Localisation Focus

THE INTERNATIONAL JOURNAL OF LOCALISATION

ISSN 1649-2358

Special Standards Issue

The peer-reviewed and indexed localisation journal

VOL. 12 Issue 1

PUBLISHER INFORMATION

Guest Editors: David Filip, University of Limerick & **Dave Lewis**, Trinity College Dublin, Ireland

Production Editor: Karl Kelly, Localisation Research Centre, University of Limerick, Ireland

Published by: Localisation Research Centre, CSIS Department, University of Limerick, Ireland

AIMS AND SCOPE

Localisation Focus – The International Journal of Localisation provides a forum for localisation professionals and researchers to discuss and present their localisation-related work, covering all aspects of this multi-disciplinary field, including software engineering, tools and technology development, cultural aspects, translation studies, project management, workflow and process automation, education and training, and details of new developments in the localisation industry. Proposed contributions are peer-reviewed thereby ensuring a high standard of published material. Localisation Focus is distributed worldwide to libraries and localisation professionals, including engineers, managers, trainers, linguists, researchers and students. Indexed on a number of databases, this journal affords contributors increased recognition for their work. Localisation-related papers, articles, reviews, perspectives, insights and correspondence are all welcome.

To access previous issues online go to http://www.localisation.ie/ and navigate to the Localisation Focus Section

Subscription: To subscribe to Localisation Focus - The International Journal of Localisation www.localisation.ie

Localisation Focus – The International Journal of Localisation (ISSN 1649-2358) is published and distributed annually and has been published since 1996 by the Localisation Research Centre, University of Limerick, Limerick, Ireland. Articles are peer reviewed and indexed by major scientific research services, including: Bowker, Cabell's Directories and St Jerome Publishing Translation Studies Abstracts Online. It is also included in the Library of Congress Collections.

FROM THE EDITORS

Welcome to a Special Standards Issue of Localisation Focus based on papers and presentations that first appeared at the second FEISGILTT event.

Here is probably a good place to explain what a FEISGILTT is. According to Jörg Schütz, one of the FESGILLT programme committee members and by that token a member of the Editorial Board for this issue, FEISGILTT is a YALA, Yet Another Localisation Acronym. Not good enough? Sure, FEISGILTT stands for a **F**ederated **E**vent on **I**nteroperability **S**tandardisation in **G**lobalisation, **I**nternationalisation, **L**ocalisation, and **T**ranslation **T**echnologies (or Globali**z**ation, Internationali**z**ation, Locali**z**ation, and Translation as per the US spelling? another good reason to have the YALA).

This Search Engine Optimised YALA is supposed to be pronounced and read as "fesh-gilt", where "feis" is an Irish (Gaeilge) word for a festival of music and dance, which seems cheerfully appropriate because localisation interoperability, in much the same way as music and dance, needs orchestration and we do not want the federated event to be a gloomy academic event but rather a constructive gathering of standards workers, practitioners and the wider community of users, such as corporations and other multilingual content owners, service providers and so on. As the dancers at a traditional Irish feis, the participants present their work to their peers and openly discuss the pros and cons of solutions and approaches to standardisation and standards implementations.

The first FESGILTT consisted of two tracks: the 3rd International XLIFF Symposium and the ITS Track; it also had an important guest appearance by Helena Chapman who presented on the efforts of the Unicode Localization Interoperability Technical Committee (ULI TC). We managed to collect most of the presentations from that inaugural FEISGILTT conference and published them on the event's webpage at *http://www.localizationworld.com/lwseattle2012/feisgiltt/* (kindly hosted by the co-host of the FESGILTT events, the Localization World conference), we did not, however, manage to publish a representative collection of full papers in 2012. So we are immensely thankful to Mr. Reinhard Schäler who invited Dave Lewis and I, as FESGILTT Conference Chairs, to become Guest Editors of this Special Standards Issue of Localisation Focus.

All submissions made to the second FESGILTT event recieved no less than three blind peer reviews by our diligent Programme Committee, which in turn became the Guest Editorial Board for this Localisation Focus issue. The second FEISGILTT had three tracks, the 4th International XLIFF Symposium, the ITS Track, and the ETSI (**E**uropean **T**elecommunications **S**tandards Institute) ISG (**I**ndustry **S**pecifcation **G**roup) LIS (Localisation Industry Standards) Track. The ETSI ISG LIS Track featured a Linport presentation that became a full paper in this issue. Further, we have two papers dealing primarily with the XLIFF standard. In the first, researchers from the University of Salamanca present a practical approach to XLIFF standard based localisation of Joomla! Websites, while the second XLIFF paper explains how XLIFF 2.0 addresses interoperability issues based on lessons learnt from XLIFF 1.2 adoption. The remaining four papers look at the ITS 2.0 standard (two papers by researchers from Trinity College Dublin, an industry take on ITS 2.0 visualisation, and a detailed description of an ITS 2.0 driven CMS-TMS integration). All of these papers convey the FESGILTT baseline message, i.e. that successful industry standards must work in concert to achieve true standards based interoperability.

The FEISGILTT events that provided the basis for this special collection of papers would not have been possible without sponsors, most importantly Microsoft (Platinum Sponsor of XLIFF Symposium 2012) and CNGL who sponsored both FEISGILTT 2012 and 2013. So here is the appropriate place to thank them.

A million thanks go out from the Guest Editors to the Guest Editorial Board (aka FEISGILTT Programme Committees), Production Editor Karl Kelly, and last but not least all of the authors, who found the time during this turbulent year to turn their oral FEISGILTT presentations into camera ready papers.

Sincerely Yours, the Guest Editors
David Filip & **Dave Lewis**

Finally as a postscript, this journal's normal policy is to enforce academic style and UK spelling. We have modified these policies slightly for the issue at hand. This special issue brings together academics and practitioners and strives to provide practical and actionable information about localisation and internationalisation standards. We haven't enforced UK spelling in papers that were submitted with consistent US spelling and we did NOT overhaul specific styles of, in particular, industry practitioners to achieve conformance with the conventions of academic writing beyond readability and citation format.

The Editors

Localisation Standards for Joomla!
Translator-Oriented Localisation of CMS-Based Websites

Jesús Torres del Rey[1], Emilio Rodríguez V. de Aldana[2]
[1]Department of Translation and Interpreting
[2]Department of Computer Science and Automatics
University of Salamanca
Spain
jtorres@usal.es, aldana@usal.es

Abstract

For a localiser, the shift from static to CMS-based dynamic websites usually involves assimilating a new editing environment, acquiring administrative rights for the site, and relinquishing the various benefits of using CAT tools. However, the possibility of integrating CAT tools in the localisation process is now becoming a reality by means of localisation standards (mainly ITS and XLIFF). In this paper, we introduce an experimental Java application we have developed for the import/export of multilingual web content for the Joomla! CMS (with the FaLang extension). We go through the workflow and explain the lessons learnt from our experiments with this and other related tools. As our research is translator-oriented, we discuss some current limitations for localisers' work in the theoretical and practical approaches taken for the multilingual management and translation of CMS-based websites and suggest some alternatives for the future.

Keywords: *web localisation, localization, Content Management System, CMS, standards, Internationalization Tag Set, ITS, XLIFF, roundtrip, interchange, Translation-Oriented Localisation Studies, communication, text, meaning*

1. Introduction

The development of websites has quickly evolved over the last half decade, as Esselink (2002, p.5) announced for digital content in general, from being "traditionally written" using html editors, such as Dreamweaver, FrontPage, Expression Web, Amaya, Nvu or Kompozer, to being "dynamically built using database driven publishing systems or content management systems", particularly thanks to the boom of FOSS web CMSs such as Drupal, Joomla! or Wordpress (Torres del Rey and Rodríguez V. de Aldana 2011, 2014).

On the client side, it could be argued that things have remained essentially the same for the last two decades due to the consolidation of HTML as the main content language and file type, and of browsers as the leading web surfing application. Of course, user *experience has* changed dramatically with the introduction of more dynamic client-side technologies such as Javascript and other scripting languages, Ajax or CSS, the embedding of Java applets or flash animations (Mata Pastor 2005, pp.197-198], the gradual move to XML vocabularies and HTML 5, and so on. And yet, the way end users experience web *navigation* "macrostructurally" (Mata Pastor, pp.200-202), "hyperstructurally" (Torres del Rey and Rodríguez V. de Aldana 2014) or "superstructurally" (Jiménez Crespo 2013, pp.92-94) still revolves around concepts such as webpages as units, hyperlinks and forms as the main functional devices, and document tree structures stemming from a homepage and branching out through a series of sections and subsections.

It is on the server side where the main revolution —as far as developers and webmasters, but also translators and localisers are concerned— has taken place. To continue with Esselink's words: "Where translators could get started quickly by just working in Word or importing the *document* into a *translation memory system*, now often a localization *engineer* is needed to produce a 'translation *kit*' from a series of *complex SGML or XML files* containing the *manual text*" (Esselink 2002, p.5. Emphasis added). It is here, where the notions of whole documents and of straightforward import-export processes via translation memory systems are being challenged, that we decided to focus our research, spurred by the needs of our undergraduate localisation course at the University of Salamanca.

2. Motivation and nature of our research

Our main interest in new localisation processes for

dynamic webpages started in 2008, when we were asked to translate our Faculty's site from Spanish into our other working languages, with the collaboration of students. The website had been built with Joomla! 1.5, and was later made multilingual with the Joom!Fish extension[1]. We were given editing rights to this component, which allowed us to search for web articles, menus and other translatable elements, and to write or paste translated HTML content onto the editor window. In order to replicate the localisation process used for static HTML websites, client-side webpages were saved as only-HTML files, translated by means of a CAT tool with the aid of translation memories, terminology management and other integrated utilities, and the resulting HTML content was pasted onto the appropriate Joom!Fish editing environments.

Very soon, we decided we wanted to further explore how dynamic website localisation processes could be made more translator-friendly and, at the same time, to integrate them as seamlessly as possible with the whole development and publication cycle. However, the main drive behind this was to be able to analyse, understand and explain this evolving infrastructure and the new localisation needs and opportunities in order to enhance our localisation course, where students had learnt advanced website localisation concepts and strategies such as essential file types, languages and technologies, website (super, macro, micro and hyper) structures, including directory organisation and hyperlink types, folder structure cloning and hyperlink management, automation strategies (search/replace with or without regular expressions), etc.

In 2012, Joomfish was no longer available for the newest Joomla! 2.5, so we moved to the Joom!Fish-fork extension FaLang[2], which was also compatible with the more recent Joomla! 3.x version. It is important to note that the main goal of these third-party *internationalisation* extensions is to easily and automatically duplicate, for each newly activated site language, the monolingual web structure created with Joomla!, and to enable the editing and publication of translated content in all non-native languages. However the process of *localisation per se* was only facilitated when modules were created for the export/import (*interchange*) of translatable data and, particularly, when both technologies started to be merged: multilingual management and localisation interchange tools[3].

For the purposes of our research and, particularly, for our teaching practice, we adopted a twofold strategy: to try and generalise common features in the localisation of CMS-based websites and to illustrate this general process by setting up appropriate mechanisms and procedures to experiment it. We started looking at the architecture of other CMSs such as Drupal or Wordpress and the way internationalisation and localisation extensions were integrated. At the same time, we started developing an experimental tool that allowed us to automate a roundtrip export/import workflow for the Joomla! CMS we used for our teaching, and to identify and describe the main concepts, processes and possible breakdowns for translators' and localisers' work. This article mainly deals with our experiments in this process, particularly with the tool we developed for our teaching, the comparison with other available tools, and some conclusions regarding the general process of localisation and suggested basic translator and localiser needs.

One of the crucial questions in our roundtrip between the CMS and localisers' workstations was the format in which the interchange would take place, so, naturally, we looked at the possibilities offered by the two main standards in our field: the W3C Internationalisation Tag Set (ITS) and the OASIS XML Localisation Interchange File Format (XLIFF). While versions 2.0 of both standards will undoubtedly offer many new advantages to this process, at the time of our experiments and of writing this article they were still in draft status, so we used the latest approved versions, ITS 1.0 (W3C 2007a) and XLIFF 1.2 (OASIS 2008).

We have already introduced the three main components of our localisation research focus: 1. the product and the underlying technology relevant to localisation; 2. the interchange format for localisers to process; 3. relevant translation-oriented technologies for the processing of localisable texts, notably CAT tools. Our approach, as mentioned earlier, has to do with the integration of all three components in a way that is translator- or localiser- oriented, since it is these professionals who are in the best position to account for the intercultural task of negotiating the meanings, purposes, expectations and conventions which come into contact (and often into conflict) in the process of localisation, and to interpret objects, texts and meanings not for their own sake but for other people, users (at both –or the multiple– ends of communication), for whom translation and localisation is performed, for whom the translator is ethically responsible, and which determine meaning and transformations (Melby 1995, pp.122-132). However, as advocated in our localisation courses, in order for localisers to claim this expert position, they

must acquire a basic knowledge of the nature and mechanics of dynamic, CMS-based websites, particularly in so far as the technology influences both the communication production assemblage and localisers' own place in the development cycle.

In this regard, we feel that it is our duty to contribute to the reversal of the current wave of disempowerment for website localisers as we move from static to dynamic websites. The three components mentioned earlier empowered localisers working with static websites by providing them with (Torres del Rey and Rodríguez V. de Aldana 2011, 2014):

1 a high degree of visual and functional context;
2 specialised productivity, QA-performing tools;
3 the possibility of taking over engineering tasks for the multilingual restructuring of the overall website;
4 the possibility of delivering publication-ready directories and files.

CMSs have made the editing of individual articles within webpages and of interface items across the website easier. This has provided some visual content by allowing for (limited) in-context translation of articles. However, translators need to process the texts in the web product by means of their own tools in order to take advantage of the consistency, analysis, quality-check and terminology extraction functions (among others) built into them, and, often, to exchange the textual elements with other collaborators, who may use different tools. In CMSs, however, texts are still very much "locked" into databases. Localisers would also need write-access rights to the database and a multilingual component installed in order to try and recover some of the possibilities 3 and 4 above.

Even though automation seems to make the whole multilingual generation and publication easier and less error prone, any new web technologies and content management systems affect the way content is created and signifies. Localisers, as techno-linguistic experts, should not be nudged aside. This deskilling perspective, called the "idiot-proofing myth" by Adler and Winograd, "is more concerned with how to keep operators from creating errors than with enabling operators to deal with the inevitable contingencies of the work process". However, translation and localisation are all about dealing and negotiating with (linguistic, cultural, technological, contextual) dependencies, so we had better rise to the "*usability challenge*" (emphasis in the original) of making new technologies more effective by augmenting rather than replacing skills of localisers, by making the most of them (Adler and Winograd 1992, p.3).

3. Our experimental research

3.1 Overview

As indicated earlier, our main research goal was to provide localisation students with the conceptual and methodological tools to experiment with the process of localising a CMS-based dynamic website, and to do it on the basis of the three basic components for this task, i.e. the product and its technology; the interchange format; and the CAT tool. As we looked into the way these could be integrated, we also expected to draw insightful lessons for a translator-oriented approach to the task at hand.

It is only very recently that XLIFF extraction/merge tools have started being developed for Joomla! That is the main reason why we decided to build our own software, with the main purpose of experimenting with the data that needed to be exported and the way XLIFF could accommodate such data and the whole localisation process. Our tool was based on the multilingual extension to Joomla! developed by FaLang. As the extraction and merge operations were mainly made on the database tables created and managed by this plugin, we called our software FaLang2XLIFF.

FaLang2XLIFF[4] has proved very useful for our purposes, particularly considering our scarce resources, both economically and in terms of our available time. However, it has some limitations that need to be taken into account. Most importantly, it has been written in Java and is a stand-alone application which, unlike other related tools, is not embedded into the CMS as a module. Although this would make it potentially applicable to other database structures, both for Joomla! or other CMSs, it would also need to be given access rights to the database or to be run in the relevant network security zone or in a localhost.

Our application uses Schnabel's XSL stylesheet from his XLIFF Roundtrip tool[5], which converts XML files into XLIFF and back again. Drupal's XLIFF Tools is also based on that XSL file, so we briefly analysed its extraction performance. However, the alternative tool that we tested the most was JDiction[6], a multilingual management extension for Joomla! 2.5 which added an XLIFF extraction/merge tool in March 2013. Before describing the workflow of FaLang2XLIFF we will present some relevant conclusions from our analysis of JDiction.

All these tools, as well as our own, only deal with

content stored in the CMS database, i.e. articles or pages, modules (for instance, text in an add-on calendar), categories and other small user interaction elements, such as weblinks. The three main database elements (stored in text fields) for these contents are exported: web structure or interface (In Price and Price's terminology, cited in Jiménez Crespo 2013, p.58]) elements, longer (X)HTML article contents, and the technical parameters for the above elements. At this point, we have not yet considered processing dependent or linked files. Neither have we looked at CMS administration or content editor interface text, typically inserted in active pages (such as PHP, ASP or JSP) or externalised to text files (e.g. INI, PO).

3.2 Other extraction strategies

Our tests with JDiction for Joomla! 2.5 revealed certain problems for the localisation process. To start with, this tool shares with other multilingual managers, such as the current FaLang version, a shortcoming in that the article that is cloned for the target language cannot be edited in-context, on the webpage itself. Instead, the translation must be inserted in a separate environment, where the original text is not shown in parallel[7].

If we look at JDiction's XLIFF extraction/merge tool, we can appreciate considerable room for improvement too. For instance, items cannot be selected and exported individually; besides, this bulk process is applied on any one content type (categories, article contents, menus, menu items and modules) indiscriminately, regardless of whether individual elements are new, updated or they have previously been translated and approved. Finally in our analysis, all titles and article content receive "needs-translation" state values, irrespective of their actual status. On the other hand, parameters are always marked as "translated" in JDiction ("final" in Drupal's XLIFF Tools), which may cause the processing translation tool to edit unlocalisable values, resulting in the corruption of the database.

What is more, all extracted elements are lodged inside CDATA sections (Savourel 2001, pp.229, 298), which would commonly prevent parsing of the (X)HTML structure and segmentation based on it. This makes the XLIFF export no different from its pre-existing CSV export in JDiction. When filtered into CAT tools, articles would be processed whole, and tags would be mingled with actual text and unprotected. Even if, as Virtaal does by means of regular expressions such as <[^>]+/?>, tags are visually marked, translators would have a hard time trying to mentally reconstruct texts, identify and separate subtexts such as *alt* values, or correctly assign functional or layout tags to the appropriate words, phrases, sentences or paragraphs. Not to mention the high risk of messing with the code that this behaviour would entail.

CAT tools might alleviate the latter problem by integrating a WYSWYG editor for HTML content, which would also be triggered whenever the XLIFF *datatype* attribute value is "htmlbody", allowing users to switch between raw source HTML text and the visual representation of HTML tags on text. However, one thing that must be taken into account with CAT tools is that their XML filters are not always versatile enough and too often they only allow for the use of regular expressions to further filter translatable content, internal or external tags, and so on. In fact, XHTML should be processed with XML processors (e.g. XPath processors), in order to help interpret meaningful structures, which would produce shorter segmentation for better matches and translation memory leverage. This problem could be more easily solved, nonetheless, if the XLIFF export was carried out as HTML text and tags rather than plain text, or if, previously, the database could actually manage XML structures, as we will see later (see note no. 9).

title	introtext	metakey
Administrador de componen...	<p>Todos los componentes se utilizan también en el...	
Modulo Archivo	<p>Este módulo muestra una lista de los meses del ...	modules, content

Figure 1. An example of the three main database elements: title (structure), introtext (content) and metakey (parameter).

3.3 Workflow of our experimental application

Before using our tool, elements should be prepared for translation by means of FaLang's administrative interface (step 1). Currently, there are two important disadvantages in the behaviour of this extension: elements need to be selected and opened one by one, and the relevant source content must be copied onto the target content window.

Once target elements have been created for translation, it is the turn for our tool to connect to the database (step 2), for which it is necessary to provide the machine server name, the communication port (usually, 3306 for standard TCP/IP connections), the user ID having administrative rights, the user

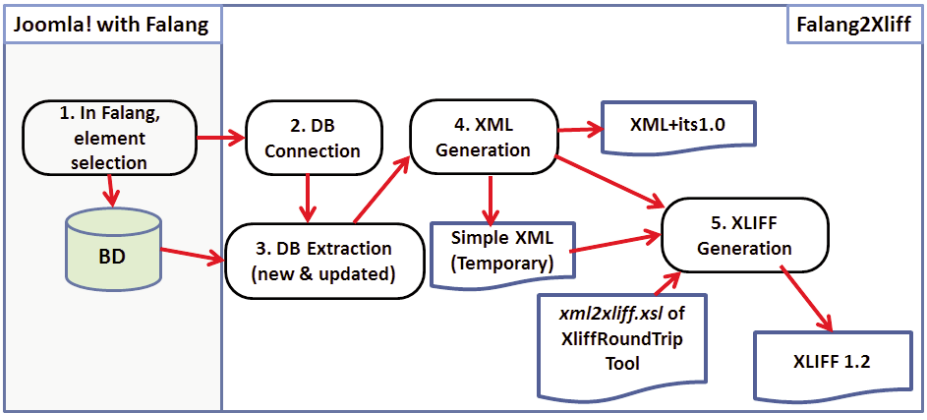

Figure 2. FaLang2XLIFF Workflow.

password, the name of the Joomla! database and the prefix or alias typically added to Joomla! table names.

Our application queries FaLang tables (step 3) but also the original content tables so as to check which data is new (i.e. established as translatable by the project manager by using the "Copy Source" procedure described in the first paragraph of this subsection) and which has been modified or updated in the source website[8]. In order to identify both types of data (new and modified), the MD5 hash code of both the translation record in the FaLang table and the corresponding record in the original content table are compared. Updates are identified whenever the source and target hash codes differ. On the other hand, translation content is considered as "new" when two conditions are met: source and target hash codes are the same and the "published" field of the FaLang record equals "0", i.e., it has never been published before, as otherwise it might mean that the source content has consciously been transferred to the target record (e.g. in the case of some proper nouns, trade names, etc.). Once new and modified translatable data are identified, their structural (e.g. titles) and content elements are extracted, but not technical parameters, as editing them may corrupt the database. However, in the future we will further analyse extractable parameters, as they may provide important contextual information for the localiser.

It is important to mention that the Joomla! HTML editor would have rewritten HTML fragments typed by users as XHTML (i.e. as correct XML)[9]. Nonetheless, our tool uses Jericho HTML Parser to recheck it and then rewrites data if necessary to make sure restricted characters in XML are escaped with their corresponding predefined character entity references (e.g. *&* for the *&* ampersand characters) (Savourel 2011, pp.44-47), attribute quotes are closed, and node hierarchies are kept. A current limitation is that all unpaired tags found in the XML hierarchy would be changed by our tool to self-closing tags without further analysis.

At this point (step 4), we would generate both an XML file with ITS rules and a temporary "simple" XML file that would serve as the basis for conversion to XLIFF by means of Schnabel's XLIFF Roundtrip XSL. It is worth noting that while intro and full texts are stored as HTML (<tags> and text) in the database, title fields contain only plain text and that no HTML/XML entities are processed. This means that we need to convert single characters, such as the ampersand, that may appear in the title field to their corresponding entity (*&* in this case) when processing the XML files, and then back to the single character when importing back to the database.

The XLIFF 1.2 file is successfully created (step 5) via the simple XML file just mentioned: here, database records are exported with <records_falang> as the root node and two child tags (see Fig. 3): <record_falang> carries, in its attributes, the administrative data that are needed to be merged back into the database; as a child node of the latter, <value_falang> contains translatable text, including HTML tags. We have adapted Schnabel's xml2xliff.xsl file used for the conversion so that the source language is variable (by using the XPath expression *{./@xml:lang}* as the value of the *source-language* attribute of the root element <file>) rather than just English ("en"). For clarity's sake, we have abbreviated some of the illustrated code by means of the ellipsis symbol "(…)".

However, we encountered several difficulties in the processing of the XLIFF file in CAT tools, as we will discuss later on. For that reason, after considering what might solve the problems we had identified, we

```
<?xml version="1.0" encoding="utf-8" ?>
<records_falang xml:lang= "es" host="..." db="..." alias="..." >
    <record_falang id = "453" (...) original_value ="3de7..."
                                        type="modificado">
        <value_falang><p><img (...)/>Usted tiene...
        </p></value_falang>
    </record_falang>
    <record_falang id="..." (...)>
        <value_falang>Usando Joomla! &amp;amp; Componentes
        </value_falang>
    </record_falang>
</records_falang>
```

Figure 3. XML file generated by FaLang2XLIFF.

decided to produce a second version of the "simple" XML file described earlier, injecting it with ITS 1.0 rules regarding the processing of translatable elements and of segmentation-related text-element relationships (W3C 2008), as follows (see Fig. 4):

- We used global (not local) rules, directly embedded in the resulting XML file.
- All <value_falang> nodes and their child nodes were made translatable; all other nodes are not translatable[10].
- Within <value_falang> nodes, HTML attributes typically carrying text are made translatable. Href attributes are also localisable when they start with "http://" or "https://" (i.e., generally, when they are external site references) [11].
- HTML elements that can occur inside text sentences (such as <a> or) are considered Within Text, which prevents segmentation (see later).

The return trip to the database is also performed by FaLang2XLIFF, so far irrespective of translation status (e.g. nodes marked as "needs-translation" will still be imported back to the database). Again, a temporary XML file needs to be produced from the XLIFF 1.2 file before SQL generation. The database can be updated directly online, although an SQL file will also be produced, in case the update is to be done in batch mode.

3.4 Analysis and Discussion: interchange problems

The application of general-purpose XSL transformation files to specific mark-up languages such as XHTML when written by CMS HTML editors may present a series of limitations. One of the consequences of this is that attribute values would not be extracted to XLIFF <trans-unit> elements in XLIFF. Take, for instance, the Joomla! article in Fig. 5, with the following source code:

```
<?xml version="1.0" encoding="utf-8" ?>
<records_falang xml:lang= "es" host="(...)" db="(...)" alias="(...)"
 xmlns:its="http://www.w3.org/2005/11/its" its:version="1.0">
<its:rules xmlns:its="http://www.w3.org/2005/11/its" version="1.0">
<its:translateRule selector="//*" translate="no"/>
<its:translateRule selector="//value_falang//* | //value_falang"
translate ="yes"/>
<!--translatables attributes-->
<its:translateRule selector="//value_falang//@alt | (...) |
//value_falang//@href[starts-with(.,'http://') or
starts-with(.,'https://')] |
 (...)" translate="yes"/>
<!--Elements within text-->
<its:withinTextRule selector="//value_falang//a | (...) |
//value_falang//img | //value_falang//span |(...)" withinText="yes"/>
</its:rules>
<record_falang id ="...">
    <value_falang>(...)</value_falang>
</record_falang>
</records_falang>
```

Figure 4. ITS rules injected in the output XML file. Ellipsis (…) is used.

- ** **<*img alt=”Joomla! Spanish, versión 2.5”*>Usted tiene un sitio Joomla! 2.5 adaptado y traducido por Joomla! Spanish** **
- ****Joomla! es de <*a href=”http://opensource.org” title=”Iniciativa Open Source”*>código abierto</*a*>. Joomla! hace que sea:****
- **<span style=”line-height: 1.3em;”>**Fácil <*strong*>crear</*strong*> y construir un sitio web de manera que quiera.****
- **<span style=”line-height: 1.3em;”>**Bastante <*strong*>sencillo</*strong*> de <*strong*><*em*>actualizar y mantener.</*em*></*strong*>****

The XliffRoundTrip transformations to XLIFF 1.2 are as follows:

- tags without text are included in <group> elements (html tags without text; highlighted in grey in our source code);
- tags with text are inserted in <trans-unit> elements (in **bold**);
- inline or within text tags are transformed into <g> </g> pairs or into <x/> xliff elements (in italics).

This would allow us to localise the *alt*, *title* and *href* attribute values (provided the latter starts with http:// or https://). We could also transform this XML file to XLIFF successfully by means of Okapi Rainbow, which also supports global *Translate*, *Elements Within Text* and *LocNote* ITS rules (W3C 2007b), and then import it into any XLIFF-supporting CAT tool.

Another problem that could be averted with ITS rules has to do with HTML overtagging, typically produced by CMS HTML editors. If, for instance, we are writing the above article in the CMS editor and we undo the list item or the whole ordered list and then change the paragraph configuration, Joomla! would add pairs of tags with style attributes around paired , or <a> tags including text. According to the transformation rules for XLIFF indicated earlier, that would produce undesired oversegmentation, as text within new paired elements would be included in their own <trans-unit> nodes (i.e. in independent segments or translation units). To sum up, many reformatting actions on the CMS html editor cause html overtagging, which can hardly be safely undone by CMS Clean-html functions.

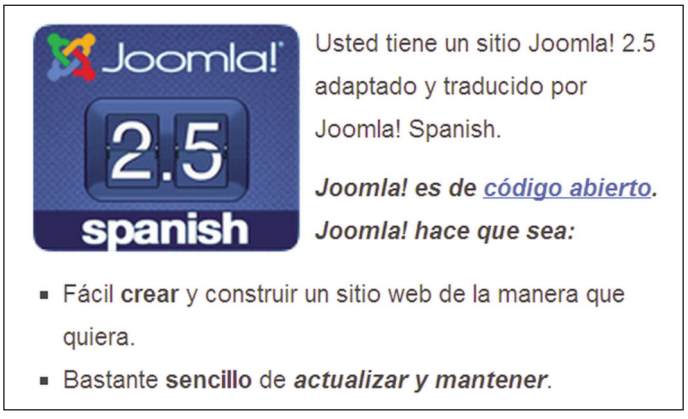

Figure 5. Sample Joomla! article.

The resulting XLIFF file would not include translatable attribute values within <trans-unit> nodes. Instead, inline tags would have an id reference to said values, which would be kept in the skeleton part of the file for later merging.

However, the XML file with ITS rules that Falang2XLIFF generates would be processed more effectively by a CAT tool – such as SDL Trados Studio– that does support global and embedded ITS rules for features *Translate* and *Elements Within Text*.

4. Towards translator-oriented localisation of CMS-based websites

Solutions to the internationalisation and localisation of CMS-based websites tend to focus rather narrowly on the technical aspects related to the extraction/merge roundtrip of translatable data or on the user-friendliness of integrated multilingual management and in-context article edition. However, little or no attention is paid to the overall communication needs that translators and localisers

must address in order to do their job successfully from the point of view of the pragmatic, intercultural, interlinguistic exchange that they are commissioned to perform.

It is true that the technical solutions mentioned earlier bring the localisation process a step closer to human, translation-oriented concerns:

- by using an XLIFF file, translation data can be enriched with information on the localisation process, objects, intentions, and so on;
- by partaking in the roundtrip, the localiser may not be seen as a "dysfunctional" agent in the technological process, but as an "enabler" in the infrastructure of (multilingual) content management;
- by handling a standard interchange format, localisers can use their computer-assisted tools and benefit from translation memories, terminologies, quality control, and all other integrated translation aids;
- alternatively, by being provided a simple system to enable and manage the multilingual structure of the website, they can devote more attention to translation matters, including negotiating contrastive conventions of web genres (Jiménez Crespo 2013, Ch.4);
- finally, by allowing localisers to place (and replace) translations in allocated webpage spaces (for some CMS content types), they benefit from a more contextualised approach to the translation of web articles.

However, the above advantages are currently far from being fully realised, particularly the combination of in-context visual translation and XLIFF support (which also show some limitations, as we have seen in the previous section). In general, a holistic view is missing as regards the part that the different technological, textual and semiotic components play in the task of the localiser, and how they can be realised in the translation process.

A translator or localiser is an intercultural mediator who makes sense of a text (or an interrelated series of texts)[12] that has been produced in a specific cultural, professional and technological context, and creates a version of that text in a different human language, taking into account differences between source and target contexts, the (explicit or assumed) purpose of the textual exchange and foreseen effects of the resulting text (in the target system or context, but also as regards the source context of production), and formal or informal norms and conventions regulating translation and localisation, usually related to culture-bound ideas of equivalence, adequacy, comparativeness and functional adaptation.

All these contextual, cultural, technological, purpose-bound considerations have a huge impact on the task of the localiser, just as they implicitly or explicitly condition the original text production process. The intercultural mediator, furthermore, needs to stand astride (or to constantly move across and back) the source and the target cultures and language systems, and to make informed decisions in order to communicate or negotiate global and particular meanings, functions, intertextual relations, purposes and (intended or unintended) effects which have been formed, structured and expressed in a linguistic mould and in a cultural context which can never be symmetric or equivalent with the target language and culture.

The other major communication, sign-producing system that greatly influences the production of meaning is the technological – here, the website as a product and the CMS as an agent mediating structure, communication, document or text boundaries, and, in general, the interaction of knowledge between users (designers, contributors, consumers, "browsers-by", and so on), the web genre and the information to be displayed.

However, web CMSs tend to gear their modus operandi towards monolingual, monocultural production, not only because multilingual management extensions are often a later add-on (and not as user-friendly or flexible as the interface for original content editing), but also because there is a source-oriented inherent assumption of direct, objective, unproblematic, ungrounded semantic correspondence (Winograd and Flores 1986, p.18; Melby 1995, pp.122-132) between the genesis of meaning and intention and the infrastructure and applications enabling and conditioning their expression. This kind of correspondence is circular (meaning > production structure, materials and mechanisms [language, writing and technological systems] > meaning adjustment > system adjustments > meaning, etc.) and is not recreated and rarely unveiled for localisation. Thus, shockingly enough, localisation tends to be left out of the meaning-production cycle.

Localisers are usually provided small chunks of text (either in the CMS editing environment or as translation units in bilingual interchange files) for

ease of exchange and integration in the localisation or publication technologies. Even if "experts most likely develop strategies either in a pre-translation stage (by acquiring prior knowledge of the global hypertext […]), or during the translation process ([…] from a prototypical of the digital genre in question and [by] negotiating the macro and microstructural levels) to compensate for the lack of context" (Jiménez Crespo 2013, p.64), localisation efficiency can be severely disrupted by forcing constant negotiation between meaning-structure levels, context recreation, and, particularly when "translating interaction", i.e. when texts and messages are not on the immediate surface visible webpage layer.

Any web content is meaningfully integrated in a larger information unit (e.g. a bigger article or a web page), next to other subunits (or subgenres), and also within a larger whole (the website, or even the World Wide Web). Localisers usually receive only the small subunits, with little or no information of relative position, order or functional dependencies. However, these units are coherently and cohesively (Jiménez Crespo 2013, pp.59-62) inserted (at least) in:

- the more general or particular communicative or performative functions they are part of;
- the regions or positions they appear in, which also have communicative or semiotic significance;
- the hypertextual, interactive relationships they are part of or which they include;
- macrostructural relationships (e.g. the particular location in the sitemap or the order they appear within a group or element such as a menu);
- the conventions for the type of element they are in (a more or less ephemeral article, a more stable basic page, a module, a category classifying blog entries, etc.);
- potentially indexed search results.

In this regard, CAT tools need to be able to offer relevant contextual information to prevent localisers from concentrating exclusively on the microtextual level (Jiménez Crespo 2008, pp.5-6), and for this, XLIFF development and CAT support (and visualisation and interaction) of this interchange format must grow closer together. However attractive, in-context translation in the CMS editing environment without the benefits of Computer-Aided Translation Technology can be dangerously insufficient as it would not profit from some of the main benefits of CAT technology, if used properly:

- translation and terminological consistency, particularly as regards specialised knowledge, web genre conventions, brand or client-related phraseology and terminology, and so on;
- quality checks;
- productivity functions;
- filtering and transferring format and presentation;
- language/knowledge building and annotation;
- team work and exchange functionality, particularly as large websites tend to be localised by more than one professional.

An important step forward would be for web CMSs to incorporate localisers and localisation into their content management agents, definitions and mechanisms, since the amount of content that localisation handles and transforms is substantial. One way to do this is to support and encourage the generation of ITS 2.0 (W3C 2013) metadata for translatable elements and attributes, text analysis (content, structure, relations between parts), external resources (e.g. relevant intertextual, intermedia references, whether explicit or implicit, that are important for overall meaning construction), size or other restrictions, linguistic annotation and any other features that may affect data interchange (via XLIFF 2.0 [OASIS 2013]).

Another complementary way would be to provide the appropriate mechanisms for a localisation project manager (PM) profile in CMSs. This user would be able to annotate content and include relevant metadata (e.g. specific localisable external links, localisation notes, text analysis, and so on), or prepare localisation interchange files, by grouping translatable content with contextual non-translatable content, including an html preview skeleton file, linking appropriate XSL/CSS files for better visual contextualisation, or, simply, providing URL links for each group of localisable content.

5. Conclusions and future work

The declared purpose of CMSs is managing content in a structured, knowledge-sensitive (and sensible) way. Localisation should therefore be part of their core concerns, and it would be sensible if CMSs integrated the Internationalization Tag Set with their content generation strategies, and XLIFF with their multilingual content interchange mechanisms.

Particularly, localisation of whole or large sections of websites (as opposed to periodic translation of individual, more-or-less independent articles) and

web localisation training would greatly benefit from textual signposting and contextualisation strategies, which could be included in internationalisation metadata for the original content (by authors or localisation PMs) and transferred or enriched in the XLIFF files that localisers would process with their CAT tools.

This is one of the avenues of experimental research that we will pursue in the near future: the extraction of contextual information that can be useful for CMS website localisers and can be integrated in XLIFF files for CAT work. At the same time, we will intensify our analysis of other roundtrip tools (and other possible localisation strategies) for web CMSs, and the way these content management systems design their interaction with web objects, concepts, conventions, meaning and interrelationships. Finally, we will continue to look into CAT integration of current and future CMS-based web localisation processes.

After all, we need to understand the way information, knowledge and communication is conditioned and shaped by technology (expanding some possibilities, reducing others, creating new meanings) in order to try and reach an understanding between the different professional languages involved in dynamic web localisation, to build (by assimilation, contact, translation, etc.) common metaphors that may help translators and localisers (and their trainers) to "inscribe" translation values and meanings in the operating system of CMS technologies (Torres del Rey 2005, pp.105,121-134), often by means of standard languages.

Acknowledgements

The work presented here has been carried out in the framework of the research project "Regulación de los procesos neológicos y los neologismos en las áreas de neurociencias" (FFI2012-34596), which receives funding from the Spanish Ministry of Economy and Competitiveness.

References

Adler, P.S. and Winograd, T. (1992) 'The Usability Challenge', in Adler, P.S. and Winograd, T., eds., *Usability. Turning Technologies into Tools*, New York & Oxford: Oxford University Press.

Esselink, B. (2002) 'Localization Engineering: The Dream Job?', *Tradumàtica*, (1), available: http://www.fti.uab.es/tradumatica/revista/articles/beselink/art.htm [accessed 24 Oct 2013].

ISO/IEC (2011) *Information technology — Database languages — SQL — Part 14: XML-Related Specifications (SQL/XML)*, 9075-14:2011 [online], available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=53686 [accessed 24 Oct 2013].

Jiménez Crespo, M.A. (2008) *El proceso de localización web: estudio comparativo de un corpus comparable del género* sitio web corporativo, Ph.D. dissertation, Granada: Universidad de Granada.

Jiménez Crespo, M.A. (2013) *Translation and Web Localization*, London & New York: Routledge.

Mata Pastor, M. (2005) 'Localización y traducción de contenido web' in Reineke, D., ed., *Traducción y localización: mercado, gestión y tecnologías*, Las Palmas de Gran Canaria: Anroart, 187-252.

Melby, A.K. (1995) *The Possibility of Language: a discussion of the nature of language with implications for human and machine translation*, Amsterdam/Philadelphia: John Benjamins.

OASIS (2008) *XLIFF 1.2 Specification.* OASIS Standard 1 February 2008, available: http://docs.oasis-open.org/xliff/xliff-core/xliff-core.html [accessed 24 Oct 2013].

OASIS (2013) *OASIS XLIFF Wiki Frontpage*, available: https://wiki.oasis-open.org/xliff/ [accessed 24 Oct 2013].

Savourel, Y. (2001) *XML Internationalization and Localization*, Indianapolis: Sams.

Torres del Rey, J. (2005) *La interfaz de la traducción: formación de traductores y nuevas tecnologías*, Granada: Comares.

Torres del Rey, J. and Rodríguez V. de Aldana, E. (2011) 'La localización de webs dinámicas: presente y futuro', accepted for *1st International T3L Conference*, Universitat Autònoma de Barcelona, June.

Torres del Rey, J. and Rodríguez V. de Aldana, E. (2014, forthcoming) 'La localización de webs dinámica: objetos, métodos, presente y futuro', *JoSTrans*, (21), available: http://www.jostrans.org.

W3C (2007a) *Internationalization Tag Set (ITS) Version 1.0*, W3C Recommendation 03 April 2007 [online], available: http://www.w3.org/TR/its [accessed 24 Oct 2013].

W3C (2007b) 'Test Suite', in *Internationalization*

Tag Set Version 1.0, Version: $Id: Overview.html,v 1.56 2007/02/27 06:20:03 fsasaki Exp $, available: http://www.w3.org/International/its/tests/Overview.html [accessed 24 Oct 2013].

W3C (2008) '5.1.4. Associating existing XHTML markup with ITS', *Best Practices for XML Internationalization*, W3C Working Group Note 13 February 2008 [online], available: http://www.w3.org/TR/xml-i18n-bp/#relating-its-plus-xhtml [accessed 24 Oct 2013].

W3C (2013) *Internationalization Tag Set (ITS) Version 2.0*, W3C Proposed Recommendation 24 September 2013 [online], available: http://www.w3.org/TR/its20 [accessed 24 Oct 2013].

Winograd, T. and Flores, F. (1986) *Understanding Computers and Cognition: A New Foundation for Design*, Norwood (NJ): Ablex.

Notes

1 http://www.joomfish.net/

2 http://extensions.joomla.org/extensions/languages/multi-lingual-content/18210.

3 Josetta (http://anything-digital.com/josetta/) is another multilingual manager for Joomla. XLIFF Tools (https://drupal.org/project/xliff) is both a multilingual manager and an XLIFF roundtrip tool for Drupal, just like WPLM (http://wpml.org) for Wordpress.

4 http://diarium.usal.es/codex/desarrollo.

5 http://sourceforge.net/projects/xliffroundtrip.

6 http://jdiction.org.

7 In the case of FaLang, this seems to be a bug, as the editing window for the target language does work (and with the original text visible) but the result is inserted in the native language tables.

8 For an analysis of the database tables and attributes that are queried, see our article (Torres del Rey and Rodríguez V. de Aldana 2014). As mentioned earlier, an in-depth analysis of the way other CMSs (or their multilingual managers) organise tables and translatable elements would allow us to extend the functionality beyond Joomla! with FaLang.

9 XHTML elements should be stored in databases as XMLElements, as recommended in ISO/IEC (2011). Unfortunately, at the moment XML support is low in MySQL, which is the favoured database management system for CMSs. We believe that it would be beneficial to adopt other systems with more advanced XML functions such as PostgreSQL or to press for further XMLsupport in MySQL.

10 "In case of conflicts between global selections via multiple rule elements, the last selector has higher precedence" (W3C 2007a, Sec. 5.4).

11 ITS 1.0 supports XPath 1.0, which does not support regular expressions, which would have made a few conditions simpler than with Xpath syntax.

12 For the purposes of this article, "text" also means hypertext and associated multimedia and interaction. All technical adaptations that may be necessary in the localisation process are considered as part of the interpretation of the text as we have just defined, and will not be covered here mainly because our focus is on the export/import of textual material from CMS-based websites.

Interoperability Frankfurt-Madrid: ITS 2.0 CMS/TMS use case

Pedro L. Díez Orzas[1], Karl Fritsche[2], Mauricio del Olmo[1], Stephan Walter[2]
[1]Linguaserve I.S. S.A., Seminario de Nobles, 4
28015 Madrid, Spain
[2]Gutleutstraße 30
60329 Frankfurt am Main, Germany
pedro.diez@linguaserve.com, Karl.Fritsche@cocomore.com,
mauricio.delolmo@linguaserve.com, stephan.walter@cocomore.com

Abstract
This paper is the description of how ITS 2.0 allows a better integration between a CMS and a TMS when translating content stored in a CMS, and how the localization workflow of the contents benefits from each ITS 2.0 data category implemented.

This use case has been developed during the MultilingualWeb-LT project that receives funding by the European Commission (project name LT-Web) and in the W3C MultilingualWeb-LT Working Group.

We will exemplify how the contents are generated in Drupal, a Content Management System (CMS). Before they are sent, the contents are annotated with ITS 2.0 metadata in two ways: automatic annotation and manual annotation. XHTML + ITS 2.0 is used as interchange format. Once created, they are sent to the Linguaserve Global Business Connector Server (GBC Server) translation server, processed in the Linguaserve internal localization workflow Platform for Localization, Interoperability and Normalization of Translation (PLINT). Afterwards, once the annotated content is translated and the metadata is treated, they are downloaded by the client and imported into the CMS. The ITS 2.0 selected data categories for integration are: Translate, Localization Note, Domain, Language Information, Allowed Characters, Storage Size, Provenance, and Readiness (ITS 2.0 extension).

Keywords: *Multilingualweb, Internationalization Tag Set 2.0, ITS 2.0, web localization, interoperability, Content Management System, Translation Management System, metadata, web translation.*

1. Introduction

The large volume of information and web content justifies the use of CMS systems for medium to large companies and organizations. They provide benefits as content control, several user profiles, abstraction and workflows.

When we introduce the multilingual variable to the CMS picture, a translation workflow is highly recommended. The advantages of using an external localization provider and computer assisted and automated Translation tools gives added value as the use of translation memories, glossaries and the experience with translation management.

This paper will exemplify how ITS 2.0 allows a better integration between CMS and TMS and how the localization workflow of the contents benefits from each implemented data category.

2 CMS and TMS Integration with ITS 2.0

In the setup described in this paper, Cocomore and Linguaserve have worked together with a real customer, the "VDMA - Verband Deutscher Maschinen- und Anlagenbau - German Engineering Federation" (www.machines-for-plastics.com/kug/). The languages combinations were German into French and into Chinese, and around 75,000 words were enriched with metadata, translated and processed.

The basic steps of an ITS 2.0-aware content creation and translation- process are as follows:

- VDMA has content produced in the Drupal CMS.
- Before being sent, the content is annotated with ITS 2.0 metadata by using automatic and manual annotation. This localization workflow is an XML based tool chain; hence, XHTML + ITS 2.0 is used as the interchange format.
- The content is sent to the Linguaserve Global Business Connector Server (GBC Server),

processed in the Linguaserve internal localization workflow "Platform for Localization, Interoperability and Normalization of Translation" (PLINT).

- The ITS 2.0 metadata is used during the LSP (Linguaserve) internal processing for several localization tasks (providing context to the translators, blocking the non-translatable contents in the CAT tool, selecting terminology and translation memories…) and also updated in some cases as a result of the process (Provenance: the translator and proofreader that have done the job).
- Afterwards, once the annotated content is translated and the metadata is treated, they are downloaded by the client and imported into the CMS.

local metadata is added by hand. In addition, automated annotation tools can be integrated through a standardized interface to support the user in creating such local markup.

Manual annotation features are available in all generally expected interaction modes (toolbar buttons, context menu, keyboard shortcuts).

Two annotation approaches are supported:

a) Annotation may be done as part of the content creation process, via features that have been added as plugins to the out-of-the-box Drupal WYSIWYG editor (see Fig. 2).

b) Annotation may be carried out as a separate step, without the ability to modify the content. This allows workflows that separate content know-how and translation management.

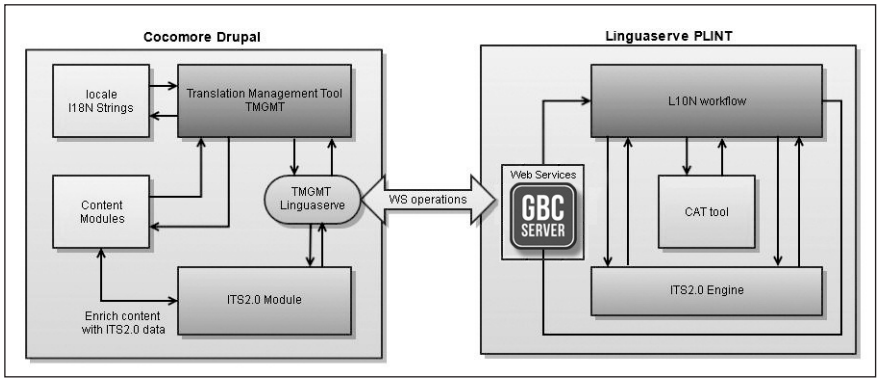

Figure 1. CMS-TMS ITS 2.0-aware architecture

This integrated approach affects practically all areas of the traditional translation workflow. Accordingly, it requires modifications and extensions throughout the tool chain. Fig. 1 shows a vision of the architectural entities that are involved in ITS 2.0-aware content and translation handling.

3. ITS 2.0 Roundtrip

Some of the features of the solution that we created based on this architecture can be assigned to either the content provider's or the LSP side of the picture.

3.1 Content provider's side

On the content provider's side the creation of the ITS 2.0 metadata aware workflow involves the following areas:

1 *Annotation of source language content with ITS 2.0 metadata within the Drupal CMS.* Structural annotation rules can be specified as global rules on a page/content type level, while

2 *Transparent data round-tripping* Triggered from within Drupal, this is carried out in the background via export/import of files XHMTL+ITS 2.0 markup, to be automatically

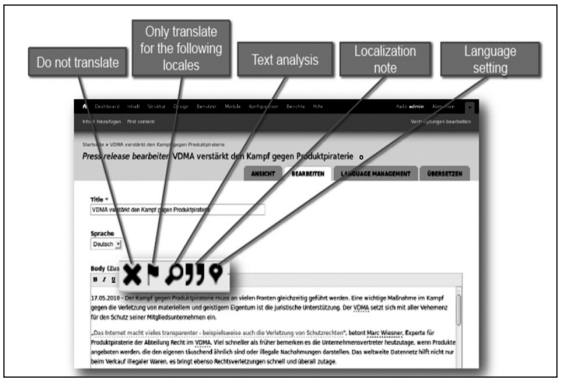

Figure 2. Edit menu for local ITS markup

sent to/received from the LSP. The process is based on an extended version of the Drupal translation Management (TMGMT)- module.

3 *Translation review*
ITS 2.0 markup is retained in this step so that annotated information can be taken into account for QA purposes.

3.2 LSP side

On the LSP side, the creation of the ITS 2.0 metadata aware workflow encompasses three areas:

1 *Pre-production/post-production engine for processing content files annotated with ITS 2.0.*
2 *LSP internal localization workflow to provide support to project management and production processes.*
3 *Computer Assisted Translation (CAT) tool usage for translation, proofreading and post-editing with ITS 2.0 annotated content.*

Fig. 3 illustrates the life cycle of each data category in the complete roundtrip.

4. ITS 2.0 Implemention in the CMS

Cocomore integrated ITS 2.0 into the open-source Content Management System (CMS) Drupal. This required the development and adaptation of several modules:

- Drupal TMGMT-module (extension to allow workflows with ITS 2.0 annotation)
- Drupal WYSIWYG editor: Plugin for ITS 2.0 annotation
- JQuery plugin for ITS 2.0 annotation in a separate step (new implementation)
- Interfacing with Global Business Connector Contents (GBCC) and web services (implementation of data export/import and client implementation)

4.1 ITS 2.0-aware translation workflow in the CMS

4.1.1 Workflow management with TMGMT

The workflow of translation and ITS 2.0 handling within the open-source CMS Drupal can be done by extending Drupal with modules, and there are already a couple of modules available to help the user with translation processes.

We used and extended the "Translation Management Tool" (TMGMT). This module provides the basic translation workflow, which comprises the following steps:

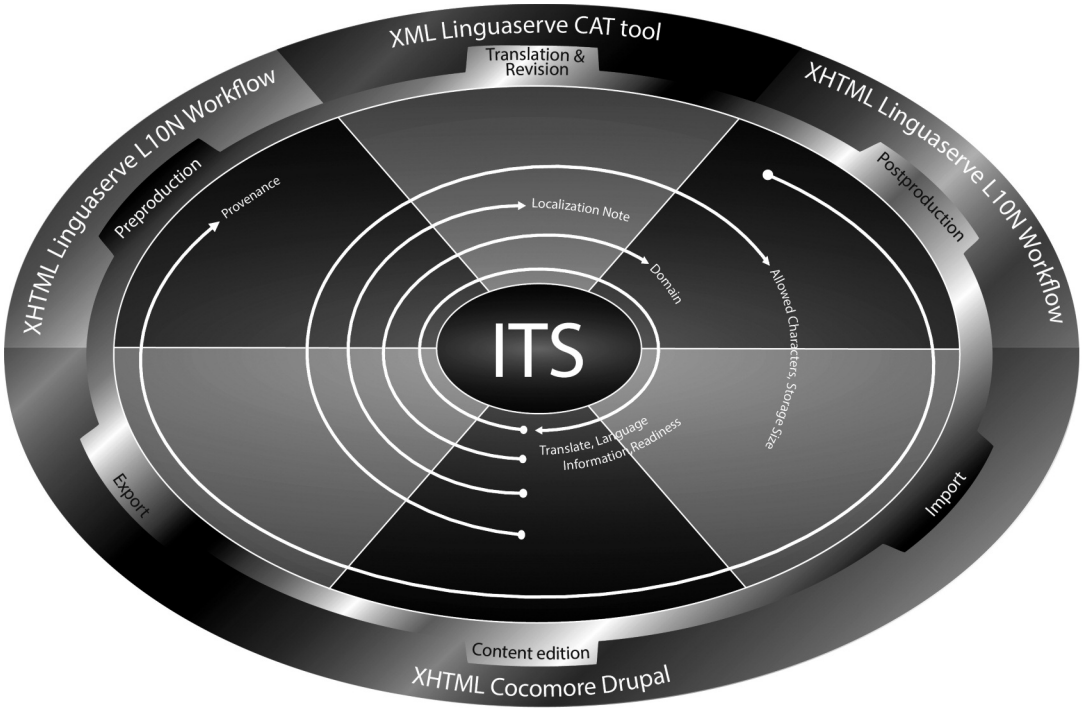

Figure 3. ITS2.0 data category roundtrip

1. Create translation job
2. Send job to translation service
3. Reintegrate translated job into Drupal

Furthermore, TMGMT is designed to work with any content and any translation service. It provides interfaces for handling different sources and services. These are complemented by a default implementation of the source interface, which creates TMGMT jobs from content pages (called nodes in Drupal). For strings that are not part of Drupal nodes (such as menu links, error messages, etc.) we created an additional implementation, which generates a TMGMT job for the untranslated strings in the Drupal CMS, this module will be part of TMGMT in the future. To operate with the Linguaserve Web Service we implemented a translation service for TMGMT to send TMGMT jobs to Linguaserve and retrieve these jobs when they are finished. With these modules the general translation workflow is covered. The described functionality is implemented in the Drupal module TMGMT Workflow.

The TMGMT Linguaserve module which is a translation service for TMGMT handles all SOAP calls to Linguaserve and creates an XHTML file from a TMGMT job. This XHTML file is used as exchange format between Cocomore and Linguaserve. This file uses script-tags for global data categories and the normal HTML markup as described in the ITS 2.0 and only contains the content, no menu or styling information. In this way it can be easily interpreted by other services. The described functionality is implemented in module *Drupal TMGMT Translator Linguaserve*.

4.2 ITS 2.0 Annotation

4.2.1 Local markup via WYSIWYG

For the integration of ITS 2.0 we had to develop another module. This module provides the integration of ITS data categories into Drupal. It extends the WYSIWYG editor with new buttons to allow the user to add and edit local ITS markup in content pages. The following ITS data categories can be set with the WYSIWYG while creating or editing a content page:

- Translate
- Locale Filter
- Text Analysis
- Localization Note
- Language Information
- Directionality
- Terminology

The described functionality is implemented in module *Drupal ITS 2.0 Integration*.

4.2.2 Support for global markup

Apart from being able to set these data categories as local markup, there are also a few data categories that can act as global markup. Support for such global markup is managed on a per-content-type basis. Enabling ITS support for a given content type creates a new section in the edit form for content of this type. In this section, global XPath rules can be entered. It is possible to set default global rules for each content type or globally for the complete site.

For global markup the following data categories are available:

- Domain
- Translate
- Localization Note
- Revision/Translation Agent (from the Provenance data category)

The described functionality is also implemented in module *Drupal ITS 2.0 Integration*, but has to be enabled manually after installation

4.3 Annotation as a separate workflow step

4.3.1 Functionality

In extension to the normal WYSIWYG editor in the content edit form we added a new "Language Management" form. The form provides an editor to only work on (add, remove change) the ITS 2.0 markup of a node, while the actual content is all write-protected. This supports a separation of content editing and ITS 2.0 annotation into two distinct workflow steps: A special user role (e.g. a translation manager) can add ITS data very easily after content creation without accidentally changing the content itself. This role will also be able to see and can edit the global markup.

4.3.2 User Interface

Local and global markup can be highlighted separately in the content. This is controlled in the UI by using checkboxes. In this way the user can choose what he wants to see and doesn't get overwhelmed with all data categories at once. If the user selects content next to the selection, a small window pops up. In this window the user can choose a data category to add to the selected content. There are also keyboard shortcuts available for the data categories to support even faster tagging. For the simple data categories like "Translate" they just add the attribute with the most

contextually likely value. For instance, a translate attribute will be set to the negation of the value pertaining to the current context, thus translate=”no” within text that is not in the scope of any other translate attribute (because the default translate=”yes” is assumed for such text). For all other data categories a new modal window appears where the needed data can be edited, like the note and note type for “Localization Note”.

4.3.3 Implementation

The functionality described above depends highly on JavaScript and is built on top of the ITS 2.0 jQuery plugin, which was also developed by Cocomore. This jQuery plugin provides a functionality for the selection of text nodes with special data category values, and for getting the ITS values of a text node. It is released independently of the Drupal modules. Thus other frameworks or users can use it in their implementations as well. For example a programmer can quickly get all non-translatable text nodes of a HTML and XHTML page to add special styles to it. The plugin correctly handles both local and global markup, including global markup in a script tag and external linked global markup. The module performs all the ITS 2.0 tests.

4.3.4 Data categories with automatically determined values

There are several data categories that have a special status when integrating ITS 2.0 in a CMS due to the fact that they allow for an especially high degree of integration. This may be because the CMS provides specific means for handling them out of the box, or because adequate values for them can be derived automatically from other information that is available from various sources within the CMS and workflow. This special status is also reflected in the Drupal ITS 2.0 integration module.

For the Domain data category you can select that the area where the user can type in the domain shouldn’t be a text field, instead you can use the taxonomy system from Drupal. With this you can create your own vocabulary or use an existing one and just select the domains on content editing. The Provenance data can’t be edited by the user, it just shows and stores this information and it will be automatically set by the translation service. In a similar manner, additional data categories are embedded in the translation process. Data categories like Allowed Characters, Storage Size and Readiness from the ITS extension will be added automatically to the content sent to the LSP depending on Drupal’s field definitions of a particular field. As an example, there is a maximum length of 255 characters for the title field, and in this case the storage size category is added to the title field with the respective values set. The user doesn’t have to care about this at all. As another example, the expected finalization date and priority are added by the translation manager before the translation job is submitted to the LSP.

The described functionality is implemented in module *Drupal ITS 2.0 Integration*.

5. ITS 2.0 Implementation in the TMS

This section explains which ITS 2.0 data categories have been implemented, their usage and application on the different phases of the localization workflow.

There are also explanatory details on their implementation and examples of ITS metadata.

The Java classes involved in the ITS 2.0 processing of the contents are three:

- To manage the paths of the files and the data base records.
- To parse the documents and traverse the nodes in the pre-production and post-production phases.
- To provide the methods related with ITS 2.0 data categories integration.

A general view about the use of each data category in the Linguaserve localization workflow is shown in tables 1, 2 and figure 4.

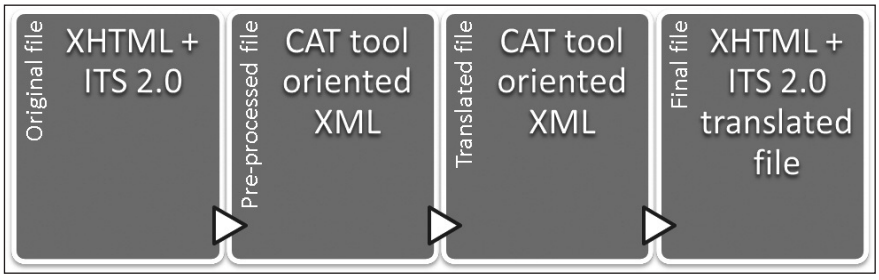

Figure 4. Format Transformation Workflow

Pre-production phase			
Data category	**L10N workflow**	**XHTML Global**	**XHTML Local**
Translate	-	Omit selected not translatable contents.	A particular node could be not translatable.
			Mark parts of the content marked as not translatable for blocking.
Localization Note	When alert type, send a notification to the project manager and add tooltip visualization in the workflow.	Create reference node to inform the translator.	Inform the translator.
Domain	Automatic selection of terminology and translation memories.	Create reference node to inform the translator.	-
Language information	Quality check to ensure the source language content is according to the webservice parameter.	-	Inform the translator.
Allowed Characters	-	-	-
Storage size	Quality check for the original content.		Inform the translator.
Provenance	Possibility to reassign the same translator/proofreader in new versions of the same content (based on identifiers).	-	-
Readiness (*)	Priority checked with webservice.	-	-

Table 1. Data Category Treatment in the Internal Pre-Production Phase

Pre-production phase			
Data category	**L10N workflow**	**XHTML Global**	**XHTML Local**
Translate	-	Omit selected not translatable contents.	A particular node could be not translatable.
			Mark parts of the content marked as not translatable for blocking.
Localization Note	When alert type, send a notification to the project manager and add tooltip visualization in the workflow.	Create reference node to inform the translator.	Inform the translator.
Domain	Automatic selection of terminology and translation memories.	Create reference node to inform the translator.	-
Language information	Quality check to ensure the source language content is according to the webservice parameter.	-	Inform the translator.
Allowed Characters	-	-	-
Storage size	Quality check for the original content.		Inform the translator.
Provenance	Possibility to reassign the same translator/proofreader in new versions of the same content (based on identifiers).	-	-
Readiness (*)	Date control for availability and delivery.	Update the data category node.	-

Table 2. Data Category Treatment in the Internal Post-Production Phase

```
Der <span translate="no">VDMA</span> setzt sich
          mit aller Vehemenz für den
        SchutzMitgliedsunternehmen ein.
```

Example 1

5.1 Use of ITS 2.0 data category

5.1.1 Translate

a) Translate in the pre-production phase
A method that obtains the ITS global rules and another method that obtains the global translatable rules from the ITS global rules were implemented.

After that, the global translate rules (translate="yes") and the global non-translate rules (translate="no") are stored in two different objects. The document nodes are traversed and for each node:

If a global translate rule applies to the node (xpath) then the current state of translate is updated for direct application and inheritance. In the local translate rules, the current state of translate and the defaults are checked to know the treatment of the node. The current state is also accordingly updated for inheritance: if the node is not translatable, jump to the next node; else, if the node is translatable, mark the node as translatable, then extract the content.

For local rules application, if any, traverse the HTML content, add tags for blocking content with translate="no" in the CAT tool, and put the content in the CAT tool oriented XML in a translate node. See example 1.

b) Translate in the post-production Phase
First, we traverse the nodes of the document, if the node was marked as translatable, besides recuperate the translation from the translated CAT tool oriented XML.

If there is HTML mark-up in the content, remove the marks for blocking non-translatable parts and insert the translation in the document. See example 2.

```
<span translate="no">VDMA</span> milite avec
véhémence la protection de ses sociétés membres.
```

Example 2

5.1.2 Localization Note

a) Localization Note in the pre-production phase
A method obtains the ITS global rules, while another method obtains the global localization note rules from the ITS global rules. A third method obtains all the localization notes of alert type.

If there is at least one *alert* type localization note, an e-mail is sent to the project managers and the comments of the file are updated in the database of the system for tooltip visualization in the localization workflow. After that, the document nodes are traversed and for each node:
If a global localization note rule applies to the node, then a reference node is created in the CAT tool oriented XML for the translators/proofreaders. See examples 3 and 4 (global and local usage).

5.1.3 Domain

a) Domain in the pre-production phase
A method obtains the ITS global rules, another method obtains the global domain rules from the ITS global rules, and a third one stores the domains associated with the file in the system's

```
<its:locNoteRule locNoteType="description"
        selector="/h:html/h:body">
            <its:locNote
translate="no">Pressemitteilung</its:locNote>
```

Example 3

```
<span its-loc-note="Bitte korrekte sinngemäße
Übersetzung mit Marc Wiesner absprechen." its-
loc-note-type="description">Internet donne de
   transparence à beaucoup de choses, dont
      notamment à la violation du droit
            d'auteur</span>
```

Example 4

database.

In this way, the document nodes are traversed and for each node:

If a global domain rule applies to the node, then a reference node is created in the CAT tool oriented XML for the translators/proofreaders. See example 5.

b) Domain in the CAT tool project creation step

```
<meta name="DC.subject" content="'Angewandte
Wissenschaft', 'Unternehmen', 'Maschinenbau',
'Allgemein', 'Anlagenbau', 'Kunststoff- und
    Gummimaschinen', 'Technologien'"/>
                    […]
                <its:rules>
              <its:domainRule
domainPointer="/h:html/h:head/h:meta[@name='DC.s
ubject']/@content" selector="/h:html/h:body"/>
                </its:rules>
```

Example 5

The domains associated with each selected file are retrieved and listed.

The dictionaries corresponding with each domain are obtained and associated to the CAT tool project.

The paths of the translation memories corresponding with each domain are obtained and associated to the CAT tool project. In this manner, the CAT tool can then use the dictionaries selected based on the domain values.

c) Domain in the CAT tool project export step

When the translation and proofreading tasks have ended in the CAT tool, the files are exported. In this step, when the CAT tool project is closed, the memory files are stored in the paths corresponding with each domain. The translated files advance in the localization workflow to the post-production phase.

5.1.4 Language Information

a) Language Information in the pre-production phase

The document nodes are traversed and for each node:

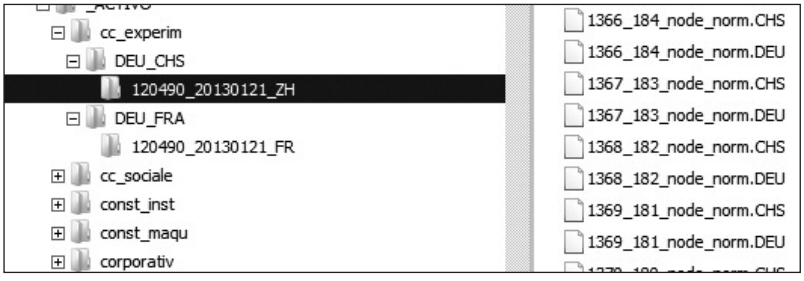

Figure 5. File system showing the translation memories classified by domains

If there is language information, it is checked to see if it is the same than the source language information declared in the system; if not, a warning for the project manager is shown in the workflow. See example 6.

```
<body id="36672" lang="de">
```

Example 6

b) Language Information in the post-production phase

The document nodes are traversed and for each node:

If the node has language information, update the value of the original language code with the target language code. The same process is made in the contents with HTML, but only within the parts that have been translated. See example 7.

```
<body id="36672" lang="fr">
```

5.1.5 Allowed Characters

a) Allowed Characters in the post-production phase

Here, the document nodes are traversed and for each node:

If the allowed characters restriction is declared, it is checked with the regular expression, but if the restriction is not fulfilled, an exception is raised, the process is aborted and the user is informed about the reason. See example 8.

```
<div id="36672-node_title" its-allowed-characters="[^<>]">VDMA renforce la lutte contre le piratage des produits</div>
```

Example 8

5.1.6 Storage Size

a) Storage Size in the pre-production phase

We traverse the document nodes and for each node:

If the node is translatable and has storage size limitation declared, the maximum size is informed in an attribute of the translatable nodes of the CAT tool oriented xml. The size of the original content (in another attribute) is also reported. This information will be available for the translators/proofreaders in the CAT tool. The size is calculated using the encoding.

It is also checked if the original content fulfils the restriction and, if not, a warning is shown to the project manager. See example 9.

```
<div id="36672-node_title" its-storage-size="255">VDMA verstärkt den Kampf gegen Produktpiraterie</div>
```

Example 9

b) Storage Size in the post-production phase

In the post-production phase, if the node is translatable and has storage size limitation declared, a method checks the maximum storage limitation compliance, for which it also takes into account the encoding declared for the content. See example 10.

5.1.7 Provenance

a) Provenance in the pre-production phase

Here, for each node, if there is provenance information available from a previous translation, the database is updated to register the translator and the language pair. On the other hand, if there is provenance information available from a previous proofreading, the database is updated to register the proofreader and the language pair. See example 11.

```
<div id="36672-node_title" its-storage-
size="255">VDMA renforce la lutte contre le
piratage des produits</div>
```

Example 10

```
<body id="36814" its-org="Linguaserve" its-
person="21686" its-rev-org="Linguaserve" its-
rev-person="20697">
```

Example 11

b) Provenance in the translation CAT tool phase
The system proposes the project manager the last translator who performed the same task for the same language pair.

Figure 6. Page of CAT Workflow - Translation

c) Provenance in the revision CAT tool phase
The system proposes the project manager the last proofreader who performed the same task for the same language pair.

Figure 7. Page of CAT Workflow - Revision

d) Provenance in the post-production phase
The attributes related with provenance information are updated: the translator, the proofreader and the organization that has done the job. See example 12.

5.1.8 Readiness (ITS 2.0 Extension)

a) Readiness in the pre-production phase
A method obtains the ITS global rules and another method obtains the global readiness rules from the ITS global rules were created.

The expected delivery date is updated in the system, taking into account the time zone, and the priority of the translation is checked with the information available in the system. If there is no concordance, a warning for the technical department is shown in the workflow. See example 13.

a) Readiness in the post-production phase
The date of availability for the next step in the chain is updated (attribute *ready-at*) having into account the time zone.

The attribute with the processes to be done is updated (attribute *ready-to-process*), removing the completed tasks (human translation and proofreading for quality assurance).

```
<body id="36672" its-org="Linguaserve" its-
person="21686" its-rev-org="Linguaserve" its-
rev-person="20697">
```

Example 12

```
<itsx:readinessRule ready-at="21/01/2013
13:48:56:000 CET" priority="1/3" complete-
by="19/02/2013 16:00:00:000 CET" ready-to-
process="hTranslate, reviseQA, hReview,
publish"/>
```

Example 13

```
<itsx:readinessRule complete-by="19/02/2013
  16:00:00:000 CET" priority="1/3" ready-
 at="30/01/2013 17:46:27:744 CET" ready-to-
       process="hReview, publish"/>
```

Example 14

If the file is processed after the expected delivery date, a warning for the project manager is shown in the workflow. See example 14.

6. Links to Information

6.1 Drupal documentation of components

All the implementations are released under the GNU General Public License 2 and can be downloaded and modified.

They are available at the following URLs:

- Drupal Community 'kfritsche's sandbox' (2103) 'Drupal TMGMT Workflow' [online], available: https://drupal.org/sandbox/kfritsche/1908598 [accessed 22 Oct 2013].
- Drupal Community 'kfritsche's sandbox' (2103) 'TMGMT Translator Linguaserve' [online], available: https://drupal.org/sandbox/kfritsche/1908422 [accessed 22 Oct 2013].
- Drupal Community (2103) 'Drupal ITS 2.0 Integration module'[online], available: http://drupal.org/project/its [accessed 22 Oct 2013].
- The jQuery Foundation (2013), ' jQuery ITS 2.0 Parser Plugin' [online], available: http://plugins.jquery.com/its-parser/, [accessed 22 Oct 2013].

The ITS-Drupal module uses the ITS 2.0 jQuery Plugin, which we published separately for users who do not use Drupal as their CMS, but want to work with ITS 2.0 in an HTML context too. This plugin is tested with the W3C ITS 2.0 Test-suite and conformant to the standard.

Our Drupal implementation is extensible with other modules. Cocomore developed an interface to allow other systems to do work before or after a translation. This can for instance be used to add a QA service after the translation is done or to integrate a service for additional automatic annotation of ITS 2.0 metadata. An implementation that integrates an *Enrycher*

Figure 8. Pre-production/post-production engine for Drupal XHTML files with ITS 2.0

service to generate text analysis markup is accessible at the following URL:

- Drupal Community 'kfritsche's sandbox' (2103) 'Drupal Enrycher Integration' [online], available: https://drupal.org/sandbox/kfritsche/1966286 [accessed 22 Oct 2013].

6.2 Videos and demo of the TMS Processing

- MultilingualWeb-LT (2013) 'L10n workflow interaction for the pre-production phase' [video online], available: http://www.w3.org/International/multilingualweb/lt/wiki/images/6/67/Linguaserve_Preproduction_step_demo.zip [accessed 22 Oct 2013].
- MultilingualWeb-LT (2013) 'CAT tool usage with ITS 2.0' [video online], available: http://www.w3.org/International/multilingualweb/lt/wiki/images/c/ca/Linguaserve_ITS_CAT_Tool_usage_demo.zip [accessed 22 Oct 2013].
- MultilingualWeb-LT (2013) 'L10n workflow interaction for the post-production phase' [video online], available: http://www.w3.org/International/multilingualweb/lt/wiki/images/a/aa/Linguaserve_Postproduction_step_demo.zip [accessed 22 Oct 2013].
- MultilingualWeb-LT (2013) 'Demonstration of how the pre-production/post-production engine for Drupal XHTML files with ITS 2.0' [online], available: https://www.linguaserve.net/las_demos/control/MLWLTWP3DemoEngine (user: demos; password: demosLingu@serve) [accessed 22 Oct 2013].

Acknowledgements

Many thanks to the MultilingualWeb-LT project (funded by the European Commission, through the Seventh Framework Programme in the area of Language Technologies, project name LT-Web, number FP7-ICT-2011-7-287815), to the W3C MultilingualWeb-LT Working Group, and to the Internationalization Tag Set Interest Group.

Also, thank you very much to Laura Guerrero, Giuseppe Deriard, Pablo Badía, and Felix Fernández (from Linguaserve), and Clemens Weins (from Cocomore), who participated in this use case.

References

ITS Interest Group (2013) 'Access to many different presentations and videos on ITS 2.0 and implementations' [online], available: http://www.w3.org/International/its/wiki/Outreach_material [accessed 22 Oct 2013]

The Apache Software Foundation (2013) 'Apache OFBiz' [online], available: http://ofbiz.apache.org/ [accessed 22 Oct 2013]

Drupal Community (2013) 'Drupal Official Web' [online], available: http://drupal.org/ [accessed 22 Oct 2013]

MultilingualWeb-LT Working Group (2013) 'ITS 2.0. 2013. *Internationalization Tag Set (ITS) Version 2.0. W3C Last Call Working Draft 24 September 2013' [online],* available: http://www.w3.org/TR/its20/ [accessed 22 Oct 2013]

del Olmo, M., Guerrero, L., Díez Orzas, P.L., Deriard, G., Badía, P., Fritsche, K., Weins, C., Walter S. (2013) *B2B integration showcase. Multilingualweb-LT Deliverable D3.2.2, public report* [online], available: http://www.w3.org/International/multilingualweb/lt/wiki/Deliverables [accessed 22 Oct 2013].

W3C (2004) 'SOAP specifications' [online], available: http://www.w3.org/TR/soap/ [accessed 22 Oct 2013].

W3C (2013) 'Internationalization Tag Set Interest Group' [online], available: www.w3.org/International/its/ig/ [accessed 22 Oct 2013].

W3C (2013) 'MultilingualWeb-LT Working Group' [online], available: www.w3.org/International/multilingualweb/lt/ [accessed 22 Oct 2013].

W3C (2002) 'XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition)' [online], available: http://www.w3.org/TR/xhtml1/ [accessed 22 Oct 2013] .

W3C (2008) 'Extensible Markup Language (XML) 1.0 (Fifth Edition)' [online], available: http://www.w3.org/TR/xml/ [accessed 22 Oct 2013].

W3C (1999) 'XML Path Language (XPath)' [online], available: http://www.w3.org/TR/xpath/ [accessed 22 Oct 2013].

Generalizing ITS as an Interoperable Annotation Technique for Global Intelligent Content

Dave Lewis, Leroy Finn, Rob Brennan, Declan O'Sullivan and Alex O'Connor
Centre for Next Generation Localisation
Trinity College Dublin, Ireland
dave.lewis@scss.tcd.ie, finle@tcd.ie, rob.brennan@scss.tcd.ie,
Declan.osullivan@scss.tcd.ie alex.oconnor@scss.tcd.ie

Abstract

This paper considers how the interoperable content annotation techniques developed to address the needs of localization processing chains could be applied to a broader class of content processing. We extract the content annotation patterns developed for the Internationalization Tag Set standards at the W3C. These provide a means for annotating content with common meta-data that addresses different aspects of content localization from content creation, through extraction, segmentation, terminology management, automated translation, post-editing, quality assurance to publication of the translated content. This paper explores the lessons learnt in developing ITS 2.0 as a suite of interoperable content annotation in the form of a pattern language. Interoperability problems arise when end-to-end content processing spans different: content formats; content processing tools and engines; and content processing service providers. This paper aims to make it easier to leverage these annotation patterns in the same way across these different interoperability mechanisms. In particular we propose annotations that follow the ITS annotation patterns but address personalization content processing. From this proposal the potential for integrated localization and personalization processing is considered.

Keywords: *ITS 2.0, Interoperable Annotation Technique, Internationalization Tag Set, W3C*

1. Introduction

One of the most significant changes to people's lives in recent years has been the explosion of content available to users, enterprises and communities via the Web. Enterprises and users have adopted new roles as creators, curators and consumers of content, in social and corporate contexts. Increasingly, organizations, communities and individuals seek to access content not only in their own language, but also according to their own needs, preferences and context. Fundamental challenges must be addressed, however, if content is to be dynamically created, curated, processed and delivered for consumers in global markets. The content processing value chains that deliver content from creators to consumer must address the volume, velocity and variety of content. The increased volume and velocity with which enterprises, institutions and users generate content requires new levels of automation to maximally leverage the limited capacity for professionals to exercise appropriate linguistic judgments in processing content from creator to consumer, e.g. translating content or quality assuring content for consistency. Language technologies such as machine translation, text classification, and named entity recognition can support such automation, but only if used at the appropriate stages in the content processing chain and only if tailored to the characteristics of the content being processed and the need of the targeted consumers. A major interoperability challenge however is the variety that exists in content formats used and in the linguistic domains, lexica and styles exhibited by content. This currently limits the efficiencies possible through language-technology automation, both in terms of consistently processing unstructured content and in training language technology to a particular content stream.

We propose a new unifying concept called 'Global Intelligent Content' as a basis for addressing these interoperability challenges. This concepts calls for embedding new levels of interoperable knowledge and intelligence into content to enable advanced intelligent content services to automatically process and transform that content in a more consistent and responsive manner. These intelligent content services will combine data driven language technologies and semantic reasoning capabilities. In this way, Global Intelligent Content will be more discoverable, semantically rich, adaptable, contextually aware and reusable across different granularities across global markets, right down to the individual. Global

Intelligent Content should therefore be dynamically transformed based on current user interaction, perceived user intention or current delivery context.

We identify the Global Content Value Chain as the business context for the processing of multilingual content from creation through to consumption (Emery et al, 2011). The central premise of the chain is that value can be added to content as it moves through the chain by leveraging of human judgments in combination with intelligent content service components. Today's Global Content Value Chain is best exemplified by the need to integrate between enterprise content management systems and the language services industry. Here workflows focus on enterprise-driven content creation, localization, management and publication functions. However, these value chains typically employ predefined workflows and complex decision making to pass content through the processing chain. The need to handle content variety often leads to specialization in the value chain, where companies, often SMEs leverage niche human skills (e.g. domain-specific translation in a certain language pair) or the specialized knowledge needed to leverage specific language resources using language technologies, e.g. a specific domain lexicon or bi-lingual corpora. This specialization however heightens the need for smooth interoperability since otherwise the overhead of manual intervention required for the exchange and processing of content will inhibit the growth of the market.

In this paper we examine the interoperability requirements of two important classes of content processing that we regard as key to the formation of global content management chains, namely *localization and personalization*. Localization is the industrial process of adapting content to a target locale. This is primarily concerned with the translation of textual content, but may also involve the adaptation of images; currency, date and other data formats and layouts to the norms of the target market. Personalization describes a range of techniques used to adapt content to an individual user's needs. It depends on a user model and employs techniques of navigation adaptation (hiding or prioritization of hyperlinks), adaptive discovery (adapting content indexing and queries), content adaptation (e.g. selection and filtering of content elements) and content composition (Levacher at al 2009, Koidl et al 2011, Wade 2009). Currently, Localization is the more mature field in terms of interoperability standards. We therefore review existing approaches to standards to examine the content annotation solutions they offer that might best provide common content meta-data that may persist across a workflow of heterogeneous components. From this analysis we see that the approach to content annotation defined in the Internationalization Tag Set (ITS) standard from the W3C (Savourel et al 2008) best addresses the needs of interoperable content annotation. We then extract these annotation techniques, based on the current ITS2.0 specification, to generate a set of reuable annotation patterns. We end by proposing new personalization-specific meta-data that could exploit these patterns to provide interoperable content meta-data annotation specifications.

2. Content Interoperability Challenges

At its simplest, content can be regarded as digital media specifically created by people with the express intent to be consumed by other people (thereby allowing us to distinguish it from digital data either solely generated or solely consumed by automated systems). When considering content communicated via the web, it will typically consist of unstructured content such as text, audio or video accompanied by structuring markup and by meta-data which serves to annotate both unstructured content and the markup. The mark-up and annotating meta-data plays a key role in the processing of content, including its transport, indexing, aggregation, selection, filtering, adaptation, composition and presentation. Content interoperability therefore relies on a common understanding of how to process the content markup and annotation that can be shared between different content processing components. It is therefore the extant variety of content mark-up and annotation techniques that makes content interoperability complicated and often expensive to achieve when attempting to form real world content processing chains.

If we consider content on the Web in particular, interoperability has been considerably eased by the widespread adoption of document formats that adopts tree based serializations. This has enabled a common programmatic abstraction for document processing to be standardized in the form of the document object model (Le Hors et al 2004). This in turn has enabled development of common declarative mechanisms for selecting tree nodes within a document (Clarke & deRose 1999) and performing transformations on document contents (Clarke 1999). This has in turn proved powerful in developing content processing chains in enterprise content applications, which typically span web, print and

other content delivery channels. However, for native web content applications these benefits have been diluted somewhat in the drive towards HTML5, which has integrated several elements that dilute common DOM serialization of content to bring benefits of enhanced interactivity and rich content media delivery, e.g. ECMA Script, audio and video content format.

In addition, the Web has experienced the growth of the semantic web and interest in its potential role in content discovery and delivery. The semantic web offers a fine grained graph of data nodes accessible as web resources, i.e. by dereferencing a URI, together with navigable links between these data resources. This has enabled newly standardized mechanisms, such as RDFa (Herman 2013) and schema.org[1], to be employed for interlinking linking web resources in the form of content-bearing documents and external meta-data in the form of linked data nodes. The result is a rich but complex set of mechanisms that can be employed in content processing and which therefore must be accommodated when attempting to implement efficient integration of content processing components into content processing value chains.

This is particularly challenging to the classes of content processing that we are considering in this paper, namely localization and personalization. Both often suffer in practice from being employed in a post-hoc manner, such that downstream localization and personalization processing is not adequately considered in the up-stream content processes where content is created, structured and annotated. This therefore adds to the cost and complexity of localization and personalization processes as they must accommodate and often also preserve the diversity of content mark-up and annotation as they traverse these downstream processes. This is required in order to maintain the validity of assumptions about mark-up and meta-data made in subsequent downstream processing components involved in content publication, indexing, search engine optimization, archiving and reuse. Therefore, making extensive changes to the mark-up of content to accommodate localization or personalization processing may not be an attractive option for enterprise. In the first instance this is because it would prove too disruptive to other downstream processes (including between personalization and localization processes). Also, such changes may result in personalized and/or localized content being 'forked' away from parallel versions of the same content passing through pre-existing content processing chain (e.g. for print publication or search indexing), making it difficult to recombine or reuse that content in future iterations. For this reason, we therefore focus here on the mechanisms available for annotating content for localization and personalization, rather than consider alternative mark-up formats that would ultimately be more difficult to deploy in the context of existing content value chains.

The next section examines the start of the art in open, interoperable content mark-up and annotation specifications for the more mature field of localization, in terms of their capabilities for marking up and annotating content.

3. Analysis of Content Interoperability Mechanisms for Localization

Localization is a well-established part of the content processing chain for many multinational companies. However, content processing value chains involving localization workflows can be varied and complex and overheads due to poor data and meta-data interoperability are estimated as being upto 20%. Moreover, the distribution of providers by size exhibits an extremely long tail, with 99% being SMEs, who therefore struggle to both handle the overhead of poor interoperability and to reap the benefits of large scale language data reuse arising from large volumes of translation traffic.

The localization industry consists of content generating enterprises and the Language Service Providers (LSPs) they contract to translate source content. In recent decades, the main technological innovations to yield productivity improvements in this industry have involved the collection and reuse of language data resources. Specifically these resources take the form of: term-bases (multilingual glossaries that improve consistency in both authoring and translation of terms) and translation memories (databases of previously translated sentences that assist translators in translating identical or similar sentences, phrases or terms). The leverage of translation memories is supported by well-established norms for translation discounts based on the corresponding human translation effort savings. More recently, translation memories (TM) and term-bases are being reused by LSPs as good quality training corpora for Statistical Machine Translation (SMT) engines. Therefore the collection, distribution and reuse of both parallel text and bi-lingual term bases is a key part of the localization workflow.

Poor interoperability experiences arise in many localization workflows due to the multiple parties

involved using a variety of content formats, workflow systems and translation tools. Though there are several standards serving this industry, standardization efforts are somewhat fragmented between several different organizations.

To avoid this fragmentation disrupting our analysis, the interoperability standards examined below are categorized by the type of interoperability function they perform.

Content authoring and publication formats: These include standardized electronic publication formats such as HTML (Berjon et al 2013), OASIS DITA (Eberlein et al 2010) and DocBook[2]. There is however widespread usage of content authoring and publication formats are open in that the specification is published, but are proprietary in that the design of the format is not subject to a consensus forming process that is open to broad industry input and consultation. Examples are PDF, Rich Text Format, Microsoft Office and Open Office formats and Adobe XX formats. Often, authoring is performed in a different format to publication, where HTML and PDF have become dominant. This requirement has made XML content authoring formats more popular, as XSLT declarations can be used and exchanged to offer reliable transforms for authoring to one or more publication formats. This in turn promotes the uptake of component or topic based authoring, where content is authored in discrete units designed to be easily recombined at the publication stage. These formats are not primarily focused on the needs of localization, sometime then requiring supplementary annotations for internationalization and localization purposes. This has been somewhat addressed by the W3C through the standardization of the Internationalization Tag Set (ITS v1.0). This aims to reduce elements of the interoperability overhead cost by defining a set of well-defined independent standard meta-data attributes that can be used to annotate XML content to address specific use cases. These use cases are: whether to translate content or not; where content is a term or not; identifying subflow in text to assist translators; offering localization notes for the translator; providing language information when absent in the source format; and providing directionality and ruby annotation information often needed in non-latin scripts. So while the wide range of source content format is a major source of complexity in localization content processing chains, as ITS is agnostic of the XML format used for the source it can be used consistently, in concert with conformant ITS processors, across any XML format, including bi-text exchange formats discussed below. Further, it defines its annotation, known as data categories, in an abstract manner that is independent of the XML implementation and could be potentially applied to other non-XML formats, though this is not yet in common practice. Addressing this requires the development of content extraction filters, which are needed because the translation processes is performed largely separately from the content authoring and publication processes. This makes translating content in the context of the publication format problematic and also complicates the synchronization of translation processes with ongoing changes made to the source content. The development and maintenance of extraction filters is a complex task, with limited support for open solutions, meaning that extraction components must be developed and used in tandem with reassembly components. Defining content annotation that can be easily processed in content filters is therefore an important objective of ITS.

Language resources: The reuse and leverage of language resources is a key productivity driver in localization processes. Principle amongst these is translation memory, which provides a searchable database of previous translations to avoid effort in replicating similar translation. The Translation Memory Exchange (TMX) standard provides an XML vocabulary for exchanging parallel text (or bi-text) that capture source language content and its translation at the level of segments as used in the translation processes that generated them. TMX is well supported in translation management systems (TMS) and computer assisted translation (CAT) tools. The widespread use of TMX has also prompted its increasing use as a format for providing parallel text to processes training statistical machine translation components. Consistent use of terminology from authoring to translation (human and machine based) and translation review is important in achieving good quality translation. Within the localization process exchange of this information between tools in the form of term bases is supported by the Term Base eXchange XML vocabulary (TBX). ISO has been active in promoting open formats for lexical repositories. In recent years, mapping of these lexical repository formats into the Resource Description Framework (Manola & Miller 2004) that underlies the semantic web, for publishing as linked open data have been explored (Windhouwer & Wright 2012). Other, RDF vocabularies have been proposed for publishing of lexical resources directly as linked open data (Chiarcos 2008, Buitelaar et al 2008). In parallel large open cross lingual and lexical

repositories are emerging, based on existing resources such as Wikipedia and WordNet, with their increasing usage presenting de facto standardization of their vocabularies – reflecting an increasing trend in the development of common formats in the linked open data community.

As natural language technologies have become increasingly viable, there has also been interest in developing language resource formats that can convey the output of language processing, including lexical parsing, semantic tagging and named entity recognition. This has resulted in a proposal for an RDF vocabulary supporting the exporting of language resource resulting from NLP component processing, termed the NLP Interchange Format (NIF) (Hellman et al 2013).

Bi-lingual Tool exchange formats: The various stages of the translation process, e.g. machine translation, TM leverage, post-editing, human translation and translation review, may be undertaken by different workers, service providers each using different tools and processing components. It is therefore important that content and its translations to be passed between reliably between such bi-lingual content processing tools. One approach popular in software UI translation is the user of the PO format for passing translatable content to translation processes and be returned matched with translation. Though a popular format, especially in open source software projects, it does not benefit from an open industry agreement process. A more concerted standardization has been conducted by OASIS in the development of XML Localization Interchange File Format (XLIFF) (Savourel et al 2008). This offers a bi-text exchange format that accommodates a wide range of meta-data needed for the localization process, including integration of TM leverage, human post-editing, translation and review and terminology.

Processing instructions: The effectiveness and fidelity of a localization process chain is particularly sensitive to how certain processes are conducted. In such cases having the ability to exchange instructions between tools and worker in an open format is important. One of the most crucial process instructions is the segmentation of text into translatable segment, since efficient leverage of translation memories requires consistent segmentation. The Segmentation Rule Exchange (SRX) format allows such rules to be exchanged and segmentation outcomes to therefore be accurately reproduced between tools.

Discussion

It can be seen from this brief analysis that interoperability formats for localization suffer from fragmentation in goals, the bodies that produce them, the formats they use, the use case they address and their uptake within the localization process. Two recent initiative has attempted to address this fragmentation.

A small industrial consortium, known as 'Interoperability Now!' (IN!), has formed specifically for the task of developing a Translation Interchange File Format. This defines how several related open formats can be packages and zipped for exchange between tools, including XLIFF, TMX and TBX. While this performs a useful consolidation function, it has progressing in parallel with a revision of the XLIFF standard with many of the same goals, including the restriction of options that was perceived to slow uptake of XLIFF 1.2. In this sense IN! has also served to add to the sense of fragmentation in the industry. A key factor here, which is similar phenomenon in web services interoperability, is that the ease with which a format can be extended using name spaces means that the key concepts represented by the format can be changed. This adds unforeseen complexities to the updates required to third party components intending to implement the extension. The key here is to ensure that the semantic role of different format elements is clearly defined separately from the syntax of the format – however this is a complex task to achieve in practice. The result is a complex set of interlinked XML vocabularies that are carefully tuned to the need of localization process interoperability, but which as a result is poorly suited to more general content processing.

The other initiative has been the Multilingual Web – Language Technology at the W3C. Rather than attempting to develop a broader container format, it follows the approach adopted in ITS1.0 to define independent data categories that annotate existing formats either for stand-alone use cases, or used in combination to support interoperability across the content processing chain, regardless of mapping between different formats used within it. The result is a draft ITS2.0 Recommendation (Filip et al 2013). This expands the implementations of ITS from just XML to include HTML5 and RDF. The key insight, continued from ITS1.0, is that the data being annotated is the textual content of documents. Annotation schemes oriented toward the semantic web and linked open data, i.e. RDFa and microdata, are not well suited to this task as text is treated only

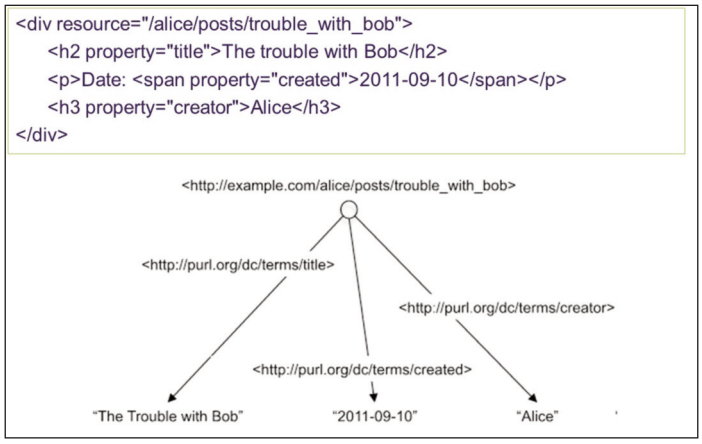

Figure 1: RDFa content annotation

as literal objects of data triples and not the subject of meta-data annotations as outlined in figure 1.

However, to support close integration with content processing and localization tool chains, ITS associated meta-data with textual content either through well-defined attribute added to enclosing elements (e.g. HTML span) or through rule element that associate attributes with enclosing elements (or attributes) using XPath selectors. Well defined inheritance, override and default rules enable dedicated ITS processor functions to be implemented and conformance tests for such processors to be formulated. Ease of adoption is supported by conformance being attainable through implementation of a single data category, presenting a lower cost migration path than the wholesale adoption of a specific source or bi-text interchange format. In addition to the data categories in ITS1.0, ITS2.0 adds further data categories designed to ease the integration of language technologies and linked open data into the localization process. Machine translation integration is supported by annotation of the content's application domain and of automated translation confidence scores. Text analysis is supported with annotation to associate words or phrases with external resources, e.g. DBpedia for classification and definitions or WordNet or BabelNet for lexical definitions. Such annotation may be generated by text analysis components such as Named Entity Recognition (NER) engines. ITS2.0 therefore offers a flexible palette of well-defined data categories to support the generation and consumption of content annotations by multiple processes and the translation workflow, spanning from content creation to its translation, consumption and reuse. In this sense ITS2.0 fulfills a role for the multilingual Web similar to that which the Dublin Core has played for interoperability of monolingual content publishing.

In the rest of this paper we examine the content annotation techniques used in ITS2.0 separately from the semantics of the data categories it defines, with the aim of generalizing these annotations into a set of reusable patterns.

4. Generalizing ITS to Content Annotation Patterns

In considering content annotations that are suitable for deployment in existing content process chains several important principles can be derived:

a) The annotation should minimize impact on the original content so as to minimize the burden on other components in the content processing chain in handling that annotation. Impact can be assessed in terms of complexity.
b) Annotation should be well-defined in an open manner so that they can be successfully exchanged between separately implemented content processing components.
c) The mechanism for associating annotations to content should be flexible enough to accommodate different content mark-up schema, so that the processes using the annotation are not unnecessarily limited to specific content formats.
d) Consistent with point (c), annotation mechanisms should aim to be flexible enough to be associated with new content markup formats, i.e. it should be extensible
e) Consistent with points (b), (c) and (d), annotations should possess unambiguous semantics even when the mechanism for associating the annotation to content varies.
f) It should be possible to reliably remove the

association of the meta-data from the content in situations where, for example, the impact of localization or personalization relate processing is not longer relevant for content reuse or other downstream processes.

The ITS approach seems to address many of the requirements, but to be able to generalize this more formally we deconstruct the various annotations into the following set of patterns. It is important to note that the specification of ITS is not based on these patterns explicitly. Therefore any attempt to build a conformant implementation should follow the ITS2.0 specification. The provisions of those specifications are written, as with any interoperability specification, to maximize the unambiguous interpretation of its provisions when building and testing a conformant implementation. In contrast, the description of patterns presented here convey some core reusable design principles underlying the ITS specifications. The aim therefore is to encourage the development of further interoperability specifications that can avail of the tried and tested interoperable content annotation solutions contained in the ITS specifications, or to extend existing ITS parsers with new data categories. Any such specification would however need to be prepared in the unambiguous manner adopted in an interoperability standard, supported by a conformance test suite.

The following annotation patterns are generalized from the established text annotations mechanism over which consensus has been reached in the standardization of ITS 1.0 and ITS 2.0. These patterns are split between a basic set of patterns concerned with the direct annotation of textual content with attribute values, and those that offer indirect ways of associating annotation values with textual content. The pattern description describes the problem it tries to solve, the constraints under which it must be applied, the advantages of its use and where relevant explains how it is used in ITS2.0.

4.1 Direct Annotation of DOM Structured Content

P1. Annotation of Textual Content in a DOM conformant document

This specifies that annotation of text nodes (i.e. the textual content of element nodes) and the textual content of attribute nodes in a DOM conformant document can be specified by association with well-defined attribute nodes. This can be implemented by a DOM-conformant parser that enacts specific actions when detecting such a special attribute nodes associated with an element or attribute node. See figure 2 for an example of text, element and attribute node in a DOM parse tree.

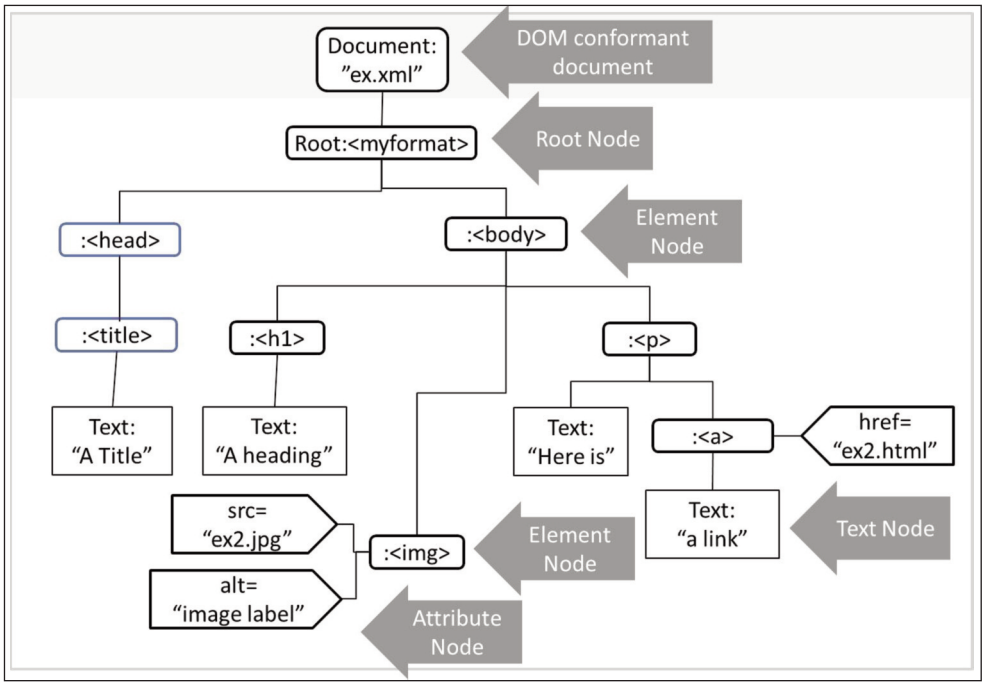

Figure 2: Example of element, attribute and text node in a DOM parse tree

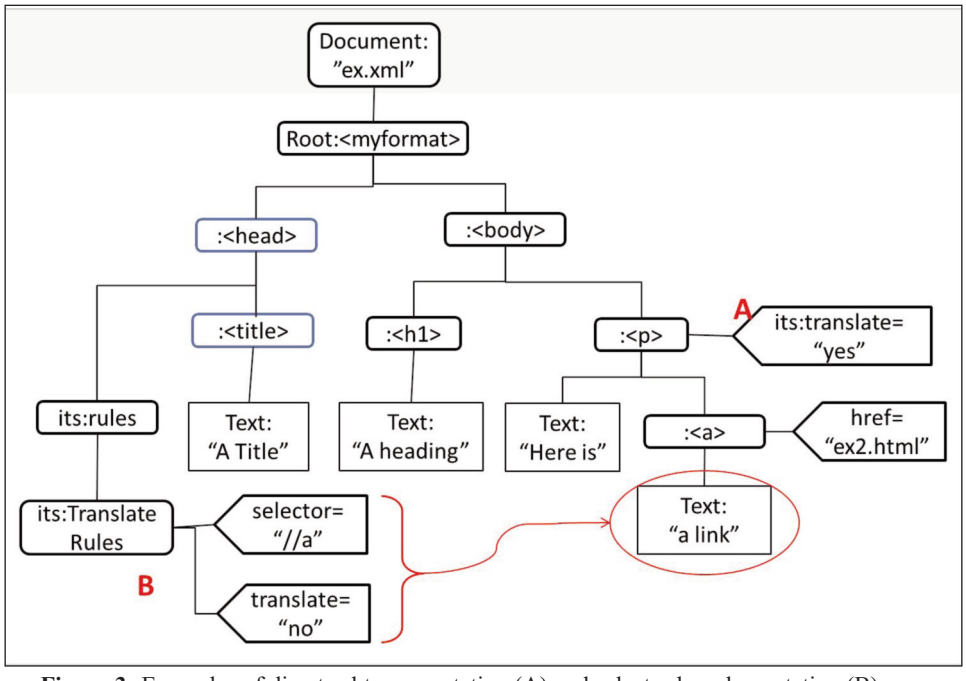

Figure 3: Examples of direct subtree annotation (A) and selector based annotation (B)

This is a base pattern of the pattern language, i.e. all the other patterns rely on this one. This therefore requires that these text annotation patterns can only be applied to DOM conformant documents. The advantage of this pattern is that by using well defined attributes to specify annotations allows these annotations to co-exist with other DOM conformant schemas in a variety of applications.

In ITS, annotating attributes are defined for XML using a specific name space and for HTML by a set of attributes with a common attribute name prefix, i.e. "its-".

P2 Direct sub-tree annotation

In this pattern all the text nodes and text values of attribute nodes within a sub-tree of a document's DOM representation are annotated by a well-define attribute annotating the root element of that sub tree. The advantage of this pattern is that it allows contiguous sub-portions or a document to be easily annotated.

A constraint on this pattern is that the semantics of the annotation may not be appropriate to propagate over the text nodes and/or the text values of attribute nodes across the sub-tree. This propagating behavior therefore must be well defined for specific annotation types.

In the ITS specification, such an annotation is referred to as a local selector. The propagation of ITS annotation from a node to its sub-tree nodes is described in terms of those nodes 'inheriting' the annotation to the annotated sub-tree root element.

P3 Selector-based annotation

This pattern exploits the standardized specification of node selector language that can operate with DOM-conformant language, such as XPath and CSS selectors. An annotation therefore can be associated with a set of nodes by associating it with a selector statement that specifically identifies that set of nodes. A constraint of this pattern is that a new annotating element must be added to the document to house the selector-to-annotator bindings.

An advantage of this approach is that this element can be placed outside of the main content-bearing portion of a document, e.g. in the <head> element of a HTML document. This approach also offers the flexibility to easily annotate a non-contiguous set of parts of a document. Also, as pattern P2 annotates an element it cannot be used to annotate the textual content of an attribute separately to the element which that attribute decorates. Using selector based patterns allow such attribute text values to be individually annotated.

In ITS, selector based annotations are referred to as

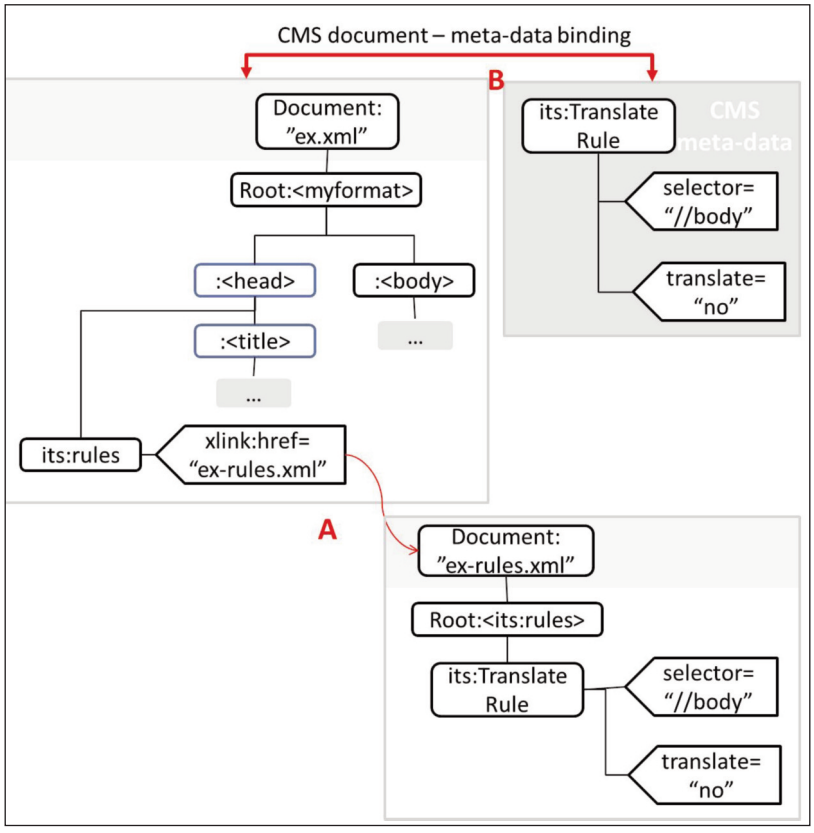

Figure 4: Examples of referenced external selector-based annotation (A) and External binding to selector-based annotation

'global rules-based selection'. They are specified in a defined set of rule elements, which bind a specific annotation type to a specific selector. Rules elements are placed in a defined <rules> element, where multiple rules can be collected. Where rules select overlapping sets of document nodes, the order of the rule declaration is used to determine which takes precedence in parsing ITS annotations.

P4 Referenced External Selector-based Annotation

Selector based annotation rule can be defined in an external file that can be referenced from within a document that uses those rules for annotation.

This has the advantage that the same set of rules can be easily applied in a consistent manner to a whole set of the document. This is useful, for example, when the rules define annotations that relate to a schema used by a number of documents. It also allows the rule in the references files to be modified without altering the referencing files.

In ITS, references to an external file with an its:rules element can be made from an Xlink hyperlink ('href') attribute from an its:rules element within the file. Rules applied in this way have a lower precedence that those declared within a document.

P5 External binding to selector-based annotation

An external definition to selector based annotation may also be bound externally to a document.

The advantage of this is that the binding can occur with no impact on the structure and content of the document.

ITS does not specify such external bind mechanisms beyond specifying that any rules applied in this manner have lower priority that those bound via an internal selector-based annotation or a references selector-based annotation. In (Ó hAirt et al 2012) we present an approach to externally binding ITS meta-data to a document in a content management systems, using the folder meta-data and multi-filing capabilities of the Content Management Information

Service API standardized by OASIS (Choy et al 2010).

4.2 Indirect Annotation of Structured Content

These patterns address situations where the value of annotation is not included in the attributes annotating the text, but instead the value is contained is some other meta-data that is referenced.

P6 Referenced Annotation

Here the annotation is not held in an attribute value, but instead the attribute specifies an Internationalized Resource Identified (IRI) that can be dereferenced (typically retrieved with a HTTP GET) to yield the meta-data value.

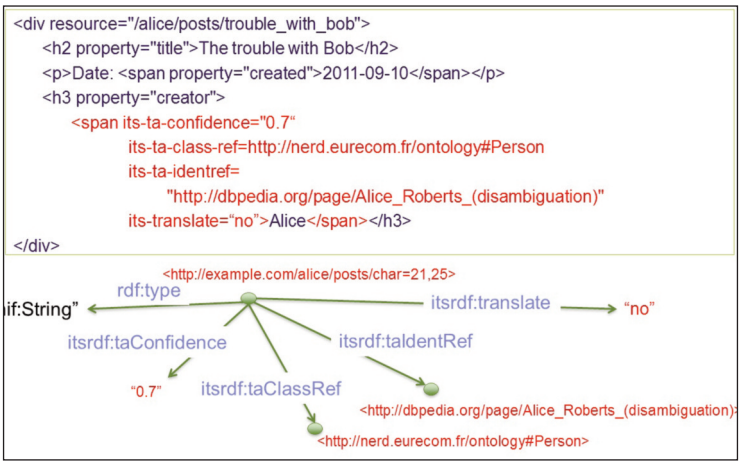

Figure 5: Example of conversion of ITS annotated content to RDF using the ITS and NIF Ontologies

The constraint is that the annotation parser must be able to access and dereference the IRI.

The advantage of this pattern is that the IRI can point to structured data so that annotation of a more complex type than is permitted in attribute node values can be used. The value of the annotation could in fact be any media or media fragment type, from a fragment in a DOM-conformant document, to an RDF node or even rich media content such as an audio or video resource. This pattern also allows for many annotated nodes to easily reference the same meta-data and it allows for that meta-data to change independently of changes to the annotated document.

Several ITS data categories contain a reference pattern data attribute, typically using the suffix 'Ref'. ITS does not specify the type of the referenced meta-data which in some cases necessitates an additional data attribute being defined to explicitly refer to a schema or classification resource.

P7 Pointer Pattern

Meta-data that can be used to annotate text nodes may sometimes already exist in the document, but as an ad hoc text node or attribute node value, which is therefore difficult to parse in an interoperable way. This pattern makes explicit that another part of the document can be used to annotate textual content.

The constraint in applying this pattern is that it is appropriate to use only with the selection based annotation, i.e. it should operate as a schema level mapping, matching all selected instances of textual content to existing accompanying meta-data within a defined schema.

The advantage of this pattern is that it allows existing piece meta-data to be reused to provide interoperable textual annotation with a minimal impact on the document, thereby minimizing the necessary addition mark-up needed to achieve a new interoperable annotation

Several ITS data categories make use of this pattern, using a data attribute with a 'Pointer' suffix, the value of which must be a relative XPath selector.

P8 Multi-Annotated Text

The lack of semantic ordering for attributes in a DOM conformant document means that only one attribute node of a given name may be associated with a given element node. However in some circumstances an annotation of a given type may need to be applied several times to some text in a document. This may be because we wish to record that different values for an annotation where applied

at different points in time, or that different annotating agents had different views on what the value of the annotation should be. Where multiple values need to be applied to the same text the following options can be adopted:

a) The attribute values can be specified in nested elements around the annotated text, e.g. in HTML using nested elements. This has the advantage of not requiring any specialized parsing. It has the disadvantage of adding a lot of otherwise unnecessary element mark-up to the document. This solution is not adopted explicitly in ITS.
b) The data attribute can itself have multiple values, e.g. separated by spaces. This has the advantage of being simple for single value attributes. However if the annotation requires the specification of more than one data attribute types, then a structuring convention is needed for the value, which requires its own parsing rules. These can become complex if the specification of values for all types is not mandatory. ITS adopts such a convention in the domainMapping attribute of the Domain data category. Here the multi-value is a tuple and an algorithm for parsing the values is defined. This approach also has the disadvantage that the number and size of value is limited by the maximum attribute value size.
c) Multiple annotation values may be captured as attributes of separate instances of the same element type that are collected in a special stand-off element placed elsewhere in the document and referenced by a reference pattern annotation of the text. The advantage of this pattern is that it allows straightforward DOM parsing of multiple annotations with no limit on value sizes, or the number and optionality of attribute types in a particular annotation. The disadvantage is that it introduces additional element into the annotated document. ITS2.0 implements this standoff solution for multi-annotation for the Provenance and Localization Quality Issue data categories.

P9 Annotation Meta-data

This pattern allows the annotation itself to be associated with additional meta-data. This is useful if the way in which the annotation was generated has a bearing on how it should be interpreted. It is performed by a direct sub-tree annotation whose values associate the instances of an annotation type in that sub-tree with additional meta-data.

This has the advantage of being able to annotate a large set of annotations with meta-data, without adding that meta-data to each individual annotation.

ITS 2.0 uses this pattern to associate a reference to the engine that has generated an annotation containing a confidence score with that annotation's data category. This is important since confidence scores are not comparable across engines, so identifying the engine involved is key to making use of the score. The annotation is done with the annotatorRef sub-tree annotation which can be applied to the Terminology, Text Analytics and MT Confidence data categories. This is efficient since typically all the annotation of a particular data category in a document will be performed by a single tool.

P10 External annotation of document fragments

A document may also be annotated by externally associating external meta-data with a fragment identifier in the document. The following approaches are possible:

a) An ID-based fragment IRI is used, e.g. http://ex.xml#sect2. This is constrained however to elements with an id (or in HMTL a name) attribute defined.
b) A selector-based fragment identifier is used, using xpath e.g. http://ex.xml#xpath (/html/body[1]/h2[1]/text()[1]). This has the advantage of being able to reference any text node even if no id attribute is present. It also able to reference attribute node values. It is constrained to XML documents however, as xpath fragments are currently not defined for HTML documents.

ITS does not use either of these external fragment reference approaches directly. Instead it specifies an indirect means of externally referencing specific annotated text. This is specified as part of a mapping of ITS annotation into RDF. This involves both parsing the ITS content of document and indexing this against a version of the document where all the markup and extraneous white space has been removed and just the text characters remain. The resulting RDF model contains a string resource which uses a char format IRI, e.g. http://ex.txt#char=21-25 to identify the text segment between character count 21 and 25 inclusive, see example in figure 4. This approach can only be used with a conversion algorithm that generates such a plain text document since char fragments are not defined for XML or HTML.

However, this approach does have the potential

advantage of being able to specify annotations for text that is not delimited by mark-up.

5. Requirements for Personalization Annotations

The previous section shows how the wide range of annotation approaches used in ITS2.0 can be generalized into a pattern language of reusable annotation patterns. As with any pattern language, patterns can be successfully applied in combination and this also is visible in the ITS2.0 specification. The benefit of this generalization is in the potential to more easily apply these annotation patterns in various combinations to the definition of new data categories.

We can therefore more easily design a new set of annotation semantics and then use a process of trial implementation prototyping and consensus forming amongst concerned actors to define new sets of content processing annotation which maintain many of the benefits resulting from the design of ITS.

As a start to developing possible interoperable content annotation data categories for personalization content processing we consider the following:

- 'personalize': which indicates to downstream processes where the annotated content should or should not be personalized (analogous to 'translate' in HTML5 and ITS).
- 'slice': indicate the boundaries of a slice, perhaps with references to slicing mechanism used and a confidence score on the positioning of boundaries.
- 'domain': indicates the subject domain or domains of the content for consumption by an adaptive process, which may have an optional confidence score. In ITS, this primarily identifies existing meta-data annotation (such as HTML meta annotations) as the domain identifiers that should be used by downstream personalization processes.
- 'text analytics': this annotates content based on the output of text analysis processes to identify content for later processing. Examples of such annotation include named entity recognition or text classification. ITS has an existing annotation that can identify entity and classifying resources as URIs, accompanied by a confidence score.
- 'axes-filter': indicates the types of adaptation modes that should or should not apply to the content, e.g. language, graphical, layout, navigation, modal, phrasing, précising. ITS has a similar data category that filters content from downstream processing based on existing BCP-47 locale codes, though here a personalization-specific coding of axes would be required.
- 'adaption-provenance': indicating what adaptation has been already applied to the content. Again there is an equivalent data category in ITS for specifying translation provenance, which can be useful in quality assurance workflows and in harvesting bi-text corpora from localization workflows using provenance parameters as a quality selection criteria. A similar role could be fulfilled for personalization, however a richer definition of agent types would be required, including: content slicer, domain annotator, text analytics annotator, indexer, filter, query rewriter, adaptive content rewriter, adaptive content composer etc. As these processes are either human driven, human checked or increasingly driven by machine learning techniques, knowing exactly which processing agents are involved in an instance of adaptation, is key in acting upon feedback received from users.
- 'adapt-script': a pointer to an executable adaptation script. This can be useful when some content is best bound directly to specific adaptation instruction that travel with the content, which may override more general processing driven by the values of other types of annotation.

These new data categories would therefore offer an abstract definition and a set of implementations, similar to ITS, enabling their implementation in HTML5, XML vocabularies and RDF data stores. However, while the evolution of ITS has been somewhat constrained by the well-established workflows already practiced in the localization industry, for personalization the pattern language presented interconnecting content and its annotating meta-data provides a well-tested starting point.

Acknowledgements

This research is supported by the European Commission as part of the MultilingualWeb-LT project (contract number 287815) and by the Science Foundation Ireland (Grant 12/CE/I2267) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Trinity College Dublin. The authors would like to thank all members of the MLT-LT WG for freely giving their knowledge and guidance in support of the ideas presented in this paper.

References

Berjon, R. (2013) *HTML5, A Vocabulary and Associated APIs for HTML and XHTML*, W3C Candidate Recommendation, available: http://www.w3.org/TR/html5/.

Buitelaar, P., Cimiano, P., Haase, P., Sintek, M. (2009) 'Towards Linguistically Grounded Ontologies', in Aroyo, L., Traverso, P., Ciravegna,

F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M. and Simperl, E., eds., *The Semantic Web: Research and Applications*, Lecture Notes in Computer Science, Springer Berlin Heidelberg, 111–125.

Chiarcos, C. (2008) 'An ontology of linguistic annotations', *LDV Forum*, 23(1), 1–16.

Clark, J. (1999) editor, *XSL Transformations (XSLT) Version 1.0*, W3C Recommendation, available: http://www.w3.org/TR/xslt. [accessed 29 October 2013]

Clark, J, DeRose, S., (1999) *XML Path Language (XPath), Version 1.0*, W3C Recommendation, available: http://www.w3.org/TR/xpath[accessed 29 October 2013]

Choy, D., Brown, A., Gur-Esh, E., McVeigh, R., Muller, F. (2010) *Content Management Interoperability Services (CMIS) Version 1.0*, OASIS Standard.

Emery, V., Kadie, K., Laplante, M. (2011) *Multilingual Marketing Content: Growing International Business with Global Content Value Chains*, Content Globalization Practice Research Report, Outsell.

Eberlein, K.J. et al (2010) *Darwin Information Typing Architecture (DITA) Version 1.2*, OASIS Standard

Filip, D., McCance, S., Lewis, D., Lieske, C., Lommel, A., Kosek, J., Sasaki, F., Savourel, Y., (2013) *Internationalization Tag Set (ITS) Version 2.0*, W3C Proposed Recommendation 24 September 2013, available: http://www.w3.org/TR/its20/. [accessed 29 October 2013]

Hellman, S., Lehmann, J., Auer, S., Brümmer , M., (2013) Integrating NLP using Linked Data, in *proceedings of the 12th International Semantic Web Conference*, 21-25 October 2013, Sydney, Australia

Herman, I., et al (2013) *RDFa 1.1 Primer - Second Edition, Rich Structured Data Markup for Web Documents*, W3C Working Group

Koidl, K., Conlan, O., Wei, L., Saxton, A.M. (2011) 'Non-invasive Browser Based User Modeling Towards Semantically Enhanced Personlization of the Open Web', in *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*, IEEE, 35–40.

Le Hors, A., et al (2004) *Document Object Model (DOM) Level 3 Core Specification, Version 1.0*, W3C Recommendation [accessed 29 October 2013]

Levacher, K., Hynes, É., Lawless, S., O'Connor, A., Wade, V. (2009) 'A framework for content preparation to support open-corpus adaptive hypermedia', in *International Workshop on Dynamic and Adaptive Hypertext: Generic Frameworks, Approaches and Techniques*, Citeseer, 1–11.

Lieske, C., Sasaki, F., 2007, *Internationalization Tag Set (ITS) Version 1.0*, W3C Recommendation, available http://www.w3.org/TR/its/.

Manola, F., Miller, E. (2004) *RDF Primer*, W3C Recommendation

Ó hAirt, A., Jones, D., Finn, L., Lewis, D (2012) '*An Open Localisation Interface to CMS using OASIS Content Management Interoperability Services'* at 17th LRC Internationalisation and Localisation Conference, Limerick, Ireland, available http://www.localisation.ie

Savourel, Y., Reid, J., Jewtushenko, T., Raya, R.M., (2008), *XLIFF Version 1.2*, OASIS Standard, available http://docs.oasis-open.org/xliff/v1.2/os/xliff-core.html [accessed 29 October 2013]

Wade, V. (2009) 'Challenges for the multi-dimensional personalised Web', in *User Modeling, Adaptation, and Personalization*, Springer, 3–3.

Windhouwer, M., Wright, S.E. (2012) 'Linking to linguistic data categories in ISOcat', in *Linked Data in Linguistics*, Springer, 99–107.

Notes

[1] Http://schema.org/docs/gs.html

[2] Http://www.docbook.org

ITS2.0 and Computer Assisted Translation Tools

Pablo Porto, Dave Lewis, Leroy Finn, Christian Saam, John Moran, Anuar Serikov, Alex O'Connor
Centre for Next Generation Localisation
Trinity College Dublin, Ireland
portovep@tcd.ie, dave.lewis@scss.tcd.ie, finnle@tcd.ie, Christian.Saam@scss.tcd.ie, moranj3@cs.tcd.ie, serikova@tcd.ie, alex.oconnor@scss.tcd.ie

Abstract

Version 2.0 of the Internationalization Tag Set (ITS) introduces a set of data categories designed to ease the interaction between content management, language technologies and the localization workflow. Many of these data categories capture meta-data related to manual language processing tasks including terminology management, source and target quality assessment and post-editing of machine translation. This paper examines the specific role that ITS2.0 may play in the design of Computer Assisted Translation (CAT) tools. It outlines the requirements ITS2.0 places on CAT tools design. We then examine the implementation of these requirements in a custom ITS/XLIFF-based CAT tool implemented as a client-based web application using JavaScript and compare this to an attempt to implement similar features in a open source Java-based CAT tool.

Keywords: *Internationalization Tag Set, Localization, Computer Assisted Translation, ITS 2.0, XLIFF, CAT, Java, Internationalization, Localisation, JavaScript, Requirements, Design, Client-based web application*

1. Introduction

Computer Assisted Translation (CAT) tools are a key part of the tool chain that supports the localization workflow. As efficiencies are sought in the localization process, the design of CAT tools is becoming an increasingly important area for investigation, especially in relation to how it supports integration with automated language technology components such as machine translation, terminology extraction and quality assessment. The MultilingualWeb-Language Technology (MLW-LT) working group at the W3C has recently finished specifying the successor to the Internationalization Tag Set (ITS) 1.0 (Lieske & Sasaki 2007) in the form of the ITS2.0 specification (Filip et al 2013). This extends the scope of ITS beyond the internationalization concerns of version 1.0 and addresses the exchange of content meta-data across the localization workflow, encompassing the CAT tool. It specifically addresses meta-data intended to assist in the integration of language technologies into the localization workflow, and therefore includes meta-data that should be displayed, created and manipulated via CAT tools. This paper examines the use cases in which CAT tool users may interact with different ITS2.0 data categories. We then review two exploratory implementations of these use cases. First, in detail, we examine the implementation of a green-field CAT tool implementation as a web client application using JavaScript. Then we briefly review an attempt to introduce ITS features into an existing open source CAT tool implementation, OmegaT[1].

2. CAT tool Use Cases for ITS2.0

This section outlines requirements for supporting data categories from ITS2.0 that are relevant to the operation of CAT tools. We assume ITS2.0 meta-data both consumed and generated by a CAT tool will be accessed and stored as part of an XLIFF 1.2 file (Savourel et al 2008). Therefore these requirements aim to align with the work on ITS-XLIFF mapping[2] being undertaken by the ITS Interest Group.

The ITS standard associates meta-data, in a standard format, with both source and target text. A number of text annotation meta-data types, called data categories, are defined for ITS2.0, including support for ones defined in ITS1.0. Figure 1 summarises the set of data categories, with the ones relevant to CAT tool design underlined. With reference to XLIFF concepts, where content is presented to translators as individual pairings of source and target segments, ITS mark-up may be presented to a CAT tool user in association with the following:

- source segments or sub-segments
- suggested target segments or sub-segments (e.g. taken from the XLIFF 'alt-trans' elements)
- target segments or sub-segments based on

suggested target selection and post-editing by the tool user
- target segments or sub-segments provided in the XLIFF file and being reviewed or revised by the tool user.

response to a corresponding sub-segment protected section in the source. Note this behavior is not specified for the translate data category in the standard but seems a useful feature.

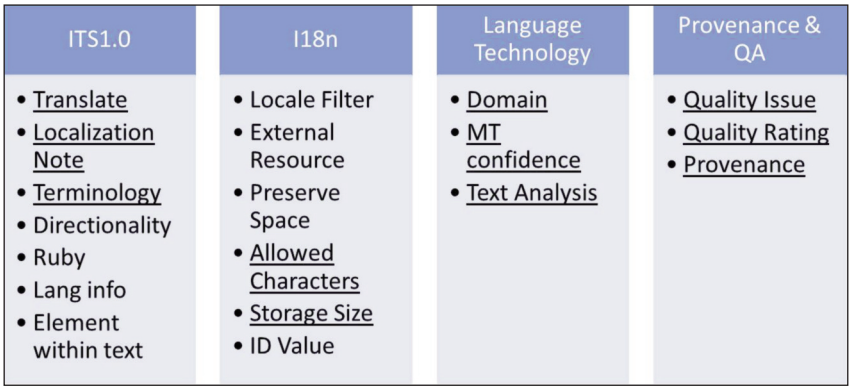

Figure 1: ITS Data Categories with CAT tool relevant ones highlighted

Use cases are given below on individual ITS data categories consistent with the ITS notion that systems can conform to each data category independently of support for the other. However, any support for multiple data categories in CAT tools must also support their concurrent usage. So it is important that all visual indications at the segment or sub-segment level should be visually distinct from each other. Care is needed however to ensure that CAT tools users do not suffer cognitive overload due to any proliferation of displayed meta-data associated with the text being translated. The following use cases are listed against the relevant ITS2.0 Data Category. Reference is made to the relevant ITS markup indicated by prefix "its:" and the XLIFF mark-up by the prefix "xlf".

Translate

This data category indicates whether the annotated text should be translated or not:

- **TraUC1:** View segments marked not to be translated (using its:translate="no") as context to the CAT user
- **TraUC2:** View highlighted source sub-segment that are marked to be not translated (using xlf:mrk mtype= "protected" as the equivalent of as its:translate="no" per the MLW-LT ITS-XLIFF mapping) to guide segment translation.
- **TraUC3:** View a highlighted sub-segment of a suggested target translation marked to indicate where an MT engine has specifically not translated text in the xlf:alt-trans/target in

Localization Note

This provides a way to convey a note from content authors or other down-stream internationalization workers to workers in the localization workflow.

- **LocUC1:** View the note text and note type (description or alert) associated with source segments or sub-segments marked.

Terminology

This indicates if the annotated text constitutes a term and references associate meta-data)

- **TrmUC1:** View source sub-segments annotated as terms (with its:term="yes") together with, where present,: a clickable link to further information on the terms (from its:termInfoRef); the confidence score associate with this term identification (from its:termConfidence) and a clickable reference to more information on the tool that generated this meta-data (from its:annotatorsRef).
- **TrmUC2:** Create source sub-segment term annotations, together with: an optional reference to further information (populating its:termInfoRef) and an optional manually determined score of the users confidence in the term annotation (populating its:termConfidence). If the latter is added, then the corresponding its:annotatorsRef attribute must be added by the CAT tool. This should reference the CAT tool itself, but could also usefully provide information about the user. If the schema of the referenced terminology

information resource is known, the values could be pre-fetched and displayed instead of presenting the references.

- **TrmUC3:** Existing source sub-segment term annotations may be deleted by the tool user.
- **TrmUC4:** Edit existing source sub-segments term annotations. This may involve changing or adding the value of reference additional information (modifying its:termInfoRef). It may also involve the changing the its:term value ("yes" or "no") or changing the its:termConfidence according to the terminology procedure being followed by the user in checking and correcting. If present the its:annotatorsRef value cannot be changed by the tool user.
- **TrmUC5:** View term annotation of a target language sub-segment. This indicates that an automated translation component has either attempted to preserve terminology annotation of the corresponding source segment or has added the annotation based on internal terminology information. This can be useful in assuring target terminology quality and consistency.

Domain

This indicates the application domain addressed by the annotated text.

- **DomUC1:** View domain annotations associated with the entire source document, a specific subsection of it, segments or sub-segment (as represented by annotations of xlf:file, xlf:trans-unit, xlf:source-segment or xlf:mrk respectively by its:domains). Differences in domain annotation that is different from surrounding text should be differentially highlighted. Note that the value of the its:domains attribute can be multivalued.
- **DomUC2:** View an automatically generated translation (i.e. the xlf:target in an xlf:alt-trans) that has to be annotated with its:domains to indicate the actual domain values used in the translation. This may be important if the value used by the MT engine differs from those specified in the corresponding source segment or sub-segment.

Text Analysis

This annotates text with reference to lexical or semantic information.

- **TxaUC1:** View source sub-segments annotated with text analysis. Where present, the clickable values of the class of entity the text represents (from taClassRef) and the specific instance it represents (from its:taSource and its:taIdent or its:taIdentRef). If the schema of the reference resources is known, the values could be pre-fetched and displayed instead of presenting the references. Also, if an score of the confidence in the annotation is present (from its:taconfidence) this should be presented together with a clickable reference to more information on the tool that generated this meta-data (from its:annotatorsRef).
- **TxaUC2:** Create source sub-segment text analytics annotations. In such cases, if an its:taConfidence score is added then the corresponding its:annotatorsRef should identify the CAT tool and possibly also the user.
- **TxaUC3:** Delete existing source sub-segment text analysis annotations.
- **TxaUC4:** Edit existing source sub-segments text analysis annotations. This may involve changing or adding the value of its:taClassRef, its:taSource and its:taIdent or its:taIdentRef attributes. It may also involve changing the value of the its:taConfidence attribute according to the text analysis processing procedure being followed by the user in checking and correcting. If present, the value of the its:annotatorsRef attribute cannot be changed by the tool user .
- **TxaUC5:** View target language sub-segments that have been annotated with text analysis annotation. This can be used to indicate that an automated translation has either attempted to preserve source text analysis annotation of the corresponding source segment or adds the annotation based on text analysis functionality integrated with translation workflows. This can be useful in supporting target terminology consistency.

MT Confidence

This provides a confidence score resulting from an automated translation of the annotated text)

- **MtcUC1:** View the confidence score of a machine translation represented at a suggested target segment level by an xlf:alt-trans/target (from its:mtconfidence) and a clickable reference to the MT engine that produced the annotation (from its:annotatorsRef).
- **MtcUC2:** If a post-editor selects an xlf:alt-trans/target element as the translation of the corresponding source segment such that it is replicated in the xlf:trans-unit/target element and if that translation remains unaltered (i.e. it is not post-edited) then that element should be

annotated with its:mtConfidence and its:annotatorsRef attributes from the corresponding xlf:alt-trans/target element.

- **MtcUC3:** In situations where sub-segments have a differential MT confidence (whether in an xlf:alt-trans and the xlf:target element) this need to be visually indicated to the tool user. If the differential sub-segment confidence score is the result of translation by different engines, then the corresponding different engine annotation (its:annotatorsRef attributes) should be used. Note, that sub-segment confidence score are not currently supported in the ITS-XLIFF mapping []

Provenance

This records the people, tools and/or organizations involved in translating or revising the translation of the annotated text.

- **PrvUC1:** View translation provenance annotation applied to target segment and suggested target segments, displaying the value of the tool, organization and person involved if present (from its:tool, its:org and its:person attributes) or presenting clickable links for the same (from its:toolRef, its:orgRef and its:personRef). in a way that the user can opt to retrieve the referenced information. Similarly, view translation revision annotation associated with target segments that have undergone post-diting. In both cases multiple records may apply, so the display of attributes must indicate their grouping into individual records.
- **PrvUC2:** View a clickable reference to further provenance information (from its:provRef) if present. Where the tool to be able to determine the type of information being referenced, view it directly in an appropriate format, e.g. W3C provenance format or iOmegaT transLog post-editing logs.
- **PrvUC3:** For each translation or post-editing session, populate translation or translation revision provenance information for the segments being addressed in the session and optionally provide a UUID value for the its:provRef attribute. If the tool, organization and person values are identical to an existing record, then the same record reference should be used, but a new UUID should be appended to the value of the its:provRef attribute.

Localization Quality Issue

This records a encoding of a quality assessment applied to either source or target text.

- **LqiUC1:** View localization quality records annotating any source or target segments and any source or target sub-segments. Each record may include a type string, some comment text, a severity value (between 0-100), a profile reference that can be clicked to display details of the localization quality reference schema used and an flag indicating whether the issue is currently in active or not. These are taken,respectively, from:
  its:locQualityIssueType,
  its: locQualityIssueComment,
  its: locQualityIssueSeverity,
  its: locQualityIssueProfileRef
  its: locQualityIssueEnabled.
- **LqiUC2:** Add new localization quality issue annotations to either source or target segments or to source or target sub-segments.
- **LqiUC3:** Edit existing localization quality issue annotations to correct errors they made in previous annotations. Changes to annotations provided by previous users should be restricted according to localization quality checking procedures, including changing the status of the issue enabled flag.
- **LqiUC4:** Delete an existing localization quality issue annotation to correct erroneous annotation they made previously. Deletion of annotations provided by previous users should be restricted according to localization quality checking procedures.

Localization Quality Rating

This allows annotation of an overall quality rating for a target document or section or of a quality vote for a particular document, section, segment or sub-segment (including suggested segments and sub-segments).

- **LqrUC2:** View the meta-data associated with the annotation, namely
  its:locQualityRatingScore,
  its:locQualityRatingScoreThreshold,
  its:locQualityRatingVote,
  its:locQualityRatingVoteThreshold
  its:locQualityRatingProfileRef.
- **LqrUC2:** Annotate the whole document, a translation unit, a segment or a sub-segment with a localization quality rating as a score or as a vote. For specifying a vote, some external mechanism is required for tallying the vote. The option should be offered for the user to enter a threshold value for the rating or vote and a reference URL to the assessment framework

used.
- **LqrUC3:** Delete an existing annotation.

Storage Size
This specifies the maximum storage size allocated to the annotated content.

- **StsUC1:** View storage size information for a target segment or sub-segment (from its:storageSize, its:storageEncoding and its:lineBreakType).
- **StsUC2:** Annotate a target segment or sub-segment with storage size information, indicating size restrictions on the textual content (populating its:storageSize, its:storageEncoding and its:lineBreakType).
- **StsUC2:** View report on breaches of the storage size restriction of the annotated textual.
- **StsUC3:** View proportion of the allowable storage size restriction available on the annotated textual content as it is being edited and be altered when the maximum is reached.

Allowed Characters
Specifies the characters that are permitted in a given piece of content.

- **AlcUC1:** Annotate a target segment or sub-segment with an its:allowedCharacters attribute to indicate which characters are permitted in the textual content.
- **AlcUC2:** Be alerted where target or suggested target text (from xlf:trans-unit/target and xlf:alt-trans/target) conflicts its:allowedCharacters value, indicating which characters in the text are in conflict.

3. Implementing ITS/XLIFF based CAT tool as a Web Client Application

In this section we describe an initial proof of concept implementation of these requirements that was implemented as a Web Client application using Java Script such that stand alone CAT tool functionality could be offered in a web browser. This was in part an assessment of the level to which a CAT tool based on ITS and XLIFF standard could be built using the Open Web Platform[3]. To put this in context, Table 1 summarizes the level to which the features of ITS2.0 and XLIFF integration are supported by equivalent features offered by other existing CAT tools.

The application, named Escriba, needed to retrieve and store both the localization content and the ITS meta-data associated with it, so an ITS parser is required. An existing parser implemented in jQuery was considered. This was called, jQuery ITS2.0 Parser[4] and is developed and maintained by Cocomore, one of the active members in the MLW-LT WG. However this library worked with XHTML, so for this CAT tool implementation, its use would require a conversion from XLIFF to XHTML and back again. Initial feasibility testing show this to be less efficient that developing a single ITS2.0+XLIFF parser, so this latter option was adopted.

	Wordfast Anywhere	Google Translator Toolkit	Pootle	XTM Cloud	PO Editor	Microsoft Translator Hub
XLIFF Support	N	N	Y	Y	N	N
ITS Integration	N	N	N	Y v1.0	N	N
Web MT	Y	Y	Y	Y	Y	Y
Glossary/ Termbase	Y	Y	Y	Y	Y	Y
Integrated Spell Checker	N	Y	N	Y	N	un known
Project Management features	N	Y	Y	Y	Y	Y
Project statistics	Y	Y	Y	Y	Y	Y

Table 1: Feature comparison of existing web client CAT tools

Implementation Components
The overall design consisted of the following modules and constituent components:

ITS2 Module:
This encapsulates components which provide support to view, edit and delete ITS 2.0 meta-data. To date implementation supports the Translation, Localization Note, MT Confidence, Provenance and Localization Quality Issue data categories. As per the ITS Interest Group's ITS-XLIFF mapping only the local style of ITS annotation is supported, i.e. global selector style was not supported. This module is formed of the following components:

- its-metadata-editor: Allows insertion, edition and deletion of ITS 2.0 meta-data in a given XLIFF file.

- its-metadata-visualizator: Contains all the logic which specifies how the information extracted from the ITS 2.0 metadata should be displayed.
- its-metadata-extractor: Provides the required functionality to extract ITS 2.0 from a given XLIFF file

XLIFF module:
This module contains all the components that provide support for handling XLIFF files. It is formed of the following components:

- xliff-data-manipulator: Allows for insertion, editing and deletion of the XLIFF elements of a given XLIFF file.
- xliff-data-selector: Provides support for selecting specific XLIFF elements (e.g. target elements) of a given XLIFF file.

Core module
This module contains the core functionality of the system. It is formed of the following components:

- content-navigation: Controls how the content of a project file should be displayed and in what order. It contains almost all the User Interface (UI) functionality.
- core: Provides support for down-loading and uploading XLIFF files and the functionality for set up a new project.

It also contains the remaining UI functionality which is not defined in the content-navigation module.

- user-pref-controller: Allows the configuration of preferences for a specific user. It stores and retrieves user preference information in a user model.
- keyboard-shortcuts: Controls the keyboard shortcuts supported by the system and how to enable or disable it.

User Interface Implementation
The implementation of the user interface (UI) is one of the key aspects of the project. It is necessary to select the right set of technologies which allow the implementation of a clear and appealing interface to engage users. It is also designed to enable Adaptivity in the UI so that various features can be presented in different ways for different types of users. There are several UI front-end frameworks which ease the design and implementation of a richer user experience (UX). Such frameworks collect best practices and UI and UX conventions and bundle for used by developers who are less expert in the UI/UX area. The one selected for this project was Bootstrap[5], which was released by Twitter as an open source UI front-end to provide a simple and quick way of creating clean and highly usable applications.

We have used Bootstrap version 2.3.2 to quickly develop the basic functionality of the UI of the

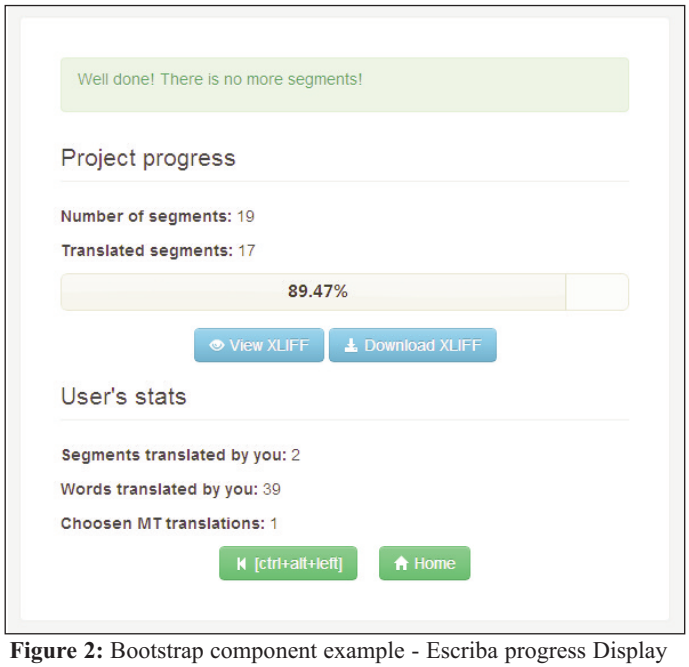

Figure 2: Bootstrap component example - Escriba progress Display

system. Moreover, one of the benefits of implementing the presentation layer using Bootstrap is that this framework integrates a set of responsive features that will allow the prototype to be adapted to mobile phones very quickly in the future. Figure 2 shows an example of how different elements of the Bootstrap toolkit were used to provide an appealing and simple UI for progress tracking. Bootstrap aids this through high level provision of different types of buttons, icons and typography. However UX is about more than a clean and usable user interface. The UI should react quickly to the user interaction and provide visual clues that this interaction is happing like component animations, transitions, effects, etc. The Escriba implementation achieves some of these effects using jQuery[6]. This is an open source library developed by the UX community to ease the implementation of highly interactive web applications. Figure 3 shows an example of how the jQuery UI accordion widget was used to present the user with different alternative translations for a source segment using an animated expanding widget. The combination of Bootstrap and jQuery UI have allowed us to quickly achieve an acceptable UI and UX design for the Escriba prototype match the needs of CAT tool users interacting with ITS meta-data.

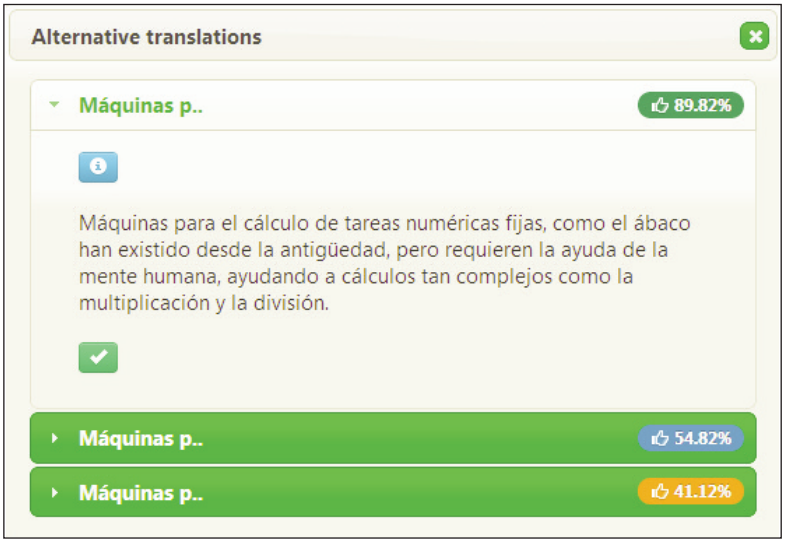

Figure 3: jQuery accordion example for revealing alternative sugested translations

XLIFF Manipulation

As detailed earlier a dedicated XLIFF parser was developed to efficiently support import and export into the UI components. There are several query languages that can be used for selecting nodes of an XML document using JavaScript. We have developed a custom parser that uses CSS selectors to retrieve specific nodes of the XLIFF document. This was selected over XPATH due to the apparent simplicity and familiarity of using these CSS selectors for styling purposes in HTML pages. The CSS selectors worked fine for all usage scenarios except when selecting an element based on an attribute whose name contained a colon, though a work around was achieved.

Main UI Features

We now describe the main feature of the Escriba UI. A demonstration version of Escriba is available on the web[7].

Home page

The main page of the web application allows the user to create a new project by uploading a XLIFF file. As can be seen in Figure 4, this page is also used to provide access to the different tasks that make up the provided CAT tool functionality as well as a list of XLIFF+ITS2.0 input samples to see the implemented ITS 2.0 support in action.

Translation panel

The translation panel (see figure 5) is the main component of the application. It allows the user to translate segments, edit translations and see alternative translations for a given segment. Moreover, it allows the user to access to more advance features related to ITS meta-data creation, edition and deletion process. The segments are presented in a vertical list. Segment number, source text and target text are shown for each of the segments. The user can select a segment by clicking

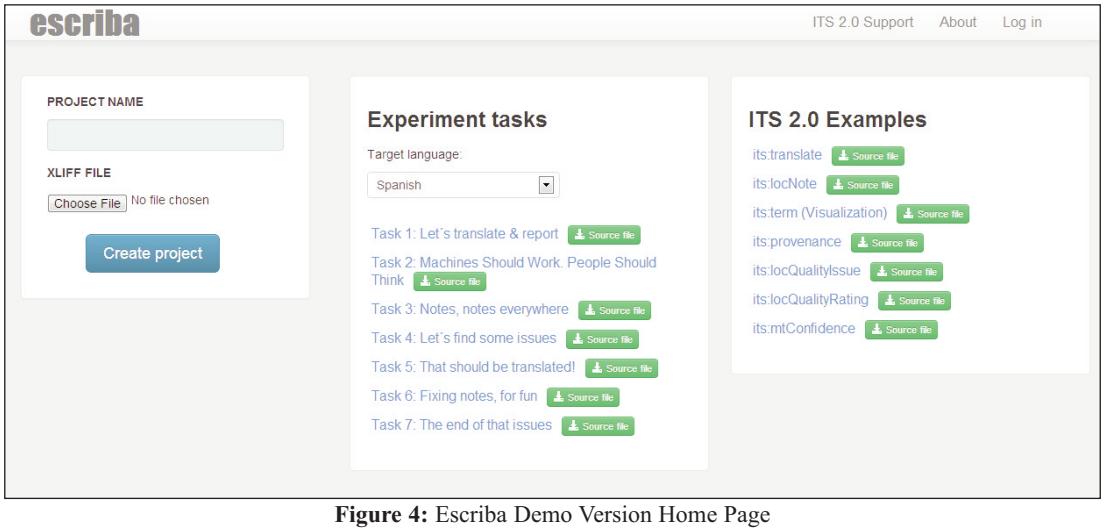

Figure 4: Escriba Demo Version Home Page

on it and see which segment is selected at any time. The ITS 2.0 information associated with a given segment is shown above the source text or target depending on the one with which it is associated. The user can create new ITS 2.0 annotations for Localization Quality Issues and Localization Quality Rating through the buttons situated below the source and target text. Finally, two buttons can be found at the bottom of the page which allow forward and backward navigation through the segment list.

Progress panel

The progress panel (see Figure 2) can be accessed from the top of the translation panel. It shows project related information such the project name as well as the current progress and the user performance in the current project. The progress is represented by a percentage and the user can also see the total number of segments that the project includes and the number of already translated segments. With regards to the user statistics, the user can see how many segments were translated by him, the number of words translated and finally the number of alternative translations selected.

As can be seen in Figure 2, the panel also allows the

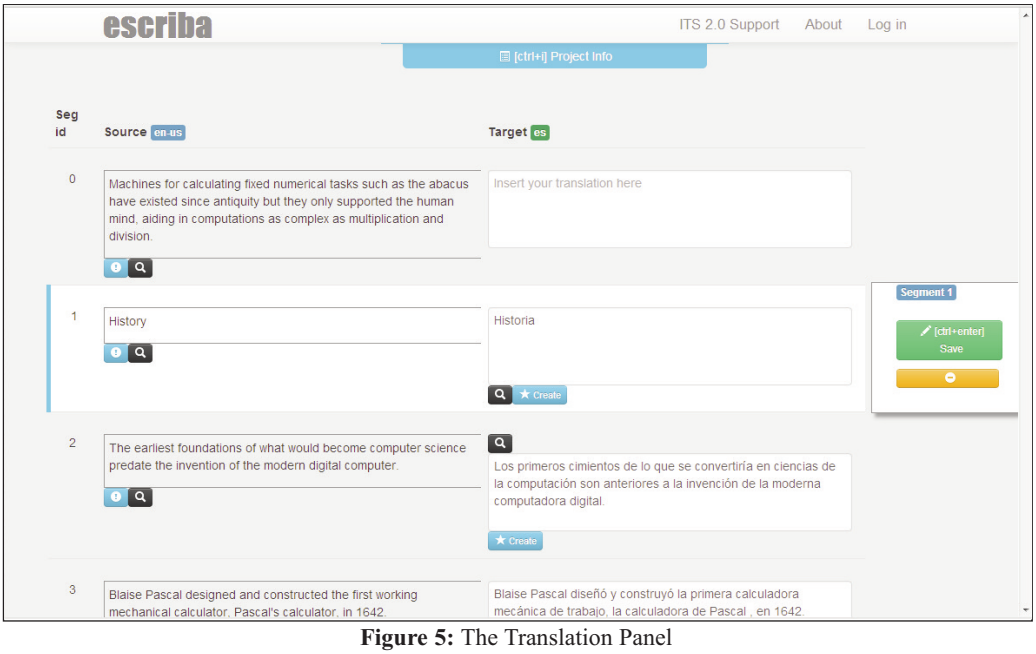

Figure 5: The Translation Panel

user to download or view the current state of the XLIFF file associated with the project.

4. Alternative Implementation

An internal document data structure, that reflects the structure of the source/target document well and in a manner agnostic of specific document type, greatly eases some aspects of rendering and manipulating the contained meta-data. As we have seen in the implementation of Escriba, ITS lends itself well to W3C DOM data structures and in turn to simple implementation using widely available open source web application libraries in JavaScript.

As a comparison, we also investigated the handling of ITS enriched translation interchange documents in a Java-based open source CAT tool called OmegaT. The Swing GUI toolkit used in this application provides a rich graphical editor API. The backing data model of the component representing the main editor is based on a hierarchical (tag) structure inspired by HTML/XML and thus similar in spirit to the W3C DOM. In principle this allows one to easily represent any kind of document structure/type with or without (in-line) ITS meta-information. In the data model the structure and content are held separately. The same content may even be associated with several different structures representing different aspects. Although OmegaT's editor interface is based on the general swing editing components, the part of the editor's backing document model representation structure is basically circumvented in favor of a secondary data structure that represents the source document's content as well as those parts of its structure deemed necessary for achieving the purpose of translation. This secondary data structure is a list containing the equivalents of segments and is kept in sync with the editor's content model. The list is populated when the source document is read in. Document filters analyse input documents in terms of content-bearing (i.e. translatable) and non-content-bearing structural elements. Content is extracted from the content bearing elements and the general structure is copied into a backup document skeleton that is not accessible from the editor. Thus upon import a lot of the structural information is lost to further manipulation along with any meta information embedded at that level.

In undertaking this alternative implementation it became clear that the extent of changes needed for a full information round-trip from document to editor and back required a much greater the time frame and skill level compared to the green-field implementation of Escriba. For some aspects of ITS that can be expressed by features of XLIFF (such as MT confidence, non-translate segments and localization notes) and that survive the existing input filtering process, implementations of GUI representations and interaction patterns were attempted. For clarity of presentation, several GUI components including the main editing component were changed from a line oriented to a tabulated display. To visualise these unsupported features we prototyped a number of tabular layouts for ITS 2.0 segment-level information and terminology information in the GUI, though these additional meta-data were not supported by existing data import filter.

.

5. Conclusions

ITS2.0 defines a set of meta-data that can be associated with content as is passes through various stages of the localisation process. Many of these data categories therefore must be viewed and manipulated by translators using CAT tools. In this paper we explore the requirements that CAT tool developers need to satisfy in order to support such interactions. We also conduct some implementation trials. As may be expected, support for these new meta-data features was found to be much easier in a green field implementation than when trying to refactor an established CAT tool. However, the Escriba implementation experience also showed the potential for rapid development of flexible and engaging CAT UI function that operates in a web browser by leveraging modern Open Web Platform libraries such as Bootstrap and JQuery. The Escriba implementation was developed by a master's student in a few weeks, and was then easily extended to include adaptive UI features. The OmegaT implementation was also able to support efficient UI prototyping using Java Swing, but was restricted by the complexity of refactoring the import and export feature to accommodate new meta-data types. If the integration of new technology such as machine translation, text analytics and quality assessment, is to be integrated into the localization workflow using meta-data annotations as advocated by the W3C in ITS 2.0, then such filters must accommodate more flexible means for accommodating new meta-data.

Acknowledgements

This research is supported by the European Commission as part of the MultilingualWeb-LT

project (contract number 287815) and by the Science Foundation Ireland (Grant 12/CE/I2267) as part of the CNGL – the Centre for Global Intelligent Content (www.cngl.ie) at Trinity College Dublin and under Grant 12/TIDA/I2424. The authors would like to thank Sharon O'Brien of CNGL at DCU as well as the members of the MLT-LT WG for feedback provided on initial drafts of ITS-CAT tool requirements.

References

Filip, D., McCance, S., Lewis, D., Lieske, C., Lommel, A., Kosek, J., Sasaki, F., Savourel, Y. (2013) '*Internationalization Tag Set (ITS) Version 2.0*', W3C Recommendation, available http://www.w3.org/TR/its20/ [accessed 29 October 2013]

Lieske, C., Sasaki, F., 2007, *Internationalization Tag Set (ITS) Version 1.0*, W3C Recommendation available: http://www.w3.org/TR/its/ [accessed 29 October 2013]

Savourel, Y., Reid, J., Jewtushenko, T., Raya, R.M., (2008), *XLIFF Version 1.2.,* OASIS Standard available: http://docs.oasis-open.org/xliff/v1.2/os/xliff-core.html [accessed 29 October 2013]

Notes

[1] http://www.omegat.org/
[2] http://www.w3.org/International/its/wiki/XLIFF_1.2_Mapping
[3] http://www.w3.org/wiki/Open_Web_Platform
[4] http://plugins.jquery.com/its-parser/
[5] http://getbootstrap.com/
[6] http://jquery.com/
[7] http://4.mobile-webcat.appspot.com/

Linport as a Standard for Interoperability Between Translation Systems

Alan K. Melby[1], Tyler A. Snow[2]
[1]Department of Linguistics & English Language
[2]Translation Research Group
Brigham Young University, Utah, United States
akmtrg@byu.edu, tylerasnow@byu.edu

Abstract

In the translation industry there are a plethora of emerging and evolving technology standards, and a great need for interoperability among them. Some interoperability standards include XLIFF, for translatable bi-texts; TIPP, which adds a package format to coordinate files including XLIFF for a single translation task; and finally Linport, which contains translation data at the project level. This paper introduces and analyzes the viability of Linport, a solution to translation tool incompatibility issues. Linport and XLIFF are complementary interoperability standards.

Keywords: *standards, interoperability, linport, XLIFF, bi-texts, translation data*

1. The Need for Linport

The modern world shows ever increasing opportunities for connection and communication with others. Thanks to modern technologies such as satellites, cell phones, and the Internet, instant global communication is possible, a phenomenon unimaginable even a few decades ago. This level of global interoperability, or the capacity to work with others to accomplish tasks quickly and easily, is a defining achievement of our era. The great advancement of interoperability in the age of globalization will continue to make life easier. However, increased interconnectivity presents problems of its own. Being able to talk to someone on the other side of the world does not guarantee *communication* can be achieved. This is why the translation industry exists. There is also a need for technological communication within this industry. This suggests that new standards should be developed in order to promote interoperability.

A classic example of interoperability breakdown comes from the shipping industry. For thousands of years countries and companies would ship products internationally, attempting to maximize trade profits by cutting costs and travel time. Attempts to streamline these processes revealed that cargo ship containers did not fit onto the trains and boats that carried them to their final destinations. Container types and sizes varied greatly. Consequently, trucking and train companies found it difficult to plan for the movement of incoming goods. The trade goods would have to be unpacked from the shipping container and then repacked into containers which would fit on a truck or train, wasting valuable time and money in the process. The industry needed a standard. Ultimately, the international community agreed upon a standard shipping container size. These standardized shipping containers could be easily moved from ship to truck without removing the contents as both the ship and the trucks were made to handle the exact dimensions of the pre-specified containers. These improved measures of interoperability enhanced the shipping industry's productivity on a global scale.

As mentioned earlier, this need for standards applies not only to the shipping industry but to many other fields as well, including translation technologies. Translation companies are constantly vying for their translation software to be used and recognized in the world of commercial translation. Some of these companies include SDL, LingoTek, MultiLing, Kilgray, SYSTRAN, and XTM-Intl, among many others. Each translation software system has a different interface that handles translation projects in different ways, often creating interoperability difficulties.

For example, if one company starts a translation project using SDL tools and then subcontracts out of house to a freelance translator who uses Kilgray, the same problem ensues as found in the shipping container example. The various project

components, such as source text, specifications, and file types, all have to be "unpacked" from the SDL format and "re-packed" or reformatted in the Kilgray-style format in order for the freelance translator to do their allotted portion of the project. Finally, the completed translation then has to return to SDL format. Similar to the shipping industry's need for standardized containers, the translation industry needs a standard "container" of its own, to allow for interoperability between the numerous translation software tools now available to professional translators and companies worldwide. Fortunately, a standardized translation project container will soon be available.

2. Linport: A Standardized Container for the Translation Industry.

Linport (Language Interoperability Portfolio), a complete and interoperable container solution for all translation processes and projects, is already under development. A Linport container documents the details of an entire translation project, and carries each of the individual components that comprise the various tasks relative to a translation project. These tasks could include the initial translation of a text from one language to another, a translation revision, a review, or a proofreading task. Each of these tasks would be accessible to any participant in the project who, upon task completion, would be able to "pack" their goods into the Linport container for further use. Therefore, keeping with the shipping example, a Linport portfolio represents an overall project view much like a shipping container but also would be able to define particular translation tasks within the project. These are comparable to standardized boxes that are shipped in the larger container and could be represented by any number of translation formats, including XLIFF and TIPP.

2.1 Elements of Linport

The current implementation of Linport is represented by a directory structure format containing two sub-folders. The first is the *portInfo folder*, similar to an HTML header element. It contains information about the portfolio as a whole, such as specifications and support files that apply to all relative subtasks, and universal identifiers to facilitate breakdown and reintegration of the portfolio.

The second element is the *payload folder*, which contains translated or un-translated documents for review, as well as the supporting resources needed for translation or any other required task, such as revision or review. Examples of these resources could include translation memory files, textual references, terminology files, and style guides, among others.

Linport can contain almost any file format, as long as it fits into the predefined directory structure. To facilitate this methodology, the payload folder is divided into language folders, such as the "en" folder for English or "es" folder for Spanish. These folders then contain document folders which house exactly one document in a "doc" folder and its supporting files, such as glossaries and translation memories, along with the document's specifications, in a support folder. In this manner, translation tools know how each file correlates to another and can handle them appropriately. The directory structure outline can be found at: http://dragoman.org/linport/ldm.txt.

2.2 Structured Translation Specifications (STS)

Structured Translation Specifications (STS) enhance Linport's ability to store and transfer the information necessary for translation tasks. It allows companies or translation project managers to specify important metadata about the translation itself, such as the target audience and intended use of the translation. An STS file includes 21 important translation specifications that should be considered during, or even before initiating a translation task or project. The 21 specifications are provided in Table 1 and are also available at http://ttt.org/specs. Table 1 was made by the Globalization and Localization Association (GALA) and the Localization/Translation and Authoring Consortium (LTAC).

3. History of Linport

Interestingly enough, Linport itself is an example of a conglomeration of companies and organizations working together. The project comes from three main project streams. In March 2011 many of the organizations that participated in the former LISA standards organization agreed that a container type format was needed in the translation industry. The Globalization and Localization Association (GALA) and the Localization/Translation and Authoring Consortium (LTAC) began work on what was named the Container Project The first presentation of their work was given a month later in Torino,

A. Linguistic [1–13]

Source-content information [1–5]

[1] textual characteristics
- a) source language
- b) text type
- c) audience
- d) purpose

[2] specialized language
- a) subject field
- b) terminology

[3] volume
[4] complexity
[5] origin

Target content information [6–13]

[6] target language information
- a) target language
- b) target terminology

[7] audience
[8] purpose
[9] content correspondence
[10] register
[11] file format
[12] style
- a) style guide
- b) style relevance

[13] layout

B. Production tasks [14–15]

[14] typical production tasks
- a) preparation
- b) initial translation
- c) in-process quality assurance

[15] additional tasks

C. Environment [16–18]

[16] technology
[17] reference materials
[18] workplace requirements

D. Relationships [19–21]

[19] permissions
- a) copyright
- b) recognition
- c) restrictions

[20] submissions
- a) qualifications
- b) deliverables
- c) delivery deadline

[21] expectations
- a) compensation
- b) communication

Table 1. Structured Translation Specifications

Italy at the JIAMCATT translation technology conference (JIAMCATT). After the presentation, a representative of the European Commission's Directorate General for Translation (DGT) indicated that their organization already had started on a similar project, known as the Multilingual Electronic Dossier (MED) project. In MED, they aimed to represent an entire translation project in what they called a translation "dossier." After a series of discussions, the Container project and the MED project were merged to form the Linport project in July 2011, hosted by LTAC Global a non-profit organization. It was decided that the Linport container would be called a portfolio and would contain all data pertaining to a translation project, be it an authoring, translation, or publication project.

3.1 Work by Interoperability Now!

In 2010, unbeknownst to the Linport project, an initiative company called Interoperability Now!, or IN!, had already started work on yet another similar project. The participants in the Linport project and those involved in the IN! project became aware of each other and then held a series of discussions. IN! agreed to integrate its "container" format into the Linport project.

3.2 IN!'s Translation Interoperability Protocol Package (TIPP)

IN!'s primary contribution to the Linport project is the Translation Interoperability Protocol Package (TIPP). TIPP represents a single translation task to be performed using exactly two languages in a translation workflow. A Linport portfolio by contrast contains the whole translation project, which could potentially involve many languages and tasks. By design then, a Linport portfolio could theoretically be broken down into multiple TIPP task packages which could be accessed, completed, and then reintegrated back into the portfolio for transportation to another translation tool. (History by Melby et al. 2012, *Multilingual Magazine*).

TIPP was designed with XLIFF in mind. Information about the TIPP format, including the parser tool, can be found online at: http://code.google.com/p/interoperability-now/.

4. How does Linport work and where does XLIFF fit in?

Linport portfolios are designed to efficiently move translation data between translation environment

tools. XLIFF users will find that a Linport portfolio incorporates XLIFF files at its very heart. A Linport portfolio can contain any number of translatable and/or previously translated documents. Although any bi-text, monolingual, or multilingual document can be contained within a Linport portfolio, it is anticipated that XLIFF will be the most common format used.

An XLIFF document will often be accompanied by several non-XLIFF supporting files, including: terminology files (e.g. .tbx), translation memory files (e.g. .tmx), translation metadata (e.g. an STS file), among countless others. All of these files can be optionally grouped together within a TIPP package or directly into the payload folder of a Linport portfolio. In this way, several XLIFF documents with their support files can easily be packaged together into one Linport portfolio, as shown in the diagram below.

-Linport provides project level interoperability for both XLIFF and non-XLIFF translation projects.

Whether you are working with a pure XLIFF-style project in house, or multiple freelance translators, Linport provides a standardized way to move project data between all stages of the translation workflow.

-Linport can handle multiple extension types.

Linport allows XLIFF to easily associate with non-XLIFF file types. Translation file formats for source texts range from XLIFF to DOCX to PDF. Glossary formats are equally diverse. Linport's standardized file structure will help machines and humans quickly associate all of the various parts of a translation project. Future Linport-aware XLIFF tools will be able to convert translation data to and

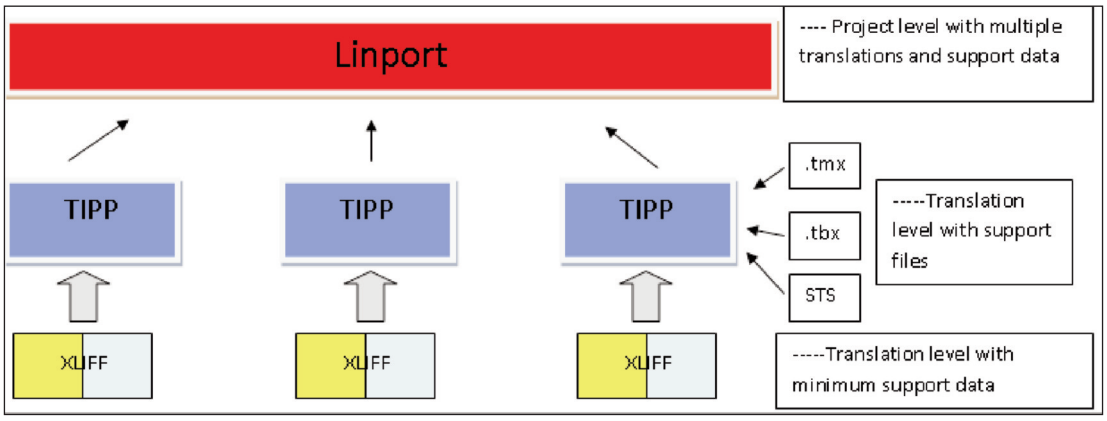

Figure 1: Linport porfolio organisation

5. Why should I use Linport? Isn't XLIFF enough?

Linport does not replace XLIFF. The two formats work together to promote and enhance organization and interoperability in the translation workflow.

Linport adds a new level of abstraction to the XLIFF format.

XLIFF is designed to organize the translation of source text to target text, whereas Linport is designed to organize multiple translations before and after the actual translation. Entire translation projects can be efficiently organized, broken down into translation tasks including XLIFF, and finally reintegrated back into a project portfolio for a reviewer or final publication.

from XLIFF format with ease.

-Linport incorporates ISO 11669.

ISO 11669 defines the Structured Translation Specifications (STS) a set of 21 translation parameters that enhance translation quality by providing additional information to the translators and reviewers of a specific translation project. More information can be found at http://ttt.org/specs

-Linport allows for translation quality assessment metrics such as QTLP.

Linport can easily incorporate any quality assessment metric, thus allowing enhanced interaction between translators and reviewers in order to produce higher quality translations. QTLP

is an emerging translation quality assessment metric format which is customizable for different projects or documents. QTLP or any other metric format can be contained in a Linport portfolio.

-Linport is an immediate solution that is easy to implement.

Although no builder tool currently exists to create or validate a Linport portfolio, though online Linport portfolio builders and validation systems are under development, all that is really needed to build one is an operating system with a directory structure. All existing Linport style portfolios to date have been made by hand in five minutes or less. As long as Linport guidelines are understood, any tech savvy or non tech-savvy person can create a viable Linport portfolio.

-Linport is free.

Linport is non-profit and non-proprietary. Any translation company or individual can use Linport royalty free.

6. Current and future development

Linport has much ongoing and future work to be done. The portfolio data model needs to be refined and formalized then submitted to a standards body, eventually becoming an ISO standard. Various software projects are being developed such as an online portfolio builder, a split website that breaks a portfolio into TIPPs, a merger website that integrates TIPP responses back into a portfolio, and a Linport validation system and schema, among others.

7. How to get involved

There are many ways to get involved in the ongoing development of Linport. You can join the Linport community and participate in monthly Linport conference calls by going to http://www.linport.org or joining the GALA Linport community group at http://www.gala-global.org (Search for Linport in their search box.)

Contact the authors at: tylerasnow@byu.edu or Alan Melby: akmtrg@byu.edu to contribute real project data that can be used to test the Linport portfolio model. All contributed data must be non-confidential. You can also contribute by testing apps developed for Linport, developing your own Linport applications, or introducing Linport into your company's translation workflow as an early adopter. The Linport community is open and thankful for any support you and your company are willing to provide.

References

International Annual Meeting on Computer-Assisted Translation and Terminology (JIAMCATT). (2012) Retrieved from http://jiamcatt.org/.

Linport: The Language Interoperability Portfolio Project. (2012) Retrieved from http://linport.org.

Melby, Alan. (2011) "The Seoul of Standards and You." *The ATA Chronicle.* Oct: 12-16.

Melby, Alan. Chandler, Brian. Lommel, Arle. (2012) "Linport addresses translation package compatibility." *Multilingual Magazine*. July/Aug: 45-47.

Melby, A., Lommel, A., Rasmussen, N., & Housley, J. (2012) "The Language Interoperability Portfolio (Linport) Project: Towards an Open, Nonproprietary Format for Packaging Translation Materials." (Unpublished article submitted to *The Journal of Internationalisation and Localisation).*

Structured Specifications and Translation Parameters. (2012) Retrieved from www.ttt.org/specs.

Glossary of Acronyms Used

DGT – **D**irectorate **G**eneral for **T**ranslation
›A part of the EC; JIAMCATT Partner
›ec.europa.eu/dgs/translation

EC – **E**uropean **C**ommission

ETSI – European Telecommunications Standards Institute
›www.etsi.org

IN! – **I**nteroperability **N**ow**!**
›A group working to improve the interoperability of tools and technology within the localization industry
›code.google.com/p/interoperability-now

ISO – **I**nternational **S**tandards **O**rganization
›JIAMCATT Partner

›www.iso.org

Linport – The **L**anguage **In**ter-operability **Port**folio Project
›www.linport.org

LISA - Localization **I**ndustry **S**tandards **A**ssociation
›*Ceased to exist March 2011*

OASIS - Organization for the **A**dvancement of **S**tructured **I**nformation **S**tandards
›www.oasis-open.org

QTLP or **QTLaunchPad** - Quality Translation Launch Pad.
EC-CORDIS-PF7-LT project 296347
cordis.europa.eu/projects/rcn/103949_en.html
(2012-07-01 to 2014-06-30)
›www.qt21.uk

TAUS – Translation Automation
›www.translationautomation.com

TIPP – Translation Interoperability Protocol Package – an **IN!** project

XLIFF – **X**ML **L**ocalisation **I**nterchange **F**ile **F**ormat
›XLIFF 1.2
›XLIFF 2.0
›XLIFF 1.2: docs.

ITS 2.0 Validation Techniques

Jirka Kosek
University of Economics, Prague
Prague, Czech Republic
jirka@kosek.cz

Abstract

ITS 2.0 (Internationalization Tag Set) is a new W3C Recommendation, which defines a set of universal elements and attributes that can be used in host vocabularies like HTML or XML to improve localization and translation processing. The fact that ITS markup can be combined with almost any other markup makes validation of ITS content more challenging than usual. This paper discusses various approaches to validation of ITS markup both in XML and HTML documents. Advantages and disadvantages of various approaches are discussed. Special attention is given also to validation of HTML5 content.

Keywords: *XML, HTML, XML schema, validation, NVDL, ITS*

1. Introduction

ITS 2.0 (Internationalization Tag Set) is a new W3C Recommendation which defines set of universal elements and attributes that can be used in a host vocabularies like HTML or XML to improve localization and translation processing (Filip, D., McCance, S., Lewis, D., Lieske, C., Lommel, A., Kosek, J., Sasaki, F., Savourel, Y.; Eds. 2013). The most common way to use ITS is to attach special attributes from the ITS namespace to elements containing content that can benefit from additional language related metadata.

```
<para>It would certainly be quite a <phrase its:translate="no">faux
pas</phrase> to start a dissertation in a pub…</para>
```

Example 1: Local ITS markup in an XML document expressed as an attribute

In example 1, you can see `its:translate` attribute in action. This attribute indicates that content of the `<phrase>` element should not be translated.

There are dozens of other attributes similar to `its:translate` available in ITS. Using this so called "local markup" is arguably the most common way of using ITS.

Another option is to define global rules. This is done by using dedicated rules elements. Example 2 shows a rule that forbids translation of labels in user interface in a DocBook document. Please note that rules are usually placed in some metadata wrapper element, such as `<info>` or `<head>`.

From examples 1 and 2, it is apparent that attributes for local ITS markup need to be allowed on almost

```
<article xmlns="http://docbook.org/ns/docbook"
         xmlns:db="http://docbook.org/ns/docbook"
         xmlns:its="http://www.w3.org/2005/11/its"
         its:version="2.0" version="5.0" xml:lang="en">
  <info>
    <title>An example article</title>
    <its:rules>
      <its:translateRule selector="//db:guilabel" translate="no"/>
    </its:rules>
  </info>
  <para>This is a short article. Title of article is shown in
        <guilabel>Title</guilabel> field.</para>
</article>
```

Example 2: Global ITS Rules

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=utf-8>
    <title>Terminology test: default</title>
  </head>
  <body>
    <p>We need a new <span its-term=yes>motherboard</span>
    </p>
  </body>
</html>
```

Example 3: Local ITS markup inside HTML document

any element, while special ITS elements are better to be allowed only inside of specific elements that already serve as metadata containers in the host format when ITS markup is integrated.

In HTML, the situation is similar. The only difference is that namespaces cannot be used. So instead of using a namespace prefix followed by a ":" (colon), such as `its:`, HTML has to use the hardcoded prefix `its-` as shown in example 3.
Let us now see various validation options for ITS content.

2. Schema Languages

In the XML world, validation is done using schema languages. A schema describes constraints on a document structure (elements and attributes you can use), datatypes (values allowed inside elements and attributes) and sometimes it can also express more advanced checks.

Over the time, several schema languages emerged. Currently, the two most common schema languages are W3C XML Schema (Fallside, D.C., Walmsley, P.;Eds., 2004) (Thompson, H.S., Beech, D., Maloney, M., Mendelsohn, N.; Eds., 2004) (Biron, P.V., Malhotra, A.; Eds., 2004) and RELAX NG (Clark, J., Murata, M.; Eds., 2001). Both of them are grammar based, which means that they can precisely list all possible element/attribute combinations in a very concise way. However, this approach has some limitations, especially when more complex relationships in documents need to be described. In such cases, the Schematron language (*Document Schema Definition Languages (DSDL) — Part 3: Rule-Based Validation — Schematron.* 2006) is very popular, as it can describe complex constraints over XML documents using XPath expressions.

There are also special schema languages that are useful in particular cases. One of them is NVDL (*Document Schema Definition Languages (DSDL) — Part 4: Namespace-Based Validation Dispatching Language — NVDL.* 2006), which can be very useful if you use several namespaces in your document and there is no single schema for such compound document available.

3. Validating ITS markup alone

In case you do not have any schema for a document and just want to validate ITS markup used inside it, you can use the NVDL schema available as a part of

```
<rules xmlns="http://purl.oclc.org/dsdl/nvdl/ns/structure/1.0">
  <namespace ns="http://www.w3.org/2005/11/its">
    <validate schema="its20-elements.rng"/>
  </namespace>
  <namespace ns="http://www.w3.org/2005/11/its" match="attributes">
    <validate schema="its20-attributes.rng"/>
  </namespace>
  <anyNamespace>
    <allow/>
  </anyNamespace>
</rules>
```

Example 4: NVDL schema for ITS

the ITS specification.

This schema finds all elements and attributes from the ITS namespace in a document and sends them separately for validation against the RELAX NG schema for ITS elements and attributes. Everything else that is not ITS markup is ignored during this validation.

validation does not detect misplaced elements with ITS markup, usually rules. For attributes, this is not such an issue, as ITS attributes are usually available on most elements of a host language.

4. Validating host vocabulary together with ITS markup

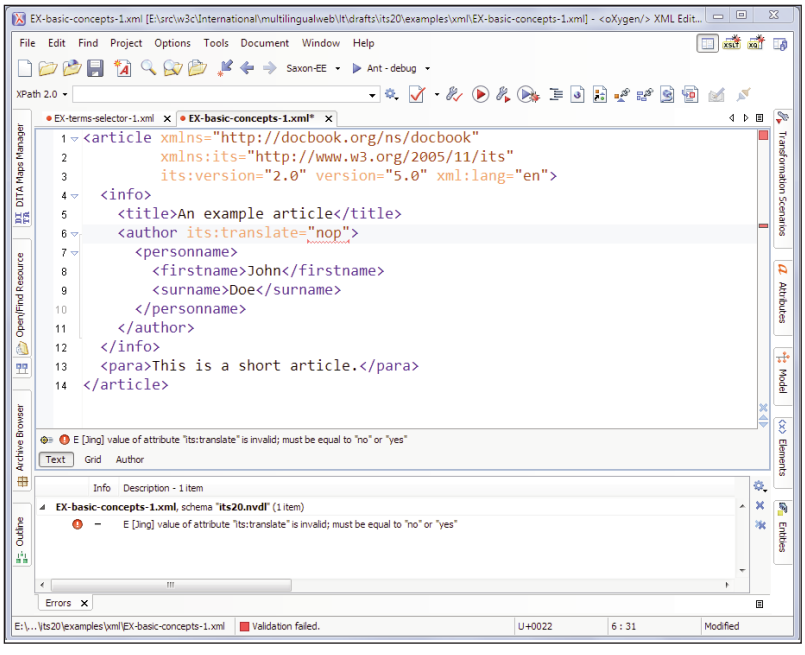

Figure 1: Validation of invalid ITS markup inside oXygen XML editor

The main advantage of this approach is that any file containing ITS markup can be validated without an additional effort. The main disadvantage is that such

If you need to have precise control over where in your existing vocabulary ITS markup can appear, you need to create a new schema that combines the

```
# Include base DocBook schema
include "docbook.rnc"

# Include base ITS schema
include "its20.rnc"
{
    # Disable ITS directionality as DocBook has its own attribute
    its-attribute.dir = empty
}

# Add local ITS attributes to all DocBook elements
db.common.base.attributes &= its-local.attributes & its-
attribute.version?

# Allow its:rules inside info element
db.info.extension |= its-rules
```

Example 5: DocBook + ITS schema

schema of the host vocabulary with the ITS schema. In order to make this easy, the ITS specification contains highly modular schemas in RELAX NG and W3C XML Schema languages. It is rather easy to take ITS building blocks from these schemas and combine them with the host vocabulary.

Example 5 shows how to integrate ITS schema into the schema for DocBook. ITS rules are added into the `<info>` element and ITS attributes are allowed to appear on any element. Because DocBook already contains its own attribute `dir` for specifying directionality, the corresponding attribute `its:dir` is removed from the ITS schema.

Please note that the schema in example 5 had to be simplified for the purposes of this publication. The complete schema can be found at https://github.com/docbook/docbook/tree/master/relaxng/schemas/dbits.

This approach to creating a combined schema of the ITS and a host vocabulary has many advantages and is thus preferable. The resulting schema will catch both, errors in the host markup and in the injected ITS markup, and shall also identify any misplaced ITS markup. For an even more reliable check, the documents to be validated can be additionally checked against the Schematron schema that is also included with the ITS specification.

There is one obvious disadvantage, this approach requires that your document has a schema and this schema needs to be extendable with ITS support. Sometimes, this is easy – for example schema for DocBook has many hooks that make extending it very easy. Unfortunately this is not the case with all of the potential host vocabularies.

Validation of the ITS markup within HTML5 documents is rather easy because support for ITS was added as an option to some of the online validation services, such as http://validator.w3.org and http://validator.nu.

Internally, validation of HTML+ITS is driven by RELAX NG schemas. That basically means that an approach similar to the one described in Section 4, *Validating host vocabulary together with ITS markup* has been used. The underlying schemas are available from https://bitbucket.org/validator/validator/src/.

Elements based ITS rules or standoff markup must be placed inside the `<script>` element because the HTML language lacks extensibility. Unfortunately,

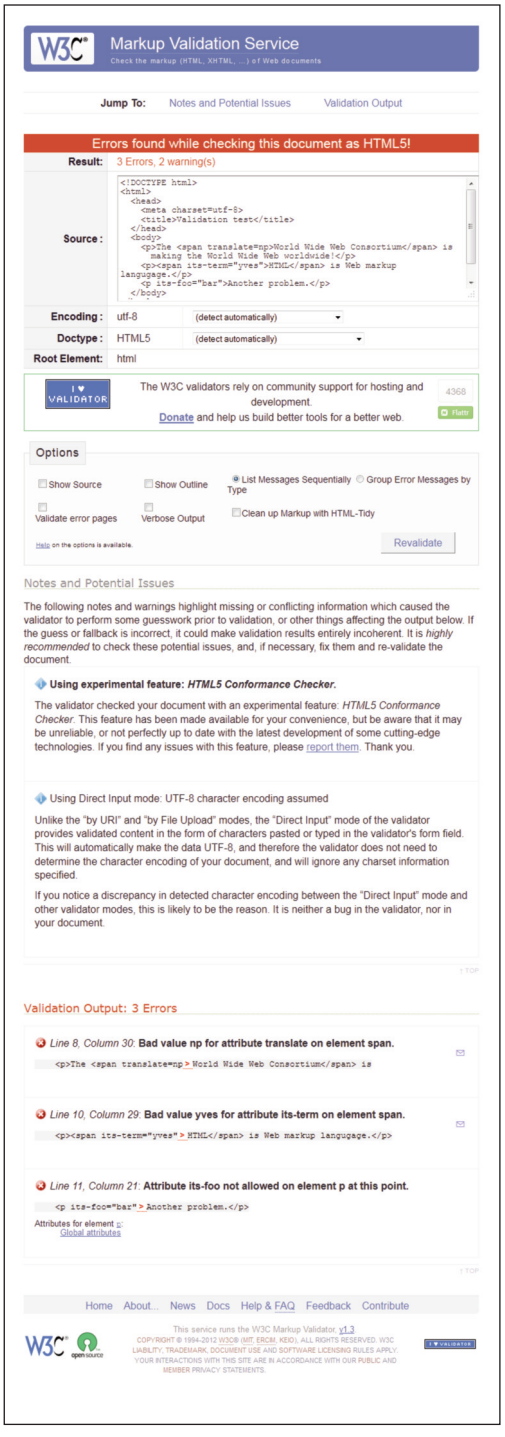

Figure 2. Result of HTML+ITS validation in W3C validator

from the validation point of view the content of this element is just an opaque string and cannot be reasonably validated. Because of this issue, it is recommended not to use any ITS elements inside an HTML page, and restrict the use to just ITS attributes. The ITS elements holding rules can then be stored in separate XML files and linked from the HTML page using the `<link>` element.

6. Tools

There are many implementations of validators. From the user perspective, the easiest option is to use a validator integrated in a popular XML editor such as the oXygen XML editor. If a commercial tool cannot be acquired, an open-source tool, such as the Jing tool (available from http://code.google.com/p/jing-trang/) is an option.

7. Conclusions

We have shown and discussed several approaches to validation of documents containing ITS markup. Potential ITS implementers should definitively include validation as one of the initial steps in procuring their ITS tool chain. This is critical to make sure that any manually or automatically produced ITS markup is conformant and thus can be successfully processed by a variety of ITS ready tools.

References

Biron, P.V., Malhotra, A. (Eds.) (2004) *XML Schema Part 2: Datatypes Second Edition* [online], Recommendation. ed, Recommendation, W3C, available: http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/ [accessed 10 Dec 2013].

Clark, J., Murata, M. (Eds.) (2001) *RELAX NG Specification* [online], Committee Specification. ed, Standard, OASIS, available: https://www.oasis-open.org/committees/relax-ng/spec-20011203.html [accessed 10 Dec 2013].

Document Schema Definition Languages (DSDL) — Part 3: Rule-Based Validation — Schematron. [online] (2006) International Standard, ISO/IEC, available: http://standards.iso.org/ittf/PubliclyAvailableStandards/c040833_ISO_IEC_19757-3_2006%28E%29.zip [accessed 10 Dec 2013].

Document Schema Definition Languages (DSDL) — Part 4: Namespace-Based Validation Dispatching Language — NVDL. [online] (2006) International Standard, ISO/IEC, available: http://standards.iso.org/ittf/PubliclyAvailableStandards/c038615_ISO_IEC_19757-4_2006%28E%29.zip [accessed 10 Dec 2013].

Fallside, D.C., Walmsley, P. (Eds.) (2004) *XML Schema Part 0: Primer Second Edition* [online], Recommendation. ed, Recommendation, W3C, available: http://www.w3.org/TR/2004/REC-xmlschema-0-20041028/ [accessed 10 Dec 2013].

Filip, D., McCance, S., Lewis, D., Lieske, C., Lommel, A., Kosek, J., Sasaki, F., Savourel, Y. (Eds.) (2013) *Internationalization Tag Set (ITS) Version 2.0* [online], Recommendation. ed, Recommendation, W3C, available: http://www.w3.org/TR/its20/ [accessed 11 Nov 2013].

Thompson, H.S., Beech, D., Maloney, M., Mendelsohn, N. (Eds.) (2004) *XML Schema Part 1: Structures Second Edition* [online], Recommendation. ed, Recommendation, W3C, available: http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/ [accessed 10 Dec 2013].

Process and Agent Classification Based Interoperability in the Emerging XLIFF 2.0 Standard

David Filip, Asanka Wasala
Localisation Research Centre,
University of Limerick, Ireland
david.filip@ul.ie, asanka.wasala@ul.ie

Abstract

In this paper, we are going to present the XLIFF 2.0 Agents classification that has been developed based on XLIFF 1.2 Interoperability research, proposed to the XLIFF Technical Committee (TC) and a simplified version of it adopted in Committee Specification Drafts of XLIFF 2.0.

We are also reviewing the state of the art and literature assessing the interoperability of localisation data interchange based on the XLIFF 1.2 OASIS standard. This paper shows how the XLIFF 1.2 based interoperability research provides valuable input for the XLIFF 2.0 standard development.

Keywords: *XLIFF 2.0, agents, classification, oasis, standard, interoperability*

1. State of the art and Literature Review

1.1 XLIFF Support in Tools

A survey conducted by Morado Vázquez and Filip (2012) reports the status of XLIFF support in Computer Aided Translation (CAT) tools. This report appeared twice so far – second time as Filip, D., Morado Vázquez, L. (2013) and tracks changes in XLIFF support in all major CAT tools. The survey was based on a questionnaire designed by the XLIFF Promotion and Liaison Sub-Committee of XLIFF TC and it is aimed mainly at CAT tool producers but also at owners of corporate XLIFF generators. The main objective of this survey is to iteratively collect information that is useful for understanding the level of tool support for XLIFF. The subcommittee is now retiring the full XLIFF 1.2 oriented questionnaire and develops a similar method for tracking XLIFF 2.0 support.

The survey reports a detailed characterisation of these tools with respect to XLIFF version support, use of custom extensions and XLIFF element and attributes support. In their survey, they avoid the use of the word "support" due to its ambiguous and prompting nature. Instead, they used the phrase "actively used elements" during the data collection phase. Only "Yes" and "No" answers have been collected. As such, the level of tool support for a certain element or attribute is questionable (e.g. given that a tool "actively uses" the <file> element, it does not necessarily imply that it conforms to the XLIFF mandatory requirements for the <file> element). Also it is important to note that this survey is based on the toolmaker's self-assessment which is not being technically verified beyond spotting and eventually reporting grave inconsistencies in their answers. The work of the XLIFF promotion subcommittee was originally inspired and partially based on the work of Bly (2010), who analysed XLIFF support in commonly used tools and presented a matrix containing tools and their level of support for individual XLIFF elements.

Despite serious limitations in the methodology used by Bly (2010), he concluded a valuable insight, namely that tool vendors can conform to standards but still lock in users with their tools. Moreover, he discussed various problems associated with the XLIFF standard, such as its inability to support all the features offered by tools, and its lack of tight definitions. All this notwithstanding, Bly is convinced that "XLIFF is the best (only) hope for interoperability."

In another study, Morado-Vázquez and Wolff (2011) present the Open Source CAT tool "Virtaal" that claims to support XLIFF. They compare its level of XLIFF support with the matrix presented by Bly. Bly's (2010) top-down analysis of XLIFF implementations show their level of support for different XLIFF elements.

Morado-Vázquez and Wolff conclude that Virtaal is better in terms of XLIFF support than the average XLIFF editing tools checked in Bly's study.

Interestingly, Morado-Vázquez and Wolff point out a weakness in Bly's methodology. Bly's analysis does not take into account the relative importance of different parts of the XLIFF specification. To address this issue Morado-Vázquez and Wolff propose a "weighted sum model" as a possible improvement, however details have not been included. Furthermore, they highlight the importance of an element's attribute and attribute values for tool interoperability. Similar to Bly's, the exact methodology used to evaluate their tool, the test suites or the use of the term "support" are not included in their publication. Although they mention the use of Bly's analysis methodology to evaluate Virtaal, the paper does not make explicit references to Bly's methodology or test suites for evaluating Virtaal.

In Anastasiou and Morado-Vázquez (2010), several interoperability tests were performed with three XLIFF compliant CAT tools. Like Bly (2010), they classified selected tools into two categories: XLIFF converters (i.e. generators or extractors) and XLIFF editors. Out of the three CAT tools selected, they found that two had the capability to generate XLIFF content and three had the capability to edit XLIFF content, so they were interested in four combinations: for each converter they wanted to see if the other two editing tools could edit the generated content. The researchers' methodology involved five steps:

1) conversion of a selected HTML file into XLIFF (using the two converters);

2) validation of the converted XLIFF file;

3) manipulation of the XLIFF file using the editors;

4) manual analysis of the XLIFF file;

5) back conversion of the file into HTML and a manual analysis of the converted file.

The results showed that out of the four combinations (i.e. XLIFF generators and editors) considered in this research, only one pair of tools seems to interoperate successfully. The authors recommend "simplicity, better communication between localisation stakeholders and clarification of specification" of the standard and suggest future work on expanding the experiment with more CAT tools as well as different file types. It should also be noted that their experiment only considers the back-conversion of the XLIFF files using the tool used to generate the XLIFF file. A better analysis could be carried out if all possible scenarios were taken into account during the back-conversion process too.

One of the reasons behind lack of tool support of XLIFF standard is the absence of a proper conformance clause in XLIFF (Filip 2011; Anastasiou and Morado-Vázquez 2010). This also reflects Bly's 'lack-of-definition' finding. Anastasiou (2011) stresses that "conformance clauses should include criteria about compliance with both Localisation and Semantic Web standards."

1.2 Different Levels of Tool Support

In this research, we compiled a large XLIFF corpus by collecting over 3000 XLIFF files with the aid of CNGL industrial partners and by crawling openly available XLIFF files in the web. We then analysed the XLIFF files for their XLIFF feature usage characteristics. In the following, we present the overall frequency distribution of XLIFF elements in our corpus in Fig 1.

An analysis of the above Graph reveals a connection between the lack of tools support for certain XLIFF elements and the frequency of use of XLIFF elements. Our research revealed that many XLIFF features are either not supported or only partially supported by tools (Anastasiou 2010; Bly 2010; Lieske 2011). This has inevitably led to interoperability issues.

According to Shan and Kesan (2008) a valid reason for tool developers not to offer full interoperability is the lack of need to support all the features in their tools. This seems to be due to two main reasons:

1. Business case requirements

   Depending on the requirements of different business cases, different tools have been implemented to support different parts of the XLIFF specification. There are localisation tools that do not support XLIFF at all, which is mainly due to the fact that there is no strong business case demanding XLIFF support from these tool vendors.

2. Complexity and limitations of the standard

   Although XLIFF's formal tool compliance is easy to achieve, complete XLIFF feature implementation in tools is difficult due to the

complexity of the standard (Anastasiou and Morado-Vázquez 2010). Eventually the document conformance is relatively well addressed in the XLIFF 1.2 OASIS standard despite the lack of a formal Conformance Clause. However, there are virtually no conformance hints targeting application conformance. We will see how the XLIFF TC learnt from this and introduced an explicit application conformance target in its current Public Review Drafts (Comerford, T., Filip, D., Raya, R.M., Savourel, Y.; Eds. 2013a, b). The application conformance target was explicitly added between the 1st and 2nd public reviews along with the Processes and Agents classification, based on the discussions concluded in the June 2013 XLIFF TC face to face meeting.

2. Agent Classification: Proposed Methodology I

Shah and Kesan (2008) state that users expect 100% interoperability among implementations for various reasons including actual requirements as well as avoiding potential problems. In order to achieve 100% interoperability among agents, ideally the agents need to implement all the features specified in the standard. However, as we identified in Section 1.2, this is not realistic, at least not in the XLIFF 1.2 case.

In this paper, we propose some alternative methodologies for improving interoperability among agents. This is by classifying agents based on their supported features or process driven feature subsets.

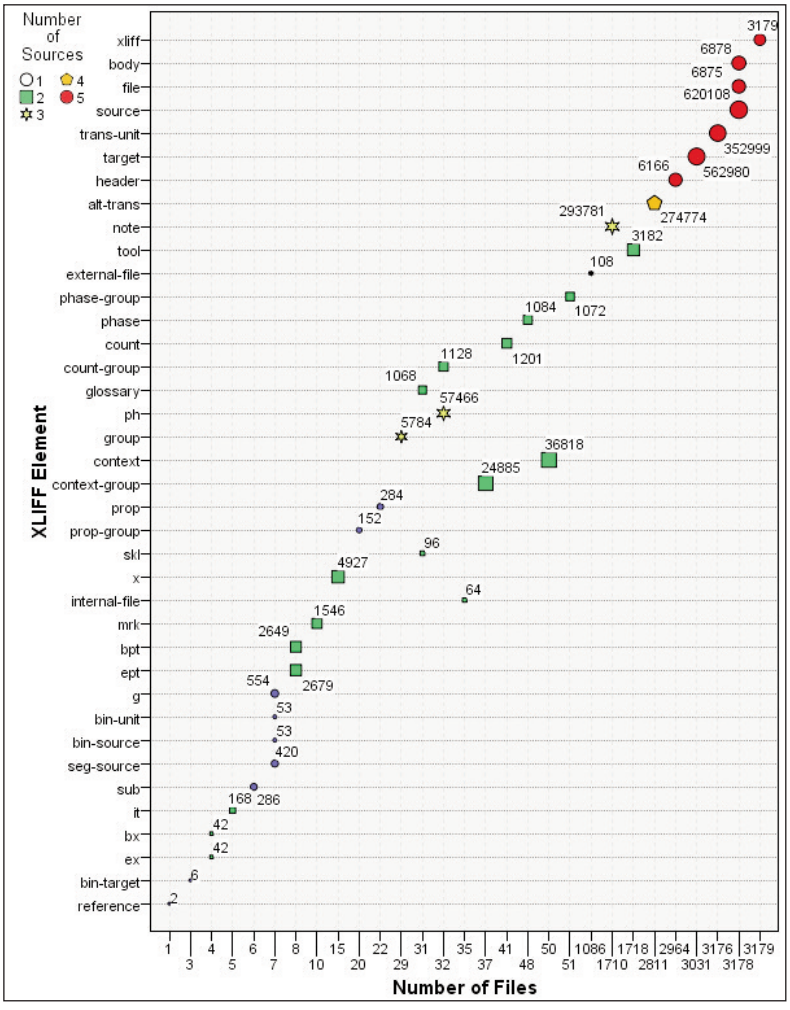

Figure 1. XLIFF 1.2 Feature Distribution in Files and Organisations

The first presented methodology is based on the assumption that agents that implement the same set of features will have full interoperability among those agents.

The features used in both a majority of organisations and files can be considered as most important features of the standard in terms of their usage. Then based on the importance, features can be categorised into several levels. This categorisation should be carried out by the standardisation committee, such as the XLIFF Technical Committee (TC). The frequency distribution of features can be used as an aid to define different levels. An example classification is given below.

level 4: xliff, file, header, body, trans-unit , source, target;

level 3: alt-trans, note, ph, group;

level 2: tool, external-file, phase-group, phase, count, count-group, glossary, context, context-group, prop, prop-group, skl, x, internal-file, mrk, bpt, ept;

level 1: g, bin-unit, bin-source, seg-source, sub, it, bx, ex, bin-target, references.

The above example has been mainly derived by analysing the XLIFF 1.2 elements distribution in our corpus. Levels have been primarily defined based on the importance of features. For example, under the level 4, the features used by all five organisations are listed. From the corpus, it is evident that these features are also used in majority of files. Similarly, the features used by at least 4 organisations have been categorised as level 3 and so forth.

It is important to note that elements cascade from top to bottom in these levels. For example, elements that belong to level 3 include all the elements listed under level 4 in addition to the explicitly listed elements (i.e. the complete feature set of level 3 consists of all level 4 elements and elements: alt-trans, note, ph, group).

Once different levels have been established in agreement with the TC of the standard, agents can be classified into these levels based on agent's feature support. An agent that has implemented features "xliff, file, header, body, trans-unit , source, target" is classified under the level 4, whereas an agent that has implemented features "xliff, file, header, body, trans-unit , source, target, alt-trans, note, ph, group" is classified as level 3. Therefore, an agent that has implemented all the features is classified as a level 1 agent.

After defining the levels, the next major step involves preparations of test suites. Separate test suites have to be developed covering the feature subset defined for each of the above levels. Then these test suites can be used to evaluate agents' level of XLIFF support. Finally, the agents can be classified by their level of XLIFF support based on the test results.

However, it is likely that agents may only support a sub-set of features of each level. In such scenario, the TC should come to agreements on essential tests of each level that need to be passed by an agent, in order to be able to classify it under a certain level.

This proposed methodology has been devised based on XLIFF 1.2 and would be usable as is if XLIFF TC continued in development of an interchange standard backwards compatible with 1.2. However, XLIFF 2.0 uses a different set of elements and cannot therefore use an XLIFF 1.2 elements based categorization of agents. The solution for XLIFF 2.0 however builds on lessons learnt from the XLIFF 1.2 based interoperability research.

3. Agent Classification: Proposed Methodology II

This agent classification has been developed specifically for XLIFF 2.0. Instead of generic support levels without any specific process focus, as discussed in Section 2, this classification methodology is based on the generalized Business Process Model of the XLIFF payload and metadata interchange. See Fig 2. This approach was inspired by the business process driven development of the UBL standard (Bosak, J., McGrath, T., Holman, K.G.; Eds., 2006, 2013).

Based on lessons learnt from the XLIFF 1.2 adoption that have been extensively discussed in Section 1, the XLIFF TC decided to specify a small core specification as the smallest common denominator and thus a secure base for interoperability.

Based on XLIFF TC discussions and a formal ballot, the TC decided to include in core only elements and attributes that are essential for extraction, translation

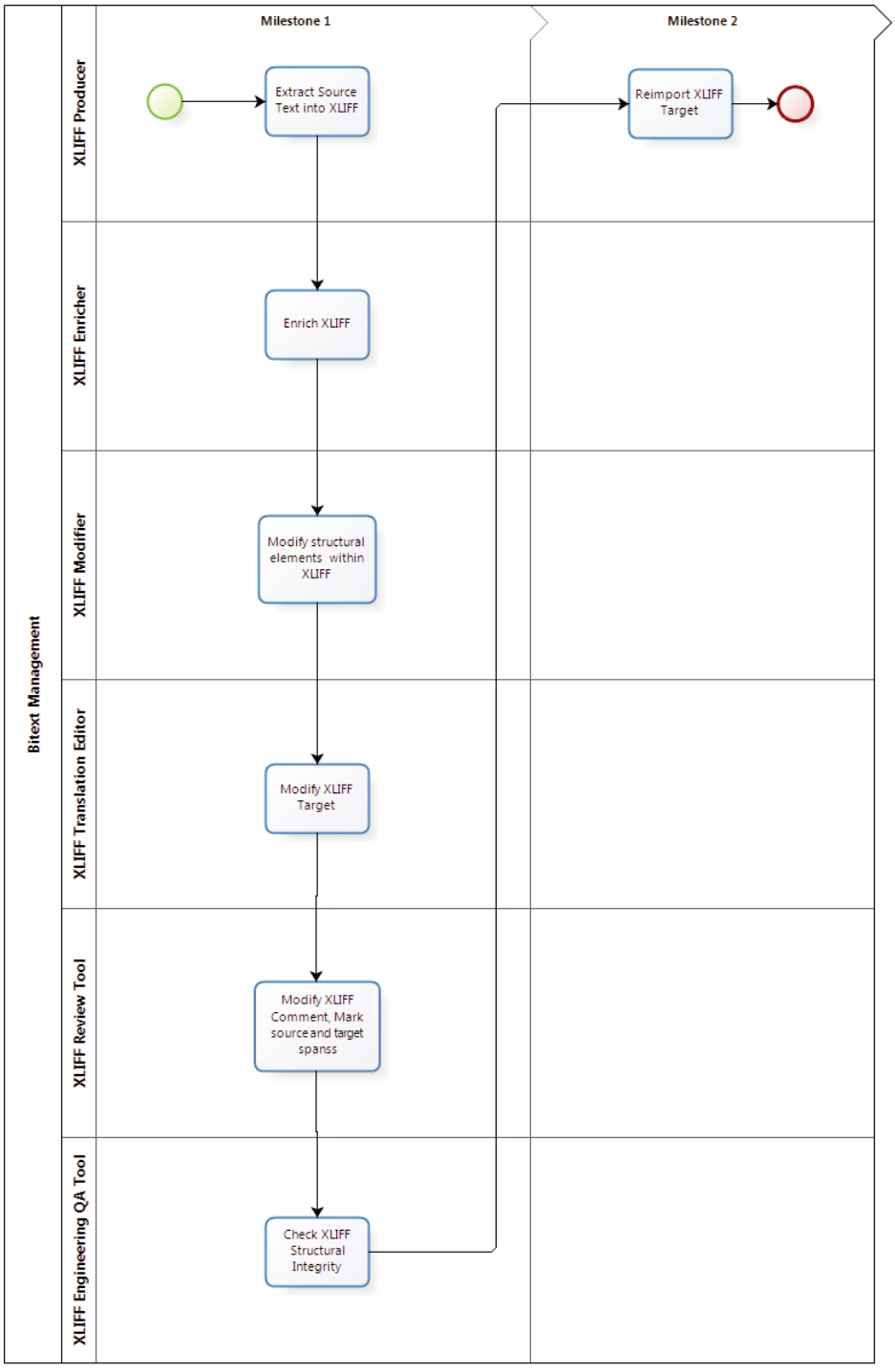

Figure 2. Generalized Business Process Model of the XLIFF Payload and Metadata Interchange

and merging back of content into the source format in the target natural language. Another possible approach would have been to look at all possible process areas that are being served by XLIFF as the interchange format and include basic functionality from each area in the core while handling the advanced functionalities in modules and possibly extensions. This approach however was not chosen by the TC. So there is, for instance, not even a basic size restriction mechanism in XLIFF core, while a comprehensive and extensible general size and length restriction mechanism has been specified as one of XLIFF 2.0 modules.

Among other popular features known from XLIFF 1.x that did not make it into the core of XLIFF 2.0, we can mention the `<alt-trans>` element. Instead of the former core element for candidate translations (and other related versions) there is a comprehensive Translation Candidates module that however sheds the semantic overload of the original XLIFF 1.2 element.

3.1 The original proposal for Normative Process and Agent related Terminology in XLIFF 2.0

The original proposal has been presented to the TC on various occasions the last time as an Initial Public Review Comment (Filip 2013).

The following has been proposed as a set of normative process and agent definitions and requirements, along with non-normative notes and warnings:

[Definitions]

Agent – a tool that does anything to an XLIFF file from extract to merge inclusively.
Extract/Extraction – the process of encoding localizable content from a native content or User Interface format as XLIFF payload, so that localizable parts of content in the source language are available for translation into the target language along with necessary context.
Extractor (Agent) – an Agent that performs the Extraction process.

Merge – the process of importing XLIFF payload back to the originating native format, based on the full knowledge of the extraction mechanism, so that the localized content or User Interface strings replace the source language in the native format.

Merger (Agent) –an Agent that performs the Merge process.

Warning:

Unless specified otherwise, Merger is deemed to have the same knowledge of the native format as the Extractor throughout the specification. Mergers independent of Extractors can succeed, but it is out of scope of this specification to specify interoperability for merging back without the full extractor knowledge of the native format.

[Definitions]

Enrich/Enriching – the process of associating module and extension based metadata and resources with the extracted XLIFF payload.

[Processing Requirements] PR:

Enriching MAY happen at the time of extraction.

[Definitions]

Enricher (Agent) – an Agent that performs the Enriching process.

Note:

Extractor knowledge of the native format is not assumed while Enriching.

[Definitions]

Modify/Modification – the process of changing core XLIFF structural elements and bin-file module elements.

Modifier (Agent) – an Agent that performs the Modification process.

[Processing Requirements] PR:

Structural elements MAY be Modified and Enriched at the same time. Modifier MUST be able roll back the core and bin-file structure of an XLIFF that it has previously modified.

Note:

Enricher or Extractor knowledge of the native format is not assumed while Modifying.

Warning:

Rollback of Modifications performed by a different Modifier is out of scope.

[Definitions]

Edit Source/Source Editing – the process of changing payload of <source> children of

 elemensts.

Source Editor (Agent) – an Agent that performs Source Editing.

[Processing Requirements] PR:

Source editing MAY be performed at the time of Modification.

[Definitions]

Translate/Translation - a rendering of the meaning of source text, expressed in the target language. As an XLIFF based process it means creating and changing <target> children of elements.

Translation Editor (Agent) – an Agent that performs the Translation process.

Review/Revision – the process of creating or changing any annotation elements, attributes or values specified in core, modules or extensions. This includes marking spans of <source> and <target> children of elements.

Revision Agent – an Agent that performs the Revision process.

[Processing Requirements] PR:

Revision MAY be performed at the time of Translation.

Revision MAY be performed at the time of Modification.

[Definitions]

Validate/Validation – the process of checking XLIFF payload against any rules specified via the Validation Module or any general, core or module specific Processing Requirements.

Validator (Agent) – an Agent that performs the Validation process.

[Processing Requirements] PR:

Validators MAY add test results via the Validation Module.

Validator MUST not modify any other XLIFF elements, attributes or values.

Validation MUST NOT be performed at the time of Merge.

Note:

Extractor/Merger Agents will benefit from use of a Validator. Even though the Validator will often be a part of the same tool/platform, it is important to distinguish between validation and the actual Merge. Validation will routinely precede Merge, while failed Validation will lead to exception handling, such as returning the XLIFF file to a Revision Agent or Translation Editor, Modifier etc.

3.1 Further Developments

The original Processes and Agents proposal as presented in Section 3.1 has developed based on XLIFF TC discussions, notable the face to face discussion at the second FESGILTT event in London, June 2013. The proposal also had to adapt to developments of the XLIFF 2.0 core and modules specification.

The first TC approved draft of the solution appeared in the Second Public Review of the XLIFF 2.0 Committee Specification Draft (Comerford, T., Filip, D., Raya, R.M., Savourel, Y.; Eds. 2013a). The TC did not agree to inclusion of Processing Requirements that would prescribe a partial ordering for the XLIFF facilitated processes and the TC also did not use the full detailed scale of process and agent definitions.

As result, the specification only recognizes Extractors, Mergers, Enrichers, Modifiers, and Writers, as specific types of Agents. All Processing Requirements targeting unspecified Agents are deemed to target any type of agents irrespective of process specialisation. This debate also led to explicit differentiation between static Constraints (on top of data type constraints based on elements and attribute definitions that can be expressed in schema) and Agents targeting Processing Requirements. Processing Requirements in the specification now always specify what types of transformations unspecified or specialized Agent can perform on the static documents.

Other types of agents were not deemed necessary for the normative provisions that the standard specification has to provide. So Validators and Revision Agents are considered specific types of Enrichers. Source and Translation Editors are considered special types of Modifiers. Interestingly, source payload editing is not an allowed XLIFF transformation as per the current XLIFF 2.0 Working Draft; source content can be only enriched or re-

segmented but not modified.

Those subtypes that did not make it into the normative specification might be still useful to discern in the context of a non-normative interoperability debate, for instance for describing use cases in a language accessible for industry.

Writers that did not feature in the original proposal are a superset of Extractors, Enrichers, and Modifiers. However, there might as well be Writers that are neither of the former, since XLIFF is intended as an interchange format and not as a processing format.

3.3 The Resulting Solution

The following is the resulting XLIFF 2.0 classification of Processes and Agents that is included in the XLIFF 2.0 specifications as of the 2nd Public Review Draft (Comerford, T., Filip, D., Raya, R.M., Savourel, Y.; Eds. 2013a).

Definitions

Agent

any application or tool that generates (creates), reads, edits, writes, processes, stores, renders or otherwise handles ***XLIFF Documents***.

Agent is the most general application conformance target that subsumes all other specialized user agents disregarding whether they are defined in this specification or not.

Enrich, Enriching

the process of associating module and extension based metadata and resources with the ***Extracted*** XLIFF payload

Processing Requirements

- ***Enriching*** MAY happen at the time of ***Extraction***.

Note

Extractor knowledge of the native format is not assumed while ***Enriching***.

Extract, Extraction

the process of encoding localizable content from a native content or User Interface format as XLIFF payload, so that localizable parts of the content in the source language are available for ***Translation*** into the target language along with the necessary context information

Extractor, Extractor Agent

any ***Agent*** that performs the ***Extraction*** process

Merge, Merging

the process of importing XLIFF payload back to the originating native format, based on the ***full knowledge*** of the ***Extraction*** mechanism, so that the localized content or User Interface strings replace the source language in the native format

Merger, Merger Agent

an ***Agent*** that performs the ***Merge*** process

Warning

Unless specified otherwise, any ***Merger*** is deemed to have the same knowledge of the native format as the ***Extractor*** throughout the specification.

Mergers independent of ***Extractors*** can succeed, but it is out of scope of this specification to specify interoperability for ***Merging*** back without the full ***Extractor*** knowledge of the native format.

Modify, Modification

the process of changing core and module XLIFF structural and inline elements that were previously created by other ***Writers***

Processing Requirements

- XLIFF elements MAY be ***Modified*** and ***Enriched*** at the same time.

Note

Extractor or ***Enricher*** knowledge of the native format is not assumed while ***Modifying***.

Modifier, Modifier Agent

an ***Agent*** that performs the ***Modification*** process

Warning

Unless specified otherwise, any ***Merger*** is deemed to have the same knowledge of

the native format as the ***Extractor*** throughout the specification.

Mergers independent of ***Extractors*** can succeed, but it is out of scope of this specification to specify interoperability for ***Merging*** back without the full ***Extractor*** knowledge of the native format.

Translation, Translate
a rendering of the meaning of the source text, expressed in the target language

Writer, Writer Agent
an ***Agent*** that creates, generates, or otherwise writes an ***XLIFF Document*** for whatever purpose, including but not limited to ***Extractor***, ***Modifier***, and ***Enricher Agents***.

Note
Since XLIFF is intended as an exchange format rather than a processing format, many applications will need to generate ***XLIFF Documents*** from their internal processing formats, even in cases when they are processing ***XLIFF Documents*** created by another ***Extractor***.

These definitions provide a normative base for writing Processing Requirements throughout the whole XLIFF 2.0 specification and allow for specific application conformance targeting in the Conformance Section:

2. Application Conformance

a XLIFF *Writers* MUST create conformant *XLIFF Documents* to be considered XLIFF compliant.
b *Agents* processing conformant *XLIFF Documents* that contain custom extensions are not REQUIRED to understand and process non-XLIFF elements or attributes. However, conformant applications SHOULD preserve existing custom extensions when processing conformant *XLIFF Documents*, provided that the elements that contain custom extensions are not removed according to XLIFF Processing Requirements or the extension's own processing requirements.
c All *Agents* MUST comply with Processing Requirements for otherwise unspecified *Agents* or without a specifically set target *Agent*.
d Specialized *Agents* defined in this specification - this is *Extractor*, *Merger*, *Writer*, *Modifier*, and *Enricher Agents* - MUST comply with the Processing Requirements targeting their specifically defined type of *Agent* on top of Processing Requirements targeting all *Agents* as per point 3.[c] above.
e XLIFF is a format explicitly designed for exchanging data among various *Agents*. Thus, a conformant XLIFF application MUST be able to accept *XLIFF Documents* it had written after those *XLIFF Documents* were *Modified* or *Enriched* by a different application, provided that:
   i. The processed files are conformant *XLIFF Documents*,
   ii. in a state compliant with all relevant Processing Requirements.

4. Defining "Support"

The term "support" is a widely used term related not to XLIFF implementations (tools), it is a key interoperability term in general. A few examples pertaining to XLIFF specifically are given below:

- *the tool supports XLIFF;*
- *the tool supports feature X (e.g. tool supports in-line mark-up elements);*
- *the tool partially supports feature X.*

As discussed in detail in Section 1.1 lack of a proper definition for this term had led to many confusions. It can also be conjectured that the term "support" has been used by some of the tool developers just as a marketing slogan. Therefore, we recommend that one of the first steps towards addressing interoperability in the new incarnation of XLIFF is a rigid definition of the term "support" in the above contexts.

4.1 Defining "Support" Based on a Feature Complete Reference Implementation

One of the possible approaches to defining support assumes that a comprehensive open-source reference implementation exists, and given that this assumption has been fulfilled, "support" may be defined as follows:

- The tool must operate on X as expected and described by the XLIFF specification and in all possible variations of X (e.g. for all possible

attribute/value combinations),
AND
- The tool must operate on X in the same manner as the reference implementation does
OR
- The outcome of the tool manipulation of the X must be equivalent to the outcome of the feature manipulation by the reference implementation in all possible manipulation scenarios.

In order to claim that a tool ***supports*** XLIFF:

- Given any possible valid variation of XLIFF content, if and only if the outcome of the manipulated content by the tool is equivalent to the outcome of the same manipulation performed by the reference implementation, the tool ***supports*** XLIFF.

As far as the above approach to defining the term support depends on the existence of a feature complete reference implementation, it is not realistic to rely on the above definition in practice, at least not until such a comprehensive open source implementation that is backed by industry consensus exist.

Anyway, given the modular character of XLIFF 2.0 and following the current efforts of implementers that are likely to provide Statements of Use that are required for XLIFF becoming a Candidate Standard and an official OASIS Standard in the due course, it is possible that there soon (early 2014) will be a comprehensive Open Source implementation of the core. However, if we are looking for coverage for the whole XLIFF 2.0 specification including its 8 modules, we are more likely looking for an ecosystem of Open Source and closed source tools.
Also standards specifications must be implementation agnostic. Of course, as standards they must be implementable, yet must not prescribe any specific implementation. Therefore "support" must be defined at a lower theoretical level that is grounded in the specification itself without referencing a particular implementation.

Defining "Support" Based on the Normative Provisions of the Specification

The new XLIFF specification contains a dedicated Conformance section. This is partially to conform to a new non-negotiable requirement that OASIS introduced as part of its TC process and, more importantly, due to the lessons learnt with compromised interoperability of XLIFF 1.x implementations. It is fair to say that while XLIFF 1.2 had covered static validity fairly well due to having relatively rigid element definitions and XML Schema based validation there had been zero guidance for application conformance. The good practices of static validation were of course adopted also for XLIFF 2.0. However, on top of explicit static document conformance, the XLIFF 2.0 specification now explicitly addresses XLIFF agents as its application conformance targets.

In analogy with the reference implementation based attempt, we could try and define "support" as follows.

In order to claim that a tool ***supports*** feature X:

- The tool must operate on X as expected and described by the XLIFF specification and the tool outcomes must satisfy all static conditions and constraints as well as processing requirements related to the feature, that in all possible variations of X (e.g. for all possible attribute-value combinations).

In order to claim that a tool fully ***supports*** XLIFF, the tool must ***support*** all the XLIFF features.

The above can be called a maximalist definition of "support" that is unfortunately not very useful in real life scenarios. Luckily the spec now works with different subsets of agents based on what processes the agents are capable of ***supporting***.

1 For processing requirements addressing any or unspecified XLIFF Agents, all agents must conform to the given Processing Requirement
2 For processing requirements addressing XLIFF Writers, all writer agents must conform to the given Processing Requirement. As discussed above the Writers are a superset of Extractors, Enrichers, and Modifiers. However, there might as well be Writers that are neither of the former, since XLIFF is intended as an interchange format and not as a processing format.
3 For processing requirements addressing XLIFF Extractors, all Extractor agents must conform to the given Processing Requirement.
4 For processing requirements addressing XLIFF Enrichers, all Enricher agents must

conform to the given Processing Requirement.

5 For processing requirements addressing XLIFF Modifiers, all Modifier agents must conform to the given Processing Requirement.

6 For processing requirements addressing XLIFF Mergers, all Merger agents must conform to the given Processing Requirement.

Thus the defined support extent can be effectively sub-setted, so that specialized tools can claim support in an unequivocal and sensible way without the need to support irrelevant features or transformations.

So for instance a tool that specializes in Enriching XLIFF Documents with Translation Candidates does not need to worry about a significant subset of Processing Requirements that are addressing Modifiers, Extractors, and Mergers, while it must conform to all Processing Requirements set forth for (unspecified) Agents, Writers, and Enrichers.

5. Conclusion

The approach to assessing support described in Section 4.2 has immense advantages; it lowers the cost of standards based interoperability within and across supply chains. Instead of requiring that all tools of a certain level support all touched elements and attributes in all respects, adopters can look into building modular workflows of specialized contributing tools between the process bracket of Extraction and Merging of the translatable content. This surpasses the interchange paradigm of XLIFF 1.2, which was intended as a non-transitive "fire-and-die" format, whereas XLIFF 2.0 explicitly targets the whole tool chain between and including Extraction and Merging back of content.

If your Extractor/Merger supports the XLIFF Core and a module X, you are looking for Modifiers and Enrichers that support the core in their respective capacities and are capable of processing the required module. You can build a workflow that contains a tool that is unaware of a specific well defined subset of functionality of core or modules, yet you can be sure that if the specialized tool had done just its own job and stuck to all requirements relevant to its own transformations, the rest of the payload and metadata won't be harmed.

Because XLIFF 2.0 Core is the smallest possible common denominator, support for all static constraints and Processing Requirements targeting just the relevant type of Agents (as explained above) is binary (yes/no) and non-negotiable. Nevertheless, the Core and Modules can be supported by tools in different capacities.

The modular and process classification based approach to conformance targets allows the assessment of tools support in a practical way based on normative requirements set out in the specification itself, which was not possible in the XLIFF 1.2 predecessor standard. XLIFF 2.0 shall be a better standard thanks to the lessons learnt from XLIFF 1.2 adoption.

References

Anastasiou, D. (2010) 'Open and flexible localization metada', *MultiLingual,* 21(4), 50-52.

Anastasiou, D. (2011) 'The Impact of Localisation on Semantic Web Standards', *European Journal of ePractice,* 12(March/April 2011), 42-52.

Anastasiou, D. and Morado-Vázquez, L. (2010) 'Localisation Standards and Metadata' in Sánchez-Alonso, S. and Athanasiadis, I. N., eds., *Metadata and Semantic Research*, Springer Berlin Heidelberg, 255-274.

Berges, I., Bermudez, J., Goñi, A. and Illarramendi, A. (2010) 'Semantic interoperability of clinical data', in *Proceedings of the First International Workshop on Model-Driven Interoperability*, Oslo, Norway, 1866275: ACM, 10-14.

Bly, M. (2010) 'XLIFF: Theory and Reality: Lessons Learned by Medtronic in 4 Years of Everyday XLIFF Use', in *1st XLIFF Symposium*, Limerick, Ireland, 22 Sept, University of Limerick.

Bosak, J., McGrath, T., Holman, K.G. (Eds.) (2006) *Universal Business Language v2.0* [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/ubl/os-UBL-2.0/UBL-2.0.html [accessed 11 Dec 2013].

Bosak, J., McGrath, T., Holman, K.G. (Eds.) (2013) *Universal Business Language v2.1* [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/ubl/os-UBL-2.1/UBL-2.1.xml [accessed 11 Dec 2013].

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2013a) *XLIFF Version 2.0* [online], Committee Specification Draft 02 / Public Review Draft 02. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/csprd02/xliff-core-v2.0-csprd02.html [accessed 17 Oct 2013].

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2013b) *XLIFF Version 2.0* [online], Committee Specification Draft 01 / Public Review Draft 01. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/csprd01/xliff-core-v2.0-csprd01.html [accessed 22 Jul 2013].

Filip, D. (2011) 'XLIFF 2.0', in *Multilingual Web Workshop*, Pisa, Italy, 4-5 Apr.

Filip, D. (2013) 'Re: XLIFF 2.0 csprd01 comment by David Filip - classification of processes and agents to improve precision of conformance statements', available: https://lists.oasis-open.org/archives/xliff/201305/msg00054.html [accessed 11 Dec 2013].

Filip, D., Morado Vázquez, L. (2013) *XLIFF Support in CAT Tools*, Subcomittee Report 2, XLIFF State of the Art, OASIS XLIFF TC, available: http://www.localisation.ie/resources/SurveyReport2ndEditionApproved.pdf [accessed 3 Dec 2013].

Frimannsson, A. and Lieske, C. (2010) 'Next Generation XLIFF: Simplify-Clarify-and Extend', in *1st XLIFF International Symposium* Limerick, Ireland, 22 Sept, University of Limerick.

Heiler, S. (1995) 'Semantic interoperability', *ACM Comput. Surv.,* 27(2), 271-273.

Imhof, T. (2010) 'XLIFF – a bilingual interchange format', in *MemoQ Fest*, Budapest, Hungary, 5-7 May.

Lewis, G. A., Morris, E., Simanta, S. and Wrage, L. (2008) 'Why Standards Are Not Enough to Guarantee End-to-End Interoperability', in *Proceedings of the Seventh International Conference on Composition-Based Software Systems (ICCBSS 2008)*, 1343630: IEEE Computer Society, 164-173.

Lewis, D., O'Connor, A., Molines, S., Finn, L., Jones, D., Curran, S., & Lawless, S. (2012a). 'Linking localisation and language resources'. In *Linked Data in Linguistics*. Springer Berlin Heidelberg, 45-54.

Lewis, D., O'Connor, A., Zydron, A., Sjögren, G., & Choudhury, R. (2012b). 'On Using Linked Data for Language Resource Sharing in the Long Tail of the Localisation Market'. In *LREC,* 1403-1409.

Li, W. and Li, S. (2004) 'Improve the semantic interoperability of information', in *Proceedings. 2004 2nd International IEEE Conference Intelligent Systems*, 22-24 Jun, 591-594.

Morado Vázquez, L. and Filip, D. (2012) 'XLIFF Support in CAT Tools', available: http://www.localisation.ie/resources/XLIFFSotAReport_20120210.pdf [accessed 5 Mar 2012].

Morado-Vázquez, L. and Wolff, F. (2011) 'Bringing industry standards to Open Source localisers: a case study of Virtaal', *Tradumàtica,* 9, 74-83.

Ouksel, A. M. and Sheth, A. (1999) 'Semantic interoperability in global information systems', *SIGMOD Rec.,* 28(1), 5-12.

Park, J. and Ram, S. (2004) 'Information systems interoperability: What lies beneath?', *ACM Trans. Inf. Syst.,* 22(4), 595-632.

Ray, S. R. (2009) 'Healthcare interoperability - lessons learned from the manufacturing standards sector', in *Automation Science and Engineering, 2009. CASE 2009. IEEE International Conference on*, 22-25 Aug, 88-89.

Sartipi, K. and Yarmand, M. H. (2008) 'Standard-based data and service interoperability in eHealth systems', in *ICSM 2008. IEEE International Conference on Software Maintenance* 28 Sept-4 Oct, 187-196.

Savourel, Y., Reid, J., Jewtushenko, T., Raya, R.M. (Eds.) (2008) *XLIFF Version 1.2* [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/v1.2/os/xliff-core.html [accessed 3 Dec 2013].

Shah, R. and Kesan, J. (2008) 'Evaluating the interoperability of document formats: ODF and OOXML as examples', in *Proceedings of the 2nd international conference on Theory and practice of electronic governance*, Cairo, Egypt, 1509141: ACM, 219-225.

Lieske, C. (2011) 'Insights into the future of XLIFF', *MultiLingual,* 22(5), 51-52.

Wasala, A., Filip, D., Exton, C. and Schäler, R. (2012a) 'Making Data Mining of XLIFF Artefacts Relevant for the Ongoing Development of the XLIFF Standard', in *3rd International XLIFF Symposium, FEISGILTT 2012*, Seattle, USA, 17-19 Oct.

Waters, J., Powers, B. J. and Ceruti, M. G. (2009) 'Global Interoperability Using Semantics, Standards, Science and Technology (GIS3T)', *Computer Standards & Interfaces,* 31(6), 1158-1166.

Visualization of ITS 2.0 Metadata for Localization Process

Renat Bikmatov[1], Nathan Glenn[2], Serge Gladkoff[1], Alan Melby[2]
[1] Logrus International
[2] LinguaTech International
www.logrus.ru, www.linguatech.com
renatb@logrus.ru, nathan.g.glenn@gmail.com, sgladkoff@logrus.ru, akmtrg@byu.edu

Abstract
The Internationalization Tag Set (ITS) 2.0 specification was introduced by W3C in 2013 as a complement to the XML, HTML5 and XLIFF specifications. ITS 2.0 format can be used to exchange localization instructions and other context metadata between data processing tools, and also to deliver metadata to any person who is working on the content. Previewing ITS 2.0 metadata was the goal of this project. We present preview tools which enable translators and reviewers to refer to localization context information visually presented in a web browser window while working on the content in their content editor, CAT or other tool. The core ITS metadata categories, such as Localization Note, Terminology, and Translate, will help to bind the localization instructions to the content, and also to fill the gap between limited functionality of current CAT tools and required access to context information.

Keywords: *ITS 2.0 metadata, metadata, context, preview, localization, localization instruction, translation, CAT, editing, machine translation, post-editing, XML, HTML, HTML5, XLIFF*

1. Introduction

The postulates listed below in this section are based on the rich experience of Logrus as a localization company.

Localization in general is still suffering from the following unresolved problems: 1) previewing the source content in the final or publication format, and 2) supplying localization-related context information and instructions to translators and editors. With the increased use of CMS and a mass transition to asynchronous update and fragmented translation of content by bits and pieces, these problems are only becoming more severe. Linking glossaries, translation instructions and style guides to the source content presents another problem.

The source content is usually provided to translators as XML or XLIFF files. The source content provided for preview usually comes as raw XML without any support for its visualization. These formats are not easily readable; contextual information is often missing. Lack of mapping of terminology, trademarks, client instructions and other context information to the content to be localized has been identified as major cause of disruption of human work. The Internationalization Tag Set (ITS) was created to relieve these problems.

2. Existing Limitations on ITS Usage

In real world production environments, direct integration of ITS 2.0 or any other metadata into content to be localized is hampered by the following issues:

- Support of ITS 2.0 by available CAT and other authoring tools is either missing or limited.
- The separate implementations of ITS 2.0 by many CAT tools are sure to contain many discrepancies.
- Even if one day all authoring tools could fully support ITS 2.0 format, compatibility with legacy Translation Memories (TM) would still be an issue. New pieces of content enriched with ITS metadata markup would not fully match the same pieces without the metadata. As a result, you would not be able to reuse 100% matches, for example.
- Full support of ITS 2.0 metadata, local markup, global rules and external data by existing translation memory formats and engines still remains an open issue.
- ITS 2.0 representation in XLIFF format is still a work in progress. For more details, see: http://www.w3.org/International/its/wiki/XLIFF_Mapping/,

http://www.w3.org/International/its/wiki/XLIFF_1.2_Mapping/, and http://www.w3.org/International/its/wiki/XLIFF_2.0_Mapping/.

- At the moment, the content stored in CMS databases as XML is often converted to XLIFF for localization purposes using proprietary tools which control the segmentation of the content into translatable items (translation units). These tools do not use ITS metadata for the purpose of fragmentation. As a result, the ITS metadata embedded in the source XML files do not control the distinction between translatable and untranslatable content, for example. Moreover, many ITS metadata may be lost during conversion to XLIFF.

3. The Solution

The solution we propose is to separate data processing (localization) from previewing context information (visualization of metadata), while still providing some synchronization between the production and preview environments.

The solution developed by Logrus and LinguaTech, known as Work In Context System or simply WICS, implies generating a reference file (viewable source) that is provided to the translator/editor in addition to the pieces of source text to be translated (translatable source) using a CAT tool. The reference file is standard HTML5 containing the same ITS 2.0 metadata as the source file, and specialized CSS/JavaScript code is used to display the ITS metadata. This preview does not require additional proprietary or specialized software – any supported web browser is sufficient, making the solution very portable (see Fig. 1):

To support the previewing of XML and XLIFF files, text conversion utilities were developed to transform these formats into an HTML5 preview format with equivalent ITS 2.0 metadata. Mapping ITS metadata from XML to HTML format turned out to be non-trivial task. (See the documentation published at https://github.com/renatb/ITS2.0-WICS-converter/ for more details on the format conversion related issues.)

The conversion of source HTML files enriched with ITS 2.0 markup to preview-ready HTML files has been introduced for technical reasons to ensure that all ITS rules are gathered inside the document and no external rule files exist. It simplifies parsing rules and implementation of rule priority and inheritance logic.

When the viewable source is loaded in a browser, ITS 2.0 information is highlighted, color-coded and augmented with popups. Additional information contained in the ITS metadata is also shown, such as definitions, comments, instructions, parts of speech, semantic information, reference web sites (both extranet and intranet), reviewer's comments, etc. Actual translation or editing might be carried out in another format in a CAT tool, but a parallel preview is certain to improve the view of the context for translators, editors, reviewers and other text workers in a wide variety of scenarios, including content authoring, translation, MT post-editing, knowledge transfer, etc.

See Figure 2 for an example of content enriched with ITS 2.0 localization metadata and rendered in a web browser. The content and localization metadata

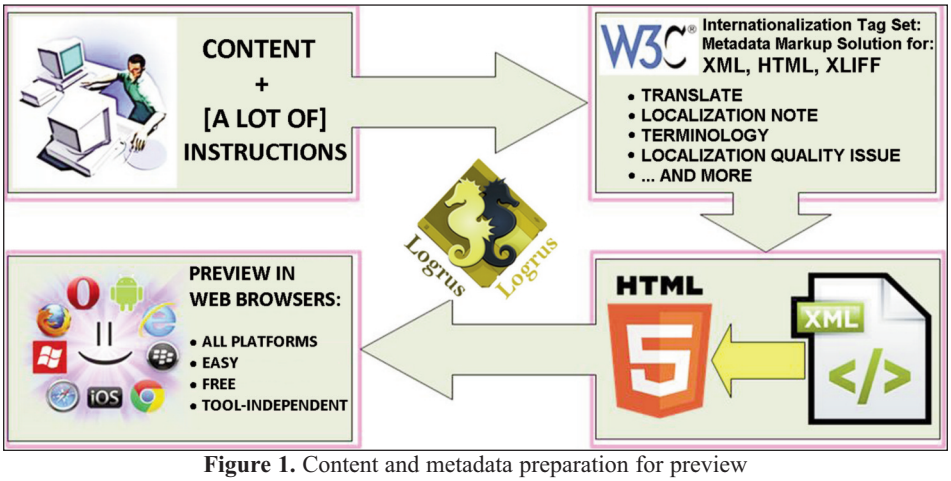

Figure 1. Content and metadata preparation for preview

preview and navigation functionality is provided via JavaScript. There are two preview modes: brief and extended. In brief preview mode, the pieces of content linked to metadata are highlighted, but the metadata are not displayed. In extended preview mode, the metadata linked to any particular piece of content or several metadata items within the active fragment of content are displayed in a separate metadata preview panel in the browser window. This solution supports metadata nesting: you can assign some metadata to the inner part of a phrase even when there is already existing metadata assigned to that phrase.

- Ready to use JavaScript files and other auxiliary files automatically referenced by web browser when opening the preview-ready HTML files.
- The complete set of source code packages, auxiliary files and instructions on building all the project utilities.
- The extended project report for the end-users and solution developers.

The project deliverables have been published at GitHub as an Open Source project available under MIT license. See the references at the end of the paper.

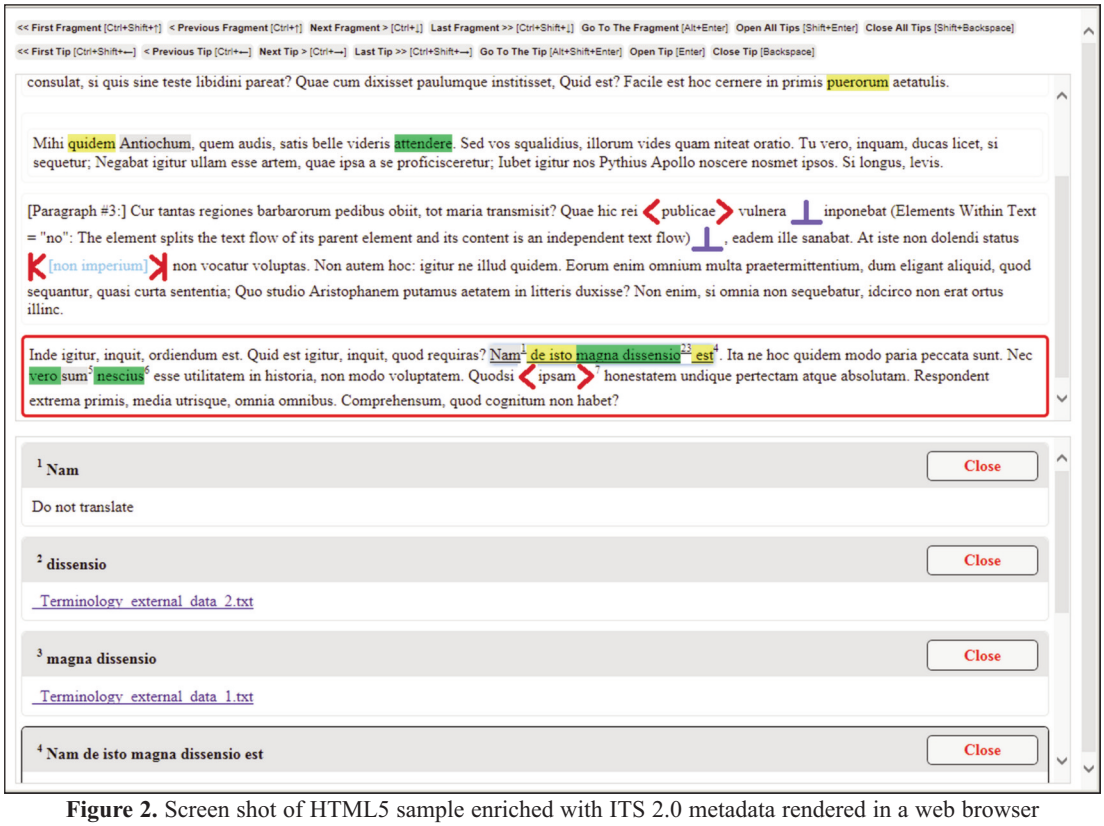

Figure 2. Screen shot of HTML5 sample enriched with ITS 2.0 metadata rendered in a web browser

4. Project Deliverables

The project deliverables include the following:

- Ready to use executables (CLI and GUI) of data converters used to transform XML, HTML or XLIFF files enriched with ITS 2.0 markup into preview-ready HTML files enriched with equivalent ITS 2.0 metadata.

5. Conclusions and Future Work

The main questions motivating this project were: how to preview ITS 2.0 metadata and how to use these metadata in real-life localization processes. This project sought to provide a portable ITS previewing solution, and to research the challenges associated with such previewing. With the tools we developed, localization instructions or other information can be shared effectively and

consistently with text workers via viewable versions of any files provided for reference including full versions of source files or any reference files enriched with ITS metadata, regardless of preferred platform or CAT tool.

With the development of the ITS 2.0 specification, the localization industry gained a carrier of localization metadata for major formats of content: XML, HTML, and XLIFF. One of the tasks for future work could be parsing and automatic or semi-automatic mapping of relevant rules from any stand-alone localization instructions to the relevant pieces of content via ITS markup. In an ideal case, such natural language processing solution should be able to apply ITS markup to the content according to any external localization instructions represented in some machine-readable format.

Acknowledgements

This work has been supported by Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI) and carried out as a part of one of the tasks of the W3C LT-Web project (http://www.w3.org/International/multilingualweb/lt/), funded by the European Commission through the Seventh Framework Programme (FP7) under contract No. 287815.

References

Logrus International (2013) The Project Deliverables: ITS 2.0 Visualization Toolset Including Samples of Localization Metadata at www.github.com [online], available: https://github.com/renatb/ITS2.0-WICS-converter, https://github.com/renatb/ITS2.0-WICS-viewer [accessed 20 November 2013].

W3C (2013) 'Internationalization Tag Set (ITS) Version 2.0' [online], available: http://www.w3.org/TR/its20/ [accessed 2 October 2013].

W3C (2008) 'Extensible Markup Language (XML) 1.0 (Fifth Edition)' [online], available: http://www.w3.org/TR/xml/ [accessed 3 June 2013].
W3C (2013) 'HTML5' [online], available: http://www.w3.org/TR/html5/ [accessed 3 June 2013].

OASIS (2008) 'XLIFF Version 1.2' [online], available: http://docs.oasis-open.org/xliff/xliff-core/xliff-core.html [accessed 3 June 2013].

BONUS ARTICLE

The Intricacies of Translation Memory Tools:
With Particular Reference to Arabic-English Translation

The following bonus article is not standards specific, and as such did not fall under the remit of the Guest Editorial Board for this issue of Localisation Focus.

The article was peer reviewed under the standard blind review process utilised for this journal by the regular Editorial Board of Localisation Focus - The International Journal of Localisation. Members of the editorial board are listed below.

Enjoy the article.

The Intricacies of Translation Memory Tools: With Particular Reference to Arabic-English Translation

Mohammad Ahmad Thawabteh
English Department, Al-Quds University,
PO Box 20002, Abu Dies, Jerusalem
Occupied Palestinian Territories
mthawabteh@arts.alquds.edu

Abstract

Translation Memory (TM) technology has been enjoying a good deal of popularity among translation theorists and practitioners since it came onto the market in the 1990s. A theoretical framework for TM *vis-à-vis* Machine Translation (MT) is first discussed. The paper then examines the applicability of a TM tool, namely Translator's Workbench (TWB), to Arabic, and the ensuing problems as illustrated by the translation output of ten postgraduate translation students at Al-Quds University for the academic year 2012/2013. The paper reveals that beyond the translation problems with which translation is usually replete, particularly between languages of little cultural and linguistic affinity, e.g., Arabic and English, the students encounter several problems arising from the inherent structure of TWB. The study concludes by assessing some of the pedagogical implications of these difficulties, in a way that will hopefully help Computer-Assisted Translation (CAT) trainers deal with the problems in future.

Keywords: *Translation memories; machine translation; computer-assisted translation; technical problems; Translator's Workbench*

1. Introduction

In its essence, translation is an act of interlingual communication across languages and cultures. It includes the Source Language (SL), the language from which we translate, and the Target Language (TL), the language into which we translate. For the past few decades, translation has had echoes further afield in a panoply of disciplines such as film studies, semiotics, sociology, conflict studies, technology, narrative theory etc., thus viewed as eclectic in nature. Perhaps the most important innovation for translators today is the introduction of technology such as corpus-analysis tools, terminology managers and machine translation (MT) among many others. (For more details on the tools available to translators, see Esselink 2000; Austermühl 2001; Bowker 2002; Gil & Pym 2006 and Pym 2012).

Defined as "the process that utilises computer software to translate text from one natural language to another" (Systran 2004, as cited in Zughoul & Abu-Alshaar 2005: 1023), MT is the Translation Technology (TT) "with the most sway over the popular imagination" (Gil & Pym 2006: 16). Since its inception in the late 1940s, MT has given translation activity a new lease on life. But, no sooner had the translators counted their chickens in the use of MT, hoping that "the intelligent use of machine translation should mean that our best human efforts are focused where they are most needed" than they concluded that it is full of fiendish difficulties in view of the fact that the "technology is not perfect, and translators must be very aware of those imperfections" (Gil & Pym 2006: 18).

It ensues, therefore, that much effort should be exerted in a search for more developed tools that would assist in the translation process. The technology of Translation Memory (TM), which originated in the 1970s, came to the fore in the 1980s, "but only since the late 1990s has [it] developed into a significant commercial entity" (Melby 1995: 187 as cited in Bowker 2002: 92). Wallis (2008: 623) argues that TM computer programmes such as SDL Trados, Déjà vu, SDLX, Transit, etc. are "the most popular tools today […,] which contain an aligned database of previous translations that can be searched to find solutions for new translations." TMs[1] are "specifically designed to recycle previously created translations as much as possible" (Esselink 2000: 362), and are also considered "invaluable aids for the translation of any text that has a high degree of repeated terms and phrases, as is the case with user manuals, computer products and versions of the same document (website updates)" (Gil & Pym 2006: 8). TMs are labour-saving translation tools with a view to providing high translation quality, seeking

increased productivity, preserving the consistency of translation quality and expediting large amounts of information in a split second (see also Esselink 2000; Bowker 2002; Zughoul & Abu-Alshaar 2005; García 2006; Gil & Pym 2006; Elimam 2007). In brief, TMs are a family of Computer-Assisted Translation (CAT) tools (Austermühl 2001: 11), and have contributed to the welfare of translation in a job market in which translating must take place at a competitive price and with consistent terminology, "not to mention quality service, tight time frames, and so many other things the translators are learning to deliver along with their work [which] would be enough to justify, in the technical area, the use of translation memory and terminology management software" (Azzam 2004: 87-88).

It is noticeable that there is a dichotomy between MT and TM systems. Simard and Langlais (2001) claim that the constraints are much less stringent in the context of CAT than in MT. Whilst the former emphasises partiality (i.e., proposing partial translations to the translator) the latter focuses on entirety, i.e., covering the whole of the source text. Likewise, García (2009: 29-30) states that "it could categorically be said that MT was language-specific while TM was not; that MT came with sets of language specific-rules and vocabularies while TM came as a kind of empty receptacle into which translators poured sentences and terms." A distinction between TM and MT is made by Smith (2012, The difference between TM and MT). The former takes its point of departure from breaking down a source text into segments.

> A segment is a manageable bite sized chunks. As these source segments are translated, they are saved to the TM. At the same time segments are being saved for new translations, the TM is also being used to leverage previously translated content. When you move to a new segment for translation, the software checks in the TM if there is an identical or similar translation and automatically enters the result which is most appropriate into the new target. Any match with the TM is given a percentage score depending on how accurate it is.

The latter, however, highlights using a computer at the expense of a human translator in transferring a text from one language into another. Smith (2012, Machine translation) further explains: "Untrained MT does not provide you with a match percentage for each translated segment, so it relies on the translator or reviewer to judge how accurate the suggested translation is. The quality of translations can vary significantly, and sometimes the results provided by machine translation can be quite amusing."

2. SDL Trados[2]

It is perhaps true that one of the omnipresent leading technologies in translation industry is SDL Trados (with its different versions) which is now synonymous with the concept of a TM environment (Hutchins 1998). More than twenty years ago, Trados began as "a language service provider, and only later, from 1989 onwards, did it special[ise] in software development –with the first product in the Trados stable, MultiTerm, hitting the market in 1990" (García 2005: 19).

It appears reasonable to assume that, other things being equal, SDL Trados has given the translation profession technological impetus. The stereotypical image of a translator as an "overworked, slightly grey woman or balding man nailed to a desk under a heap of dictionaries and encyclopaedias, leading a rather solitary life" (Vintar 2008: 40; see also Austermühl 2001: 11) is beginning to fade away. Vintar (ibid) further argues that "a more realistic picture of a translator at work would inevitably feature a computer with an internet browser minimised on the task bar and the heap of dictionaries similarly replaced by an array of desktop icons" (see also García 2006: 89). SDL Trados is a case in point. It "can be used to translate any kind of document that can be opened by Microsoft Word. TWB generates a statistical overview of the number of the internal repetitions, and fuzzy or exact matches in the translation memory" (Esselink 2000: 368). Exact match refers to the process in which the TM programme "pairs text segments in a revised source text that match the original source text exactly; however, any text in the document that does not exactly match the original will not be translated" (Webb 1998: 9). On the other hand, fuzzy match is the process by which the TM programme "pairs text segments in a revised source text with similar text segments from a previously stored translation based on the original source text. Fuzzy matching will find segments that are very similar to the original and

suggest the original translation" (ibid).

3. Research on the Technology of TMs

Technology has gained momentum and weight in different translation activities. Since there are many potential problems in the use of technology, TM-oriented research should then be carried out to keep abreast of the difficulties the translator is likely to face in translating from one language into another, and to work out suitable solutions. It is perhaps true that research on TMs in relation to translation is embryonic. Translators have only recently begun using TM tools on a wide scale, so "there has not yet been a substantial amount of research into the impact that they have on translators or their work" (Wallis 2008: 623). This explains the very few works published treating the subject in scholarly translation journals or books. A search in BITRA[3] (a prestigious bibliography of interpreting and translation studies) returns only 94 entries on TMs, with the abbreviation 'TMs' in the title, and no article on TMs with Arabic as an object of study. A similar search in Translation Studies Bibliography[4] returns only 20 hits on TMs, and of these nothing with the word 'Arabic' in the title. Research on TMs seems to be nothing to write home about.

4. TMs and the Arabic Translator

TT seems to be esoteric in the Arabic-speaking World, and only recently has it begun to fight for the recognition of its own place within Arab translation studies. It is also safe to argue that even MT is at an early stage in the Arabic-speaking World. For more details on the MT-related studies, see Homeidan 1998; Zantout and Guessoum 2000; Gaber 2002; Guidère 2002; Zughoul and Abu-Alshaar 2005; Diab, Ghoneim *et al*. 2007; and Hammadah 2008. The Pan-Arab Translation Centre in Beirut, as Raddawi and Al-Assadi (2005: 66) state, "does not have a record for any machine translation program[me]s or applications available in the Arab countries." By the same token, few attempts to address TMs are made in the Arabic-speaking World (see Elimam 2007, Fatani 2009 and Thawabteh 2009).

The principles of the process of MT and TMs are quite different, but they have grown together in the last few years. Compared to MT, TM is a relatively new technology whose presentation is likely to befuddle its users in doing translation tasks. However, no sooner has a user-unfriendly system come out than it becomes user-friendly with the passage of time and with the proper training. The use of SDL Trados is no exception. The better versed the translator is in the technology of SDL Trados, the more s/he seems to stand in awe of it. For instance, introducing TT into translator training at Al-Quds University usually stirs up unnecessary panic among the students, but it eventually turns out to be a blessing in disguise.

The present paper argues that the stereotypical image of the translator described by Vintar (2008), still pervades the Arabic-speaking World. Apparently, translation in its old sense is the be-all and end-all to many Arabic translators. Such images can be understood in terms of an underdeveloped translation industry[5] and university translation programme curricula that are mostly linguistic-oriented. In Saudi Arabia, Egypt, Jordan and the Occupied Palestinian Territories, to mention only a few, TT receives scant attention at both industrial and academic levels. In the translation industry in Saudi Arabia, "[n]o translation software is used, and in many cases translators are still searching for terms in a dictionary instead of having online access to a term bank" (Fatini 2009, Conclusion and major implications). Fatani concludes that out of 40 companies surveyed, none "were contemplating teaming up with a global translation supplier such as Trados since they were satisfied with outsourcing their work" (2009, Common practices). Taking Aramco as a case in point, Fatani (2009, Aramco) notes that MT contributes to the reinforcement of translation quality:

> The changeover to MT did indeed increase the speed, consistency and overall quality of translation. Despite the laying off of employees, Aramco translators report a high job satisfaction since the Trados system succeeded in eliminating all the tedious and repetitive aspects of translation. When probed, informants exhibited no aversion to MT, nor did they believe that computers had taken over their jobs. […] The presence of a large multinational staff made it imperative for the company to search for a translation solution that would facilitate communication among company employees, cut down on costs and speed up the

translation process.

In Arabic third level educational institutions, TT is as yet not a recognised field of study. Hammadah claims that "although TMs are precise, they are a neglected area of study in the Arab World" (2008, MTs in International World Market; researcher's translation). Gaber (2002, Prerequisites for translation instructors) states how little used TT in Egypt is, and further stresses that "translation teachers should be acquainted with the latest developments in information technology and electronic tools for translators." Similarly, the Occupied Palestinian Territories, are no exception as 'technologising' translation goes slowly. Many translation instructors at Palestinian universities have never had any exposure to technical software applications such as Déjà vu, Wordfast, SDL Trados etc. However, four graduate-level CAT courses are taught as part of the curriculum at Al-Quds University (see Thawabteh 2009: 166). In Jordan, a new CAT course is housed in the Department of Translation[6] at Yarmouk University.

There is still one caveat about introducing TT academically. The academic and industrial worlds diverge. Thus, translating in its traditional sense as envisaged by Vintar (2008) and Fatini (2009) is still shaping the overall translation industry in the Occupied Palestinian Territories (see Thawabteh 2009) and probably many (if not all) Arab countries. In a study conducted by Li (2002: 521), "nearly two-thirds of the respondents thought that [translation programmes] did not reflect the market very well." This might be true in the Occupied Palestinian Territories, especially at Al-Quds University. Even with such training savvy, only a few postgraduate students with an MA in Translation from Al-Quds University use TMs in their translation activities, and many jettison them. They have come full circle and 'traditional' translation methods are once again employed in translation tasks. This gloomy picture should not, however, be an obstacle in the way of using technology, which has become a determining factor in today's translation world.

In a nutshell, the Arabic translators appear to dislike the use of technology in connection with translation. Arguably, a lack of technical knowledge may be one reason. Another reason is that technology suffers unpredictable and annoying behaviours—manipulating PDF formats, scanning texts, or dealing with peculiarities in the encoding. Furthermore, the Arabic-speaking World lies among the low-rate-low-cost countries, which means that the Arabic translators are scraping by, and investing in relatively expensive TM systems is not feasible.

5. Methodology

5.1 Design of the study

This paper aims to investigate the problems encountered by ten MA Arab translation students using TWB. The data are derived from an Arabic-English task at the Comprehensive Examination in the first semester of the scholastic year 2012/2013. The task involved translating a highly repetitive text, designed for the purpose of the study (see Appendix 1), from Arabic into English. A carefully designed exam consisting of 69 words was used to examine the difficulties the students are likely to encounter in the course of using TWB. To ensure maximum reliability and validity, an Arabic professor checked the exam before the students sat it. The criterion for choosing the subjects was their prior experience.

The MA translation programme at Al-Quds University offers a combination of core and elective courses[7] amounting to 39 credit hours, with two options: a thesis option and a comprehensive examination option. Therefore, the students had received considerable training for at least two years in special TT courses, which aim at furnishing students with knowledge of electronic tools including some TM systems (e.g., Wordfast, Trados). For the sake of the present study, only TWB was used by the students, whereas other wide-ranging SDL Trados components e.g., WinAlign, TagEditor, T-Window for Clipboard, etc. are beyond the scope of the study. The figures of screenshots represent the students' actual translations. The examples are used to further explain the linguistic and/or technical difficulties the students were faced with in the translation exam.

6. Significance of the Study

Perhaps it would be safe to assume that TT seems to be of little interest in the Arabic-speaking World where linguistic-oriented approaches to translation are still seen as the academic norm. TT has only recently begun to gain significance as Thawabteh (2009: 165) points out: "TT has shifted somewhat towards lifelong training on account of the rapid expansion in market demand for qualified translators." Therefore, in view of a lack of interest in TT, and the dearth of basic and up-to-date Arab literature on TT, the present paper may be deemed significant because it addresses itself to the applicability of TWB to Arabic. Hopefully, this paper

will increase translators' awareness of the technology of TMs as a growing discipline in TS, offer an insight into the complexities of employing TWB in an Arabic-English context and delineate a path for further research in Arabic and other languages.

7. Discussion and Analysis

With the theoretical framework sketched, we now have an approximate idea about TM tools, particularly TWB which is superseded by something newer, thanks to the rapid pace of technology development. We shall examine some examples in order to corroborate and diversify our argument. To facilitate the analysis of the data collected in the experiment, a taxonomy of TWB-related problems was elaborated. It has been found that three major problems permeate the translations of the students, namely (1) linguistic problems, i.e., orthography and gemination; (2) discourse problems; and (3) human-computer interface.

7.1 Orthography

Orthography refers to the conventional spelling system used by a language to map phonology to or from the language script (Habash 2010). It is an oft-quoted truism that letter combinations that represent sounds in one language are different from those in another. This is quite true in (un)related languages e.g., Arabic and English. Whilst the former belongs to the Semitic language family, the latter is an Indo-European language. Orthographic disparity between Arabic and English may include capitalization, word breaks, emphasis, punctuation, graphemes and diacritics. These differences may bring about orthographic ambiguity, which, according to Habash and Sadat (2006: 2), may arise because the "form of certain letters in Arabic script allows suboptimal orthographic variants of the same word to coexist in the same text."

To see how this operates in practice in relation to students choices, let us indulge in a few illustrative examples:

Example 1

1a وانجبت القدس العديد من الكتاب والشعراء.

*wa **anjabat** al-Quds al-'adīd min al-kuttāb wash-shu'arā.*

('Several writers and poets were born in Jerusalem!')

1b وأنجبت القدس العديد من الكتاب والشعراء.

*wa **'anjabat** al-Quds al-'adīd min al-kuttāb wash-shu'arā.*

('Several writers and poets were born in Jerusalem!')

Example 1 shows orthographic variation between وانجبت *wa anjabat* ' begets' whereby the omission of *hamza* is noticeable and وأنجبت *wa 'anjabat* 'begets' in which the glottal stop (i.e., *hamza*) is observed. Figure 1 illustrates the point:

In Figure 1, the omission or writing of *hamza* in stem-initial position is clear, luckily with no semantic differences. This can result in translation errors. Nevertheless, the writing or omission of diacritics is important in Arabic and may have a deleterious effect on meaning. Observe the following example:

Example 2

2a ما أجمل القدس!

*mā **'ajmal** al-Quds!*

('How beautiful Jerusalem is!')

2b ما أجملَ القدس!

*mā **'ajmala** l-Quds!*

('How beautiful Jerusalem is!')

2c ما أجملُ القدس؟

أسوارها وحاراتها القديمة وقبة الصخرة وكنيسة القيامة.

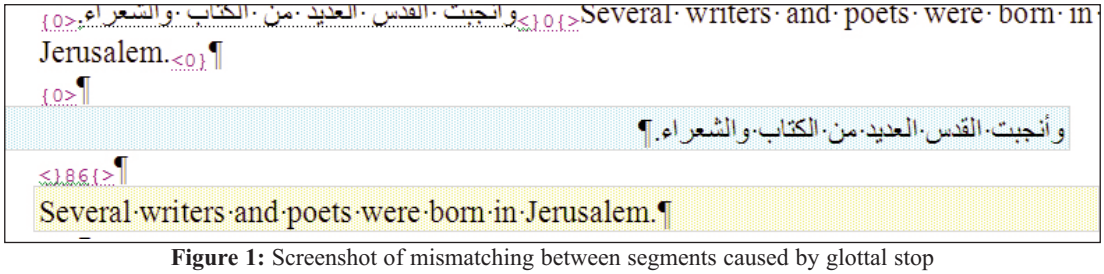

Figure 1: Screenshot of mismatching between segments caused by glottal stop

*mā **'ajmalu** l-Qudsi? 'aswāruha, wa ḥārātuha l-qadīmah, wa Qubbatu ṣ-Ṣaxrah wa Kanīsatu l-Qiyāmah.*

('How beautiful Jerusalem is? Its walls, old quarters, the Dome of the Rock and the Church of Sepulchre.')

It is necessary to account for the highlighted items in Example 2b and Example 2c. These are orthographically more or less the same, but syntactically different, thus bringing about different semantic meanings. In Example 2b, an exclamatory particle ما *mā* 'what' with the elative form of the adjective أجملُ *'ajmala* 'the most beautiful' is used with a diacritical mark *fatḥa* - َ [a] attached to the ending of the adjective to create an exclamation. In contrast, the diacritical mark *ḍamma* -ُ [ū] attached to أجملُ *'ajmalu* 'the most beautiful' in Example 2c in the subjective case is used to express a question (for more details on case in Arabic, see Aziz 1989: 128). Therefore, diacritics are notable features in Arabic. This kind of difference leads to a semantic gap, clearly observed in Example 2b and 2c.

On the other hand, Example 2a aims at examining the applicability of TWB to undiacritized text, a phenomenon that is typical of Arabic; diacritics are almost always absent in running text in written Arabic situations (Habash and Sadat 2006: 2). Reliance on our linguistic competence on the one hand and the context of a situation on the other may help us understand the acute differences in an exchange. The undiacritised utterance in Example 2a also poses a great challenge as it can either mean what Example 2b or Example 2c is intended to mean. Though semantically different, TWB, as Figure 2 shows, could orthographically recognise a high fuzzy match between segments in question: 2a and 2b (75% similarity) and 2b and 2c (84% similarity).

In Figure 2, the lexis أجمل in segment no. 2 has no full diacritics, thus rendering the word a homograph. Put simply, the segment ما أجمل القدس! has a multiplicity of meanings— either 'How beautiful Jerusalem is!' or 'What is the most beautiful place in Jerusalem?'. It is only the former that is intended in this situation. The bracketed number next to segment no. 2 shows zero matching as the source segment is sent to a built-in database, i.e., the TM did not contain this segment previously. For the subsequent segment (i.e., segment no. 3), the memory has proposed "How beautiful Jerusalem is!" as a translation, based on the translation of the previous segment, with a 75% match. The memory has suggested for segment no. 4 a similar translation to the previous one i.e., 'How beautiful Jerusalem is!', now with an exact match of 100%. Most importantly, the problem arises in segment no. 6, ما أجملُ القدس 'What is the most beautiful place in Jerusalem?' because TWB recognised an 84% match. In terms of meaning, segments no. 2, 3 and 4 are semantically different from segment no. 6.

The student translator seems to take a leap of faith and trust the TM system and/or is encouraged to work fast and uncritically with the translated segments, thus killing the spirit of the SL text. The TM is a false friend as the erroneous translation in segment no. 6 shows, for instance. We may also argue that the student decided to accept the 84% fuzzy match translation so one of the deficiencies of the TWB insofar as Arabic is concerned is its inability to handle diacritics on the one hand and student carelessness on the other. Webb (1998: 11) explains that although "fuzzy matching is quite useful, the user must also be aware of problems that may arise during post-editing of matched text segments". It is obvious that consistency in TMs is questionable (see also Moorkens 2012). For more elaboration, take Example 3:

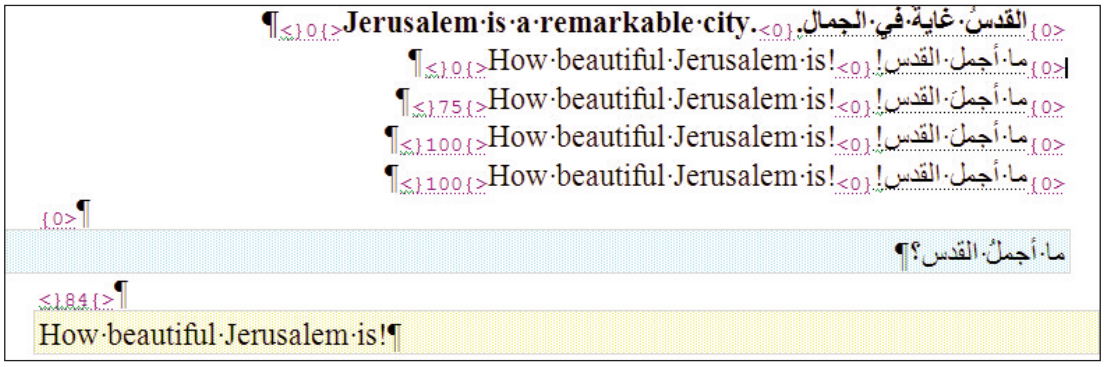

Figure 2: Screenshot of fuzzy match between segments

Example 3

3a أحبُ القدسَ وخاصةً أسوارَها.

'uḥibu l-Qudsa ***wa xāṣatan*** *'aswāraha.*

('I love Jerusalem, especially its walls')

3b أحبُ القدسَ خاصةً أسوارَها.

'uḥibu l-Qudsa ***xāṣatan*** *'aswāraha.*

('I love Jerusalem, especially its walls')

3c أحبُ القدسَ وبخاصةٍ أسوارُها.

'uḥibu al-Qudsa ***wabi-xāṣatin*** *'aswāruha.*

('I love Jerusalem, especially its walls')

3d أحبُ القدسَ وخصوصاً أسوارَها.

'uḥibu l-Qudsa ***wa-xuṣūṣan*** *'aswāraha.*

('I love Jerusalem, especially its walls')

In Example 3, segments 3a وخاصةً *wa xāṣatan* 'and especially', خاصةً *xāṣatan* 'especially', 3c وبخاصةٍ *wabi-xāṣatin* 'and especially' and 3d وخصوصاً *wa-xuṣūṣan* 'and especially' are synonymous and all have more or less the same meaning in Arabic, but with different orthographies. However, Figure 3 indicates a 70 percent fuzzy match for خاصةً *xāṣatan* (segment no. 2), an 80 percent matching for وبخاصةٍ *wabi-xāṣatin* (segment no. 3) and وخصوصًاً *wa-xuṣūṣan* for (segment no. 2).

as-Sakakini learned Arabic in Jerusalem schools.

4b ودرّس السكاكيني العربية في مدارس القدس.

wa ***darrasa*** *as-Sakakīniyy al-'arabiyyata fi madāris il-Quds.*

as-Sakakini taught Arabic in Jerusalem schools.

In Example 4, gemination is observed in the highlighted items in 4b by the reiteration of [*r*] resulting in a totally different meaning from that in 4a. However, as the software matches strings based on characters and sentence length, TWB does not recognise the acute differences between 4a and 4b, giving a 93% match as Figure 4 shows. The students seem to have been misled by a higher match percentage.

It should be noted that the absence of gemination in ودرسَ *wa darasa* 'and he learned' (segment no. 1) and presence of gemination in ودرّسَ *wa darrasa* (segment no. 2) are not treated appropriately by TWB. Actually, it reinforces a malapropism: "the mistaken use of a word in place of a similar-sounding one" (Concise Oxford English Dictionary 2004). Character-based indexing poses one of the pitfalls of TWB, which obviously affects the translation retrieval performance. As Figure 3 shows, it would be indeed bizarre for the translation students to accept the suggested translation without editing it. The translation in Example 3 is then fraught with peculiar

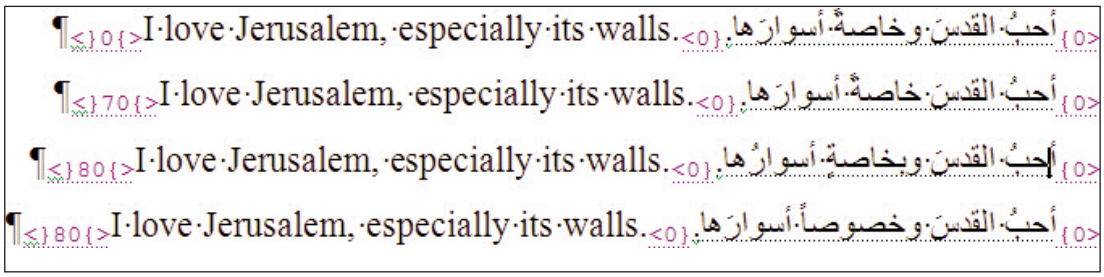

Figure 3: Screenshot of fuzzy match between synonymous items

7.2 Gemination

Gemination is orthographically "signalled in Arabic by a symbol called ***shadda*** above the sound in question […] Absence of such symbols leads to confusing the different parts of speech of words" (Al-Jabr 2008: 112, emphasis in original). Consider Example 4:

4a ودرسَ السكاكيني العربية في مدارس القدس.

wa ***darasa*** *as-Sakakīniyy al-'arabiyyata fi madāris il-Quds.*

perils. This is due to the fact that meaning is posited to be both the point of departure and end product of translation.

7.3 Discourse-related problems

Preserving meaning(s) expressed in an SL when translating into a TL is the ultimate goal of translation. A semiotic interaction of various signs within the boundaries of a text should be given due attention by the translator. Hatim and Mason (1997:

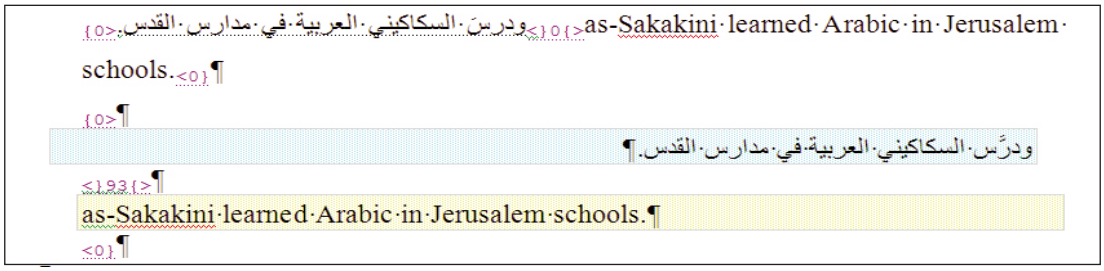

Figure 4: Screenshot of matching between geminated segments

223) point out that such an interaction paves the way for "a dimension of context which regulates the relationship of texts or parts of texts to each other as signs." Failure to abide by such a relationship gives rise to a breakdown in communication in the TL text. To illustrate problems in discourse, take Example 5 in which the coherence of the text is not well respected in the student's translation.

Example 5

5a	How beautiful Jerusalem is!
5b	Its walls, old quarters, the Dome of the Rock and the Church of Sepulchre.

It seems plausible to argue that 5b, as Example 5 shows, is a response to the Arabic question ما أجمل القدس؟ 'What are the most beautiful places in Jerusalem?' The syntactical and contextual information supplied by أجمل indicates to the translator the interrogative mood. As can be noted, segment 5b is recalcitrant to 5a, that is, does not flow communicatively, thus leading to a discourse-related problem (i.e., an incoherent translation). The suggested translation by TWB as Figure 5 suggests may cause a TL audience to raise eyebrows. One might understand the translation in Example 5, but still not intuit the underlying relations between different signs of the text. This boils down to human error on the part of the translation student, and may be related to training issues and the use of TM tools at the university.

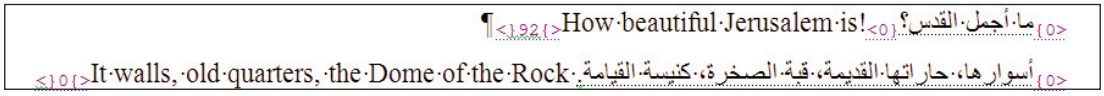

Figure 5: Screenshot of incoherent translation

7.4 The human-computer interface

The platform used by TWB is problematic for novice translators or even experienced ones on account of the user interface of TWB. Here we have Right-to-Left (RTL) SL text followed by Left-to-Right (LTR) TL text. As Esselink (2006: 25) puts it: "TM technology could only deal with text files. Hardly any technology [is] commercially available for the localization of software user interfaces." Dennett (2011: 29) further describes the user interface saying that TWB "not only has four windows of its own, but also adds a toolbar with a number of extra buttons to Word for Windows. This screen layout is a generic problem with all the program[me]s. The user interface is simply too cluttered for easy working." Dennett (ibid) further adds that TM programmes "are typically attempting to display four windows on screen at once: source language, target language, dictionary and fuzzy match." The window in Figure 2 is cluttered with several things: SL text in tandem with TL translation, bracketed numbers indicating match value, segmentation and alignment of the segments.

Alignment is an area of imbalance between Arabic and English by virtue of disparity in the writing systems. "Whereas the former is a [RTL] language, in which the letters of a single word can normally work with joined-up by 'ligatures' or cursive script, the latter is a [LTR] language" (Thawabteh 2007: 126; emphasis in original). TWB handles Arabic as a bi-directional language, which has special "reading order, visual appearance and alignment" (File Formats Reference Guide[8], Glossary-2). TWB "allows you to input content in any language into translation memory text and attribute fields. It is possible to input any Unicode character into a translation unit" (Trados TWB User Guide[9], 1-13).

Alignment poses a problem for Arabic- English translators using TMs, particularly Trados TWB. Figure 3, for instance, explains alignment complexities in which the text is full of clutters, with a likely problematic visual presentation of the SL and TL on the screen, and some of the translation problems may be attributed to issues with the human-

computer interface.

8. Conclusion and Pedagogical Implications

Technology has grown into an area of study worthy of research in its own right and provided the translator with several powerful communication tools, thus perhaps bringing about unparalleled prosperity in the translation industry. The image of the translator has changed from that of the past decades. Besides being an all-round person, the translator of today must be prepared to acquire technological skills which can support the translation process.

We should take cognisance of the fact that TT in the Arabic-speaking World should be streamlined. Attempts are being made to give TT a jump-start at some Arab universities; for instance, an initiative has been taken to teach TMs at the postgraduate level at Al-Quds University. It is no doubt an interesting initiative and, thus, sharing the experience of teaching and conducting this course would help other universities in Arab countries in outlining and updating their translation programmes.

TT is an under-researched area in Arab translation literature, but if Arab translation scholars began to address TT from a research point of view, this might pave the way for more development in the Arab translation industry. However, perhaps we should admit that the fact that TT is a lifeline to the job market in the Arabic-speaking World is questionable. It is therefore vital that translation programmes offering courses on CAT are responsive to industry demands. Perhaps it is futile to offer courses that are unrelated to the local job market as is the case with the Master's programme at Al-Quds University. However, the courses may be considered pioneering in that they qualify translation students to compete with peer translators all over the world. Equipped with sufficient technological savvy, students may enter the job market worldwide because, as Gil (2006: 90), explains: "customers and translators no longer need to be in the same geographical area, and members of the same translation team may live and work in different places."

We come to the conclusion that not all translation students are sufficiently equipped to employ TT in their future careers, and TT becomes a gruelling activity or a curse "based on a deep feeling of frustration in many translators […] due to the perceived steep learning curve needed to master TM" (García 2006: 98). The paper also reveals that the onus is on software developers to re-design TM tools to handle genetically remote languages against a backdrop of linguistically and culturally different systems. Example 4 is a case in point. The paper also shows that translating from morphologically-rich languages (e.g., Arabic) remains a challenging task.

Insofar as Arabic is concerned, the paper concludes that the use of TWB is associated with a number of complexities— problems with matching, recognition of spelling and diacritical variance and embedded morphological elements. Therefore, the issue of creditable performance of TMs is rather dubious, with respect to Arabic. It is perhaps true that TMs are buggy and unreliable. The translator should therefore aim for an acceptable compromise between usability and tractability. It is problematic to use TMs as "a translator will only see a few sentences, strings or one paragraph on the screen at a time during the translation process" (Elimam 2007, Parg. 6). Elimam further points out that the translator will only be able to work out of context. A corollary to this, the translator "may need to change some of his/her translations afterwards, which again means wasting some more time depending on how many corrections s/he needs to introduce in the translation" (Elimam 2007, Parg. 6).

It is safe to assume that MT is less efficient than TM tools. The former gives rise to many translation problems, especially in the translation of remote languages as is the case with Arabic and English. Unless it is meticulously used by the translators, MT may have disastrous consequences insofar as any translation activity is concerned. The latter, however, offer a gateway to success in translation profession if fastidious attention to technical details is paid.

Notes

[1] This abbreviation stands for translation memory tools.

[2] SDL Trados 2006 freelance is used by the sample of the study. For the sake of the present study, the sample worked with TWB as other software devices, e.g., SDL Studio 11/12 environment has not been implemented by the MA translation programme yet.

[3] https://aplicacionesua.cpd.ua.es/tra_int/usu/buscar.asp?idioma=en [accessed on August 31, 201]

[4] http://www.benjamins.com/online/tsb/ [accessed on August 31, 2012]

[5] Information on translation from and into Arabic is provided by Index Translationum: World Bibliography of Translation**,** available at: http://databases.unesco.org/xtrans/xtra-form.shtml, [accessed on September 13, 2012]

[6] {http://www.yu.edu.jo/index.php?option=com_docman&Itemid=332 [accessed on May 20, 2011]

[7] Core Courses (totalling 24 credit hours) are*:* Advanced Linguistics for Translators; Translation History and Theory; Editing, Documentation and Publishing Methods; Introduction to Interpreting; Audiovisual Translation I; Translation Practice I; Translation Practice II; and Translation Technology and Term Management. Electives (totalling 15 credit hours) are: Conference Interpreting I; Conference Interpreting II; Audiovisual Translation II; Literary Translation I; Literary Translation II; Translation Practice III (for three-language candidates); Technical and Business Translation I; Technical and Business Translation II; Legal Translation; Translation and Arabicization; Seminar in Translation and Thesis.

[8] Manual of SDL TRADOS7 Freelance.

[9] Manual of SDL TRADOS7 Freelance.

References

Al-Jaber, A. (2008) 'Impact of e-dictionaries on Arab students' translation strategies', *Babel,* 54(2), 110–124.

Austermühl, F. (2001) *Electronic Tools for Translators*, Manchester: St. Jerome.

Aziz, Y. (1989) *A contrastive Grammar of English and Arabic,* Mosul: University of Mosul.

Azzam, F. (2004) 'Gerenciamento de memórias de tradução e de glossários', *Cadernos de Tradução,* 2(14), 87-119.

Bowker, L. (2002) *Computer-Aided Translation Technology: A practical Introduction*. Ottawa: University of Ottawa Press.

Dennett, G. (2011) 'Translation memory: Concept, product, impact and prospects', available: http://www.tradulex.org/Bibliography/Dennett.pdf [accessed 2 April 2011].

Diab, M. Ghoneim, M. & Habash, N. (2007) 'Arabic diacritization in the context of statistical machine translation', Available" http://www.mt-archive.info/MTS-2007-Diab.pdf [accessed 20 October 2011].

Concise Oxford English Dictionary, (2004) 11th ed. Oxford: Oxford University Press.

Elimam, A. (2007) 'The impact of translation memory tools on the translation profession', *Translation Journal* [online], 11(1), available: http://accurapid.com/journal/39TM.htm [accessed 12 January 2011].

Esselink, B. (2000) *A practical Guide to Localisation,* Amsterdam: John Benjamins Publishing.

Esselink, B. (2006) 'The evolution of the localisation', in: Pym, A., Perekrestenko, A. & Starink, B., eds., *Translation Technology and its Teaching (with Much Mention of Localisation),* Tarragona: Intercultural Studies Group, 21-29.

Fatani, A. (2009) 'The state of the translation industry in Saudi Arabia', *Translation Journal* [online], 13(4), available: http://www.accurapid.com/journal/50saudi.htm [accessed 25 March 2011].

Gaber, M. (2002) 'A skeleton in the closet: Teaching translation in Egyptian national universities', *Translation Journal* [online], 6(1), available: http://www.accurapid.com/journal/19edu.htm [accessed 24 March 2011].

García, I. (2005) 'Long term memories: Trados and TM turn 20', *The Journal of Specialised Translation* [online], 4, 18-31, available: http://www.jostrans.org/issue04/art_garcia.pdf [accessed 24 March 2011].

García, I. (2006) 'Translators on translation memories: A blessing or a curse?', in Pym, A., Perekrestenko, A. & Starink, B., eds., *Translation Technology and its Teaching (with Much Mention of Localization).* Tarragona: Intercultural Studies Group, 79-105.

García, I. (2009), 'Research on translation tools', in

Pym, A., Perekrestenko, A. & Starink, B., eds., *Translation research projects 2*. Tarragona: Intercultural Studies Group, 27-33.

Gil, J. (2006) 'Teaching electronic tools for translators online', in Pym, A., Perekrestenko, A. & Starink, B., eds., *Translation Technology and its Teaching (with Much Mention of Localization).* Tarragona: Intercultural Studies Group, 89-97.

Gil, J. & Pym, A. (2006) 'Technology and translation. A pedagogical overview', in Pym, A., Perekrestenko, A. & Starink, B., eds. *Translation Technology and its Teaching (with Much Mention of Localization).* Tarragona: Intercultural Studies Group, 5-21.

Guidère, M. (2002) 'Toward corpus-based machine translation for Standard Arabic', *Translation Journal* [online], 6(1), available: http://accurapid.com/journal/19mt.htm [accessed 12 June 2011].

Habash, N. and Sadat, F. (2006) 'Arabic preprocessing schemes for statistical machine translation'. *Proceedings of the Human Language Technology Conference of the North American,* available: http://www.mt-archive.info/Coling-ACL-2006-Sadat.pdf [accessed 20 June 2011].

Habash, N. (2010) *Introduction to Arabic Natural Language Processing*, Morgan/Claypool Publisher series.

Hammadah, S. (2008) '*Dhākiratu at-Tarjama al-'Arabyyiah* A. M. T', available: http://www.atida.org/makal.php?id=146 (accessed 13 March 2011].

Hatim, B. & Mason, I. (1997) *The Translator as Communicator,* London and New York: Routledge.
Homeidan, A. (1998) 'Machine translation', *Language and Translation,* 10, 1-21.

Hutchins, J. (1998) 'The origins of the translator's workstation', *Machine Translation,* 13(4), 287-307.
Li, D. (2002) 'Translator training: What translation students have to say', *Meta,* 47(4), 513-531.

Moorkens, J. (2012) 'A mixed-methods study of consistency in translation memories', *Localisation Focus-The International Journal of Localisation* [online], 11(1), 14-26, available: http://www.localisation.ie/resources/locfocus/pdf.htm [accessed 12 January 2013].

Raddawi, R. & Al-Assadi, W. (2005) 'Machine translation in the Arab World: Overview and perspectives', *Translation Watch Quarterly,* 1, 59-81.
Pym, A. (2012) Translation skill, sets in a machine-translation age', available: http://usuaris.tinet.cat/apym/on-line/training/2012_competence_pym.pdf [accessed 12 December 2012].

Simard, M. & Langlais, P. (2001) 'Subsentential exploitation of translation memories', available: http://www.mt-archive.info/MTS-2001-Simard.pdf [accessed 10 October 2011].

Smith, K. (2012) 'The difference between translation memory and machine translation', available: http://www.sdl.com/community/blog/details/17449/the-difference-between-translation-memory-and-machine-translation [accessed 2 June 2013].

Thawabteh, M. (2007) *Translating Arabic cultural signs into English: A discourse perspective*, unpublished doctoral dissertation, University of Granada.

Thawabteh, M. (2009) 'Apropos translator training aggro: A case study of the Centre for Continuing Education', *The Journal of Specialised Translation* [online], 12, 166-176, available: http://www.jostrans.org/issue12/art_thawabteh.pdf [accessed 12 January 2011].

Vintar, Š. (2008) 'Corpora in translation: A Slovene perspective', available: http://www.jostrans.org/issue10/art_vintar.pdf [accessed 23 May 2011].
Wallis, J. (2008) 'Interactive translation vs. pre-translation in TMs: A pilot study', *Meta,* 53(3), 623-629.

Webb, L. (1998) 'Advantages and disadvantages of translation memory: A cost/benefit analysis', unpublished thesis (M.A.), California: Monterey Institute of International Studies Monterey.

Zughoul, M. & Abu-Alshaar, A. (2005) 'English/Arabic/English machine translation: A historical perspective'. *Meta,* 50(3), 1022-1041.

Zantout, R. & Guessoum, A. (2000) 'Arabic machine translation: A strategic choice for the Arab World', *Computer and Information Sciences,* 12,

117-144.

Acknowledgements

I warmly thank Dr. Omar Najjar, Al-Quds University and Mr. Tom Sperlinger, University of Bristol who commented on an earlier version of the paper. I would also like to thank MA translation students at Al-Quds University for cooperation and fruitful in-class discussions.

Appendix 1

Translate the following text by using Trados TWB.

القدسُ غايةً في الجمال.
ما أجملِ القدس!
ما أجملَ القدس!
ما أجملَ القدس!
ما أجملِ القدس!
ما أجملُ القدس؟
أسوارها وحاراتها القديمة وقبة الصخرة وكنيسة القيامة.
وانجبت القدس العديد من الكتاب والشعراء.
وأنجبت القدس العديد من الكتاب والشعراء.
انجبت القدس السكاكيني.
ودرسَ السكاكيني العربيةَ في مدارس القدس.
ودرّس السكاكيني العربية في مدارس القدس.
أحبُ القدسَ وخاصةً أسوارَها.
أحبُ القدس خاصةً أسوارَها.
أحبُ القدسَ وبخاصة أسوارُها.
أحبُ القدس وخصوصًاً أسوارَها.

Guidelines for Authors

Localisation Focus
The International Journal of Localisation
Deadline for submissions for VOL 13 Issue 1 is 31 July 2013

Localisation Focus -The International Journal of Localisation provides a forum for localisation professionals and researchers to discuss and present their localisation-related work, covering all aspects of this multi-disciplinary field, including software engineering and HCI, tools and technology development, cultural aspects, translation studies, human language technologies (including machine and machine assisted translation), project management, workflow and process automation, education and training, and details of new developments in the localisation industry.

Proposed contributions are peer-reviewed thereby ensuring a high standard of published material.

If you wish to submit an article to Localisation Focus - The international Journal of Localisation, please adhere to these guidelines:

- Citations and references should conform to the University of Limerick guide to the **Harvard Referencing Style**
- Articles should have a meaningful title
- Articles should have an abstract. The abstract should be a minimum of 120 words and be autonomous and self-explanatory, not requiring reference to the paper itself
- Articles should include keywords listed after the abstract
- Articles should be written in U.K. English. If English is not your native language, it is advisable to have your text checked by a native English speaker before submitting it
- Articles should be submitted in .doc or .rtf format, .pdf format is not acceptable
- Excel copies of all tables should be submitted
- Article text requires minimal formatting as all content will be formatted later using DTP software
- Headings should be clearly indicated and numbered as follows: 1. Heading 1 text, 2. Heading 2 text etc.
- Subheadings should be numbered using the decimal system (no more than three levels) as follows:

  Heading
  1.1 Subheading (first level)
  1.1.1 Subheading (second level)
  1.1.1.1 Subheading (third level)
- Images/graphics should be submitted in separate files (at least **300dpi**) and not embedded in the text document
- All images/graphics (including tables) should be annotated with a fully descriptive caption
- Captions should be numbered in the sequence they are intended to appear in the article e.g. Figure 1, Figure 2, etc. or Table 1, Table 2, etc.
- Endnotes should be used rather than footnotes.

More detailed guidelines are available on request by emailing LRC@ul.ie or visiting www.localisation.ie

Localisation Focus
The International Journal of Localisation

VOL. 12 Issue 1 (2013)

SPECIAL STANDARDS ISSUE

CONTENTS

Editorial
David Filip & Dave Lewis 3

Research articles:

Localisation Standards for Joomla!
Translator-Oriented Localisation of CMS-Based Websites
Jesús Torres del Rey, Emilio Rodríguez V. de Aldana …….....4

Interoperability Frankfurt-Madrid:
ITS 2.0 CMS/TMS use case
Pedro L. Díez Orzas, Karl Fritsche, Mauricio del Olmo, Stephan Walter 15

Generalizing ITS as an Interoperable Annotation Technique for Global Intelligent Content
Dave Lewis, Leroy Finn, Rob Brennan, Declan O'Sullivan, Alex O'Connor 27

ITS2.0 and Computer Assisted Translation Tools
Pablo Porto, Dave Lewis, Leroy Finn, Christian Saam, John Moran, Anuar Serikov, Alex O'Connor 40

Linport as a Standard for Interoperability Between Translation Systems
Alan K. Melby, Tyler A. Snow ..…….....50

ITS 2.0 Validation Techniques
Jirka Kosek 56

Process and Agent Classification Based Interoperability in the emerging XLIFF 2.0 standard
David Filip, Asanka Wasala.....61

Visualization of ITS 2.0 Metadata for Localization Process
Renat Bikmatov, Nathan Glenn, Serge Gladkoff, Alan Melby.....74

BONUS ARTICLE

The Intricacies of Translation Memory Tools:
With Particular Reference to Arabic-English Translation
Mohammad Ahmad Thawabteh.....79