The Cross-Media-Publishing Framework *openFuXML*

Thorsten Kisner
Department of Communication Systems
Faculty of Mathematics and Computer Science
FernUniversität in Hagen, Germany
thorsten.kisner@fernuni-hagen.de

Jan Gellweiler
Department of Communication Systems
Faculty of Mathematics and Computer Science
FernUniversität in Hagen, Germany
jan.gellweiler@fernuni-hagen.de

Firoz Kaderali
Department of Communication Systems
Faculty of Mathematics and Computer Science
FernUniversität in Hagen, Germany
firoz.kaderali@fernuni-hagen.de

Abstract: The cross-media-publishing and rendering framework *openFuXML* is a further development of parts of the content management framework *FuXML* which is in use at the FernUniversität in Hagen for the production of printed and electronic course material. It became open source in February 2007 and is offered in the CampusSource Software Exchange. *openFuXML* can be used to author scientific documents and courses and then render them in diverse output formats. In this paper we summarize the features of *openFuXML*, hi-light the latest developments and give an overview of further developments.

Introduction

Since its founding in the year 1972 the FernUniversität (University of Hagen) – the first and only distance teaching university in German-speaking countries – has offered course material in form of printed paper to its students. Traditionally these printed materials are posted to the students every two weeks, 7 times per semester. For an increasing number of courses the printed study material is supplemented by slides, audio and video, educational software or home kits. The multimedia format was introduced in 1995; the current multimedia format is HTML with Flash animations and Java applets as well as Portable Document Format (PDF) for printing.

Most of the 1,400 courses at the FernUniversität exist only as printed material created with word-processing or typesetting software such as Microsoft Word or LaTeX. Some courses are published as CBT on CD-ROM and are typical examples for specially and individually funded multimedia course development. It is economically infeasible to raise funds usually required for the production of high quality multimedia courses for each individual course offered by the FernUniversität. Our experience shows that students happily accept hypermedia courses but also request high quality printed course material. Thus a single (multimedia) version does not meet their needs.

This illustrates the need for a cross-media-publishing system. Due to the fact that such systems were previously not in use, the task of creating multiple presentation versions of course content was left to the faculty staff. Of course this situation was not satisfactory and the solution to this was considered to be cross-media-publishing. The basic idea hereof is the production of different media types such as web pages, CD-ROMs or books from one single data repository (single source production). Content has to be saved in a media-neutral manner, which results in the clear separation of content and layout.

At the end of 2002 we evaluated different existing frameworks but found nothing suitable for our needs and started the development of our own cross-media-publishing system *FuXML*. The project started in 2003 and was funded up

to 2005. Development was then continued by the Department of Communication Systems and the Centre for Development of Distance Teaching. In February 2007 *FuXML* was renamed to *openFuXML* and then published as open sources software (GPL license) in the CampusSource Software Exchange. CampusSource is the largest repository worldwide specializing in learning management systems and tools under the GPL and is financially supported by the Ministry of Science and Research of the Federal State of North Rhine-Westphalia (Germany). There are currently 10 complete learning management systems (LMSs) and 9 other modules available in the software pool. In Germany nearly half of all the 330 universities are using CampusSource LMS, worldwide more than 5,300 institutions are registered. There are several conditions for the acceptance of open source software in CampusSource: Each product must be used on a long term basis, it is reviewed by two independent experts and basic requirements like documentation, an open source license and the non-use of commercial software must be fulfilled.

Media Concept

Tresman (Tresman 2002) describes a strategy for addressing student retention issues developed at the Open University in the UK. This strategy consists of six points; one of them is course design. The course design and structure should anticipate possible student problems. The introductory information should be useful and clear, and the course structure should be motivating and engaging. The use of media and technology should be suggestive.

The media and didactical concept of *openFuXML* is an inherent part of the system. The technical realisation of this concept is an XML Schema Definition (the current version still uses a Document Type Definition) which represents the underlying concepts. One of the best known frameworks for XML authoring is *DocBook*. Although we used some concepts from *DocBook* there are several reasons against using this format for e-learning content. The main argument is the simple fact that *DocBook* is highly optimized for technical documentation and not for e-learning.

Another aspect is the huge amount of elements in *DocBook*. Authors have to get at least an overview of all available elements and the diversity of about 400 elements is, especially for new users, confusing, while there are only about 100 elements in *openFuXML*. *DocBook* also enforces strict logical markup. For high-level structures like chapters, sections or paragraphs this is valuable, but often the author just wants to highlight a few words. While this is prohibited in *DocBook*, authors can use these formatting elements in *openFuXML* without the need of a special didactical intention.

However, *DocBook* still provides excellent examples of context-independent markup elements. For example we use exactly the same models for tables and subscripts and optimized the *openFuXML* DTD for e-learning content. The following figure shows the five main categories of elements of the *openFuXML* schema.

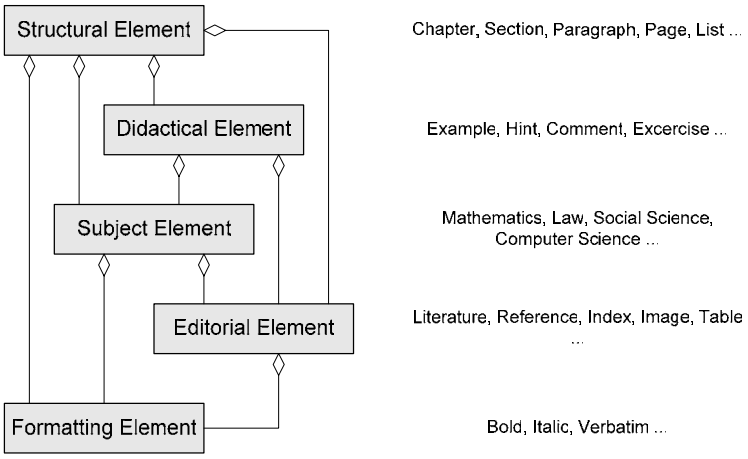

Fig. 1: Hierarchy of didactical und structural elements

The **structural** and **formatting elements** as well as editorial elements are well known from other word processing systems: common structural elements like sections, paragraphs or lists are available as well as visual formatting like bold, italic, verbatim or underline. Different markups like footnotes, marginalia or literature references are supported by openFuXML and belong to the **editorial elements**. The numbering of all enumerable elements is done automatically and lists of abbreviations, formulas and figures are generated and linked with the corresponding origin. Mathematic formulas are inserted in LaTeX style and rendered as images for the HTML output. During the development we considered MathML, but many browsers interpret formulas in the wrong way or need additional plugins to show them. Therefore we decided to use the established LaTeX syntax and rendered the formulas to PNG or GIF.

We worked together with professors from different faculties to be able to offer special environments for particular target groups such as mathematics, law, social sciences, engineering or computer science. These environments are represented in different **subject elements**. Fig. 2 shows a subject element from a computer science module, on the left side the author's view (editor) and the corresponding XML document on the right side.

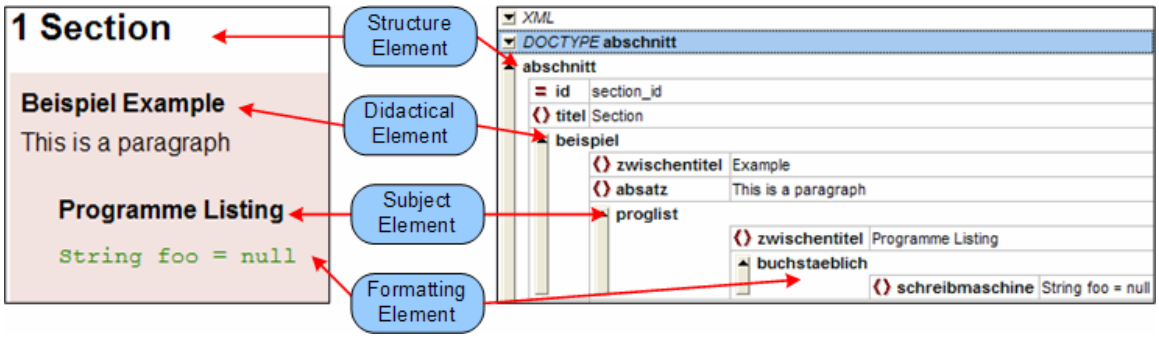

Fig. 2: Example of a subject element from a computer science module

The available **didactical elements** cover a wide range of concepts like examples, hints, exercises and solutions, author information, prerequisites and learning targets.

System Design

The rule-based rendering process produces the desired output. Currently there are two major formats available: the Portable Document Format (PDF) as an optimized format for printing and HTML for online viewing. With specially designed CSS style sheets it is even possible to produce an optimized HTML output for handheld devices.

The rendering engine of *openFuXML* is able to process a whole course from different XML documents. This makes it possible to separate the course structure from the content and thus the author is able to create different courses (e.g. for Bachelor and Master Studies) from the same course content. Only the structure definition varies and makes it easy for the authors to maintain both courses, because only a single content source must be edited.

The software package available at CampusSource contains a complete spectrum of rendering and publishing features. The rendering server is currently optimized for installation on Linux based systems. Users can access repository and output folders via file shares. The user front-end is a clearly designed java application which starts the production and displays the results.

An enterprise edition for a distributed environment features load balancing for pluggable rendering servers, a distributed content repository and an improved client application. The main part of this release is a J2EE application server which manages the connections to clients and handles the different rendering servers. Rendering servers can be connected and disconnected during runtime, if they are currently rendering a job, it will be forwarded to the next available application server.

Pictures and other media elements are stored in different formats, each optimized for the output type. For printing we use Encapsulated Postscript (EPS) and for HTML normally Portable Network Graphics (PNG). Animations and simulations can be inserted in all available formats, mainly Flash animations and Java Applets. For printing an alternate media-object is available. This can be a simple screenshot of an animation, or a completely different media object like a text description or a flowchart.

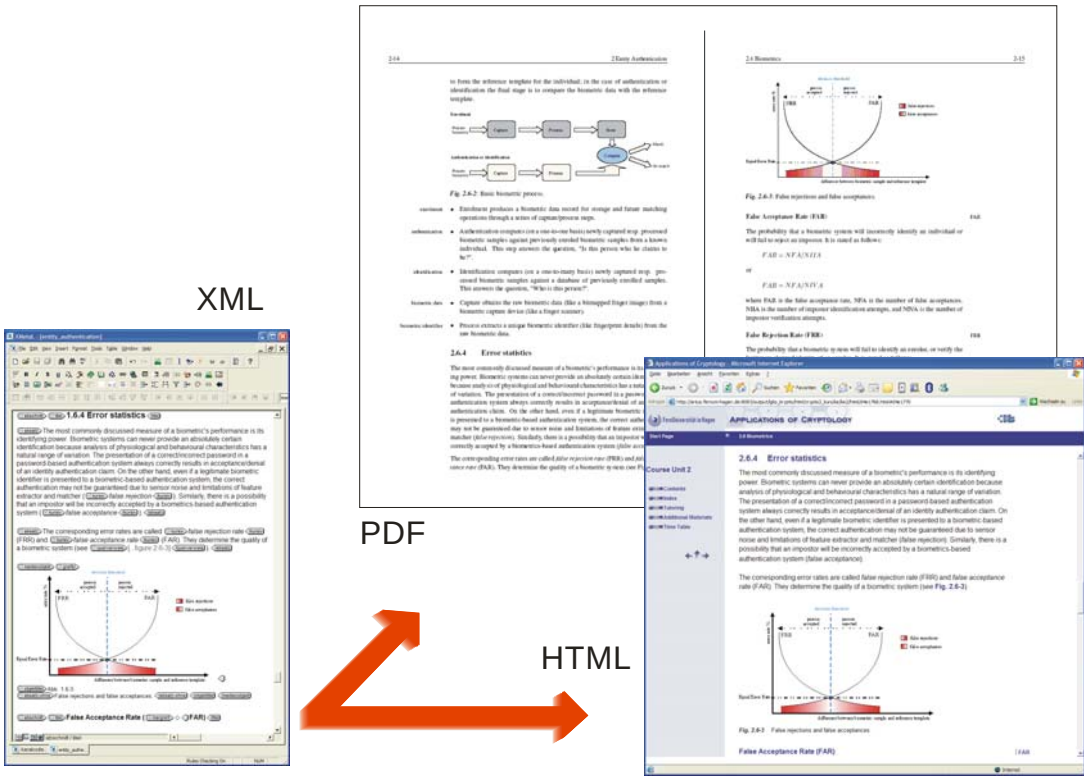

Fig. 3: Cross-media-publishing of an openFuXML-document

Lessons learnt

The decision for the structured markup language XML as source format for our publishing process has proved to be a good choice. All structural and didactical requirements of authors and study branches could be mapped with XML and aggregated in a Document Type Definition (DTD). The changeover of the authors from an unstructured authoring environment such as Word or FrameMaker to an authoring environment which demands a structured format turned out to be a bigger problem than the technical side dealing with XML. On account of this it was important for the success of the project to accommodate the authors as far as possible with the selection of the authoring tool, i.e. to offer a tool that in most instances hides the underlying XML structure from them and conforms with the everyday usage of Microsoft Office products. We achieved this by customizing the editor XMetaL of JustSystems.

Furthermore it turned out that in order to ensure author acceptance of such a system it is necessary to soften the strict separation of content and design in some instances. Thus our DTD directly supports basic markups, e.g. bold, italic or underlined, instead of abstract elements like emphasize1, emphasize2 and emphasize3.
For the rendering of the printout in the Portable Document Format (PDF) the usage of XML Formatting Objects (XML-FO) is convinient. However the results, which are achievable with non-commercial XML-FO packages do not meet our requirements regarding the micro- and macro-typography of our printed study material. The use of LaTeX as an intermediate step of the rendering process results in the desired quality of the print version.

Conclusions and Future Work

Since the beginning of 2006 *FuXML* is being used by the whole university for content production and has been greeted positively by authors. We offered tutorials for introducing *FuXML* and for the migration of the present content to the new authoring system. Especially for non-technical oriented faculties this was very helpful to overcome reservations against structured authoring.

Further development of openFuXML is still in progress. For the end of 2007 we have planned a completely windows based version and are working on the integration of additional editors. OpenOffice already offers XML support and Microsoft will also offer this in Office 2007. In Microsoft's zip compressed file format "dox" embedded media objects will also be stored along with XML files. Our goal is to use Word 2007 and OpenOffice as an editor and to use the word file directly as the source file for further processing.

Another near-future goal of our project is to make SCORM content packages available as an additional output format and hereby provide an easy possibility to use *openFuXML* contents with current learning management systems (LMSs). The rendering process of *openFuXML* provides an easy way to extend the produced HTML-pages with SCORM function calls (e.g. LMSInitialise() for version 1.2 or Initialize() for version 2004) whereby interactive sharable content objects (SCOs) instead of pure assets could be produced. Due to the XML-based source format of *openFuXML* contents and the structural elements defined in the schema definition all necessary prerequisites for the partitioning of the contents in SCOs and the generation of the manifest file are given. Both SCORM 1.2 and SCORM 2004 should be supported by *openFuXML*. SCORM 1.2 because of the wide range of LMSs supporting this specification and SCORM 2004 because of the support of inter-SCO linking.

In cooperation with other CampusSource projects an integration process of a common XML structure for e-learning content has been started. Our goal is the convergence of different authoring tools and learning management systems to create exchangeable content.

References

CampusSource. (2007). http://www.campussource.org

Steinkamp, G., Bullig, A., Heuel, E., Kaderali, F., Müller, H., Schlageter, G. (2005). Developing and Deploying an XML-based Learning Content Management System at the FernUniversität Hagen, *e-learning and education (eleed).*

S. Tresman. (2002). Towards a Strategy for Improved Student Retention in Programmes of Open Distance Education: A Case Study From the Open University UK, *International Review of Research in Open Distance Learning*, September 2002.

Walsh, N., Muellner, L. (1999). *DocBook: The Definitive Guide*, O'Reilly Media.

SCORM 1.2. (2001). http://www.adlnet.gov/scorm/history/12/documents.cfm

SCORM 2004 3rd Edition. (2006). http://www.adlnet.gov/scorm/20043ED/Documentation.cfm