjecture. It is difficult to say whether Brulez crossed out the word 'helaas' first and the word 'niet' second. Nevertheless, this TEI/XML transcription yields a large amount of information (and incidentally, it makes for no easy reading). We can represent the first layer of writing, the 'ground' layer, by scripting the XSLT stylesheet we use for transforming this file to display the characters in the <del> element and ignore the <add> element and its text content altogether. Or, vice versa, in order to display the top layer of the text. Such a visualization would result in a semi-diplomatic transcript that respects line breaks and indicates the approximated location of the insertion (figure 20):

![Figure 20. A semi-diplomatic visualization of the TEI transcription in figure 19.](image)

The values of the attributes @rend, @place, and @hand supply further information that can be used for display purposes. Since the manuscript is written and revised in Brulez' hand, it makes little sense to display the @hand value, but considering that Brulez often used a red pencil to make corrections in the structure of the story, it would be worthwhile to include the value of the
An additional advantage of visualizing different writing tools is that they implicitly indicate an order in the writing. Thus readers/users are given some additional tools to assist their interpretation and analysis of the text.

Another option is to visualize the top layer text. This makes for a better reading text (see figure 21), but it no longer shows the instances of ‘creative invention’ and hides the traces of Brulez’ composition process.

In this case, displaying only the top layer would leave out the previous sentence 0018 altogether (‘Doch ook zijn haremmeisjes was de sultan stilaan beu geworden’). This goes against Van Hulle’s argumentation that ‘finding ways to draw reader’s attention to the dead ends in the genesis is a crucial aspect of the study of writing processes’ ([forthcoming]). Moreover, although sentence 0018 is indeed crossed out in the manuscript, it does appear in an alternated version in the folio typescript (figure 22):

Non-Dutch readers may be able to spot the identical words ‘Doch’ and ‘beu geworden’ in the manuscript (figure 18) and the typescript (figure 22); for the rest the versions are only semantically related. In fact, three versions of sentence 0018 can be found in the draft manuscripts. They are located in different places and two of them are crossed out entirely. Of course, the correlation between the three versions of sentence 0018 could also become clear in other ways; quite possibly through a collation of both witnesses (see Chapter 2) or by arranging them in a synoptic overview (cf. the BDMP, discussed below). Still, if we would discard the visualization of the cancelled versions of the sentence and only display the top layer of the text, readers/users would not be able to see the

---

119 Note that in the case of such a synoptic overview, the connection between the sentences is already established and encoded by the editors.
correspondence between these separate versions and, moreover, they would miss a part of the sentence’s development.

In Chapter 2, I discussed that it is nearly impossible to establish with certainty the order of writing and revision on complex draft manuscripts. Editors are thus compelled to choose a certain order and hardcode this order in the TEI/XML transcription. The previous chapter already discussed the transcription and encoding method of the ADS edition (chapter 2.2.2). In the visualization of this transcription, the various deleted passages are encoded with <del> elements and displayed as crossed out. The added text is classified into additions, additions within an addition, and addition within an addition within an addition, displayed in blue, red, or green respectively (figure 23).

![Figure 23. Selection of the ADS edition: a visualized transcription of sentence 005 from the manuscript AdsM1, with the various types of additions in blue, red, or green.]

Evidently, scholarly editors have ways to indicate how (un)sure they are about the order, but it remains a challenge to visualize this in an intuitive way. Realistically, the majority of users/readers will skip the editorial rationale and more or less accept the text as it is presented to them. In other words, the user/reader’s conception of the development of the text is (subconsciously) influenced by the edition’s presentation of it.

With that in mind it is interesting to consider Galey’s visualization tool ‘Animated Variants’. His intention was to ‘make a virtue of ambiguity’ by presenting the textual variance within a sentence in an active manner by rotating variant words in a sentence. In his own words, Galey wants to ‘combine the elements of time and motion to represent variants in ways that challenge the idea that texts are fixed and immutable’ (Galey, online). He hopes to make readers/users conscious of the notion of textual transmission by showing that texts constantly change ‘even when the reader is not looking’ (idem). The Visualization Variation project’s website contains detailed account of Galey’s approach. Although the current code is in an experimental phase of prototyping, the site’s documentation outlines in a clear and insightful manner the considerations behind his method and referring to the underlying code.


120
example used on the website is a snippet from a parallel segmentation transcription in TEI/XML of William Shakespeare’s *Hamlet* that contains three variants of the word ‘solid’:

```
<sp><speaker>Hamlet</speaker>
<lb n="129">Oh that this too too
<app>
  <lem wit="F1">solid</lem>
  <rdg wit="Q2">sallied</rdg>
  <rdg type="emendation">sullied</rdg>
</app>
  flesh would melt,
<lb n="130">Thaw, and resolve itself into a dew.
</sp>
```

This ambiguity is subsequently visualized by rotating the words ‘solid’, ‘sallied’, and ‘sullied’ in the online transcription (see figure 24 and figure 25). To be fair, a screenshot alone does this way of presenting variance no real justice, as the point Galey want to make is that no variant is fixed:

![Figure 24](image1.png)

Figure 24. Visualization of the transcription above; screenshot taken at the moment the word 'solid' is displayed.

![Figure 25](image2.png)

Figure 25. The same visualization; screenshot now taken at the moment the word 'sullied' is displayed.

We can infer from the underlying TEI/XML that the present example concerns two witnesses and one lemma, but it is easy to imagine this type of visualization would also work for variants within a witness. A combination of the two would nevertheless require a more advanced visualization that distinguishes between variation within a witness (intradocumentary) and variation between witnesses.
(interdocumentary). Since these types concern different aspects of textual genesis, they call for a visualization that takes this difference into account.

Both the ADS and the Visualizing Variation examples concern visualizations of the writing process and of textual variance at microlevel: Galey indicates that this visualization currently works best for single words or short phrase, and the step-by-step composition of a sentence to the ADS edition also concerns primarily the endogenesis of a text. Nevertheless, as discussed in Chapter 1, we can consider the study of textual genesis as a continuum or a scale, ranging from microlevel to macrolevel and back. This idea of being able to 'zoom in' and 'zoom out' is in line with the close reading and distant reading methods discussed in the previous section. From the visualizations at word level or sentence level, we now move on to larger units of text. An illustration is provided by the prototype of one of Marcel Proust's notebooks created by André and Pierazzo (2012). Here, the unit of text is defined by the moment when it was written. Units therefore vary from one sentence to a few paragraphs. Figure 26 (next page) reproduces how the transcription of each separate unit is plotted on a certain zone of a document page. The transcription includes line breaks and mimics the distribution of the text over the page.

---

121 Pierazzo describes this edition as 'ultra-diplomatic', saying that its format tries to mimic they lay out of the manuscript page as much as possible (Pierazzo 2012).
Despite its small size, the screenshot above gives an idea of how the text is spread over the pages of the notebook that contains marginal and interlinear additions. The editors encoded the notebook’s text from two perspectives: the order in which Proust inscribed the page, and the order in which – in all probability – the text needs to be read. Consequently, readers/users can decide to visualize the transcription of the notebook from two different perspectives: the writing sequence or the reading sequence. The process of writing is also emphasized by means of the dynamic way in which the transcription is plotted over the facsimile: with each mouse click a segment appears in reading or writing order. As is to be expected André and Pierazzo were not always sure about the chronology of writing. They even found that in some cases, the order in which Proust filled his notebook went against the expectations they had based on the published versions of the text (2013, 157n15). These different levels of certainty are conveyed by different shades of yellow: for instance, the dark yellow section in figure 26 implies a high level of uncertainty.

Figure 27 shows a selection from the underlying code of the Proust prototype, illustrating how the editors encoded folio 47r conform the TEI
manuscript encoding guidelines\textsuperscript{122} as a \texttt{<surface>} divided in \texttt{<zone>}s, each with their unique identifier (\texttt{<xml:id>}). Each zone is numbered according to its (hypothetical) chronological order (see also André and Pierazzo 2013, 56).

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{selection}
\caption{Selection of the TEI encoding of page 47r, showing four uniquely identified zones. The coordinates of each zone are required to map it topographically on the digital facsimile. The text transcription is contained in the \texttt{<zone>} element (not visible in this figure).}
\end{figure}

In principle, working with zones and surfaces is simply another way of dividing and grouping text. Instead of the more conventional chapters and paragraphs, the textual features (i.e., the content) and temporality of the writing (i.e., when is the unit of text written) form the basis of the divisions. In other words, the zones consist of a unit of text that can theoretically be considered independently of the document page. The Proust prototype only visualizes the writing and reading sequences on individual openings, but André and Pierazzo do acknowledge the value of having a visual overview of the writing process across the pages of the notebook – what has been described before as interdocumentary variation (André and Pierazzo 2013, 160). Evidently, this could be extended further, to Proust’s other notebooks for example. Nonetheless, the Proust prototype takes important steps towards a more inclusive representation of textual genesis by including the temporal aspects of genesis as well as the dynamicity of the process.

With regard to a dynamic visualization of the writing process across documents borders, a notable experiment has been carried out by my colleagues at the CMG; Dillen, Neyt, and Van Hulle. At present, the BDMP already offers an overview of the development of each sentence in the works in the genetic edition. Readers/users can select a sentence for a so-called ‘Synoptic Sentence View’ of each version of that sentence, with the versions displayed vertically in chronological order. During the transcription phase, the editors of Beckett’s works give each sentence in a work a number. For practical reasons, the

\textsuperscript{122} See chapter 11 and 12 of the TEI Guidelines (P5 version 3.1.0).
numbering is based on the first published text of the work, which functions as a provisional, temporary base text. The editors subsequently implement this numbering into the remaining versions of the work. Readers/users can select one sentence and are then presented with an overview of all its versions.

For the sentences of *L’Innommable*, the BDMP presents an additional, macrogenetic perspective in a visual juxtaposition of the draft manuscript (consisting of two notebooks) and the base text (i.e., the first published text, in this case Editions de Minuit in 1953). Similar to the *Proust* prototype, readers/users can choose to have the sentences appear in the order they are arranged on the page of the manuscript (i.e., the writing order) or in the order they appear in the published text (i.e., the reading order). One of the main intentions of the visualization is to make clear the relationship between the draft manuscript and the published text. The visualization of *L’Innommable*’s genesis shows the work from a distant reading perspective, because it does not show any actual text: sentences are instead represented by little dots or squares in different colours (see figure 28). Clicking on a square brings you back to the aforementioned Synoptic Sentence View, showing each version of the selected sentence. The visualization makes clear when a sentence from the manuscript does not make it into the published text – or vice versa, when a sentence in the published text does not appear in the manuscript.

![Figure 28. Selection of the macrogenetic visualization of the sentences in *L’Innommable*. The right grid shows where the sentence is located in the published text (the base text witness), and the left table shows where the same sentence is located in the manuscript (the notebook witness). The screenshot was taken when sentence 0020 was selected, which is visible as a highlighted white square in the right grid. Since sentence 0020 is not present in the manuscript, the square is transparent in the left table.](image)


124 The visualization is introduced by an explanation of the editorial rationale and a ‘manual’ of how it can be used. See [http://www.beckettarchive.org/writingsequenceofinnommable.jsp](http://www.beckettarchive.org/writingsequenceofinnommable.jsp) (last accessed January 10, 2017).
Dillen et al. also visualized the temporal aspect of the genesis by animating the writing sequence of the manuscript of *L’Innommable*. In the animated visualization, a grid is filled with squares in the order the sentences were – or are presumed to have been – written. As figure 28 shows, this visualization is not intuitive. Before readers/users can use it to its full advantage, it is necessary to understand what the coloured squares stand for, and why the table and the grid are juxtaposed; in short, to read the introduction provided. The introduction is, in fact, highly relevant for the present discussion, because it gives a detailed account of the editors’ modelling process of finding a suitable visualization. Talking readers/users through five different visualizations of the same data (namely the TEI/XML encoded sentences of *L’Innommable*), Dillen et al. show the advantages and shortcomings of such a representation. For instance, in the third visualization the writing sequence is also presented as a grid (instead of a table). However, the editors found that ‘some crucial information’ about the order in which the sentences were written had been lost, so they decided to add diacritical symbols to the grid view. Instead of discarding the third ‘in-between’ visualization, they kept it in the explanatory section to this feature. So, when Dillen writes that this digital visualization assists the editors in conveying their argument about the text (2015, 240), I’d like to add that by communicating the intermediate steps in the modelling process, the editors also manage to inform readers/users of the purpose and bias of digital visualization.

By linking the macrogenetic visualization to the Synoptic Sentence View, which, in turn, is linked to the manuscript containing the sentence, the BDMP allows its readers/users to freely roam the spectrum of textual variation. The TEI/XML transcribed manuscripts are at the base level of all visualizations. In general, the sentence is used as the textual unit and acts as a navigation mark that leads the user through the range of perspectives on text. Sentences can be viewed either in minute detail within the context of a transcription of the document page (close reading perspective), in comparison to other versions of the same sentence (through the Synoptic Sentence View or through collation), or as a specific point in the development of the work (distant reading perspective). The moment a sentence is viewed in the Synoptic Sentence Viewer or in the macrogenetic visualization, it is connected to the document page by means of a thumbnail, or a mouse-click on the related square. In other words, a

---

sentence can be viewed from a documentary point of view, a textual point of view, or a juxtaposition of both.

Another project that explores the representation of various access points to the text’s genesis is the Critical and Synoptic Edition Online of James Joyce’s Ulysses. In an experiment with a so-called ‘Diachronic Slider’, developed with technical assistance of Joshua Schaüble, Gabler intends to show the continuous development of a textual segment within both a manuscript and across the borders of the document. This necessitates a distinction between the intra-documentary variation – the ‘layers’ of text – and the interdocumentary variation or the ‘levels’ of text. As discussed earlier (see chapter 1.3.2), layers in a text come about through authorial revisions and consist of edit operations such as additions and deletions. They are the writing traces manifest on a document page. The levels of a text’s genesis consist of the same edit operations, but they only become clear after a collation. They are established by edit operations between witnesses: a sentence that is present in witness A and absent in witness B is considered a deletion or omission. In other words, the levels constitute themselves in the transition from document to document (Gabler and Schaüble, 2016). With the Diachronic Slider, Gabler and Schaüble present a visualization that combines both layers and levels and thus merges different perspectives.

This ambitious approach results in an information-rich visualization. Figure 29 shows the Diachronic Slider applied to lines 741 to 756 of Ulysses’s text. A pop-up panel shows two visualizations of the underlying TEI/XML transcript. The upper half displays a visualization of the transcription of a specific witness – in this case ‘level B’ that represents the changes made to the final working draft after fair copying. The edit operations made within this witness (i.e., the layers of text) are indicated with colours – red for deletions and green for additions – and by displaying the words as strike-through or superscript respectively.

126 The line numbers are based on the Critical and Synoptic Edition of Ulysses (Joyce 1986).
The lower half of the panel displays a visualization of a transcription of *Ulysses*’ text that is based on the print publication of the critical edition (Joyce 1986). The transcription is enriched with diacritical marks and alphanumeric sigla referring to the text’s witnesses. By moving the slider from left to right, readers/users can switch between different witnesses (i.e., textual levels) in the upper panel and see how the text develops over time and across witnesses. The synoptic view in the lower panel remains constant and, as such, it appears to function as a navigational aid or an additional key to understanding the text’s development. As this visualization conveys information and editorial knowledge that is already present in the TEI/XML transcriptions, which are in turn based on the print editions, the Diachronic Slider can therefore be considered as a digital expansion upon existing scholarship. It is primarily created to facilitate the use of this edition and to make the different witnesses easier to access.

An additional feature is that readers/users can also choose a transcription view that displays the interdocumentary variation. We could call it a simple synoptic transcription: since it concerns only two witnesses there is no need for diacritical marks. The changes that occur between documents are represented inline and indicated in red (deletions) or green (additions). Figure 30 reproduces this visualization of a transcription of Virginia Woolf’s ‘Sketch of the Past’ (1976), displaying the variation occurring from manuscript to typescript. With its simplified, coloured rendering of the writing traces on the
draft manuscripts, the project shows the value of a simple, direct model to study certain features of a text.

Figure 30. A synoptic transcription view of the textual variation occurring between the manuscript and the typescript of Sketch of the Past. The modifications in grey are intradocumentary edit operations on the level of the manuscript.

It needs to be said that both the Ulysses-project and the Woolf-project are work in progress. I decided to include both projects in this discussion because they provide an insight into the editorial thought processes and the modelling of a digital visualization. Similar to the editors of the BDMP, Gabler and Schaüble set out to model the text’s genesis from different points of view, generated from one underlying digital transcription and simultaneously making use of existing knowledge about the text.

Paralipomena in the Picture

Thus far, we have seen how digital visualization can be employed to visualize transcriptions from a documentary point of view or a textual point of view; how it can show the macrogenetic development of a text; and how it can clarify successive steps in the writing of one sentence. In all these cases, the documents that are visually included have a direct connection with the text. However, the

127 I am very grateful to Hans Walter Gabler for giving me the possibility of a ‘sneak peek’ into this ongoing work and for providing documentation in an extensive ‘Read Me’ file, which helped me understand the project’s terminology and methodology.
‘avant-texte’ being so versatile, there is a myriad of documents that may be linked to a stage within the work’s development, but do not contain a proper version of the text. This type of material is described as ‘paralipomena’ (Mathijsen 2003, 283). Paralipomena could also consist of tables, diagrams, doodles or sketches – in short, it is a category difficult to define. They are of value to the text’s genesis, because they are related to the text’s origins and development in the broadest sense. Van Hulle points out that this type of material offers exciting possibilities to explore the theme of imagination and written invention (2001, 18). Excluding paralipomena from an edition, then, implies discarding potentially interesting information about a work’s development. Examples of paralipomena in the work of Brulez are the notes he made prior to or during the writing process of Sheherazade (see figure 31 and figure 32).

Figure 31. Note 5 from the ‘avant-texte’ of Sheherazade, reading: ‘Shiriar: / De gramofoonmuziek [sic]: / Strauss “Salome”: muzikaal / flagellantisme / flagellantische muziek.’ (AMVC Letterenhuis B917 / H2a).

Figure 32. Note 15, reading ‘Gramofoon [sic] / De xxx microfoon <det> vlotte deinen op den zwarte / draaikolk der <ebonieten> plaat <en> / waarop^ <om> ^als gold het^ de / symbolische gelijkenis met een naar zijn snel einde / <xxx> spoedend levensbestaan te

128 For a discussion of the definition of ‘version’, see chapter 2.1.2.
The phrases on the notes are unmistakably related to sentence 0061 that is added in the margin of the manuscript MS-B917/H2a, and incorporated in later textual versions (typescripts, print proofs, and publications). On the folio typescript, the sentence reads

‘Shiran. Hij laat den microfoon deinend vlotten op den ebonieten draaikolk der plaat, waarop - als gold het de symbolische gelijkenis met een naar zijn snel einde spoedend levensbestaan te volledigen - het lichtspel den schim van den zandlooper der eeuwigheid spiegelt’ (TS-B917 / H2a).

This version of the sentence is, albeit more fleshed out, not much different from the version on note 15. In other words, the note contains one of the first versions of the sentence and thus represents a stage in the text’s development. There are, of course, practical reasons for excluding paralipomena from scholarly editions: the printed books lack space to provide them a place other than in the footnote section or appendices where they, like other textual variants, do not receive much attention from readers. Some theorists also argue that, because of their lack of syntactic structure, paralipomena cannot be considered a proper textual version (Scheibe 1991, 26; Mathijsen 2003, 282-4; see also in Van Hulle 2001, 11). This is not to say that scholars do not recognize the value of that type of documents: the poetic and prose works of Goethe (first published 1836/1837) already included paralipomena. The digital medium, then, does not so much bring about a change in the methodology of scholarly editing, but it offers the possibility to make visual what before could only be communicated textually.

Since digital scholarly editions often also function as digital archives, the inclusion of paralipomena presents little difficulty. Their classification is a little harder. Given that they often originate in the earliest moments of creation, they are direct reflections of the messy writing process and such activity cannot easily be classified into strict categories. Moreover, it means setting aside the idea that a version needs to consist of a complete text. Instead we could consider paralipomena as ‘the sole material witnesses’ of an intermediate stage in the writing process (Vanhouette 2001, 149). The editors of the BDMP follow this approach, acknowledging the gain of including ‘loose jottings, ideas, false starts and potential alternatives’, such as the early versions of sentence 0061 from

volledigen - het lichtspel / der schim van den zandloper der eeuwigheid spiegelde’. (AMVC Letterenhuis B917 / H2a).
By taking the sentence as the smallest textual unit, the editor can include paralipomena in the digital edition and visualize them in their ‘Synoptic Sentence View’. This visualization presents the sentences entirely separate from the document they were written on, thus showing the work’s textual development.

Of course, the relationship between paralipomena and the text of the work is exactly where it gets tricky. If the paralipomena bear no direct relation with the published text, the BDMP includes it in the Synoptic Sentence View as ‘loose jotting’. This is achieved by encoding the phrase with a `<seg type="paralipomena">`. When the editors suspect a relationship between the jotting and a sentence, the `<seg>` is given an additional attribute (@n) with the number of the sentence in question. I have already discussed the (animated) macrogenetic visualization of the macrogenesis of the BDMP, which linked discarded sentences to the published text by visualizing them as blank squares that corresponded with coloured ones in the published text grid. A similar visualization, which shows the place of paralipomenon in relation to the work and permits this zooming in and out, is used for the digital edition of Goethe’s Faust (Goethe 2016).

The textual tradition of Faust includes a significant number of paralipomena, which have already been treated quite extensively in earlier research publications, although these did not really attempt to establish their role in the genesis of the work (Bohnenkamp et al. 2011, 28). As a result, the digital Faust edition includes a wide range of paralipomena, such as early versions of a verse (Konzept), letters, diary fragments, and schemas. They have a prominent place on the interface of the ‘landing page’, next to the text of Faust I and Faust II (figure 33, next page).

---


Clicking on ‘Paralipomena’ gives access to a complete list of all paralipomena of *Faust*; the individual paralipomenon can subsequently be viewed from different perspectives (documentary or textual transcription, metadata, and digital facsimile). At the foundations of these different visualizations lies a complex encoding system, which is more elaborately discussed in ‘Perspektiven auf *Faust*’ (Bohnenkamp et al. 2011, specifically the section ‘Text- und Dokumentmodell’). Each source document in the *Faust* archive is encoded twice (‘doppelten Kodierung’): once from the documentary and once from the textual (genetic) perspective. Both transcriptions are linked to one another; from these transcriptions different visualizations are generated.\(^{131}\)

A comprehensive overview of the genesis of *Faust* is presented in the form of interactive diagrams that show the writing process of the entire work. The diagram resembles a conventional genetic plan, but with each mouse-click readers/users can dive deeper into the texts of *Faust*: the work is first divided into *Faust I* and *Faust II* and then, clicking further, into separate acts and scenes. To illustrate, we take a look at the scene ‘Studiezimmer II’ (figure 34, next page).

\(^{131}\) A TEI/XML transcription can express different perspectives on a source document. Nevertheless, these transcriptions generally prioritise one structure (text or document) in order to avoid conflicting hierarchies. The *Faust*-editors aim to capture both the documentary aspects and the textual aspects of Goethe’s manuscripts. In an attempt to circumvent the difficulties with nested elements and overlapping hierarchies, they decided to encode the source documents once from the documentary perspective and once from the textual (genetic) perspective.
Figure 34. Selection from the Faust-edition showing the genetic plan of the scene 'Studierzimmer II'.

The horizontal axis of the diagram represents the verses of the base text; the vertical axis contains the different sigla representing either a version of the work or a paralipomenon (indicated by a siglum starting with a ‘P’). Taken together, the coloured bars represent the extant texts of the scene Studierzimmer II. The visual overview quickly illuminates how the paralipomena are related to the other documents. Clicking on a specific bar in the diagram, for example the bar representing paralipomenon H P15, opens a digital facsimile view on the source document that allows for close reading. In this window there are other views to choose from: a documentary transcription, a textual transcription, and a critical apparatus. The documentary transcription can be plotted over the digital facsimile, or studied independently. The textual transcription is a linear transcription of the scene with, in the margins, references to other paralipomena that are linked to the text. As figure 35 (next page) demonstrates, this specific part of text is linked to paralipomena P9, P43, and P62.

132 What is more, this particular source document containing H P15 illustrates how an extant document may contain several different texts, for it also contains paralipomena P16 and P17.
Figure 35. Selection from the Faust-edition, showing the reading text transcription of the scene Studierzimmer II with in the right margin the numbers of the associated paralipomena.

The editors of Faust are also confronted with paralipomena that have no direct connection to the text. The text on paralipomenon P6, for instance, does not recur in any later version. It constitutes what Van Hulle calls a cul-de-sac: a dead-end in the writing process (see chapters 1.3.3 and 4.2.1 ‘Teleological/Chronological Classifications’). For that reason, it could not be included into the schematic diagram discussed above (Bohnenkamp et al. 2011, 64). Moreover, since the diagram uses the verses of Faust as a reference system, it does not include any prose paralipomena (like letters). The editors recognize the limitations and challenges of their encoding system. After all, a literary work does not develop in a structural and linear way. Nevertheless, they consider it their role as editors to build plausible hypotheses about its development, to formalize them, and to use those formal models as the foundation for a digital visualization of each step in the text’s genesis (Bohnenkamp et al. 2011, 65). With that statement they confirm the premise of this chapter: digital visualization is always based on a model of the genesis that is constructed by the editor. According to that model, the editor captures and records certain information in a digital transcription (and excludes other information by necessity). Visualization requires a selection from the information in the transcript; even the combination of documentary and textual transcriptions cannot prevent certain information to be excluded. The possibilities of digital visualization, then, do not ensure a complete representation of all texts. Rather,
they can be employed to support and enhance the editorial argument; in this case by showing a type of document that has often been left out but that help to examine the earliest moments of genesis.

In conclusion, the digital *Faust* edition contains an impressive amount of documents (digital facsimiles) and texts (transcriptions) as well as scholarship. Even though the complexities of the interdocumentary and intradocumentary relationships are captured in the underlying code (Brüning et al. 2013), visualizing them is not straightforward. Especially for readers/users not familiar with the textual tradition of *Faust*, the use of the edition requires at least a reading of the editorial guidelines and some practice in understanding the way the various views are linked.¹³³ For example, the *Faust* edition provides a view where the documentary and the textual perspective are juxtaposed, but the textual transcription clearly contains more text than the paralipomena in question.¹³⁴ The reason for this is not entirely clear, especially since the digital edition does not present any editorial annotations to the paralipomena: it only references existing scholarship. That being said, the different visualizations and the interactive genetic plans make great tools to understand the connections between the different documents of *Faust*. Moreover, by not only including the paralipomena in the archive but also giving them an important place in the interface, the editors ensure the inclusion of the earliest traces of creation into the digital edition.

The BDMP and the *Faust* edition make clear, then, that digital visualizations can be used to extract a wealth of knowledge encoded in a digital transcription. A considerable strength of the digital medium is that the source documents can be considered from different angles: readers/users can switch between a digital facsimile, a documentary transcription, a textual or linear transcription, or a combination of the three. The editor can make use of a variety of visualization methods to assist further endogenetic study, ranging from colour-coding the revisions to plotting the documentary transcription on a semi-transparent digital facsimile. Such methods reduce the flattening effect of transcriptions (see also chapter 2.2.1): although the inscriptions on a draft manuscript are forced in a structural, linear order when they are transcribed, to

---

¹³³ The digital edition is currently only available in beta (accessed on January 15, 2017).
some extent the ensuing digital visualization restores the multidimensional source document. We could even say that it adds more dimensions, enabling readers/users to view the document and the information it contains from original perspectives (like seeing only the top layer, or only additions in a specific hand). In all cases, it holds true that the visualization is an abstraction of the source transcription that is, in turn, an abstraction of the source document. Following Pierazzo and Stokes, who write that ‘a certain level of simplification and abstraction is required for the model to be useful’ (2011, 421), we can say that as long as readers/users are aware that what they see is a certain abstraction is understood, the visualization can be used to the advantage of research.

The different perspectives on text can be extended to a macrolevel where, similar to the concept of distant reading, the structural development of the text is shown in graphics. Since the visualizations are derived either from one and the same digital transcription or from several systematically linked transcriptions, the aforementioned principle of ‘filter and zoom’ (chapter 3.1) applies: readers/users can zoom in and out to navigate between different aspects of genesis. Although the macrogenetic graphics may not be immediately comprehensible – the more complicated a textual tradition, the more complex its visualization – editors do not appear to eschew presenting the readers/users with a complicated view. The visualizations are complex but (to paraphrase Van Hulle\(^\text{135}\)) in their complexity they draw attention to the places of textual revision and thus highlight the writing process.\(^\text{136}\) With digital visualization, in other words, the complexities of manuscripts are not concealed but rather revealed; complicated writing processes are not necessarily simplified but put forward; and the alerted readers/users are provided with several tools to further explore these places of revision.

If we think about the potential functionalities of a typical genetic digital edition and the wealth of information it contains, it is easy to imagine that first-

\(^{135}\) Making the case for an automated collation of genetic variants, Van Hulle proposes that in one version of a text all corrections (deletions and additions) are encoded and that the user can choose to visualize the ‘ground layer’ (including the deletions) and the ‘top layer’ (including the additions). The result of this collation might be a ‘slightly non-sensical’ text, but that is by no means problematic: the syntactical inelegance signals the genetic variant and as such the genetic writing process rather than the end product (Van Hulle 2009a).

\(^{136}\) This is not to say that the visualization should be more complex to decipher than the original documents. As Bryant remarks about a traditional critical apparatus: ‘if the textual apparatus will always be difficult and perhaps more difficult than the original upon which it is based[,] it is a damning testimony of the ineffectuality of an apparatus to clarify textual fluidity’ (Bryant 2002, 129).
time read-users won’t use the edition to its full capacity. In that respect, some commercial websites and smartphone applications (‘apps’) set an interesting example by offering their visitors and users the option to ‘take a tour’ through the interface. During this tour, the user becomes acquainted with its functionalities and how these can be used. Explanations are often given in the form of pop-up tooltips through which the user can click, ending with the confirmation ‘I got it’. Naturally, almost all digital scholarly editions come with a manual of some sort to explain their content and functionalities, but the threshold to take a quick tour is significantly lower. An example of this is provided by the Kundige Bok (KB) edition (primary editor Malte Rehbein) in which the functionality of each button is explained with a mouse-over tooltip (figure 36).

Figure 36. Screenshot of KB edition showing the tooltip that pops up upon a mouse-over (here the mouse hovered over the option ‘Kürzungen ein/aus’). The text displayed in the transcription panel is classified as ‘Wein’ and consists of four versions (see ‘Textschichten’ in the upper right corner as well as the timeline with four dots). Here we see the version dated from 1472–1496.

Moreover, the difference between text versions is briefly explained in similar text blocks that appear upon mouse-over (in figure 36 the dark grey block in the timeline section). In short, the hover boxes in the KB not only demonstrate the workings of the interface, but also inform readers/users about the contents of the edition, and introduce the textual variance between versions. This brings me
to the last – but certainly not least – instrument to illustrate the development of a text: the genetic narrative.

Narratives of Genesis

With respect to the visualization of the genetic narrative, the editorial annotations remain one of the principal aids. Besides the tooltips of the KB (above), we find examples of such editorial comments in Bryant’s fluid-text theory (2002). Bryant observes that ‘exposing readers to a fluid text ... through the physical complexities of the change itself – requires a high degree of editorial guidance’ (2002, 138). Although readers/users may have an understandable impulse to go for the clear reading text instead of the pains of scrutinizing a complex documentary transcription, Bryant thinks the editor can guide readers/users through the various stages of the writing process, until they experience the ‘deeper pleasure’ of understanding textual revision (135). Bryant describes a rather hesitant reader who, not used to textual variance, needs to be coaxed or enticed into enjoying the space between two versions of a text. The task of this persuasion obviously befalls the editor who can engage her knowledge of the text’s genesis and development to present the revision in compelling narrative form. This is what Bryant calls a ‘revision narrative’.

The revision narrative entails more than the usual editorial annotation. Instead, it sketches the complete story of the textual development, presenting revision in a linear way.\textsuperscript{137} This linearity is inevitable, argues Bryant, as ‘variants and versions […] are sequential and grow in linear fashion’ (2002, 148). Since the order of revisions cannot be established with absolute certainty, a revision narrative is an editorial construct; it is subjective yet based on the editor’s meticulous analysis of the extant documents.

A practical implementation of this theory can be found with the fluid-text edition of Herman Melville’s Typee (2006). In addition to the features that we have seen in the previously discussed editions (i.e., a digital facsimile of the source document and different renderings of the transcription), the textual transcription (figure 37; lower left corner) indicates the places where textual revision narratives can be found. Bryant describes this as a ‘revision narrative’.

\textsuperscript{137} What Bryant means with the ‘complete’ story is explained using the Typee manuscript of Herman Melville. In this case revision narrative includes ‘revisions of all versions associated with the Typee manuscript throughout the life of Typee, by offering revision sequences as a visual component of revision narratives, and by creating full, not partial, narratives that hypothesize upon motives of revision’ (Bryant 2002, 190).
revision took place, the so-called ‘revision sites’. Clicking on a highlighted revision site opens a panel that shows the revision step-by-step in the chronological order it presumably took place.

Figure 37. Selection from the revision site of ‘comprising an extensive oblong pipi’ on 10ms50-51 in the digital Typee-edition.

Similar to the ADS edition (chapter 2.2.4), the separate steps in this particular revision sequence do not contain complete versions of the sentence. Instead, the editor disentangles the various corrections made in the text and presents them in a hypothetical chronological order. The revision itself is indicated in bold; the surrounding text is provided to place the revision in context and make its meaning more understandable. Bryant considers this method of presentation advantageous over a documentary transcription that graphically renders all deletions and additions, especially in the case of heavily revised manuscripts with a ‘miasmic cascade of separately cancelled and inserted words’ (Bryant 156). He may have a point, especially if we think about this hesitant reader who still needs to be lured into the pleasurable depths of genetic analysis and may be put off by a more legible, but still incomprehensible documentary transcription. Accordingly, the revision sequence and the revision narratives are visually
separated from the documentary transcription and specifically linked to the reading text\textsuperscript{138} of *Typee*.

In the end, the revision narrative is specifically intended to further explain what happens at the revision site in question. Figure 38 shows that the narrative is, in fact, a story with a plot, characters, motives, and agents.

In Bryant’s theory, the revision narrative may ‘tell a complex tale of intentions and maneuvers with novelistic intensity and suspense’ (Bryant 159). Accordingly, the editor is given carte blanche when it comes to describing the poetics of textual variance – naturally within the boundaries of editorial methodology – which makes the revision narrative the primary means of communication between editor and reader.\textsuperscript{139} It is pre-eminently the point where the editor can fully reveal herself and display her knowledge of the text, but also communicate her enthusiasm about its development and her marveling about its genesis.

The different panels are confronting in the sense that they do not allow for a fluid, immersive reading of the text, but that is exactly the point of the

---

\textsuperscript{138} The reading text is most probably the top layer of the manuscript text. The editors describe it as ‘Melville’s final draft, as evident from the manuscript’ (on ‘This Site’ An Elementary Guide’ on http://rotunda.upress.virginia.edu/melville/site.xqy; last accessed January 12, 2017).

\textsuperscript{139} Consequently, in the same vein as I have been talking about a reader/user of a digital edition, Bryant refers to an ‘editor-narrator’ (2002, 161).
visualization. The underlying aim of the revision sites and the revision narratives is to ‘foreground variants’ and by interrupting the reading, readers/users are made aware of the fluidities of the text and the editorial judgments made about them (Bryant 2002, 119). That is, the fluid text-approach not only foregrounds textual variation, but it also gives the editor a more prominent place. So what readers/users of the Typee-edition see when looking at revision site ‘10ms50-51’ is the editor’s hypothesis about the order in which the text is written and revised, and the same goes for the narrative of that revision. Each source file of the Typee edition is a TEI/XML transcription that consist of three parts: a documentary transcription of the manuscript, followed by a <div> with the revision sequences, and finally a <div> with the revision narratives. The edit operations in the manuscript text are recorded with <del> and <add>, but the revision sequence and the revision narrative are plain text (see figure 39). By means of a reference pointer with a unique <xml:id> the sequence and narratives are linked to the exact location in the manuscript transcription. As a consequence, the TEI transcription proper remains free of the interpretative aspect of the revision narrative and revision sequence. Both the revision sequence and the revision narrative can thus be considered extensive editorial annotations to the text.

Figure 39. Selection of the XML-based TEI transcription of revision sequence RS10ms50-51, showing the successive steps as <items> in a <list>.

This separation of concerns, with on the one hand a documentary transcription that serves as the source for different visualizations and on the other hand an editorial interpretation of the writing process, appears to be the best solution for

140 Although the TEI Guidelines offer a specific tagset to encode a writing order or revision sequence (using for instance the <mod> element or the @seq attribute), the editors of the Typee edition have refrained from using them.
the encoding and representation of a text’s genesis because, as Dillen observes, ‘XML is an inferior medium for developing “revision narratives”’ (2015, 201). The technology of XML requires an explicit formalization of rules that does not correspond to the idea of expressing tentative hypotheses. The latter are better expressed, it seems, in a narrative form and stored separate from the digital transcription. In addition to his narratives of revision, Bryant talks about a combination of the printed book and an electronic edition (2002, 122). In the same vein, the editorial team of the BDMP publishes in concurrence with each module of the digital edition a print monograph that elaborates upon the editor’s analysis of the genesis.

Reflection

This section attempted to provide a brief summary of the literature and the practice regarding the visualization of digital transcriptions. In line with the idea of modelling, this practical discussion was not intended to be all-encompassing: it only scratches the surface of a wide variety of editing projects and some practices have necessarily been left out for reasons of space. From Caedmon’s Hymn-edition to the BDMP and the Proust prototype, the projects provide examples of the many digital visualization methods used to showcase textual genesis and make it accessible to readers/users. Based on the overview, we can roughly distinguish three uses of digital visualization. The first function is to assist scholarly editors during their analysis, recording, and processing of the text. Whether editors encode a static HTML file that represents a certain interpretation of the text, or figure out the optimal dynamic representation of the macrogenesis of a literary work: visualization is used as a modelling tool in constantly refining the argument.

The second usage of digital visualization is therefore the presentation of the editorial analysis and argument. Traditionally this is done in introductions to the edition proper, in footnotes or editorial comments. In the form of revision narratives they receive a new, more prominent place: their presence is allowed to be disruptive and meant to be engaging. A third use of digital visualization, or rather an effect of it, is that readers/users are given more tools to become actively involved with the visualizations, toggling between them to find the most suitable view or, conversely, to experience the dynamic process of writing. They can zoom out for a macroperspective and understand the work’s genesis on a structural level (think of the interactive genetic plan of the Faust
193

edition), and subsequently zoom in to a microperspective for a closer study of one specific manuscript witness. It follows that the (presumed) interests of the readers/users play an important part in the modelling of each textual representation.

One of the main findings to emerge from this discussion is that the technology of digital visualization provides editors with the means to present multiple perspectives on both a text and its development. In most digital editions readers/users can switch between digital visualizations of the document and the text. Ideally, readers/users gain a heightened awareness of the fact that text has no fixed form. The various perspectives on text are related, either because the link between them is hard-coded, or because they are all derived from the same source file. This requires the editor to make a decision on a foundational and recurring topic in digital scholarly editing theory: the separation between text and document. Even when this separation is made on a conceptual level, its implementation prompts what Galey describes as 'the kind of dilemma known as a “religious issue” in programming jargon: is it desirable, let alone possible, to divide the content of a text from its material form for the purposes of machine-readability and large-scale computation?' (2010, 110) We have seen several interpretations of the concept of ‘documentary view', ranging from ultradiplomatic transcriptions plotted over a digital facsimile to semi-diplomatic that represent the revisions in different colours. In all cases, the documentary perspective is intended to make the inscriptions on the manuscript readable and help readers/users to discern layers of writing and stages in the writing process. Conversely, the textual perspective is intended to facilitate the reading of the text. For genetically encoded text this means that the revision process is concealed in favour of a clear reading text, usually by visualizing the top layer.

Interestingly, the separation text/document is often not implemented in the underlying code: the TEI/XML transcription ‘just’ contains sufficient information to generate a documentary view and a textual view. Capturing all elements of a source document as well as the meaning of its inscriptions seems impossible. TEI/XML transcriptions can quickly become unwieldy and difficult to process. In order to avoid a complicated and dense encoding, most digital edition projects aim for a middle road between document-oriented and text-oriented encoding. When different perspectives are to be generated from the same source file, the transformation from one format to another is a procedure of selection and obfuscation. Accordingly, the XSLT files, too, contain the
editor’s argument about what information is required for a documentary perspective and what information is required for a textual perspective. Since this method can take up quite some processing power when done on the fly, it is sometimes (partly) prepared in advance. The example was from the Caedmon’s Hymn-edition, which prepared static HTML pages for each view on the text, but the BDMP also preprocesses its TEI/XML transcriptions to speed up the visualization of the Synoptic Sentence View (Dillen 2015, 207). Conversely, the Faust edition has implemented Zeller’s separation between the documentary record and the editor’s analysis, which resulted in two separate TEI/XML files.

The next section picks up where Chapter 2 left off: it discusses the visualization of collation output. Since collation is, like transcription, an instrument to examine textual variance, we will see that much of the current findings apply. Visualization of collation output can be used to support an argument, to make readers/users aware of the different views on text, and to support close and distant reading. At the same time, it requires dealing with the separation between document and text, and finding ways to work around the hierarchical data structure of XML.
3.2.2. Visualization of Collation Output

'It is not as difficult to compare texts by computer as it is to record the results of the comparison in a way that will be easy and unambiguous to read.'

(Dearing 1970).

**Introduction**

As Vinton Dearing remarked, making the results of automated collation understandable and easy to read can seem a more daunting task than the automated collation proper. While his observation dates from the 1970s, it is still a challenge to visually render the collation output. Overall, the principles of digital visualization as summarized in the introduction to this chapter (3.1. ‘Framework’) continue to apply. With regard to automated collation, these principles translate to (1) the cognitive exercise of thinking about visual representation of the collation output; (2) the iterative modelling process of refining the model of the text (i.e., the edition); and (3) the mediating, communicative aspect of a visualization as a critical argument. In other words, both editor and reader/user can use the visualization of the collation output. The editor may use it as an intermediate step in the textual analysis, to validate for instance the results of a collation, to check the transcriptions, or to think about how to convey a certain amount of information recorded in the transcriptions. For the readers/users, a visualization of the collation output can facilitate exploring the variation between versions, i.e., the levels. As mentioned previously (3.1. ‘Framework’), the effectiveness of a visualization is difficult to pin down without user testing. With that in mind, we can ask ourselves to what extent a familiarity with a certain output format impacts the user’s preferences. It may be that readers/users of scholarly editions expect a specific format, like a certain representation of a critical apparatus, simply because they are used to it. Why change a winning format? If that were true, readers/users may be better served with (adjusting) an existing format than with an innovative but unknown design.

Incidentally, ‘following existing standards’ need not be negative. The refurbishing of an output format for the digital medium can thoroughly change its function and use. Let’s take for instance the transmediation of the critical

---

141 This at least was the observation of some of the speakers at the Interface symposium in Graz (September 23, 2016), namely Hugh Clayless and Roberto Rosselli Del Turco.
apparatus. In principle, any practical reason for storing textual variants in an appendix (or discarding them entirely) is ruled out thanks to the functionality and storage space of the digital medium. In digital form, then, the critical apparatus would no longer be a place where textual variants are respectfully buried and forgotten (‘a graveyard’), and instead could become a dynamic tool that displays variance and allows you to study the patterns of textual variation. It continues to exist, but in new and improved format.

The previous chapter has presented the workings of collation software programs and emphasized that an understanding of how the text transforms ‘under the hood’ adds to a more accurate textual awareness. With each step in the automated collation process (e.g., preprocessing, collation, and postprocessing), information is added or taken away. The text files are collated and then outputted to a selected file format. This format depends on the collation program: if we look at CollateX, the collation results are internally stored in a graph structure and subsequently outputted to JSON, TEI/XML, or a variant graph. Visualizing the collation output will always reduce the information stored in the output format that, in turn, is extracted from the data stored in the computer’s memory. Although this reduction is not undesirable – in its raw form the collation result is not very informative or understandable for human readers – we need to be conscious of the ways in which information is both presented and concealed in different visualizations: it affects the study of the textual variants.

Furthermore, Chapter 2 described the difference between textual layers and textual levels, respectively the editor’s genetic analysis of the order in which revisions are made and the interdocumentary variance established by collating different witnesses to the work. It follows that we need to distinguish layers and levels in the visualization. Closely related to the idea of layers and levels is the concept of version. Genetic editors employ flexible definitions of ‘version’ that are highly influenced by the nature of the revisions that they are dealing with. In brief, versions are larger or smaller ‘units of comparison’, ranging in size from characters, words, or revision units. In the present section, five existing visualizations are described, illustrated with implementations in digital editions. The focus of the discussion is on their value for textual genetic research.
3.2.2.1. Alignment Tables

For most philologists, an alignment table is a natural or familiar format to present interdocumentary variation. The collation result is presented in tabular format where witnesses are displayed vertically or horizontally. Each cell contains one or several tokens, and if a token has no match an empty cell is introduced (a ‘gap’) to visually align the surrounding tokens. Some tools (like CollateX) offer the choice to either group matching tokens in one cell as segments, or have one individual token per cell. Figure 40 presents an example of a vertical alignment table generated by CollateX, which presents the results of a collation of the first paragraph of Charles Darwin’s *Origin of Species* in the six editions published during his lifetime. The matching tokens are grouped together as a segment in one cell. We see here that three witnesses (1859, 1860, and 1861) lack the words ‘Causes of variability’ because that cell is empty.

<table>
<thead>
<tr>
<th>1859</th>
<th>1860</th>
<th>1861</th>
<th>1866</th>
<th>1869</th>
<th>1872</th>
</tr>
</thead>
<tbody>
<tr>
<td>WHEN we</td>
<td>WHEN we</td>
<td>WHEN we</td>
<td>WHEN we</td>
<td>WHEN we</td>
<td>WHEN we</td>
</tr>
<tr>
<td>look to</td>
<td>look to</td>
<td>look to</td>
<td>compare</td>
<td>compare</td>
<td>compare</td>
</tr>
<tr>
<td>the individuals of the same variety or sub-variety of our older cultivated plants and animals, one of the first points which strikes us</td>
<td>the individuals of the same variety or sub-variety of our older cultivated plants and animals, one of the first points which strikes us</td>
<td>the individuals of the same variety or sub-variety of our older cultivated plants and animals, one of the first points which strikes us</td>
<td>the individuals of the same variety or sub-variety of our older cultivated plants and animals, one of the first points which strikes us</td>
<td>the individuals of the same variety or sub-variety of our older cultivated plants and animals, one of the first points which strikes us</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>

Figure 40. HTML alignment table output, generated by CollateX after collating the first paragraph of six witnesses of Darwin’s *Origin of species*. The witness sigla are represented in bold at the start of each column.

Considering the simplicity of the alignment table format, it is evident that its capacities to represent genetic information are limited. Nevertheless, as described in chapter 2.3, it is possible to pass along and to visualize markup. This results in a more informative alignment table: as we can see in figure 41, showing the collation output of a sentence with intradocumentary revisions in the BDMP. By processing the simple HTML output of CollateX with an XSL transformation, the BDMP can adjust the visualization and use colours to make it easier to identify invariance.
More importantly, the alignment table contains deletions and additions. As described in chapter 2.2, this markup is passed along for visualization purposes: the tokens are processed and collated in their normalized form. That means, for instance, that the tokens ‘to and fro and’ in witness 09 align with the tokens ‘<del>to and fro and</del>’ in witness 10. In this horizontal presentation the witnesses can be read from left to right; for longer sentences the readers/users can scroll horizontally. It is also possible to click through the sentences (with the <prev |next> buttons) and read the entire text of a witness text, sentence per sentence.

An interesting textual phenomenon, and especially challenging for collation software, is the transposition. If we look at the full transcription of witness 6 (MS-UoR-2935-3-2; figure 42), we see that it contains an intradocumentary transposition: Beckett initially wrote ‘Paced to and fro in the dark and back and forth’ and then indicated that the order of the words needed to be changed: ‘Paced to and fro and back & forth in the dark’. The alignment table in figure 41 shows that he continued to experiment with the segments ‘to and fro’ and ‘back and forth’, and that these segments were subsequently discarded altogether in favour of ‘from wall to wall’. What the table does not show, however, is the very first version of this sentence. The previous chapter explained how TEI/XML transcriptions are processed and ‘cleaned up’ before they are presented to CollateX: apart from ignoring certain markup tags this
cleaning process entails the implementation of transpositions. Only the readers/users that select the option to view the individual witness will find out about the transposition. The visualization is thus only a selective representation of textual variation: it is a tool to explore variance and it does not provide a comprehensive representation of the full genesis of the text.

Figure 42. Selection of the image/text transcription of witness 6, indicating the transposition.

The Charles Harpur Critical Archive (CHCA), which uses the MultiVersionDocument (MVD) technology discussed in chapter 2.2.4, employs another kind of alignment table. The concept of version as understood by the MVD and the CHCA has already been discussed, but in short: a text that contains layers of revision is encoded as two or more witnesses (one witness per layer). The witnesses are collated on character level, i.e., each character is one token. Readers/users have the possibility to consider the collation from different viewpoints: ‘Compare’, ‘Tree View’, ‘Alignment Table’. In this discussion, I will limit myself to the alignment table visualization. Here, too, the witnesses are displayed horizontally. Unlike the BDMP that works with sentences, the CHCA edition collates entire witnesses so that readers/users may need to scroll horizontally through the table before encountering any variant characters.
Since comparison is on the character level, the segmentation of the tokens in the cells in the alignment table is not necessarily on words or phrases, but on characters. Accordingly, sometimes a cell contains only one character (see figure 43). The decision comes from the editors who found that a character-by-character comparison of the versions better supports their analysis of the textual variance. Indeed, this granular, character-level visualization supports a microgenetic analysis of the text. The CHCA also employs colour coding: variant characters are indicated in blue to distinguish them from the matching tokens in grey. A macrogenetic overview of the text’s development is provided in the form of a Tree View, but regrettably the tree and the alignment table are not linked. Since the witnesses are stored in the MVD format, we can reasonably assume that the relationships between them are already present in the underlying encoding and that it is just a matter of visualizing them. Considering the aforementioned value of combining distant reading and close reading perspectives, linking the tree and the alignment table may improve the readers/users’ study of the textual dynamics.

Figure 43. Selection of the alignment table visualization of the collation of the poem Abed Ben Haroun. The character-level collation becomes clear at the word ‘wealth’ which has been split up on ‘wea’ and ‘[l]th’. Another remarkable example of a dynamic alignment table of collation results is CATView, developed by the working group on eHumanities of the Computer Science Institute of Martin Luther University Halle-Wittenberg.\textsuperscript{142} It makes it possible to view the textual variation at different degrees of granularity: from a

\textsuperscript{142} See project website of CATview on http://catview.uzi.uni-halle.de/ (last accessed February 2, 2017).
The macrolevel view (figure 44) readers/users can zoom in to see a more detailed view on the text (figure 45). On the macrolevel view, the witnesses are displayed horizontally with different rectangles representing a segment of text (e.g., word, phrase, or paragraph). On the vertical axis, the sigla of the witnesses are displayed. Segments are aligned vertically (in columns); in figure 45 we see that some segments have a darker shade, which indicates the level of similarity.

Figure 44. Selection of the macrolevel perspective generated by CATview.

Figure 45. Selection of the same collation on a more detailed level, with different shades indicating the degree of similarity between segments.

CATview is not widely known in the digital editing community, but it aims specifically to visualize the genesis of texts consisting of many manuscripts (Pöckelmann et al., 2015). Users can immediately spot the degree of dissimilarity and navigate through the different witnesses by scrolling and zooming in and out. In a review of the tool, Gioele Barabucci mentions a number of downsides of CATview (2016, 26-7), namely the fact that all rectangles are of the same size and should thus represent segments of text of a similar size (which is not always the case in genetic editing). Furthermore, it is not possible to select only a subset of witnesses to collate (unlike the collation offered by the BDMP, for instance). Other downsides of the CATview visualization are, in fact, inherent to any tabular representation of textual variation. To name but one: the grid visualization only supports texts with a regular development, making it nearly impossible to show transpositions.
Alignment tables are a good start for a discussion of collation visualization, because they are relatively straightforward and can be easily adjusted to whatever the editors choose as textual unit of comparison. As in all cases the text of the witnesses is presented in linear sequence (either horizontally or vertically); it is well-suited to a study of the relationships between witnesses on a detailed level, but less so to acquire an overview of the patterns of variation. A synoptic overview is more suited for that type of examination, if we take 'synoptic' in its most general definition: ‘of or forming a general summary or synopsis’ (OED). Other collation visualizations can be classified as 'synoptic'; I will discuss the inline critical apparatus and parallel segmentation.

A synoptic edition contains a visual representation of the collation results from the perspective of one witness, where the variants are indicated by means of a system of signs or diacritical marks. When using the term 'synoptic' in the context of scholarly editing, one thinks automatically of Gabler’s print edition of *Ulysses* (1986). The synoptic notation of *Ulysses* has two aims: to draw attention to the revision sites and to make explicit the editorial decisions made during the process of editing. Diacritical marks indicate both the layers and the levels of textual genesis. The synopsis interweaves with a reading text that functions as a stable text against which the variation is shown (not to be mistaken with ‘base’ text). As mentioned in chapter 2.2, the editors used TUSTEP’s technology to collate the witnesses.

Employing diacritics to signal variation is sometimes considered to hinder the use and the reading of an edition. Since each system of diacritical signs is slightly different (see Mathijsen 2003, 11.2.3 for an overview of these diacritical systems), it takes time to become acquainted with it; and some scholars reckon that this learning curve hinders the accessibility. In that respect, it is relevant to return to the remark of Bryant who asserted that the adaptation to a new reading system indeed requires some effort but that it will lead to more a more pleasurable reading experience (cf. ‘no pain, no gain’). In a similar vein, McGann argues that learning the diacritics used by Gabler presents no serious problems to readers once they have mastered the grammar, concluding that ‘when we shall have learned to “read” the synoptic text as we read the reading

---

143 BDMP has a very useful ‘Synoptic Sentence View’, but that doesn’t use collation but the aforementioned sentence numbering. Similarly, its dynamic macrogenetic viewer is also an example of a visualization of the sentence numbering (see 3.2.1 ‘Visualization of Transcription’).
text, we shall have gone a long way toward understanding the nature of texts in general’ (1985, 291). Nevertheless, the clear advantage of a digital synoptic edition is that the diacritical signs may be replaced by visual indications that have a lower usability threshold. In the Faust edition, for example, the presence of alternative readings of a verse is indicated with a slightly darker background of the text.

Accordingly, a synoptic overview can also draw our attention to the places in the text that underwent heavy revisions. The Diachronic Slider (discussed in 3.2.1) provides an exemplary illustration of this principle. Both layers of revision within a witness, encoded in the TEI/XML transcriptions, and variation across witnesses, established through collation with TUSTEP. Figure 46 shows how both layers and level are visualized synoptically in the transcription of the text.

![Selection of the synoptical transcription of Woolf’s ‘A Sketch of the Past’, showing the layers of revision present on the manuscript in grey, and the interdocumentary variation between the manuscript and the typescript in red and green.](image)

3.2.2.3. Parallel Segmentation

Parallel segmentation is the side-by-side display of segments to show the similarities or variations. The results of a collation are visualized in a non-hierarchical fashion. Contrary to the synoptic method discussed above, where
variant witnesses are displayed in relation to a stable ‘base’ text, parallel segmentation entails that witnesses are presented as variations of one another. Keep in mind that, like the critical apparatus, parallel segmentation is a method to visualize the results of a collation, not a collation method proper. In other words, the witnesses are first collated and then the results are encoded in a parallel segmentation format using the elements <app>, <lem>, and <rdg>. The scopes of the segments – i.e., the unit of comparison – are defined by the editor and range in size from words to (parts of) paragraphs. The TEI Guidelines notes that this method will always be ‘satisfactory’ for a comparison between two texts. Accordingly, the method is a widely used in digital scholarly editing: in the following paragraphs I will discuss three prominent tools that offer a visualization of text encoded according to the parallel segmentation method.

One of these tools is the Versioning Machine (VM), an open source software application developed and maintained at Maynooth University (Ireland). It accepts input text that is encoded according to the TEI <app/> <rdg/> construction and, additionally, it allows for a number of elements to be nested within the <rdg> element. This includes the <del> and <add> elements, so that it is possible to indicate layers of revision within one witness. Consequently, cancellations and additions, if desired, can be visualized in the output by means of XSLT transformations. The end user of the edition is thus offered an insightful overview of the places in the text where variation occurs within one witness and, if the different witnesses are placed side-by-side, also across witnesses. The digital edition of Walden is a good example. The edition of the work by American author Henri David Thoreau is created based on (a number of) the principles of Bryant’s fluid-text theory (2002). It may even be called a genetic edition, according to my colleague Aodhán Kelly in his review of the edition, although it does not include the complete avant texte of Walden. Still, the editors aim to present simultaneously multiple versions of the work including their textual variants (Kelly 2016, §7, online). The VM is used to process and visualize the critical apparatus. It processes the parallel

---

144 See chapter 12.2.3 of the TEI (P5, version 3.1.0).
145 The Versioning Machine is conceived by Susan Schreibman. See http://v-machine.org/documentation/ (last accessed September 20, 2016).
147 See http://digitalthoreau.org/fluid-text-toc/ (last accessed September 18, 2016).
segmentation encoded files and displays the more simple cancelled and added text by means of, respectively, strikethroughs and a colour scheme (figure 47).

In the editorial introduction to the *Walden* edition, the editor Paul Schacht asserts that in many places in the manuscripts, Thoreau has made revisions that are ‘too complicated to represent through this simple scheme [of the VM]’ (Schacht, online). These are, for instance, the transposition of paragraphs across versions, or highly complex revisions resulting in multiple writing layers. Moreover, Schacht recognizes the necessity of including ‘facts about the material state of the manuscript to grasp how a passage fits into Thoreau’s workflow’ (Schacht, online). That kind of information is supplied in pop-up notes at the paragraph level. The editorial presence in the collation is obvious; the hermeneutical decisions are made during the creation of the critical apparatus and in the first place the VM is a tool to display the textual content of the apparatus.

The Edition Visualization Technology project (EVT) also presents, as its name suggests, a visualization tool.¹⁴⁸ The project is primarily orientated towards

---

¹⁴⁸ The EVT is initiated and developed by Roberto Rosselli Del Turco (project leader) and Chiara Di Pietro, Raffaele Masotti, Julia Kenny, and others. See the project’s website on https://visualizationtechnology.wordpress.com and its SourceForge page on https://sourceforge.net/projects/evt-project/ (last accessed May 21, 2017).
visualizing critical editions and offers, among others, an informative parallel segmentation view (figure 48). Similar to the visualization of the VM, witnesses are presented side-by-side in separate panels. The critical text is displayed in the left panel, variant witnesses can be added in the panels to the right. Depending on their interest, readers/users can choose from different perspectives on the witnesses. The so-called ‘heat map’ view is perhaps the most insightful. Not only does it make immediately clear where textual variance occurs, but it also indicates different types of variance by means of colour codes (their meaning is explained in a legend). Clicking on a textual variant provides a small pop-up window with additional information about the individual units of variance. It is also possible to look at the TEI/XML encoding of the witnesses. A final interesting point is that the EVT makes use of a set of Java scripts that parse the TEI/XML file directly, without resorting to XSL transformations. In other words, a transcription is not transformed: the relevant information is simply extracted and visualized. However, this only applies to a selected number of text characters and TEI tags. Text characters that are tagged with the <del> element are displayed as such, but that is about the amount of genetic variance that can be visualized.

Figure 48. Selection of the parallel segmentation representation by the EVT.

As briefly discussed in section 2.1 and 2.2, Juxta Commons also offers several visualization options. Among these are a histogram and a heat map, the latter being probably Juxta Commons’s signature visualization. Similar to the MVD, Juxta Commons suggests creating multiple witnesses from one source text if the
latter contains several textual layers. From one source text containing layers of revision, the editor thus creates two or more witnesses: one with the base text and one (or more) with the additions. The two texts are collated against each other and the results are stored in a separate file - a method similar to that of TUSTEP and even, in some way, to the superwitness in the Native XML Collation (chapter 2.3). However, in Juxta Commons it is not possible to revert the process of extracting the textual layers and storing the information in separate files. Given that this section concentrates on visualization, I will not discuss this issue further. Juxta Commons also offers a heat map-visualization, which highlights in blue the places where textual variance occurs. Readers/users can take a closer look by clicking on one of these revision sites, and see the deletions are shown in red and the additions in green.

An additional feature is Juxta Commons’s histogram, which provides a visual overview of the variation (figure 49). The higher the bars, the more textual variation occurs. Since the histogram is intended for readers/users to ‘step back’ and explore the textual variation on a larger scale, it can help to study patterns in variation. Together with a heat map, the histogram provides another example of the productive combination between distant and close reading.

Figure 49. Selection of Juxta Commons’s histogram visualization.
3.2.2.4. Variant Graph

Last in this overview – but surely not least – is the concept of the variant graph. Since graphs are discussed throughout this thesis, it is important to emphasize the difference between a variant graph and a graph as a data structure. A graph consists of sets of nodes that may be connected through (directed) edges. The flexibility of the graph structure ensures that it can store a variety of information and it allows for an extensive analysis of that information (see also chapter 4.2 and 4.3). Conversely, a variant graph is a way of representing textual variance. Here, too, information is represented in nodes and edges that are to be read from left to right. This reading order makes it a directed, acyclic graph: it can be read in one direction only – from left to right – and without ‘looping’ back. All paths through the variant graph begin at the empty start node (left) and end at the empty end node (right). By following one path (i.e., following a certain route over nodes and directed edges), readers/users can trace the tokens of a specific witness.

One of the main advantages of a graph is that it does not impose a hierarchy. If we look at the variant graph depicted below (figure 50) we see different ways to traverse it. In the visualization no variant is given preference over the other. Although textual scholars, more specifically textual genetic scholars, tend not to be familiar with the representation of variance in a graph visualization, the variant graph is a useful format. Because of its non-hierarchical structure, it is easier to visualize layers of writing without preferring one to the other. The graph facilitates recording and structuring non-linear structures in manuscript texts. It provides a comprehensive representation of the alignment, capable of including more information than for instance an alignment table, so it is useful to analyse the collation outcome in detail.

The concept of the variant graph forms the basis of the MVD approach. Schmidt and Colomb argue that a variant graph has an ‘intuitive’ format to represent texts in multiple versions, because its non-hierarchical structure allows for versions to overlap. In fact, the authors assert, the variant graph can represent all known forms of textual variation: deletions, insertions, transpositions and open variants (2009). A number of the MVD visualizations of the variant graph were presented in chapter 2.2. These variant graphs have the witness information (both the text and the sigla) on the edges, while transpositions are represented by means of a dashed edge (a dotted line) between two nodes.
Conceptually, the variant graph of CollateX extends the variant graph proposed by Schmidt and Colomb, but in CollateX’ visualization the information is distributed over the nodes and edges, with the nodes containing textual information (tokens) and the edges containing the witness identifiers (sigla). CollateX also makes uses of graphs for internal processing purposes. In brief, the most optimal alignment for the input tokens is stored in CollateX’ memory in the form of a graph. The internal graph can be outputted in GraphML, an XML-based format for graphs. From this, a variant graph representation can be generated using GraphViz. The current variant graph visualization gives a good deal of additional information: each node contains a small table with the first column showing the normalized and original token, and the right column indicating the sigla of the witness(es) that contain that particular token or text fragment. The sigla are given as labels on the edges. To read the CollateX variant graph, a user can follow the path of a certain witness siglum and study the tokens in their original as well their normalized form. An example of the variant graph visualization can be seen in figure 50 below. The graph shows the alignment of the tokens ‘a b c d’ (witness 1); ‘a c d b’ (witness 2); and ‘b c d’ (witness 3).

Figure 50. Example of a variant graph output with the witness sigla on the edges and the (normalized and original) tokens in the nodes.

Since the variant graphs of the MVD and ColleteX are used either for internal processing or (intermediate) editorial analysis, the design of both applications is relatively simple. A more elaborate variant graph visualization is offered by TRAViz, a set of JavaScript packages primarily intended to generate and visualize variant graphs. TRAViz presents a good example of how the

---

149 The workings of CollateX are outlined in chapter 2.1 and illustrated by means of a practical implementation in chapter 2.3.

150 GraphViz is open source graph visualization software that has several packages for different graph visualizations. It takes as input GraphML files. See [http://www.graphviz.org/](http://www.graphviz.org/) (last accessed February 2, 2017).

151 TRAViz is one of the outcomes of the eTRACES project. See
affordances of digital visualization can inform design, assign a more prominent role to visualization, and include a focus on the end-users. One of its main developers, Stefan Jänicke, has been researching data visualization for some time now, which resulted among others in a set of rules for variant graph output (Jänicke et al. 2014; 2015). These rules are specifically developed to improve the non-verbal communicative aspect of a variant graph, for example by using a larger font size if a token is present in many witnesses (cf. rule 1 ‘Vary vertex label sizes’) or giving each edge a different colour instead of a label (cf. rule 3 ‘Do not label edges’). Moreover, the developers of TRAViz depart from the horizontal scrolling and propose line breaks instead (cf. rule 5: ‘Insert line breaks’). The reasoning behind this last rule is that the readers/users are accustomed to vertical scrolling when working on a computer and, when it comes to non-digital reading, to the arrangement of text on the page of a book. In consultation with humanities scholars, the developers of TRAViz found that this design also reminded its readers/users that ‘it’s a book to be read, not just some string of letters’ (Jänicke et al. 2014). Figure 51 presents an example of this variant graph.

Figure 51. Selection of the TRAViz variant graph visualization of a collation of Genesis 1:4.

Where the variant graphs of MVD and CollateX are non-interactive by design (since they are visual renderings of a collation output), TRAViz lets users interact with the graph and adjust it to match their needs and interests. Users can, for instance, adjust the relative edit distance (see chapter 2.1) to allow near matching or by readjusting the computer-generated alignment.

As mentioned earlier, transpositions present a particular challenge for collation software. Roughly defined as an omission and an addition, transpositions can occur both intradocumentary (inside one witness, in a layer

of revision) and interdocumentary (between two witnesses). While a human reader easily recognizes the relationship between an omission and an addition, collation software generally records them as two independent edit operations. Since this is a complex phenomenon that is not easily computable, in many cases it will be the editor who establishes a transposition by studying the outcome of a first collation. Transpositions also present some original challenges for visualization. They can be visualized in variant graphs as a dashed edge (i.e., a dotted line) between two nodes, but this becomes complicated in case of a considerable distance between the deletion and the addition. An interesting feature of TRAViz is that it visualizes potential transpositions by indicating which nodes are related (here, ‘related’ means that they share one or more tokens). By clicking and zooming, the user can study the relationship between the tokens and determine whether it is indeed a transposition. This interaction with the variant graph and the collation output is similar to Smith’s Collation Editor discussed in chapter 2.2. Since the latter runs in a collaborative editing environment, it is primarily oriented at editors/users who do not wish to engage with the code directly and prefer to work in a User Interface.

In conclusion, we can say that the ‘ideal’ visualization tool would show both variation within a single witness and variation between several witnesses. Existing collation tools do visualize both types of variation, but rarely at the same time. In fact, only Gabler and Schaüble’s synoptic prototype attempts to visually combine the layers and the levels of writing. Nevertheless, the visualization of deletions and additions in the CollateX alignment table implemented in the BDMP, or the combination of the heat map and the histogram of Juxta Commons also deal with intradocumentary and interdocumentary variation. The structure of variant graphs appears especially suitable to record and display multiple levels of variance, but existing outputs are limited to either a variant graph of intradocumentary variation (the MVD), or a variant graph representing the results of a collation of multiple witnesses without their internal variation. The combination of intra- and interdocumentary variation would entail a kind of ‘merge’ of the two variant graphs. A conceptual visualization of such a variant graph was already discussed in chapter 2.2.3 (‘The BDMP: Experimenting with the Collation of Manuscripts’); chapter 4.3 expands on this theory with a practical experiment.

---

152 The latest variant graph output of CollateX does not visualize transpositions, because that information is not recorded in the internal graph.
For the time being, though, we will put aside the topics of automated collation and transcription, and return once more to Brulez’ Sheherezade. The following section discusses the endeavour to display Brulez’ manuscripts and represent the work’s genesis using a type of revision narrative.
3.3. Theories and Practices Applied: Exhibiting Literary Genesis

‘Each editor, in short, should use his knowledge, skill, and judgement to make his contribution to the better understanding of his subject in his own way.’
(Gaskell 1978, 10)

‘Shiriar enjoyed to check the steady grow of these stories and to observe how the writer refined them with the flowery rigging of style ornaments, like a chef who makes a minitature galleon out of a fowl.’
(Brulez 1946, 28, own translation).

Introduction

When it comes to visualization, it is important to keep in mind that the main aim is not necessarily to mimic or imitate the dynamic process of writing; the aim is to provide a suitable environment to study it. This could very well mean that the digital representation is more static than the reality. In fact, it is only reasonable that complex textual situation cannot be represented in a simple and uncomplicated manner. At the same time, the representation needs to be understandable or else it will miss its goal. Over the course of this chapter we have seen an increasing appreciation for the documentary evidence of a textual change on a digital facsimile combined with an immaterial, or ‘invisible’, explanation. Such an explanation can be offered explicitly in the form of an editorial narrative, implicitly by juxtaposing the sentences (collation output), or presented in a complementary critical edition in print. Two additional aspects of visual presentation of textual genesis have also been discussed, namely the act of modelling as a cognitive exercise for the editor (section 3.1), and the revision narrative as a way of communicating the editor’s knowledge to readers/users (section 3.2.3).

Both aspects come together in a virtual exhibition of Raymond Brulez’ Sheherazade, created as a joint effort of the CMG and the web design company Prophets. In the present section, I examine the Brulez Digital Exhibition

154 Prophets is an Antwerp-based company that specialises in web design and information architecture. Among their clientele are other non-profit and cultural heritage institutes (see http://www.prophets.be/; last accessed February 15, 2017). Our contact persons at Prophets were Sofie Desmedt and Kris van Hauwmeieren, but the BDE itself could not have existed without the hard work of their back-end and front-end developers.
(BDE) from the perspective of the editorial team of the CMG, namely Van Hulle, Neyt, Kelly, and myself. The section is partly based on a paper given together with Kelly at the symposium ‘Digital Scholarly Editions as Interfaces’ (Graz 2016). While that paper focused on the role of interfaces in digital scholarly editions (following the theme of the symposium), the present chapter looks at the role of the editors in creating a representation of the writing process. A number of theories provide a useful framework to consider the editor’s role in the making of the BDE. To begin with, the abovementioned concept of revision narratives as proposed by Bryant. Furthermore, we take inspiration from the idea of a ‘genetic path’ described by D’Iorio in the Pearl Model of the HyperNietzsche. This model is based on the metaphor of the editor as a pearl diver who collects beautiful gems. For the HyperNietzsche, each gem – a pearl – is a special element or unit in the archive. Pearls can consist of textual fragments, editorial commentary, or complete notebooks. If a editor/user is interested in the development of a certain word, she can search for all pearls that mention this word, string them together in a chronological order, and thus generate a new sequence called a ‘genetic path’. The paths in HyperNietzsche could be predetermined by the HyperNietzsche editors, or newly generated by scholars with a HyperNietzsche account (Bartscherer 2003, online). A third influence for the present section is the work of Nick Sousanis. He presented his PhD research in the form of a graphic novel to demonstrate how text and images can be interwoven to reinforce each other (Sousanis 2015). His work shows that meaning is not only conveyed by what is depicted, but also through structure and organization. Readers are made aware of a continuous interaction between the linear, hierarchical nature of words and the rhizomatic character of the visual. In a similar vein, HyperNietzsche employs a rhizomatic display to illustrate the interconnectedness of the various genetic paths in the archive.

The concept of a ‘path’ that follows the development of a textual instance also underlies the revision narrative Bryant’s fluid-text theory (2002; chapter 3.2.1 ‘Narratives of Genesis’). The narrative generally starts from a microgenetic perspective, that is, it takes a small textual instance like a crossed-out word. Subsequently, Bryant expands upon the deletion by placing it in a larger framework of the revision history across documents and versions. In describing how the revision site develops, Bryant takes into account – where

155 Although he prefers the term ‘comic’ for his work (Sousanis 2015, 60).
applicable - the exogenetic and epigenetic processes that underlie the deletion. It is with this ‘discourse of narrative script’ that the fluid-text editor guides the readers/users and introduces them to the existence of other versions of the text. In addition to giving the readers/users a pleasurable experience with the historicity of the text, the editor can illustrate the ‘critical relevance of textual scholarship’ (2002, 160). As we saw in section 3.2.1, a revision narrative is presented to the readers/users of the *Typee* edition in the form of a pop-up window that appears by clicking on a revision site. With this unconventional take on the classic form of the critical apparatus, Bryant proposes a means to ‘map out variation, chart paths from one version to another, and enable users to lead themselves along those paths’ (2002, 123). The underlying idea is that the editor guides readers/users and shows them how the text develops, in the process presenting them with interesting discoveries and persuading them to explore the ‘avant-texte’ themselves.

The BDE is at least in concept similar to Bryant’s revision narratives and the genetic paths of HyperNietzsche: here, too, we chart out a path through the textual genesis for the user to follow. One of our principal aims was to showcase the ‘avant-texte’ of *Sheherazade*, to introduce readers/users to the stories the documents convey, and to articulate our analysis of the endo-, exo- and epigenetic processes. Our intentions are fittingly captured by Robert Darnton in his article ‘The New Age of the Book’:

> Any historian who has done long stints of research knows the frustration over his or her inability to communicate the fathomlessness of the archives and the bottomlessness of the past. If only my reader could have a look inside this box, you say to yourself, at all the letters in it, not just the lines from the letter I am quoting. If only I could follow that trail in my text just as I pursued it through the dossiers, when I felt free to take detours leading away from my main subject. If only I could show how themes crisscross outside my narrative and extend far beyond the boundaries of my book (1999).

Darnton is a book historian, but the yearning he describes is well-known among scholarly editors too, whose research brings them to literary archives and boxes filled to the brim with documents. After an in-depth study of the content of these boxes, scholarly editors can form a plausible picture of the history of the text, but how to convey that to the readers/users? In a similar vein, Darnton muses over ways to make available ‘the raw material embedded in the story’ and specifically to provide readers with an awareness of ‘the complexities involved in
construing the past’ (1999). His musings lead him to imagine a layered kind of ebook that contains various sorts of information on the text, from scholarly essays to pedagogical documentation and forms of social reading. They led us to develop the BDE in which we aim to use the appeal of multimedia content (digital facsimiles, images, music, video, and text) to make readers/users our associates, ‘researchers in arms’ as it were, by giving them an awareness of the diverse materials that make up the ‘avant-texte’ and how they can be interpreted.

3.3.1. B917/H2: The Archival Boxes of Sheherazade

We start with the ‘genetic’ part of genetic criticism and have a look inside the boxes containing the documents associated with the collection of stories Sheherazade. These boxes are kept in the AMVC Letterenhuis, the literary archive of Flanders in Antwerp, Belgium. Since Raymond Brulez was active in the cultural scene of Belgium as director of the Belgian Radio and Television and president of a committee to promote Dutch literature, his archive comprises a variety of materials. The documents directly related to Sheherazade have been collected in three boxes with the identifiers B917/H2a and B917/H2bis. For the most part, the boxes have notes and draft manuscripts in the author’s hand, as well as typescripts and page proofs with authorial revisions. Of course, one can find numerous and endless genetic paths within these documents. It is up to us, the genetic editors, to decide where such a path starts or ends. We may want to follow a certain chronology or start at a single textual instance and trace its development. Evidently, every genetic dossier contains numerous paths. We present three of them in the BDE; in the following I will highlight one particular path.

The path in question is called after the frame story of Sheherazade: ‘Wat is liefde zonder verleiding’ (translated to ‘What is Love without Seduction’). We want to show Brulez’ particular way of writing and the importance of the materiality for this process. The stories of Sheherazade were written between 1928 and 1930, during the aforementioned golden age of the contemporary manuscript (chapter 1.1.1). Like many writers at the time, Brulez used the materials to support his writing process. In the terminology of De Biasi, Brulez’ work method can be placed between écriture programmatique and écriture à processus (Fierens 2015; De Biasi 1996). Brulez himself described his writing as a sea voyage. He set out with a certain idea and structure, but he allowed himself
to be ‘blown off course’ during the writing process proper. The genetic dossier of Sheherazade contains a page on which the story’s outline is sketched (figure 52) as well as a considerable number of loose jottings on pieces of paper and on the back of envelopes.

Figure 52. The writing plan of Sheherazade, with the story division indicated with the letters ‘A’, ‘B’, and ‘C’ (AMVC Letterenhuis, B917/H2a).

In length the notes range from a few words to rough paragraphs. When the text of a note is used in the first draft, it is marked with a coloured pencil to avoid using it again. An example of this can be seen in the notes 13 and 28 (figure 53a and figure 53b) that both were used for the story’s first part, which can be inferred from the letter ‘A’ in blue pencil. Moreover, Brulez would cut up certain notes in order to easily reorganize elements of the story. The notes in figure 53 offer proof of the physical aspects of Brulez’ writing process.
The narrative of Brulez’ Sheherazade is loosely based on the famous Arabian Nights tale, where Sultan Shiriar has the women from his harem murdered after spending one night with them. In the original story, Sheherazade tells Shiriar stories, having him long for more and eventually falling in love with her. In Brulez’ – arguably more ironic – version, Sheherazade attempts to assassinate the Sultan, but fails. As he does not want to have her arrested, Shiriar contrives an original punishment. The text on note 22 (figure 53a) has Shiriar musing about the sentence he will inflict upon Sheherazade. His first idea is to make her write out lines (‘a thousand-and-one’; a clear reference to the alternative title of Arabian Nights) but he considers that too light a sentence. Seeing the notebook she carries, he orders her to fill that with stories. The cut-off note 13 (figure 53b) bears only two phrases: in grey pencil it reads ‘17 sept. 1928’ and the words ‘[XXX]’s avonds: de maangolf omspoelde de boeg van een nachtwolk’ (‘[XXX] evening: the moon wave washed against the prow of a night cloud’) and in red pencil the comment ‘Voor Sheherazade begin van Sinbad’ (‘for Sheherazade the start of Sinbad’).

As the story goes, Sheherazade is frustrated with her punishment because she does not know what to write, and she leans out of the window hoping for some inspiration. Looking at a moonlit cloud that resembles a ship, she is reminded of Sindbad the Sailor, and she decides to write an eighth
adventure for him. In a narrative that thematizes the occasional struggle that is writing, the moon cloud-scene describes the moment of inspiration that sets off the protagonist’s writing. The notes thus represent first material traces of a double genesis: for a struggling author and his protagonist. In brief, there is a textual connection between note 22 and note 13 (respectively the sultan’s idea for a punishment and Sheherazade’s frustration over that punishment) and a material connection (their physical place on the document page). By cutting up what was once one page, Brulez separated the two story elements and he rearranged their place in the narrative.

Although note-making was one of the first steps in Brulez’ writing process, the notes themselves do not yet constitute a first draft. They are generally too fragmentary and unordered. Conversely, the pages of the draft manuscript are ordered and numbered and they form a coherent text. Nevertheless, as Fierens has shown in her study of the genetic dossier of Sheherazade, the first complete draft is composed through a ‘creative cut-and-paste’ technique (2015, 29). A telling example of this creative engagement with the materiality of the documents is found in notes 22 and page 18 of the draft manuscript. Figure 54 shows that note 22 is cut in two. The lower half (to which I will refer as note 22-b) is pasted on a larger page that forms part of the complete first draft as MS page B917-H2-18 (figure 55a).

![Figure 54. Note 22-a (AMVC Letterenhuis, B917/H2a).](image)

---

156 What is more, the fact that the note is dated suggests that Brulez, too, may have looked out a window on the night of September 17, 1928 and found some inspiration in the cloudy sky.
It is likely that MS B917-H2-18 of the draft manuscript first was a note. The text contains a significant number of false starts and revisions, and only a very rough version of the text. It describes sultan Shiriar reflecting on what would happen were he to arrest Sheherazade for attempted murder. He makes an inventory of the consequences that turn out far from favourable. In his lively imagination, the sultan envisions people protesting in the streets, the opposition party taking advantage of his weakened position, himself going down in history as a tyrant… Note 22-b adds more unpleasantries to Shiriar’s imagined list. Thus, by combining note 22-b with another note, Brulez expands the text sufficiently to insert the document page in the first draft. This hypothesis is supported by the fact the page number ‘18’ is written on note 22-b, not on the underlying page (figure 55b).
However, the upper half of the note (note 22a) is not lost. When Brulez typed out the draft manuscript into a first typescript, he decided to include the text on note 22a as well (thus extending Shiriar’s reveries of revolution even further still). This method of *bricolage*, which allows for a rearrangement of story elements, is an important step in Brulez’ composition process. The notes provide material traces that transcend document borders. Stringing these traces one after another allows us to reconstruct genetic paths that – indeed – crisscross through the ‘avant-texte’, demonstrating how the textual elements constitute a rhizomatic web of interrelated fragments.

3.3.2. Challenges for the Representation of Textual Genesis

The genetic paths described above are only a fraction of what the ‘avant-texte’ contains, but – as Darnton noted – it is already quite a challenge to describe the textual and material relationships in sufficient detail and with respect for alternative routes. Genetic criticism comes with certain necessary boundaries as well as interesting challenges. First and foremost, it’s nearly impossible to define the origin of a text’s genesis. As Almuth Grésillon states, ‘le début, l’origine du processus d’écriture, est inaccessible’ (2001, 10). What we can establish is a verbal expression of the intention to write, but not the intention itself – and even that expression is often established retrospectively in some autobiographic testimony. Authors often designate a certain moment of inspiration but the original point of genesis usually goes back even further, e.g., to the books read.
by the author, or even to something she once heard. During the reconstruction of the origin of a literary work, genetic scholars need to avoid the pitfalls of source criticism; nevertheless such early sources can be considered in the light of this dialectical relationship with the literary work. The same goes for the end point of textual genesis: when is a work truly completed? Genetic criticism typically takes the bon à tirer moment as a point of completion, but the theory of epigenesis (chapter 1.3) makes clear that a text may continue to develop after publication.

Another challenge is the representation of the temporal aspect of writing. Several attempts have been made to include this third dimension in a digital edition, such as the Teleurgang van de Waterhoek edition (2001, eds. De Smedt and Vanhoutte; chapter 4.2) and the Proust prototype (2012, André and Pierazzo; chapter 3.2). Both require an active engagement of readers/users by letting them access the textual elements from different vantage points (Waterhoek) or by clicking through a dynamic presentation of the writing sequence (Proust).

Given that we want to use the BDE to present the multiplicity of genetic research, we need to pay attention to the five aspects of genetic criticism discussed in chapter 1.3.2: endogenesis, exogenesis, epigenesis, microgenesis, and macrogenesis. Furthermore, we want to emphasize that textual genesis is not a linear process towards a ‘final’, finite state of the text, but a transformative and dynamic process in which each textual state is of equal importance. We can consider these states as interrelated textual elements and explore the network of elements from different angles (thematic, lexical, syntactical, chronological, material). Since the BDE is not a digital scholarly edition, it does not attempt to provide an all-encompassing picture of this network. Instead, it intends to present readers/users with a first impression, touching upon the different aspects of genesis and illustrating them with material evidence from the boxes of Sheherazade.

There is a certain duality in genetic criticism that is also reflected in our visualization of the ‘avant-texte’ of Sheherazade. On the one hand, we try to represent the development of text. Text is linear by nature and can be represented in a sequential manner, but these characteristics do not apply to its development and composition process. When we construct a path, we establish a chronology in the textual constitution and we imply a linearity. On the other hand, we try to capture and convey the nature of thought – both the thoughts of Brulez and our own thoughts – that can be seen as an interconnected
network of associations and fragments, interwoven in a rhizomatic fashion. This duality is not problematic but suggests specific ways of visualization in which the two means of perceiving – linear and rhizomatic – are not mutually exclusive but mutually informative.

3.3.3 Making the BDE: Objectives and Approach

In addition to showcasing genetic criticism in general and the genesis of *Sheherazade* in particular, the CMG had a number of objectives. First was the evaluation of Kelly’s user-study (2015) and general interface design principles (primarily found in the aforementioned Ruecker et al. 2011). Three findings of Kelly’s questionnaire are especially useful for the development of the BDE:

1. Desktop interfaces offer a suitable environment for textual research with complex tools; conversely, tablets present a good medium for outreach publications;
2. Tablets are popular for learning activities;
3. Native apps, that is, specifically designed for tablets, achieve less outreach than applications derived from a responsive website.\(^1\)

We can add to that a number of development principles formulated by Ruecker et al. (2011), namely the idea that a homepage provides an overview of the entire content (3-4), and that the organization of the images should always be meaningful to the users (89). We were keen to develop the BDE as a responsive website that could be easily displayed on tablets and desktop screens alike. The tablet would be placed in the permanent exhibition space of the AMVC Letterenhuis, but the BDE itself would be accessible online from any computer.

Another objective was the collaboration with external parties, which can be seen in the light of finding alternative ways to increase the outreach of our research. Next to a space in their exhibition room, the Letterenhuis offered us a large touch screen device, and high-resolution scans of the material. They gave us complete freedom in regard to the development of the BDE proper. For this, we engaged the web design company Prophets, which had some previous experience with cultural heritage institutions (e.g., they had developed the website of the Letterenhuis). Engaging an external company for the development is not usual in the field of digital scholarly editing. At the same

\(^1\) A responsive website is developed according to the principles of ‘responsive web design’, meaning that the layout and presentation adapts to the device it is displayed upon: it responds to the size of the screen or web browser.
time, we were aware of the statement by Kurosu and Kashimura that the aesthetics of a website ‘promotes the perceived usability and therefore the overall usability of a website’ (qtd. in Ruecker et al. 2011, 13). In other words, when readers/users see a well-designed website they assume it is easy to use and therefore it usually is. The strengths of our research center lie with genetic criticism and not web design, so we felt that in order to reach such a level of aesthetic design we could use some external input.

The collaboration with the developers of Prophets was interesting right from the start, because we were compelled to express our arguments about the genesis of Sheherazade to a willing but uninformed audience. In practice, that meant trying to make the idea of genetic criticism appealing without oversimplifying it, and leaving room for input while adhering to our primary intentions with the digital exhibition outlined above. Explaining the concept of genetic criticism to Prophets turned out to be a good measurement to test how to communicate with a wider public. As outlined in the previous section, the aim of the BDE was two-fold: (1) to describe our understanding of the constitution of the Sheherazade, touching upon the different aspects of the genesis; and (2) to convey the relational, rhizomatic nature of the writing space. The former is by necessity a linear and sequential path, narrated by the editor and based on a chronological interpretation of the creative process. In fact, the MS PowerPoint we used to present these genetic paths to Prophets turned out to work quite well for this narrative aspect. The final design is therefore set up in a similar way by letting readers/users click through the successive stages in a path. As for the networked structure of the writing space, this is addressed by setting up the home page as a messy writer’s desk where, conform the overview-principle of Ruecker et al., all items in the BDE are displayed (figure 56, next page).
3.3.4. Presentation of Editorial Findings

The three different genetic paths in the BDE are indicated in the upper left corner of the home page, which can be accessed at any point. The attention of readers/users is primarily directed to the first path with a start button and the caption ‘Ontdek hoe Raymond Brulez Sheherazade schreef’ (‘Discover how Raymond Brulez wrote Sheherazade’). This first path introduces the writing process of Sheherazade in a general manner, describing Brulez’ moment of inspiration at a concert (thus touching upon the concept up exogenesis) and his conceptualization of writing as a sea journey. The second path deals with the materiality of writing and Brulez’ working method of cut-and-paste (bricolage) as it is described in section 3.3.1. If readers/users click on ‘Start’, they are taken through the narrative step-by-step. Each step is a slide with visual and verbal information: several images and a caption with a short explanation. Figure 57 represents a screenshot from the slide that introduces Brulez’ working method of note taking, structuring them with the alphabetic system, and marking them with different coloured pencils.
In the successive slides, we zoom in on notes 22 and 13 (shown and discussed in figures 54a and 55a-b) and recount the story behind their development, pointing out that the note is cut in two and that both halves are used to form a page of the draft manuscript. The slides go on to show a digital facsimile of the typescript that contains the fragment in question and finally a digital facsimile of the table of contents of the print edition that no longer bears any traces of what happened before. With this last slide, we aim to demonstrate the contrast between the messy, apparently disorderly process of writing and the innocently clean result of that process.

The third genetic path is called ‘De Opstand der Voetnota’s’ (‘The Rise of the Footnote’s) and recounts in detail the genesis of the story by the same name in the collection of Sheherazade, taking into account both the story’s metaphorical meaning and again the material aspects of the writing process. Each narrative of these genetic paths is set up as a linear story with the manuscript fragments in chronological order. If they want, readers/users can choose to be guided by us and follow our premeditated route through the ‘avant-texte’. One can imagine that first-time users prefer this option to exploring the material on their own. However, they are not bound by the constraints of the narrative: they can leave it or access it at any point. The transition between slides, too, does not have a ‘linear’ feel; rather, it swivels and spins from one frame to another. It is one of the features that – ideally – conveys the sometimes undirected process of writing even though the story of that process is told in a linear narrative.
The BDE thus presents three genetic paths in the form of a narrative that can be accessed at any point, used to provide readers/users with in-depth information about the textual genesis of Sheherazade. Each path constitutes a scholarly hypothesis with a minimal threshold. By clicking through the path from beginning to end, readers/users follows along with our editorial argument. Keeping in mind the duality of genetic criticism (mentioned in the introduction to this section), we can say that these genetic paths are visual representations of the linear structure of text, or more precisely, of our argument about its composition. The other side of the medal is addressed by the landing page that, in its simulation of a writer’s desk, represents the networked or rhizomatic nature of the writing space. Here, the visual arrangement expresses an aspect of writing that is difficult to put into words. It is only by stitching together the different fragments that a coherent, meaningful whole emerges.

A last feature of the digital exhibition I would like to highlight is made up by video animations. That is, two of the three genetic paths are introduced by a short animation that ‘sets the scene’ and provides a visual framework for the path in question. On the score of Rimsky Korsakov’s Sheherazade, the two videos show the animated manuscripts of Brulez: letters appear on the page, documents are blotted with ink or marked with red pencil, words are crossed out – in short, the full dynamics of writing. The animations cannot stand alone, because without contextual background information viewers do not understand what they are seeing. In other words, the information presented in the genetic path clarifies the animations, while they, in turn, contribute to the narrative by making the story of the genesis more dynamic. This is consistent with the definition of digital visualization (chapter 3.1) that understands visualization as an important rhetorical device, not supportive to but on equal level with the text.

As in any exhibition (analogue or virtual), the material contained in the BDE is a representative selection of a larger whole. In this case, we as editors have taken on the role of curator. Within the confined space of the exhibition, readers/users are free to roam around or to follow a designated (‘curated’) path that represents our editorial argument. As Sousanis argues, the combination of text and images, of linear and networked, has the potential of engaging the reader in ‘an act of imagination’ (2015, 61). Accordingly, readers/users of the BDE are the actors who animate words and images, and who transform a static presentation into something dynamic. Finally, the visual arrangement and structure of the BDE also attends to distant and close reading as outlined in
chapter 3.1. Distant reading is facilitated by the writer’s desk visualization that provides an overview; close reading by the option to zoom in at any point and experience the text from a linear perspective. Ideally, the interaction of these two means of perception contributes to a fuller understanding of what is at play in textual genetic research.

3.3.5. User Testing, or: Did You Get My Message?

Setting up an environment in a certain way does not mean it will be used as intended. For that reason, user testing and user surveys are an important part of developing an interface. At the moment of writing, the BDE is in the final stage of development but it has not yet been included in the physical exhibition of the Letterenhuis. The following paragraphs discuss the results of our own user testing and propose a number of hypotheses based on findings of previous research. As mentioned before, the BDE is informed by the conclusions of Kelly’s user survey (2015). Where the survey finds that tablets are especially suited for disseminating digital scholarly editions, the BDE is created for use on a tablet as well as a desktop. User tests were conducted both on a tablet and on a desktop. The test audience consisted of native Dutch speakers with and without a background in the study of literary manuscripts.

An interesting finding was that one reader/user did not click on the ‘start’ button in the middle of the landing page, but was drawn instead to the menu in the upper right corner indicating the three genetic paths (figure 56) and clicked on the – admittedly tempting – title ‘Wat is liefde zonder verleiding’ (‘What is Love without Seduction?’). He then accessed this particular genetic path without being introduced to the general background of Shehrazade or Brulez’ work method, which influenced his understanding of the work’s genesis and the communicative strength of the specific narrative. One could say this is the consequence of giving readers/users the liberty to enter the narrative at any point: if its linearity is disregarded, the editorial argument may lose some of its force. At the same time, the associative and intuitive reading strategy that the BDE apparently stimulates might also underscore the creative nature of writing.

At this point it is probably fair to note that previous studies found that, on average, museum visitors spend between 30 seconds and 2 minutes per individual item in an exhibition. Moreover, they are more likely to ‘use trial-and-error methods at interactive exhibits than to read instructions’ (Bachta et al. 2012). Realistically, we can hold the attention of museum visitors for a limited
amount of time only. This may be significantly more for the online readers/users, because they will have made a conscious decision to navigate to the BDE. At the same time, it is possible that the physical environment in which the BDE is accessed influences how it is perceived. That is, we may assume a certain level of interest in literary works and writing among the public of the literary museum. Moreover, the exhibition space of the Letterenhuis, with its open displays of authorial manuscripts and artefacts related to the writing process, may already provide a suitable background and thus inspire readers/users to spend more time exploring the contents of the BDE.

3.3.6. Reflection

It is a rare and precious opportunity for any scholarly editor: creating a digital exhibition of the documentary material in collaboration with a partner specialized in web design. It is furthermore a good exercise in modelling, although in a different way than I have discussed so far. Here, the modelling is in identifying aspects of our research (and research findings) that are potentially of interest for the wider public. It entails presenting our content and our argument in a clear manner without oversimplifying. During our work on the BDE, we had to keep in mind that we wanted to give readers/users an insight in textual genetic research whilst keeping the threshold as low as possible. By letting the user interact with both aspects, thus using the strengths of linear sequences (the genetic paths) as well as a complete overview (the landing page as the writer’s desk), we hope to have created a meaningful representation of the concept of genetic criticism. With a visualization of both the linear aspects and the rhizomatic nature of the writing process, we intend to convey the dualism of the writing process. Overall, we feel that, while the research can be complex, there is no reason why the key findings cannot be understandable and accessible to more than a small community of dedicated scholars.

When trying to classify the BDE, it is easiest to start with a definition ex negativo. It is not a digital scholarly edition in the definition of Pierazzo (see the Introduction to this thesis), although the BDE does have some characteristics of a digital scholarly edition in the sense that it is a fluid publication. Nor is it a derivative of a digital scholarly edition, since the digital scholarly edition of Sheherazade is still under construction. It has been argued that the scholarly edition is no longer ‘an authoritative, final statement’ (Sahle 2016, 29), which implies that, just like the process of writing, the editorial process is ongoing and
produces a variety of products along the way. In line with that hypothesis, the BDE can be seen as a product of the act of editing. It is based on the same texts, materials, and scholarly research findings that underlie a digital scholarly edition and, as a result, can be considered a scholarly product as well.

Finally, the collaborative effort that underlies the creation of the BDE emphasizes that the activities of digital scholarly editors include reflecting on outreach and dissemination. This is also suggested by Pierazzo:

instead of denaturising existing scholarly editions, the involvement of textual scholars in the provision of new editions for the wider public that are able to support leisure reading, as well as to present some of the aspects of textual variance could be a [...] culturally meaningful and responsible approach (Pierazzo 2015, 153).

With the BDE, we attempt to represent the temporal dimension of writing and to capture Brulez’ writing process in a meaningful way. Following the theories behind genetic paths and revision narratives, we intend to convey our hypothesis regarding the genesis of Sheherazade, and to bring to life the documents of its ‘avant-texte’. The approach we followed combines the strengths of different modes of visualization (sequential and simultaneous), using the linear nature of the verbal expression in combination with the relational nature of the writing space. As such, we hope to show readers/users how exciting the content of the boxes in a literary archive can be.
3.4. Concluding Remarks

‘Editions ... give new life in new forms to texts or works from the past.’
(Shillingsburg 2013a)

‘The solution [to information overload] is visualizing information so we can see the patterns and connections that matter and designing that information so it makes more sense or tells a story.’
(McCandless 2012)

In his TED talk entitled ‘The Beauty of Data Visualization’, David McCandless readily admits that if visualizations fail to convey meaning, at least they ‘look really cool’ (2012). He may have a case in point, and it sure is no disadvantage if a digital scholarly edition looks ‘really cool’. However, such a statement also confirms the bias of some scholars against digital visualization. Over the course of this chapter, I have argued that for digital visualization to be more than a pretty picture, for it to be considered of scholarly nature, it needs to have a steering or influential role in the edition. It could for instance further advance analysis of collation output, or communicate information that is difficult to explain in words. A good example is a genetic plan, the schematic overview of the genesis of a work. In a print edition, the genetic plan has primarily an illustrative role, supporting the editorial analysis as explained in an editorial rationale or introduction to an edition. In a digital edition, however, the genetic plan could be made interactive and thus function as interface that provides additional access points into the text. We have seen this with the digital Faust edition and the macrogenetic overview of the BDMP among others.

These examples underline how digital visualization can bring together close and distant reading. Diagrams, graphs, or coloured squares add new perspectives to the various ways we look at text. Each perspective puts another textual characteristic before the footlights, while – ideally – making readers/users aware of the fact that there is still so much more behind the scenes. What is more, certain overview representations point to the existence of textual fragments like paralipomena that were previously ignored or delegated to footnotes, critical apparatuses, or separate publications. Thus, with digital visualization, editors can foreground the multivocal nature of text.

The frequently mentioned possibility to generate various transcriptions from the same source file has a similar effect. The coexistence of multiple transcriptions and a digital facsimile underlines the argument that text is fluid and does not exist in a fixed form. What is more, it addresses a long-standing
question asked by genetic scholars, namely is a transcription primarily intended to ‘donner à voir’ or ‘donner à lire’? (see Lebrave 2014) By placing the transcription alongside – or on top of – a digital facsimile, editors give readers/users the chance to study either the one or the other, or the one in relation to the other. This particular presentation gives equal prominence to document and text, something that can only be partially established in a TEI/XML transcription. As a result, the visual complements the textual, thus promoting a fuller understanding of the contents of a genetic dossier.

Another relevant finding is that visualization is first and foremost an editorial affair. The selected projects discussed here and in the previous chapter show that, even within the genetic orientation, a wide variety of visualization practices exist that are a reflection of the editor’s interests and objectives. Bryant extends on the mediating function of the editor who narrates the revision; the editors of the BDMP experiment with an interactive and dynamic genetic plan; and Gabler and Schaüble draw the attention of their audience to the distinction between layers and levels of revision. Each of these visualizations contributes significantly to the edition’s research potential.

Being quite unique and standalone instruments, the visualizations also raise questions regarding their assessment and evaluation. Keeping in mind the four conditions of digital tools outlined in chapter 1.2 (under ‘Models as Research Output: Assessment’), we would at least need to know the underlying theory and intentions, and verify if the visualization accords with this theory. This seems to justify an editorial rationale or technical documentation that includes a thorough explanation of the visual aspects of the edition in question. Not incidentally, the selected projects all included such a clarification.

In closing, we can say that the digital editing projects discussed here testify of significant advances in the areas of interface, accessibility, and editorial enrichment. This progress is enabled by forming alliances with partners from inside and outside the digital editing community. Based on my own experience working with Prophets on the development of the BDE, I think it safe to conclude that not only do such collaborations result in a creative adaptation of digital technology. They also press scholarly editors to reiterate their preconceptions and refine their argumentation. As a consequence, textual genetic research is taken out of its close surrounding; the field is opened up to influences from design studies, computer science, and information modelling that, in turn, improve editorial methodology and enhance the study of the multidimensionality of draft manuscripts. Undeniably, there is still abundant
room for further progress and developments. One of the main questions propelling this development is ‘how can we as editors successfully model the multidimensionality of the (draft manuscript) text in such a way, that this multidimensionality continuous to play a role in further processing?’ This question forms the subject of the next chapter.
Chapter 4 - Organizing Documents and Versions

4.1. Preamble

‘Such is the power of tools and representations alike: to shape thinking, both through the conclusions they enable and the metaphors they deploy.’

(Galey 2010, 100)

‘The intertextual and relational capabilities of electronic media offer an intriguing set of possibilities for the future of textual criticism.’

(Sargent 2013, 233-4)

Introduction

Digital scholarly editions have emerged as powerful platforms to present the ensemble of digital transcriptions, digital facsimiles, and other related materials. The fact that multiple visualizations can be generated from one digital transcription is, of course, one of the most potent characteristics of digital editing. The previous chapter discussed a number of ways to look at text, for instance through the presentation of a digital facsimile alongside a transcription, or by providing the option to switch between a reading text and a diplomatic transcription (chapter 3.1). Another, equally potent feature is that the individual files can be organized in various ways. In this respect, digital scholarly editions are seen to provide an ideal support for an unobstructed and flexible exploration of their contents. Organizing documents and texts is of course an important task of any type of scholarly editing, but it plays a key role in the genetic orientation, which investigates the relationships between documents, text, and versions. D'Iorio summarizes this point as follows:

Grâce à la souplesse du support, l'édition génétique numérique fait coexister différents systèmes de classement en redéployant les documents selon les intérêts du lecteur qui peut choisir de feuillerter les textes selon la succession des pages ou dans l'ordre chronologique, ou alors de suivre les parcours génétiques (2010, 53 ; see also chapter 1.3.4).

Simply put, this coexistence of various ‘systèmes de classement’ or systems of classification means that the same collection can be studied from different angles. A frequently mentioned classification is documentary (e.g., according to the archival organization of the documents), but one could also think of
thematic, linguistic, or geographic classifications. The flexible infrastructure of a digital edition provides a favourable platform to apply them.

Before we go on, it is useful to distinguish between a system of classification and an infrastructure. The term ‘infrastructure’ can be rather vague, or at least difficult to define. Within the context of digital scholarly editions, infrastructure primarily indicates the entire architecture or set-up of the edition. It can also be defined as ‘the resources – as personnel, buildings, or equipment – required for an activity’ (Merriam-Webster). Expanding on that definition, we can see how the integration of external software such as CollateX into an edition creates an infrastructure for digital collation. With ‘system of classification’, then, I mean the various ways in which the contents of the edition can be arranged. This arrangement is supported by the infrastructure. For example, CollateX can be used to collate all sentences from a teleological perspective; or a system of hyperlinks can be used to produce the genetic paths that present a thematic classification.

Classifying material begins on a conceptual level, in the first phase of genetic editing, together with a transcription and organization of the material in the genetic dossier (see chapter 1.1.2). It is part of the editor’s traditional work: a task that can be carried out independent of the digital medium. Subsequently, the edition’s infrastructure is set up to support the actual organization of the documents. In this respect, the creation of systems of classification entails a form of modelling: editors need to formulate in detail their underlying assumptions about the chronological or teleological development of the text, and then construct a digital environment in which readers/users can explore them. Although readers/users may discover novel things on their wanderings and explorations, the editor remains undeniably present in the technical organization of the edition.

If we look at the components of a genetic digital edition that are identified in section 1.3.4 and that form the basic structure of my thesis, we notice some inevitable overlap. Both ‘visualization’ and ‘classification’ entail different ways of looking at text; while ‘collation’ and ‘classification’ provide ways to structure the contents of the genetic dossier. In fact, collation is a

---

158 In a research context, infrastructure is most commonly defined as the underlying framework as the basis of a system or organization. For instance, we often speak of research- or institutional infrastructures (e.g., ‘NINES’ stands for ‘Networked Infrastructure for Nineteenth-Century Electronic Scholarship’ and DARIAH stands for ‘Digital Research Infrastructure for the Arts and the Humanities’).
method to classify material from a textual perspective. Nevertheless, I take ‘classification’ as an individual topic, because it entails storing and providing access to the information produced by the other components. The preceding chapters have described the wealth of information contained in digital transcriptions, produced by automated collation, and extracted by digital visualization. Systems of classification are a way to organize this information in a flexible manner. As such, they allow readers/users to take advantage of the full potential of the digital edition as a medium and as a knowledge site.

**Genetic Classifications**

Chapter 1.1.2 defined the ‘avant-texte’ as the result of a first analysis and classification of the ensemble of documents that is the genetic dossier. But in what different ways can these documents be classified? Since the genetic orientation deals with documents and versions, a documentary and chronological ordering seems evident. Nevertheless it is useful to take a closer look at what a ‘genetic classification’ could further entail. In ‘Collation de réécritures: l’avant-dernier texte bilingue de Samuel Beckett’ (2008, 234-5), Van Hulle describes how genetic material could be organized either from a teleological perspective or, conversely, from a chronological perspective. Teleology, from the Greek ‘telos’ meaning ‘end’, comes down to the idea that there is meaning or purpose behind a process, and studies phenomena by the purpose they serve, starting from the end result. For scholarly editing this translates to the selection of textual fragments that have made it into the ‘final stage’ of the text.\(^{159}\) The opposite would be a chronological classification, which focuses also on the fragments that did not reach this final stage. In the electronic Waterhoek edition, this classification is described as ‘ontological’ (Vanhoutte 2007, 163). A good example is the ‘cul-de-sac’, an impasse in the writing process, or an unachieved version that represent a certain moment in the genesis.

Furthermore, as D’Iorio suggests in the citation above, the genetic dossier can be ordered by means of a thematic ordering based on ‘parcours génétiques’ or genetic paths. The material is then arranged according to the

---

\(^{159}\) The teleological classification has long been the primary way to organize a collection of text. It may be clear now that scholars working from a genetic orientation to text do not believe text actually has a ‘final’ stage. A teleological ordering, therefore, rather implies selecting a later version, for instance the *bon à tirer* – text or the first publication, and using it as a temporary ‘final’ version.
development of a certain theme, such as the concept of natives in *Typee* or the motif of the gramophone in *Sheherazade*. A thematic ordering is thus similar to the genetic paths discussed in chapter 3.2 and illustrated in chapter 3.3. Finally, the nature of the edited work in question plays a role as well. Beckett, for instance, wrote in English as well as French, and often translated the texts himself. The elements of language(s) and (self-)translation are therefore highly significant for the genesis of his bilingual works, e.g., *Stirrings Still/Soubresauts*. It would make sense, in this particular case, to include in a digital edition a means to organize the transcriptions according to original text and translation (Van Hulle 2008, 235). A second example can be found in the research of Ronan Crowley who examines how location influenced the writings of James Joyce. Here, it is not hard to imagine the additional value of a location-based classification of Joyce’s documents.

In conclusion, the ways to classify and organize a text are influenced by the nature of the text and directed by the findings of editorial research. We can distinguish four classifications or ways to access a genetic dossier: (1) **documentary** – following the classification of documents according to the archive that holds them; (2) **teleological** – starting from a published version (a ‘final’ stage) and working back through the preceding versions; (3) **chronological or ontological** – starting from the first writings and working up to a ‘final’ stage (e.g., a published version if there is one); (4) **thematic** – according to specific characteristics of the material and the nature of the work.

**Outline**

The goal of this chapter is to examine how the multiple systems of classification of textual and non-textual materials can promote research into textual genesis. The chapter’s structure corresponds to Chapter 2 and Chapter 3, starting with an overview of the state of the art (4.2 ‘Theories and Practices’). This overview will show that most genetic digital editions provide a number of structural ways to access and organize their contents. In general, editors seem to have little difficulty with modelling on a conceptual level, but all the more with the technical realization of their model(s). Much comes down to the separation of form and content in the encoding. A related challenge is the combination of the

---

160 In the FWO/Pegasus research project *James Joyce’s Geographies of Reading: Trieste, Zurich, Paris*’ carried out at the CMG (University of Antwerp, Belgium).
archival and the editorial impulse. Therefore, the section also looks at the technological aspects of providing an infrastructure for systems of classification, from knowledge sites to modelling text as a network. It comes as no surprise that most genetic scholars find the idea of a web of interrelated textual fragments quite appealing, and that they were among the earliest supporters of the theory of hypertext. I will take a closer look at the graph data structure and specifically the concept of Text as a Graph (TAG; briefly mentioned in chapter 2.3). The main advantage of graphs is that, unlike XML, they do not impose a hierarchy. Therefore, the graph structure would be especially convenient to model the genetic dossier as a network of non-hierarchical and discontinuous elements.

Chapter 4.3 discusses a practical implementation of these theories, taking the genetic dossier of Sheherazade as a case study. In a small-scale experiment, the TAG technology is applied to a particularly challenging sentence that contains overlapping structural elements like a sentence break. The sentence is encoded in TEI/XML from a genetic orientation, with a focus on text but including aspects of the material document. I will discuss how the use of these structural elements can improve a textual classification. Finally, the section places the experiment within a framework of digital philology and modelling. The experiment, albeit small and partly theoretical, demonstrates how both modelling and hands-on computational research can make a positive contribution to textual analysis.

Paraphrasing Kaltenbrunner, we can say that digital editing depends on the creative uptake and adaptation of technological possibilities (2015, 189). In the domains of text modelling and digital editing, no single technology is to be preferred over the other because each has its own advantages and drawbacks. At the same time, certain knowledge of and flexibility towards the use of technologies ensure that we continue to explore new directions of textual research. For instance, although the idea of text as an interconnected network comes naturally to textual scholars, most of them are trained to model text according to a hierarchical structure, which in turn shapes their thinking and their practice. In designing and creating different sets of classifications, the editor can (re)consider the ways in which the edition’s content is interconnected and related. Each choice made here reflects the ways in which the materials are imagined, while it also influences the future use of the edition. In other words, the editorial choices and perspective can also be found in the overall structure of a digital edition and the ways to access its content.
4.2. Theories and Practices

‘Organising our received humanities materials as if they were simply information depositories, computer markup as currently imagined handicaps or even baffles altogether our moves to engage with the well-known dynamic functions of textual works.’

(McGann 2007b, 211)

‘Is it possible to think that today’s research is ushering in a new golden age of criticism accomplishing yesterday’s dream of achieving the harmony of contraries?’

(Hay 2004, 25)

Introduction

When he talks about ‘the harmony of contraries’, Louis Hay refers to the many complex and contradictory processes of writing, namely the internal and external forces that shape a literary work and are covered by endo-, exo- and epigenesis. We can think of the genetic dossier as a collection of seemingly unmoving material that needs a few good stirs and some reorganization before it reveals the manifold stories it holds, before the contrary forces can be studied. McGann and Hay are not the first or the only scholars who welcomed the digital medium as a powerful means to study the multiple dimensions of draft manuscripts. Indeed, D’Iorio notes that ‘la plupart des spécialistes s’accordent à reconnaître que seuls les supports et les technologies numériques peuvent permettre la réalisation d’une véritable édition génétique et garantir sa diffusion auprès du public’ (2010, 49). Grésillon describes the ‘précieux’ contribution of the computer that makes it possible, for the first time, to represent in full the dynamic aspect of the writing process by interrelating the various stages in the text’s genesis (2001, 14). In the same vein, Gabler considers the digital edition ‘a superior base of operations for genetic criticism’ (2010, 51).

Let’s start by taking a closer look at the structural foundations of a technological framework that could facilitate this. At the basis of digital editions is a database of some sort. This database can be referred to as a text base or a digital archive, but the principle remains the same. The database of a genetic edition contains at least the digital transcriptions, most probably combined with digital facsimiles, and other non-documentary material related to the text’s genesis – in short: it contains a digital representation of the genetic dossier. A database needs to be set up in such a way that the genetic dossier can be easily queried, so that a reader/user can ask for instance ‘give me all the additions in a hand different from the author’ (cf. the SG A) or ‘show me the development of
this particular sentence’ (cf. the BDMP). In the previous chapters, I claimed that markup influences the way in which transcriptions are processed by software and tools. Since systems of classification are an alternative means to query and display a selection of the digital content, it follows that there is also a direct relationship between markup and the potential ways in which transcriptions could be classified and organized.

A first thematic classification can quite easily be produced by means of the search function of a digital edition. This is mainly a question of structurally tagging specific phenomena in the digital transcription. Examples of this are drawings, doodles, the Geolocation information of places mentioned in a literary work, or (fictional) people based on named entity recognition. Examples of edition projects that offer such functionalities are the Van Gogh Letters (which lets users/readers search for Van Gogh’s sketches), the BDMP (where one can specifically search for Beckett’s doodles) and the Kundige Bok (which provides a number of themes that recur in the text). Now each individual file or element in the database needs to contain structural information and/or metadata based on which a system can query and organize them. The relationships between all textual fragments may need to be hard coded in the transcriptions, e.g., by numbering the sentences vis-à-vis a base text as in the practice of the BDMP. The same goes for a documentary organization, a thematic organization, and so forth. In other words, editors have to think about any potential classifications before they start to encode the materials.

As I mentioned in the previous section, editors and textual scholars can easily organize a genetic dossier on a conceptual level. I distinguished four classifications: documentary, teleological, chronological or ontological, and thematic. For reasons of convenience, ‘teleological’ and ‘chronological’ may be grouped together under ‘textual’. Since the distinction between text and document is fundamental for all editing projects, this allows us to include some projects that are not necessarily created on the basis of a genetic orientation to text.
4.2.1. Documentary and Textual Classifications

Linking Documents and Versions

Most scholarly editors think it makes perfect sense to consider the content of an edition from a textual perspective as well as a documentary perspective. At the same time, encoding these systems of classification can lead to issues with overlapping hierarchies similar to those discussed in chapter 2.1. I concluded then that most editing projects follow an encoding practice that gives preference to either text or document. At this point, we can take a closer look at some attempts to encode ‘all’ textual and ‘all’ documentary features and how that decision influences the organization of the edition’s content.

The project Textual Communities (TC) makes an interesting case to start with. TC, created by Robinson and Bordalejo, is an online editing platform where editors/users can upload their documents, make digital transcriptions using a TEI-based tagset, and carry out other editorial tasks like automated collation. By design, TC allows its editors/users to encode all features of a manuscript in XML without resulting in conflicting hierarchies. This basically means that the project attempts to realise the holy grail of digital text editing, an accomplishment they are surprisingly modest about: ‘along the way’, writes Robinson, ‘TC solves the notorious “overlapping hierarchy” problem which has bedevilled the making of scholarly editions in digital form for more than two decades’ (Robinson 2012a). The project instantiates several of Robinson’s principles or ‘desiderata’ for scholarly editions, the most relevant principle being that ‘a digital edition should encode both the text of the document and the text of the work’. It is difficult, Robinson concurs, but it can and should be done (2012a; 2013a). Accordingly, TC is founded on two ‘pillars’: Documents and Text (or ‘textual entities’), granting both pillars equal importance. With the technical assistance of Xiaohan Zhang, TC developed an encoding approach in which both documents and texts are subdivided into smaller units. For instance, a document can be divided into pages, columns, lines; and a text into books, chapters, verses, etc. These units are subsequently linked to each other by means of the TEI element <refsDecl>. Let’s take a look at a snippet of a TEI/XML transcription from the project’s website:

```xml
<lb/>
<div n="Book of the Duchess">
<head n="title">The Boke of the Duchesse</head>
```

241
I haue grete wondir
be this light

The first line ‘I have grete wondir’ is given the following identifier:


In this identifier, ‘document’ and ‘entity’ (indicated in bold) respectively link the unit to the second line of folio 110v of the Bodley manuscript and to the first line of verse in the Book of the Duchess. With the TEI attributes @prev and @next, lines of text are connected. All units and their unique identifier are stored in the TC-database. So when a reader/user requests to see the text of the work Book of the Duchess, an API queries that database using XPath expressions and extracts the relevant units, and orders the sentences based on their @prev and @next attributes. Taken together, this encoding approach implies that entire text of the Bodley document and entire text of the Book of the Duchess can be generated.

Although TC claims to be able to deal with all sorts of documents, one of the prerequisites of using their approach is that the document and the text it contains must be representable in XML (Robinson 2012b). This condition is rather ambiguous. As I have argued in this thesis, it is near to impossible to capture in XML the sum of all dimensions of the draft manuscript. Thus far, genetic editors have had to make concessions or find workarounds when editing a genetic dossier of a work. It is not clear whether Robinson means that TC is able to work with files made under these concessions, or whether the project actually proposes an innovative encoding system that enables its editors/users to encode not only the document and the text on the document, but also a reconstruction of how the text appears on the document. One indication the TC would not be able to easily process draft manuscripts is that their Text reference system is currently based on the premise that the text on the document can be divided into lines. But, as Gabler asserts, a typical draft document cannot be divided into lines without considerable difficulty and interpretation: ‘every manuscript page [...] offers a writing space not already pre-
determined by expectations of linear text reading’ (Gabler 2013). A possible solution comes from Dillen, who suggests that the TC encoding system can be
adjusted to handle something like the TEI element <zone> (Dillen 2015, 127). Until this – certainly commendable – effort has been made, however, TC appears to be especially suitable for texts and documents that can be structured in a straightforward manner.

In order to better frame TC and consider its potential for genetic editing, let’s take a look at an electronic edition that was made about a decade earlier and follows a similar approach. The electronic edition of De Teleurgang van den Waterhoek (TW) presents, in the words of its editors De Smedt and Vanhoutte, ‘an excellent case for textual and genetic criticism’ (2002, 2). It offers its readers/users a number of tools to reconstruct the genetic history of the Flemish novel De teleurgang van den Waterhoek (Streuvels, 1927). Because it combines approaches from different editorial theories (‘documentary, historical-critical, diplomatic, facsimile, study and reading’), the edition can be defined along the lines of Pierazzo’s paradigmatic edition (2015, 26). The editors’ intention is epitomized by the following phrase: ‘a work develops from version to version and it has become the editor’s task to render this evolution as clearly as possible’ (Vanhoutte and De Smedt 2002, 2). They propose a so-called ‘linkemic approach’ to visualize how text develops over time. In this approach, a large number of ‘linkemes’ (i.e., the smallest unit of linking; in this case a paragraph) are connected in a hypertextual environment. Readers/users can annotate a linkeme and even cross-reference it to other linkemes they might consider related. The internal structure of these linkemes is achieved through the structural markup of the Xpointer syntax (Vanhoutte 2000). This Xpointer syntax also provides a ‘bidirectional linking’ that is necessary to jump back and forth between linkemes. In that respect, there are significant correspondences between TC and the linkemic approach of TW.

Let’s take a closer look at the structure of the linkemes and the way in which their relation is explained to readers/users. The edition opens with a ‘dynamic-interactive table of contents’ that provides links to the four main parts of the edition; the rationale, the hypertext edition, the documentary sources, and the diplomatic edition of the correspondence (Vanhoutte 2000). The editors aim to present the user with a ‘navigational system’ to traverse the edition (cf. Shillingsburg, qtd. in Vanhoutte 2000). The editorial commentary narrates the textual genesis and is enhanced with a number of notes and hyperlinks to ‘digital facsimiles of relevant material from the Streuvels archive’ (2002). Another section describes the (linear) textual transmission history of the document material and links to the location of these sources.
Although the editors state that the rationale includes the genetic and transmission history of the text, a base text, and an explanation about the text constitution, Van der Weel laments that ‘this introduction fails to explain what sort of an edition the “electronic-critical edition” really is, and leaves the user with many questions’ (2002, 346). As said, the edition can be considered paradigmatic, but apparently it does not make clear why some of the witnesses are presented in different formats, e.g., diplomatic, clear-text critical, facsimile (346). Vanhoutte indirectly responds to this in a later contribution, where he explains that their ‘underlying rationale was to enable the reading and study of multiple texts’, which would stress ‘textual qualifications such as variation, instability, and genetic (ontologic/teleologic) dynamism’ (2006, 163).

The main value of the edition lies in the possibility to ‘display all the variants of each paragraph in all six of the versions on the screen’ where they can be ‘orderly consulted’ (Vanhoutte 2000). In this sense, the edition offers a way to present the text of a work regardless of document-boundaries, indeed, to organize the material from a textual perspective. Additionally, the linkemic approach also supports a thematic organization of the genetic dossier. That is, variants are juxtaposed to the orientation text in a pop-up window. In this pop-up window the user/reader can click on a particular variant, which leads to a digital facsimile of the complete document of that variant. Vanhoutte argues that ‘this linkemic approach provides the user with enough contextual information to study the genetic history of the text, and introduces new ways of reading the edition’ (2002), but if this is at all confusing, there is a manual that explains the possible routes using illustrations. Van der Weel, however, found that ‘the edition does not alert the reader when choices have been made by the editor, nor does it direct the reader to the alternative readings’ (idem 348).

In conclusion, Vanhoutte states that although the edition is ‘far from perfect’ it did challenge the conventional editorial theories and allowed him and Marcel De Smedt to explore other possibilities. Discussing and analyzing the failures of their model brought them closer to generating a better one (2006, 164). This better model would include, among others, a step away from print editorial conventions such as a static presentation of variants. Overall, we can consider Vanhoutte’s argumentation in the same light as Pierazzo’s pleas for ‘more daring textual scholarship’ (2015, 204) that digital philology would instantiate: a type of editing that allows room for modelling and experimental editions. The creation of TC can surely be regarded along those lines.
Standoff Markup

It is an appealing idea: one digital archive with genetic material from which a variety of visualizations and classifications can be generated. As Van Hulle and Neyt point out, this idea is based on Theodor Nelson’s principle of transclusion, meaning that ‘the same piece of information can have meaning in a variety of contexts and in each of these contexts the shared data can be retrieved without duplicating them’ (BDMP’s Editorial Principles and Practice 2016). Accordingly, Vanhoutte and Robinson were far from the only ones to take up the task of developing a technological infrastructure to support such a digital edition. Dillen describes how Robinson was inspired to create TC after a research stay at the De Montfort University in Leicester together with Shillingsburg and Paul Eggert (Dillen 2015, 123). While Robinson decided to address XML’s overlapping hierarchy-problem by breaking down the text of the document and the text of the work in small units, Eggert and Shillingsburg opted for a different approach: standoff markup.

The idea of standoff markup is that the markup is stored separate from the data that is described by it. The TEI Wiki on standoff markup suggests that ‘the source text can be left as read-only (and thus secure and possibly even located on a remote server) and the markup that describes it can constitute a separate layer, linked to the original by appropriate pointers’.161 This notion of a secure, read-only source text fits the idea of an archive of documentary transcriptions to which editorial knowledge and scholarship is applied in a later phase. An additional advantage of standoff markup is that it theoretically solves the problems with overlapping hierarchies. Theoretically, editors could add as many layers of markup to the base transcript without them overlapping; markup is stored elsewhere and only called on when needed.

Eggert’s take on standoff markup resulted in the development of an editorial ‘WorkBench’, instantiated by the Australian Electronic Scholarly Editing project (AustESE). Work on AustESE has ceased, but originally it was conceived as an online platform consisting of several modular components, much like the bricks approach discussed in chapter 1.2. The starting point was to understand the edition as a project: ‘a collection of digital surrogates of original documents, such as page-images and transcriptions, but also digital records or metadata about physical artefacts, such as books, as well as derived

classes of entities such as versions and works’ (Schmidt and Eggert 2015). Transcriptions were made in plain text format and annotated in the form of standoff markup. Editors/users could apply the markup on the fly (or ‘Just In Time’; the name of the markup language) and generate the view of their interest.

After AustESE ended in 2013, its principles were reviewed and in some cases adjusted, resulting in the Ecdosis project. Ecdosis makes a sharper distinction between back-end tools (for editors/users; namely transcription and text-to-image-linking) and front-end tools (for readers/users; namely visualization and annotation). One notable change in the transcription tool is the shift from transcriptions in plain text to transcriptions with MML (Minimal Markup Language). In this case, MML is a simple tag set used to encode ‘structural features’ of the document, e.g., line breaks and italics (Eggert and Schmidt 2015). The transcripts are subsequently stored and processed as an MVD file, the workings of which were discussed in chapter 2.2.2.

Ecdosis is similar to Shillingsburg’s concept of a ‘knowledge site’. The latter defines a knowledge site as a set of editing tools integrated in a modular infrastructure and based on a Content Management System (CMS). The Woolf Online project discussed in Chapter 3 is one example of a knowledge site. It makes use of a CMS called ‘Mojulem’, specifically created according to Shillingsburg’s principles, and it works with transcriptions that have a minimal amount of embedded markup for textual features only (such as ‘strike through’ or ‘pencil’). Any additional editorial analysis is stored in standoff markup. As Shillingsburg explains, allowing for a minimal form of markup on the textual properties ‘respects their standing as essentially textual features’ and, accordingly, ensures that the textual features are taken into account during collation (Shillingsburg 2013c).

The word ‘annotation’ in digital text editing can be ambiguous. On the one hand, annotations are traditionally understood as editorial notes to the text (cf. Mathijsen 2003, section 12.3.1.1). They are usually short (foot)notes used to explain or illustrate lexical terms or provide a short historical background. Digital editions can have this kind of annotations, which usually take the form of a clickable item or a pop-up window. ‘Web annotations’ on the other hand, take this principle and extend it to an infinite user base, basically allowing ‘everyone to annotate everything everywhere’, thus creating a connected network of annotations to the World Wide Web. Of course, this interoperability required a standard, a model, and an ontology (see https://www.w3.org/annotation/; accessed on March 15, 2017). Finally, annotations have come to be synonyms for markup elements like TEI tags, which does make sense if you consider they are editorial notes to a text. In the last sections of this chapter, 4.2.2 and 4.2.3, I use the word ‘annotation’ especially in that context.
An elaborate discussion of the pros and cons of standoff markup for genetic editing is not within the scope of this chapter (or thesis). Instead, I like to focus on one particular disadvantage of standoff markup for genetic editions. Both the AustESE/Ecdosis WorkBench and the Mojulem knowledge site take as point of departure a basic level of agreement upon the base text: an ‘indisputable transcript’ (Dillen 2015, 134) or uncontested layer of base text. However, draft manuscripts are often too complex and require significant editorial interpretation, making it hard to reach the required levels of objectivity required for such an indisputable transcript. Revisions that are clearly made at a later stage can indeed be stored as standoff markup to a base text transcription, but what are the consequences for instant revisions, the corrections that are made during the writing of a sentence and immediately corrected (currente calamo)? Obviously, leaving the revisions out of the base text is not an option. But if we include them in the base text without any markup to flag them, we would miss a significant phase of the writing process. A simple example may suffice. Let’s take a look at figure 58 below, showing a sentence from Brulez’ manuscript of Wat is liefde zonder verleiding.

Figure 58. Selection of MS B917/2a-3 (AMVC Letterenhuis B917/2a)

The sentence starts in the upper left corner with ‘Tegen In het avonduur den namiddag wanneer de’. While we could still argue that the deletion of the word ‘Tegen’ and the addition of the word ‘In’ took place in a later stage of revision, the deletion of ‘het avonduur’ quite clearly happened currente calamo, because Brulez continued writing on the same line. The words ‘den namiddag wanneer de’ could therefore not be marked as an addition; rather they are part of what would be called the base text. In short, and as I discussed more extensively in Chapter 2, the flattened sequence of characters that is plain text does not suffice to capture the manifold dimensions present in a draft manuscript. Nevertheless, the idea of using standoff markup remains promising, if we understand it in the sense of externally stored annotations to text. I expand upon this in a later section of this chapter (4.2.3).
Finally, when discussing knowledge sites we have to keep in mind the observation Vanhoutte made a few years ago with respect to the TW. He pointed out that the concept of knowledge sites as defined by Shillingsburg would most probably not work for Dutch or Flemish authors. Where the names of Virginia Woolf or Walt Whitman are sure to engage a large reader base, Vanhoutte concludes that ‘the idea of the active involvement of a computer-literate and critical community with a knowledge site built around a modern Dutch or Flemish text is but an idle fantasy’ (2010, 120). Since the work of a Flemish author (Raymond Brulez’ Sheherazade) is also the case study supporting my thesis, his comments may prove to be relevant here.

A Textual Classification

Thus far, we have seen two methods to encode and structure the features of text and document: the first method divides material into small units or linkemes that are subsequently cross-referenced; the second method uses standoff markup on a minimally encoded transcription. The projects discussed share several important principles. Most relevant for the present discussion is that they agree that both textual features and documentary features need to be recorded as extensively as possible, without giving preference to one over the other. Moreover, they envisage an online platform that offers a number of analytical tools that can be used to query or analyse an underlying database of digital material related to the text, such as transcriptions, facsimiles, metadata, etc.

At this point, it is interesting to look at a third approach that is based on a text-oriented encoding scheme, but that also takes into account how certain textual segments cross document borders. The module Krapp’s Last Tape / La Dernière Bande (KLT) of the BDMP is a good example. Each module of the BDMP is dedicated to a specific work, with the digital archive containing transcriptions and facsimiles of the ‘avant-texte’ of that work, and the accompanying book providing scholarly essays related to the work and its genesis. In general, Beckett’s work method was relatively structured in the sense that he often dedicated specific notebooks to a certain work and used them consistently. Nevertheless, in some cases a notebook contains several texts related to different works. This is the case with the notebook called ‘ETE 56’ (MS-UoR-1227-7-7-1), which contains textual fragments of several works, namely Fin de Partie, All That Fall, Krapp’s Last Tape, Pim (which would later become How It Is), and Willie-Winnie (which would later become Happy Days). At the
moment of writing, KLT is the only work on this list that is available as a BDMP module. Since the BDMP is an ongoing project, the transcriptions of the other works will be made available in due time. Figure 58 reproduces the table of contents of the notebook ETE56. Beckett indicated the type of text ('scraps', 'first draft', and 'notes') next to the title of the work in question.

Of the ninety or so folios of the notebook, a total of forty-five folios contain writings, of which only folios 10v to 20 and folios 24 and 25 contain the first draft of KLT. However, readers/users can leaf through the digital facsimiles of the entire notebook. When they find themselves in the module of KLT, the BDMP assumes their primary interest is in KLT. Accordingly, only the transcriptions of that specific work are available. This means that if readers/users find themselves on a folio of notebook ETE56 that is not related to KLT, certain functionalities and tools to further study the genesis, such as the double page view, the image/text view and the text view, are switched off. Upon mouse over, readers/users see a message saying ‘This page does not relate to the genesis of [KLT]. The relevant pages for [KLT] are 10v-25r’. An advantage of the current situation is that there is less risk of confusing readers/users who, looking for the text of KLT, are not suddenly confronted with the text of another work, say Fin de Partie. Still, notebook ETE56 as a whole also remains available as a documentary unit.

In conclusion, the BDMP presents a valuable alternative take on dealing with the text/document dichotomy. They may give preference to the textual
perspective in their encoding scheme, but the infrastructure of the online archive enables readers/users to study the document as well. This approach of the BDMP is consistent with the observations of Gabler. About the study of draft manuscripts, Gabler states that we can only gain a full understanding when the writing and its material support are presented as an essential and inseparable unit (Gabler 2013). Although he primarily refers to the detailed study and close reading of an individual draft manuscript, I believe his argument can be extended to include a broader perspective, that is, to see the set of documents as one physical object, and to study how its material reality – in this example, a notebook – relates to the textual inscriptions it carries.

Teleological/Chronological Classification

For a final example of the textual classification, we turn to a different module of the BDMP: Stirrings Still / Soubresauts (SS). I mentioned earlier that when editors classify the content of a genetic dossier from a textual perspective, they could follow either a teleological or a chronological/ontological point of departure. The former means that they take the final structure of the text as leading; the latter means that they follow the chronological genesis of the text. Taken together, these two points of departure ensure that theoretically all textual fragments of a work are included. For instance, Van Hulle points out that by classifying the genetic dossier of SS according to the chronological development of the text, the editors came upon three unachieved parts of the work, presented in the module as ‘Abandoned Sections’ (2008, 237-8). They are visible in the ‘Chronology’ view on the work, which is basically a genetic plan (figure 59).
These textual fragments have never been published, but they are now made accessible in the SS module of the BDMP. The readers/users can select a manuscript of the abandoned sections and study it with the usual tools (text view, image view, text/image view, zoom, etc.). Van Hulle describes this as ‘étudier les culs-de-sac’: by selecting the last version of the unachieved segment, readers/users examine its development retrospectively (2008, 238). Only the option ‘Compare Sentences’ is not available, since this takes the first published version SS as directive to the sentence numbering: since none of the sentences made it into the published text, they cannot be included in the ‘Compare Sentences’ tool.

The technology behind ‘Compare Sentences’ brings us right to the teleological classification. This option groups together in the ‘Synoptic Sentence View’ all sentences that made it into the first publication, so that the readers/users are presented with an overview of how each sentence developed up until the final stage.\textsuperscript{163} As described in chapter 2.2.3 and chapter 3.2.2.2, BDMP’s method of sentence numbering also facilitates a detailed automated collation, allowing readers/users to select two or more sentences and compare

\textsuperscript{163}As said earlier, the use of the adjective ‘final’ does not imply that the text’s genesis has reached its end. It is rather used for reasons of convenience: the BDMP includes only the non-published material like draft versions and page proofs of Beckett’s works, and ‘excludes epigenetic material, such as later annotated editions’ (Neyt and Van Hulle 2016). Accordingly, the ‘final stage’ simply means the last version before publication.
them with CollateX. In his discussion of these functionalities, Dillen notes that ‘they disregard the document’s physical structure of pages and zones in favour of the text’s internal structure of sections, paragraphs, and sentences’ (2015, 146). Indeed, readers/users can follow the development of a sentence across document borders, which indicates that the genetic dossier is primarily organized from a textual perspective.

Overall, these features illustrate that the chronological and the teleological classification are distinctly textual. Readers/users can lift the inscriptions off of the document page and consider them solely in the light of textual genesis. At the same time, the BDMP aims to present the continuum from text to document (Van Hulle [forthcoming]) and, as a consequence, the document is always just a mouse-click away. It is in this respect especially that the project corresponds to Gabler’s statement that digital editions must organize information about the materiality of document and inscription in ways that allow readers/users to study near-simultaneously the genetic succession of textual elements and their spatial distribution on the material document (cf. Gabler 2013). One significant exception is the notion of a ‘virtual document’, for instance the books in the personal library of an author. The physical copies may have been lost over time, but if they are virtually reconstructed the contents of the library can still promote exogenetic research. Veronica Bālā at the CMG carried out valuable research in this direction, reconstructing Beckett’s student library (Bālā 2015). Another example is the project Melville’s Maginalia Online that aims to virtually reunite the works read by Melville (Olsen-Smith, Norberg and Marnon 2008; see also Bleeker 2015b). For a genetic reading of text, the materiality of the document is highly significant, but as the previous chapters have shown, the challenge is to create a digital textual model that does justice to both material and textual aspects.

**Genetic and Thematic Paths**

The digital genetic edition, as it moves along the document-text continuum, is driven by an archival as well as an editorial impulse. Yet it has an additional focus, namely ‘the actions of invention as implied by the chronological succession of documents and the transformations within and between them’ (Shillingsburg and Van Hulle 2015, 36). Textual genetic scholars look at a document’s materiality primarily to distill from the inscriptions the various layers of writing and revision. Through this kind of close manuscript research,
they aim to understand ‘the sequence in time of the filling of the writing area’ (Gabler 2013); and ‘the sequence of discoveries, inventions, and actions of the writer(s)’ (Shillingsburg and Van Hulle 2015, 36). The choice of words of Gabler and Van Hulle/Shillingsburg is not coincidental, and points to one of the main responsibilities of genetic editors: to place textual elements in a sequential order. Throughout this thesis we have seen instances of sequentiality in genetic editing, from flattening and de-flattening text in a transcription (Chapter 2) to visualizing the appearance of writing in sequence (Chapter 3). These instances confirm that there is always an interpretative aspect to genetic editing; a moment where editors guide the reading of readers/users.

The Waterhoek edition (TW; above) illustrates how such editorial guidance can be realised in a thematic organization of a genetic dossier. TW intends to ‘offer the users the possibility to organize their own visual collations of the versions they were interested in from an orientation text of their own choice’ (Vanhoutte 2007, 163). With the linkemic technology to support this objective, the editors apply certain hypertext technologies. First and foremost, they build an infrastructure that supports the cross-referencing of textual variants, taking the paragraph as unit of comparison (i.e., the linkeme). A principal structure is pre-established and hardcoded by the editors, but readers/users can construct additional genetic paths of their own by defining bidirectional hyperlinks between textual fragments or digital facsimiles. Although the user-added hyperlinks cannot override the editorial organization, the readers/users can save and share their hypertext structure as a ‘Personal Webfile’, a feature intended for teaching and similar pedagogical activities. Incidentally, this feature is for a large part supported because the TW makes use of an SGML browser, which facilitates linking and annotating fragments.164

The HyperNietzsche project (HN) presents a second example of an alternative organization of a genetic dossier that is inspired by the characteristics of the material in question. In fact, the parallel between Nietzsche’s use of ‘aphorisms’—short textual units that are organized in a larger, interrelated network—and the principle of hypertext is so striking, that it would be inconceivable had no-one picked it up:

Of variable length but always relatively brief, [the aphorism] is a single unit that can stand independently, and yet is also deeply integrated into

---

164 Many thanks again to Vincent Neyt for pointing this out to me, and for explaining the workings of SGML, of which I have only second-hand knowledge.
the structure of the [work as a] whole; this unit is linked to the aphorisms that proceed and follow it, and is often linked by allusion or thematic concern to aphorisms in other texts (Bartscherer 2003).

Aphorisms are thus an ideal starting point to experiment with a hypertextual structure. HN was set up as a ‘research hypertext’ and conceived by D’Iorio in 1996. Unfortunately it has been discontinued in its original form, but its design and specifically developed markup language (HyperNietzsche Markup Language; HNML) have in fact influenced digital editing to a large extent, especially the Encoding Model for Genetic Editions created by the WG GE, which that informed the present chapter 11 of the TEI P5 Guidelines. Although HN has officially evolved into the NietzscheSource project, the latter lacks several of the features that make its predecessor such an interesting case of digital genetic editing.

One of those features is the aforementioned Pearl Model, which specifically echoes many of the principles of hypertextual technology mentioned earlier. ‘Pearl’ is defined as an element, like an aphorism or a draft manuscript. Moreover, one pearl can contain multiple pearls: a draft manuscript may contain a number of aphorisms, each of which constitutes a separate pearl. Finally, a pearl can also be the definition of a link between two or more pearls. Each pearl has its own, unique identifier. This structure sounds perhaps confusing, but in essence it means that (1) editors or readers/users are entirely free to define their own understanding of a pearl; (2) pearls can be joined together in ‘strings of pearls’ that can be constantly reordered and rearranged (Bartscherer 2003). This particular aspect of hypertext is especially relevant for genetic editing, for it presents a flexible tool to organize textual fragments of varying size, and to structure the genetic dossier according to any aspects of interest to the editor. Since these new paths need to be somehow stored in the HN database, they were most probably only available for the HN editors or registered readers/users.

Also of notable interest is the HN’s visualization of the rhizomatic structure of a genetic path. As I mentioned in chapter 3.3, the concept of a rhizome is quite effective to conceptualize the character of the writing space. One example from HN shows the genetic development of a specific aphorism (indicated as aphorism 338) and demonstrates how ‘various genetic paths cross, diverge, fork, conclude, and connect to other paths’ (Bartscherer 2003; figure 60).
Evidently, the editors construct these paths by placing elements (‘Pearls’) in sequential order. Readers/users can click and follow these paths, and are thus presented with an editorial reading of the genesis. As such, this Pearl model is an implementation of D’Iorio’s genetic paths and constitutes a blueprint for any later theories on the subject. Gabler, for instance, took some inspiration from the HyperNietzschean ‘sub-segmentation’ of manuscripts when working on the manuscripts of Joyce’s *Ulysses*. His findings confirm that models of one specific writer/text are not necessarily applicable to any other writer/text. That is, Nietzsche’s aphorisms do not interact with their documentary writing space in the same manner as texts of Joyce. The genetic paths of HN could easily disconnect text from its material carrier, because the draft aphorisms were not defined by their original documentary context. They are not Ferrer’s ‘protocols for making a text’ but rather autonomous elements. In Joyce’s case, on the contrary, there is a relationship between the drafts of *Ulysses* and the document they were written on. Gabler therefore decided on an alternative structure, which rather resembles the Proust prototype of André and Pierazzo (2012; see chapter 3.2.1 ‘Visualizing to Advance Genetic Analysis’). Intending to present the creative moments in Joyce’s compositional writing process in a dynamic way, Gabler’s model of a double-spread manuscript page has the individual textual fragments appear in a certain order on the document (Gabler 2013).

Taken together, the projects discussed in this section demonstrate how the nature of text and the work method of the author influence the editorial
approach. At the same time, they show that, despite technological advancements over the years, the conceptual ideas behind these projects continue to be relevant. The TW and HN, for example, share a similar principle: not only do the editors set up a framework of thematic paths for the readers/users to explore, but they also provide readers/users with tools to add information to existing data. These enriching and personalizing functionalities emphasize the continuous evolution of text as a network of interrelated discourses.

4.2.2. A Network of Text

Hypertext and Semantic Networks

As was pointed out in the introduction to this chapter (4.1), genetic editors were naturally drawn to the ideas behind hypertext and they experimented with the technology early on.\textsuperscript{165} Indeed, concludes Lernout, ‘the metaphor of hypertext has helped the généticiens to formulate more efficiently what genetic criticism can mean to the study of texts’ (2013, 73). The fundamentals of hypertext – at least as they are usually understood – facilitate the incorporation of a variety of material (including non-textual media), and the display of intertextual and intratextual relationships.

Another, significantly different method for linking textual segments can be found in a younger field of research: the area of semantic networks. Semantic networks are usually championed as an ideal way for representing complex and large-scale collections, which may be the reason why they are as of yet little used in the field of literary studies. Nevertheless, the theory of the semantic web offers an interesting perspective on our challenge of analysing and representing textual relationships. A semantic network can, for example, construct

‘an overall vision of inter-documentary relationships in the corpus. It supports navigation, query and reasoning operators that lead the user towards his goal. It also allows the creation of a network of high-level concepts that represent the users’ understanding’ (Project INRIA)\textsuperscript{166}.

\textsuperscript{165} The volume Texte et ordinateur. Les Mutations du Lire-Ecrire (1991) presents a useful overview of some of these early practices in France, namely the contribution of Jean-Louis Lebrave ‘L’hypertexte et l’avant-texte’ (pages 103-119), recounting the experiment Ferrer and Lebrave carried out with the HyperCard software.

\textsuperscript{166} Project INRIA https://raweb.inria.fr/rapportsactivite/RA2010/gravite/uid73.html, last accessed January 14th, 2016.
Not incidentally, semantic network theory echoes the concept of the literary
work as a web of interrelated texts and contributed to the idea of the scholarly
edition as a relational or structured database.

Schmidt also points to the value of the network model for text (2006;
Schmidt and Colomb 2009). In short, he argues that structuring variants as a
networked database could solve the problem of representing the variants
between texts. The main focus, Schmidt argues, should be on the encoding and
structuring of the variant witnesses; the visualization of textual relationships and
the representation of the structure are secondary issues of interface design
(2006). Together with Schmidt, Yin Liu and Jeff Smith assert that ‘a network
[allows] more possibilities for connections to be made between the elements of
the structure’ (2008). With regard to labelling paths, the latter stress that ‘the
meaning of a connection between two elements is attached to the connection
itself’ (idem), that is, not to the elements.

Both contributions address the problem of overlapping XML
hierarchies, generally agreed upon as one of the main disadvantages of encoding
texts in an XML-based model. The approaches of Schmidt and Liu and Smith
are potentially very interesting, yet they have considerable implications. To
begin with, the use of a relational database would mean a rigorous departure
from XML. And, while the debate about addressing overlapping XML
hierarchies with standoff markup is ongoing, so far no conclusive alternatives to
XML have been proposed. The alternatives might properly address the problem
of overlapping hierarchies, but they do not necessarily measure up to XML’s
advantages in other areas (see e.g., Pierazzo 2015, 121-2).

This is also the case for David and Sandra Schloen’s proposal to make
more use of databases for text encoding and processing. In short, the authors
consider ‘document-markup solutions to the problem of overlapping hierarchies
awkward workarounds’ (§35) and propose to abandon the single character string
representation of text in favour of an atomized database system in which both
textual relations and other observations can be stored. In line with Liu and
Smith, the Schloens state that databases are suitable for certain structures within
texts (2014, §5). Furthermore, they argue that the current text encoding
methods (i.e., TEI) are closely related to the document paradigm, which results
in the all-too-familiar problem of overlapping hierarchies. The solution
proposed by the authors – who both hold a degree in computer science – comes
down to a database linked to a hierarchical ontology. This database system is
comparable to standoff markup, but makes use of an ‘atomised and keyed
database’ (§31) over the more common ‘single-character-string representation’ of text (§33).

It is a promising yet technologically complex solution. Schloen and Schloen implicitly agree when they identify as one of the main advantages of XML that it is intuitive and relatively easy to use, even for people with limited programming skills (which would be most present-day digital editors). Indeed, Pierazzo notes that learning a high-level programming language or gaining a solid understanding of RDF OWL or a database language might be a lot to ask, or at least quite ‘demanding’ (2015, 121). It is safe to say that most of the digital editing community makes use of XML, often following (a subset of) the TEI Guidelines to encode their texts. Even if a relational database structure were the most convenient or comprehensive way to encode the relations between texts, the fact that the last three approaches do not use XML is a major disadvantage. However, this methodology is especially useful to large text corpora, but less so for the genetic orientation to text. The main reason is that a relational database requires structured data, whereas draft texts and certain markup (like &lt;delSpan&gt;) are unstructured data.

All practices make use of a system of links, creating a network of interrelated textual and/or documentary fragments with unique identifiers. In different ways, they have set up or implemented the concept of a digital archive consisting of a digital transcription and a digital facsimile, and a set of digital tools to navigate and contextualize the material. Consciously or not, in doing so they act in accordance with the original concept of hypertext. Based on a study by Vanhoutte (2000), we can summarize this concept in four principles:

(1) The hypertextual environment consists of nodes (e.g., fragments or documents). The data in these nodes have structural markup, for instance XPointer anchors, to enable a system of exact linking;
(2) Multiple views can be generated from the same node, because the data is encoded with structural markup and processed by different stylesheets (e.g., XSLT or CSS);
(3) The links between the nodes are bidirectional, which means that the anchors encoded in the data work both ways, functioning simultaneously as a start and an end reference;
(4) The environment supports link classification, that is, the user should be able to label links and discern between relevant and non-relevant links.
The current World Wide Web (WWW) is loosely based on these principles, but does not strictly adhere to them. The main point of divergence is the third principle of bidirectional linking. When Tim Berners-Lee created the WWW (launched in 1990), he judged the implementation of bidirectional linking as unwieldy and restrictive. Instead, he opted for a combination of HTML and HTTP that supports simple, unidirectional links, giving users the freedom to link everything to anything without requiring permission. Because of this system, the Internet has been able to evolve into the magical place it is today.

While the open environment and unidirectional linking system were essential for the Internet’s success, it was not what hypertext inventor Nelson originally envisioned. In fact, he considers the structures of sequential and hierarchical data imposed by strongly embedded markup absolutely ‘impeded’ and finds that they ‘add obstacles’ to the ‘exact representation of human thought, especially that thought put into words and writing’ (Nelson, qtd. in Buzetti 2002, 74). Indeed, the attentive reader may have noticed that many of the issues present-day editors are confronted with when digitally modelling human thoughts and the act of writing echo Nelson’s critique of a sequential and hierarchical representation of data. Genetic criticism did not only take inspiration from early hypertext theory; it also encountered similar setbacks at the technical implementation of this theory.

The editorial practices described thus far demonstrate that the appeal of modelling text as an interconnected network continues to exist. Assuredly, as Vanhoutte points out, hypertext is definitely not the ‘sole answer’ to the concerns of digital genetic editing (2000). Nevertheless, I believe we can assume that a combination of several of the hypertextual features provides a suitable environment for digital genetic editing: a closed system of bidirectionally linked nodes, each with a unique identifier; the possibility to group and ungroup the nodes when desired, plus the option to further identify the relationship between certain nodes.¹⁶⁷ Many of these features are, at least theoretically, part of the Text as a Graph (TAG) model. Therefore, the last part of this section considers the technology underlying the TAG model in more detail and from a genetic orientation to text.

¹⁶⁷ We can say, for instance, that two nodes are both part of the TEI element <p>, and that in the physical document they are placed in a certain order.
The idea of modelling text as a graph – especially text of the anarchical kind – has come up before. Chapter 3.2.2.4 described the graph data structure sets of nodes (sometimes called ‘vertices’) that may be connected through directed and undirected edges. Each node represents an object; it is up to the user to define that object. In textual editing, a node may represent a (string of) text character(s), but it could also be a TEI tag or a stage in the text’s genesis. Given that a graph does not impose a hierarchy, and thus allows for multiple layers to overlap, a variety of graph models for text have been proposed before. The graph used in these models is known in computer science and mathematics as a DAG: a Directed Acyclic Graph. Like any graph, it is a collection of nodes and edges; here, one edge connects one node to another in a directed manner. Hence, in the example DAG below (figure 61) overlap is possible, but it is not possible to start in the node \{x\} and follow the edges to end up at node \{x\} again. The arrows indicate that the edges are directed.

![Figure 61. Example of a Directed Acyclic Graph (DAG) with the root node Ø as starting point.](image)

A well-known example of a DAG implementation for text editing is the GODDAG model that stands for General Order Descendant Directed Acyclic Graph (Sperberg-McQueen and Huftfeldt 2000). The model presents a data structure similar to XML but that can deal with overlapping hierarchies because it allows ‘multiple parenthood’: a node can have multiple parents. Although GODDAG generally is accepted as a powerful data structure, it is difficult to apply to the practice of text encoding (see e.g., Moore 2010; Middell 2013) and not widely adopted by the digital editing community. Another example is GrAF, a ‘Graph-based Format for Linguistic Annotations’, which as the name suggests focuses on linguistic annotations (Ide and Suderman 2007). Finally, the WG
GE discussed the concept of a Genetic Graph to model genetic relations. They describe how the objects represented by the nodes could be documents, fragments, text stages, or textual alterations. The edges of a Genetic Graph represent a genetic relationship between the two nodes. As in the hypertext structure described above, the edge (or link) can be given a type. Figure 62 shows an example where the directed edges are represented as arrows. The type of relationship is that of predecessor - descendant: node A and node B are precursors of node C, etc.

In general, a DAG is a logical choice to model non-hierarchical textual entities, especially if the graph is used to model and visualize information about the genetic relationships. In that sense, the graph’s nodes and edges are not unlike genetic paths through the content of a genetic dossier (cf. the WG GE 2010). Furthermore, a DAG provides for many of the hypertextual principles outlined above, namely determining objects of varying size, defining their relationships, grouping them in a non-hierarchical order. But these functionalities may be supported even better in a different kind of graph structure, which we can call a HyperGraph (HG). Although HGs are used extensively in mathematics and computer science, their application in the domain of text modelling is experimental at best. The following discussion is informed by ongoing research at the Huygens ING and therefore subject to change. Nevertheless, a HG is already of interest from a conceptual perspective as it builds on the characteristics of a DAG and adds some qualities that are specifically valuable for the modelling of unstructured data.

One of these qualities is that a HG’s edges can join together any number of nodes (instead of the one-on-one relationship in a DAG). For that reason they are called hyperedges. Because a hyperedge joins together a set of nodes in the HG, we can also say that a hyperedge is a subset of nodes. We can furthermore distinguish between hypernodes (that form part of the subset
defined by a hyperedge) or nodes in the HG that are not part of a hyperedge. At this point a simple example of a HG visualization is probably welcome:

![Figure 63. An example of a hypergraph visualization consisting of seven nodes (v1 – v7) and four hyperedges (e1 – e4). Source: Claudio Rocchini (2008) via WikiMedia Commons.](image)

Each hyperedge in this HG is indicated with a different colour. Hyperedge e3, for instance, forms a subset of three hypernodes (v3, v5, and v6). Node v7 is not part of a hyperedge and therefore not a hypernode. This visualization clarifies that the HG structure is essentially about ‘sets’ or ‘collections’. Nodes are grouped in sets; sets can overlap. A node can thus be part of different sets at the same time. HGs can have directed edges and undirected hyperedges. Undirected hyperedges contain one set of nodes, and directed hyperedges contain two sets of nodes: one for the source nodes and one for the target nodes. The direction of the (hyper)edge implies the order or the hierarchy of the nodes.

The visualization of ‘real-world’ graphs is a challenging operation and almost a profession in its own right, since they concern highly complex and vast sets of data. The more information the graph contains, the heavier its cognitive load and the more difficult for readers/users to comprehend the matter. Hypergraphs, therefore, only exist in the memory of the computer, which means that figure 63 does not show a HG, but a visualization of a HG. The HG in figure 63 contains only a small number of nodes, but in theory the edges can link n nodes to n nodes. If each of those relationships would be visualized with a single directed edge, the number of lines would result in an entangled and confusing web. Hence, it makes more sense to go for a kind of visualization that does not show the edges, like the one in figure 63. Other HG visualizations exist, but they are primarily used in computer science and mathematics; to date no visualizations exist that work with literary texts.

---

168 See for instance Junghans 2008 for a comprehensive overview of the complications of (hyper)graph visualization.
A HG structure can help to model a digital text with overlapping markup. A good way to illustrate this is by means of an acrostic: a piece of text of which the first letters of each new line spell out a word or message. Let’s take a look at a short poem by Edgar Allen Poe, fittingly entitled ‘An Acrostic’ (1829):

Elizabeth it is in vain you say
‘Love not’ – thou sayest it in so sweet a way:
In vain those words from thee or L.E.L.
Zantippe’s talents had enforced so well:
Ah! if that language from thy heart arise,
Breath it less gently forth – and veil thine eyes.
Endymion, recollect, when Luna tried
To cure his love – was cured of all beside –
His follie – pride – and passion – for he died.

The first letters of each line spell out the name ‘Elizabeth’. The letters L.E.L. on line 3 refer to English poet Letitia Elizabeth Landon (1802-1838). In all likelihood, ‘Zantippe’ refers to Xanthippe, the wife of Ancient philosopher Socrates: Poe appears to have misspelled her name for the sake of the acronym. The text can be marked up in a number of different ways, but let’s assume we want to do at least some Named Entity tagging. This entails marking the words ‘Elizabeth’, ‘Zantippe’, ‘L.E.L.’, ‘Endymion’, and ‘Luna’, as well as the acrostic itself. A sketch can be found in figure 64 below:

Figure 64. Simplified example of the different Named Entities that can be tagged in an acrostic poem.
What we are looking for, ideally, is a way to express information about the
acrostic ‘E L I Z A B E T H’ in a similar way as information about ‘Zantippe’ or
‘Endymion’. The TEI Guidelines do not suggest a specific encoding for
acrostics, but a likely method would be to work with anchors, e.g., wrapping
each first letter in an element with an <xml:id> that refers to the acrostic. With
the help of some lines of XSLT code we would then be able to extract the letters
of the acrostic. This is, in fact, the downside of capturing discontinuous
elements: we end up with nine elements (one element per letter) that have to be
chained together, whereas the acrostic actually consists of one element proper.

When we read a text from left to right, the continuous sequence in
which the characters are ordered is meaningful and conveys information to the
readers. The meaning of an acrostic, however, does not depend on this
continuity; in fact, it depends on a discontinuity. An acrostic is a particular
stylistic exercise, but examples of discontinuity are plentiful. What we need,
then, is a way to express both continuity and discontinuity in text. Since this
will inevitably lead to overlap in markup, it also needs to be expressed in a data
format that allows for overlap. At the moment of writing, the IT department of
the Huygens ING is developing a repository that uses a data and processing
model that combines these two features. Although the project, entitled
Alexandria, is still under development, it brings together a number of the issues
discussed in this section. It offers a virtual environment consisting of
interconnected nodes (or objects) that can be labeled (or ‘marked up’ or
‘annotated’). These annotations, in turn, are also objects that can be annotated,
so that the data structure theoretically supports an infinite number of
annotation layers to be added to the text. Alexandria is based on a HG model
and ensures that both continuity and discontinuity can be expressed. The
remaining paragraphs of this section take a closer look at the project and the
data model behind it.\footnote{The discussion is informed by many conversations with Ronald Haentjens Dekker, the lead
developer of Alexandria, and his team at the Huygens ING. The background material Haentjens Dekker created for the NEH Institute Make your Edition (taking place from 10-19 July, 2017 at
Pittsburgh University) was at the moment of writing not available, but will in the near future
provide a reference page (see https://github.com/Pittsburgh-NEH-Institute/Institute-Materials-
2017/blob/master/alexandria/data_and_processing_model_specification.md; last accessed
March 18, 2017).}
Alexandria and TAG

Alexandria is primarily intended as a text annotation repository: it stores annotations on text. Here, annotations are understood in the sense of markup. The current data model is a hypergraph structure. Consequently, it consists of a number of objects that can be divided into nodes, edges, and hyperedges. There is a wide variety of node types, consistent with the wide variety of information contained in a digital transcription. The types are listed below:


b. Text nodes. Text nodes present segments of text.

c. Empty Markup nodes. Empty Markup nodes represent empty elements in a text like `<pb/>` or `<lb/>`.

d. Markup nodes: Markup nodes represent tags/markup present in the text of the document.

e. Annotation nodes. Annotation nodes represent metadata on markup node. As said, the model allows annotation nodes to be connected with other annotation nodes or, in other words, it allows editors/users to structure or markup the annotations to tags.

The model also defines different kinds of edges. Keep in mind that edges are used to connect nodes. Corresponding with the variety of nodes, there are different types of edges. All edges are directed, which means that they consist of a start node (the source) and an end node (the target):

A. Document-to-text directed edges. A document-to-text directed edge connects a document node (source) with the first text node (target) contained in that document.

B. Text node-to-text node directed edges. The text nodes themselves are connected with directed edges, which order them, so that the linear order of the text is preserved.

C. Markup-to-text nodes directed hyperedges. Markup hyperedges connect the tags/markup nodes (source) to text nodes (target) tagged with the markup in question.

D. Annotation-to-text node directed edges. This edge connects annotation nodes to the first text node contained in that annotation. The term ‘text’ is ambiguous here: it refers to the text of the annotation, not the text of the source document.
E. Annotation directed edges. This edge can be split in two functions: (1) it connects annotation nodes with other annotation nodes; (2) it connects an annotation node with a markup/tag node.

*Alexandria* supports files encoded in XML or in LMNL. Currently, the developers at the Huygens ING are working on an experiment in which a text encoded in LMNL is represented in the TAG model. The choice for LMNL is based on two characteristics of the markup language that accord largely with the idea of graphs: first of all, LMNL works with ‘ranges’ that can overlap. Secondly, these ranges can contain annotations containing text, which in turn can be annotated. It is not difficult to imagine that this data model is the dream of editors (cf. ‘Encode all the things!’) but that it also results in extremely detailed and complex transcriptions.

At the moment of writing, the experiment has not yet advanced enough to be included in this present study. Without a concrete example of how the model can be applied to a text, it is difficult to estimate the potential of the TAG model. Nevertheless, the short description above of the various components of the model already has a number of considerable implications. First, the model differentiates between document and text, treating them as individual entities that are related by means of a directed edge. Secondly, by connecting text nodes to one another with a directed edge, the sequential (continuous) order of text is captured. Third, the use of the markup-to-text directed hyperedges ensure that n markup nodes can be connected to one text node, and vice versa. Most importantly, in the *Alexandria* system, text and annotations (i.e., markup, editorial annotations, and metadata) are modeled on the same conceptual level, so both text and annotations can be processed and queried in similar ways.

This approach finds theoretical backing in statements by Gabler, who identified the editorial commentary and annotations as ‘essential strands in an edition’s set of interrelated discourses’. In fact, he argues, they need to be considered as equal to and interrelated with the textual introduction, text, and textual apparatus ‘not in serially additive arrangements, but in functional interdependence’ (Gabler 2010, 46-7). He goes on to assert that ‘the content of

---

170 LMNL is a markup language for text encoding, developed by Wendell Piez. LMNL is different from XML in the sense that it does not impose a hierarchical structure on a text. Instead, it works with annotations to arbitrary ranges of text. These ranges may overlap. Another important feature of LMNL is that the annotations themselves can be structured and annotated. For more information, see [http://lmnl-markup.org/](http://lmnl-markup.org/) (last accessed April 19, 2017).
the commentary and annotations would be treated as articulated text in its own right’ and envisions that

Submitting the apparatus and commentary contents to grid connectivities and search patternings would allow bi-directional linkings, say, of any given note with any other note of correlatable content within the body of the edition, as well as with the edition’s backbone, the edition text (Gabler 2010, 49)

Before we go on, it is important to point out that Gabler’s definition of ‘annotation’ is not identical to the concept of ‘annotation’ in the TAG/LMNL model. Where Gabler refers primarily to editorial commentary, LMNL annotations are comparable to XML attributes. They provide extra information about markup tags, and can contain text themselves. This text in turn can contain markup, which can contain text, *ad infinitum*. At the same time, it seems that the TAG model could realise many of the features Gabler outlines in his theoretical model of a digital scholarly edition. That is, TAG would provide editors with a powerful tool to model and structure their knowledge about a text in such a way that the knowledge is not only stored, but can also be used for further processing. What is more, the fact that each node can have multiple edges (and each edge a different type) allows editors to express relationships in detail. The text and annotations can be structured either in a hierarchical order (as linear text), or as networked entities. So not only can this rich data set be extensively queried, it is also possible to export (a selection of the) data in different file formats (e.g., TEI/XML) that may then be processed further.171

In summary, the past two decades have seen a growing trend towards the combination of a database and a set of editing tools combined with editorial knowledge (i.e., a knowledge site or digital research site) as a way to integrate the archival impulse and the editorial impulse. It is felt that in order to take full advantage of the potential of digital technology, editors need a system to store and express information about text. Such a system would need to support the encoding of the continuous as well as the discontinuous aspects of text, without leading to conflicting overlapping hierarchies. This section reviewed a number of noteworthy projects that each present an original and innovative take on that particular challenge. The projects share the aims to generate multiple views on the same (set of) data, and to offer their readers/users multiple ways to classify

171 See for more information the project description of Alexandria at the Huygens Institute’s GitHub: http://huygensing.github.io/alexandria/ (last accessed March 18, 2017).
and organize the text represented by these data. Early hypertext theory and, later on, graph theory appear to support many of these requirements. The data model behind the Alexandria project is promising in that respect, as it implements a HG structure to store complex information about text, taking into account continuity and discontinuity, overlapping hierarchies, and editorial annotations. Users can query the content of the repository and export the data in several formats.

These conclusions again confirm the value of modelling as a research instrument. When they use the TAG model — or any other model discussed above — editors are compelled to reflect upon what aspects of the draft manuscript they want to capture. Assuming that, for genetic editors, it is essential to capture various aspects of the document as well as the text, to record the relationship(s) between them, and to express the continuity and discontinuity of textual elements, the TAG model appears to have potential for text modelling. With respect to thinking of text as a graph, this may be confusing for textual scholars who only just got used to considering text as a tree. All the more since the actual graph only exists in the computer’s memory and it is difficult to create an informative visualization. In that case, it may be helpful to conceptualize the graph model by thinking of layers, since genetic editors are already familiar with the idea of multilayered text as well as the concept of markup layers or annotation layers. Each layer in the model, then, can be seen as an object in the graph. This object can be connected to one or more other objects, starting with the source document and mounting up to annotation layers on annotation layers, etc. The acrostic in figure 64 is an example of how annotation layers can be applied on a microlevel, but the layers could just as easily be applied to a larger data set.

Finally, the advantages of a graph to model and store information about text do not imply that editors must cease to use XML for text modelling — on the contrary. Following Eide, we can conclude that ‘keeping XML as a part of the information system, adding non-hierarchical modules as needed, is a better way forward for text encoding in general and for TEI specifically’ (Eide 2013, §5). As I emphasized in Chapter 2, it is important to be aware of the pros and cons of each format for text modelling. XML has proven itself a valuable and powerful technology, but sometimes the imposed hierarchy can be a limitation. Being aware of this limitation, and of the existence of other technologies that are more flexible when it comes to hierarchies, grants the editors more freedom — both conceptually and practically.
4.3. Theories and Practices Applied:

Two-Step Alignment

‘Attempts to see provisional or temporary relations and even harmonies in the complexity are not to be abandoned because perfection is not available to us.’

(Shillingsburg 2006, 23)

‘We see straight away, as our forebears tended not to, that we need to make a deliberate effort to understand how it works.’

(Eggert 2013, 97)

Introduction

The previous section analysed a number of ways to store information about text (e.g., with embedded markup, standoff markup, or in system of layered annotations). While they use different technologies for storage, all methods aspire to facilitate the organization of the data according to various classifications. The genetic orientation to text appears to benefit most from a thematic, documentary, and textual classification (subdivided into ontological and teleological classifications). Since the latter concerns textual versions and variation, automated collation is a useful tool. We can assume, therefore, that collation is primarily a textual activity. At the same time, we have established that the strict separation between document and text is difficult to maintain in any TEI/XML transcription, and especially in a genetically oriented transcription, since our study of the draft manuscript is so clearly informed by a combination of both features. A digital genetic transcription thus permeates the boundaries of document and text, of structure and semantics, containing the editor’s analysis of both.

Chapter 2.3 discussed an experiment with using certain information encoded in a TEI/XML transcription to influence the alignment in order to achieve a better collation result. Instead of extracting text characters from a TEI/XML transcription, we decided to give equal importance to its structural features. In other words, we collated entire XML trees against each other. The experiment is a good illustration of the characteristics of modelling: we abstracted a small selection that we took as representative of a larger issue – collating draft manuscript text with several layers – and customized the algorithms of CollateX, all the while carefully documenting every
transformation the text underwent with each iteration. It took several attempts before reaching a satisfactory outcome, namely a superwitness with a correct alignment of the linearized token sequences of both manuscript witnesses. However, the last step - transforming this superwitness into an XML tree output - was complicated because the program interpreted the superwitness’ edit graph differently than expected. In short, the program took the document’s structure as leading instead of the textual content. This led to an alignment result that did not match our expectations.

The present section revisits the experiment of chapter 2.3 and tries to make use of additional information in the TEI/XML transcription to produce a collation output that is foremost text-oriented, but uses information about the document structure when it may improve the alignment. It shows how a textual classification can benefit from documentary information, and thus it diffuses the distinction between text and document even further. The working title of the new experiment is a ‘Prioritized Collation’, since priority is given to the semantic quality of text. It could also be called a ‘Two-Step Alignment’, since the alignment takes place in two phases: the tokens are first aligned on their textual content, then the tokens marked as a mismatch are aligned a second time on structural features (i.e., their type). The experiment is again carried out at the IT department of the Huygens ING under supervision of Dekker and in the form of a DiXiT secondment. Although we are at the moment of writing still in the midst of the experiment, it is nevertheless valuable to reflect upon several intermediate results and, more broadly, their implications.

**Outset**

At the core of the Two-Step Alignment lie the same principles discussed in chapter 2.3 on Native XML Collation. We continue to work with the same sentence – number 0021 – from the story ‘Wat is liefde zonder verleiding’. As said, the TEI/XML transcriptions of the text of Sheherazade primarily follow a textual perspective. At the same time, of course, they are made from a genetic orientation to text, so most of the documentary features are encoded as well. At the outset, we have the following XML-based TEI transcriptions of witness A (folio TS B917/H2bis):

```
<s>Hoe zoet moet nochtans zijn dit
<lbr/>
<del>werven om</del>
```
<add>trachten naar</add> een vrouw , de ongewisheid vóór de liefelijke toestemming!
</s>

and witness B (quarto TS B917/H2bis):

<s>Hoe zoet moet nochtans zijn dit <del>werven om</del> <add>trachten naar</add> een<br/>vrouw ! </s>
<s>Die dagen van nerveuze verwachting vóór de <lb/>liefelijke toestemming.</s>

Witness B consists of two <s> elements; witness A has a similar sentence but only one <s> element. As a human reader – and keeping in mind the tree hierarchy of this XML file – it is not difficult to understand that the extra <s> element in witness B is a parent to the text tokens ‘Die dagen van nerveuze verwachting’. After all, by placing an exclamation mark after the word vrouw, Brulez started a new sentence. So what we expect is something like the following superwitness:

```xml
```

Here, the closing tag </s> and the opening tag <s> are marked as an addition, and the semantically related tokens ‘, de ongewisheid’ and ‘! Die dagen van nerveuze verwachting’ are considered a replacement (i.e., an omission followed by an addition). However, the current configuration of the collation program results in the following superwitness:

```xml
['s', 'Hoe', 'zoet', 'moet', 'nochtans', 'zijn', 'dit', '-lb', '-/lb', 'del', 'werven', 'om', '/del', 'add', 'trachten', 'naar', '/add', 'een', '+lb', '+/lb',
```

---

172 Keep in mind that the actual superwitness exists in the computer’s memory as a graph (more specifically an Edit Graph; see chapter 2.3). What we see here, then, is a linear token sequence derived from that graph. Omitted tokens are preceded by a -; added tokens preceded by a +.
The closing tag </s> and the opening tag <s> are indeed marked as additions (highlighted). But in this output, the semantically related text tokens are not considered as a replacement. In other words, the current superwitness does not match the expectancies of a human reader, who looks first at semantic information and therefore expects the semantically related words ‘, de ongewisheid’ in witness A and ‘Die dagen van nerveuze verwachting’ to be placed in each other’s vicinity, and not separated by the added <s> element.

The logic behind this collation result is quite simple: the algorithm that takes care of the alignment looks for the matches, which in this case are the tokens vrouw and voor. In between the two matches are a number of omissions and additions. These are considered as one long replacement. We cannot blame the alignment algorithm: as it is written, it treats the text tokens equal to element tokens. Furthermore, it does not know that the tokens ‘, de ongewisheid’ in witness A are semantically related to the tokens ‘! Die dagen van nerveuze verwachting’ in witness B. In fact, neither would a non-Dutch speaking reader, but he or she would have to take my word for it.

Our first inclination was that the issue could be solved by collating first all text tokens, then all markup tokens, and finally merging the two resulting superwitnesses into one text/markup superwitness. However, this solution would probably not work as planned. Let’s see if we can work out what would happen if we give the aligner an uninterrupted flow of markup tokens. The input would be as follows:

Witness A: ['s', 'del', '/del', 'add', '/add', '/s']
Witness B: ['s', 'del', '/del', 'add', '/add', '/s', 's', '/s']

The last <s> element in witness B (highlighted) will be considered as the addition, resulting in the following superwitness of markup tokens:

['s', 'del', '/del', 'add', '/add', '/s', 's', '+s', '+/s']

Technically this outcome is correct, but after merging it with the superwitness of the text tokens, it would result in the following text/markup superwitness:
This text/markup superwitness is not desirable because the first closing</s> tag (highlighted in green) is now indicated as a match, whereas this is not the case: in witness A the closing tag is not located directly after the text token vrouw. In the same vein, the last closing </s> tag is indicated as an addition (highlighted in yellow), meaning that it is only present in witness B. This is, however, not correct: the closing tag is present in both witnesses. Briefly put, the solution would not work because it does not take into account that in reality, the markup tokens are not placed so close together in a linear sequence: they are nodes in a tree with text tokens as children. The presence of these text tokens should influence the way tokens are grouped together. Let’s take another look at the current superwitness:

Apart from the misplaced closing </s> and opening <s> discussed above, the – and + prefixes, indicating whether the token is an omission or an addition, are correctly attributed. The problem is, then, that the tokens are not grouped together as we as human readers expect. This suggests the solution may be found at the level of the replacements: after all, a replacement consists of an omission followed by an addition, grouped together based on their relative position. Therefore we decide to carry out a second round of alignment on the tokens that constitute a replacement. Since a replacement means that the tokens do not match, we look at other features of the tokens, such as their type: XML element, punctuation, numeric, etc.

We can reuse parts of the Tokenizer code written for the experiment with Native XML Collation, because we continue to distinguish between a TextToken and an ElementToken. The Aligner code from this experiment...
requires some customization. To recap, the Aligner takes care of a number of operations (see also chapter 2.3). After tokenizing both witnesses,

1. An edit table is constructed and filled with the tokens of each witness;
2. The cells of the table are scored by the Scorer, based on a certain definition of ‘match’;
3. The Aligner calculates the best path through the table based on the total score;
4. Segments are created whenever an edit operation takes place, e.g., from a match to a non-match.

Since we require a second round of alignment, we need to alter our definition of ‘match’ and customize the Scorer. Through the scoring we can indicate what we consider important: the punctuation, the markup, possibly also the XML attributes and their values, etc. Another vital point for this approach is also the segmentation. The segmentation not only groups together aligned tokens, it also groups together omitted and added tokens as a replacement. Where the segmentation works well for sequences of aligned TextTokens, the superwitness indicates that the segmentation does not go as expected for other types. For instance, we may expect the tokens [−,] and [+!] to be considered a replacement and grouped together based on the fact that they are both punctuation located at roughly the same place in the witnesses.

In other words, we are looking at an approach that will ‘refine’ the concept of replacement and the resulting segmentation. By taking the type of the tokens into consideration during the second round of alignment, and adjusting the scoring algorithm accordingly, we may be able to produce the following super-superwitness:

```plaintext
```

Here, both the XML tokens [/s] and [s] and the punctuation tokens [,] and [!] (highlighted) are considered a replacement and therefore grouped together. This means that during the merge of the text superwitness and the markup superwitness, the text tokens are inserted in the expected place.
Scoring and Segmentation

The Native XML Collation program described in chapter 2.3 is written in Python; this experiment is written in Java. I do not go into detail regarding the code and the workings of the program, but it is important to know that both Python and Java are Object Oriented programming languages. This means that, when we model the text and the process of collation, we have to think in separate objects. Each object has certain properties, roles, and responsibilities: for example, a Segment object contains a list of tokens from witness A and a list of tokens from witness B. The objects interact with each other within the program. Within the Prioritized Collation program, we distinguish the following objects:

1. A Token object, subdivided into a TextToken object and an ElementToken object.
2. A Segment object;
3. A Scorer object, subdivided into a ContentScorer object and a TypeScorer object;
4. A Node object.

The Token object speaks more or less for itself: it contains a string of characters from a witness. We distinguish TextTokens and ElementTokens. As described in chapter 2.3, the input file is parsed and tokenized. These Token objects are subsequently aligned. This experiment requires two rounds of alignment: the first round matches on content of the tokens, and – after singling out the segments marked as ‘replaced’ – the second round of alignment looks at the type of the tokens. To this end we create a ContentScorer object, used in the first round, and a TypeScorer object, used in the second round of alignment. With regard to the type of the tokens, we decided for the moment to distinguish three different token types: text, markup, and punctuation. Note that the TypeScorer looks only at the types of the tokens, not at their content. In other words, the markup elements `<del>` and `<p>` will be grouped together in the second alignment based on the fact that they are both markup (and, obviously, only if they are in the same relative position). The same goes for the tokens `[1]` and `[ . ]`, since they are both punctuation.

173 The code of both experiments can be consulted on GitHub: https://github.com/bleekere/xml_collation and https://github.com/bleekere/prioritised_xml_collation_java.
As mentioned above, the Aligner produces a table. The collation program fills the table with the tokens of witness A and tokens of witness B. This means that each cell in the table represents a unique combination of a token of witness A and a token of witness B. Subsequently, the program scores the cells, and then calculates the optimal path through that table. Now, following that path, we can start adding the tokens to the superwitness in the form of Segments. A new Segment object is made whenever a 'state change' occurs between the cells in the Edit Table, from 'match' to 'replacement' for instance. This information is given to the Segment object. Each Segment object then contains two lists of tokens (the tokens of witness A and the tokens for witness B) and knows its type, i.e., whether the Token objects constitute a match, an omission, an addition, or a replacement. The diagram below (figure 65) offers a simplified visualization of this process:

![Diagram](image)

Figure 65. Schematic representation of the workings of the Two-Step Alignment program.

The last step is to build a tree in the memory of the computer that contains these objects. This is where the Node object comes in. To avoid confusion, it may be useful to point out that the tree is not an XML tree and that it only exists in the computer’s memory. It is used for internal purposes only; in the end the tree will be flattened into another data structure that we can use for further processing purposes like visualization. To test the code, we start with a set of simple witnesses that contains only matches and one replacement:
Witness A2: <TEI><s>c</s></TEI>
Witness B2: <TEI><s>a</s></TEI>

When we run the collation program and align the witnesses A2 and B2, the objects described above are created and filled with information about the alignment. First, an empty root node is set. After a first round of alignment, a Node is created for each Segment. If the Segment has the type ‘replaced’, which is the case for the segment in the second Node, the tokens inside the segment are aligned again but this time on type. The result of this second alignment is stored in a new Segment that is appended as a child node to the existing Node. Figure 66 shows a simple version of what the tree would like:

![Figure 66. A schematic representation of the tree containing information about the alignment of two simple witnesses.](image)

The result of the first round of collation is indicated in blue: a root node with three child nodes, which each have a typed Segment – ‘aligned’ or ‘replaced’ – containing Tokens. The tokens of the second node in the tree, with the segment type ‘replaced’, are collated again (indicated in red). Based on their type (they are both text tokens) the tokens are considered a match and aligned. The resulting superwitness (in a linearized sequence format) is as follows:

Root node ['aligned' TEI, s | TEI, s], ['replacement' c | a ['aligned' c | a ]], ['aligned' /s, /TEI | /s, /TEI ]
Thus far, the code works. We see that the superwitness starts with a root node. This root node has three child nodes (marked with [ ] and separated by a comma) that each contains a segment with a type (‘aligned’, ‘replacement’, and ‘aligned’) and the individual tokens of witness A2 and witness B2 respectively (separated by a |). The tokens inside the segment marked as ‘replacement’ are realigned. They are now considered a match too, because they are both of the type TextTokens. A new segment is created with these nodes and appended as a child node to that node.

Running the code on the example sentence number 0021 produces similar, positive results: in the second alignment the program recognizes the type of the punctuation tokens [, ] and [!] and aligns them. However, the resulting superwitness is currently too arduous to read. The example witnesses A2 and B2 above contain only five tokens, and already it takes some practice to read and understand the superwitness resulting from their collation. It is clear that the efficacy of the current visualization leaves something to be desired when collating slightly larger witnesses. This brings us to the final part of this section.

Reflection

We are currently in the midst of the Prioritized Collation experiment. The preliminary results are promising and imply that a two-step alignment yields potential: they suggest that we can use the information encoded in a TEI/XML transcription to examine in great detail textual variation. Furthermore, I can conclude that engaging with a computer in such a detailed way is a valuable method to get to know not only a text in all its intricacies, but also to become aware of one’s own research questions and objectives. The iterative, step-by-step process that is modelling continuously confronted us with questions like ‘what do we want to obtain as a final result?’ and ‘how can we visualize that result in a meaningful way?’ Each time that the code failed – and this happened quite often – we returned to our initial aims and compared them to the failed outcome, in order to see at what point in the alignment process the two did not match.

The concept of Object Oriented programming, too, turned out to be an effective instrument for the modelling process. Conceptualizing the notions of ‘TEI transcription’ and ‘collation’ into individual components (Objects) with certain responsibilities, helped to clarify the workings of digital collation.
Essentially, when genetic editors make a TEI/XML transcription of text, they capture revision as an explicit information component. Within this experiment, then, we took those components and used the information they contain to improve the modelling of a draft manuscript, using their semantic as well as their structural features.

While we continue to work on developing the code base, it is crucial that we think about how we clearly visualize the tree. A classic critical apparatus would not suffice to visualize the results of this Prioritized Collation unless in a highly simplified format. At the same time, it is difficult to come up with a representation that is meaningful for end-users who are familiar with the visualizations discussed in chapter 3.2.2. It is quite possible that the HG visualization of the TAG model discussed in section 4.2 could offer a solution, but this model is still highly abstract and may not be as easily and intuitively understood as existing visualizations. Transforming the tree into a less complicated format entails a loss of information, which compels us to think about what information exactly we could stand to lose. So while the Prioritized Collation makes inventive use of the information about both text and document to improve the automated collation of draft manuscripts, it has also thrown up many questions in need of further investigation.
4.4. Concluding remarks

‘Book historians and editors gaze into the past for their evidence, but they aim at informing or changing thinking in the present and the future.’
(Eggert 2005, 98)

‘The technology and the methodology for representing texts and documents awaits the will and direction of scholars to fulfill the promise of a new but present age.’
(Shillingsburg 2009a, 175)

This chapter set out to answer the question, ‘how can systems of classification promote research into literary genesis?’ In answering that question, the chapter touched upon topics that came up throughout this thesis. From mapping textual and documentary aspects in a TEI/XML transcription to interconnecting the fragments and storing this information: it comes down to an editor’s endeavour to create a digital environment in which she conveys her knowledge of the genesis of a literary work and lays out a selection of tools for others to explore. Either the readers/users decide to follow a genetic path and let themselves be guided by the editor, or they decide to examine the genetic dossier according to their own interests. But what does this ‘free exploring’ actually mean when every genetic relationship is hard coded by the editor? Is there room for new findings, or do readers/users actually follow in the editor’s footsteps unknowingly?

The question is, really, whether that would be a drawback at all. It is highly plausible that Vanhoutte is right and that knowledge sites of less famous authors would indeed not attract the active and involved community editors dream about. In that case, the editor’s presence – whether discretely and indirectly through the provision of certain classifications, or prominently in the commentary and annotations to text – is key to the success of a digital edition. Her traditional role as mediator between text and reader is emphasized and expanded in two ways: by giving prominence to the edition’s commentary and placing it on a similar level with the text, and by involving the editor in the creative and challenging process that is text modelling.

174 So as to avoid readers/users from feeling lost, Dillen suggests that the interface is set up in such a way that editorial guidance remains at calling distance, ready to provide guidance when needed (cf. Dillen 2015, 227). Examples of this subtle editorial presence are an omnipresent search box, or that certain analysis tools only ‘pop up’ when they are relevant for the view the user/reader has selected.
The main obstacle here is not so much setting up a database or a digital editing environment. Instead, the challenge lies in the text encoding and the ways in which information about that text is expressed. We have seen – and know – that the technical realization of capturing information about a document and the text it contains is highly complex; it will continue to engage the digital editing community for some time to come. With each new solution, new ways of conceiving of text are introduced. It may take some external influence from disciplines like computer science – in the human form of curious IT specialists – to introduce the concept of hypergraphs, but it takes editors and textual scholars to envision its implications and its practical use for literary and textual genetic studies. This is clearly the case for the Prioritised Collation experiment, its consequences for genetic research and the best visualization of the outcome. The same applies to the visualization of the TAG model when it is applied to a literary work. While I assert that the TAG model offers a valuable data model and a powerful research instrument, it will take more experimentation and, especially, practical implementation before it can develop its full potential for textual genetic studies.

Working closely with computer scientists, software developers, and IT specialists at the Huygens ING has shaped my thinking about text in trees, graphs, ranges and layers. Therefore, when Van Hulle asserts that digital editing can serve as a tool for manuscript research and genetic criticism, I can add that writing code for an automated collation tool that deals directly with the composite reality of a draft manuscript can also promote manuscript research. At the same time, I am aware that this fortunate position is not reserved for all (young) scholars in our field. Nevertheless, the fact that many textual scholars theorized early on about digital scholarly editions of which the infrastructure is similar to the one currently proposed by the Alexandria project demonstrates that it takes first and foremost the ability to think outside existing models, and to conceptualize about text, the shapes it takes, and the circumstances under which it can best be studied.
Conclusion

The double-vision that characterizes textual scholarship: to see at once both the signifying surface and what lies beneath’
(Galey 2010, 106)

‘There are certain immediate dangers to improvisation, and one of them is that everybody coalesces immediately ... The interesting place is not chaos, and it’s not total coherence. It’s somewhere on the cusp of those two’
(Eno, qtd. in Wells 1995)

One of the artists who worked with Brian Eno was the late David Bowie. In an interview with Time Out (1995), Bowie noted that ‘more and more people are moving into areas they’re not trained for’ – he mentions a movie director that had recently made a music record – which is in his opinion a highly positive development. Evidently, very few among us can claim the chameleon-like nature of Bowie, for whom the boundaries of professions were as fluid as those of gender. But I do not believe Bowie advocates embarking on a project entirely untrained. Rather, he hints at a certain openness of mind, a willingness to look beyond the institutionalized borders of a domain, a boldness to work with or create new methods. Throughout this study I have tried to demonstrate that a similar mindset can be highly beneficial for digital editing. I focused on bringing together the methods of scholarly editing for the genetic orientation and the particular methodology of computer science called ‘modelling’. With that aim, this study extended previous work by McCarty, Pierazzo, Ciula, and Galey among others, who have emphasized the value of digital modelling for humanities research. I further expanded on the line of argumentation introduced by Van Hulle and Dillen, who have demonstrated a rapprochement between scholarly editing and genetic criticism, and I argued in favour of a rapprochement between computational methods and textual genetic research.

Building upon Van Hulle’s concept of ‘textual awareness’, this study showed how digital technology can be employed to examine that ‘enormous amount of potential energy’ of text (Van Hulle 2004a, 158). I distinguished four components of genetic editions that appear to have particular potential for modelling, and subsequently evaluated this potential from both a theoretical and a practical perspective. As such, I made clear how modelling can assist the scholarly editor in mapping written invention, and how this enhances the textual awareness of both editors and readers/users. The main premise of the
study was to understand the digital genetic edition as a research instrument, or more precisely as a combination of several research instruments or interacting modular components. The components under discussion were ‘transcription’, ‘collation’, ‘visualization’, and ‘classification’. In reality there is evidently overlap between them, but I chose to maintain the dividing lines not only for reasons of clarity, but also because the process of modelling thrives by formalization and by working with small, definable elements. The components informed four research questions that, in turn, each constituted a chapter of this study.

Accordingly, Chapter 1 looked into the methods of editing for the genetic orientation (defined by Shillingsburg and Van Hulle as a combination of genetic criticism and genetic editing) and the use of digital models in textual scholarship. At first sight, the requirements of modelling appear to contradict the intentions of the genetic editor: for tools to be interoperable we need to formalize and standardize our research objects, whereas textual genetic research pays attention to the smallest intricate detail of individual texts. Taking the perspective of digital philology, which revolves around experimental textual research and is as much interested in the process as in the end product, allowed me to synthesize modelling and genetic editing. The chapter thus highlights the importance of understanding every step in the process of editing. Such understanding goes beyond knowing how to model a text in TEI/XML: editors equally benefit from having a detailed understanding of how different tools process TEI/XML files, or from being able to argue why they use TEI/XML in the first place. This awareness contributes to a productive self-reflection within the field of textual scholarship, but at the same time it produces a number of original obstacles: the (lack of a) consensus on standards, the evaluation of output, and the technical schooling of humanities scholars. Although the case for research models is being made for over a decade now, institutions and funding bodies continue to have difficulty with the evaluation and financing of projects that have no clear end products. I concluded that the value of this type of research can best be measured against its methodological and epistemological implications.

Chapter 2 expanded upon these three points – the discrepancy between general standards and idiosyncratic detail, an awareness of the process, and the complexity of assessing the output – and considered them in the light of transcription and automated collation. I focused on texts that are marked up according to the genetic orientation, because the encoding of this type of text creates original challenges that have to do with the flexible definition of
‘version’ and the combination of a document-oriented and a text-oriented perspective. The chapter thus addressed several crucial topics within digital editing and text modelling, namely the separation of text and document, and the extent to which a multidimensional draft manuscript can be represented in an XML tree structure. Assuming that transcriptions contain the text of the document as well as the editor’s analysis of these multiple dimensions, I examined how this editorial knowledge could be used to improve the automated collation of TEI/XML transcriptions. Together with Dekker, I developed a software program that collates entire XML trees (‘Native XML Collation’). As point of departure, we used examples from the genetic dossier of Brulez’ Sheherazade. Although the program’s code is in essence correct, we found that the dominance of the document structure did not result in the desired output, since collation is primarily a text-oriented activity. Despite its apparent ‘failure’, the experiment thus presents an exemplary case of the trial-and-error character of modelling. I returned to the experiment in the last part of Chapter 4. With a second program called ‘Prioritized Collation’, we gave preference to the text and adjusted the alignment only in the case of a non-match. For both experiments, I gave a step-by-step account of our proceedings and the underlying considerations and made clear what happens ‘under the hood’ of a software program. I thus intended to contribute to a better understanding of the operations that software programs perform on the text and, by extension, to promote a critical way of thinking about code and computing.

Working on the automated collation programs required a constant thinking about ways in which the outcome could be presented. Chapter 3 therefore examined how visualizations can be taken a step further to effectively enhance our understanding of textual genesis. It was found that this could be primarily established if the visual was placed on the same level as the textual; thus creating two complementary perspectives. Collectively, they would provide a means to present both the rhizomatic and the linear character of the ‘avant-texte’. Other features that lead to similar results are a revision narrative as proposed by Bryant, and interactive visualizations like the genetic plan of the Faust edition or the animated macrogenetic overview developed by Dillen et al. The findings of this chapter were combined in a digital museum application of Brulez’ Sheherazade, which presents an interactive genetic narrative aimed at a wider audience. The chapter concluded that thinking about visual ways to represent the results of a study into a text’s genesis, digital visualization constitutes a cognitive exercise in information modelling. As such, it provides
not only a means of communication and outreach, but also a research instrument for the scholarly editor.

The fourth and last chapter builds on the idea of modelling the linear, textual features as well as the rhizomatic, interconnected nature of the ‘avant-texte’. It addressed what I consider one of the most valuable affordances of the digital medium, namely the ability to represent the coexistence of text and documents, the interaction between textual versions and their material reality. The chapter’s analysis of the prevailing theories and practices in text modelling confirmed that it is paramount to keep an open mind towards new technologies without discarding the existing ones. For instance, XML is an effective and widely used data structure that is quite sufficient for simple forms of text modelling. Only when its tree hierarchy becomes a limitation can we look for ways to complement it. Therefore, continued efforts are needed to make knowledge about data structures, data models, and their function for textual research and scholarly editing more accessible. In discussing a number of technologies to capture and store information about a text and its genesis, I kept coming back to the crucial role of the scholarly editor. Whether it pertains to theoretical modelling on a conceptual level, actively engaging with the underlying code, or to assessing the value of technological resources for textual genetic research, the editor’s contribution is pivotal. Her mediating function is broadened as she introduces literary scholars as well as computer scientists to the nature of the draft manuscript, and as she demonstrates to them the traces of writing that are so compelling and yet so problematic to capture.

As I focused on the editor’s role in the making of genetic editions, the scope of this study was necessarily limited. In my attempt to avoid generalizations, I treated a selected number of editorial theories and practices in great detail, thus inevitably leaving out other laudable projects. Furthermore, in my discussion of the editor as mediator between text and reader, I concentrated on one side of the equation (‘text’) and only briefly touched upon the experience of readers/users of editions. Considering the importance of the readers/users in the durability and overall success of digital editions, it is a good thing others have taken this topic to heart. The same applies to the publishers whose role in creating sustainable environments for digital editions cannot be
underestimated. It also goes without saying that, although the ongoing work on automated collation programs should be considered in the light of ‘process over product’ (cf. digital philology), the experiments described in chapter 2.3 and chapter 4.3 have raised a number of questions that need to be investigated. Further research is required to determine the most successful output visualization of the Prioritized Collation program. Additionally, I see a significant need to test the TAG technology and to find out how it can be used to model the editor’s knowledge of the writing process.

A final issue that has surfaced throughout this study is the training of (future) textual scholars. First and foremost, I support the emphasis on collaboration and knowledge exchange. One of the most valuable humanities skills is the ability to ‘zoom out’ and reflect, and this skill is widely applicable. Nevertheless, this study has shown that a more applied technical schooling can expand our ways of seeing, studying, and understanding a literary text and its genesis. This does not imply that scholarly editors all need to become programmers, but a broader understanding of certain technologies would make the affordances of the digital medium more accessible.

The findings of this study can help to define what such a technical knowledge might entail. Digital technologies can be used to gain a fuller understanding of the object to be modelled – in this case, text. The application of various digital technologies, from XML and XSLT to OO-programming languages like Python and Java, deepened my examination of Sheherazade’s draft manuscripts. It resulted in an understanding on several levels: from manuscript analysis and TEI/XML text encoding to developing algorithms that can process these text encodings, and devising visualizations that allow us to narrate Brulez’ writing process. At times, these activities took place outside my comfort zone, but digital editing is a job one never does alone. A close collaboration with experts from different domains created an environment that Eno would approve of: productive and experimental, on the cusp of chaos and coherence. So, in addition to studies that argue for the prominence of the editor (Gabler 2010) and that advocate digital documentary editions (Pierazzo 2014), we can say that textual genetic research is further boosted by the use of computational methods like modelling. The interaction with specialists from other fields and

---

175 The work of Pierazzo in that area shows that some valuable alliances can be formed between editors and publishing houses. See for instance the NeDiMAH workshop ‘Toward a new social contract between publishers and editors’ that took place on January 26, 2015 in Grenoble.
the confrontation with unfamiliar technologies generate new ways of perceiving text and documents, and deepen our insight into the fluid nature of text.

If textual scholars see both 'the signifying surface and what lies beneath', this study investigated what tools and instruments can improve their vision. It did so by alternating between a bird's eye perspective on various editorial theories and a microscopic study of specific lines of code. The findings indicate that the days of textual genetic research are far from over. We can in fact observe a new golden age, not of creating modern manuscripts but rather of the study thereof. We have seen that, by their very nature, draft manuscripts offer a stimulating challenge for digital technologies. But the documents and texts cannot speak for themselves; they need somebody to tell their story. This study has shown that editors are becoming central figures in portraying manuscript research as an exhilarating venture for scholars, scientists, and readers from all backgrounds.
Acknowledgements

Although the word suggests otherwise, this monograph is not a one-woman’s work. In the following paragraphs, I’d like to name some of the exogenetic forces that have shaped and influenced my research. The first moments of inspiration can be traced back to Adriaan van der Weel, who incessantly urged me to go ahead and do a PhD already. I’m ever grateful to Peter de Bruijn who introduced me to the crucial phrase ‘with a probability bordering on certainty’. He revealed the marvels of deciphering the inscriptions on a document page and showed me the value of scholarly editing. A sincere word of thanks goes to Ronald Haentjens Dekker, who took me under his wings and patiently explained to me the magic behind algorithms and computers, thus making them all the more interesting. I would also like to thank Elena Pierazzo, Hans Walter Gabler, David Birnbaum, and Peter Boot, whose willingness to share their wisdom (and the occasional pint of beer) has been both stimulating and encouraging. Last but surely not least, my supervisor Dirk Van Hulle, who had an unwavering belief in this research and firmly kept me on track. His passion for textual genetic studies, his open mind, and his tireless enthusiasm for research are truly inspiring. Together with Vincent Neyt, Dirk has made the CMG the productive and ever-expanding research hub it is today.

Of course, the CMG could not exist without its team members. Wout Dillen did make me feel welcome right from the start – it was happy day when he joined DiXiT and we became double colleagues. Many thanks go to Olga Beloborodova, whose unconventional remarks constitute a book of their own, to Vincent Neyt, technical genius and master of comic dryness, for all his advice and comments; and to Ronan Crowley, a terrific flatmate with a knack for bad movies and thesis revisions. My Brulez-buddy Sarah Fierens, Tom de Keyser, Arent van Korlaar, Alison Luyten, Pim Verheyen, Pim Verhulst, Ellen Vanderstraeten – you’ve all taught me so much. The same goes for Vanessa Joosen, Mike Kestemont, and Geert Lernout: it has been a true privilege to have you as colleagues. Thanks also to my fellow-doctoral students at the University of Antwerp, especially to Edith Cassiers, feminist in padded gloves and Leslie Knope incarnated, and to the ever-cheerful and ‘atypical Flemish’ Johanna Ferket. Furthermore, I would like to thank the AMVC Letterenhuis for letting me spend many hours in the sunlit calm of their reading room, and for their readiness to collaborate with us and Prophets on the BDE.
And then there is DiXiT. What can I say? It was the best of times, period. The moment I read the call for applications in 2013, I was sold. Our first official meeting in Cambridge felt like coming home to old friends. With everything still before us, we set out to change the world. I’m grateful to the team of supervisors for letting the fellows indulge in this belief, while quietly steering us in more realistic directions. Working with these guardians of the digital editing galaxy was as big an honour as it was a pleasure. My fellow DiXiTeers - Richard Hadden, Misha Broughton, Daniel Powell, Gioele Barabucci, Linda Spinazzè, Magdalena Turska, Federico Caria, and Roman Bleier - it was an honor to serve with you (and I still think we can change the world). Extra purple mentions go to Frederike Neuber, smartest and most determined German Kurbis; Aodhán Kelly, my partner in PhD crime; Merisa Martinez and her sharp sense of humour; Elena Spadini and Anna Maria Sichani, fierce Southern Europeans at the Huygens ING; and Fran and Tuomo, the most unlikely and yet endearing duo imaginable. You guys...

I am greatly indebted to my parents, grandmother, and brother. Thank you for never doubting me; your support means more than I could ever put in words. Much gratitude also goes to my friends for their puzzled yet sincere interest in my research, for their patience, and especially for dragging me out of my voluntary isolation and into a social setting every now and then. You kept the darkness at bay with friendship and Cabernet. A special word of thanks to Rosalinde Stadt and Rebecca Bego, two of the most intelligent and hard-working women I know. And finally, I’d like to thank Stephane Kaas for showing me that curiosity is the key to life. I could not have done this without his assuring presence and superhuman patience throughout the years. He should actually be on the cover of this book.
Abstract / Samenvatting


Vier componenten van een digitale editie kunnen worden ingezet als digitaal onderzoeksinstrument voor tekstgenetische studie: transcriptie, automatische collatie, digital visualisatie, en de classificatie en organisatie van tekstversies en documenten. De kennis van een editeur, waaronder informatie over de tekstgenese en de verschillende schrijflagen op een document, kan – tot op zekere hoogte – worden opgenomen in de transcriptie. Collatie software kan gebruik maken van deze gecodeerde informatie om tot een beter resultaat te komen. Digitale visualisatie biedt de mogelijkheid om de tekst gelijktijdig vanuit meerdere invalshoeken te bestuderen, en geeft editeurs de gelegenheid om op visuele wijze hun analyse van de tekstgenese presenteren. Het systematisch ordenen van tekstversies en documenten, tenslotte, ondersteunt een studie naar zowel de lineaire kenmerken van tekst als de netwerkstructuur van de ‘avant-texte’.

Het uitvoeren van deze handelingen met behulp van de computer biedt editeurs een waardevol nieuw perspectief op tekst. Inzicht in de werking van de gebruikte technologie is hierbij onmisbaar, en resulteert in een diepere bewustwording van de tekst en de tekstgenese. De dissertatie concludeert dat de inzet van computationele methodes als modelleren een gunstig effect heeft op digitaal editeren en tekstgenetisch onderzoek. Hieruit volgt dat de editiewetenschap en computationeel onderzoek nader tot elkaar komen.
Bibliography

All websites cited have been accessed on May 25, 2017 in a Chrome browser.


BECKETT DIGITAL MANUSCRIPT PROJECT. See: Van Hulle, Dirk et al. 2013-


BRULEZ, RAYMOND. 1929-1930. ‘Wat is liefde zonder verleiding’ [notes]. AMVC Letterenhuis B917/H2a.

BRULEZ, RAYMOND. 1930. ‘Wat is liefde zonder verleiding’ [manuscript]. AMVC Letterenhuis B917/H2a.
BRULEZ, RAYMOND. 1930-1932. ‘Wat is liefde zonder verleiding’ [typescript folio]. AMVC Letterenhuis B917/H2bis.

BRULEZ, RAYMOND. 1929-1930. ‘Wat is liefde zonder verleiding’ [typescript quarto]. AMVC Letterenhuis B917/H2bis.


BRYANT, JOHN. 2006. ‘Editing a Fluid Text.’ http://rotunda.upress.virginia.edu/melville/intro-editing.xqy


CUMMINGS, JAMES. 2015. ‘Childish Toys.’ In my <element/> May 14. https://blogs.it.ox.ac.uk/jamesc/2015/05/14/childish-toys/.


EGGERT, PAUL. 2005. ‘These Post-Philological days.’ Ecdotica 2: 80-98.


GREG, WALTER WILSON. 1913. ‘What is bibliography?’ The Library 1: 39-54.


HAENTJENS DEKKER, RONALD AND GREGOR MIDDELL. CollateX. https://collatex.net/.


http://gde.upress.virginia.edu/00Cgde.html.


LIU, YIN, AND JEFF SMITH. 2008. ‘A Relational Database Model for Text Encoding.’ Digital Studies/Le champ numérique 0.12.


and Raffaele Viglianti.’ RIDE: A review journal for digital editions and resources 2. DOI: 10.18716/ride.a.2.5.


O’DONNELL, DANIEL PAUL. 2010. 'Different Strokes, Same Folk: Designing the Multi-Form Digital Edition.' Literature Compass 7.2: 110-119.


http://digitalmedievalist.org/journal/7/rosselliDelTurco/.


SHILLINGSBURG, PETER. 2009a. ‘How literary works exist: Implied, represented, and interpreted’ Text and genre in reconstruction: Effects of digitalization on ideas, behaviours, products and institutions: 165-82.


STREUVELS, STIJN. 1927. De teleurgang van de Waterhoek.


VAN DER WEEL, ADRIAAN. 2010. ‘New Mediums: New Perspectives on Knowledge Production.’ Text Comparison and Digital Creativity. The Production


VAN PEURSEN, WIDO. Ernst Thoutenhoofd, and Adriaan van der Weel, editors. 2010. Text Comparison and Digital Creativity. The Production of Presence and Meaning in Digital Text Scholarship. Leiden: Brill.


