

CaseML – Enabling Multifaceted Learning Scenarios with a Flexible Markup Language for Business Case Studies

Hülya Topcuoglu, Katharina Reinecke, Stefanie Hauske, Abraham Bernstein

Department of Informatics, University of Zurich, Switzerland
topcu@ifi.unizh.ch, reinecke@ifi.unizh.ch, hauske@ifi.unizh.ch, bernstein@ifi.unizh.ch

Abstract: This paper introduces a markup language for multimedia and modularized business case studies in the field of information systems. While most case studies, both paper-based and e-learning, are created for a well-defined teaching and learning scenario with precise instructions for the overall setting in the teaching notes, case studies as described in this paper should be suitable for a flexible and modularized usage for different assignments and in various teaching and learning scenarios. The necessary flexibility for such case studies can be guaranteed by our markup language - CaseML - that fulfills the various requirements.

Introduction

Case studies are an efficient method to achieve higher learning objectives like problem-solving and decision-making. Furthermore, the usage of multimedia case studies in an e-learning environment allows to put learners into a more realistic business environment by incorporating videos and simulations. Within the context of the e-learning project “Cases in Information Systems (CasIS)”, four case studies in the style of Harvard business cases (Roberts, 2001) have been created in cooperation with companies. An early challenge during the project were following essential requirements to the case studies’ design, which arose from the project partners’ need for a highly flexible and modularized usage of the CasIS case studies: Different views on the case for teachers and learners, flexible use of case studies in various teaching and learning scenarios, support of different assignments, realistic case studies through the use of multimedia, different output formats, and platform independence. In order to meet these requirements, we developed a markup language. The markup language CaseML is the first attempt to describe a case study in a structured manner and to modularize the content so that the case can be adapted to different assignments or employed in various teaching and learning scenarios. CaseML thereby constitutes a versatile foundation for various application possibilities and their amplification for additional requirements, such as adding adaptivity.

After giving a project overview, we describe the requirements mentioned above in more detail and examine existing markup languages with regard to their relevance for CasIS. The markup language CaseML is introduced in the main part of the paper. Here, we focus on describing how CaseML meets the project’s requirements. The main part closes with an overview of the first experiences and with a discussion of the benefits of CaseML. We finish our paper with a conclusion and give an outlook on future work.

Project Overview

The e-learning project “Cases in Information Systems (CasIS)” is a collaborative project of four Swiss institutions of higher education. Apart from the University of Zurich, the University of Lausanne and the Swiss Federal Institute of Technology in Lausanne and Zurich are involved. During the course of the project from January 2006 to July 2008, e-learning modules for case-based courses on master level in the field of information systems are developed and implemented into the project partners’ curricula. The overall project aim is to meet higher learning objectives and to address a heterogeneous target audience. Courses in information systems on master level typically aim at higher level learning goals, such as the application of knowledge and the evaluation of data and information, and students learn how to deal with complex problems and how to make well-founded decisions. A highly appropriate method to achieve these learning goals are case studies, in which a realistic situation or a problem of an organization are described and the learner has to adopt the position of a decision-maker and problem-solver (Roberts, 2001). Participants of master courses have in general very different pre-knowledge and competencies depending on their prior academical education and/or their work experience.

The case studies created during the CasIS project are not only realistic and lively by incorporated multimedia like video interviews and simulation. They are also more interactive due to the possibility to interact with teachers and co-learners. In order to address the heterogeneous target group and to reach the project goal, CasIS comprises the following four e-learning components:

- *Online Entrance Test*: Learners can assess the range of their knowledge required for successfully working through the whole case study process and get hints to relevant learning materials to fill their knowledge gaps.
- *Online Preparation Modules*: Learners can acquire missing declarative knowledge by working through self-contained e-learning modules that cover a wide range of information system relevant topics.
- *CasIS Toolbox*: Learners can make use of a toolbox during their case study work. The toolbox contains a collection of relevant methods and tools for analysis and problem-solving that are needed to prepare a specific case.
- *Real Case Studies*: Learners get the necessary data and information on the specific case from a multimedia online case study. Four case studies were produced in cooperation with companies on information system relevant topics.

Based on the initial online entrance test, learners will be offered a proposed learning path through the online preparation modules. Working on the online preparation modules should guarantee that learners subsequently share a similar level of declarative knowledge. The actual case study work is mainly based on the real case study modules and is complemented by the toolbox.

The Project's Essential Requirements on the Implementation of Case Studies

Out of the overall project aim and the project partners' need for a highly flexible and modularized use of the case studies, arise a number of essential requirements, which are described in some detail below:

1. Harvard Business Case: A Harvard Business case is a description of a real-life business situation. The case specifies a person in an organization who is involved in a decision, a problem or a challenge. By using cases for teaching, learners are supposed to take on the role of the decision-maker and to apply knowledge, to analyze or to evaluate data and information. Therefore, we decided to develop field-based and decision-oriented cases according to the Harvard business case standard.

2. Different views for teachers and learners: As in conventional learning environments, where teachers often have access to additional material, the web-based work on case studies requires two separate views, providing supplementary information and teaching notes for the lecturer, and an adequate learning environment for the student. Essentially, the arbitrary access to all materials for teachers, while at the same time denying learners access to teaching notes, has to be ensured.

3. Support for different assignments: A single case study can offer a variety of applicable student assignments, which usually necessitates the knowledge of only certain text and multimedia paragraphs of the case, whereas others can be completely unimportant for problem-solving. The support of different assignments, hence, requires the possibility to assign parts of the text and relevant multimedia elements to the particular exercise(s). According to the selected assignment, only associated parts of the case study should be presented to the learner.

4. Flexible teaching and learning scenarios: In order to allow for customizable teaching and learning scenarios, the online platform has to incorporate a modifiable frame for case studies. For instance, scenarios have to be applicable for the use in a 90-minutes-class to the continuous use in a course over the period of four weeks, thereby providing high adaptability to the teacher's preferences.

5. Increasing realisticness through the use of multimedia: Case studies live from a realistic description of the case. Hence, one of our aims in this project is the development of web-based multimedia case studies, where, for example, a video presents the manager of an organization who explains the business problem, or a diagram visualizes data-sets.

6. Different output formats: Our main focus lies on presenting the developed case studies in a web-based learning environment, which enables an easy linking of all additional materials, such as the entrance test, the preparation modules and the toolbox. However, from previous e-learning projects, we have gained the experience that learners advocate the possibility to get a print version of textual information.

7. Platform independence: As our project partners use different learning management systems (LMS), such as OLAT, WebCT Vista, or Moodle, all case studies must be developed in a way that they are independent of a specific e-learning platform, so that the re-usability of the case will be ensured.

Related Work

Due to the complexity and the variety of these project's requirements, the most applicable solution was the use of XML. Therefore, we restricted our evaluation of related work to existing markup languages in the field of e-learning. A markup language is either very common, so that it can be used in a wide area, or it considers the whole domain knowledge and can only be used in a specific area. In this regard, CaseML primarily bases on the markup language eLML (eLesson Markup Language) for declarative knowledge (Bleisch & Fisler, 2005). Previous approaches for mapping case studies to eLML showed that their structure is strongly predetermined through the requirements of eLML (Niederhuber et al., 2005). As the markup language accounts to neither the need for content modularization nor to our request to include domain knowledge, it did not sufficiently comply with our requirements for the use of case studies. Another applicable approach is the development of a markup language for the representation of IT specific case studies by Stübing (2003). His work allows to reference available learning resources. However, it still structures the content hierarchically. We have assumed different ideas from both approaches and have taken on the structural composition of presentation elements, such as tables and lists, as they are specified in eLML.

CaseML

The development process of CaseML not only comprised an initial analysis of requirements and their potential conformance in related research, but also necessitated a thorough review of implementation possibilities. In the following, we will introduce CaseML with a special focus on the realization of the requirements listed above. Subsequently, we will go into detail about the transformation possibilities and their implementation, before recapitulating the benefits of CaseML in the whole.

Requirements and their Realization in CaseML

Derived from the general requirements on the online presentation of our case studies, we will now describe their realization in CaseML.

1. Harvard Business Case: In order to develop field-based and decision-oriented cases, we followed the approach proposed by Leenders et al. (2001). They recommend following framework, which guaranteed a focusing process: A case should always start with the introduction, followed by general company background, specific area of interest, specific problem or decision, alternatives and end with the conclusion. Furthermore, the case is complete with the exhibits, providing raw materials for learners to develop analytical skills, the appendixes, which are complementary or useful materials, a glossary as well as endnotes. CaseML comprises this framework and is therefore structured as shown in Figure 1.

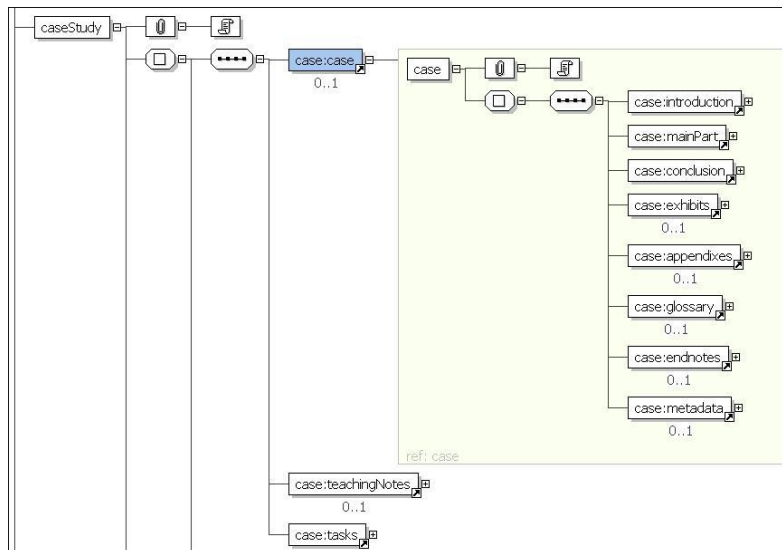


Figure 1: Cut-out from the graphical representation of the XML scheme showing the upper levels of CaseML

Directly derived from the root element *caseStudy*, the scheme is divided into the elements *case*, representing the text of the case and realizing the framework, *teachingNotes*, containing information for the teacher to prepare for class, and *task* holding several assignments for learners. The element *case* consists of a sequence of the elements *introduction*, *mainPart*, *conclusion*, *exhibits*, *appendixes*, *glossary*, *endnotes* and *metadata*. The element *introduction* presents a summary of the issue, identifies the name and location of the organization, and provides, for example, information about the decision maker, the problem, or the timeframe. To capture all this information, CaseML provides optional elements, which help the case's author to incorporate this information. Example 1 shows how an introduction in CaseML could be written. The main part of the case is mapped through the element *mainPart*, which contains elements like *companyBackground*, *specificArea*, *specificProblem and alternatives*, which cover information on and a description of the organization's background and its industry, the area in which the key decision maker works, the specific problem or decision, and possible alternatives. Additionally, the scheme provides other semantic elements such as *organizationHistory*, *industry*, *majorProduct*, *staffing*, *requirement*, etc. The element *conclusion* takes the reader back to the introduction and reinforces time limits that are set by the case. Therefore, elements like *deadline*, *nextMeeting* or *customerExpectations* can be used within this element.

2. Different views for teachers and learners: The two separate views for teachers and learners have been implemented in the upper level of the CaseML hierarchy. Directly derived from the root element *caseStudy*, the scheme is divided in *case*, *teachingNotes* and *tasks*, as shown in Figure 1. The first level, therefore, represents the learner's and the teacher's perspective: The content of the element *case* is visible to both, while the optional element *teachingNotes* contains information for teachers only.

3. Support of different assignment: A further requirement is to reveal suitable content for a certain assignment. In this regard, the modularization of the case study's content provides the scope for additional customizations. As previously described, each module has been allocated an element *task* and an element *content*. The task is defined to be of type integer and has an unbounded occurrence. However, the specification requires the indication of at least one assignment to each module. An example module in the XML file could therefore be composed as in Example 1.

```

<introduction title="Introduction">
  <moduleIntro>
    <taskID>1</taskID>
    <taskID>2</taskID>
    <contentIntro>
      <paragraphIntro>
        Since <time>1999</time> the management of the
        <organization>XYZ transport company</organization> has pursued
        <aim>to prepare a competitive urban company for the
        European market. </aim>
        <problem>Thereby, they detected an increased need for
        communication among the employees, and between several
        business units. </problem>
      </paragraphIntro>
    </contentIntro>
  </moduleIntro>
</introduction>

```

Example 1: An extract of a CaseML XML document

According to this extract, the module's content (and subordinated element definitions, such as *paragraph*) appertains to Task 2 and 3. The interconnection to the related assignment, as they are defined in the element *tasks*, has been established by the definition of attributes. Correspondingly, *tasks* contains the attribute *title* and *taskID*. The actual linking of a task to a relevant module is thereby achieved by the latter attribute *taskID*, which, accordingly, is also of type *integer*. With the allocation of one or more tasks to each module defined, we have ensured the possibility to adapt the presentation of the case study for the learner to the amount of modules needed to cope with an assignment. The modular structure additionally allows for an adaptation of the learning scenario through a reduction of the case study's extent.

4. Flexible teaching and learning scenarios: The adaptation of case studies to the favored work duration within a teaching and learning scenario has been enabled by a modularization of the content. Text paragraphs and multimedia, potentially comprised in lists, tables or other content versions, are included in the element *module*, which requires the specification of the belonging assignment(s) and the content. The length of the case study is thereby controlled by the teacher's choice of assignments, which triggers the presentation of belonging modules. Furthermore, CaseML provides the option to specify semantic elements within the element *teachingNotes*. For example, suggestions for the use of the case study with regard to different teaching and learning scenarios can be made in the semantic element *timePlan*. As such, both the author, who defines the information for the element

teachingNotes, and the teacher are given a wide range of possibilities for determining the scope of the case study.

5. Increasing realism through the use of multimedia: As described before, one of our aims in this project is the development of web-based case studies, which are enriched with multimedia materials. With the element *multimedia*, CaseML provides the possibility to easily integrate video clips, flash animations, java applets, and images. Furthermore, it is possible to integrate XML standards like mathML, SVG, or x3d using this element.

6. Different output formats: Each case study is stored in a single XML file and can be transformed into different output formats. Currently, we have implemented style sheets using XSLT for transforming the case studies in HTML for an online version and in LaTeX for a print version. The style sheet for the online transformation produces XHTML code. To follow the paradigm of separating content and presentation, all layout and style definitions are defined in a separate cascading style sheet (CSS) file. Therefore, the style of the generated HTML pages can be changed easily by adapting the corresponding definitions in the CSS file. The style sheet for the print version produces LaTeX code, which can be used to generate a PDF document, with the help of the LaTeX engine. Our project partners can use the generated LaTeX file to easily perform individual changes on the case, such as writing the name of the course and teacher on the title page before generating a PDF document.

7. Platform independence: In the requirements section we pointed out the importance of being able to use the case studies in different LMS. Nowadays, most LMS support standards like IMS Content Packaging (IMS CP, 2004) and Sharable Content Object Reference Model (SCORM, 2004) to import online class material into the learning platform. We, therefore, implemented packaging scripts that transform CaseML case studies to online versions of either standard.

CaseML in Practice

The implementation of requirements in CaseML, as described in the previous section, allows for the generation of a case study from the text version to the point of using it online. We have tested this process with four case studies that have been written within our project. It was notable, that different authors did follow the structure of a Harvard business case, but nevertheless strongly adapted this structure to suit a desired learning and teaching scenario. The resulting case studies showed strongly differing structures: A case study about the introduction of an intranet at the public transportation services in Leipzig, Germany, for instance, leads the learner from the problem to the recommendation of different solutions before defining the implementation of a chosen strategy. Thus, the case study not only introduces the problem but also anticipates the solution as implemented in real life. Contrariwise, a second case study about the migration of a securities clearing system at the Zurich Cantonal Bank in Switzerland only introduces the problem but leaves the choice of an appropriate solution to the learner. Our experience with CaseML showed that these differences can be easily integrated. The four case studies are used in various scenarios, whereas even a single case study can be used within a shorter or longer time frame with the help of the modularization possibility in CaseML. Specifying the affiliation to a certain assignment for each module, the extent of the case study can be adequately adapted to one's own needs. We also used semantic information to annotate text paragraphs and multimedia in our case studies in order to allow for the search of certain criteria. Subsequently, the XML files were automatically transformed into two output formats, LaTeX and HTML, according to the selected style sheet. The latter further enabled an automated content packaging for the use online: Both a version for the teacher and one for the student are generated for each of the previously defined assignments. These content packages were then manually integrated into OLAT, a learning platform developed at the University of Zurich.

The Benefits of CaseML at a Glance

CaseML distinguishes itself from previously implemented markup languages through the possibility of flexible content *modularization*. With the allocation of modules to specific assignments (see requirement no 3), the case study can be customized to present only relevant parts to the learner. During the process of mapping modules of the case study to designated assignments, it is thus possible to *reduce the case study's extent*. Teachers can thereby avail themselves of *various teaching scenarios* and are able to *adapt the level of difficulty* to learners by either providing relevant information only, or forcing learners to decide for significant information themselves. In this respect, one and the same case study can also be easily adapted for use within a single class as well as for the more continuous use over the period of four weeks. Suggestions for the use of the case study in different teaching and learning scenarios are given in the form of *teaching notes* that build an associated unit to the case's content, but are only accessible for teachers. An auxiliary advantage of the use of CaseML is the *domain*

knowledge that allows for functional extensions, such as the readout of semantic information. With this, it is not only possible to search for all occurrences of a certain detail within the case, but the facultative semantics also yield an indication for the case study's content. Altogether, the author is highly flexible in the preparation of the content for various requirements and scenarios, but simultaneously, CaseML provides guidance to the crude structure of a business case study conforming with the Harvard standard. From a technical point of view, CaseML has further advantages for the modifiable use of case studies, as it can be transformed into *different output formats*, such as HTML, LaTeX, and PDF. Additionally, the markup language enables the use in any LMS supporting content packages. Due to this platform independence, it has, therefore, potential to be deployed at numerous schools and universities, allowing teachers and learners to stick to their familiar learning environment.

Conclusion and Future Work

After an introduction to our e-learning project CasIS, we presented our essential requirements for the use of web-based case studies. We found that the best way to fulfill all requirements is to use XML. Therefore, we investigated existing XML developments in the area of e-learning and showed their limitations: a strongly predetermined structure and insufficient adaptability to provide field-based case studies that can be used in flexible teaching and learning scenarios. Furthermore, we presented the markup language CaseML, which we developed based on our requirements. Unique features of CaseML are the *modularization* possibility of content for the support of different assignments using one single case, the capability to use *semantic elements* to mark up the domain knowledge, and the ability to transform case studies written with CaseML into *different output formats* for the use in various learning management systems. The advantage of using domain knowledge is, at the same time, an inevitable constraint of CaseML, as the focus lies on business case studies. However, since the semantic elements in CaseML are optional, it can still be used for case studies in other fields by simply using a subset of these semantic elements, leaving out the business specific ones. In the future, we will implement the functionality to read out semantic information. One possible scenario for this is that learners can search for all occurrences of a certain detail within the case such as to get information about the business units or the strategic goals of the company. As a further extension of CaseML, it is planned to describe methods and tools that are contained in the CasIS toolbox.

References

Bleisch, S., Fisler, J., 2005. eLesson Markup Language eLML – eine XML basierte Applikation für die beschreibende Auszeichnung von nachhaltigen und flexiblen e-Learning Inhalten. In: Muttentz, Switzerland: Fachhochschule beider Basel (FHBB).

IMS CP (2004).IMS Content Packaging Specification. <http://www.imsglobal.org/content/packaging/>

Leenders, M. R., Mauffette-Leenders, L. A., Erskine, J. A., (2001). Writing cases. Western Ontario, Ivey.

Niederhuber, M., Heinimann, H., Hebel, B., (2005). e-Learning basierte Fallstudien zur akademischen Ausbildung in der Geoinformatik: Methodisches Konzept, Umsetzung und Erfahrungen, Proceedings of DeLFI 2005: 3. Deutsche e-Learning Fachtagung Informatik, Rostock.

Roberts, M.J. (2001). Developing a Teaching Case (Abridged). Harvard Business School Case No. 9-901-055.

SCORM (2004). Sharable Content Object Reference Model. <http://www.adlnet.gov/scorm/index.cfm>

Stübing, M., 2003. Metadaten-gestützte Integration von Fallstudien in IT-gestützte Lehre. Diploma thesis, Technical University of Berlin.

Acknowledgements

The project has been funded by the Swiss Virtual Campus (P-4-011).