HANDLING MULTILINGUAL CONTENT IN DIGITAL MEDIA: A CRITICAL ANALYSIS

Samuel Cruz-Lara, Satyendra Kumar Gupta, Javier David Fernández García, Laurent Romary, Philippe Sébire, Françoise Prêteux, Titus Zaharia, Marius Preda, Son-Minh Tran

▶ **To cite this version:**

Samuel Cruz-Lara, Satyendra Kumar Gupta, Javier David Fernández García, Laurent Romary, Philippe Sébire, et al.. HANDLING MULTILINGUAL CONTENT IN DIGITAL MEDIA: A CRITICAL ANALYSIS. [Research Report] 2006, pp.60. inria-00001120

HAL Id: inria-00001120
https://inria.hal.science/inria-00001120

Submitted on 18 Feb 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HANDLING MULTILINGUAL CONTENT IN DIGITAL MEDIA: A CRITICAL ANALYSIS

WP 2 Progress Report
Activity 2.1
Jules Verne

Reporting period: 1 May 2004 – 31 October 2004

Project number: ITEA 02002
Edited by: Samuel CRUZ-LARA on 25th October 2004

Contributors:

- Samuel CRUZ-LARA (LORIA / INRIA Lorraine, France)
- Satyendra GUPTA (LORIA / INRIA Lorraine, France)
- Javier David FERNANDEZ GARCIA[1] (LORIA / INRIA Lorraine, France)
- Laurent ROMARY (LORIA / INRIA Lorraine, France)
- Philippe SEBIRE (LORIA / INRIA Lorraine, France)
- Françoise PRETEUX (INT, France)
- Titus ZAHARIA (INT, France)
- Marius PREDA (INT, France)
- Son-Minh TRAN (INT, France)

[1] ERASMUS student from the University of Valladolid, Spain.

HANDLING MULTILINGUAL CONTENT IN DIGITAL MEDIA: A CRITICAL ANALYSIS

SUMMARY

This document expresses and analyzes the need to define a generic method for representing multilingual information in multimedia data. It describes the basic requirements that would bear upon such representations and establishes the potential link with ISO committee TC 37/SC 4 (Language Resource Management) [1] and with XMT (eXtended MPEG-4 Textual format) [2].

SUPPORTING COMPANIES AND INSTITUTIONS

CEA (Commissariat à l'Energie Atomique, France), CRP-HT (Centre de Recherche Publique “Henri Tudor”, Luxembourg), INRIA (Institut National de Recherche en Informatique et Automatique, France), INT (Institut National des Télécommunications, France), Philips Digital Systems Lab. (The Netherlands), Thomson (France).

TABLE OF CONTENTS

1 INTRODUCTION *4*

2 STANDARDIZATION BACKGROUND *5*

3 DATA CATEGORIES *6*

3.1 Notation **6**

3.2 Definitions **6**

3.3 Data Category Registry **8**

3.4 Data Category Selection **9**

4 The Multi Lingual Information Framework (MLIF) *12*

4.1 Introduction to MLIF **12**

4.2 Hierarchical overview of MLIF **13**

4.3 Example: A MLIF document **15**

4.4 XML canonical representation of the generic model **16**

4.5 Tool specific inputs **19**

4.5.1 TBX 19

4.5.2 TMX 19

4.5.3 XLIFF 20

4.5.4 TIMED TEXT 20

4.5.5 I18n 20

4.5.6 MPEG-4/MPEG-7 21

5 IMPLEMENTING LANGUAGE TRANSLATION BY USING MLIF: THE ITEA "JULES VERNE" PROJECT EXPERIENCE *22*

5.1 The RAMO Model **22**

5.2 RAMO and MLIF **23**

5.3 SMIL and XMT **24**

5.4 The XMT-O localisation round trip **27**

5.5 Identifying monolingual content in XMT-O **29**

6 USING XSLT TO DEAL WITH DIFFERENT MULTILINGUAL CONTENT FORMATS. *32*

7 SOFTWARE APPLICATIONS *33*

8 CONCLUSION. *34*

9 REFERENCES AND BIBLIOGRAPHY. *35*

10 APENDIX *37*

10.1 The XMT_{french} document cited in 5.4 ..... **37**

10.2 The $XMT_{english}$ document cited in 5.4 ..... **40**

10.3 The XLIFF file used by the MPEG-4 Multimedia presentation developed by INT for the ITEA Symposium (Seville, Spain 2004) **43**

10.4 An XSLT style sheet transforming XMT-O documents into MLIF documents **53**

10.5 An XSLT style sheet transforming MLIF documents into XLIFF documents **54**

10.6 An XSLT style sheet transforming XLIFF documents into MLIF documents **57**

1 INTRODUCTION

Linguistic information plays an essential role in the management of multimedia information, as it bears most of the descriptive content associated with more visual information. Depending on the context, it may be seen as the primary content (text illustrated by pictures or videos), as documentary content for multimedia information, or as one among several possible information components in specific contexts such as interactive multimedia applications. Linguistic information can appear in various formats: spoken data in an audio or video sequence, implicit data appearing on an image (caption, tags, etc.) or textual information that may be further presented to the user graphically or via a text to speech processor.

In this context, dealing with multilingual information is crucial to adapting the content to specific user targets. It requires one to consider potential situations where the linguistic information contained in a multimedia sequence is either already conceived in such way that it can be adapted on the fly to the linguistic needs of user, or by using an additional process where content should be adapted before presenting it to the user.

Finally, there are a wide variety of applications within which multilingual information may appear, which supports development of a generic framework for dealing with multilingual content: subtitling of video content, dialogue prompts, menus in interactive TV, descriptive information for multimedia scenes, karaoke management, etc. Such information should be considered in the light of the experience of more specialized communities traditionally dealing with multilingual content, namely the translation and localization industry (see section 2).

Main requirements.

It is suggested that a generic mechanism for representing and dealing with multilingual content in multimedia information shall be designed within the MPEG-4 / MPEG-7 context. Such a mechanism should comprise a flexible specification platform for elementary multilingual units that may be either embedded in other types of multimedia content (e.g. as an extension to XMT) or used autonomously to localize existing content. Such a specification platform should be coupled to a reference set of descriptors that should be used to parameterize specific applications comprising multilingual content. Those descriptors may either be specific to the multimedia context (representational features, temporal synchronization, spatial insertion; e.g. as available in the various MPEG-7 DS) or dedicated to the linguistic management of the content proper (content segmentation, phonetic representation, links to relevant terminologies, etc.). It should be able to deal in a uniform way with both monolingual or multilingual embedded content and with various levels of granularity of the linguistic content (from large portions of texts down to phoneme level features (e.g. for synchronization purposes).

2 STANDARDIZATION BACKGROUND

The scope of research and development in localization and translation memory (TM) process development is very large. There are numerous independent groups LISA [3], OASIS [4], W3C [5], ISO [6] working on these aspects namely LISA, OASIS, W3C, ISO, etc. LISA is working for the GILT (Globalization, Internationalization, Localization and Translation) business community. It has evolved as the premiere organization for developing language-technology standards. OSCAR [7] "Open Standards for Container/Content Allowing Re-use" is LISA's special interest group (SIG) for the creation of open standards. OASIS evolved to drive the development, convergence, and adoption of structured information standards in the areas of e-business, web services, etc. OASIS is driven by various technical committees (TC) formed by its members. OASIS XML Localization Interchange File Format TC (OASIS XLIFF TC) was formed with the purpose to define, through XML vocabularies, an extensible specification for interchange of localization information. The World Wide Web Consortium (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to bring the Web to its fullest potential. W3C organizes the work necessary for the development or evolution of Web technologies into Activities. Most of the activities under W3C are developing specifications/tools for management of language/text on the web. ISO is a network of national standards institutes from 148 countries working in partnership with international organizations, governments, industry and business and consumer representatives. ISO is a bridge between public and private sectors. It has also formed some technical committees, working in language-technology and IT applications in information, documentation and publishing.

Under the guidance of above-mentioned groups, many formats have been developed. Some of the major formats of specific interest for localization and TM are TBX [8], TMX [9] (LISA/OSCAR), XLIFF [10] (OASIS), Timed Text [11] (W3C), i18n [12] (W3C), TMF [13] (ISO). TBX is an open XML-based standard format for terminological data. This capability will greatly facilitate the flow of terminological information throughout the information cycle both inside an organization and with outside service providers. TMX is a vendor-neutral, open standard for storing and exchanging translation memories created by Computer Aided Translation (CAT) and localization tools. The purpose of TMX is to allow easier exchange of translation memory data between tools and/or translation vendors with little or no loss of critical data during the process. The purpose of XLIFF vocabulary is to store localizable data and carry it from one step of the localization process to the other, while allowing interoperability between tools. The Timed-Text specification covers real time subtitling of foreign-language movies, captioning for people lacking audio devices or having hearing impairments, karaoke, scrolling news items or teleprompter applications. It provides interoperability between different existing formats. ISO 16642:2003 [14] specifies a framework designed to provide guidance on the basic principles for representing data recorded in terminological data collections. This framework includes a meta-model and methods for describing specific terminological mark-up languages (TMLs) expressed in XML. The mechanisms for implementing constraints in a TML are also defined in ISO 16642:2003. ISO 16642:2003 is designed to support the development and use of computer

applications for terminological data and the exchange of such data between different applications. In the framework of digital media, MPEG4 and MPEG7 deal with multilingual data. MPEG-4 is standard for multimedia for the fixed and mobile web and MPEG-7 is standard for description and search of audio and visual content.

When we closely examine the different standards or formats developed by these groups, we find that they have many overlapping features. For example all these formats are based on XML schemas, provide extensibility, and bridge the gap between two systems or tools in different languages. There are many identical requirements for all the formats irrespective of the differences in final output. For example, all the formats aim at being user-friendly, easy-to-learn, and at reusing existing databases or knowledge. All these formats work well in the specific field they are designed for, but they lack a synergy that would make them interoperable when using one type of information in a slightly different context, giving rise to the fear of competition between them.

3 DATA CATEGORIES[2]

3.1 Notation

- **"/grammatical gender/"** means we are speaking about a Data Category called "grammatical gender". This form is used in documentation.
- **"grammaticalGender"** (camel writing) means we are talking about the computer implementation of the /grammatical gender/ Data Category. This form is used in the computational format.
- **"/definition/./note/"** means we are talking about the /note/ which is a refinement of the /definition/
- **"LS./definition/"** means we are talking about the /definition/ anchored on the LS level (Language Section) of one meta-model
- **"TMF->LS./definition/./note/"** means we are talking about the /note/ which is a refinement of /definition/ anchored on LS level of the TMF meta model.

3.2 Definitions

- A **Data Category** (DC or DatCat) is the generic term to talk about the concept. In this document, it defines the Data Category accepted by the ISO committee. ISO 12620 "Data Categories" [15].
- The **Proposal** is a candidate Data Category suggested by an expert. When an ISO committee accepts a proposal, it becomes a "standard Data Category".
- The **Descriptor** is a standard Data Category or a proposal defined in the ISO12620-1 and used for the description of a Data Category (!). In fact, the ISO12620-1 defines the computational way to describe a Data Category in a very simple XML format called the Generic Mapping Tool (GMT) defined in the ISO16642-2003.

Some precisions:

[2] SYNTAX Help Manual [18].

- All Data Categories are called by a unique set of descriptors {/identifier/, /version/, /registration authority/}
- The linguistic content of the descriptor (like /definition/) of a Data Category can be translated. In this case, a specific label called “working language” gets the value from the ISO639-2 language codes [16] (ex: “en” for English, “fr” for French). The translation is not an official description of a Data Category but a more comfortable way for a lot of people to access to the work of the committee and to use it in their applications. That is the reason why we are talking about "label" and not about "descriptor".

Figure 1 shows that there are two kinds of Data Categories, coming from the ISO 11179 [17] description: complex and simple. A complex Data Category uses a set of simple Data Categories like possible values. Data element and Value domain are all the possible vocabulary for implementation of the Data Categories in different XML formats.

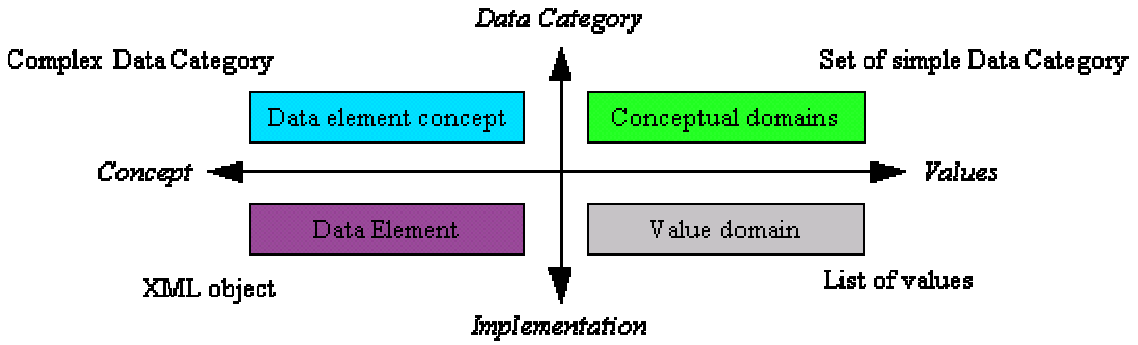

Figure 1: Data Categories from ISO 11179.

Figure 2 shows an example with the complex Data Category /grammatical gender/ which can get the values /masculine/, /feminine/ and /neuter/ as simple Data Categories. The implementation of these Data Categories in TMX (a specific XML format) are respectively “gen” then “m”, “f” and “n”.

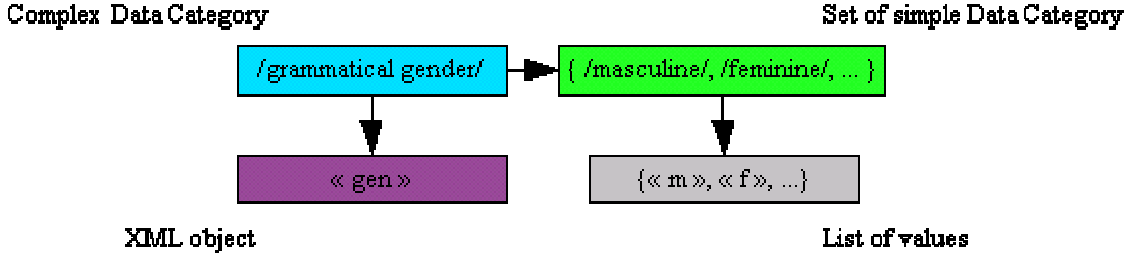

Figure 2: Example with the complex Data Category /grammatical gender/.

3.3 Data Category Registry

A Data Category Registry (DCR) is the collection of all Data Categories (complex and simple) and all proposals:

1. Data Categories **accepted** by the ISO normalisation process. The status of these Data Categories informs about the state and the history of the Data Category:
   a. STANDARD is the more recent Data Category,
   b. SUPERSEDED means that it exists a more recent Data Category,
   c. DEPRECATED means the Data Category is obsolete,
   d. CANDIDATE means we are talking about a proposal and not a Data Category.
2. Proposals **in creation** in the private working space of experts (i.e. the CANDIDATE-PRIVATE Data Categories)
3. Proposals **in progress** in the ISO normalisation process (i.e. the CANDIDATE-IN PROGRESS)
4. Proposals **rejected** by the ISO normalisation process (i.e. the CANDIDATE-REJECTED)

So, we used 2 descriptors to record the status of a Data Category or a proposal: /registration status/ and /administrative status/. Figure 3 below resumes the behaviour of a DCR.

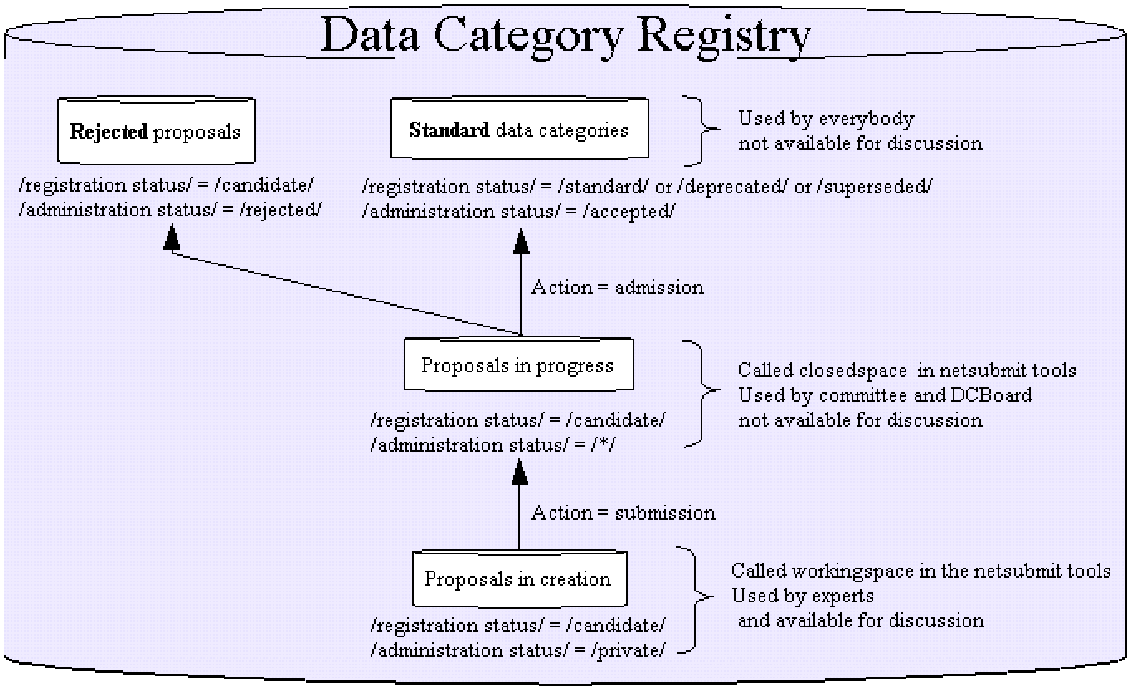

Figure 3: Behaviour of a Data Category Registry.

All Data Categories in a DCR are “one and only one”, that means there is “one and only one” Data Category in the DCR with an implementation of the set {/identifier/, /version/, /registration authority/}; for example: {/grammatical gender/,/1.0/,/ISO/}.

3.4 Data Category Selection

Figure 4 illustrates the notion of Data Category Selection (DCS) is a set of Data Categories (complex or simple) or proposals. It is composed of:

- Data Categories (the last version or one coming from its history)
- Proposals which will be submitted to an official ISO committee
- Proposals needed only for the user without submission.

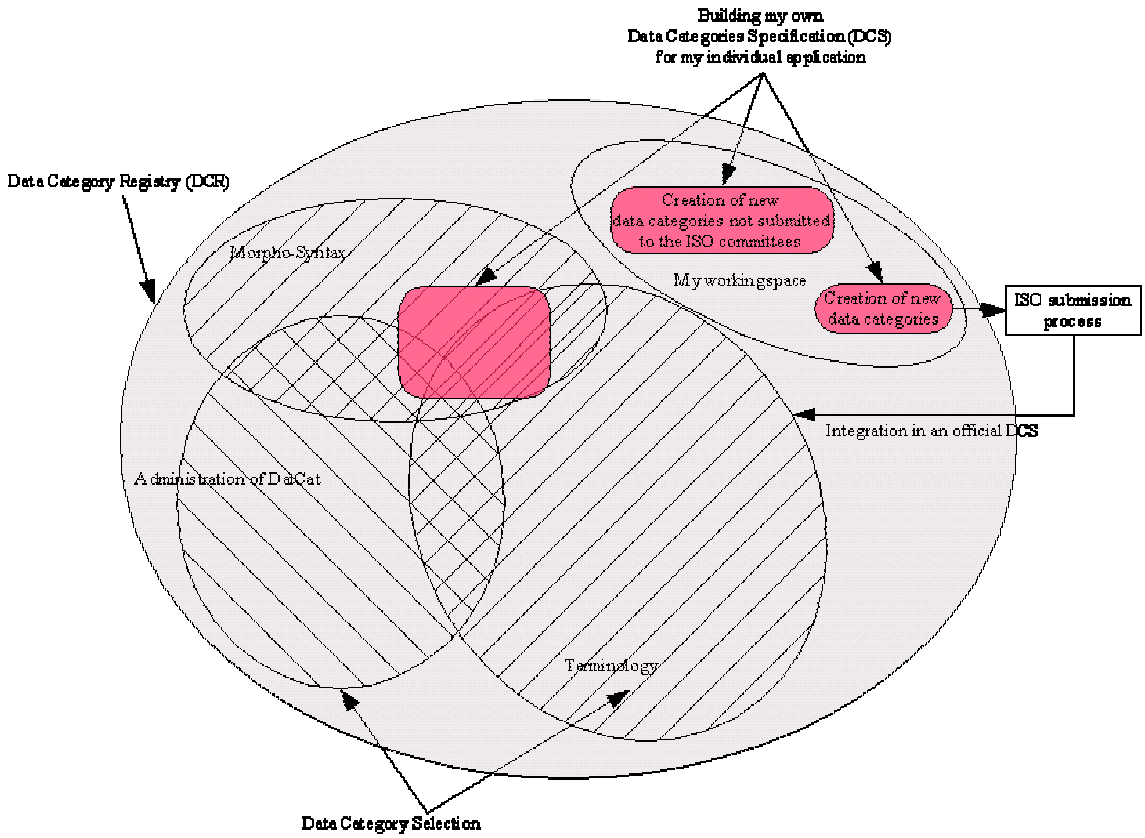

Figure 4: Data Categories (DatCat), Data Categories Selection (DCS) and Data Category Registry (DCR).

A DCS can represent:

- the model of an individual application,
- the model of a specific domain. In this case, some DCS corresponds to an ISO committee like morpho-syntax, terminology, etc.

Note that:

- All Data Categories are called by a unique set of descriptors {/identifier/, /version/, /registration authority/},
- All Data Categories in a DCR are "one and only one", that means there is "one and only one" Data Category in the DCR with the set {/identifier/, /version/, /registration authority/},
- Actually the database is not "clean". That means that all required working language are not filled and some needed descriptors are missing. The reason is that the most part of the database are /candidate/ (ie: in creation or in progress).
- In this case, the working language combobox can get 2 more values:
    1. "none" means that the working language is missing,
    2. "undefined" means that the tool cannot find a common working language in all the selected Data Categories.

DCS are needed to identify the individual information units making up a data collection, or annotation scheme, for a given language resource. Figure 5 below shows possible uses for a DCS. As exemplified in ISO 16642, Terminological Mark-up Framework (TMF), a Data Category Selection is needed in order to define, in combination with a meta-model, the various constraints that apply to a given domain-specific information structure or interchange format (e.g. expressed in XML). These constraints can be typically expressed as a DTD, a RelaxNG schema or an XML schema that will allow a computer application to check the validity of a language resource data collection against the intended specifications or to utilize a set of XSLT filters that will map the collection from one mark-up language to a neutralized dumped format (such as GMT in the case of ISO 16642) for archiving purposes and back, with the global purpose of mapping one mark-up language or format to another.

From a wider perspective, a formal model for representing Data Categories must account for the fact that apart from pure computer use, a Data Category Selection can be intended for human use as well. For instance, such specifications can form the core of a Data Category Registry, which can be published either as a paper document (such as the printed version of ISO 12620-2) or an electronic resource, such as the global Data Category Registry (DCR) for TC 37 language resources. Typically, the designers of a given mark-up language or data management system will query such a registry in order to create their individual application profiles by selecting a DCS from the global DCR. As a consequence, the formal representation of a Data Category shall comprise the specific attributes that document it (e.g., the Data Category name, definition, examples, comments, etc.). Finally, providing a precise description of the Data Categories used within a given data collection in reference to certified registries allows for a quick diagnosis of the compatibility of this collection with any particular computer application and thus acts as metadata for this collection.

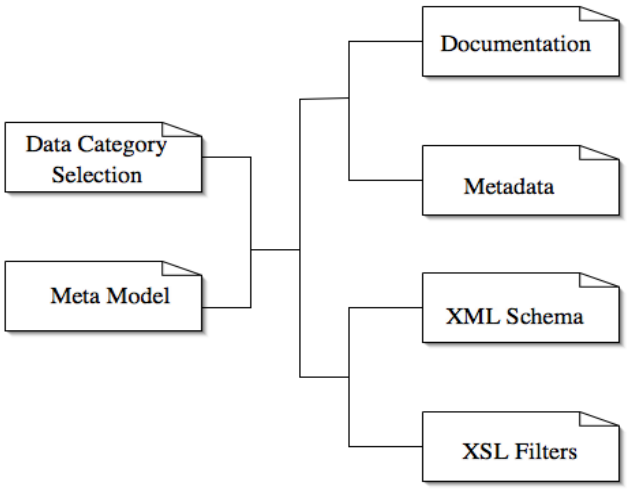

Figure 5: The role of a DCS in the context of the definition of linguistic annotation schemes.

If the specification also contains the provision of styles and vocabularies (cf. TMF-ISO 16642) for each Data Category, the DCS then contributes to the definition of a full XML information model which can either be made explicit through a schema representation (e.g. a W3C XML schema), or by means of filters to and from the GMT representation.

In addition, the DCS can be seen as a documentary source for the linguistic annotation scheme in question. Indeed, the fact that it contains the list of all data items that the annotation scheme can make use of, it is probably the best source of information for potential users or implementers who want to know whether a given item corresponds to their needs.

Furthermore, the Data Category Selection can be attached (or referenced) in any data transmission process to provide the receiver with all the information needed to interpret the content of the information being transmitted. In particular, this procedure should allow linguistic data expressed in various kinds of XML representations to be sent or received in the most transparent way.

Figure 4 above illustrates also the relationship between Data Category Specifications, the DCR, and any one of the possible DCSs that can be subsetted from the DCR. The patterned cells represented in the drawing correspond to individual DCS, each describing a given Data Category concept using the Data Category attributes set down in ISO 12620-1 with reference to the attributes defined in ISO/IEC 11179. Some Data Categories included in the DCR for terminology and other language resources are pertinent to a single thematic domain within this field. For instance, a concept identifier is probably unique to terminological resources (although not prescriptively), or a sense number is probably specific to lexicographical resources. Nevertheless, many Data Categories, frequently those of a strictly linguistic nature such as part of speech, grammatical gender, grammatical number, etc. are common to a wide variety of resources. To be sure, these categories may not always have the same function in different thematic domains, but they nevertheless represent the same essential token relationship in different kinds of resources. Hence each thematic domain contributes all its Data Categories in the form of Data Category specifications to the global Data Category Registry, while at the same time identifying those Data Categories that it shares with other kinds of resources. A standard listing the subset of Data Categories used in a thematic domain will comprise a domain-specific Data Category Selection (DCS) taken from the DCR. The oval shapes in the Venn diagram represent such DCS subsets. A further, smaller subset can be selected from the domain DCS for use in a given application or collaborative environment. The pink coloured shapes appearing in Figure 4 represents such a

smaller subset. Note that while some of the Data Categories contained in this subset are common to several different kinds of language resources, this particular application is wholly contained within the DCS for terminological entries, so we can conclude that it is designed for use with a terminological application.

4 THE MULTI LINGUAL INFORMATION FRAMEWORK (MLIF)

4.1 Introduction to MLIF

The Multi Lingual Information Framework (MLIF) is designed with the objective of providing a common platform for all the existing tools developed by the groups listed in the previous section. It promotes the use of a common framework for the future development of several different formats: TBX, TMX, XLIFF, Timed Text, TMF, etc. It does not create a complete new format from scratch, but suggests that the overlapping issues should be handled independently and separately. It will save time and energy for different groups and will provide synergy to work in collaboration. Presently, all the groups are working independently and do not have any mechanism for taking advantage of each other's tools. MLIF proposes to concentrate on only those specific issues that are different from others and specific to one format only, so it will create a smaller domain for the groups' developers. It gives more time to concentrate on a subset of the problems they are currently dealing with and creates a niche that helps in providing a better solution for problems of multilingual data handling and translation issues.

In MLIF, we deal with the issue of overlap between the existing formats. MLIF involves the development of an API through which all these formats will be integrated into the core MLIF structure. This is done through the identification and a selection of data categories as stated in ISO DIS 12620-1. MLIF can be considered as a parent for all the formats that we have mentioned before. Since all these formats deal with multilingual data expressed in the form of segments or text units they can all be stored, manipulated and translated in a similar manner. This kind of data can easily be stored in data categories and in terminological mark-up. The results of IST SALT project [19] clearly show that it is not difficult to edit, store and reuse data categories. The SALT project combines two interchange formats: OLIF [20], which focuses on the interchange of data among lexbase resources from various machine translation systems, and MARTIF [21], which facilitates the interchange of termbase resources with conceptual data models ranging from the simple to the sophisticated. It provides a graphical user interface that can be used to access or to define new data categories or modify them.

Figure 6 shows how it should look like when the MLIF is integrated with other formats.

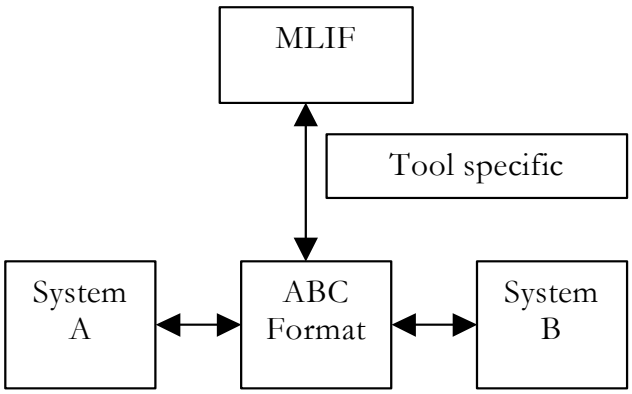

Figure 6: Schematic representation of MLIF.

In Figure 6 ABC format can represents any existing format (e.g. TMX, XLIFF, Timed Text). It can also be possible to develop new formats those may be more flexible and easier-to-use then these formats. System A and system B represents corresponding systems between which any of these format operates. These systems have very different architecture in themselves, but they are integrated to these formats as a separate identity, so there seems to be no requirement to make any change in the working of systems A/B, these should continue to operate as they are operating nowadays and suggested in these existing formats. The only issue where we need to put our attention is identification of "Tool specific inputs". These inputs will make it easy to flow data between MLIF and other formats. Since these will be dependent on the format in question so we need to discuss all these in separate sections.

4.2 Hierarchical overview of MLIF

All the different models have very similar hierarchical structure but they have different terms and methods of storing metadata relevant to them in particular. MLIF provides a generic structure that can establish basic foundation for all other models. This model will provide flexibility to make any element or attribute of this format to be defined explicitly or not. If the value is not defined explicitly it will take default value. Most of the models will also defined their own elements and attributes those will fit into this using extensibility that is one of the basic requirements of MLIF model. A MLIF document has a hierarchical structure as shown in Figure 7. This document will have MLIF as the highest level element, which consists of a mlifHeader element and a mlifBody element. mlifHeader contents meta data related to the document content and other necessary information about the document. Most of the data of this metadata can be formed with the help of core metadata elements suggested in Dublin Core Approach. This is comprised of all the major elements needed to define the metadata related to content of any document and can be used at any level of metadata development. mlifBody contents one multiLingualEntry element. multiLingualEntry can have one or more multiLingualUnits. Each of these multiLingualUnits can have any number of monoLingualEntry. monoLingualEntry are used to store monoLingualSegs, these contains

monolingual data. There can be any number of monoLingualSegs inside one monoLingualEntrys, but will contain data in one language only. These monoLingualUnit and monoLingualEntry can be used in any order in the MLIF document.

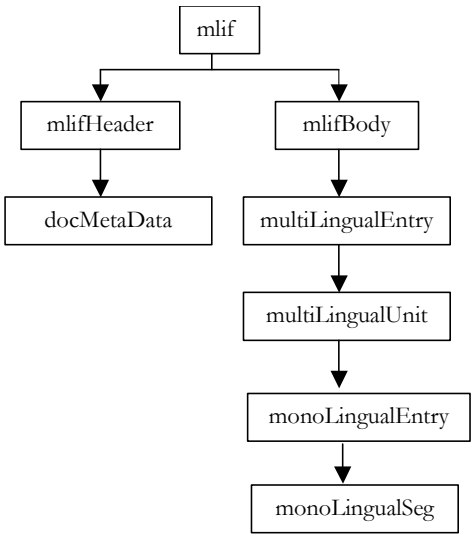

Figure 7: Hierarchical representation of MLIF.

monoLingualSeg is the basic unit of MLIF document. It contents all data, in one language, related to the segments. These elements content information about lexicographical entries, phonetics, lexemes related information, ontological concept defined by segment and some note. In section 4.3 we have included one example that gives more insight of how MLIF document looks like. MLIF is XML compliant so it carries extensibility from the XML, and it become highly extensible. It carries following properties from XML schema format and hence inherits the advantage of using XML schemas:

- An XML declaration, which, beyond identifying that the current document is an XML one, allows one to declare the character encoding scheme used in the document (e.g. iso-8859-1, utf-8, etc.);
- XML is both Unicode and ISO 10646 compatible, which means that, in the context of TMX, there is no limitation in the use of writing systems and languages in the content of the information exchanged within the dialogue system or with the external multimedia database;
- XML comes along with a specific mechanism, called namespaces, allowing one to combine, within the same document, mark-up taken from multiple sources. This very powerful mechanism, which is in particular the basis for XSLT and XML schemas, allows more modularity in the definition of an XML structure and also to reuse components defined in other context;
- XML provides a general attribute, 'xml:lang' to indicate the language used in a given element.

4.3 Example: A MLIF document

This example shows how a very simple MLIF document looks like:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Example of MLIF document  -->
<mlif version="1.0" type="general" xml:lang="en">
    <mlifHeader>
        <docMetaData approach="dublinCoreApproach"/>
    </mlifHeader>
    <mlifBody>
        <multiLingualEntry>
            <multiLingualUnit>
                <monoLigualEntry>
                    <monoLingualSeg xml:lang="en">
                        <texte>
                             MLIF is foundation for all other formats.
                        </texte>
                        <note>
                             this is simple example for demonstration.
                        </note>
                    </monoLingualSeg>
                </monoLigualEntry>
                <monoLigualEntry>
                    <monoLingualSeg xml:lang="fr">
                        <texte>
                             MLIF est la base de tous les autres formats.
                        </texte>
                        <note>
                             ceci un exemple simple pour démonstration.
                        </note>
                    </monoLingualSeg>
                </monoLigualEntry>
            </multiLingualUnit>
        </multiLingualEntry>
    </mlifBody>
</mlif>
```

Figure 8: A very simple MLIF document.

The MLIF structure of document shown above can be mapped on any other format with the help of Data Categories identified in a later section. The XML document represented above can be mapped to the abstract model described below (for simplicity we have extended only one branch in the model, but in general it can be extended from any point)

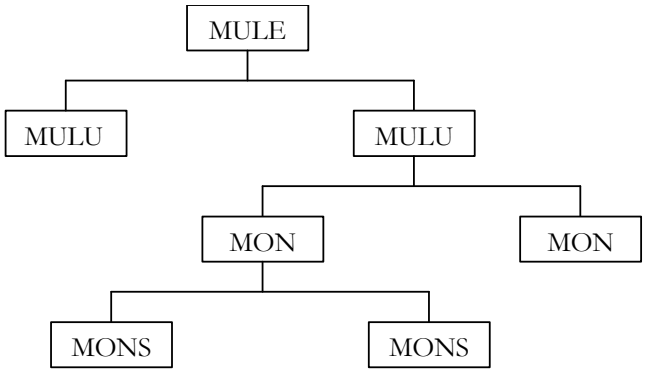

Figure 9: Mapping an MLIF document to an abstract model.

4.4 XML canonical representation of the generic model

The final component of MLIF is the definition of a simplified XML application that can be used to map any given format onto the abstract components of MLIF. This format, also known as GMT (Generic Mapping Tool), is based on a reduced set of XML elements and attributes, as we shall see, serve as containers for nodes of the structural skeleton (identified by <struct> tags) and Data Categories (identified by <feat> tags). Any mapping to any other format can be implemented as the composition of two elementary mappings through GMT. The following example illustrates how the information contained within the MLIF example in section 4.3 can be encoded in GMT format.

```
<struct xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="file:/MLIF/GMT.xsd"
    type="mlif">
    <struct type="MetaData">
        <feat type="approach">dublinCoreApproach</feat>
    </struct>
    <struct type="MultiLingualEntry">
        <struct type="MultiLingualUnit">
            <struct type="MonoLingualEntry">
                <struct type="MonoLingualSegment">
                    <feat type="language" xml:lang="en"/>
                    <feat type="text">
                        MLIF is foundation for all other formats
                    </feat>
                    <feat type="text">
                        this is a simple example for demonstration
                    </feat>
                </struct>
            </struct>
            <struct type="MonoLingualEntry">
                <struct type="MonoLingualSegment">
                    <feat type="language" xml:lang="fr"/>
                    <feat type="text">
                        MLIF est la base de tous les autres formats
                    </feat>
                    <feat type="text">
                        ceci est un example simple pour démonstration
```

```
                    </feat>
                </struct>
            </struct>
        </struct>
    </struct>
</struct>
```

Figure 10: Canonical representation of a MLIF document.

Here we have chosen one TMX example for the purpose of showing how MLIF can be mapped on other formats. This TMX example is very simple and contents very basic structure of TMX document. In this example we are discussing some elements of TMX format in body of TMX document. But as we discuss further details about MLIF, it will be clear that all features can be identified and mapped through Data Categories. Later on we will also show that elements of header and attributes can also be mapped similarly.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE tmx SYSTEM "file:/TMX/tmx14.dtd">
<tmx version="1.4">
    <header creationtool=""
            creationtoolversion=""
            segtype="phrase"
            o-tmf=""
            adminlang=""
            srclang=""
            datatype=""/>
    <body>
        <tu>
            <prop type="group">1</prop>
            <tuv xml:lang="en">
                <seg>First segment</seg>
            </tuv>
            <tuv xml:lang="fr">
                <seg>Premier segment</seg>
            </tuv>
        </tu>
        <tu>
            <prop type="group">1</prop>
            <tuv xml:lang="en">
                <seg>Second segment</seg>
            </tuv>
            <tuv xml:lang="fr">
                <seg>Second segment</seg>
            </tuv>
        </tu>
    </body>
</tmx>
```

Figure 11: A TMX document.

In the following table we have shown comparison of MLIF and TMX elements.

TMX	MLIF
Header	MlifHeader
Body	MlifBody
Tu	MultiLingualUnit
Tuv	MonoLingualEntry
Prop (defines the properties of parent)	Wrap
Seg	Text

Tableau 1: Comparaison of MLIF and TMX elements.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<struct xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="file:/MLIF/GMT.xsd" type="mlif">
    <struct type="Head">
        <feat type="approach">dublinCoreApproach</feat>
    </struct>
    <struct type="Body">
        <struct type="MultiLingualEntry">
            <struct type="MultiLingualUnit">
                <struct type="Wrap">
                    <struct type="MonoLingualEntry">
                        <struct type="MonoLingualSegment">
                            <feat type="language" xml:lang="en"/>
                            <feat type="text">First segment</feat>
                        </struct>
                    </struct>
                    <struct type="MonoLingualEntry">
                        <struct type="MonoLingualSegment">
                            <feat type="language" xml:lang="fr"/>
                            <feat type="text">Premier segment</feat>
                        </struct>
                    </struct>
                </struct>
            </struct>
        </struct>
        <struct type="MultiLingualEntry">
            <struct type="MultiLingualUnit">
                <struct type="Wrap">
                    <struct type="MonoLingualEntry">
                        <struct type="MonoLingualSegment">
                            <feat type="language" xml:lang="en"/>
                            <feat type="text">Second segment</feat>
                        </struct>
                    </struct>
                    <struct type="MonoLingualEntry">
                        <struct type="MonoLingualSegment">
                            <feat type="language" xml:lang="fr"/>
                            <feat type="text">Deuxième segment</feat>
                        </struct>
                    </struct>
                </struct>
            </struct>
        </struct>
    </struct>
```

```
</struct>
```

Figure 12: An MLIF document corresponding to the TMX document (cf. Figure 11).

As we have seen in this table all of the elements from TMX format can be matched with some other elements in the MLIF, same kind of conclusion can be drawn for other formats too. One thing that is very important in the development of MLIF is it has to be very flexible because it has to support many formats in one format. It is extremely important to understand the requirements of the different formats and their specific inputs. In the next section we have identified specific inputs from different formats, later on we will discuss how these inputs can be incorporated with the design of MLIF.

4.5 Tool specific inputs

After study of different formats we find that many of the requirements are common and exist with all the formats, so they will be an integral part of MLIF core structure itself. For example all the formats in question are either XML compliant or are in XML itself. So for of obvious reasons MLIF is also an XML format and provides all the functionalities of XML. Some other common features are using standards for time/date, language codes, and country codes. MLIF will follow all these standards at core level so these issues will not surface on individual specifications levels. In this section we will discuss about the inputs from all different formats those are required from these formats to implement over the MLIF format to make them as they look today without the help of MLIF. Specific inputs for different formats have been identified based on the current available versions of these formats.

4.5.1 TBX

This format is the simplest one to fit into MLIF format because it already uses Data Categories or more specifically “terms” to define and design to support machine processing of terminological data. TBX does also provide flexibility because it is based on different standards. Its core structure is based on MARTIF, ISO 12200 and it uses inventory of well defined “Data Categories” from ISO 12620. Not only this, it also allows the users to define their own Data Categories. MLIF should also be flexible enough to incorporate these different standards and terminological environments. In TBX this is done with the help of TBX eXtensible Constraint Specifications (XCS) file. MLIF should also provide some mechanism to use similar concept, as a part of MLIF itself or should be able to import it through extensibility of MLIF. MLIF should also provide mechanism so that it remains valid according to TBX core-structure module.

4.5.2 TMX

This format is also close to the proposed MLIF format. In MLIF format we have used term Data Categories and TMX uses term segments to define very similar concepts. These segments are bundles of information specific to particular translation unit. In TMX all the elements and attributes are defined in lower case, so in MLIF either we need to follow the

same rule or have to provide some other strategy to handle this issue. MLIF should also continue to support two levels of implementation as are supported in TMX, this is required for the adoption of MLIF by the people who will have their data in currently available two-level TMX implementation.

4.5.3 XLIFF

OASIS XLIFF defines extensible specification for interchange of localization information. MLIF should be able to mark-up and capture localization information and interoperate with different processes and phases without loss of information. It should fulfil specific requirements of being tool-neutral. It should support the localization related aspects of internationalization and entire localization process. It also needs to support common software and content data formats. This should also provide an extensibility mechanism to allow the development of tools compatible with an implementer's proprietary data formats and workflow requirements.

4.5.4 TIMED TEXT

This format is a content type that represent times text media for interchange among different authoring tools. Timed text media is textual information associated with timing information of media. This format helps in synchronization of textual information with the audiovisual content of the multimedia object. This format needs to keep timing information precisely for all the content, intrinsically or extrinsically. MLIF should provide capability to embed this information into the multimedia objects descriptions. There are many specific requirements of the "timed text" format those can be handled if information is supplied with all inputs. For example it should be able to handle multiple natural languages, it should keep information about hyper-linking, embedded graphics, flow of textual information etc.. More data items needed to be provided for "Timed Text" are listed under Data Categories in timed text of next section.

4.5.5 I18n

The W3C Internationalization Activity has the goal of proposing and coordinating any techniques, conventions, guidelines and activities within the W3C and together with other organizations that allow and make it easy to use W3C technology worldwide, with different languages, scripts, and cultures. The Internationalization (i18n) Working Group comprises three task forces: Core, Web Services and GEO (Guidelines, Education & Outreach).

The following terminology is widely used: [22]

- I18n (internationalization) means modification of a software or related technologies so that a software can potentially handle multiple languages, customs, and so on in the world.
- L10n (localization) means implementation of a specific language for an already internationalized software.

However, this terminology is valid only for one specific model out of a few models that we should consider for i18n. Now we will introduce a few models other than this I18n-L10n model:

1. L10n (localization) model
   - This model is to support two languages or character codes, English (ASCII) and another specific one. Examples of software which is developed using this model are: Nemacs (Nihongo Emacs, an ancestor of MULE, MULtilingual Emacs) text editor which can input and output Japanese text files, and Hanterm X terminal emulator which can display and input Korean characters via a few Korean encodings. Since each programmer has his or her own mother tongue, there are numerous L10N patches and L10N programs written to satisfy his or her own need.
2. I18n (internationalization) model
   - This model is to support many languages but only two of them, English (ASCII) and another one, at the same time. One has to specify the "another" language, usually by LANG environmental variable. The above I18N-L10N model can be regarded as a part of this I18N model. Gettextization is categorized into I18N model.
3. M17n (multilingualization) model
   - This model is to support many languages at the same time. For example, Mule (MULtilingual Enhancement to GNU Emacs) can handle a text file that contains multiple languages - for example, a paper on differences between Korean and Chinese whose main text is written in Finnish. GNU Emacs 20 and XEmacs now include Mule. Note that the M17N model can only be applied in character-related instances. For example, it is nonsense to display a message like 'file not found' in many languages at the same time. Unicode and UTF-8 are technologies that can be used for this model.

4.5.6 MPEG-4/MPEG-7

MPEG-7 is formally known as "Multimedia Content Description Interface". Audiovisual information plays important part storing any kind of data and increasing everyday. This information needs to be defined and stored. This format offers a comprehensive set of tools for audiovisual description. These tools store information in the form of metadata elements. As such MPEG-7 does not depend upon the form of representation of audiovisual material, it is going to describe but still it can take help if the data is described in MPEG-4 formats. This format needs to define information about the content of the multimedia objects, so it needs to describe all the data in a systematic and structured way that can be used later by any agent interest in the details of the data it describes.

5 IMPLEMENTING LANGUAGE TRANSLATION BY USING MLIF: THE ITEA "JULES VERNE" PROJECT EXPERIENCE

5.1 The RAMO Model

The RAMO (Reactive & Adaptive Multimedia Object) concept has been developed [23], [24] within the ITEA "Jules Verne" project. RAMO is based on the notion that a new dimension of interactivity can be achieved by enabling multimedia objects to fulfil the following criteria:

- To become fully autonomous,
- To be independent from predefined scenarios,
- To fully emulate the characteristics and behaviours of the represented entities.

Such objects are able to react and adapt themselves to any contextual situations resulting from interactions with other objects of the application and/or from users actions. Interactivity in such a concept has to be seen at two different levels: the object level and the system level (running environment). Interactivity is not deterministic at the system level. At the object level it deals with two kinds of scenarios: predefined and evolving ones. Predefined scenarios consist in predefined behaviours that are initially set up and based on a stimulus/reaction model. The evolving scenarios of an object refer to the notion of intelligent adaptive and learning entities. The dynamic evolution of the scene composition, the contextual situations and the object adaptations create multiple and complex combinations that lead to unpredictable solutions. Mixed time-based and event-based multimedia scenarios will be designed.

The RAMO model will be based on a flexible standard able to describe scene taking into account the following objects configurations that are:

- The structures,
- The relations,
- The constraints, and
- The attributes.

In addition to this, it should specify the basis upon which object rendering will be performed for the corresponding contexts with the use of the relevant technologies in order to be platform independent.

A RAMO MPEG-7 based description scheme (RAMO DS) is being developed in order to enable to completely describe RAMO objects in terms of states and behaviours. The states will be described based on "Sensorial / AV Representations" and in terms of "Temporal and Spatial Matches". The behaviours will be described in terms of "Processes" and of "Events". The MPEG-7 Multimedia Description Schemes (MPEG7-MDS) is an interesting technical choice to describe the RAMO Scenes and Objects. It brings standardized means to the description and the specification of the following main features:

- The structures of entities and their internal relations between elements;
- The multimedia content for attributes with their type definition;
- The multimedia resources for control and access;
- The references to external resources;
- The relations and interactions among entities;
- The granularity, the flexibility and the openness that the XML standard offers.

The RAMO model has to specify the communication protocols to set up between the objects as well as the way RAMO components (Application, Scenes and Objects) are processed. This has to be achieved starting from the object management level up to the system level (application supervision). The RAMO model aims at being a generic model usable for many kinds of interactive multimedia presentation. It should be implementation independent and open to and compatible with extension profiles.

5.2 RAMO and MLIF

The main idea is to create specific Data Categories dedicated to generic RAMO scenes, objects and their related properties and to use MLIF as a meta-model [25], [26].

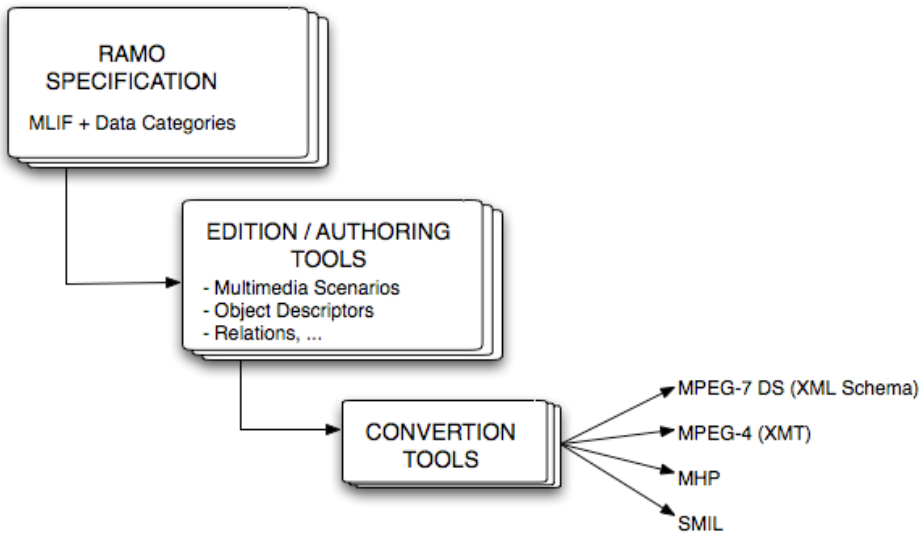

Figure 13: RAMO and MLIF.

High-level categories may consist in RAMO Actors Data Categories, Behaviour Data Categories and Event Data Categories, for instance. Low-level categories of generic RAMO descriptors may be media-oriented descriptors, directly inspired by the MPEG-7 standard; for instance Video Data Categories, Audio Data Categories, Animation Scenario Data Categories, etc.

5.3 SMIL and XMT

Currently, in the framework of ITEA “Jules Verne” project, we are experimenting some basic scenarios by using XMT[3] (“eXtensible MPEG4 Textual format”) [2] and SMIL (“Synchronized Multimedia Integration Language”) [27].

The “Synchronized Multimedia Integration Language” enables simple authoring of interactive audiovisual presentations. SMIL is typically used for "rich media"/multimedia presentations which integrate streaming audio and video with images, text or any other media type. SMIL is an easy-to-learn HTML-like language, and many SMIL presentations are written using a simple text-editor. SMIL has the two following design goals:

- • Define an XML-based language that allows authors to write interactive multimedia presentations. Using SMIL, an author can describe the temporal behaviour of a multimedia presentation, associate hyperlinks with media objects and describe the layout of the presentation on a screen.
- • Allow reusing of SMIL syntax and semantics in other XML-based languages, in particular those who need to represent timing and synchronization. For example, SMIL 2.0 components are used for integrating timing into XHTML and into SVG.

- SMIL is defined as a set of mark-up modules, which define the semantics and a XML syntax for certain areas of SMIL functionality.

The “eXtensible MPEG4 Textual format” is a textual representation, using XML, of MPEG-4 Systems (ISO/IEC 14496-1). The MPEG-4 specification provides conformance points to ensure interoperability of bitstreams with receivers so that they can decode and render the content. MPEG-4 allows the representation of content that is both complex and highly interactive, as well as, of course, much simpler content. Binary coded MPEG-4 content can be stored in the MPEG-4 Systems file format commonly known as the mp4 format. While the mp4 file is exchangeable it is often difficult to subsequently use it to edit or re-purpose the stored content. This is because the binary coded representation often cannot be "reverse-engineered" in a consistent manner to represent the content author's original intentions.

The XMT has been designed to provide an exchangeable format between content authors while preserving the author's intentions in a high-level textual format. In addition to providing an author-friendly abstraction of the underlying MPEG-4 technologies, another important consideration for the XMT design was to respect existing practices of content authors such as the Web3D X3D, W3C SMIL and HTML.

The XMT is suitable for many uses including manually authored content as well as machine-generated content using multimedia database material and templates. The XMT may be encoded and stored in the exchangeable mp4 binary file or may also be encoded directly into streams and transmitted. XMT encoding and delivery hints exist to assist this process.

[3] We have directly reproduced here several paragraphs included in [2].

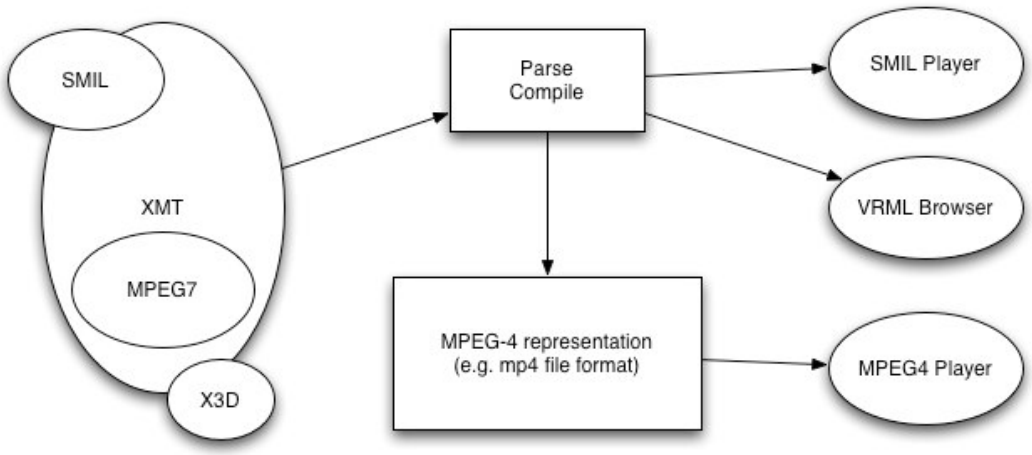

Figure 14: Interoperability of XMT with other standards.

The XMT consists of two levels of textual syntax and semantics: the XMT-O format and the XMT-A format.

The XMT-A is an XML-based version of MPEG-4 content that closely mirrors its binary representation. The goals of the XMT-A format are to provide a textual representation of MPEG-4 Systems binary coding, including supporting a deterministic one-to-one mapping to the binary representations for conformance, and to be interoperable with the X3D [[2]], which is being developed for VRML 200x (X3D) by the Web 3D Consortium.

XMT-A contains a textual representation of BIFS for both the nodes and the commands for scene updates. The XMT textual format for the nodes is fully compatible with the XML representation of the subset of X3D that is contained within MPEG-4. MPEG-4 Systems scene description originally used VRML 97 as a base and added further nodes of synthetic representation, audio and 2D support. VRML 97 is an ASCII textual representation while X3D is based on XML. Currently the X3D specification has additional function that is not present in MPEG-4; hence XMT contains its subset rather than complete coverage. The XML for additional nodes in MPEG-4 has been based on the same representation philosophy defined by X3D that originally guided their XML conversion from VRML plain text. This strategy will facilitate change if future harmonization efforts between MPEG-4 and X3D make more functionalities common.

XMT-A also has textual representations of features unique to MPEG-4 Systems, not found in X3D, such as the Object Descriptor Framework that includes ODs (Object Descriptors), other descriptors and commands. The ODs are used to associate scene description components to the Elementary Streams that contain the corresponding coded data. These include visual streams, audio streams as well as animation streams that update elements of the scene more efficiently than BIFS commands for complex animations. XMT-A also

includes a textual representation to allow MPEG-J streams to be created either from Java classes or zip files.

The XMT-O is a high-level abstraction of MPEG-4 features designed around SMIL 2.0. Also, as the MPEG standardization process is not yet complete for XMT, the XMT samples provided in this deliverable are based on the latest version of the XMT specification.

Using XMT-O authors can describe the temporal behaviour and layout media objects to form multimedia presentations, as well as associate hyperlinks with media objects in the presentation. Animation and interactive event based behaviour can also be easily expressed, as well as transition effects. Unlike SMIL the XMT has features that enable control over many intrinsic properties of the media objects in support of the capabilities present in MPEG-4.

For each XMT-O construct, one or more possible mappings exist to MPEG-4 Systems that can represent the same content. Such mappings may be represented in XMT-A by various combinations and sequences of XMT-A elements. From this discussion it can be concluded that there is no single deterministic mapping between the two levels, for the obvious reason that a high-level author's intentions can be expanded to more than one sequence of low-level constructs. However, with such non-determinism comes flexibility. XMT-O content can thus be converted and targeted to MPEG-4 players of varying capability. For example, animation in a scene may be mapped using BIFS commands for a simple player, for a more capable player the animation may be mapped, potentially more efficiently, to an animation stream.

Within XMT-A and XMT-O a set of common elements such as MPEG-7, authoring elements, encoding hints, delivery hints and publication hints have been identified and collected into a common section used within both formats that is designated XMT-C.
The goals of the XMT-O format are to provide ease of use for content creation, to facilitate content exchange among authors and authoring tools, and to provide a content representation that is compatible and interoperable with the Synchronized Multimedia Integration Language (SMIL) 2.0.

XMT-O describes audio-visual objects and their relationships at a high level, where content requirements are expressed in terms of the author's intent, using media, timing and animation abstractions, rather than by coding explicit nodes and route connections. Higher-level constructs facilitate, among other aspects, content exchange between authors and authoring tools, and content re-purposing.

An algorithm, or an authoring tool, would compile (or map) an XMT-O construct into MPEG-4 content, i.e., into BIFS, OD and media streams etc., along with any appropriate audio/visual media compression or conversions that may be required. Media sources can be of a variety of formats native to the machine where the algorithm is executing, and it is the responsibility of the tool, during the compilation phase, to convert media to suitable target formats for MPEG-4 and appropriate bit-rates etc.

In converting the XMT-O format to MPEG-4 there is not necessarily only one mapping for each construct. MPEG-4 nodes and routes are very powerful tools and there can often be more than one way to represent XMT-O constructs. Also, as MPEG-4 nodes can be

'wired' together with routes in many combinations, it is often difficult to reverse engineer an author's intent from a collection of nodes and routes. When confronted by content, containing many nodes and routes, the re-authoring and maintenance can be quite challenging, if the high-level view of that presentation must be inferred. The XMT-O, however, provides a high-level view with high-level authoring constructs and thus facilitates content exchange, rapid content re-purposing, re-authoring and ongoing maintenance of content.

Recognizing though that some authors may wish to access low-level nodes/routes, XMT-O allows the embedding of the XMT-A node and route definitions, within an identified low-level escape section, to create custom media constructs.

5.4 The XMT-O localisation round trip

The general idea is that we may want to obtain, by means of a translation / localization process, a document in English (see Figure 16) corresponding to an XMT-O or SMIL document which is in French (see Figure 15). Figure 17 shows how the localization process is performed.

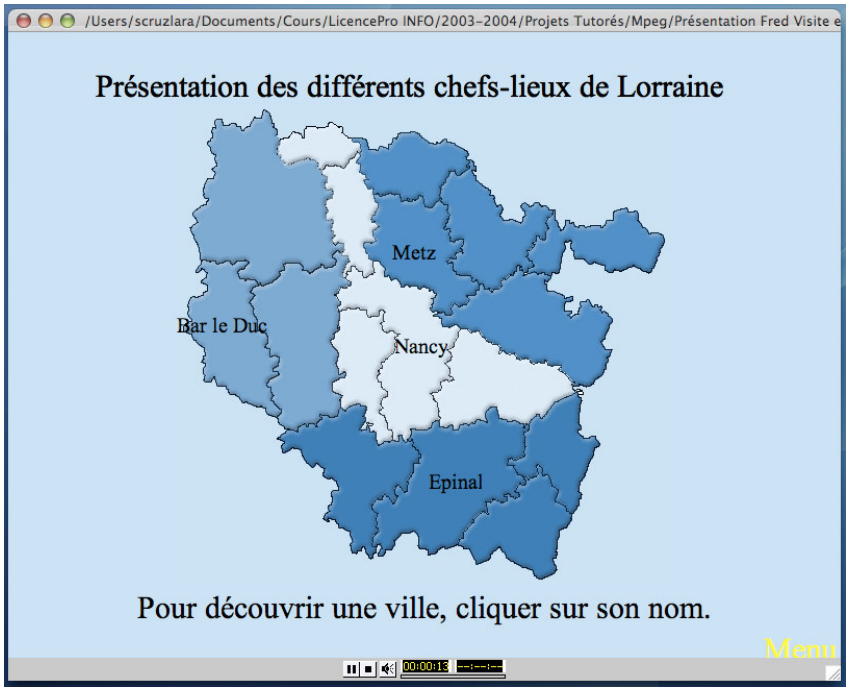

Figure 15: An interactive MPEG4 presentation in French.

And we would like to obtain,

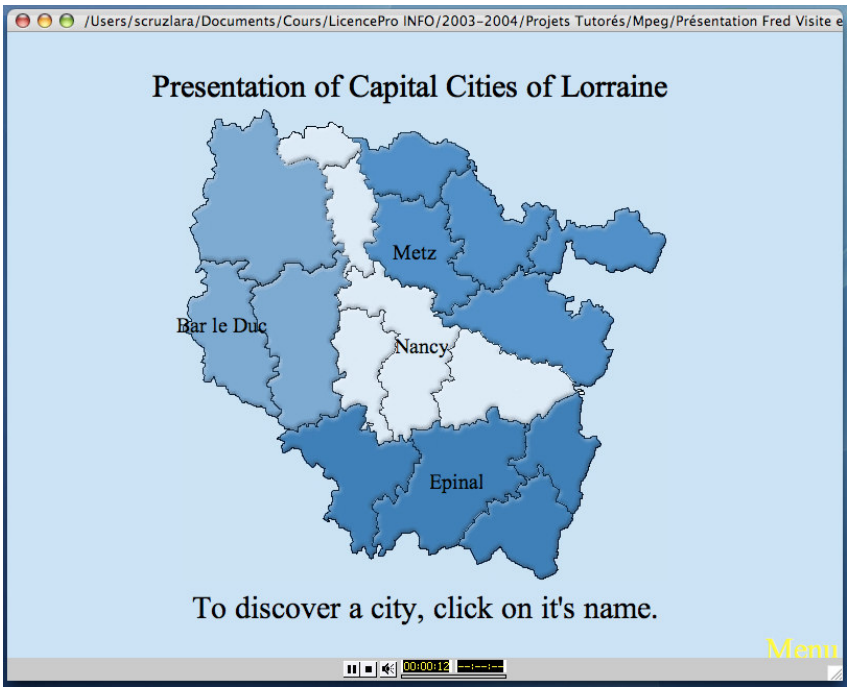

Figure 16: The same interactive MPEG4 presentation in English.

Figure 17 shows how the localization process is performed.

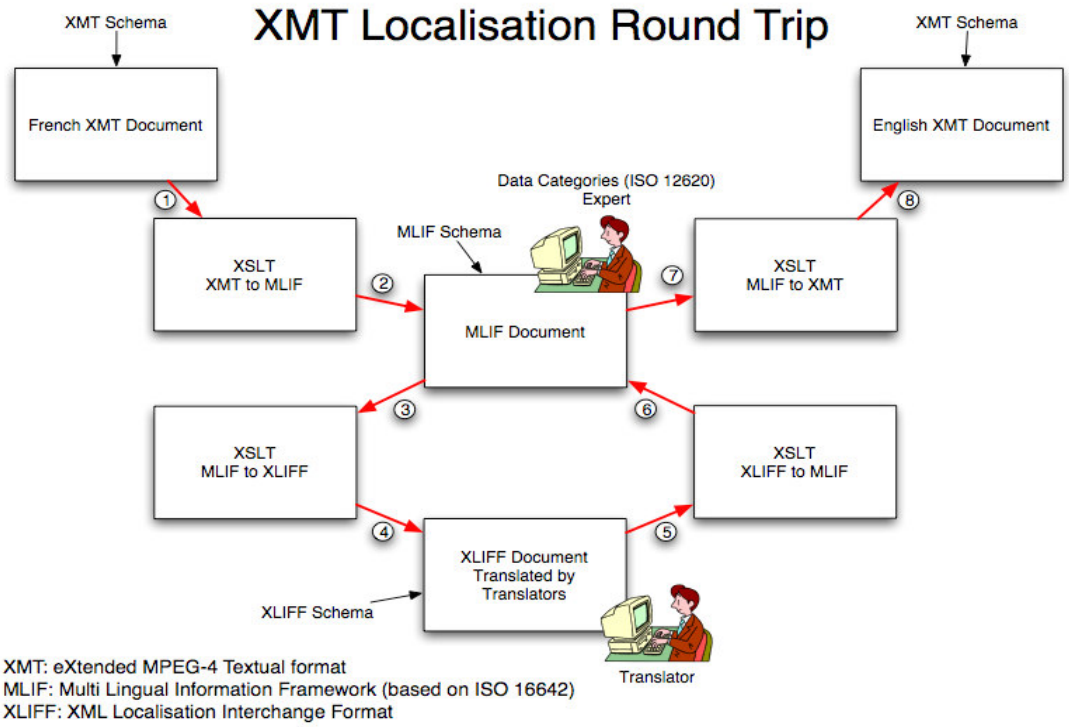

Figure 17: XMT-O Localization Roundtrip.

1. The original XMT_{french} document contains linguistic information in French.
2. Transformation of the XMT_{french} document into an $MLIF_{french}$ document.
3. Transformation of the $MLIF_{french}$ document into an $XLIFF_{french}$ document.
4. By using already existing XLIFF environments, a professional translator performs a French-English translation. We obtain an $XLIFF_{English}$ document.

5. / 6. Transformation of the $XLIFF_{English}$ document into an $MLIF_{English}$ document.

7. / 8. Transformation of the $MLIF_{English}$ document into an $XMT_{English}$ document.

5.5 Identifying monolingual content in XMT-O

Identifying monolingual content in XMT-O may be considered from two points of view:

1. Textual information related to meta data.
2. Textual information related to data (i.e., subtitles).

In the aim of simplicity, we shall only consider textual information related to data[4], that is, textual information that may be associated under the form of subtitles, to a multimedia presentation.

[4] A priori, meta data is more closely related to MPEG-7 DS, rather than to MPEG-4 nodes.

If we closely analyze the XMT-O documents [28] presented in the appendix (see **Erreur ! Source du renvoi introuvable.**), we shall see that all textual information related to data (i.e. subtitles) in a XMT-O document, is always included in the "textLines" attribute of the <string> tag, for example:

```
<string dur="800s"
        ...
        textLines="&quot;Présentation des différents chefs-lieux
                   de Lorraine&quot;; ">
        ... />
```

So, it is rather easy, by parsing a XMT-O document, to retrieve all monolingual textual information related to data. Developing a XSLT style sheet, allowing transforming a XMT-O document into an MLIF document, may also be a relatively easy task. However, we must verify that:

1. The XSLT style sheet preserves the original XMT-O structure when passing to a MLIF Document.
2. The XSLT style sheet takes into account all Data Categories related to the XMT-O original document.

So, if the task of identifying monolingual content inside a XMT-O document is rather straightforward, being able of identifying what are the actual Data Categories that must be taken into account, is a much more complex task.

The very first thing to do is to setup a Data Categories Specification (see section 3) related to Digital Media. This activity shall be very complex, because we have to:

1. Identify all existing Data Categories that may be used in the context of Digital Media knowing that several Data Categories may be common to several different kinds of language resources (see section 3.4).
2. Understand that most Data Categories related to Digital Media have not been yet identified and defined. It should be noted that identifying and defining Data Categories related to Digital Media, is a rather complex process because:
    i. Digital Media experts have to be involved in the identification of Data Categories for Multimedia.
    ii. Data Categories experts must approve all Data Categories identified (and proposed) by Digital Media Experts.
    iii. We must remember that defining Data Categories is an official ISO normalisation process.

Some multilingual content stuff has been integrated in demonstrations developed by INT (Institut National des Télécommunications d'Evry, France) and by CRP-HT (Centre de Recherche Public "Henri Tudor", Luxembourg). As most Data Categories related to Digital Media have not been yet identified and defined, in both demonstrations, the multilingual content part has been directly encoded by using XLIFF.

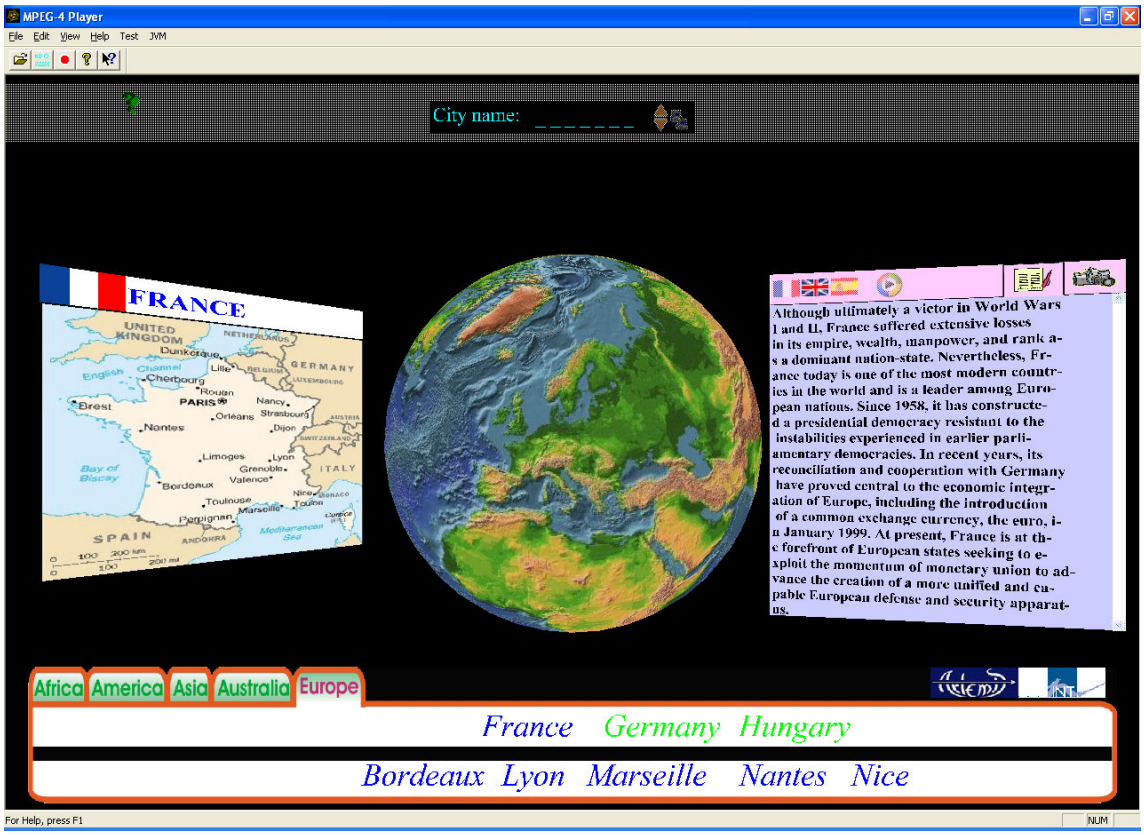

Figure 18: INT Demonstration (ITEA Symposium. Seville, Spain 2004). Textual information is in English.

Figure 18 shows a screen shot of a MPEG-4 Multimedia presentation developed by INT[5] for the ITEA Symposium (Seville, Spain 2004). In this screen shot, the textual information (in English) appearing at the right side of the screen has been retrieved from a XLIFF file. This file has been included in the Appendix (see 10.3). Figure 19 shows the same application once that the user has chosen to see textual information in French.

It should be noted, as we have clearly mentioned it before, that the multilingual content part of this application has been directly encoded by using XLIFF.

[5] © Institut National des Télécommunications d'Evry – Unité de projet "ARTEMIS".

Figure 19: INT Demonstration (ITEA Symposium. Seville, Spain 2004). Textual information is in French.

In section 6, we present two software applications that have been developed in the aim of allowing users to deal with multilingual content by using several formats: TMX, XLIFF, i18n and MLIF[6].

6 USING XSLT TO DEAL WITH DIFFERENT MULTILINGUAL CONTENT FORMATS.

Figure 17 has shown that dealing with multilingual content means transforming XMT documents, first into MLIF documents, and then into XLIFF documents. The XSLT style sheet in 10.4 is a very simple way to obtain MLIF documents (canonical from) from XMT-O documents. It should be noted that this XSL style sheet allows only to retrieve textual information from XMT-O documents and that the original structure is not preserved.

[6] We are using a preliminary version of MLIF that do not include all required Multimedia Data Categories.

Applying this XSLT style sheet to the XMT-O document shown in 10.1 allows to obtain the MLIF document shown in .

```
<?xml version="1.0" encoding="utf-8"?>
<struct xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:noNamespaceSchemaLocation="file:/MLIF/GMT.xsd"
        type="mlif">
   <struct type="MetaData">
      <feat type="approach">dublinCoreApproach</feat>
   </struct>
   <struct type="MultiLingualEntry">
      <struct type="MultiLingualUnit">
         <struct type="MonoLingualEntry">
            <struct type="MonoLingualSegment">
               <feat type="language" xml:lang="fr"/>
               <feat text="&#34;Présentation des différents chefs-lieux de
Lorraine&#34;; "/>
               <feat text="&#34;Présentation des différents chefs-lieux de
Lorraine&#34;; "/>
               <feat text="&#34;Cette présentation a été développée en XMT-
O ...&#34;; "/>
               <feat text="&#34;Cette présentation a été développée en XMT-
O ...&#34;; "/>
               <feat text="&#34;Pour découvrir une ville, cliquer sur son
nom.&#34;; "/>
               <feat text="&#34;Epinal&#34;; "/>
               <feat text="&#34;Nancy&#34;; "/>
               <feat text="&#34;Bar le Duc&#34;; "/>
               <feat text="&#34;Metz&#34;; "/>
               <feat text="&#34;Menu &#34;; "/>
            </struct>
         </struct>
      </struct>
   </struct>
</struct>
```

Figure 20: Transforming XMT-O documents into MLIF documents (canonical form).

Even if MLIF is not completely defined yet, it is possible to work on XLIFF, TMX, TBX and i18n formats. A client / server Cocoon's based application and a local Java-based applications have been developed (see section 7). Both applications allow users to perform XSLT-based transformations on MLIF, XLIFF, TMX and i18n formats.

Section X in the Appendix, show two XSLT style sheets allowing to transform MLIF documents into XLIFF documents, and XLIFF documents into MLIF documents, respectively.

7 SOFTWARE APPLICATIONS

In the framework of activity 2.1, we have built three software applications:

1. A web client / server applications allowing users to visualize, manage and combine "Data Categories". This application has been developed in the framework of the SYNTAX project. http://syntax.loria.fr
2. A web client / server Cocoon-based application allowing users to transform documents from: XLIFF to TMX, TMX to XLIFF, i18n to TMX, i18n to XLIFF, TMX to i18n, XLIFF to i18n, XLIFF to MLIF. http://loreley.loria.fr:8888
3. A local Java-based application allowing users to transform documents in the same way. This application may also be downloaded from http://loreley.loria.fr:8888

Each application has a user manual. All manuals will be included, as separate documents, in the global WP 2 progress report.

8 CONCLUSION.

We have presented MLIF (Multilingual Information Framework). MLIF is a high-level model for describing multilingual data. MLIF can be used in a wide range of possible applications in the translation/localization process of several multimedia domains. In order to experiment MLIF in multimedia, we have proposed to use MLIF and Data Categories for specifying RAMO in ITEA "Jules Verne" Project: dealing with terminology and multilingual content in the European interactive digital broadcast industry for content creation. Currently, we are specifying the MLIF API through which a wide variety of formats (i.e. TBX, TMX, XLIFF, etc.) may be integrated to the core MLIF structure.

9 REFERENCES AND BIBLIOGRAPHY.

[1] ISO TC37 SC4 “Language Resources Management”. http://tc37sc4.org/

[2] M. Kim & S. Wood
“XMT: MPEG-4 Textual Format for Cross-Standard Interoperability”
IBM T.J Watson Research
19 Skyline Drive
Hawthorne, NY. 10532 USA

[3] LISA. http://www.lisa.org

[4] OASIS. http://www.oasis-open.org/home/index.php

[5] W3C. http://www.w3c.org

[6] ISO. http://www.iso.org

[7] OSCAR. http://www.lisa.org/oscar

[8] TermBase eXchange Standard. http://www.lisa.org/tbx

[9] Translation Memory eXchange Standard. http://www.lisa.org/tmx

[10] XML Localisation Interchange File Format.
http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xliff

[11] Timed Text. http://www.w3.org/AudioVideo/TT/

[12] W3C Internationalization (I18N) Activity. http://www.w3.org/International/

[13] Terminological Markup Framework. http://www.loria.fr/projets/TMF/tmf.html

[14] Computer Applications in Terminology – Terminological Markup Framework.
ISO 16642.
http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=32347&ICS1=1&ICS2=20&ICS3=

[15] Computer Applications in Terminology - Data Categories.
ISO 12620.
http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=2517&ICS1=1&ICS2=20&ICS3=

[16] Language Codes.
ISO 639-2.
http://www.loc.gov/standards/iso639-2/

[17] Specification and Standardization of Data Elements.
ISO 11179-2.
http://www.iso.ch/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=2317&ICS1=35&ICS2=40&ICS3=

[18] SYNTAX Help Manual.

[19] Standards-based Access to multilingual Lexicons and Terminologies.
http://www.loria.fr/projets/SALT/saltsite.html

[20] Open Lexicon Interchange Format. http://www.olif.net/

[21] MAchine-Readable Terminology Interchange Format. ISO 12200:1999.

[22] Introduction to i18n.
Tomohiro Kubota.
http://www.debian.org/doc/manuals/intro-i18n/

[23] S. Renault, B. Gâteau, D. Khadraoui, F. Meinkhön, S. Cruz-Lara and T. Zaharia
"Reactive and Adaptative Multimedia Objects"
RAMO Model Proposal v1.5.
WP 1.3 Repport.
ITEA Project "Jules Verne".
December, 2003

[24] S. Renault, F. Meinköhn, D. Khadraoui, and P. Blandin.
"Reactive and adaptative multimedia object approach for interactive and immersive applications".
Proc. IEEE ICTTA 2004.
Damascus, Syria, 2004.

[25] S. Cruz-Lara, L. Romary, Ph. Sébire
"Using the concept of Data Category Registry for specifying RAMO structure"
ITEA "Jules Verne" Project.
Project Management Team.
Luxembourg.
November, 2003.

[26] S. Cruz-Lara, L. Romary, Ph. Sébire
"Managing Terminology and Multilingual Content within RAMO"
ITEA "Jules Verne" Project.
Project Management Team.
Luxembourg.
November, 2003.

[27] Synchronized Multimedia Integration Language (SMIL 2.0)
http://www.w3.org/TR/smil20/

[28] F. Alexandre.
Université Nancy 2.
Institut Universitaire de Technologie "Nancy-Charlemagne"
Licence Professionnelle "Concepteur / Intégrateur de sites Internet / Intranet".
Année Universitaire 2003-2004.

[29] http://syntax.loria.fr

10 APENDIX

10.1 The XMT_{french} document cited in 5.4

```
<?xml version="1.0" encoding="UTF-8"?>
<XMT-O xmlns="urn:mpeg:mpeg4:xmto:schema:2002"
xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:mpeg:mpeg4:xmto:schema:2002 xmt-o.xsd">
    <head>
        <meta content="fr" name="systemLanguage"/>
        <layout metrics="pixel" type="xmt/xmt-basic-layout">
            <topLayout backgroundColor="#bddcf2" height="600" width="800">
                <region id="image" size="800 600" translation="0 0"
                        z-index="1"/>
                <region id="titre" size="800 500" translation="-15 250"
                        z-index="2"/>
                <region id="explication" size="800 500"
                        translation="0 -250" z-index="3"/>
                <region id="clique_epinal" size="50 25"
                        translation="30 -130" z-index="6"/>
                <region id="clique_nancy" size="50 25"
                        translation="-3 0" z-index="6"/>
                <region id="clique_bar_le_duc" size="90 18"
                        translation="-195 20" z-index="6"/>
                <region id="clique_metz" size="50 18"
                        translation="-10 90" z-index="6"/>
                <region id="cadre1" size="800 600"
                        translation="30 30" z-index="4"/>
                <region id="cadre2" size="800 600"
                        translation="30 30" z-index="0"/>
                <region id="anim1" size="800 600"
                        translation="30 30" z-index="7"/>
                <region id="retour" size="800 500"
                        translation="365 -290" z-index="3"/>
            </topLayout>
        </layout>
        <defs>
            <fontStyle family="&amp;quot;SERIF&amp;quot;" id="defFont20"
                       justify="MIDDLE; MIDDLE"
                       size="30" style="BOLDITALIC"/>
            <fontStyle family="&amp;quot;SANS&amp;quot;" id="defFont16"
                       justify="MIDDLE; MIDDLE"
                       size="15" style="ITALIC"/>
        </defs>
        <meta
            content="(c)Copyright LORIA - INRIA Lorraine 2003-2004"
            name="copyright"/>
    </head>
    <body>
        <par>
            <audio dur="3s" src="Sons/debut.mp3"/>
            <audio begin="4s" dur="indefinite" src="Sons/fond_menu.mp3"/>
            <img dur="indefinite" id="carte" region="image"
                 src="Images/carte_lorraine.jpg"/>
```

```
<string dur="800s" region="titre"
     textLines="&quot;Présentation des différents chefs-lieux
                de Lorraine&quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="200  -200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string dur="indefinite" region="titre"
     textLines="&quot;Présentation des différents chefs-lieux
                de Lorraine&quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="-200  -200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string dur="5s" region="explication"
        textLines="&quot;Cette présentation a été développée en
                   XMT-O ...&quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="-200  200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string dur="5s" region="explication"
        textLines="&quot;Cette présentation a été développée en
                   XMT-O ...&quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="200  200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string begin="6s" dur="12s" region="explication"
        textLines="&quot;Pour découvrir une ville, cliquer sur
                   son nom.&quot;; ">
    <transformation>
        <animateMotion begin="0s" dur="3s" from="0  -50"
                       to="0 0"/>
    </transformation>
    <transformation>
        <animateMotion begin="9s" dur="3s" from="0  0"
                       to="0 -50"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<a href="anim_epinal.mp4">
    <string dur="indefinite" id="epinal" region="clique_epinal"
            textLines="&quot;Epinal&quot;; ">
        <material color="black"/>
        <fontStyle family="&quot;SERIF&quot;" justify="MIDDLE;
                   MIDDLE" size="20" style="BOLD"/>
    </string>
```

```
            </a>
            <a href="anim_nancy.mp4">
                <string dur="indefinite" id="nancy" region="clique_nancy"
                        textLines="&amp;quot;Nancy&amp;quot;; ">
                    <material color="black"/>
                    <fontStyle family="&amp;quot;SERIF&amp;quot;" justify="MIDDLE;
                               MIDDLE" size="20" style="BOLD"/>
                </string>
            </a>
            <a href="anim_bar_le_duc.mp4">
                <string dur="indefinite" id="bar_le_duc"
                        region="clique_bar_le_duc"
                        textLines="&amp;quot;Bar le Duc&amp;quot;; ">
                    <material color="black"/>
                    <fontStyle family="&amp;quot;SERIF&amp;quot;" justify="MIDDLE;
                               MIDDLE" size="20" style="BOLD"/>
                </string>
            </a>
            <a href="anim_metz.mp4">
                <string dur="indefinite" id="metz" region="clique_metz"
                        textLines="&amp;quot;Metz&amp;quot;; ">
                    <material color="black"/>
                    <fontStyle family="&amp;quot;SERIF&amp;quot;" justify="MIDDLE;
                               MIDDLE" size="20" style="BOLD"/>
                </string>
            </a>
            <a href="menu.mp4">
                <string begin="0s" dur="500s" region="retour"
                        textLines="&amp;quot;Menu &amp;quot;; ">
                    <use xlink:href="#defFont20"/>
                    <material color="yellow"/>
                </string>
            </a>
        </par>
    </body>
</XMT-O>
```

10.2 The $XMT_{english}$ document cited in 5.4

```
<?xml version="1.0" encoding="UTF-8"?>
<XMT-O xmlns="urn:mpeg:mpeg4:xmto:schema:2002"
xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:mpeg:mpeg4:xmto:schema:2002 xmt-o.xsd">
    <head>
        <meta content="en" name="systemLanguage"/>
        <layout metrics="pixel" type="xmt/xmt-basic-layout">
            <topLayout backgroundColor="#bddcf2" height="600" width="800">
                <region id="image" size="800 600" translation="0 0"
                        z-index="1"/>
                <region id="titre" size="800 500" translation="-15 250"
                        z-index="2"/>
                <region id="explication" size="800 500"
                        translation="0 -250" z-index="3"/>
                <region id="clique_epinal" size="50 25"
                        translation="30 -130" z-index="6"/>
                <region id="clique_nancy" size="50 25"
                        translation="-3 0" z-index="6"/>
                <region id="clique_bar_le_duc" size="90 18"
                        translation="-195 20" z-index="6"/>
                <region id="clique_metz" size="50 18"
                        translation="-10 90" z-index="6"/>
                <region id="cadre1" size="800 600"
                        translation="30 30" z-index="4"/>
                <region id="cadre2" size="800 600"
                        translation="30 30" z-index="0"/>
                <region id="anim1" size="800 600"
                        translation="30 30" z-index="7"/>
                <region id="retour" size="800 500"
                        translation="365 -290" z-index="3"/>
            </topLayout>
        </layout>
        <defs>
            <fontStyle family="&amp;quot;SERIF&amp;quot;" id="defFont20"
                       justify="MIDDLE; MIDDLE"
                       size="30" style="BOLDITALIC"/>
            <fontStyle family="&amp;quot;SANS&amp;quot;" id="defFont16"
                       justify="MIDDLE; MIDDLE"
                       size="15" style="ITALIC"/>
        </defs>
        <meta
            content="(c)Copyright LORIA - INRIA Lorraine 2003-2004"
            name="copyright"/>
    </head>
    <body>
        <par>
            <audio dur="3s" src="Sons/debut.mp3"/>
            <audio begin="4s" dur="indefinite" src="Sons/fond_menu.mp3"/>
            <img dur="indefinite" id="carte" region="image"
                 src="Images/carte_lorraine.jpg"/>
            <string dur="800s" region="titre"
                 textLines="&amp;quot;Presentation of Capital Cities of
                            Lorraine &amp;quot;; ">
                <transformation>
                    <animateMotion begin="0" dur="2s" from="200  -200"
                                   to="0 0"/>
```

```
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string dur="indefinite" region="titre"
     textLines="&quot;Presentation of Capital Cities of
                Lorraine &quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="-200  -200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string dur="5s" region="explication"
        textLines="&quot;This presentation has been developed
                   in XMT-O ...&quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="-200  200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string dur="5s" region="explication"
        textLines="&quot;This presentation has been developed
                   in XMT-O ...&quot;; ">
    <transformation>
        <animateMotion begin="0" dur="2s" from="200  200"
                       to="0 0"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<string begin="6s" dur="12s" region="explication"
        textLines="&quot;To discover a city, click on it’s
                   name&quot;; ">
    <transformation>
        <animateMotion begin="0s" dur="3s" from="0  -50"
                       to="0 0"/>
    </transformation>
    <transformation>
        <animateMotion begin="9s" dur="3s" from="0  0"
                       to="0 -50"/>
    </transformation>
    <use xlink:href="#defFont20"/>
    <material color="black"/>
</string>
<a href="anim_epinal.mp4">
    <string dur="indefinite" id="epinal" region="clique_epinal"
            textLines="&quot;Epinal&quot;; ">
        <material color="black"/>
        <fontStyle family="&quot;SERIF&quot;" justify="MIDDLE;
                   MIDDLE" size="20" style="BOLD"/>
    </string>
</a>
<a href="anim_nancy.mp4">
    <string dur="indefinite" id="nancy" region="clique_nancy"
            textLines="&quot;Nancy&quot;; ">
        <material color="black"/>
        <fontStyle family="&quot;SERIF&quot;" justify="MIDDLE;
```

```
                        MIDDLE" size="20" style="BOLD"/>
                </string>
            </a>
            <a href="anim_bar_le_duc.mp4">
                <string dur="indefinite" id="bar_le_duc"
                        region="clique_bar_le_duc"
                        textLines="&quot;Bar le Duc&quot;; ">
                    <material color="black"/>
                    <fontStyle family="&quot;SERIF&quot;" justify="MIDDLE;
                        MIDDLE" size="20" style="BOLD"/>
                </string>
            </a>
            <a href="anim_metz.mp4">
                <string dur="indefinite" id="metz" region="clique_metz"
                        textLines="&quot;Metz&quot;; ">
                    <material color="black"/>
                    <fontStyle family="&quot;SERIF&quot;" justify="MIDDLE;
                        MIDDLE" size="20" style="BOLD"/>
                </string>
            </a>
            <a href="menu.mp4">
                <string begin="0s" dur="500s" region="retour"
                        textLines="&quot;Menu &quot;; ">
                    <use xlink:href="#defFont20"/>
                    <material color="yellow"/>
                </string>
            </a>
        </par>
    </body>
</XMT-O>
```

10.3 The XLIFF file used by the MPEG-4 Multimedia presentation developed by INT for the ITEA Symposium (Seville, Spain 2004)

```
<?xml version="1.0" encoding="UTF-8"?>
<xliff>
  <file>
    <header/>
    <body>
      <group>
        <trans-unit id="france">
          <source xml:lang="en"> Although ultimately a victor in World Wars I and II,
            France suffered extensive losses in its empire, wealth, manpower, and rank
            as a dominant nation-state. Nevertheless, France today is one of the most
            modern countries in the world and is a leader among European nations. Since
            1958, it has constructed a presidential democracy resistant to the
            instabilities experienced in earlier parliamentary democracies. In recent
            years, its reconciliation and cooperation with Germany have proved central
            to the economic integration of Europe, including the introduction of a
            common exchange currency, the euro, in January 1999. At present, France is
            at the forefront of European states seeking to exploit the momentum of
            monetary union to advance the creation of a more unified and capable
            European defense and security apparatus. </source>
          <target xml:lang="fr"> Même si elle est finalement sortie victorieuse de la
            première et de la deuxième guerre mondiale, la France a souffert des pertes
            considérables dans son empire, dans ses richesses, dans sa population et
            dans son rang de nation dominante. Cependant, la France reste aujourd'hui un
            des pays les plus modernes du monde et c'est encore un des conducteurs des
            pays européens. Depuis 1958, elle a construit une démocratie présidentielle
            qui a su résister aux instabilités subies par les démocraties parlementaires
            précédentes. Ces dernières années, sa réconciliation et sa coopération avec
            l'Allemagne se sont avérées déterminantes dans l'intégration économique
            européenne, incluant la création de la monnaie unique européenne, l'Euro, en
            janvier 1999. Actuellement, la France est au premier rang des pays européens
            qui essayent, dans la dynamique suivant la création de l'Euro, d'aller vers
            la construction d'une défense et d'une sécurité commune aux pays européens. </target>
        </trans-unit>
        <trans-unit id="spain">
          <source xml:lang="en"> Spain's powerful world empire of the 16th and 17th
            centuries ultimately yielded command of the seas to England. Subsequent
            failure to embrace the mercantile and industrial revolutions caused the
            country to fall behind Britain, France, and Germany in economic and
            political power. Spain remained neutral in World Wars I and II, but suffered
            through a devastating civil war (1936-39). In the second half of the 20th
            century, Spain has played a catch-up role in the western international
            community; it joined the EU in 1986. Continuing challenges include are
            Basque Fatherland and Liberty (ETA) terrorism and further reductions in
            unemployment. </source>
```

<target xml:lang="fr"> Le puissant empire espagnol du 16ème et du 17ème siècles a finalement cédé le contrôle des mers à la Grande Bretagne. L'incapacité de suivre les révolutions mercantile et industrielle ont fait que l'Espagne se trouve désormais derrière la Grande Bretagne, la France et l'Allemagne dans les domaines politique et économique. L'Espagne, qui a su rester neutre pendant la première et la deuxième guerre mondiale, a cependant souffert d'une guerre civile dévastatrice entre 1936 et 1939. Dans la deuxième moitié du 20ème siècle, l'Espagne a joué un rôle important dans la communauté internationale occidentale. Elle à rejoint l'Union Européenne en 1986. Parmi les défis importants de l'Espagne d'aujourd'hui on trouve le problème des indépendantistes basques et des terroristes de l'ETA ainsi que le problème du chômage. </target>
</trans-unit>
<trans-unit id="italy">
<source xml:lang="en"> Italy became a nation-state in 1861 when the city-states of the peninsula, along with Sardinia and Sicily, were united under King Victor EMMANUEL II. An era of parliamentary government came to a close in the early 1920s when Benito MUSSOLINI established a Fascist dictatorship. His disastrous alliance with Nazi Germany led to Italy's defeat in World War II. A democratic republic replaced the monarchy in 1946 and economic revival followed. Italy was a charter member of NATO and the European Economic Community (EEC). It has been at the forefront of European economic and political unification, joining the European Monetary Union in 1999. Persistent problems include illegal immigration, organized crime, corruption, high unemployment, sluggish economic growth, and the low incomes and technical standards of southern Italy compared with the prosperous north. </source>
<target xml:lang="fr"> L'Italie est devenue un état-nation en 1861 lorsque les cités-états de la peninsule, avec la Sardaigne et la Sicile, ont été unifié sous le Roi VICTOR EMMANUEL II. Une période de gouvernement parlementaire a fini lorsqu'au début des années 1920, quand Benito MUSSOLINI établit une dictature fasciste. Sa désastreuse alliance avec l'Allemagne Nazie a provoqué la défaite de l'Italie lors de la deuxième guerre mondiale. Une république démocratique a, en 1946, remplacé la monarchie et une période de renouveau économique s'en est suivie. L'Italie est l'un des membres fondateurs de l'OTAN et de la Communauté Economique Européenne. Auu premier rang des pays européens oeuvrant pour l'unification de la politique et de l'économie de l'Union Europpénne, elle a rejoint l'Union Monétaire Européenne en 1999. Les problèmes actuels de l'Italie incluent, l'immigration ilégale, le crime organisé, la corruption, le haut niveau de chômage, une croissance économique inconstante et les grands differences existant entre le nord et le sud de la peninsule. </target>
</trans-unit>
<trans-unit id="germany">
<source xml:lang="en"> As Europe's largest economy and most populous nation, Germany remains a key member of the continent's economic, political, and defense organizations. European power struggles immersed Germany in two devastating World Wars in the first half of the 20th century and left the

country occupied by the victorious Allied powers of the US, UK, France, and the Soviet Union in 1945. With the advent of the Cold War, two German states were formed in 1949: the western Federal Republic of Germany (FRG) and the eastern German Democratic Republic (GDR). The democratic FRG embedded itself in key Western economic and security organizations, the EC, which became the EU, and NATO, while the Communist GDR was on the front line of the Soviet-led Warsaw Pact. The decline of the USSR and the end of the Cold War allowed for German unification in 1990. Since then, Germany has expended considerable funds to bring Eastern productivity and wages up to Western standards. In January 1999, Germany and 10 other EU countries introduced a common European exchange currency, the euro. </source>
<target xml:lang="fr"> Entant que plus importante économie européenne et nation la plus peuplée, l'Allemagne reste un membre clé des organisations économiques, politique et de défense du continent européen. Les rivalités entre les puissants états européens ont conduit l'Allemagne vers deux dévastatrices guerres mondiales dans la première motié du 20ème siècle et ont laissé le pays occupé par les puissances alliées victorieuses : les Etats Unis, la Grande Bretagne, la France et l'Union Soviétique en 1945. Avec l'apparition de la Guerre Froide, deux états allemands ont été fondés en 1949 : la République Fédéral d'Allemagne de l'Ouest (RFA) et la République Démocratique d'Allemagne de l'Est (RDA). La RFA démocratique a tout de été un membre clé de l'économie et de la sécurité occidentales, dont la Communauté Economique Européenne qui allait ensuite devenir l'Union Européenne, et l'OTAN. La RDA communiste se plaçant au premier rang des nations du Pacte de Varsovie. La fin de la Guerre Froide et la chute de l'URSS ont permis l'unification de l'Allemagne en 1990. Depuis lors, l'Allemagne a dépensé des sommes considérables pour adapter l'économie de sa partie Est aux standards occidentaux. En janvier 1999, l'Allemagne et dix autres pays de l'Union Européenne, donné naissance à la monnaie unique européenne, l'Euro. </target>
</trans-unit>
<trans-unit id="UK">
<source xml:lang="en"> Great Britain, the dominant industrial and maritime power of the 19th century, played a leading role in developing parliamentary democracy and in advancing literature and science. At its zenith, the British Empire stretched over one-fourth of the earth's surface. The first half of the 20th century saw the UK's strength seriously depleted in two World Wars. The second half witnessed the dismantling of the Empire and the UK rebuilding itself into a modern and prosperous European nation. As one of five permanent members of the UN Security Council, a founding member of NATO, and of the Commonwealth, the UK pursues a global approach to foreign policy; it currently is weighing the degree of its integration with continental Europe. A member of the EU, it chose to remain outside the European Monetary Union for the time being. Constitutional reform is also a significant issue in the UK. The Scottish Parliament, the National Assembly for Wales, and the Northern Ireland Assembly were established in 1999, but the latter is suspended due to bickering over the peace process. </source>
<target xml:lang="fr"> La Grande Bretagne, la plus importante puissance

industrielle et maritime du 19ème siècle, a joué un rôle majeur dans le développement de la démocracie parlementaire, de la littérature et de la sceince. A son apogée, l'Empire Britannique possédait un quart de la surface de la Terre. Dans la première motié du 20ème siècle on a assisté à une diminution considérable de la puissance de cet empire à cause des deux guerres mondiales. Dans la deuxième moitié du 20ème siècle on a vu le démantèlement de l'empire mais aussi la naissance d'une nation européenne moderne et prospère. Entant qu'un des cinq membres permanents du Conseil de Sécurité de l'ONU, membre fondateur de l'OTAN et du Commonwealth, le Royaume-Uni a une vision globale de la politique étrangère et accroît son degrée d'intégration avec l'Europe continentale. Membre de l'Union Européenne, le Royaume-Uni a choisi, pour le meoment, de ne pas participer à l'Union Monétaire Européenne. Aussi, existe-t-il actuellement une importante réforme constitutionnelle. Le parlement Ecossais, l'Assemblée Nationale Galloise et l'Assemblée d'Irlande du Nord ont vu le jour en 1999, mais cette dernière a vu ses fonctions suspendues en raison des problèmes liés à l'interruption du processus de paix.</target>
</trans-unit>
<trans-unit id="canada">
<source xml:lang="en"> A land of vast distances and rich natural resources, Canada became a self-governing dominion in 1867 while retaining ties to the British crown. Economically and technologically the nation has developed in parallel with the US, its neighbor to the south across an unfortified border. Its paramount political problem continues to be the relationship of the province of Quebec, with its French-speaking residents and unique culture, to the remainder of the country. </source>
<target xml:lang="fr"> Terre de vastes étendues et riche en ressources naturelles, le Canada est devenu indépendant en 1867 mais il est toujours attaché à la Couronne Britannique. D'un point de vue technologique et économique la nation s'est développé en parallèle avec les Etats-Unis, son voisin du Sud le long d'une frontière non-fortifiée. Son plus important problème politique continue d'être les relations entre la province de Québec, ses résidents francophones et leur culture unique, avec le reste du pays.
</target>
</trans-unit>
<trans-unit id="USA">
<source xml:lang="en"> Britain's American colonies broke with the mother country in 1776 and were recognized as the new nation of the United States of America following the Treaty of Paris in 1783. During the 19th and 20th centuries, 37 new states were added to the original 13 as the nation expanded across the North American continent and acquired a number of overseas possessions. The two most traumatic experiences in the nation's history were the Civil War (1861-65) and the Great Depression of the 1930s. Buoyed by victories in World Wars I and II and the end of the Cold War in 1991, the US remains the world's most powerful nation state. The economy is marked by steady growth, low unemployment and inflation, and rapid advances in technology. </source>
<target xml:lang="fr"> Les colonies britanniques ont rompu avec la mère patrie

en 1776 et ont été reconnus comme la nation des Etats-Unis d'Amérique suivant le traité de Paris en 1783. Durant le 19ème et le 20ème siècles, 37 nouveaux états ont été ajoutés aux 13 états d'origine, alors que le pays s'étendait dans l'Amérique du Nord et acquérait des territoires outre-mer. Les deux expériences les plus traumatisantes dans l'histoire de la nation ont été la guerre civile (1861-65) et la grande dépression des années 30. Reconfortés par des victoires dans la première et la deuxième guerre mondiale et par la fin de la guerre froide en 1991, les Etats-Unis restent la nation la plus puissante. L'économie est marquée par une croissance régulière, un chômage et une inflation réduits, et par des avances technologiques rapides.</target>
</trans-unit>
<trans-unit id="brazil">
<source xml:lang="en"> Following three centuries under the rule of Portugal, Brazil became an independent nation in 1822. By far the largest and most populous country in South America, Brazil has overcome more than half a century of military intervention in the governance of the country to pursue industrial and agricultural growth and development of the interior. Exploiting vast natural resources and a large labor pool, Brazil is today South America's leading economic power and a regional leader. Highly unequal income distribution remains a pressing problem. </source>
<target xml:lang="fr"> Après trois siècles de domination portugaise, le Brésil est devenu indépendant en 1822. De loin, le pays le plus grand et le plus peuplé d'Amérique du Sud, le Brésil a surmonté plus d'un demi-siècle d'interventions militaires pour gouverner le pays, pour accroître sa croissance et son développement dans les domaines industriel et agricole. Ayant une quantité importante de ressources naturelles et un réservoir de main d'oeuvre considérable, le Brésil est aujourd'hui un leader régional et aussi la première puissance économique de la Amérique du Sud. La répartition, très inégale, des richesses reste un problème important. </target>
</trans-unit>
<trans-unit id="argentina">
<source xml:lang="en"> Following independence from Spain in 1816, Argentina experienced periods of internal political conflict between conservatives and liberals and between civilian and military factions. After World War II, a long period of Peronist authoritarian rule and interference in subsequent governments was followed by a military junta that took power in 1976. Democracy returned in 1983, and numerous elections since then have underscored Argentina's progress in democratic consolidation. </source>
<target xml:lang="fr"> Après son indépendance de l'Espagne en 1816, l'Argentine à vécu plusieurs années de conflit entre les conservateurs et les libéraux mais aussi entre les civiles et les militaires. Après la deuxième guerre mondiale, une longue période de gouvernement des « Peronistes » ainsi que les interventions répétées de ceux-ci dans les gouvernements subséquents, une junte militaire pris le pouvoir en 1976. La démocratie est retournée en 1983, et des nombreuses élections depuis lors, ont permis de consolider les progrès de l'Argentine en matière de démocratie. La répartition, très inégale des richesses, reste un problème considérable. </target>

</trans-unit>
<trans-unit id="algeria">
<source xml:lang="en"> After a century of rule by France, and in the wake of 1948 elections rigged by French colonists to reverse the sweeping victory of a Muslim political party in 1947, Algerians fought through the 1950s to achieve independence in 1962. Algeria's primary political party, the National Liberation Front (FLN), has dominated politics ever since. Many Algerians in the subsequent generation were not satisified, however, and moved to counter the FLN's centrality in Algerian politics. The surprising first round success of the fundamentalist Islamic Salvation Front (FIS) in the December 1991 balloting spurred the army to intervene and postpone the second round of elections to prevent an extremist-led government from assuming power. The Algerian army began a crack down on the FIS, that resulted in a continuous low-grade civil conflict between Islamic activists and the secular state apparatus. The government later allowed elections featuring pro-government and moderate religiously-based parties, but did not appease the activists who progressively widened their attacks. Operations by the activists and the army resulted in nearly 100,000 deaths during the decade-long conflict. The government gained the uppper hand by the mid-1990s and FIS's armed wing, the Islamic Salvation Army, disbanded in January 2000. Many armed militants of other groups surrendered under an amnesty program designed to promote national reconciliation, but small numbers of armed militants persist in confronting government forces and conducting ambushes and occasional attacks on villages. Issues facing the winner of the April 2004 presidential election include Berber unrest, large-scale unemployment, a shortage of housing, the presence of a group in the southern regions of the country that kidnapped European tourists in 2003, as well as the need to diversify Algeria's petroleum-based economy. Algeria assumed a two-year seat on the UN Security Council in January 2004. </source>
<target xml:lang="fr"> Après un siècle de domination française, et suite à l'organisation bâclée des élections de 1948 où les colons français ont tenté d'effacer la victoire d'un parti musulman en 1947, les Algériens ont combattu durant les années 50 pour finalement obtenir leur indépendance en 1962. Le principal parti politique Algérien, le "Front de Libération Nationale" (FLN), a dominé la vie politique depuis lors. Cependant, de plus en plus d'algériens mécontents ont mis en cause la place du FLN dans la politique algérienne. Le surprenant succès au premier tour des élections de décembre 1991 du "Front Islamique du Salut" (FIS) a provoqué l'intervention de l'armée pour empêcher que des extrémistes prennent le pouvoir. Les pressions exercées par l'armée sur le FIS ont par la suite dégénéré en un conflit civil entre les activistes islamiques et l'état. Des élections où des partis pro gouvernement et des partis religieux modérés ont participé furent organisées mais celles-ci n'ont pas apaisé les activistes islamiques qui ont continué leurs attaques. Durant une décennie, les opérations des militaires et des activistes islamiques ont provoqué la mort de plus de 100,000 personnes. Le gouvernement a finalement pris le dessus au milieu des années 1990 et l'aile armée du FIS à été enfin détruite en janvier 2000. Plusieurs militants armés d'autres groupes se sont par la suite rendus dans

le cadre du programme d'amnistie qui a été lancé pour promouvoir la réconciliation nationale, mais plusieurs groupuscules armés subsistent qui continuent à provoquer les autorités gouvernementales via des embuscades et des attaques contre des villages. Parmi les défis qui attendent le gagnant de l'élection présidentielle d'avril 2004, on peut citer le problème Berbère, le degré élevé de chômage, le manque de logements, la présence de groupes terroristes dans le sud du pays (qui ont kidnappé des touristes européens en 2003), ainsi que le besoin de diversifier une économie largement basée sur le pétrole. L'Algérie occupe pour deux ans un siège au Conseil de Sécurité de l'ONU depuis janvier 2004. </target>
</trans-unit>
<trans-unit id="congo">
<source xml:lang="en"> Since 1997, the Democratic Republic of the Congo (DROC; formerly called Zaire) has been rent by ethnic strife and civil war, touched off by a massive inflow in 1994 of refugees from the fighting in Rwanda and Burundi. The government of former president MOBUTU Sese Seko was toppled by a rebellion led by Laurent KABILA in May 1997; his regime was subsequently challenged by a Rwanda- and Uganda-backed rebellion in August 1998. Troops from Zimbabwe, Angola, Namibia, Chad, and Sudan intervened to support the Kinshasa regime. A cease-fire was signed on 10 July 1999 by the DROC, Zimbabwe, Angola, Uganda, Namibia, Rwanda, and Congolese armed rebel groups, but sporadic fighting continued. KABILA was assassinated on 16 January 2001 and his son Joseph KABILA was named head of state ten days later. In October 2002, the new president was successful in getting occupying Rwandan forces to withdraw from eastern Congo; two months later, an agreement was signed by all remaining warring parties to end the fighting and set up a government of national unity. </source>
<target xml:lang="fr"> Depuis 1997, la République Démocratique du Congo (RDC - anciennement appelée Zaïre) a été le théâtre de troubles interethniques, d'une guerre civile et d'un afflux massif de réfugiés en 1994 en provenance du Rwanda et du Burundi. Le gouvernement de l'ancien Président MOBUTU Sese Seko a été renversé par une rébellion dirigée par Laurent KABILA en mai 1997. Son régime a été ensuite menacé par une rébellion supportée par le Rwanda et l'Uganda voisins en août 1998. Des troupes du Zimbabwe, de l'Angola, de la Namibie, du Tchad et du Soudan sont par la suite intervenues en faveur du régime de Kinshasa. Un cessez-le-feu a été signé le 10 juillet 1999 par la RDC, le Zimbabwe, l'Angola, l'Uganda, la Namibie, le Rwanda et par des groupes de rebelles congolais, mais des batailles sporadiques se sont poursuivies. KABILA a été assassiné le 16 janvier 2001 et son fils Joseph KABILA a été nommé à la tête de l'état dix jours plus tard. En octobre 2002, le nouveau Président a réussi à faire évacuer les régions de l'est du pays, par l'armée rwandaise. Deux mois plus tard, un accord a été signé par toutes les parties belligérantes pour arrêter les conflits et pour l'unité nationale. </target>
</trans-unit>
<trans-unit id="russia">
<source xml:lang="en"> Repeated devastating defeats of the Russian army in World War I led to widespread rioting in the major cities of the Russian Empire

and to the overthrow in 1917 of the 300-year old Romanov Dynasty. The Communists under Vladimir LENIN seized power soon after and formed the USSR. The brutal rule of Josef STALIN (1928-53) strengthened Russian dominance of the Soviet Union at a cost of tens of millions of lives. The Soviet economy and society stagnated in the following decades until General Secretary Mikhail GORBACHEV (1985-91) introduced glasnost (openness) and perestroika (restructuring) in an attempt to modernize Communism, but his initiatives inadvertently released forces that by December 1991 splintered the USSR into 15 independent republics. Since then, Russia has struggled in its efforts to build a democratic political system and market economy to replace the strict social, political, and economic controls of the Communist period. While some progress has been made on the economic front, recent years have seen a recentralization of power under Vladimir PUTIN and an erosion in nascent democratic institutions. A determined guerrilla conflict still plagues Russia in Chechnya. </source>

<target xml:lang="fr"> Les défaites répétées de l'armée russe durant la première guerre mondiale ont provoqué des émeutes dans les principales villes russes et ont conduit par la suite à la chute d'une dynastie vieille de 300 ans, la dynastie des Romanov. Les communistes menés par Vladimir LENIN ont pris le pouvoir et constitué l'URSS. L'autorité brutale de Josef STALIN (1928-1053) a renforcé la domination de la Russie sur l'URSS au prix de dizaines des millions de morts. L'économie et la société soviétiques ont stagné jusqu'à l'arrivée au pouvoir de Mikhail GORBATCHEV (1985-1991) avec l'introduction de la glasnost (transparence) et de la perestroika (restructuration) dans une tentative de moderniser le communisme. Ces initiatives cependant ont provoqué la scission de l'URSS en 15 républiques indépendantes. Depuis lors, la Russie s'efforce de consolider un système politique démocratique et une économie de marché pour remplacer les rigoureux contrôles sociaux, politiques et économiques de la période communiste. Même si des progrès ont été accomplis au niveau économique, ces dernières années ont montré une recentralisation du pouvoir autour de Vladimir POUTINE et une érosion persistante des institutions démocratiques. De plus, un conflit armé en Tchétchénie oppose la Russie à une guérilla très déterminée. </target>

</trans-unit>

<trans-unit id="australia">

<source xml:lang="en"> Aboriginal settlers arrived on the continent from Southeast Asia about 40,000 years before the first Europeans began exploration in the 17th century. No formal claims were made until 1770, when Capt. James COOK took possession in the name of Great Britain. Six colonies were created in the late 18th and 19th centuries; they federated and became the Commonwealth of Australia in 1901. The new country was able to take advantage of its natural resources in order to rapidly develop its agricultural and manufacturing industries and to make a major contribution to the British effort in World Wars I and II. Long-term concerns include pollution, particularly depletion of the ozone layer, and management and conservation of coastal areas, especially the Great Barrier Reef. A referendum to change Australia's status, from a commonwealth headed by the British monarch to a republic, was defeated in 1999. </source>

<target xml:lang="fr"> Les premiers aborigènes, en provenance de l'Asie du Sud-est, sont arrivés sur le continent australien 40,000 ans avant l'arrivée des premiers explorateurs européens au 17ème siècle. Aucune démarche formelle n'a cependant été faite jusqu'en 1770 quand le capitaine James COOK pris la possession de l'Australie au nom de la Grande-Bretagne. Six colonies ont été créées à la fin du 18ème et au 19ème siècles. Celles-ci se sont fédérées et ont fondé la Commonwealth d'Australie en 1901. Le nouveau pays a été capable de profiter de ses ressources naturelles et a rapidement développé ses industries agricoles et manufacturières qui ont largement contribué aux efforts britanniques durant la première et la deuxième guerre mondiales. Les défis a long terme incluent la pollution, la destruction de la couche d'ozone, la gestion et la conservation des côtes, tout particulièrement la grande barrière de corail. Un referendum pour le changement du statut de l'Australie, d'une Commonwealth dont la tête est la couronne britannique à une république, a été perdu en 1999. </target>
</trans-unit>
<trans-unit id="china">
<source xml:lang="en"> For centuries China stood as a leading civilization, outpacing the rest of the world in the arts and sciences. But in the 19th and early 20th centuries, China was beset by civil unrest, major famines, military defeats, and foreign occupation. After World War II, the Communists under MAO Zedong established a dictatorship that, while ensuring China's sovereignty, imposed strict controls over everyday life and cost the lives of tens of millions of people. After 1978, his successor DENG Xiaoping gradually introduced market-oriented reforms and decentralized economic decision making. Output quadrupled by 2000. Political controls remain tight while economic controls continue to be relaxed. </source>
<target xml:lang="fr"> Durant des siècles, la Chine a été une civilisation dirigeante, surclassant le reste du monde dans les arts et les sciences. Cependant, pendant le 19ème et le début du 20ème siècles, la Chine a été touchée par un malaise civil, des famines importantes, des défaites militaires et par l'occupation étrangère. Après la deuxième guerre mondiale, les communistes de MAO Zedong ont établi une dictature que, tout en assurant la souveraineté de la Chine, a imposé des contrôles très stricts sur la vie de tous les jours et qui a coûté la vie à des dizaines de millions de personnes. Après 1978, son successeur DENG Xiaoping a progressivement introduit des réformes orientées vers une économie de marché et vers un système de décisions décentralisé. Les résultats ont été multipliés par 2000. Les contrôles politiques restent rigoureux, alors que les contrôles économiques continuent d'être relâchés. </target>
</trans-unit>
<trans-unit id="indusValley">
<source xml:lang="en"> The Indus Valley civilization, one of the oldest in the world, dates back at least 5,000 years. Aryan tribes from the northwest invaded about 1500 B.C.; their merger with the earlier inhabitants created the classical Indian culture. Arab incursions starting in the 8th century and Turkish in the 12th were followed by European traders, beginning in the late 15th century. By the 19th century, Britain had assumed political

control of virtually all Indian lands. Nonviolent resistance to British colonialism under Mohandas GANDHI and Jawaharlal NEHRU led to independence in 1947. The subcontinent was divided into the secular state of India and the smaller Muslim state of Pakistan. A third war between the two countries in 1971 resulted in East Pakistan becoming the separate nation of Bangladesh. Fundamental concerns in India include the ongoing dispute with Pakistan over Kashmir, massive overpopulation, environmental degradation, extensive poverty, and ethnic and religious strife, all this despite impressive gains in economic investment and output. </source>
<target xml:lang="fr"> La civilisation de la vallée de l'Indus, une de plus anciennes du monde, date d'au moins 5,000 ans. Les tribus aryans du nord-ouest ont envahi la vallée de l'Indus en 1500 A.J; elles se sont mélangé aux habitants déjà installés pour former la culture indienne classique. Les incursions arabes ont commencé au 8ème siècle et les turques au 12ème siècle furent suivies par les européennes au 15ème siècle. Au 19ème siècle, la Grande-Bretagne pris virtuellement le contrôle de toutes les terres indiennes. La résistance non-violente de Mohandas GANDHI et de Jawaharlal NEHRU ont conduit à l'indépendance en 1947. Le sous-continent a été divisé en l'état séculaire d'Inde et un plus petit état musulman, le Pakistan. Une troisième guerre entre les deux pays en 1971 à provoqué la naissance d'un troisième état, le Bangladesh. Les défis importants de l'Inde incluent la dispute avec le Pakistan autour de la région de Cachemire, la surpopulation, la dégradation de l'environnement, l'avancé de la pauvreté, les troubles religieux et ethniques. Tout ceci, accompagné des progrès impressionnants dans le domaine économique et la productivité. </target>
</trans-unit>
</group>
</body>
</file>
</xliff>

10.4 An XSLT style sheet transforming XMT-O documents into MLIF documents

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">

    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
    <xsl:variable name="systemLanguage"
              select="/XMT-O/head/meta[@name='systemLanguage']/@content" />

    <xsl:template match="XMT-O">
        <struct xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:noNamespaceSchemaLocation="file:/MLIF/GMT.xsd" type="mlif">
            <xsl:call-template name="head"/>
            <xsl:call-template name="body"/>
        </struct>
    </xsl:template>

    <xsl:template name="head">
        <struct type="MetaData">
            <feat type="approach">dublinCoreApproach</feat>
        </struct>
    </xsl:template>

    <xsl:template name="body">
        <struct type="MultiLingualEntry">
            <struct type="MultiLingualUnit">
                <struct type="MonoLingualEntry">
                    <struct type="MonoLingualSegment">
                        <feat>
                            <xsl:attribute name="type">
                                language
                            </xsl:attribute>
                            <xsl:attribute name="xml:lang">
                                <xsl:value-of select="$systemLanguage"/>
                            </xsl:attribute>
                        </feat>
                        <xsl:for-each select="//string">
                            <xsl:call-template name="mlsegment"/>
                        </xsl:for-each>
                    </struct>
                </struct>
            </struct>
        </struct>
    </xsl:template>

    <xsl:template name="mlsegment">
        <feat>
            <xsl:attribute name="text">
                <xsl:value-of select="@textLines"/>
            </xsl:attribute>
        </feat>
    </xsl:template>

</xsl:stylesheet>
```

10.5 An XSLT style sheet transforming MLIF documents into XLIFF documents

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- This stylesheet transforms well formed MLIF documents into XLIFF
documents -->
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
    <xsl:template match="mlif">
        <xsl:variable name="SRCLang" select="@xml:lang"/>
        <!--XLIFF document starts here. -->
        <xliff version="1.1">
            <xsl:attribute name="xml:lang">
                <xsl:value-of select="$SRCLang"/>
            </xsl:attribute>
            <file>
                <!--default source language is inserted into XLIFF document
-->
                <xsl:attribute name="source-language">
                    <xsl:value-of select="$SRCLang"/>
                </xsl:attribute>
                <header/>
                <!--body element for XLIFF document starts here -->
                <body>
                    <!-- top level unit is serched from MLIF document -->
                    <xsl:for-each select="//multiLingualEntry">
                        <!--main template that will do all the
transformation work from MLIF to XLIFF -->
                        <xsl:call-template name="makeTransUnits"/>
                    </xsl:for-each>
                </body>
            </file>
            <!--XLIFF document ends here. -->
        </xliff>
    </xsl:template>
    <!--definition for main template for transformation from MLIF to XLIFF
-->
    <xsl:template name="makeTransUnits">
        <xsl:variable name="numOfUnits"
select="count(./multiLingualUnit)"/>
        <!--trans-units are inserted here -->
        <trans-unit>
            <!--calls to add the ID from MLIF document -->
            <xsl:call-template name="copyID"/>
            <!--calls to add source text from MLIF document -->
            <xsl:call-template name="copySource"/>
            <!--calls to add target text from MLIF document -->
            <xsl:call-template name="copyTarget"/>
            <!-- checks if more than one translations are available in MLIF
document. -->
            <xsl:for-each
select="./multiLingualUnit/monoLigualEntry[position()&gt;2]">
                <xsl:call-template name="copyAltTrans">
                    <!--xsl:with-param name="post1"
select="./multiLingualUnit/monoLigualEntry/position()"/-->
                </xsl:call-template>
            </xsl:for-each>
        </trans-unit>
    </xsl:template>
```

```
    <!-- This template copies id from MLIF document into XLIFF document, if
it exists -->
    <xsl:template name="copyID">
        <xsl:if test="@id">
            <xsl:attribute name="id">
                <xsl:value-of select="@id"/>
            </xsl:attribute>
        </xsl:if>
    </xsl:template>
    <!--This template copies the content of first monoLingualSeg into
trans-unit of XLIFF document as source element -->
    <xsl:template name="copySource">
        <source>
            <!-- it is assumed that first monoLingualEntry in MLIF document
is source -->
            <xsl:call-template name="copyLang">
                <xsl:with-param name="branchNum" select="1"/>
            </xsl:call-template>
            <xsl:value-of
select="./multiLingualUnit/monoLigualEntry[1]/monoLingualSeg/text"/>
        </source>
    </xsl:template>
    <!--This template copies the content of second monoLingualSeg into
trans-unit of XLIFF document as target element -->
    <xsl:template name="copyTarget">
        <target>
            <!-- this template is called to add the language of text from
monolingualSeg if it is there. -->
            <xsl:call-template name="copyLang">
                <!-- it is assumed that second monoLingualEntry in MLIF
document is target -->
                <xsl:with-param name="branchNum" select="2"/>
            </xsl:call-template>
            <xsl:value-of
select="./multiLingualUnit/monoLigualEntry[2]/monoLingualSeg/texte"/>
        </target>
    </xsl:template>
    <!-- This template search for "xml:lang" tag and insert into
corressponding element in XLIFF document -->
    <xsl:template name="copyLang">
        <xsl:param name="branchNum" select="1"/>
        <!-- Checks if xml:lang attribute exists -->
        <xsl:if
test="./multiLingualUnit/monoLigualEntry[$branchNum]/monoLingualSeg/@xml:la
ng">
            <xsl:attribute name="xml:lang">
                <xsl:value-of
select="./multiLingualUnit/monoLigualEntry[$branchNum]/monoLingualSeg/@xml:
lang"/>
            </xsl:attribute>
        </xsl:if>
    </xsl:template>
    <!-- This template is used to insert alternative translations into XIFF
document-->
    <xsl:template name="copyAltTrans">
        <xsl:variable name="post1" select="position()"/>
        <alt-trans>
            <target>
                <!-- Checks if xml:lang attribute exists -->
                <xsl:if test="./monoLingualSeg/@xml:lang">
                    <!--"xml:lang" attribute is added to target element -->
```

```
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="./monoLingualSeg/@xml:lang"/>
                    </xsl:attribute>
                </xsl:if>
                <!-- this copies text for target in alt-trans unit from
MLIF document -->
                <xsl:value-of select="./monoLingualSeg/text"/>
            </target>
        </alt-trans>
    </xsl:template>
    <!-- MLIFToXLIFF stylesheet ends here. -->
</xsl:stylesheet>
```

10.6 An XSLT style sheet transforming XLIFF documents into MLIF documents

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- This stylesheet transforms well formed XLIFF documents into MLIF
documents -->
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
    <!--looking for the starting of xliff document -->
    <xsl:template match="xliff">
        <!--sets the source and target languages from xliff document -->
        <xsl:variable name="SRCLang" select="file/@source-language"/>
        <xsl:variable name="TRGLang" select="file/@target-language"/>
        <!--inserts starting tags for MLIF document -->
        <mlif version="1.0">
            <!-- sets document language for MLIF-->
            <xsl:attribute name="xml:lang">
                <xsl:value-of select="$SRCLang"/>
            </xsl:attribute>
            <!-- sets header for MLIF document-->
            <mlifHeader>
                <docMetaData approach="dublinCoreApproach"/>
            </mlifHeader>
            <!--MLIF Body starts here -->
            <mlifBody>
                <!--calls template depending upon the structure of XLIF
document -->
                <xsl:for-each select="file/body/group/trans-unit">
                    <xsl:call-template name="mainTemplate">
                        <!-- passing the arguments to mainTemplate -->
                        <xsl:with-param name="SRCLang" select="$SRCLang"/>
                        <xsl:with-param name="TRGLang" select="$TRGLang"/>
                    </xsl:call-template>
                </xsl:for-each>
                <xsl:for-each select="file/body/trans-unit">
                    <xsl:call-template name="mainTemplate">
                        <!-- passing the arguments to mainTemplate -->
                        <xsl:with-param name="SRCLang" select="$SRCLang"/>
                        <xsl:with-param name="TRGLang" select="$TRGLang"/>
                    </xsl:call-template>
                </xsl:for-each>
                <!--MLIF Body ends -->
            </mlifBody>
            <!--MLIF document ends here -->
        </mlif>
    </xsl:template>
    <!-- main template starts here. it is called on the top level of XLIFF
document -->
    <xsl:template match="trans-unit" name="mainTemplate">
        <!-- parameters to this template. if no value is passed, it uses
default values. -->
        <xsl:param name="SRCLang" select="'Not Available'"/>
        <xsl:param name="TRGLang" select="'Not Available'"/>
        <!-- MLIF elements comes here. -->
        <xsl:element name="multiLingualEntry">
            <!-- ids are inserted here if they exist in original XLIFF
document -->
            <xsl:call-template name="elementID"/>
```

```
            <xsl:element name="multiLingualUnit">
                <xsl:element name="monoLigualEntry">
                    <!--monolingual segments with texual data. they also
have language attribute to find the language of text. -->
                    <xsl:element name="monoLingualSeg">
                        <xsl:choose>
                            <!-- checks if "xml:lang"  attribute exists in
source tag of the given XLIFF document'-->
                            <xsl:when test="source[@xml:lang]">
                                <!--attribute is inserted if it exists in
XLIFF document -->
                                <xsl:attribute name="xml:lang">
                                    <xsl:value-of
select="source/@xml:lang"/>
                                </xsl:attribute>
                                <!-- try to find the language of text in
XLIFF document and sets tags accordingly -->
                                <xsl:choose>
                                    <!-- if source language is english
"en"-->
                                    <xsl:when test="source/@xml:lang='en'">
                                        <text>
                                            <xsl:value-of select="source"/>
                                        </text>
                                    </xsl:when>
                                    <!-- if source language is french "fr"-
->
                                    <xsl:when test="source/@xml:lang='fr'">
                                        <texte>
                                            <xsl:value-of select="source"/>
                                        </texte>
                                    </xsl:when>
                                    <!-- if source language is other than
"en" or "fr". -->
                                    <xsl:otherwise>
                                        <text>
                                            <xsl:value-of select="source"/>
                                        </text>
                                    </xsl:otherwise>
                                </xsl:choose>
                            </xsl:when>
                            <!-- if XLIFF document doesn't have "xml:lang"
tag to specify language of source -->
                            <xsl:otherwise>
                                <!-- language tag is inserted same as
source language tag of XLIFF document -->
                                <xsl:attribute name="xml:lang">
                                    <xsl:value-of select="$SRCLang"/>
                                </xsl:attribute>
                                <text>
                                    <xsl:value-of select="source"/>
                                </text>
                            </xsl:otherwise>
                        </xsl:choose>
                        <!-- monoLingualSeg tag for source ends here -->
                    </xsl:element>
                    <!--monoLingualEntry tag for source ends here -->
                </xsl:element>
                <!--monoLingualEntry tag for target and alt-trans starts
here. -->
                <xsl:element name="monoLigualEntry">
```

```
                    <!--monoLingualSeg tag for target and alt-trans starts
here. -->
                    <xsl:element name="monoLingualSeg">
                        <!--checks for the "xml:lang" attribute in target
tag of XLIFF document.-->
                        <xsl:choose>
                            <xsl:when test="target[@xml:lang]">
                                <xsl:attribute name="xml:lang">
                                    <xsl:value-of
select="target/@xml:lang"/>
                                </xsl:attribute>
                                <xsl:choose>
                                    <!--if target language is specified as
english -->
                                    <xsl:when test="target/@xml:lang='en'">
                                        <text>
                                            <xsl:value-of select="target"/>
                                        </text>
                                    </xsl:when>
                                    <!--if target language if specified as
french -->
                                    <xsl:when test="target/@xml:lang='fr'">
                                        <texte>
                                            <xsl:value-of select="target"/>
                                        </texte>
                                    </xsl:when>
                                    <!-- calls if target language is not
"en" or "fr"-->
                                    <xsl:otherwise>
                                        <!--coipies the values of target
tag -->
                                        <text>
                                            <xsl:value-of select="target"/>
                                        </text>
                                    </xsl:otherwise>
                                </xsl:choose>
                            </xsl:when>
                            <!--if no language tag is specified it gives
from target language of XLIFF documents -->
                            <xsl:otherwise>
                                <xsl:attribute name="xml:lang">
                                    <xsl:value-of select="$TRGLang"/>
                                </xsl:attribute>
                                <!--copies text from target tag -->
                                <text>
                                    <xsl:value-of select="target"/>
                                </text>
                            </xsl:otherwise>
                        </xsl:choose>
                    </xsl:element>
                </xsl:element>
                <!--checks if alternative translations exists in XLIFF
document -->
                <xsl:for-each select="alt-trans">
                    <!--calls altTrans template -->
                    <xsl:call-template name="altTrans"/>
                </xsl:for-each>
            </xsl:element>
        </xsl:element>
        <!--mainTemplate template ends here -->
    </xsl:template>
```

```
    <!--elementID template starts here -->
    <xsl:template match="trans-unit" name="elementID">
        <!--checks if id exists in XLIFF document, if yes it is copied to
MLIFF document -->
        <xsl:if test="@id">
            <xsl:attribute name="id">
                <xsl:value-of select="@id"/>
            </xsl:attribute>
        </xsl:if>
        <!--elementID template ends here -->
    </xsl:template>
    <!--altTrans template starts here -->
    <xsl:template match="target" name="altTrans">
        <xsl:element name="monoLigualEntry">
            <xsl:element name="monoLingualSeg">
                <!--language tag is copied for alternative translation
units -->
                <xsl:attribute name="xml:lang">
                    <xsl:value-of select="target/@xml:lang"/>
                </xsl:attribute>
                <!--alternative translation is copied from target to MLIF
document -->
                <text>
                    <xsl:value-of select="target"/>
                </text>
                <!--monoLingualSeg tag for target ends here -->
            </xsl:element>
            <!--monoLingualEntry tag for target ends here -->
        </xsl:element>
        <!--altTrans template ends here -->
    </xsl:template>
    <!--stylesheet ends here -->
</xsl:stylesheet>
```