

Reuse of Teaching Materials in Targeteam

Gunnar Teege

Universität der Bundeswehr München

Key words: *Reuse, Adaptation, Learning Materials, XML, XSLT.*

Abstract:

Electronic representation of teaching materials enables their reuse for a more efficient development of courses. Most authors consider reuse as a matter of metadata, which describes entities of teaching materials in a way that authors can search and assess the entities before adding them to a course. In this paper we identify two additional requirements for reusable entities: Abstractness and adaptability. We explain, how these requirements are implemented in the Targeteam system, using XML for the representation of abstract entities and using XSLT for their adaptation by transformations.

1 Introduction

XML has become a universal language frame for representing data. One of its strengths is the support for both structured and semi-structured data, hence it is widely applied to many different kinds of data.

This is true also for the specific domain of education. At least four main kinds of data are considered here, and there are several efforts for developing XML representations for them. One category of data consists of the content, which should be learned (in the case of fact learning) or used for learning (in the case of skill learning). A second category consists of *descriptions* of such content, used for identifying learning modules for a specific purpose. This category is usually called “learning object metadata” [4]. A third category of data consists of descriptions of the learning *process*, of the strategies used, of the kind of interaction between teacher (if involved) and learner. Languages for describing this process are called “educational modelling languages” [3]. And a fourth category consists of the organisational data describing a specific learning group, including descriptions of the group members, of their current knowledge status, their learning history and so on. This is the kind of data which is typically managed by a “learning platform”.

In this paper we present the approach of the Targeteam (“Targeted Reuse and Generation of TEaching Materials”) system [5]. This system supports an XML based representation of the first data category: the content. For its use during the learning process it is complemented by systems supporting the management of the other three data categories. However, Targeteam mainly addresses the processes of authoring and of course design. In these processes, educational modelling languages and learning platforms need not be used.

2 Educational Content

Targeteam has been developed with the objective of managing large collections of facts and other contents which have to be studied by learners. These materials include pedagogical annotations, such as explanations, motivations, examples, and they are carefully structured and interrelated.

Targeteam is based on the assumption that this structure is crucial for providing an understanding for the learners. A simple linear sequence of standalone “learning modules” is not sufficient. Compare this with a good teaching book which has been carefully structured to be useful for learning, and contrast it with a collection of articles about a common subject, which does not provide an overview or relations among different viewpoints.

Of course, a collection of articles can be useful, if it is complemented by a corresponding learning strategy, e.g., a moderated discussion in a learning group. However, Targeteam has been designed to cover the full spectrum of learning materials, from unstructured fragments up to highly structured content collections.

There are several reasons for an electronic representation of educational content. One reason is a better support for the authoring process, as it is the case in general publication. A second reason is the possibility of using information technology for content delivery during the learning process. This includes all forms of remote learning using a computer network. A third reason is the support for hypermedia: multimedia content structured and connected by hyperlinks. A fourth reason is reuse: the electronic representation allows reusing the same content in different learning contexts with minor costs.

3 Content Reuse

While Targeteam addresses all four objectives, its main focus is on support for reuse. Reuse means, for example, to use materials about a specific subject in an introductory course, in a special course, and in a practical exercise. This improves the effectiveness of content development, however, not automatically also the quality.

Often, reusability is seen as a matter of metadata for “learning objects” [2]. However, even with good metadata available, reuse works in practice only for entities of rather low granularity, such as images or small case descriptions. As a basis for extending reuse to larger entities, Targeteam identifies two additional requirements for reusability: abstractness and adaptability.

3.1 Abstractness

In many cases, content, which has been specifically designed for one context cannot reasonably be used in another context. Hence, for being reusable, the content must be independent of a specific context, it must be *abstract*. Targeteam content is abstract in several ways.

Targeteam abstracts content as much as possible from the way it is presented. This is the usual issue of separating layout from content. Targeteam text does not support any graphical markup such as font sizes, styles or line width.

Non-text media objects, on the other hand, typically include a lot of graphical layout information. Consider an image depicting a structural schema of some kind. However, media objects in Targeteam have a small granularity, hence the need for abstractness as a prerequisite for reuse is not high. Targeteam uses media objects as atomic “islands of layout” where specific presentation aspects can be used to help represent the content in an illustrative way.

Targeteam also provides an abstract *structure* in the form of a homogeneous hierarchy of issues. Content chunks are not identified as “chapter”, “section”, “paragraph”, “list entry”, “exercise”, “slide”, “course unit” etc. Instead, the actual structure is generated automatically from the homogeneous hierarchy when the delivery format is produced.

The abstract structure makes it possible, to reuse Targeteam content chunks in differently structured contexts. The same issue can be used in one context as a subsection where the subissues become list entries, and in another context as a separate chapter where the subissues become sections. This is most useful in combination with adaptation, as described below.

Finally, Targeteam content abstracts from the way it is used in the learning process. The same content, say, a case description, may be used in problem based learning or may be presented by a teacher.

Of course, not all content is useful for all learning methods. The issue here is, however, that in Targeteam the learning method is not explicitly prescribed for the content. It is up to the educational designer, or even to the learner, which method to use for learning the content.

3.2 Adaptation

Abstract content can be moved easily between different contexts. However, it usually cannot be used there as it is. It needs to be *adapted* to the context. It should be clear, that the same content module normally cannot be used in an introductory course, in a special course, and in a practical exercise, without changes.

Hence, in Targeteam adaptation is the second important concept for content reuse. Adaptation does not happen on the level of layout, adaptation means modifications on the content level. For example, a content chunk may be adapted by omitting parts, by adding a part which explains its relation to the context in which it is reused, by replacing its title or by replacing a term by a synonymous term which is used in the context.

In particular, content adaptation in Targeteam is not restricted to some predefined selections or parameter settings. Instead, the content can arbitrarily be modified down to the character level. Hence it is not the duty of a content author, to decide how to make her content adaptable and reusable for others. The person who reuses the content has full control over the adaptation process.

Content adaptation in Targeteam does not mean to copy and modify the content. Instead, adaptation means to specify a *transformation* which is applied to the content when it is used in the new context. This has two important consequences: the author retains control over the original content and the reusing author profits if the author updates or improves the original content.

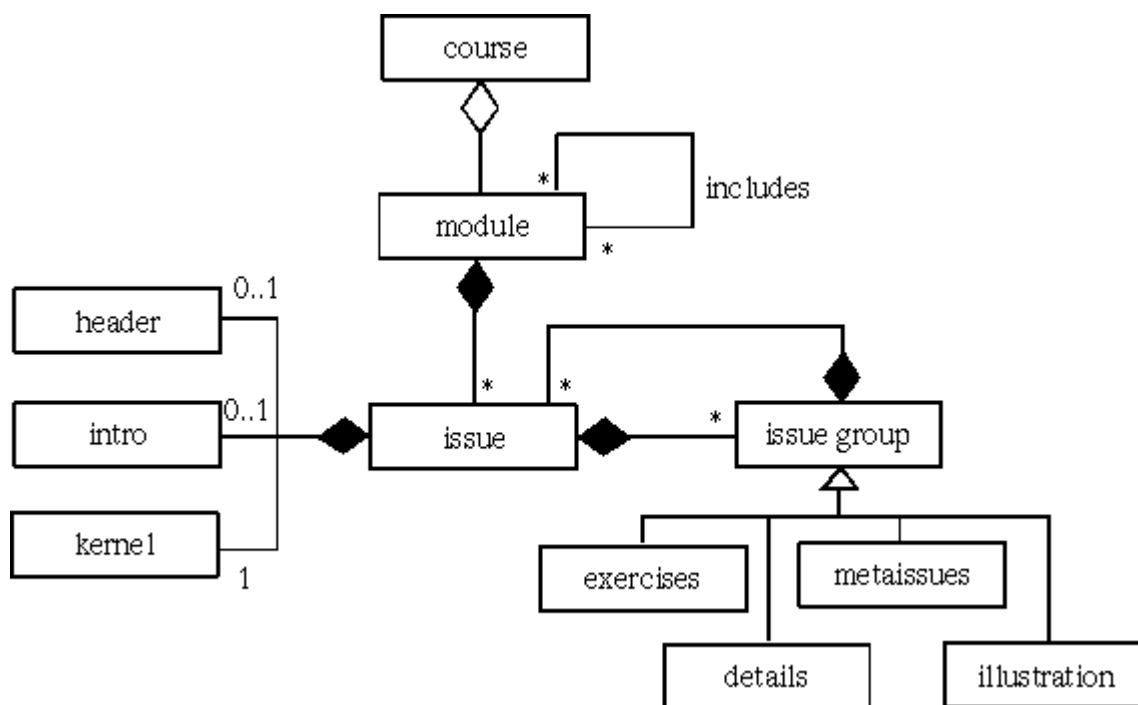


Figure 1 : UML diagram of the main structure in TeachML

4 Implementing Content Reuse in Targeteam

The approach for content reuse, as described in the previous section, has been implemented in Targeteam on the basis of XML. Abstractness is achieved by employing an appropriate XML Schema, which is called TeachML in Targeteam. Adaptation is achieved by leveraging the XML transformation language XSLT [1].

4.1 TeachML

TeachML is a deliberately simple language. Its main elements provide the abstract structuring as “issues”. An issue consists of an optional “header” an introduction, a “kernel”, and arbitrary many subissue groups. The kernel directly contains text or other media objects. The subissue groups are classified according to their role in the issue. Figure 1 depicts a simplified UML diagram of TeachML.

There are some markup elements defined for specifying, e.g., emphasized or quoted text, however, all these elements provide equivalents to “inline level” formatting. With one exception, used for formatting program fragments, there are no elements for “block level” formatting, such as lists or tables. The objective is that the issue structure should always be used instead. The content of the kernel should be no more than a usual text paragraph. Thus, authors are motivated to produce a relatively fine grained issue hierarchy.

As a consequence, in many cases adaptation can be done on the issue level, by transforming and reordering the issues in the hierarchy. The fine grained structure also helps when specifying the place where an adaptation should be applied.

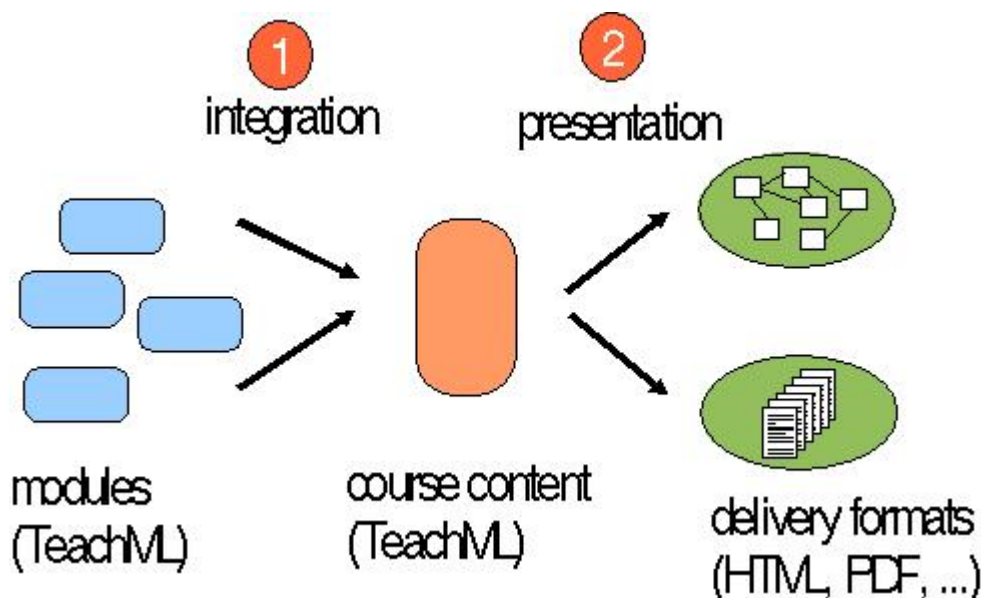


Figure 2 : Adaptation (1) and delivery format generation (2)

After adapting a content for a specific purpose (e.g., a specific course), several delivery formats can be produced (see Figure 2, step 2). During this step the abstract structure is made concrete. What structure is generated in detail, depends on the delivery format. When producing HTML pages for online use, a set of linked pages is generated, where each page has a relatively shallow structure. When producing a printed script, a classical structure with chapters and sections is generated, where issues at deeper levels become list entries or paragraphs.

In Targeteam, this processing step is implemented by a Java program. Currently, it can produce two different delivery formats: HTML and LaTeX. Both cases are highly configurable. On this basis it is possible to produce high quality output even from the abstractly structured content.

4.2 Transformation with XSLT

Adaptation in TeachML is done by specifying transformations using XSLT. Unlike in many other applications of XSLT, the goal is not to present the content by transforming it to another language, such as HTML, SVG, or FO. Instead, the transformation again produces TeachML. The modifications only affect the content, in ways described above. This is shown as step 1 in Figure 2.

The use of transformations has an important advantage in contrast to the manual modification of a copy, as it is often done when adapting content for learning. The transformation can be applied anew, whenever the original content has been changed or improved. Additionally, it is to some degree possible to combine several transformations. Together, this results in a platform for the distributed management of different adapted variants of shared contents for learning.

Targeteam organizes the content in modules (compare Figure 1), every module is a separate TeachML document. The size of the module has no influence on its reusability, since adaptation supports the extraction of arbitrary parts of it for reuse. Usually, a module will be a unit covering a certain subject.

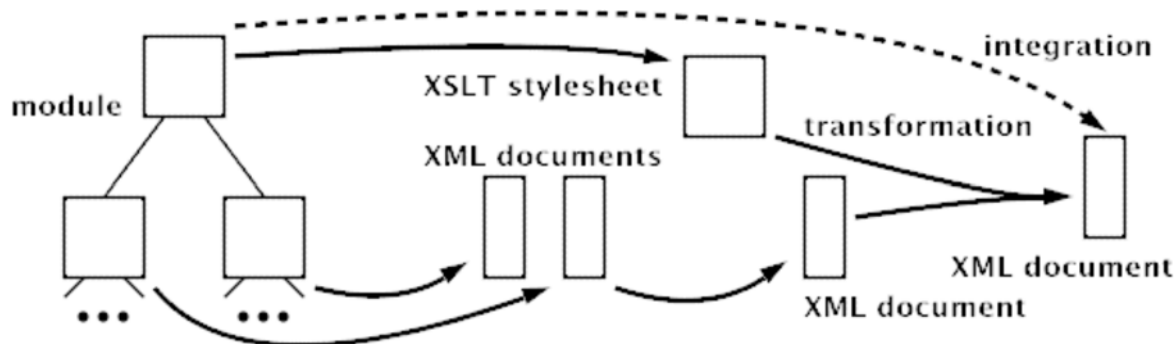


Figure 3 : The integration step in Targeteam

A set of modules is collected to form a "pool". The modules in a pool are organized into a hierarchy which is a directed acyclic graph. If a module A in this hierarchy is subordinate to a module B, this means that module B contains, among its own content, also (adapted) content from module A. The transformations of parts of A and the places where they are inserted in module B, are specified as a part of module B. In this way, the size and complexity of the overall content of a module increases according to the position in the hierarchy. Modules on the highest level usually correspond to complete course contents covering several months in the learning process.

When an author wants to create a new content, she adds a new module to the pool. In the module she can formulate new content and mix it with adapted content from other modules, by subordinating existing modules and specifying transformations for them. The Targeteam system will perform the step of integration, where all transformations are executed and all content from subordinate modules is collected. The result of this integration step is a single XML document, (as shown in Figure 2), from which the delivery formats will be produced. Whenever one of the modules changes, the integration step can be repeated.

Technically, the integration step is implemented as follows (see Figure 3): First, the module is syntactically rewritten into an XSLT stylesheet. Next, for all subordinate modules the integration step is recursively performed and the resulting documents are combined into one single XML document. Now, the stylesheet is applied to this document, selecting and adapting content from it and mixing it with the content from the module itself. The resulting XML document is the result of the integration step.

```
<teachml>
  <module>
    <issue>
      <header>Transformation with XSLT</header>
      <kernel>Adaptation in TeachML is done by ...</kernel>
      <details>
```

```

        <xsl:apply-templates select="/children/module[@id='submodule1']//
        issue[contains(header,'TeachML')]" />
    </details>
</issue>
</module>
<xsl:template match="issue[@id='theorem1']/kernel">
    <kernel>In this context the following statement is of special interest.
    <xsl:apply-templates />
    </kernel>
</xsl:template>
</teachml>

```

Figure 4 : Example TeachML module

The example in Figure 4 shows a minimal TeachML module which includes content from another module and modifies it with the help of an XSLT template. The example also gives an impression of how the XSLT language has been integrated into the TeachML language.

5 Experiences

Targeteam has been used in university level teaching since its development in 1999. Currently, it is used by several course developers at two universities. Materials have been developed in the domain of computer science which cover about 15 half year courses. Several content is used for more than one course, employing the reuse support of Targeteam. As an example, the half-year course “Informatik für Ingenieure” (Computer Science for Engineers), prepared at Technische Universität München, has been extended to a full-year course at the Universität der Bundeswehr München, by integrating and adapting materials from several non-introductory courses such as “Hypermedia”, “Computer Networks”, and “Distributed Systems”.

The experiences show, that transformation based adaptation works as an important support for reuse of teaching materials. It has been demonstrated, that XSLT transformation is in principle adequate for specifying adaptations for abstract materials. However, for a broader use, corresponding tools are necessary, which help to specify adaptations in an intuitive and easy way, using XSLT only behind the scenes. The development of this kind of tools will be an important part of the future development of Targeteam.

References:

- [1] James Clark : Xsl transformations (xslt) version 1.0. W3c recommendation, World Wide Web Consortium, 1999. Available at <http://www.w3.org/TR/xslt/>
- [2] Stephen Downes: Learning Objects: Resources for Distance Education Worldwide. International Review of Research in Open and Distance Learning, 2(1), 2001. Available at <http://www.irrodl.org/content/v2.1/downes.html>
- [3] Rob Koper: Modeling units of study from a pedagogical perspective: the pedagogical meta-model behind EML. discussion paper, Educational Technology Expertise Center, Open University of the Netherlands, 2001. Available at <http://eml.ou.nl/introduction/articles.htm>.
- [4] IEEE LTSC: IEEE P1484.12 Learning Object Metadata Working Group Homepage. <http://ltsc.ieee.org/wg12/index.html>, 2002.
- [5] Gunnar Teege: Targeteam Project Homepage. <http://www.targeteam.org/>, 2002.

Author:

Gunnar Teege, Prof. Dr.
 Universität der Bundeswehr München, Fakultät für Informatik
 Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany
Gunnar.Teege@unibw-muenchen.de