Support for Semantic Documents in Protégé

Henrik Eriksson

Dept. of Computer and Information Science
Linköping University
SE-581 83 Linköping, Sweden

her@ida.liu.se

Introduction

There is a significant amount of knowledge in written documents, such as internal reports, printed publications, and textbooks. Such documents are often a rich source of knowledge because they are authored and reviewed carefully. However, there is little integration between documents and knowledge bases. Although it is possible to use documents as the basis for knowledge acquisition, ontology engineering, and for populating large knowledge base, it is a one-way development process. In current system architectures, there is surprisingly little bidirectional interaction between documents and knowledge bases. A significant disadvantage is that it is easy to loose the references of the source documents. Furthermore, there is insufficiently support for document integration in today's development tools and performance environments.

There is much to be gained from integrating documents and knowledge bases in better ways. Our approach is to take advantage of documents enriched with knowledge bases. Such *semantic documents* support regular on-line viewing and printing while containing semantic information linked to the document itself (e.g., metadata describing document properties) and to the textual and graphical content of the document (e.g., as document annotations). In many ways, this document format is related to the Semantic Web approach [1], which among other things promises to enrich web pages with (i.e., HTML-encoded documents) with semantic information, such as metadata describing page content. However, there are other document formats that are commonly used for documents with much richer knowledge contents than most web pages. Specifically, authors and publishers use the popular Portable Document Format (PDF) [2] to store documents suitable for both printing and on-line viewing. Support for PDF in Protégé development tool [3] would thus be an important step towards improving the usability of documents and knowledge-based systems.

Because textual knowledge in documents are closely linked to knowledge in knowledge bases, we believe that semantic documents should combine these representation formats in a common format that supports the viewing, printing, and reasoning tasks equally well. We use an enriched version of PDF as the basis for semantic documents. Furthermore, we use a Protégé extension to store knowledge bases in dedicated streams inside PDF files and to link sections of the text to classes and instances/individuals in the knowledge base.

Semantic Documents

Semantic documents combine printable electronic documents with knowledge bases by annotating document pages with semantic information expressed in a knowledge-representation language. There are several advantages of using PDF as the basis for semantic documents. PDF is a widely-used file format that supports high-quality printing and on-line viewing of documents with a fixed, page-oriented layout. Because PDF is an open format [2], third-party programs can read and write the files for various purposes, such as automatic document generation and document indexing. In addition, PDF is extendible in the sense that it is possible to add information in new sections (streams) inside document.

The Extensible Metadata Protocol (XMP) is an XML-based format developed by Adobe for storing metadata [4]. Adobe uses XMP to encode metadata for several file formats, such as PDF and other storage formats used by their products. XMP takes advantage of RDF statements to represent the file metadata (e.g., title, subject, author, creation date, and producer software). Although XMP is an important step towards semantic document descriptions, it cannot be used to encode knowledge bases in the OWL format (because of incompatible uses of RDF).

Knowledge bases in OWL and other formats should be stored separately from XMP. In addition to XMP-based metadata, we use a special stream and dictionary entry in the document to store the Protégé OWL knowledge bases (see Figure 1). It is possible to annotate the pages in the PDF file with references to the knowledge base. The resulting semantic documents support multifaceted tasks, such as decision support, semantic search, reasoning, and consistency checks.

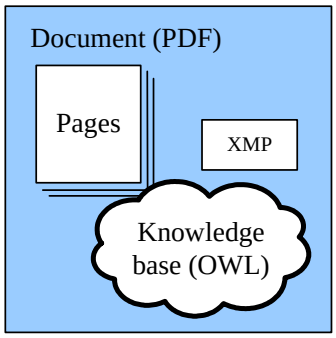

Figure 1. Document format. The PDF file contains the printable document as well as the knowledge base in a separate section.

Tool Support for Semantic Documents

Tool support is an essential component of the semantic-document approach. Because of the complexity of the internal file structure, however, the annotation process is complicated and requires adequate tools. One possibility is to generate automatically the documents complete with annotations from knowledge bases [5]. In addition, it is possible to extend Protégé to support new file formats thought the regular extension mechanisms (e.g., tab widgets).

The PDFTab extension adds support for creation and maintenance of semantic documents in PDF. A major achievement of PDFTab is that it integrates Adobe Acrobat with Protégé. The extension runs an Acrobat window inside a Protégé tab in the same way web browsers allow viewing of PDF documents by running Adobe Acrobat inside a browser window. Figure 2 illustrates the extension architecture for PDFTab.

It is often advantageous for developers to work on collections of documents associated with a knowledge base (e.g., for the knowledge-management purposes). PDFTab provides a list of documents that are referenced by the knowledge base. Developers can select a document from this list and open it in the Adobe Acrobat section of PDFTab. Here, it is possible to view the document and to add new annotations to it. These annotations link to individuals in the knowledge base. Developers can view and edit the corresponding individual by double-clicking on the annotation. Figure 3 shows annotation of an open document in PDFTab.

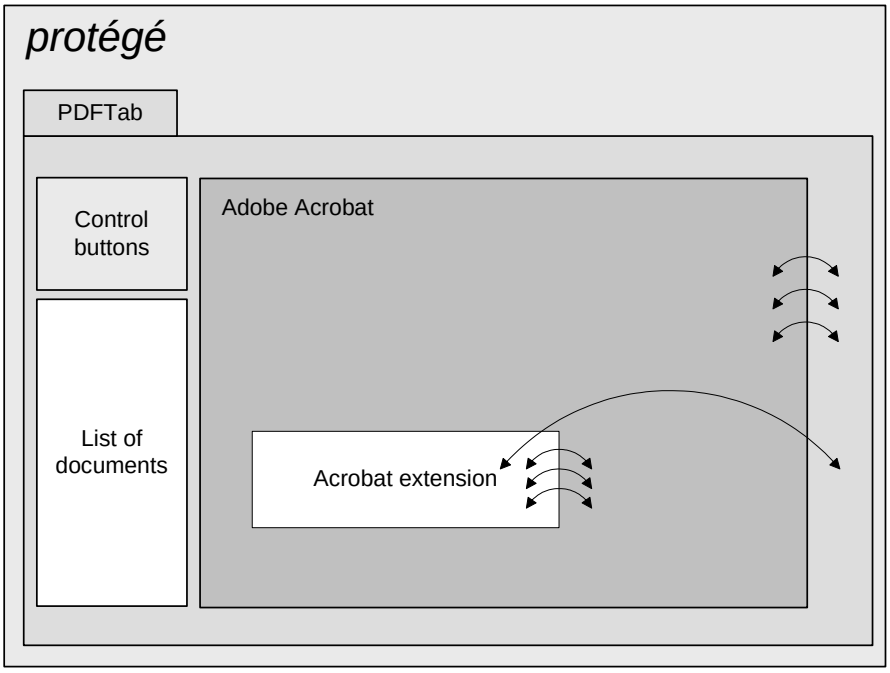

Figure 2. The extension architecture. PDFTab runs Adobe Acrobat inside its tab area. PDFTab communicates with Acrobat directly and with a custom extension to Acrobat that supports annotations.

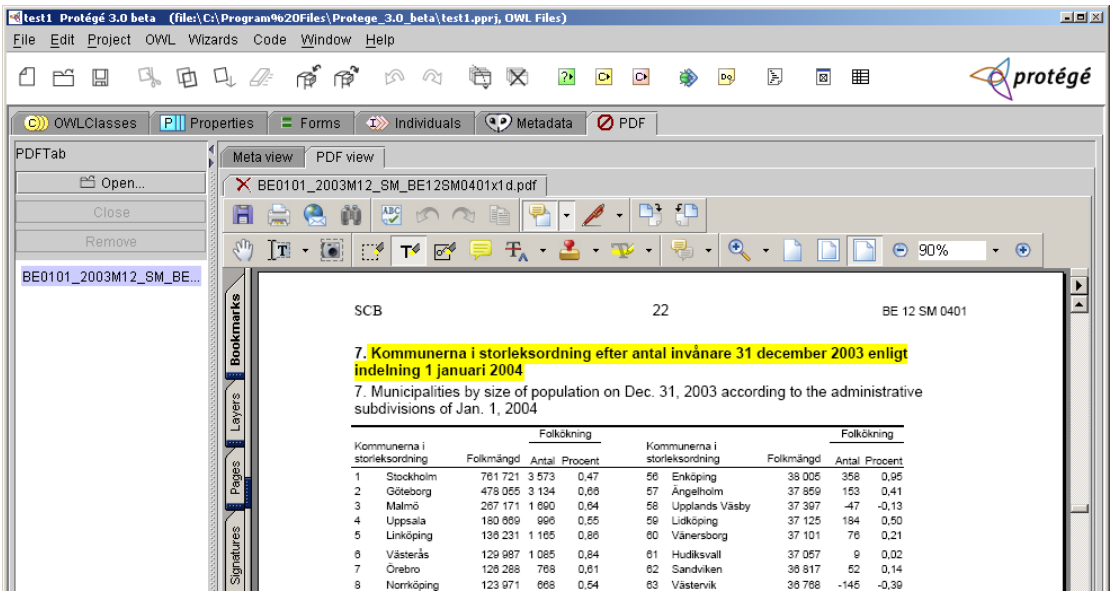

Figure 3. Annotation of a PDF document in Protégé. The annotated document text (highlighted with yellow background) refers to the corresponding instance in the Protégé knowledge base. In this example, the document is a statistics report where the annotated table headings are linked to a knowledge base describing the domain.

Summary and Discussion

We believe that the semantic-document approach is a promising method for bridging the gap between documents and knowledge bases, and that there are many uses for such enriched documents. Appropriate tool support for developers is an essential aspect of this approach. It is possible to extend Protégé to support semantic documents by adding functionality for PDF handling. The current implementation allows us to experiment with the creation and maintenance of semantic documents and the development of systems that take advantage of such documents for reasoning and search.

Acknowledgements

This work was supported by Vinnova (grant no. 2003-01415).

References

[1] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, May 2001, pp 35–43.

[2] Adobe Systems Inc. PDF Reference, fifth edition: Adobe Portable Document Format version 1.6, 2004. (Available at http://partners.adobe.com/public/developer/en/pdf/PDFReference16.pdf)

[3] John H. Gennari, et al. The evolution of Protégé: An environment for knowledge-based systems development. *International Journal of Human–Computer Studies*, 58(1):89–123, 2003.

[4] Adobe Systems Inc. XMP Specification, 2004. (Available at http://partners.adobe.com/public/developer/en/xmp/sdk/xmpspecification.pdf)

[5] Henrik Eriksson, Samson W. Tu, and Mark Musen. Semantic Clinical Guideline Documents. [Submitted for publication]