

# Discourse knowledge in device independent document formatting

Joost Geurts, Jacco van Ossenbruggen, Lynda Hardman

CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands  
Firstname.Lastname@cwi.nl

**Abstract.** Most document structures define layout structures which implicitly define semantic relationships between content elements. While document structures for text are well established (books, reports, papers etc.), models for time based documents such as multimedia and hypermedia are relatively new and lack established document structures.

Traditional document description languages convey domain-dependent semantic relationships implicitly, using domain-independent mark-up for expressing layout. This works well for textual documents as, for example, CSS and HTML demonstrate. True device independence, however, sometimes requires a change of document model to maintain the content semantics. To achieve this we need explicit information about the discourse role of the content element.

We propose a model in which content is marked-up with the discourse role it plays in the document. This way the formatter has knowledge about the function of a content element so it can make appropriate layout choices.

## 1 Introduction

The overall document model for text, i.e. chapter, section and paragraph, is applied by authors to present their message. In other words, the author uses layout functionality to support the argument to be presented. In a traditional document engineering task an author marks up the content of a document. By doing so, the semantic function of the content is made explicit. This then can be used by a style sheet to transform the document into a final form format. An example is this paper which was marked up in  $\text{\LaTeX}$ , defining sections, subsections, bibliography etc. The marked-up document can (in several transformation steps) generate various output formats for different platforms, such as Postscript for paper and HTML for hypertext. Making semantics explicit (through document structure) allows us, therefore, to abstract from the final format of the document. Although  $\text{\LaTeX}$  defines a rich environment in which semantic functions can be specified (e.g. list, emph) there is a point after which content is considered to be atomic: figures, for example, are black-boxes of which  $\text{\LaTeX}$  knows nothing except their dimensions (which are specified by the author). A figure can thus not be reformatted when it does not fit on a page. To some extent the same holds for plain-text paragraphs in which the relations between words<sup>1</sup> and sentences are not made explicit. The layout of the text can break sentences between words and make sure they fit the available space. There are few cases where a  $\text{\LaTeX}$  processor cannot solve the layout constraints satisfactorily because in text the atomic units (words) are small compared with the column width.

---

<sup>1</sup> LaTeX does know about hyphenation of words.

## 2 Multimedia versus Text

In contrast, when we apply the textual formatting model to multimedia content, the atomic units (media items) are large (in the spatial dimension) compared with atomic units (words) in a text model. As a consequence, document structure for multimedia is not as fine grained as its textual equivalent. Therefore conveying semantics in a multimedia document is, just as a figure in  $\text{\LaTeX}$ , the responsibility of the author who understands the semantics of the atomic unit.

Furthermore, a textual document model uses a spatial flow model while a multimedia document uses a temporal model. In case that the content does not fit the screen a spatial model can use different ways to compensate this shortcoming, such as scrolling or multiple pages. In a temporal model the flow dimension happens in time, interaction is, therefore, not really an option<sup>2</sup>. One needs to decide beforehand how much time is scheduled to view a scene/page and this requires knowledge about the content displayed.

The difference between text and multimedia, from a layout point-of-view, relates to the document model. A document model in general is a discourse model which the author uses to structure the content and facilitate the communication of the intended message to the reader. Within a textual document model the discourse relationships between the content elements are expressed using chapter, section, subsection etc. The visual appearance of the document reflects the document structure by using, for example, different font sizes, to express different levels of heading.

Within a multimedia document however there exist no well defined document structure to express discourse relationship of content elements through layout. Instead an author of a multimedia documents typically conveys these discourse relationships between media items (or groups of media items) by using design constructs such as alignment, by using similar background colours (spatial) or use of transitions (temporal). Note that these relationships are not explicitly defined in the document model. In contrast with textual documents where grouping of text is realized by using chapters, sections etc.

In summary, traditional document description languages convey domain-dependent semantic relationships implicitly using domain-independent mark-up for expressing layout.

## 3 Device-independent Multimedia

Furthermore, document models are traditionally developed for a defined medium, using, and being limited by, the characteristics of the media. A textual document model does not include a temporal dimension because paper does not have one. The dependency also works the other way round: devices are developed to fit a certain media [1,2]. A PDA was designed to be used as an agenda, to store addresses or view small notes. In general, any particular device typically has a limited set of document types it can display. This means that for true device-independent authoring, any particular document

---

<sup>2</sup> Although multimedia documents can potentially include interaction, this typically has more far-reaching consequences on the presentation than adding a scrollbar in textual document.

might need to switch document type. For example, a textual document with an accompanying picture might be presented on a PDA, where the screen size is too small to present both text and images simultaneously. A solution is to “reformat” the presentation for the smaller screen as a slide-show of only the images with the accompanying text presented simultaneously as synthesized speech. In this case, the document model has changed from being spatial/textual(text-flow) to spatial/temporal(audio) [4,5]. Because of the lack of explicit semantics expressing the relationships among the media items in the presentation, such a transformation can currently not be made automatically.

#### **4 Discourse Role as Explicit Metadata**

A proposed first step towards a solution is to encode explicit, in RDF, the semantic role of the content components within a document model. This means elements within a document are structured according to their discourse function. For example a typical document contains a section “Introduction”, the main body and a section “Conclusion” [3]. Currently the formatting engine sees no difference. If it would, however, then it could, for example, decide to present an executive view and show the conclusion first. From a cross device authoring perspective explicitly knowing that an image is an example of a concept explained in a piece of text enables the engine to synthesize the text for a PDA and make sure the image is displayed when the text is spoken.

#### **5 Conclusion**

Traditional document description languages convey domain-dependent semantic relationships implicitly, using domain-independent mark-up for expressing layout. Content adaptation, for example in device independent authoring, sometimes requires a transformation of document model. Because of the lack of explicit semantics expressing relationships among the content elements in the presentation, such a transformation can currently not be made automatically. In this document device independent authoring is highlighted, however, this is just one of multiple dependencies between content (the message), media, medium, user and device. Similar arguments can, for example, be made for user adaptation. In this perspective, the problem of meta data for content adaptation is not so much a problem of meta data for individual media items but more about the context in which they are used.

#### **References**

1. *W3C Workshop on Web Device Independent Authoring*, Bristol, UK, October 3-4, 2000.
2. *W3C Delivery Context Workshop*, INRIA Sophia-Antipolis, France, March 4-5, 2002.
3. *Second International Semantic Web Conference (ISWC2003)*, Sanibel Island, Florida, USA, October 20-23, 2003.
4. J. van Ossenbruggen, J. Geurts, F. Cornelissen, L. Rutledge, and L. Hardman. Towards Second and Third Generation Web-Based Multimedia. In *The Tenth International World Wide Web Conference*, pages 479–488, Hong Kong, May 1-5, 2001. IW3C2, ACM Press.

5. J. van Ossenbruggen, J. Geurts, L. Hardman, and L. Rutledge. Towards a Formatting Vocabulary for Time-based Hypermedia. In *The Twelfth International World Wide Web Conference*, pages 384–393, Budapest, Hungary, May 20–24, 2003. IW3C2, ACM Press.