# Exploiting Author-Designed Domain-Specific Descriptive Markup Languages in the Production of Learning Content

José L. Sierra[*], Alfredo Fernández-Valmayor[*], Mercedes Guinea[#]
(*) *Dpto. Sistemas Informáticos y Programación. Fac. Informática. Universidad Complutense de Madrid. 28040, Madrid (Spain). {jlsierra,alfredo}@sip.ucm.es*
**(#)** *Dpto. Historia de América II. Universidad Complutense de Madrid. 28040, Madrid (Spain). guinea@ghis.ucm.es*

## Abstract

*In this paper we describe an approach to the production of learning resources where authors (students and instructors) are actively involved in the production process. This active involvement is achieved by using descriptive markup technologies. Authors are compelled to produce learning resources in the form of documents, and to make the structure of these documents explicit by creating and using descriptive markup languages. This lets developers formalize these author-designed markup languages and provide suitable transformation specifications for translating these marked documents into their final presentations. We exemplify this approach with the production of DHTML pages by Ph.D. students in archaeology, oriented to be used as resources owned by reusable learning objects in* Chasqui, *an authoring and deployment tool used in the virtualization of academic museums at the* Complutense *University of Madrid (Spain).*

***Keywords***: *Authoring Approach, Descriptive Markup Languages, Reusable Learning Objects, Active Learning, Virtual Museums*

## 1. Introduction

During the last ten years we have been involved in the use of descriptive markup technologies in improving the production and maintenance of educational [4] and other content-intensive applications [10]. In these experiences we have realized that digital learning resources structured according to descriptive markup languages are able to survive changes and trends in the computer industry, that they can be adapted to different learning and sociocultural circumstances, and that they are easy to maintain, regardless of platforms and tools. All these features are also very valuable in the context of life-long learning scenarios.

The typical production models followed in our experiences begin with the establishment of a suitable domain-specific descriptive markup language, which is subsequently used to make the logical structure of the documents that constitute the core of the learning resources explicit. The formulation of this markup language involves both authors and computer science professionals, and it shares many features with domain analysis techniques, proposed in the field of software reuse [1], and thoroughly used in the context of software development based on domain-specific languages [12].

As stated in [12], domain analysis demands a considerable amount of time and resources to be completed successfully, even with the use of iterative and incremental approaches. To overcome this problem, in [7] an author-centred approach is proposed. This approach, which is called *infocentrism*, delegates the formulation of domain-specific languages (*jargons*, in terms of [7]), and even the construction of their processors, to authors. Although at first glance this approach can appear a bit ambitious and unrealistic, since it presupposes that there are authors with good skills in computer science (indeed, authors in [7] were experts in software configuration, and therefore they were endowed with good technical abilities), we have realized that a lightweight, document-oriented version can also be used successfully in the production of learning resources in specialized, non-technical domains. In this paper we describe and exemplify this approach.

This paper is organized as follows. In section 2 we describe our approach from a normative point of view. In section 3 we exemplify this in the context of the *Chasqui* [8][11] web-based authoring and deployment tool for the virtualization of the *Antonio Ballesteros* Museum of Archaeology at the *Complutense* University of Madrid (Spain). Finally, in section 4 we outline the conclusions and the lines of future work.

## 2. The Production Approach

The approach presented in this paper is the result of adapting our domain-analysis driven process models for the document-oriented development of content-intensive applications [10] to the more pragmatic author-centred point of view sustained by the *infocentric* posture [7]. The resulting approach can be used for the production of learning resources in domains where their contents can be organized as documents with a clear logical structure. The following subsections detail the different aspects of the approach.

### 2.1. Products and activities

The production approach contemplates the products and activities depicted in Fig 1.

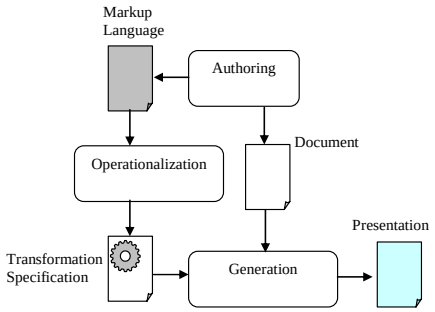


**Fig 1. Products and activities in the production approach**

The central activity of the approach is *authoring*. During this activity, a *document* with the learning contents is produced, and the structure of this document is characterized in terms of descriptive markup. As a consequence, a *markup language* is also produced and/or refined, and it is used to mark the document up. Here it is fundamental to remark that this markup language is not established *a priori*, but is created and modified as the result of this authoring activity. Therefore, the need for a domain analysis-like activity is avoided. Markup can follow the conventions dictated by a standard markup metalanguage (e.g. SGML [5] or XML [2]), so that all the companion publishing technologies can be used to produce the presentations required for the contents.

The *operationalization* activity is devoted to providing a *transformation specification* that describes how to translate the marked document into a final presentation that can be deployed in a learning platform. As suggested before, this activity can take advantage of pre-existing transformation and stylesheet technologies associated with the standard markup metalanguages (e.g. DSSSL [6] for SGML, XSLT [3] for XML).

Finally, during the *generation* activity the final *presentation* for the contents is produced by processing the marked document according to the transformation specification. Usually this activity will be carried out automatically, by using a suitable transformation engine.

### 2.2. Sequencing of the activities

The activities introduced in the previous subsection are sequenced as stated in Fig 2, where a typical authoring – production cycle, with occasionally interleaved operationalization steps, is suggested.

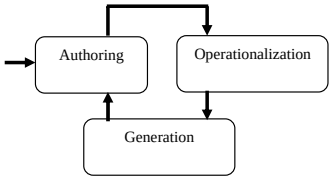


**Fig 2. Sequencing of the activities**

The iterative behaviour induced by the sequencing in Fig 2 concerns the iterative evolution and refinement of the different products involved in the process. In each iteration, the corresponding document is refined. This refinement can affect both its contents and its structure. In this case, the markup language created is also refined, and this refinement is also mirrored in the refinement of the associated transformation specification. When the authoring of a document of a similar type is faced, it will be possible to use and further refine the markup language already available and the corresponding transformation specification. Therefore, with the successive applications of the process in the same domain, a more mature markup language and a more refined transformation specification will be obtained. If the process is adequately driven, it will converge in a complete descriptive markup language and a complete specification of (possible) operational semantics for such a language.

### 2.3. Responsibilities

One of the main weaknesses of the infocentric approach defended in [7], when translated to domains other than software configuration, is the attribution to authors of language definition and operationalization responsibilities. In our production approach we relax this attribution, by explicitly introducing a community of *developers* which also actively participate in the

authoring activity. The resulting responsibilities for authors and developers are sketched in Fig 3.

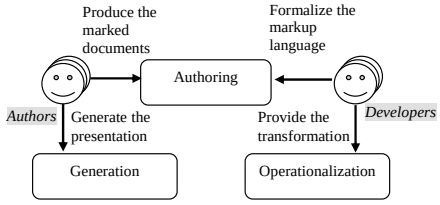


**Fig 3. Participants and their responsibilities**

As stated in Fig 3, in this production approach the leading role corresponds to authors. Authors produce the documents during the authoring activity, decide their structure, and make this structure explicit using tags with suitable identifiers and attributes. This is feasible due to the simplicity and understandability of the main concepts behind descriptive markup. In this process, they can follow the markup rules already established in the current version of the markup language, but they also can include new markup conventions, and even modify the existing ones. Developers analyze the changes and formalize them in the grammar for the markup language. In addition, developers also modify and/or extend the transformation specification, with the help of authors regarding presentation, interaction and other stylistic aspects. The resulting transformation is then used by the authors to produce the presentation.

## 3. Production of Learning Contents in *Chasqui*

In this section we exemplify the previously described approach in the context of the *Chasqui* web-based application used in the virtualization of the archaeological and ethnographical material maintained in the *Antonio Ballesteros* Musem, at the Department of *American History II* at the *Complutense* University of Madrid[1]. This application has been developed by our research group, an interdisciplinary group formed by both archaeologists and computer scientists.

As described in [8][11], *Chasqui* enables the authoring and deployment of reusable learning objects [9], which in *Chasqui* are called *virtual objects* (VOs), due to their initial orientation towards organizing the different resources and data resulting from the virtualization of the real objects in the museum.

Beyond its use as a simple authoring and deployment tool, *Chasqui* also has been used in several pedagogical experiences at the *Complutense* University [11]. In particular, the tool has been used by Ph.D. students for the publication of their research results. In this activity we have applied the production process described in this paper to let students incorporate their research reports as resources of new VO in the system. Following subsections give the details.

### 3.1. Context of the Experience

The experience was carried out in the context of the Ph.D. course on *New Information Technologies in Andean Archeology*. Students enrolled in the Ph.D. course cited are actively engaged in the production of new learning contents for the *Chasqui* system. For this purpose, the instructor assigns them several research papers about a theme, which they must elaborate as part of their homework. As result of this elaboration, the learners must prepare summaries synthesizing the main ideas contained in the papers. Then they discuss their conclusions with the rest of the group (including the instructor). The results of this process are syntheses contrasted and agreed upon by the group, which constitute very interesting material with a high educational potential. Therefore students are compelled to integrate this material into *Chasqui*, as part of thematic VOs regarding the subjects initially proposed.

Nevertheless, students interested in a course on *Andean Archeology* are hardly required to be proficient in authoring the high-quality DHTML presentations desirable for the resources finally published. To overcome this barrier, the production approach presented in this paper was adopted. As developers we involved some advanced undergraduate and graduate students in computer science who collaborate in our research projects as junior programmers. Therefore, while the main technological goal of this experience was to test and refine our production approach, the experience was also very valuable from a pedagogical point of view, since it allowed us to promote active learning among both the PhD. students enrolled, and our students in computer science.

### 3.2. Authoring of the Reports

As part of the Ph.D. course that contextualizes this experience, students playing the role of authors were first exposed to general concepts about descriptive markup. Given the simplicity of these concepts, they were easily comprehended by the students. Once this knowledge was assimilated, students were compelled to analyze the structure of their reports, and also to

[1] The production version of *Chasqui* can be found at `http://macgalatea.sip.ucm.es/chasqui.html`

describe a markup language informally reflecting this structure. These informal descriptions were usually expressed by authors using tagged and annotated trees, and also textually described as an initial documentation.

```
<documento>
   <titulo>
     <titulo-principal>Bonampak
     </titulo-principal>
    </titulo>
    <indice>
       <item referencia="El-Paisaje">
         <texto-plano>El Paisaje.</texto-plano>
        </item>
       ..........
    </indice>
    <contenido>
       <seccion identificador="El-Paisaje">
         <titulo-seccion>El Paisaje.</titulo-seccion>
          <unidad-conceptual>
           <texto-plano>El sitio de</texto-plano>
           <frase-a-imagenbd ovpropietario="1628"
                             recurso="fotositio.jpg">
            Bonampak
            </frase-a-imagenbd>
            <texto-plano>
            se encuentra en el estado mexicano de
           Chiapas, en plena selva tropical, ..........
```

**Fig 4. Fragment of a report about the mural pictures in the *Maya* excavation of *Bonampak* produced by a Ph.D. student. The markup highlighted was introduced by the student to make reference to a resource of the Chasqui VO 1628**

```
........
<!-- Indice -->
<!ELEMENT indice (item+)>
<!--El indice puede contener varios items (Aspectos
    tecnicos, Aspectos culturales-->
<!ELEMENT item (texto-plano, subitem*)>
<!-- Item es lo que quieres que ponga cada indice; ej
    (Aspectos tecnicos).-->
<!-- Ademas puede tener un atributo (opcional) que seria
     un ancla a una parte determinada del documento.-->
<!-- Esta parte determinada debe ir identificada con otro
     atributo y con este mismo valor que el atributo
     referencia. -->
<!ATTLIST item
      referencia IDREF #REQUIRED
>
.......
```

**Fig 5. Fragment of an XML DTD for the markup language used on the document of Fig 4. Shadowed comments correspond to annotations of the authors. Text in cursive corresponds with the formalization introduced by the developers.**

Following the production approach, computer science students playing the role of developers were also involved in authoring, in order to formalize the markup languages created by the authors. In this process developers started from the informal descriptions, and then they completed this documentation with initial formalizations of the languages in terms of document grammars. For simplicity they used XML DTDs (Document Type Definitions) [2] as the grammatical formalism.

The next step was for authors, who marked their documents up using the markup vocabulary and the structural relations created by them. When students tried to make the structure of their reports explicit according to the structure of their markup languages almost a third of them discovered inconsistencies in the structure of these documents. In turn they used the formally expressed structure of the language as a guide to solving the inconsistencies. In some other cases, students discovered that the language did not have enough resources (tags, attributes and/or relations) to express the knowledge structures desired and they proceeded to discuss an extension of the languages.

As an example, Fig 4 shows a fragment of one of the marked documents containing a report about the mural pictures at *Bonampak*, a *Maya* archaeological site in *Chiapas*, *Mexico*. The simplicity and power of descriptive markup languages also allowed authors (Ph.D. students) to refer to existing resources and VOs in *Chasqui*. For this purpose, they were also allowed to extend the language creating the appropriate tags and attributes (e.g. markup highlighted in Fig 4). In turn, in Fig 5 a fragment of DTD for the markup language used on Fig 4 is shown. This fragment contains both the informal annotations made by the authors and the formal grammatical rules written by the developers using the tags created by authors.

## 3.3. Operationalization

Once the marked documents were available, computer science students (developers) wrote XSLT stylesheets to transform them into DHTML presentations. As an example, in Fig 6 a fragment of the stylesheet used with the document in Fig 4 is shown.

```
.............
<xsl:template match="indice">
   <xsl:for-each select="item">
     <LI>
       <A CLASS="interno" TARGET="_self">
          <xsl:variable name="existeRef">
             <xsl:value-of select="@referencia"/>
          </xsl:variable>
          <xsl:if test="($existeRef='')">
              <xsl:attribute name="href"/>
          </xsl:if>
          <xsl:if test="($existeRef!='')">
               <xsl:attribute name="href">#
                        <xsl:value-of select="@referencia"/>
                    </xsl:attribute>
          </xsl:if>
          <xsl:value-of select="texto-plano"/>
       </A>
      .........
</xsl:template>
.............
```

**Fig 6. Fragment of the XSLT stylesheet used to translate the document of Fig 4 into DHTML**

Notice that, while an average Ph.D. student in *Andean Archeology* (authors) can be able to understand and manage descriptive markup as that in Fig 4, she/he will hardly be able to understand and use a language like XSLT, as the example in Fig 6 makes evident. Therefore, in order to be useful in this domain, the infocentric thesis must be relaxed by introducing developers knowledgeable in this kind of technologies (students in computer science in our experience).

### 3.4. Generation

The XSLT stylesheets provided during operationalization by the students in computer science (developers) were used by the authors to produce DHTML presentations, which were subsequently added as resources to *Chasqui* VOs. In Fig 7 several snapshots regarding the thematic VO about the mural pictures at the *Bonampak* excavation are shown. These snapshots includes the DHTML presentation produced from the document corresponding to Fig 4 using the XSLT transformation partially sketched in Fig 6 (*Chasqui* VO number 1628).

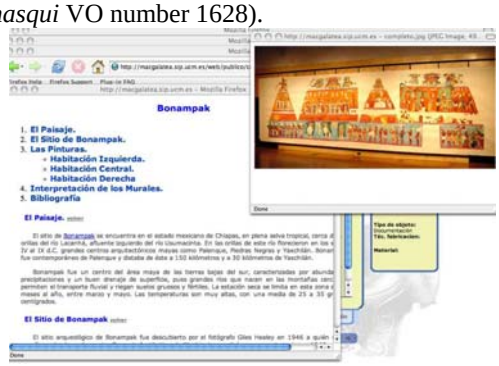


**Fig 7. Snapshots for the Chasqui VO number 1628 (mural pictures in *Bonampak*).**

Notice that, once the stylesheet were available, authors were able to iterate several authoring – generation steps until the generated presentations were satisfactory, or until a dead end was reached. This last case was usually due either to a defect in the stylesheet or a flaw in the document markup. In both cases developers were warned in order to adapt the stylesheet, and, in the second case, also to extend and/or modify the markup language.

## 4. Conclusions and Future Work

In this paper we have described an approach to the production of learning resources based on the creation of documents describing all the contents of the resource, and on the creation by the authors of markup languages for explicitly structuring these documents. The approach also contemplates the participation of developers for dealing with the more technical aspects of language formalization and operationalization. We also describe how this approach has been used in a real experience at *Complutense* University of Madrid, and how this experience has also promoted active learning among interdisciplinary groups of students.

As future work we want to experiment with the approach in other domains and to further exploit its pedagogical aspects. Besides the problem of content generation, we also want to unify it with our previous work on document-oriented development of content intensive applications [10].

## 5. Acknowledgements

The Spanish Committee of Science and Technology has partially supported this work (projects TIC2002-04067-C03-02, TIN2004-08367-C02-02 and TIN2005-08788-C04-01). We also would like to thank our students who participated in the experience.

## 6. References

[1] Arango,G. Domain-Analysis: From Art Form to Engineering Discipline. *ACM SIGSOFT Notes 14(3)*.1989

[2] Bray. T. Paoli, J. Sperberg-McQueen, C.M. and E. Maler (eds). *Extensible Markup Language (XML) 1.0 (Third Edition)*. W3C Recommendation. www.w3.org. 2004.

[3] Clark, J (ed). *XSL Transformations (XSLT) Version 1.0*. W3C Recommendation. www.w3.org. 1999

[4] Fernández-Manjón, B; Fernández-Valmayor, A. Improving World Wide Web Educational Uses Promoting Hypertext and Standard General Markup Languages. *Education and Information Tech.*, 2(3), 193-206. 1997

[5] Goldfarb, C. F. *The SGML Handbook*. Oxford University Press, Oxford, UK. 1990

[6] International Standards Organization. *Document Style Semantics and Specification Language (DSSSL)*. ISO/IEC 10179. 1996

[7] Nakatani, L.H; Jones, M. Jargons and Infocentrism. *First ACM SIGPLAN Workshop on Domain-Specific Languages DSL'97*, Paris, France, 18 January. 1997

[8] Navarro, A; Sierra, J. L; Fernández-Valmayor, A; Hernanz, H. From Chasqui to Chasqui 2: An Evolution in the Conceptualization of Virtual Objects. *Journal of Universal Computer Science*, 11(9), 1518-1529. 2005

[9] Polsani, P. Use and Abuse of Reusable Learning Objects. *Journal of Digital Information* 3(4).2003

[10] Sierra, J. L; Fernández-Manjón, B; Fernández-Valmayor, A; Navarro, A. Document-Oriented Construction of Content-Intensive Applications. *Int. Journal of Software Eng. and Knowledge Eng.*., 15(6), 975-994. 2005

[11] Sierra, J. L; Fernández-Valmayor, A; Guinea, M; Hernanz, H; Navarro, A. Building Repositories of Learning Objects in Specialized Domains: The Chasqui Approach. *5th*

*IEEE International Conference on Advanced Learning Technologies ICALT'05*. Kaohsiung, Taiwan, 5-8 July. 2005

[12] Van Deursen, A; Klint, P; Visser, J. Domain-Specific Languages: An Annotated Bibliography. *ACM SIGPLAN Notices*, 35(6), 26-36. 2000