Digitally Supported Reading

The Standardisation of Markup Languages and its Impact on Reading

Marcel Knöchelmann

University College London
Department of Information Studies
Gower Street
London, WC1E 6BT

Submitted at University College London in December 2015
Published on LePublikateur in May 2016
This essay and information about the author can be found on www.lepublikateur.de

lepublikateur

Table of Content

List of Abbreviations

ASCII	→	American Standard Code for Information Interchange
CSS	→	Cascading Style Sheets
DTD	→	Document Type Definition
EPUB	→	Electronic Publication (file format)
GML	→	Generalised Markup Language
HTML	→	Hypertext Markup Language
ISO	→	International Organization for Standardization
PDF	→	Portable Document Format
SGML	→	Standardised General Markup Language
WWW	→	World Wide Web
XHTML5	→	XML-serialised HTML5
XML	→	Extensible Markup Language
XSLT	→	Extensible Stylesheet Language Transformations

1 Introduction

A text is an "ordered hierarchy of content objects" (DeRose et al., 1990, p. 6) that reveals most information by combination of its discrete entities as well as by intertextual referencing. With a long history of reading, humans developed skills to master these tasks. The moment of the standardisation of markup languages though, marks an important moment in the history of reading. Markup languages changed the way machines support human reading, which in return changed how humans read. This essay will explore the significance of markup languages in the history of reading by specifying how they changed reading.

As this essay focusses on markup languages, device-specific displaying of texts (hardware) and text encoding will not be described. Though ASCII or Unicode encoding build the foundation for machines to display and manipulate characters, encoding does not help machines to interpret the meaning of content in the way markup languages do. Furthermore, within markup languages, this essay focusses on those developments which affect reading in general rather than specific markup languages which act in fields like financial analysis or mathematics, as shown by Russell (2005).

2 A brief description of markup languages

2.1 The standardisation of markup languages

Early digital representation of texts only displayed characters statically in a similar way printed representations did. These characters had no meaning to the machine that displayed them. As software and hardware advanced, the need grew to let the machine interpret text rather than just to display it. The basic ideas of such processing of text reach back to the early 1950s. At this stage, however, only mechanical indexes were produced and simple word counts computed to foster scholarship. With the rise of more advanced technologies in the 1960s, first achievements in digital textual analysis

were made by using a simple markup that specified structural models for a set of texts. The advantages of such encoded structure representation triggered a diversity of software developments, especially in the fields of archiving and preserving of texts. (Barnard et al., 1996; Hockey, 2004)

Parallel to these developments, IT-specialists formulated generic coding which made use of descriptive tags (e.g. "heading" for headlines in all documents) to generalise representation models in the late 1960s. In contrast, other digital representation models made use of specific coding or macros which caused the problem that every data representation was unique. Charles Goldfarb introduced the Generalised Markup Language (GML) at IBM in 1969, a descriptive encoding language "to solve the data representation problem" (Goldfarb, 1999, p. 76).

The combination of these different developments finally led into the standardisation of GML by ISO: the Standardised General Markup Language (SGML) in 1986. This early standard model comprised the idea of tagging original content with specific declarations, while the syntax of such tagged document is defined in the Document Type Definition (DTD). This was the foundation of the future of digital text processing: separating content from appearance, format, and design. (Goldfarb, 1999; Hockey, 2004; World Wide Web Consortium, 2006)

2.2 The technology of markup languages

The core advantage of the technology of markup languages is that content (mostly text) is stored as media-independent (sometimes called media-neutral), information entities. "One fundamental premise of SGML is that texts are composed of discrete content objects, and that supplying meaningful names for these delimited textual objects, their attributes and their hierarchical relationships independent of possible appearances is one of the most powerful means of transforming text into information units…" (Cover et al., 1991, p. 198). This means that the actual content will not be manipulated for

specific purposes once it is tagged[1]. Whereas the manipulation is defined in programming files (JavaScript), or stylesheets focussed on design (CSS) or on content processing (XSLT), the markup of texts focusses only on describing structure or referencing relations (Birnbaum, 2015).

As can be seen, markup makes no comments about how the content appears. It tries to reflect the structure of expression of content, specifies interpretative entities, and "…becomes an instrument for use in transforming the implicit variation of the interpretation of an identical expression into the explicit fluidity of the expression of an identical content" (Buzetti, 2002, p. 83).

3 The impact of markup languages on texts

To identify the significance of markup languages in the history of reading, it is important to explore their impact on the foundation of all reading: textual content.

Texts are linearly structured and highly hierarchical (DeRose et al., 1990). In contrast, understanding texts happens in a non-linear process. By connecting the interpretations of words, sentences, and paragraphs, the process of reading turns texts into meaning. This thereafter is referenced with surrounding elements (contextual reading). In addition to this mass of linked meaning, every reader even builds her own set of references, taking into account the individual knowledge she contributes to the reading process. This shows that reading is a massively complex cognitive process. Extracting meaning requires recognition of both intrinsic and extrinsic semantic connections as well as associations. (Warwick, 2004; Smith, 2004)

Markup languages try to model this structure of understanding. The tags which are added to the original content represent the specific interpretation of it. As Buzzetti

[1] Tags are the core elements of markup languages; they build the markup surrounding the actual content (bold): `<tagname attribute="x", ID="x">content element</tagname>`. A tagged content element thereby forms a discrete entity which can be addressed directly, or be further specified by attributes or IDs. To illustrate how markup lanugages work, an example of tagged content is attached in the appendix.

writes, "... if the text (or rather its realization through a concrete form of expression) may be conceived as the linearization of a complex content, then the markup may be considered the instrument most fit to expose their potential structural relations" (2002, p. 80). This exposure enables machines to process texts interpretatively, which is furthermore available independent of a particular machine or purpose. Cover et al. write about the processing ability before the standardisation of markup languages: "The presence or absence of explicit textual markup defining logical structure and formal content-object relationships within the data is irrelevant so long as the local computing system can manipulate the file" (1991, p. 197). The standardisation of markup languages dissolved this critical factor. With SGML, a content's structure is rendered in discrete entities so that information can be classified on a granular level (only delimited by encodable characters). As software applications were more and more programmed to handle and parse SGML, information of similar interpretative entities could be digitally compared, combined, or otherwise manipulated by machines for the first time. (Cover et al., 1991; DeRose et al. 1990)

Conclusively, the linearity of textual content was partly resolved. Machines began to handle texts with respect to the interpretation of content. This led to a diversity of developments in digital content representation and processing, e.g. the hyperlinking of internet documents (which marks the foundation of the WWW), the creation of universally interchangeable document formats like the PDF, or the establishments of reflowable, enhanceable reading formats like EPUB. All of those technologies store and retrieve content in particular forms of XML, an extendable substandard of SGML (Qiu et al., 2010): EPUB uses XHTML5 (Garrish and Gylling, 2013), and WWW content is often represented by a combination of HTML (or newer standards), XML, and CSS (Lie and Saarela, 1999).

Concerning the universal usability of tagged content it must be noted that XML is a language that can have an infinitely large number of tags, each of which can have a variation of the standard description (Bosak, 1998). Other markup languages only have a restricted number of tags, e.g. HTML. This is derived from the fact that every web

browser uses HTML and must inherently know how to deal with tagged descriptions, whereas XML files are always accompanied by explaining files (DTD, XSLT).

4 The impact of markup languages on reading

As chapter 3 shows, the impact of markup languages on reading must stem from how machines process texts, i.e. enabling machines to interpret content, and thus to partly be active in the reading process. In the following, three core aspects of this impact on reading will be explored: media-independency, hypertexts, and improved active reading.

Firstly, media-independency changes the direct reading environment. Though the concept of media-independency is a core feature of markup languages (Cover et al., 1991), its impact on reading might not be visible immediately. At first sight, media-independent text still appears as text when a reader accesses it. The impact can rather be seen in how and in which context texts are accessed.

Historically, all texts accessed by readers were highly similar in appearance. There may have been different editions of a text; these though didn't vary much as they were all based on the same product, a printed book. Thus, readers can be separated into groups, each comprising readers of one edition of a book. Within these groups, all accessed a text of the same appearance, whereas between the groups few differences occurred; e.g. typefaces or bindings may varied from one edition to another.

Nowadays, such a simple overview of textual variation is impossible. On the contrary, a nearly infinite number of appearances is possible. Many publishers offer their contents in different formats using XML (Smith, 2012). Thus, readers can access texts not only in both printed and digital form, they can also change the appearance of the digital text. With eReaders and especially with browsers, both format and layout of texts are seldomly fixed. Readers are thereby able to fit the text to their reading habits, making the reading experience as convenient as possible. It no longer has to be as the publisher wants it to be. In addition, the technology enables impaired readers (e.g. partially

sighted or dyslexic) to access textual contents despite their impairment, as Bauwens et al. examine (1996).

Furthermore, media-independent content leads to contextual individuality. For instance, while reading newspapers in print, readers relate the meaning and information of an article to the appearance of the whole page (with other articles, announcements, or advertisements surrounding it). In digital editions and especially on the web, articles mostly appear as single pages whereas advertisements vary depending on the cookies of a reader's browser. Thus, the context of an article is completely different, leading to a more individual reading experience (Chartier, 2004).

Secondly, hypertexts make reading unsteady and segmented. Another aspect of text as discrete information entities that are specified by markup is that each of these entities can be addressed directly. Text can function as anchor (a discrete textual entity which is tagged as the sender) and as target (a discrete textual entity which is tagged as the address to which the anchor is referring). In general, this is called linking whereas some links have specific names; e.g. a link between different files on the WWW is called hyperlink (Landow, 2006).

The idea behind linked content points back to 1945, even before the connection of computers to the internet or the standardisation of markup languages (Bush, 1945). Since 1945, the idea gradually developed to becoming a powerful tool that not only changed how people access content, but also how it is consumed (Landow, 2006).

Today, text links create a map of content. Most digital texts are either linked within the text itself (an index in a PDF is a list of anchor texts referring to particular textual entities) or to connecting files (every web site that is more than one static page contains links, hidden in menus, buttons, or behind text). This makes the process of reading very unsteady (Landow, 2006), or as Chartier points out: "Reading in front of the computer screen is generally a discontinuous reading process that seeks, using keywords or thematic headings, the fragment that the reader wishes to find…" (2004, p. 142).

Furthermore, textual entities in printed products always have to be searched manually. In a printed book, the reader has to flick through pages until the right chapter is found. In digital documents, the right position in a document is found much faster by clicking on a link. This behaviour, especially considering reading of non-literary texts, alters the reading itself and leads to segmented reading. Readers click through documents, skim through paragraphs and over headlines until the right information appears. They can even click from one site to another without even knowing whose (publisher's or author's) content they are currently accessing. (Brown, 2001; Liu, 2005)

The scholarly reader is even more affected by links. Scholars rather search for broad keywords, succinctly narrowing the search until the appropriate content is retrieved; an example is an index like Google Scholar which is regularly used by 60% of researchers (van Noorden, 2014). Furthermore, as publishers provide their articles digitally on platforms, these articles construct a semantic network of information. When the platform's contents are stored via XML, a single article becomes a data set itself, rather than being a linear string of words. Researchers don't have to read whole articles anymore, as links carry them directly to the searched information. The linked list of references at the end of articles enhances this data set, as well as technologies like Altmetrics which show dynamically where an article is referenced (Altmetrics, 2015). This enables readers to follow the way the insights of an article went (Landow, 2006). However, as precise as such a concept may seem, it can also narrow the view as the technology only provides the specifically searched items. Semantic based search technologies like Le et al. (2013) have developed (and for which again markup languages build the foundation) can help to re-create serendipity in research.

It must here be noted that simple text searches are not based on markup languages. For instance, searching in a WORD file for a specific word only commands the software to search for the specific series of characters. Semantic searches, however, are processed based on markup languages (Goldschmidt and Krishnamoorthy, 2008; Le et al., 2013).

Thirdly, with the use of markup languages, digital environments can offer improved active reading options. Active reading involves annotating, highlighting, comparing, or note-taking while reading (Hood and Sahari Ashaari, 2013). Especially scholars, but also readers of news articles on the WWW or readers of eBooks can add annotations to the content. These build a new layer above the text, are markup themselves, and, as the markup format is standardised, such annotations can be shared. Other services of publishers enable scholars to compare different editions of texts directly and read the specific annotation of a particular entity of a text (see for example the Oxford Scholarly Edition Online, 2015). As Martha Smith explains, these options of active reading offer scholars to work more efficiently, but also demand new skills (2004).

Furthermore, tools like Google Search, Bing Translation, or Wikipedia (all products of the efficient use of text via markup) are in use every day and are even embedded in digital reading products like the Kindle eReader (Amazon.co.uk, 2015). Readers thus don't need to search for a translation in a dictionary. They can stay within the text and send the query directly to the tool with a fingertip. This makes reading more vital and efficient (Hillesund, 2010).

In addition to these direct impacts on reading, the use of markup languages may lead to a better understanding of the semantics of textual contents. A study carried out by Grue et al. suggests that ordinary readers generally do not know the underlying structure or semantics of a text. High school students, the participants of the study, had to read a literary text and tag it with simple markup describing the narrative. They "envisaged a schematic form to represent this reading experience, but found disjunctions between these experiences and their textual realizations" (Grue et al., 2013, p. 243); e.g. the students thought about finding a particular structure in the text, but, by tagging it, realised a different structure. This suggests that digital schematic enhancement could foster textual understanding in reading processes when the schema is displayed in addition to the text.

5 Conclusion

This essay aimed to explore how markup languages changed reading. The overview of the technology showed the core aspects of markup languages, followed by their impact on texts. Thereafter, the changes in how texts are processed in digital environments show clearly that the impact of markup languages on the history of reading is immense. This can be seen in day-to-day readings as well as in particular kinds of reading, like scholarly reading.

As Culkin concisely summed up an idea of Marshall McLuhan: "We shape our tools and thereafter they shape us" (1968, p. 460). In other words, we built the new technology which now shapes our behaviour. There were not new needs emerging among readers or society that prompted IT-specialists to adapt, it was the other way around. This important aspect must be borne in mind, as the use of markup languages will have further impact not only on our behaviour, but also on how we feed the new technology with changing content. This will in return have further impact on how we read. Textual content in the future may be 3-dimensional modelling of information, semantic mapping, or granular, reacting news articles. (Lloyd, 2015; Warwick, 2004)

In addition, with a diversity of non-hierarchical approaches to processing text, digital representation of text is today even nearer to becoming an autopoietic system and thus to fully representing the diversity of natural language. "The great gain that comes with such a tool is the ability to specify – to measure, display, and eventually to compute and transform – an autopoietic structure at what would be, in effect, quantum levels" (McGann, 2004, p. 206). This can be seen in artificial intelligence approaches in which machines learn to transform information into meaning - a principal that was beforehand reserved to the human brain. As humans are still used to extracting information in a linear way, the appearance of all of these new models causes forms of defamiliarisation which in return may reveal new insights into the content (Warwick, 2004).

The New York Times, traditionally a provider of texts which are read by many people, even goes as far as to state: “The tags that we attach to articles enable nearly everything that happens to that article after publication” (nytlabs, 2015). This shows that those who want to be read digitally shouldn’t rely on the content alone, but enhance the content with markup.

Reference list

Altmetric LLP (ed) (2015) *The Altmetric Explorer* [Computer program]. Available at http://www.altmetric.com/aboutexplorer.php (Accessed 6 December 2015).

Amazon.co.uk (ed) (2015) *All-New Kindle Paperwhite* [Online]. Available at http://www.amazon.co.uk/All-New-Kindle-Paperwhite-Resolution-Display/dp/B00QJDO0QC/ref=sr_1_2?s=digital-text&ie=UTF8&qid=1449923759&sr=1-2&keywords=kindle+ereader (Accessed 12 December 2015).

Barnard, D. T., Burnard, L. and Sperberg-McQueen, C. M. (1996) 'Lessons learned from using SGML in the Text Encoding Initiative', *Computer Standards & Interfaces*, vol. 18, no. 1, pp. 3–10.

Bauwens, B., Evenepoel, F. and Engelen, J. (1996) 'SGML as an enabling technology for access to digital information by print disabled readers', *Computer Standards & Interfaces*, vol. 18, no. 1, pp. 55–69.

Birnbaum, D. (2015) *What is XML and why should humanists care? An even gentler introduction to XML* [Online], obdurodon.org. Available at http://dh.obdurodon.org/what-is-xml.xhtml (Accessed 2 December 2015).

Bosak, J. (1998) 'Media-independent publishing: Four myths about XML', *Computer*, vol. 31, no. 10, pp. 120–122.

Brown, G. J. (2001) 'Beyond print: Reading digitally', *Library Hi Tech*, vol. 19, no. 4, pp. 390–399.

Bush, V. (1945) *As We May Think* [Online], The Atlantic. Available at http://www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/ (Accessed 29 November 2015).

Buzzetti, D. (2002) 'Digital Representation and the Text Model', *New Literary History*, vol. 33, no. 1, pp. 61–88.

Chartier, R. (2004) 'Languages, Books, and Reading from the Printed Word to the Digital Text', *Critical Inquiry*, vol. 31, no. 1, pp. 133–152.

Cover, R., Barnard, D., and Duncan, N. (1991) 'The Progress of SGML (Standard Generalized Markup Language): Extracts from a Comprehensive Bibliography', *Literary and Linguistic Computing*, vol. 6, no. 3, pp. 197–209.

Culkin, J. (1968) 'A Churchman's Guide to Marshall McLuhan', *Religious Education*, vol. 63, no. 6, pp. 457–462.

DeRose, S. J., Durand, D. G., Mylonas, E. and Renear, A. H. (1997) 'What is text, really?', *ACM SIGDOC Asterisk Journal of Computer Documentation*, vol. 21, no. 3, pp. 1–24.

Garrish, M. and Gylling, M. (2013) *EPUB 3 best practices: [optimize your digital books]*, Beijing, Köln, O'Reilly.

Goldfarb, C. (1999) 'The roots of SGML: A personal recollection', *Technical Communication*, vol. 46, no. 1, pp. 75–78 [Online]. Available at http://search.proquest.com/docview/220959508?accountid=14511.

Goldschmidt, D. E. and Krishnamoorthy, M. (2008) 'Comparing keyword search to semantic search: a case study in solving crossword puzzles using the Google™ API', *Software: Practice and Experience*, vol. 38.

Grue, D., Dobson, T. M. and Brown, M. (2013) 'Reading practices and digital experiences: An investigation into secondary students' reading practices and XML-markup experiences of fiction', *Literary and Linguistic Computing*, vol. 28, no. 2, pp. 237–248.

Hillesund, T. (2010) 'Digital reading spaces: How expert readers handle books, the Web and electronic paper', *First Monday - Peer–Reviewed Journal on the Internet*, vol. 15, no. 4 [Online]. Available at http://uncommonculture.org/ojs/index.php/fm/article/view/2762/2504 (Accessed 29 November 2015).

Hockey, S. (2004) 'The History of Humanities Computing', in Schreibman, S., Siemens, R. and Unsworth, J. (eds) *A Companion to Digital Humanities,* Malden, MA, USA, Blackwell Publishing Ltd, pp. 1–19.

Hood, Z. and Sahari Ashaari, N. (2013) 'Researchers Annotation Collections and Practices', *Procedia Technology*, vol. 11, pp. 354–358.

Landow, G. P. (2006) *Hypertext 3.0: Critical theory and new media in an era of globalization*, 3rd edn, Baltimore, Md., Johns Hopkins Univ. Press.

Le, T. N., Wu, H., Ling, T. W., Li, L. and Lu, J. 'From Structure-Based to Semantics-Based: Towards Effective XML Keyword Search', in, pp. 356–371.

Lie, H. W. and Saarela, J. (1999) 'Multipurpose Web publishing using HTML, XML, and CSS', *Communications of the ACM*, vol. 42, no. 10, pp. 95–101.

Liu, Z. (2005) 'Reading behavior in the digital environment', *Journal of Documentation*, vol. 61, no. 6, pp. 700–712.

Lloyd, A. (2015) *The Future of News Is Not An Article* [Online], nytlabs ← Research, thoughts, and process from The New York Times R&D Lab. Available at http://nytlabs.com/blog/2015/10/20/particles/ (Accessed 6 December 2015).

McGann, J. (2004) 'Marking Texts of Many Dimensions', in Schreibman, S., Siemens, R. and Unsworth, J. (eds) *A Companion to Digital Humanities,* Malden, MA, USA, Blackwell Publishing Ltd, pp. 198–217.

nytlabs (ed) (2015) [Online], nytlabs ← Research, thoughts, and process from The New York Times R&D Lab. Available at http://nytlabs.com/projects/editor.html (Accessed 6 December 2015).

Oxford Scholarly Editions Online (ed) (2015) [Online], Oxford University Press. Available at http://www.oxfordscholarlyeditions.com/ (Accessed 6 December 2015).

Qiu, R., Tang, Z., Gao, L. and Yu, Y. (2010) *A novel XML-based document format with printing quality for web publishing*, International Society for Optics and Photonics.

Russell, K. (2005) 'Markup languages', *Computerworld*, vol. 32, no. 39, pp. 30–31 [Online]. Available at http://search.proquest.com/docview/216077318?accountid=14511 (Accessed 29 November 2015).

Schubert, L. (2015) 'Computational Linguistics', *The Stanford Encyclopedia of Philosophy* Spring 2015 Edition, Edward N. Zalta (ed.). Available at http://plato.stanford.edu/archives/spr2015/entries/computational-linguistics/ (Accessed 15 December 2015).

Smith, F. (2004) *Understanding reading: A psycholinguistic analysis of reading and learning to read*, 6th edn, Mahwah, NJ, Erlbaum.

Smith, K. (2012) *The publishing business: From p-books to e-books*, Lausanne, Worthing, AVA Academia.

Smith, M. N. (2004) 'Electronic Scholarly Editing', in Schreibman, S., Siemens, R. and Unsworth, J. (eds) *A Companion to Digital Humanities,* Malden, MA, USA, Blackwell Publishing Ltd, pp. 306–322.

van Noorden, R. (2014) 'Online collaboration: Scientists and the social network', *Nature*, vol. 512, no. 7513, pp. 126–129.

Warwick, C. (2004) 'Print Scholarship and Digital Resources', in Schreibman, S., Siemens, R. and Unsworth, J. (eds) *A Companion to Digital Humanities,* Malden, MA, USA, Blackwell Publishing Ltd, pp. 366–382.

World Wide Web Consortium (ed) (2006) *1.1 (Second Edition): Extensible Markup Language (XML)*: World Wide Web Consortium [Online]. Available at http://www.w3pdf.com/W3cSpec/XML/2/REC-xml11-20060816.pdf (Accessed 22 November 2015).

Appendix

The following example illustrates how markup languages work.

The content stems from the Stanford Encyclopedia of Philosophy (Schubert, 2015); all other parts are created by the author: the tags, the encoded style, as well as the graphic of a digital reading device. It must be noted that the markup is an example; the content in the original source is differently tagged. There is not one solution of markup. Markup always has to be fit for purpose and can thus take different shapes.

As shown below, the text itself comprises no information about style or formatting and does not reveal any interpretation about itself to the processing machine. On the following pages, the text is shown as stored as an XML-file (pp. 19-20), as a manipulated HTML-file (p. 21) with an accompanying CSS-stylesheet (p. 22). When interpreted by a standard browser, the three files combined trigger the browser to display the text as shown on page 23.

The content without any style, as it is appearing in the source code of the Stanford Encyclopedia of Philosophy (Schubert, 2015).

```
Computational Linguistics
Computational linguistics is the scientific and engineering discipline concerned with
understanding written and spoken language from a computational perspective, and building
artifacts that usefully process and produce language, either in bulk or in a dialogue
setting. To the extent that language is a mirror of mind, a computational understanding of
language also provides insight into thinking and intelligence. And since language is our most
natural and most versatile means of communication, linguistically competent computers would
greatly facilitate our interaction with machines and software of all sorts, and put at our
fingertips, in ways that truly meet our needs, the vast textual and other resources of the
internet.
The following article outlines the goals and methods of computational linguistics (in
historical perspective), and then delves in some detail into the essential concepts of
linguistic structure and analysis (section 2), interpretation (sections 3–5), and language
use (sections 6–7), as well as acquisition of knowledge for language (section 8), statistical
and machine learning techniques in natural language processing (section 9), and miscellaneous
applications (section 10).
First published Thu Feb 6, 2014; substantive revision Wed Feb 26, 2014
“Human knowledge is expressed in language. So computational linguistics is very important.” –
Mark Steedman, ACL Presidential Address (2007)
Author: Schubert, Lenhart
Editor: Edward Zalta
Related entries: Anaphora, Artificial Intelligence, Logic and Artificial Intelligence,
Cognitive Science, Connectionism, Discourse Representation Theory, Philosophy of Linguistics,
Logical Form, Dynamic Semantics, Montague, Natural Language Semantics
```

The content as stored as an XML-file. As can be seen, the different textual elements are attributed with interpretative names and form discrete entities. The original content is bold.

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE Stanford Encyclopedia of Philosophy 'sep_main_content.dtd'>
<content encyclopedia='stanford encyclopedia of philosophy' language='english'>
   <chapter title='c'>
      <entry id='computational-linguistics'>
         <title>Computational Linguistics</title>
         <author id='lensch_987'>Schubert, Lenhart</author>
         <editor id='edwzal_987'>Edward Zalta</editor>
         <archive xlink:type='locator'
         xlink:href='http://plato.stanford.edu/archives/spr2015/entries/computational-
         linguistics/'>
            <edition>Spring 2015</edition>
            <year>2015</year>
         </archive>
         <group>Linguistics</group>
         <related>
            <rel xlink:type='locator' xlink:href='anaphora.xml'>anaphora</rel>
            <rel xlink:type='locator' xlink:href='ai.xml'>artificial intelligence</rel>
            <rel xlink:type='locator' xlink:href='logic-ai.xml'>logic and artificial
            intelligence</rel>
            <rel xlink:type='locator' xlink:href='cognitive-science.xml'>cognitive
            science</rel>
            <rel xlink:type='locator' xlink:href='connectionism.xml'>connectionism</rel>
            <rel xlink:type='locator' xlink:href='discourse-representation-
            theory.xml'>discourse representation theory</rel>
            <rel xlink:type='locator' xlink:href='philosophy-of-linguistics.xml'>philosophy of
            linguistics</rel>
            <rel xlink:type='locator' xlink:href='logical-form.xml'>logical form</rel>
            <rel xlink:type='locator' xlink:href='dynamic-semantics.xml'>dynamic
            semantics</rel>
            <rel xlink:type='locator' xlink:href='montague.xml'>Montague</rel>
            <rel xlink:type='locator' xlink:href='natural-language-semantics.xml'>natural
            language semantics</rel>
         </related>
         <published>06022014</published>
         <revision substantive='1'>26022014</revision>
         <quote>
            <author id='marste_987'>Steedman, Mark</author>
            <reference>Mark Steedman, ACL Presidential Address (2007)</reference>
            <content>Human knowledge is expressed in language. So computational linguistics is
            very important.</content>
         </quote>
         <abstract>
            <about>Computational linguistics is the scientific and engineering discipline
            concerned with understanding written and spoken language from a computational
            perspective, and building artifacts that usefully process and produce language,
            either in bulk or in a dialogue setting. To the extent that language is a mirror
            of mind, a computational understanding of language also provides insight into
            thinking and intelligence. And since language is our most natural and most
            versatile means of communication, linguistically competent computers would greatly
            facilitate our interaction with machines and software of all sorts, and put at our
            fingertips, in ways that truly meet our needs, the vast textual and other
            resources of the internet.
            </about>
```

```
            <locator>The following article outlines the goals and methods of computational
            linguistics (in historical perspective), and then delves in some detail into the
            essential concepts of linguistic structure and analysis (<section
            xlink:type='locator' xlink:href='computational-linguistics.xml/#synpar'>section
            2</section>), interpretation (<section xlink:type='locator'
            xlink:href='computational-linguistics.xml/#semrep'>sections 3–5</section>), and
            language use (<section xlink:type='locator' xlink:href='computational-
            linguistics.xml/#langen'>sections 6–7</section>), as well as acquisition of
            knowledge for language (<section xlink:type='locator' xlink:href='computational-
            linguistics.xml/#acqknoforlan'>section 8</section>), statistical and machine
            learning techniques in natural language processing (<section xlink:type='locator'
            xlink:href='computational-linguistics.xml/#stanlp'>section 9</section>), and
            miscellaneous applications (<section xlink:type='locator'
            xlink:href='computational-linguistics.xml/#app'>section 10</section>).
            </locator>
         </abstract>
         <introduction id='goamet'>1. Introduction: Goals and methods of computational
         linguistics
            <goal id='goacl'>1.1 Goals of computational linguistics
            …
            </goal>
            <method id='metcl'>1.2 Methods of computational linguistics
            …
            </method>
         </introduction>
         …
      </entry>
      …
   </chapter>
   …
</content>
```

The content as processed with an HTML-file. A processing application like JavaScript would get the content from the XML-file to manipulate it according to the HTML-tags. By processing, the attributes are interpreted. Thus, titles are processed as h1 or an abstract as paragraph (<p>). Still, the HTML-file does not comprise any information about the style in which the content should be displayed. Furthermore, the HTML-file does not reveal all information that the XML-source contains. This can be defined individually for any process as different devices may only show certain information. In this example, it was choosen to display the related entries list not in the main sction due to the size of the display of the reading device. However, the content itself is not affected by this choice as the XML-file still contains all information.

```
<!DOCTYPE html>
<link rel="stylesheet" type="text/css" href="sep_main_stylesheet.css">
<html>
   <header>
      <h1>Stanford Encyclopedia of Philosophy</h1>
   </header>
   <body>
      <h2>Computational Linguistics</h2>
      <published>First published Thu Feb 6, 2014; substantive revision Wed Feb 26,
      2014</published></br>
      <quote>“Human knowledge is expressed in language. So computational linguistics is very
      important.”
      <author>–Mark Steedman, ACL Presidential Address (2007)</author>
      </quote>
      <p>Computational linguistics is the scientific and engineering discipline concerned
      with understanding written and spoken language from a computational perspective, and
      building artifacts that usefully process and produce language, either in bulk or in a
      dialogue setting. To the extent that language is a mirror of mind, a computational
      understanding of language also provides insight into thinking and intelligence. And
      since language is our most natural and most versatile means of communication,
      linguistically competent computers would greatly facilitate our interaction with
      machines and software of all sorts, and put at our fingertips, in ways that truly meet
      our needs, the vast textual and other resources of the internet.</p>
      <p>The following article outlines the goals and methods of computational linguistics
      (in historical perspective), and then delves in some detail into the essential concepts
      of linguistic structure and analysis (<a
      href="http://plato.stanford.edu/entries/computational-linguistics/#synrep">section
      2</a>), interpretation (<a href="http://plato.stanford.edu/entries/computational-
      linguistics/#semrep">sections 3–5</a>), and language use (<a
      href="http://plato.stanford.edu/entries/computational-linguistics/#langen">sections 6–
      7</a>), as well as acquisition of knowledge for language (<a
      href="http://plato.stanford.edu/entries/computational-
      linguistics/#acqknoforlan">section 8</a>), statistical and machine learning techniques
      in natural language processing (<a
      href="http://plato.stanford.edu/entries/computational-linguistics/#stanlp">section
      9</a>), and miscellaneous applications (<a
      href="http://plato.stanford.edu/entries/computational-linguistics/#app">section
      10</a>).
      </p>
   …
   </body>
…
</html>
```

The CSS-file that accompanies the HTML-file. This stylesheet states how the attributed content of the HTML-file is displayed in a digital device, when processed by a compiler like a standard web browser. In this case, it defines the style as shown in the digital reading device on page 23. However, another stylesheet might contain definitions for other displaying purposes, or as a responsive design that shapes the content according to the digital reading device. In other words, the content can react to different stylesheets, whereas one stylesheet could be used for a variety of contents (other databases, content for books or articles). This illustrates the flexibility that is described in the essay.

```
<!DOCTYPE CSS>

body {
   background-color: #E0E0F8;
}
h1 {
   font-family: "Garamond","serif";
   font-size: 30px;
   color: #0B0B61;
   text-align: center;
   letter-spacing: 3px;
   text-transform: uppercase;
}
h2 {
   font-family: "Garamond","serif";
   font-size: 25px;
   color: #0B0B61;
   text-align: center;
}
published {
   font-family: "Ebrima","sans-serif";
   font-size: 18px;
   color: #555555;
   font-style: italic;
}
quote {
   font-family: "Ebrima","sans-serif";
   font-size: 18px;
   color: #555555;
   font-style: italic;
}
author {
   font-weight: bold;
}
p {
   font-family: "Ebrima","sans-serif";
   font-size: 20px;
   color: #444444;
}
a {
   color: #0B0B61;
   font-style: italic;
}
…
```

The content as displayed in a digital reading device. The content looks differently than the original source (Schubert, 2015). This is due to the different stylesheets that the original source and this example use.