Localisation Focus

THE INTERNATIONAL JOURNAL OF LOCALISATION

ISSN 1649-2358

Standards
Issue II

The peer-reviewed and indexed localisation journal

VOL. 14 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 visit 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

This Special Standards Issue of the Localisation Focus is based on papers and presentation that first appeared at the FEISGILTT events held in 2014. FEISGILTT 2014 (the 3rd FEISGILTT) took place as a Localization World preconference in Dublin Convention Centre, 3rd and 4th June 2014. The event comprised the 5th International XLIFF Symposium; a track called Content Analytics meets Localization and the Federated Track. There was also a North American one-day follow up event called FEISGILTT 2014 Vancouver Edition and. Two papers in this issue originate from the follow up event, albeit their topics had actually been first introduced and extensively discussed at the Dublin event in June 2014.

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 acronym).

This Search Engine Optimised acronym is supposed to be read as "fesh-gilt", where "feis" is an Irish Gaeilge word for a festival of music and dance, which seems just appropriate because, same as music and dance, localisation interoperability 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 discuss openly the pros and cons of solutions and approaches to standardisation and industry standards' implementations.

We are very thankful to Mr. Reinhard Schäler who invited Dave Lewis and I, as FESGILTT Conference Chairs, to become Guest Editors of Special Standards Issues of Localisation Focus in 2013 and 2014/2015 (the issue at hand).

All submissions made to the third FESGILTT and Vancouver Edition events received 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.

This issue brings four papers that are primarily concerned with the XLIFF standards, two of them with XLIFF 2. One paper by TCD researchers suggests how to use Semantic Mapping methods to address heterogeneity in XLIFF 1.2 implementations. Researchers from and Geneva and Salamanca explain how teaching about XLIFF 1.2 empowers translators in training at their universities. LRC researchers address Advanced Validation techniques for XLIFF 2.x. Chase Tingley produced a write up of his inspiring keynote that asked the all-important questions, whether XLIFF 2.0 had been an evolution (or revolution) developing from XLIFF 1.2 in the right direction and if it is about to meet with adoption success.

There is a fifth paper concerned with XLIFF, and XLIFF 2.0 in particular, that brings XLIFF 2.0 and its Glossary Module mapping into and back from the TBX Basic dialect. This collaboration between primarily Provo and Limerick (that later took on board other TBX stakeholders) was sparked in Dublin and led to development and presentation of a most intuitive 1-2-1 mapping between the two standards in Vancouver.

Two papers originate from the Content Analytics track, TCD researchers offer their takes on how NLP methods can be leveraged to provide better CAT tool functionality and why CAT tools need a technology agnostic method to collect translator activity data.

The FEISGILTT events that provided the base for this special collection of papers would not have been possible without sponsors, most importantly the Platinum Sponsor CNGL and two Gold Sponsors (the Falcon and LIDER projects). So here is the appropriate place to thank them.

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

Sincerely Yours, the Guest Editors

David Filip, University of Limerick & **Dave Lewis**, Trinity College Dublin

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

Teaching XLIFF to translators and localisers

Lucía Morado Vázquez[1], Jesús Torres del Rey[2]
[1] Département de Traitement Informatique Multilingue, Faculty of Translation and Interpreting
University of Geneva, Switzerland
[2]Department of Translation and Interpreting, Faculty of Translation and Documentation
University of Salamanca, Spain
Cod.eX Research Group
lucia.morado@unige.ch, jtorres@usal.es

Abstract
The XML Localisation Interchange File Format (XLIFF) is the main standard for the interchange of localisation data during the localisation process and the most popular and widely used in the industry. Computer Assisted Translation (CAT) tools already support its version 1.2. However, the most important end users of the format, i.e. translators, still have limited or no knowledge about the standard and the possible advantages of its adoption (Anastasiou 2010). With a view to bridging this knowledge gap, we have been introducing XLIFF as a topic of study in the translation and localisation studies curricula for the last four years in four different European universities, both at undergraduate and postgraduate levels, thus satisfying one of the missions of the Promotion and Liaison OASIS XLIFF subcommittee. In this paper, we aim at sharing our experience in teaching XLIFF to translation and localisation students: the curriculum design, the topics covered, the practical exercises and the areas that we have improved and modified based on our experience over this period of time.

Keywords: *Post-editing, Machine Translation, CAT, Linked Open Data*

1. Introduction

The XML Interchange File Format (XLIFF) is a tool-neutral standard that was conceived to allow for the interchange of localisable information during the localisation process. It was devised in Dublin in September 2000 by members of Novel, Oracle and Sun Microsystems. One year later, the first draft of XLIFF 1.0 was published; and in 2002 it was officially approved as an OASIS Committee Specification (Jewtushenko 2005). Since then, three more versions have been approved (1.1 in 2003, 1.2 in 2008 and 2.0 in 2014) and the standard has been widely adopted by the software and localisation industry, particularly over the past five years.

The advantages offered by the XLIFF format can be classified depending on the different agents involved in the localisation process: localisation customer, tool vendor and service provider (OASIS XLIFF 2007, pp.6-8). Translators would fit into the latter category; for them, XLIFF represents:

a) *a tool-independent file format* (OASIS XLIFF 2007, p.8): this could be the most important advantage for translators, as it gives them the freedom to choose their preferred CAT (Computer-Assisted Translation) tool, reinforced by the fact that XLIFF version 1.2 is a widely-supported format in the CAT tool ecosystem (Filip and Morado 2013);

b) *a standardised file format* (OASIS XLIFF 2007, p.8), which could help translators to concentrate on mastering and understanding the structure of one standardised, well-established format instead of several proprietary specific ones. This advantage was also addressed by García (2006, p.18) when he stated that the use of XLIFF for freelancers could mean their way back to working on the text, rather than worrying about formatting issues;

c) *a possibility to incorporate the standard file format in the vendor's workflow* (OASIS XLIFF 2007, p.8). Some CMS already allow users to extract the translatable text of their web sites in XLIFF format and reimport it to the system once the translation has been completed (Torres del Rey & Rodríguez V. de Aldana 2013);

d) *an open standard* (OASIS XLIFF 2007, p.8): the development process of XLIFF is completely transparent and all the documents produced by the OASIS XLIFF Technical Committee are available for public

consultation. Moreover, the composition of the Technical Committee itself –with members coming from software companies, tool vendors, service providers, associations and academia– (Filip 2012, p.33) guarantees that the needs of all agents implied in the localisation process are taken into account;

e) *the advantages of XML* (OASIS XLIFF 2007, p.8): being XML-based, XLIFF represents a format that can be easily handled and modified by translators. Most web browsers can display well-formed XML documents (such as an XLIFF file); moreover, XML files can be opened and modified without the need of specific advanced software: a simple text editor such as Notepad (in a Windows based system) can be used.

It is clear from the above-mentioned advantages that translators can benefit from the standard in numerous ways. In a localisation or translation process, translators are the last of a series of agents having to deal with the standard. However, they are still not very familiar with it (Anastasiou 2010, Morado Vázquez 2012, p.155). All these reasons have compelled us to create a teaching module on XLIFF to familiarise translation and localisation students with the standard and to make them better equipped for their professional practice.

Knowledge is power. Without knowing the benefits that the standard can provide them, they will not be able to make the most of it in the future professional careers and they will never be able to claim their rights to tool independency, active participation in the workflow, accessing whole, non-fragmentary information about the content and process included in the interchange documents, contributing to the possibilities of the standard, and so on. Moreover, it is also possible that the people in charge of distributing the files to be translated within a company (i.e. project managers) are not aware of the benefits and advantages that the use of XLIFF can imply.

This paper covers our experience in teaching XLIFF to translation and localisation students and is structured as follows: in section 2 we state the rationale that led us to this choice of content for translation and localisation trainees. In section 3, we include the initial considerations that were taken into account when designing the module, followed in Section 4 with our teaching methodology. Section 5 contains a detailed description of the latest iteration of our XLIFF module taught at the Autonomous University of Barcelona. We end up this paper with a summary of the lessons learnt during our teaching practice and the future work that we intend to implement on the subject.

2. Rationale

Our main objective as knowledge facilitators is to empower our students. We want them to be in control of the process and resources that they manipulate in order to carry out a job in a way that is satisfactory to them in professional terms and gives value to society. Being in control means, in a practical sense, that they should understand the files that they need to handle and the processes that they are involved in. Accordingly, in-depth knowledge should be provided to give them the necessary means to confront typical as well as unforeseen circumstances during real situations in their future professional career.

Knowing how to use and process different file formats has been identified as one of the main localisation elements that should be included in the curriculum for translators (O'Hagan 2006, p.41). In addition to this, the first author of this paper has been involved in the OASIS XLIFF Technical Committee and in the OASIS XLIFF Promotion and Liaison (P&L) Subcommittee for the last six years, the latter focusing on the promotion of the standard within the localisation field. Both authors have also been involved in the organisation of the yearly symposia on XLIFF coordinated by the OASIS XLIFF TC and the P&L Subcommittee since 2010. Therefore, it seemed logical to us as localisation lecturers to help to spread our knowledge about the standard among the new generation of translators. The XLIFF module was first implemented in the year 2010 and since then it has been taught in different European universities. The module has been adapted and has evolved, taking into account the profile of the trainees and the feedback received from previous experiences.

3. Initial considerations

The first iteration of the module was a request made to the first author of this article in the year 2010. While conducting her PhD research she was part of a research group within the Localisation Research Centre at the University of Limerick. At that time she was already a member of the XLIFF TC and the director of the centre asked her to prepare a seminar on XLIFF to the rest of their colleagues. The seminar, that took place in early 2010, was also attended by postgraduate students pursuing the MA in Global

Computing and Localisation. The module was divided into a theoretical component (history of XLIFF, XLIFF usage, advantages, CAT tool support, XLIFF validator) and a practical component (the creation of an XLIFF file to familiarise participants with its syntax and a file inspection questionnaire). The result of that experience was very positive and it encouraged her to adapt it in the future in localisation-related courses as part of various official curricula elsewhere.

Since that constructive experience, the XLIFF module has been included: in the curriculum of an undergraduate course on localisation at the University of Salamanca, Spain, coordinated by the second author of this article, and who is now responsible for its adaptation and teaching at that institution; as one of the modules of a postgraduate course on localisation and project management at the University of Geneva, Switzerland; as one of the modules of a postgraduate course on XML and multilingual documents at the University of Geneva, Switzerland; and as a standalone standards seminar, that belongs to one of the three taught units that form the MA in *Tradumàtica* (Translation Automation or Translation and Computers) at the Autonomous University of Barcelona, Spain. All the above-mentioned courses are Localisation-related courses taught at Translation faculties. It should be noted that the LRC, where the first iteration of the module took place, is based at the Computer Science Department and Information Systems of the Faculty of Science and Engineering at the University of Limerick. Including the XLIFF syllabus in a Localisation-related module (whether this is located at a Computer Science or a Translation department) fit perfectly well in the already existing curriculum.

We always make the design of our XLIFF module pivot around two main axes: the previous technical background of the students, and the level of specialised knowledge that they need to acquire. On the first axis, we gather information beforehand about the students' experience in technical aspects of text formats, mainly of mark-up languages. During the first iterations of the XLIFF module, we realised that most of the problems that students faced were not directly related to XLIFF itself but to their lack of knowledge on XML basic concepts. Therefore, we decided to tackle that constraint by adding extra tutorials and practical exercises on XML prior to the introduction of XLIFF.

The second axis is determined by the maturity of the students and the level of specialisation of their degree. We take these factors into account because students pursuing an undergraduate diploma in translation might not need or may not be prepared to assimilate in-depth technical concepts, while this might not be be the case for postgraduate students in Localisation or Translation Technology Master's degrees.

4. Teaching methodology

The general objective of our module is to help students obtain good conceptual and practical knowledge of the XLIFF standard and other related localisation standards. The more specific objectives are: learning XML basic concepts; understanding the importance of the use of localisation standards during the localisation process; learning about the history and development of standards of localisation; learning how to manipulate some basic aspects of the various interchange file formats in localisation; getting in-depth knowledge of the XLIFF standard: main elements, attributes and most important uses; learning about similar standards used in the Open Source environment (GETTEXT system and the manipulation of PO files).

In our module, we introduce theoretical components followed by practical activities aimed at mutually reinforcing the theoretical concepts and the technical skills required to understand and manipulate XLIFF files: in the end, the nature, the mechanics and, why not, the aesthetics of the standard need to be assimilated synergistically (Torres del Rey 2005a, pp.171-186). Mixing those components is currently successful practice in the design of translation technology-related courses (Doherty *et al*. 2012, Doherty and Moorkens 2013, O'Brien 2002, Starlander and Morado Vázquez 2013) and it has been suggested as a good strategy in the design of XML-related courses to translators (Suttleworth in Núñez Piñeiro 2006, p.64).

The module always takes place in a computer room where both the lecturer and the students have access to the necessary tools to fulfil their tasks. The rationale and aims of the XLIFF module are always indicated in the first lesson to attract students' attention and make them aware of the importance of the knowledge that they are going to acquire. This information has been appreciated by students in similar courses (Doherty and Moorkens 2013, p.130).

In general, we always try to adopt what we have labelled the ECOS (Spanish acronym standing for Communicative, Object-oriented, Social) Approach

for a comprehensive, humanistic, "multiscopic" (i.e. from different perspectives) learning experience (Torres del Rey *et al*. 2014; Torres del Rey 2003; 2005a, 171ff; 2005b) – i.e. by stressing technical, *object*-oriented aspects that students must comprehend and experience visually, "manually" and proactively for better assimilation; by understanding how the standard can *communicate* information, structure, functionality, meaning; and by helping the localiser, through both the communicative and the object-oriented aspect, gain a strong foothold in the multi-disciplinary *socio*-professional circle (and process) they work in, while promoting *social* initiatives like standardisation and open knowledge.

5. Structure of the module

The latest and most detailed iteration of our module was created in the form of a seminar taught at the Autonomous University of Barcelona. The seminar was part of the second unit (web, multimedia and videogame localisation) of the MA in *Tradumàtica*. It consisted of an 8-hour course distributed in two days; lessons (mixing theoretical and practical components) were divided in two-hour periods with a short break (15mts) in between. It took place in the third week of February 2014. In this section we describe the four sections of that module at both the theoretical and practical level: Introduction to XML, Standards of localisation, XLIFF, and Open Source Localisation Standards.

Introduction to XML

As mentioned before, some of the problems that our students encountered in previous years during the XLIFF module were related to their lack of knowledge of basic concepts of XML. To avoid that problem we decided to include an introductory course on XML prior to the main lesson on standards and XLIFF. XML had already been identified as a topic that deserves its place within the localisation curriculum (Drouin 2006, p.51), and the need for adding XML and other formatting and exchange mechanisms to the translation curriculum has been mentioned by localisation scholars (Wright in Núñez Piñeiro O. and Mullamaa, K. 2006, p.61; Drouin in Núñez Piñeiro 2006, p.66). We started our introductory course with an overview of XML, its history and its current use in the localisation process. We then introduced the syntax rules and tools needed to modify it and render it.

For the practical session, three exercises were created to introduce our students to XML:

1 Creation of an XML file. After the first session on XML, we asked our students to create their own XML language in order to define the rates of a translation company. In this exercise they had to put what they had learnt about the syntax of XML into practice. They were asked to use an XML editor for the first time.
2 Fixing XML syntax errors. Once the previous exercise was completed, we introduced a second activity where we distributed several XML files containing different syntax errors. Using the XML editor debug functionality trainees had to try to fix the corrupted XML files.
3 Creating a filter in a CAT tool to translate an XML file. In the third exercise on XML, we presented our students with a simulated translation case scenario, where an XML file contained the text of the user interface (UI) of a software application that they needed to localise. We handed over an XML file that contained some translatable text. Their task consisted of creating an *ad hoc* filter in a CAT tool (SDL Trados Studio 2011)[1] that extracted the translatable text and protected the rest of the code. In that particular exercise we gave our students an XML file that contained the UI text strings of Notepad++ (an open source advanced text editor). After creating the filter and translating the first section of that file, our students were required to export the semi-translated file and import it back into the code of the program. That last step gave them the possibility of testing the result of their translation directly on the semi-localised software application. Viewing the final result on screen helped them to understand the importance of their work and prevented them from being too focused on the code alone, where the lack of context could lead to the de-humanisation of their activity.

Standards in localisation

This topic helped our students contextualise XLIFF within the localisation standards ecosystem. On the theoretical side, we introduced the following topics: introduction to the concept of standard, standardisation organisations (W3C, OASIS and LISA), W3C ITS, new localisation standardisation initiatives, and the standards developed by the LISA OSCAR group. For the practical session, we prepared an exercise with a TMX file that students had to inspect and divide into two valid files using a

tool of their choice (an XML editor with the validation functionality, an advanced text editor, a CAT tool…).

XLIFF

This section represented the core of our module. We began with an introduction to XLIFF –what it is, what it is made for, the extraction-merging paradigm and the advantages of its use. Then we presented the history of the standard and its different versions through time. We also produced an overview of the development process of the standard within the OASIS XLIFF Technical Committee –how the TC works and the transparency of the development process itself. The current level of support of the standard in CAT tools was also analysed. The new XLIFF 2.0 version was introduced along with the change of paradigm (core and modules) that it proposes. Finally the main elements and attributes of the 1.2 version were presented and discussed.

For the practical session, we conducted a hands-on session to create an XLIFF file manually. This first guided exercise helped us to introduce, one by one, the main elements of an XLIFF 1.2 file (xliff, header, body, note, trans-unit, source, target and trans-unit) along with its main attributes. At the end of the activity, trainees had to validate the file that they had created, using XLIFF Checker[2] (a tool developed by Rodolfo Raya, former secretary of the XLIFF TC).

For the second exercise, students were given an XLIFF file. A quiz with questions about that specific file followed, where students had to answer questions such as "What is the XLIFF version? What is the data type of the original file? Is the skeleton embedded? What are the source and target languages?..." As can be noted, those questions referred to basic information that could be found in that particular XLIFF file. This activity helped them to better understand the format and to be prepared for future situations where they would be able to analyse and process the files that they received before starting to translate them.

In the third exercise, students received five corrupted XLIFF files that they had to fix using the tool of their choice. They had to find the bug, create a bug report with it, and fix the code to obtain a valid XLIFF file. This latter activity helped them to acquire a better understanding of the correct XLIFF syntax and to develop their own problem-solving techniques.

Interchange standards in Free and Open Source Localisation

In the last session of the module, we introduced our students to the Open Source (OS) Localisation field. In this area, the most used bi-text interchange file format is not XLIFF but Portable Object (PO) (Frimannsson and Hogan 2005, p.9). After a brief introduction to the field pf Free and Open Source Software (FOSS), we explained the mechanism of the GETTEXT system and PO files. The syntax of PO files was widely discussed and the tools needed to modify this format were also presented.

For the practical session, and in an attempt to turn the last session of our seminar into an entertaining activity, we used a modified version of the OS game (SuperTuxKart[3]). In that modified version, we had included some errors in the Spanish translation of the UI through the PO file (which contained the UI text strings). Students had to find the errors while playing the game and filled a bug report with them. After

Figure 1: STK user interface with the included errors (left). PO file in Virtaal containing the UI text strings (right)**.**

finding a minimum number of errors, students were asked to edit the PO file to fix the errors using an OS CAT tool (Virtaal was present in the computer lab facilities) or an advanced text editor. Students could actually see the changes on the running game itself after modifying the PO file, which gave them an overview of the process and the result of their work on screen. We firmly believe that in a localisation course it is important that students can get a final view of their work in order to have a clear panoramic view of the whole process, and to become physically and emotionally involved with the concepts and objects being studied so that they can be meaningfully assimilated by the students (Torres del Rey 2005b, pp.533-534; 2005a, pp.181-183 and 196-198). Although some tasks might be done without context, at the end of the procedure the context is recovered and the final product is localised.

This last lesson can deserve a module of its own and it might be taught in relation with XLIFF or independently as a standalone unit. We have decided to include it as part of the XLIFF module to show a similar de-facto standard that is widely used in the FOSS localisation area (Wolff 2011).

In a nutshell, our module was composed of five different topics, with XLIFF being the core one. Despite representing a highly technical module, where students were forced to work mainly with code and little context, we attracted their interest by trying to represent, when possible, quasi-complete localisation processes. In those cases, the result of their code manipulations was implemented in a final localisation product. This "whole-process" strategy allowed us to give a sense of "meaning and purpose" to the task that they carried out and to contextualise their work better, as well as to place localisers in a potentially better situation, socio-professionally speaking, when having to deal with other team members from different disciplines. The de-contextualisation of the process is an inherent characteristic of the localisation process, where several agents and tools are involved at several stages to obtain the final product (Pym 2013). In fact, the XLIFF paradigm (extracting the localisable content, placing it into a XLIFF file and merging it back into the original format upon the completion of the localisation process) is a de-contextualisation process in itself. In order to tackle this "human" problem, we design activities that present a whole process with a final localised product so that our trainees understand each of the tasks as separated pieces of a bigger puzzle, which is the localisation process.

Topic	Main contents	Exercises	Tools used
XML	XML overview and basic concepts	Creation of a XML file	Exchanger XML Editor
	XML and localisation	Fixing XML syntax errors	Exchanger XML Editor
	Filters for XML	Creation of a XML filter in a CAT tool	SDL Trados 2011 Notepad++
Localisation standards	Concept of standard Standardisation organisations New localisation standardisation initiatives LISA OSCAR standards W3C ITS	Division of a TMX file	The tool of their choice.
XLIFF	XLIFF history and development Advantages Extraction-merge paradigm XLIFF support in CAT tools	Creation of an XLIFF file	Notepad++ and XLIFF Checker.
	XLIFF 2.0	Inspection of an XLIFF file and quiz	Notepad++ and a CAT tool of their choice
	XLIFF 1.2 syntax	Fixing corrupted XLIFF files	Any tool of their choice
Standards in FOSS localisation	Introduction to Open Source Localisation The GETTEXT system PO format	Linguistic QA testing of a game and error fixing	Virtaal SuperTuxKart Notepad++

Table 1: Overview of the XLIFF module taught at the Autonomous University of Barcelona

Students' feedback

A questionnaire to obtain students' feedback was distributed to students after each of the seminars and courses that were part of the MA in *Tradumàtica*[4]. Eighteen students answered the questionnaire related to the XLIFF seminar. Trainees gave an average score of 9.3 over 10 in the general appreciation category. They also declared that the learning process was adequate 18/18, the seminar contents fulfilled their expectations 18/18, the objectives of the seminar were achieved (17/18 yes, 1/18 partially), the materials of the seminar were adequate 18/18, and that the difficulty level was adequate 18/18. In the last questions, students were asked if they thought that they could apply the acquired knowledge to their professional life, 17 students answered "yes" and one student "I don't know".

In the same feedback questionnaire, an open question was left for additional comments. Here, three of the students stated that the seminar contained too much information for such short period of time:

> Student1: *I think this seminar should be taught earlier and it should have more hours, as* ***we have seen a lot of topics in a short period of time***[5]*. Moreover, what we have learnt is basic to understand how to create filters or fix hidden files, it would be better to know how to do that earlier.*
>
> Student2: *Maybe* ***it was too much information in a short period of time****. However, the practical exercises were very interesting and they helped us to better understand how a standard like XLIFF works.*
>
> Student3: ***The seminar was too "compressed"****. We should have had some more lessons on the topic.*

The "overcondensed information" issue could be attributed to the organisation of the MA itself. The eight-hour seminar was carried out during two afternoons (Tuesday and Wednesday from 4pm to 8pm). Such a timetable probably did not leave time for a calm digestion of the concepts; neither did it allow students time to finish their exercise as homework before the following practical task. It would have been better to have the seminar distributed in a longer period of time, with a maximum of two-hour sessions per day, ideally with two sessions per week. Two of those students even required more teaching hours on the subject and one of them would have preferred the module to be taught earlier. On the other hand, it was clear from the feedback that the module was perceived as a positive asset to their learning process and they could see the knowledge acquired as a component that could be useful on their professional career.

6. Lessons learnt and future directions

During the last years we have learnt to adapt our teaching approaches to the different student backgrounds. We have also modified our course contents according to the analysis of the difficulties encountered by our students and their own feedback in the form of questionnaires. In this last section we present some lessons learnt and some directions for future work.

6.1 Lessons learnt

A combination of theoretical and practical components is useful

The combination of master lessons with practical exercises has proved to be a successful teaching strategy to transfer XLIFF knowledge. However, there are two factors that need to be taken into account when undertaking practical exercises and hands-on sessions: the number of students and the periodicity of the lessons. We have seen during our different iterations that there is a direct and positive relation between a small group of students and the fluency of practical lessons: there are fewer possibilities of interruption and more time to answer students' questions. Having a teaching assistant in place during those lessons has also proved to be of great help. The periodicity of the lessons should also be taken in consideration: having a seminar of 8 hours in only two days on the topic is totally feasible, but it risks becoming too condensed and it does not allow for a serene assimilation of the proposed concepts. Extending the teaching hours during a longer period of time would allow students to repeat the practical exercises at home and finish them if necessary. It would also give them additional time to go over the acquired theoretical notions and practise their technical skills.

A previous XML course is needed

An XLIFF module could be hard to undertake without the prior introduction of other key concepts, i.e. mark-up languages and XML in particular. Needless to say, teaching XLIFF with other related localisation standards (such as TMX or TBX) can be

beneficial for the general understanding of the field by our students. Connecting both concepts is actually a win-win strategy, because learning about XLIFF is learning about XML. The mark-up concepts that are acquired in our module can be transferred to other XML languages, and the tools that we use during our exercises could be used in the future by our students when dealing with other related formats (i.e. creation of ad-hoc filters for CAT tools, use of tools such as XML editors and validators, advanced text editors, etc.).

The module should be placed at the end of the course/semester

Following on from the same idea that previous technical knowledge is required to assimilate the course successfully, we believe that a module on XLIFF should be planned as part of a localisation-related course and it should be introduced once other, more basic, technical concepts and practical skills have been acquired by students. CAT tool knowledge, for example, is taken for granted in our module. In fact, one of the exercises proposed (creating an *ad hoc* filter for a specific XML format) can be categorised as an advanced use of a CAT tool. In that particular case, command of the basic functionalities of a CAT tool (creation of a project, addition of a TM…) was assumed.

It is essential to adapt the course pace and content to students' needs

Students' different technological backgrounds are in fact one of the main difficulties that a localisation facilitator encounters when designing and teaching a localisation course (Quirion 2003, p.550). Students following localisation courses have different backgrounds and interests and we normally find different levels of computer literacy within the same group of students. This reality forces us to adapt our expectations about outcomes to each year's students as well as to adapt the pace of the course to strike a balance between high-skilled students and lower-skilled ones. The module on XLIFF could be categorised as one of the most advanced modules that our localisation courses contain, and as seen in the previous paragraph it should be ideally placed at the end of the semester, giving students time to adapt to and be familiar with raw code situations, advanced localisation tools and text editors, as well as mark-up languages.

Contextualised practical exercises help to assimilate the concepts

As Pym (2013) points out "[i]n its very nature, the localization project requires a significant division of labor". It is easy, therefore, to be "de-contextualised" and task-centred. During our practical exercises we try to tackle this issue by proposing activities where the final product can be achieved. In that sense, our trainees benefit from the view of the overall process and they are not kept within isolated stages of the process.

6.2 Future directions

The introduction of XLIFF 2.0 as the core of our course depends on its adoption by the localisation industry.

The new version has been approved in August 2014 and it has already been implemented in some CAT tools and prototypes (Morado and Filip 2014). We have included an introduction to this new version in the last iteration of our module, and we have planned to have a more detailed one in its future editions. However, we would like to maintain our activities on the 1.2 version until the new version will be completely established and supported by the main stakeholders of the localisation industry.

Based on our teaching experiences during these years, we have also envisaged the creation of a manual that contains our lessons with theory and practical exercises. With a few exceptions, the existing documentation on XLIFF is scarce and it is either too technical (as the TC specifications) or too commercially oriented (mainly case and pilot studies presented at conferences and symposia). In fact, creating and adapting theoretical material for students was the main challenge that we had to face when designing the XLIFF module. The publication of such a manual could benefit not only students, but also localisation lecturers who wish to implement a standard module in their courses.

As mentioned, the widespread use of standards –and XLIFF in particular– gives translators the freedom to work with the tool of their choice, which we consider their main advantage towards translators and localisers. Therefore we firmly believe that knowledge of XLIFF, its importance and manipulation, should be one of the core competences of a localiser – or translator specialised in translation technologies. We would like to go even further and state that it should be a concept that all translators trainees should acquire before graduating and starting their professional career, as we believe that they will have to deal with this format sooner or later. Consequently, we consider that an XLIFF module

should be part of Localisation and Translation Technologies courses. We have presented in this paper our experience on teaching it and we hope it could inspire others to continue with this educating activity.

Acknowledgements: We would like to thank some of the lecturers that have collaborated in creating and teaching this module during the last years: Emilio Rodríguez Vázquez de Aldana (from the University of Salamanca) and Silvia Rodríguez Vázquez (University of Geneva and Salamanca).

References

Anastasiou, D. (2010) 'Survey on the Use of XLIFF in Localisation Industry and Academia', In *Proceedings of Language Resource and Language Technology Standards–state of the art, emerging needs, and future developments Workshop, 7th International Conference on Language Resources and Evaluation (LREC),* Malta, 50-53.

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2014) *XLIFF Version 2.0* [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-os.html [accessed 15 Dec 2014].

Doherty, S., Kenny, D. and Way, A. (2012) 'Taking statistical machine translation to the student translator', in *Proceedings of the Tenth Biennial Conference of the Association for Machine Translation in the Americas*. San Diego, California, USA.

Doherty, S. and Moorkens, J. (2013) 'Investigating the experience of translation technology labs: pedagogical implications', *The Journal of Specialised Translation* 19, 122–136.

Drouin, P. (2006) 'Training for localization (Replies to a questionnaire)', in Pym, A., Perekrestenko, A. and Starink, B., eds., *Translation technology and its teaching*. Tarragona, Spain, 49-54.

Filip, D. (2012) 'The localization standards ecosystem', *Multilingual computing and technology*, 23(3), 29–36.

Filip, D. and Morado Vázquez, L. (2013) 'XLIFF Support in CAT Tools', OASIS XLIFF Promotion and Liaison Subcommittee Report.

Frimannsson, A. and Hogan, J.M. (2005) 'Adopting standards based XML file formats in open source localisation', *Localisation Focus–The International Journal of Localisation* 4, 9–23.

Garcia, I. (2006) 'Formatting and the Translator: Why XLIFF Does Matter', *Localisation Focus–The International Journal of Localisation*, June 2006, 14–20.

Jewtushenko, T. (2004) 'An Introduction to XLIFF', in *IV International LRC Localisation Summer School 2004*.

Morado Vázquez, L. and Filip, D. (2014) 'XLIFF Version 2.0 Support in CAT Tools', OASIS XLIFF Promotion and Liaison Subcommittee Report.

Núñez Piñeiro, O. and Mullamaa, K. (2006) 'Summary of discussion on Is localization just technology?', in Pym, A., Perekrestenko, A. and Starink, B., eds., *Translation technology and its teaching*. Tarragona, Spain, 59-62.

Núñez Piñeiro, O. (2006) 'Summary of discussion on What is XML and how do we teach it?', in Pym, A., Perekrestenko, A. and Starink, B., eds., *Translation technology and its teaching*. Tarragona, Spain, 65-68.

O'Hagan, M. (2006) 'Training for localization (replies to a questionnaire)', in Pym, A., Perekrestenko, A. and Starink, B., eds., *Translation technology and its teaching*. Tarragona, Spain, 39-44.

O'Brien, S. (2002) 'Teaching post-editing: a proposal for course content', in *6th EAMT Workshop Teaching Machine Translation*, 99–106.

OASIS XLIFF TC (2007) 'XLIFF 1.2 White Paper'.

Quirion, J. (2003) 'La formation en localisation à l'université: pour quoi faire?', in *Meta Translators' Journal* 48, 546–558.

Pym, A. (2013) 'Localization, Training, and Instrumentalization', Universitat Rovira i Virgili, Tarragona, Spain.

Reid, J. (Ed.) (2002) *XLIFF Version 1.0* [online], OASIS Standard. ed, Standard, OASIS, available: http://www.oasis-open.org/committees/xliff/documents/contribution-xliff-20010530.htm [accessed 15 Dec 2014].

Savourel, Y., Reid, J. (Eds.) (2003) *XLIFF Version 1.1* [online], OASIS Standard. ed, Standard, OASIS, available: http://www.oasis-open.org/committees/xliff/documents/xliff-specification.htm [accessed 15 Dec 2014].

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/xliff-core/xliff-core.html [accessed 15 Dec 2014].

Starlander, M. and Morado Vázquez, L. (2013) 'Training translation students to evaluate CAT tools using Eagles: a case study', in *Proceedings of ASLIB, Translation and the Computer Conference 35*, London, UK.

Torres del Rey, J. (2003) 'Tecnología y enseñanza de la traducción: panorama investigador, enfoque humanístico', in Ortega Arjonilla, E, García Peinado, M.A., eds., Panorama actual de la investigación en *traducción e interpretación*, Granada: Atrio, 343-359.

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

Torres del Rey, J. (2005b) 'Nuevas orientaciones de la formación en tecnologías de la traducción: procesos, objetos y aplicaciones', in Vilvandre de Sousa, C. *et al.*, eds., *El español, lengua de cultura, lengua de traducción*, Granada: Atrio, 525-539.

Torres del Rey, J. and Rodríguez V. de Aldana, E. (2013) 'Localisation Standards for Joomla! Translator-Oriented Localisation of CMS-Based Websites', *Localisation Focus. The International Journal of Localisation*, 12 (1), 4-14.

Torres del Rey, J., Morado Vázquez, L., Rodríguez Vázquez, S., Rodríguez V. de Aldana, E. (2014) 'La formación de localizadores en los estudios de traducción: un enfoque comunicativo, objetual y social', *II Congreso internacional sobre investigación en didáctica de la traducción*, Barcelona, Spain, 8-9 July, unpublished.

Wolff, F. (2011) Effective Change Through Localisation, Translate.org.za.

Notes

[1] This exercise was inspired by a similar one conceived by Emilio Rodríguez V. de Aldana, Localisation lecturer at the University of Salamanca, Spain, and a fellow member of the Cod.eX research group.

[2] http://www.maxprograms.com/products/xliffchecker.html

[3] http://supertuxkart.sourceforge.net/

[4] Although this is MA is widely known as "MA in Tradumàtica", its complete and official name in Catalan is *Tradumàtica: Traducció i Localització.* http://pagines.uab.cat/mastertradumatica/

[5] Emphasis added on the three statements.

Leveraging NLP Technologies and Linked Open Data to Create Better CAT Tools

Chris Hokamp
CNGL Centre for Global Intelligent Content
Dublin City University, School of Computing
Dublin, Ireland
chokamp@computing.dcu.ie

Abstract

This paper presents a prototype of a Computer Aided Translation (CAT) interface integrated with an entity extraction system to create a dynamic linked terminology component. The entity extraction system tags terms in the source sentence, mapping them to translation candidates in the target language. A usage scenario for linked data within a CAT tool is evaluated by prototyping all components necessary to construct a real-time dynamic terminology. By making use of Natural Language Processing (NLP) technologies including entity linking (Mihalcea *et al.* 2007), and statistical models for extracting and disambiguating entities (Daiber *et al.* 2013), the tool can provide translators with rich feedback about potential target-language translations of entities in the source text.

Keywords: *Post-editing, Machine Translation, CAT, Linked Open Data*

1. Introduction

Linked Open Data (LOD) can potentially be utilized at many points in the localisation workflow. By augmenting the metadata for the source or target text in a pre- or post-processing phase, linked data can provide metadata which facilitates human translation and quality assessment. Where metadata can be added in a completely automatic way, the task of determining whether or not the data is useful in the context can be pushed to the translator, who can decide where and how to make use of the additional information. The feedback from translators can then be used to augment the knowledge base.

In the dynamic terminology component presented here, a LOD resource and a statistical entity linker are combined within a translator-in-the-loop system. Translator-in-the-loop means that the design of the system explicitly includes a human, who is finally responsible for selecting the correct translation. This setup can be contrasted with a fully automatic design, where the target sentence would automatically be augmented with terminology, either via a machine translation system, or via an automatic post-editing phase.

The terms **entity** and **surface form** as used in this paper are defined as follows: an **entity** is concept (usually noun-like), represented by a unique DBPedia URI (Lehmann *et al.* 2014). A **surface form** is the text that is used to link to that entity – in other words, it is a language-specific string used to describe the entity. In Wikipedia, surface forms appear as blue hyperlinks in text (this indicates that an editor has linked the text with another page in Wikipedia). The set of possible surface forms for a given entity can be created by aggregating all links to the entity across all of a language's Wikipedia version, resulting in a (typically large) set of possible ways to refer to the entity. Table 1 shows the top ten German surface forms for the DBPedia entity "Earth".

DBPedia URI: http://dbpedia.org/resource/Earth
Surface Forms
Erde
Erdoberfläche
Welt
Erdbahn
Erdkugel
Terra
Planet
Erdkörpers
irdischen
Terra

Table 1: The most frequent German surface forms for the DBPedia entity "Earth"

In our design, the linking system tagger detects entities in a source segment, and the LOD resource provides candidate translations in the target language. By leveraging Wikipedia's multilingual graph through the DBPedia datasets, the system can provide suggestions for many language pairs. The multilingual graph of entities is thus transformed into a *dynamic terminology database*.

The term *dynamic* in this context means that the set of suggestions for a term depend upon the context in which it is being used. Because the disambiguation is done with respect to the context, the possible target forms are ranked according to their likelihood with respect to the underlying entity. A central hypothesis of this work is that this dynamic re-ranking provides a major improvement over the standard glossary or terminology lookup, which can only look for string matches for a particular token or phrase, without regard to the particular sense of the term in context.

2. Related Work

The majority of work on linked data for translation has focused on creating standards for data exchange, and on connecting backend resources such as terminologies with LOD ontologies. However, integrating linked data into the localisation workflow is an active area of research, and several projects, notably the ongoing FALCON (Lewis 2014b) project, are evaluating potential usecases as part of their goal to develop standards for linked data in localisation workflows. A prototype web-based application which integrates metadata using the Internationalization Tag Set (ITS) (Filip *et al.* 2013) within XLIFF 1.2 (Savourel *et al.* 2008) documents is presented in Porto *et al.* (2013).

3. Component Design

The motivating hypothesis for the component design is that the most difficult part of translating terminology is selecting the correct surface form for an entity. In other words, determining which entity a source language string refers to is easier than determining the correct translation for an entity, because of the nuance involved in choosing the correct surface form in the target language (formal vs. colloquial, full name vs. abbreviation, etc...). Thus, the component does not attempt to automatically select the correct surface form. The target side component is pre-populated with the translation options (ranked by frequency), and the translator must select the best option from the candidates. This design is similar to a terminology lookup or translation memory UI component, presenting translators with options instead of automatically populating the target translation with a "best" hypothesis.

We make use of the DBPedia-Spotlight (Daiber *et al.* 2013) statistical backend to perform the entity extraction step. The multilingual links in Wikipedia allow a mapping between languages to be created, so that concepts in the source language can be connected with concepts in the target language (the language specific URIs for a concept point to the same unique DBPedia URI). After the source entities have been extracted, target language translation candidates are found by moving in the opposite direction, generating the possible set of surface forms from the entity. The target-side surface forms are ranked by occurrence count with respect to the entity. Figure 1 shows a simple schematic of the flow of data through the component.

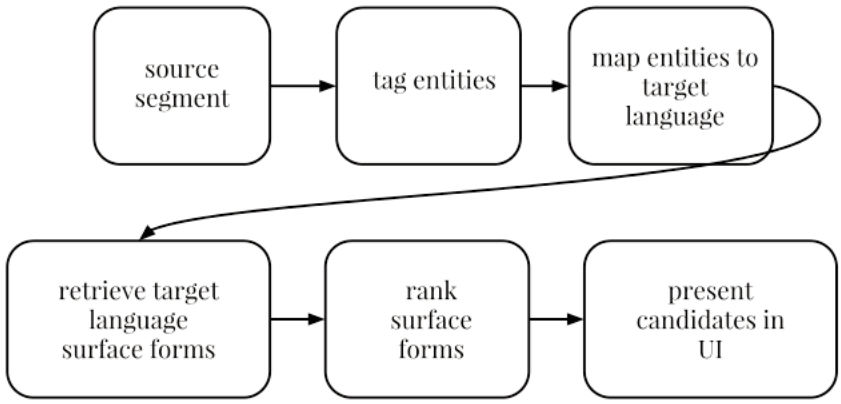


Figure 1: The dynamic linked terminology workflow

4. HandyCAT

HandyCAT is a flexible web based CAT tool (Lewis *et al.* 2014), specifically designed with interoperability and extensibility in mind. Because graphical components can easily be added and removed from the interface, it is an ideal platform for developing prototypes. The dynamic linked terminology component is designed as a standalone module that can easily added or removed from HandyCAT.

The server components are designed as microservices which are accessed using RESTful APIs, each fulfilling a single task in the dynamic terminology building process. The system is designed to operate in realtime, meaning that it does not require any offline preprocessing of the translation job.

5. Rendering Translation Options

Figure 2 shows a screenshot from an actual post-editing session. The user is evaluating the translation options for the source word "Europe". Upon selecting the best option (in this case the first option), the term will be inserted into the target area on the right side. All of the terms and markup are determined *automatically* and *on-the-fly* by the system, that is, not interfere with the normal operation of the interface.

6. Entity Linking and Labeling

Resources such as DBpedia and Freebase (Bollacker *et al.* 2008) are examples of open knowledge bases which take advantage of the implicit and explicit links in Wikipedia and other resources to construct a graph of entities with edges encoding relationships between the entities.

The process of finding the target-language surface form for a source entity requires two disambiguation steps. The first step is *entity linking*, where the entity extraction system attempts to link surface forms in the source language to the specific entity they represents. See Daiber *et al.* (2013) for details on the algorithm used to determine which entity is most likely represented given a surface form and a surrounding context.

The second step is *entity labeling*, where the translator selects the correct surface form for the entity in the target language. This requires retrieving the set of target language links for each entity, and making them available in the translation interface.

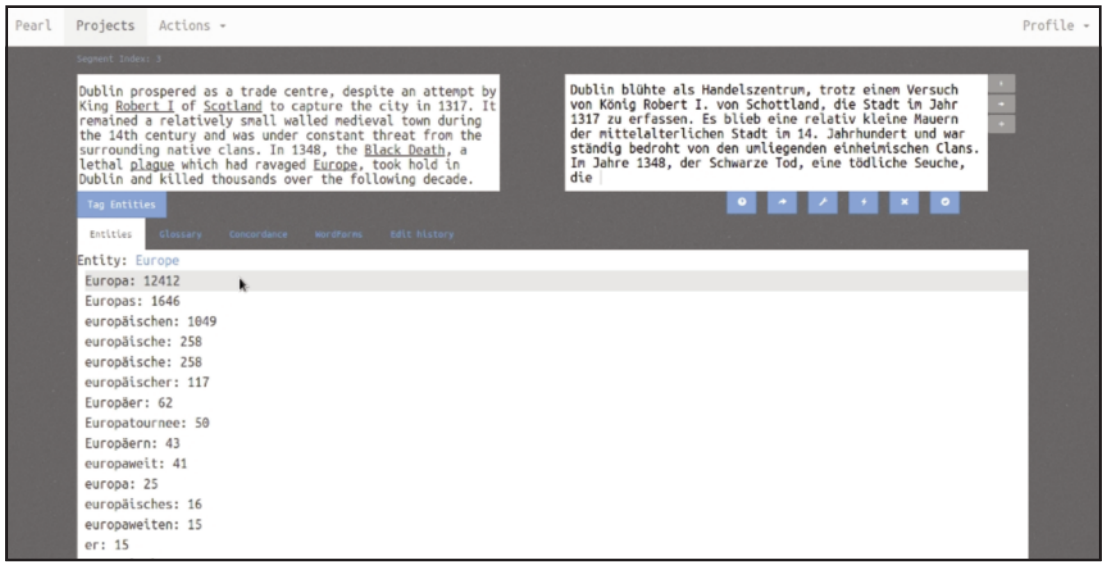


Figure 2: A screenshot of a HandyCAT editing session

there is no hard-coding of any entities or surface forms in either language, and the source text is parsed and linked when the translator enters the segment. The system can generate translation options for the entities in a source sentence in less than one second, so it does

6.1 Limitations

The terminology currently available to the component is limited to the data contained in Wikipedia, a resource that would probably not have good coverage

for many translation tasks, especially in specialized domains. Furthermore, the accuracy of the system is dependent upon the accuracy of the extraction framework. If a source entity is linked incorrectly, the translator could be presented with incorrect translation options.

Although the performance of the entity linking system is quite good, it is not perfect, so deploying the tool as part of a localisation workflow would require translators to carefully audit the translation options to ensure that they are not being presented with options that are generated from an incorrect entity. The entity linking component also requires many training examples for each entity in order to achieve good accuracy, so adding entities not contained in a large open dataset would necessitate curating a new training dataset – a process which could turn out to be prohibitively time-consuming.

7. Future Work

Evaluation of user interface components must be conducted with respect to a metric that can be measured in controlled user tests with and without the component present in the interface configuration. Some possible evaluation metrics are listed in table 2. Formal evaluation with one or more of these metrics has not yet been conducted, and the current prototype is simply a proof-of-concept.

Because the component is factored into standalone backend services (entity extraction and surface form mapping) and user interface elements, it can serve as simple enhancement to an existing interface. The backend services could also be integrated into a Machine Translation (MT) system, so that entities are added to the translation options considered by the MT system, instead of explicitly asking the user to choose the correct surface forms for the source entities.

8. Potential Integration with XLIFF and ITS

The tagging and disambiguation frameworks presented in this paper could be used as a standalone components in any localisation workflow. ITS and XLIFF are ideal for persisting translation options, and translators' choices for the best candidates in a particular context (Porto *et al.* 2013). One potential usecase could be used to add terminology to a project before it is sent to translators, allowing the information to be downstream in the translation process.

9. Conclusion

There are many opportunities to integrate existing NLP technologies into the Computer Aided Translation pipeline, but very few functional prototypes have been created to date. This work presented an end-to-end prototype of a dynamic linked terminology component implemented as part of the HandyCAT platform. The component was created to demonstrate a potential usecase for linked data within the localisation workflow, and to evaluate the effort needed to build such a system. This system enhances the resources available to translators without forcefully guiding the translation process, because translators are free to completely ignore the additional markup and terminology options if they wish. We believe that the human-in-the-loop paradigm is ideal for many CAT components, because it allows

CAT Component Evaluation Metrics
translator speed (words/min, segments/hour, etc…)
keypresses/operations per segment
quality of the resulting translation (human or automatic evaluation)
cognitive load (as measured by eye tracking or other methods)

Table 2: metrics for formally evaluating CAT UI components

translators to take advantage of additional metadata without requiring them to utilize the component(s) in cases where they do not perceive additional value.

Acknowledgments

This work was supported by the European Commission FP7 EXPERT project.

References

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J. (2008) 'Freebase: A collaboratively created graph database for structuring human knowledge', in *Proceedings of the 2008 ACM SIGMOD Interna-tional Conference on Management of Data,* New York, NY, USA. ACM, 1247–1250.

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2014) XLIFF Version 2.0 [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-os.html [accessed 22 Aug 2014].

Daiber, J., Jakob, M., Hokamp, C., Mendes, P. (2013) 'Improving efficiency and accuracy in multilingual entity extraction', in *Proceedings of the 9th International Conference on Semantic Systems,* New York, NY, USA. ACM, 121–124.

Filip, D., McCance, S., Lewis, D., Lieske, C., Lommel, A., Kosek, L., Sasaki, F., Savourel, Y. (Eds.) (2013) Internationalization Tag Set (ITS) Version 2.0 [online], W3C Recommendation, W3C, available: http://www.w3.org/TR/its20/ [accessed 16 May 2014].

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C. (2014) 'DBpedia – A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia', *Semantic Web Journal, 6(2), 167-195.*

Lewis, D., Liu, Q., Finn, L., Hokamp, C., Sasaki, F., Filip, D. (2014a). 'Open, Web-based Interna-tionalization and Localization Tools', *Translation Spaces*, vol III.

Lewis, D. (2014b) FALCON [online], available: http://falcon-project.eu/wp-content/uploads/2014/05/FALCON-Poster-mlw-madrid-may141.pdf [accessed 15 May 2014].

Mihalcea, R., Csomai, A. (2007). 'Wikify!: Linking Documents to Encyclopedic Knowledge', in P*roceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, New York, NY, USA. ACM, 233–242.

Porto, P., Lewis, D., Finn, L., Saam, C., Moran, J., Serikov, A., O'Connor, A. (2013). 'ITS2.0 and Computer Assisted Translation Tools'. *Localisation Focus - The International Journal of Localisation*, 12.

Savourel, Y., Reid, J., Jewtushenko, T., Raya, R. (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 15 May 2014].

XLIFF 2.0 and the Evolution of a Standard

Chase Tingley
Spartan Software, Inc
chase@spartansoftwareinc.com

Abstract

This paper is a write up of the FEISGILTT 2014 keynote on whether or not XLIFF 2.0 was a successful evolution of the XLIFF 1.x standard. This offers a higher level perspective on the chances that XLIFF 2.x is to become the common denominator of the localization industry. This is an honest and clear discussion of how XLIFF 2.0 has learnt from XLIFF 1.2 issues, which also suggests that the XLIFF 2.0 object model has potential to influence how localization data and metadata are exchanged at lower granularity through webservices, based not only on XML.

Keywords: *Interoperability, Standards, Feature Creep, Adoption, XLIFF*

"I find it rather puzzling," a technologist friend wrote to me an email, "that this small industry has such difficulties designing robust standards."

I agreed with his assessment.

As a consulting engineer in the localization industry, I spend a lot of my time developing solutions to help clients stitch together different pieces of technology into one coherent tapestry of automated process. In the course of this work, I spend a lot of time using — and complaining about — the various standards that define interactions across tools in our industry. In particular, I've spent a lot of time dealing with XLIFF 1.2 (Savourel et al; Eds.; 2008), a standard which I frequently utilize and almost as frequently find wanting.

XLIFF 1.2 is not a convincing standard, despite good intentions and flexible approach allows it to capture data for many different use cases. It suffers from a lack of interoperability between different implementations, characterized in particular by an inconsistent implementation of its feature set. Even mutually supported features are not always implemented in ways that are mutually intelligible. While the basic feature set can be relied upon, it is difficult to use the advanced features of XLIFF 1.2 across a tool chain containing more than than one or two different implementations.

With the recent finalization of the XLIFF 2.0 standard (Comerford et al; Eds.; 2014), it is worth taking another look at XLIFF 1.2, as well as the industry as a whole, and try to examine why producing a reliable interchange format for localization data is difficult. Using that analysis, we can look again at XLIFF 2.0 to see how it avoids pitfalls encountered in the past.

1. Difficulties in the Localization Ecosystem

1.1 Supply Chain Complexity

As an industry, localization is famously decentralized. Even a simple translation process may involve several parties, as a vast web of individual translators perform work for small local vendors, who in turn are hired by larger intermediaries who aggregate work across multiple languages. For a large customer engaging with multiple translation suppliers, the supply chain contains a large number of links, across which localization data and process information must be transmitted intact.

The problem of reliably communicating structured information through this complex chain has been compounded by the ambition of XLIFF itself. In addition to carrying enough data to serve as an interconnect, XLIFF 1.2 was designed to carry high-level process data from one end of a tool chain to another. This increased complexity by expecting different tools and organizations to agree on how and when this metadata should be processed. Any inconsistency in the tool chain in how these values are used and consumed could render the metadata incorrect or useless.

1.2 Competing Design Objectives

In order to serve this complex ecosystem, the development of XLIFF has been impeded by a bevy of competing design objectives. As with any standard, there is an inherent tension between a simple standard, which is easy to understand and implement, and a complex standard, which can provide additional features. The fluid nature of the localization industry also creates a tension between

the need for rigid standards, which provide strong guarantees of interoperability, and flexible standards, which allow for extension and customization. Similarly, it has been difficult to balance a descriptive approach which canonicalizes features of existing implementations against a prescriptive approach which demands certain functionality for implementing tools.

1.3 Why do Standards Fail?

The failure of standards is not a problem unique to the localization industry. In 2011, Carl Cargill, a Standards Principal at Adobe, published a paper called "Why Standardization Efforts Fail" (Cargill 2011) that analyzed common factors across industries that led to the unsuccessful development and adoption of standards. These covered all stages of standards development, including conceptual, developmental, and implementation failures. Of the six major categories Cargill identifies, three are of particular interest when thinking about XLIFF 1.2.

1.4 Feature Creep

"Feature Creep" in the standards world is analogous to its meaning in engineering: the standard contains such an abundance of functionality that it loses focus and becomes difficult to implement. Cargill cites an over-reliable on compromise by the standards body as one of the chief causes of feature creep in standards.

Whether or not compromise was in fact at fault, there is clear evidence of feature creep in XLIFF 1.2. Some concepts, such as inline tags, have multiple representations when fewer would have sufficed. Some pieces of metadata, particularly related to process information, have no clear semantics and do not identify the problem they are meant to solve. And some inclusions, particularly related to software localization metadata, seem unrelated to the rest of the specification.

The impact of this has led to a large set of functions without any clear guidelines from the specification about what is truly necessary. Tool vendors have tended to implement arbitrary subsets of the available feature set, as documented by Micah Bly (Bly 2010), among others.

1.5 Incompatible Implementations

The problem of incompatible implementations is simple to understand, but its causes can be subtle. In addition to the incompatibility that results from partial implementations of the standard, Cargill also recognizes ambiguity and omission in the language of the standard itself as a critical problem. Developers, he argues, prefer to pick the simplest solution that can be labelled "standard-compliant", and so any linguistic wiggle room that they find may be exploited.

Incompatible implementations may be the single greatest problem with XLIFF 1.2. It is generally understood that only the barest subset of XLIFF 1.2 features can be interchanged and correctly processed across a heterogeneous tool chain. Consistent implementation of even common features like the use of the alt-trans element for memory proposals can be difficult to achieve without tightly controlling the set of tools that are allowed to touch the file.

The causes for this incompatibility across XLIFF 1.2 implementations are varied, and extend beyond simply an inconsistent feature set. The overloading of some features, such as the `<alt-trans>` and `<mrk>` elements, led to confusion amongst tool vendors, many of whom support these features only partially. The ambiguity in the plain language of the standard that Cargill cites is also present: does the match-quality attribute support decimal points? Does it require (or allow) the presence of a percent sign? An over-broad extension mechanism allowed for the development of many tool-specific variants of the format that further hinder interoperability.

Perhaps most importantly, the standard provides little assistance in proving that a given implementation is correct or incorrect. XLIFF 1.2 lacks processing expectations, a set of test cases to verify correct operation, or even a set of requirements for compliance. The meaning of "processing XLIFF 1.2" is left up to the implementor.

1.6 Market Indifference

Cargill describes market indifference as a situation where a standard is completed, but not widely adopted in the market. Most commonly, this happens when the market has already moved on to another solution to the same problem, and no longer has a need of the standard.

For XLIFF 1.2, this has not been a problem. XLIFF 1.2 has been, and continues to be, widely supported, albeit in inconsistent ways. The more interesting question is whether market indifference could hamper the adoption of XLIFF 2.0.

There are several factors working against XLIFF 2.0.

The size and complexity of the standard may make tool vendors reluctant to invest the engineering resources to support it when the market for it is not already developed; this could create a chicken-and-egg situation. XLIFF 2.0 is also not backwards-compatible with XLIFF 1.2, which provides a migration hurdle for existing implementations. Lastly, the rise of different forms of data exchange, in particular web services, provide alternatives to XLIFF for the exchange of localization data.

2. Can XLIFF 2.0 Learn from Past Mistakes?

The development and standardization of XLIFF 2.0 spanned several years of effort by the OASIS XLIFF Technical Committee, and they made a conscious effort to address many of the shortcomings of XLIFF 1.2. How do the changes in the new version of the standard work to avoid the problems seen in the past?

2.1 Feature Creep

XLIFF 2.0 streamlines many aspects of XLIFF 1.2, but it also adds several sizeable new features, including the ability to embed terminology data and enforce size and length restrictions on content. Overall, the number of available features has increased. Will this exacerbate the complexity problems found in the previous version?

I am optimistic that it does not, thanks in large part to the addition of the module mechanism in XLIFF 2.0. Modules are a meta-feature, a system for organizing other aspects of the XLIFF format into functional clusters that must be implemented or ignored as a whole.

This change has a profound effect on the overall complexity of the standard. Although the number of elements has increased, whether an implementation now supports a given "feature" is now a discussion of whether or not it supports a "module", a much coarser distinction. This simplifies compatibility discussions and helps developers prioritize within their implementations. Additionally, modules provide a regular way for implementations to *not* support a feature, thanks to the requirements regarding the handling the markup of unsupported modules. Restrictions against re-implementation of core features provides an important check against abuse.

Additionally, the new functional capabilities of XLIFF 2.0 generally reflect conventional wisdom regarding things that were missing from XLIFF 1.2, rather than an attempt to unify disparate implementations from existing tools. For example, the lack of terminological support in XLIFF 1.2 is well-known, which has led to the common practice of bundling a standalone TBX file (or other term format) with XLIFF.

2.2 Consistent Implementations

One of the most noticeable changes when reading the XLIFF 2.0 specification is its size — it's much longer than the previous version. To a developer, the XLIFF 1.2 specification is terrifyingly slim, and this more verbose style is a welcome change. The Technical Committee has consciously focused on improving the number and quality of available examples in the text, as well as clearly describing processing instructions for a number of features.

Consistency of implementation should also improve thanks to the approach taken with some of the more problematic features from XLIFF 1.2. The functionality of `<alt-trans>`, which was used both as match proposals and as a form of segment history, has been split into two separate features in dedicated modules, Translation Candidate and Change Tracking.

2.3 Pushing for Success

Even with an improved specification, widespread acceptance of XLIFF 2.0 is no sure thing. The Technical Committee, as well as parties that wish to further the spread of XLIFF 2.0, will need to make a concerted effort to drive its adoption.

It is vital to push for the implementation of the XLIFF 2.0 core in as many places as possible, as a stepping-stone towards more advanced functionality. Open source can be a valuable tool for new standards, as it allows for other implementers to quickly embed the functionality and build on it. Investing in high-quality open source implementations, such as the Okapi Framework, should be a priority.

Translation buyers also have an important role to play in the adoption of any standard, should they so choose, as they ultimately have the most to gain from improved interoperability between tools and among vendors.

It is also worth considering the applicability of the XLIFF model to other scenarios where it has not traditionally been applied. XLIFF is an XML-based, document-oriented format, but an industry focus on web services is increasingly dealing with translation

granularity smaller than a document. A web service may exchange a segment or small group of segments, and may use a non-XML format such as JSON to exchange the data. These formats are simpler, but also encode far less information; many of them ignore inline markup entirely. An abstract version of the XLIFF 2.0 data model would be valuable for these implementations, by providing a set of common structures for exchanging rich segment data.

3. Conclusion

Although technology and business demands change, there is little evidence that the basic structure of the localization industry is poised for imminent change. Translation and its associated activities will continue to depend on a multitude of organizations and individuals spread across the globe. The challenges in tying this web together in a way that satisfies the requirements of modern translation buyers is serious.

I believe, however, that we are moving in the right direction. From an implementor's perspective, XLIFF 2.0 offers clear technical benefits over its predecessor that both strengthen the standard and should address some of the critical interoperability problems that weakened its predecessor. In many ways, the next challenge is non-technical, as the XLIFF community pushes for the broad adoption of the standard.

References

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 10 Mar 2015].

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2014) *XLIFF Version 2.0* [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-os.html [accessed 10 Mar 2015].

Cargill, C.F. (2008) "Why Standardization Efforts Fail", *The Journal of Electronic Publishing* [online], 14(1), available: http://dx.doi.org/10.3998/3336451.0014.103 [accessed 10 Mar 2015].

Bly, M. (2010) "XLIFF: Theory and Reality" [online], presented at XLIFF Symposium Limerick, University of Limerick, 22 Sep 2010, available: http://www.localisation.ie/oldwebsite/xliff/resources/presentations/xliff_symposium_micahbly_20100922_clean.pdf

Interoperability of XLIFF 2.0 Glossary Module and TBX-Basic

James Hayes[1], Sue Ellen Wright[2], David Filip[3], Alan Melby[4], and Detlef Reineke[5]
[1] BYU Translation Research Group
[2] Kent State University, Kent, Ohio, USA
[3] University of Limerick, Limerick, Ireland
[4] LTAC Global
[5] Universidad de Las Palmas de Gran Canaria
james.s.hayes@gmail.com, swright@kent.edu, david.filip@ul.ie, alan.melby@gmail.com, detlef_reineke@yahoo.es

Abstract

This article describes a bidirectional mapping between the XLIFF 2.0 Glossary Module and the TermBase eXchange format (TBX), in particular the TBX-Basic dialect. This mapping is slated to be endorsed by the OASIS XLIFF TC as a Committee Note, thus providing the canonical model for interoperability between the two complementary standards. The article recounts the history of the TBX format's evolution from SGML to XML, beginning with its development through TEI to ISO, LISA, ETSI, and TerminOrgs. It presents the core structure of the TBX term entry and explains how the XLIFF Glossary entry easily fits inside this model, facilitating interchange. The structure and data categories of the two models are discussed, followed by a mapping demonstrating the logical conversion path between the two approaches. The role played by the ISOcat Data Category Registry is also introduced. Appendices provide a more detailed view of overall structures and data category assignments.

Keywords: *XLIFF, TBX, Interoperability, Standards, Translation, Terminology, Terminology Exchange, Terminology Management*

1. Introduction

The XML Localisation Interchange File Format (XLIFF) standard from OASIS is intended to function as a file format for the interchange of localisable data in bitext form that are passed between tools during a localisation/translation process, with the primary goal of lossless information transfer (Comerford, T., Filip, D., Raya, R.M., Savourel, Y., Eds. 2014; Savourel, Y., 2014). XLIFF 2.0 allows for terms in segments of the bitext to be linked to simple entries in an optional glossary module intended to store important term-related information as fully as possible without overstepping the scope of the overall file. This specific intended purpose also allows the XLIFF glossary module to maintain compatibility with larger glossary formats which are specialized for the task of terminology management, such as the Basic dialect of TermBase eXchange (TerminOrgs, 2014).

TBX-Basic is a dialect of the XML-based ISO 30042, *TermBase eXchange (TBX)* format (ISO 30042:2009, referred to in this article as *the TBX Standard*) and is intended to be, as its name suggests, simpler than its older and more powerful cousin *TBX-Default*, which comprises the full scope of the standard. TBX-Basic is not a standard per se (although it is sometimes referred to as a *de facto* standard) and is considered a *guideline* for use in localisation environments. Where TBX-Default has more than 120 data categories and many of those can be used with multiple type values, TBX-Basic is a fully contained subset of TBX-Default that features 28 data categories (DCs) and substantially reduces the number of permissible instances assigned to various DCs. Nevertheless, TBX-Basic is still capable of storing a large amount of terminological information, is fully compatible with the core TBX standard, and adheres to the constraints of a terminological markup language (TML) as defined by ISO 16642, *Terminological Markup Framework (TMF)* (2003). Figure 1 illustrates the structure of a TMF/TBX data record, which comprises a concept-oriented container called a `<termEntry>`. In addition to conceptual information (possibly including a definition for the concept) pertaining to the entire entry, the term entry has embedded in it at least one `<langSet>` containing all terms for the concept and all related information pertaining to a given language. Included in each langSet is/are one or more `<tig>` elements, each containing a single term in that language and associated with the concept, along with related information, including

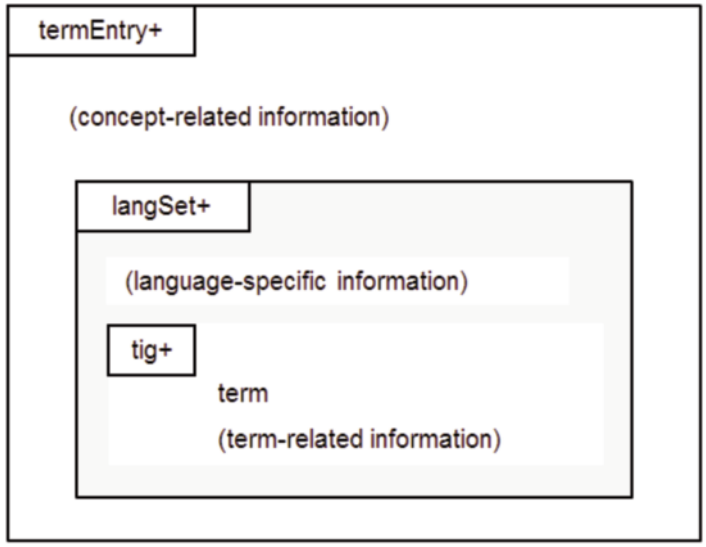


Figure 1: Structural model of a terminological entry in TBX

(optionally) one or more contexts where the term is used in text (the complete TBX core structure is illustrated in Appendix I).

Terminological data categories are generally instantiated as a value of an attribute associated with a meta data-category (`<descrip>`, `<admin>`, `<xref>`, etc.). A small number of data categories is directly instantiated in form of element names (`<term>`, `<date>`, `<note>`) or as attributes (`id`, `xml:lang`). Metadata categories can also be used to group information as shown in the following example:

```
<tig>
       <term>fish</term>
       <descripGrp>
              <descrip
       type="context">This
       is a sample fish
       context</descrip>
              <admin
       type="source">New
       York Times</admin>
       </descripGrp>
</tig>
```

TBX-Basic is intended to be a structurally compliant member of the TBX family of formats that is populated by a selected set of the most common data categories used in fairly uncomplicated terminology databases. Whether by serendipity or by design, TBX-Basic can be treated as structurally compatible with the XLIFF glossary module because the elements in the XLIFF model map easily to a subset of the elements in the TBX-Basic set.

2. TBX Development

The TBX standard has deep roots. It began as Chapter 13 of the Text Encoding Initiative's *P3* iteration of TEI's original SGML-based text markup environment. (Text Encoding Initiative, 1994/1999; Cover, R., 2002). Under the guidance of Alan Melby (Brigham Young University), Klaus-Dirk Schmitz (University of Applied Sciences, Cologne), Sue Ellen Wright (Kent State University), and Gerhard Budin (University of Vienna), it was introduced to ISO and eventually became *ISO 12200:1999, MARTIF*. Here lies the origin of the enigmatic `<martif>` root element, which has been maintained in keeping with a commitment to backward compatibility.

With the general move from SGML to XML as the primary vehicle for encoding textual data, an XML serialization of the MARTIF model was developed through the so-called SALT project under the aegis of the European research 5th framework known as the Human Language Technologies (HLT) project (SALT, 1998-2002). As the format evolved, the Localization Industry Standards Association (LISA) OSCAR (Open Standards for Container/content Allowing Re-use) Special Interest Group (SIG) picked up the project under the leadership of Kara Warburton, publishing the new industry standard

openly on the web (Lommel, A., 2007). It eventually came back to ISO in the form of a jointly published standard, ISO 30042:2008.

Unfortunately, the LISA organization experienced financial difficulties and ceased operations in February 2011, which led to transfer of LISA's intellectual Property including the TBX standard to ETSI in May 2011 (see e.g. Cuddihy, K. 2011). The then chief executive of the organization, recognizing that in its function as a standards body, LISA had produced a number of viable industry standards, TMX (*Translation Memory eXchange)*, SRX (*Segmentation Rules eXchange*) and TBX in particular, chose to transfer the intellectual property rights for the standards to ETSI, the European Telecommunications Standards Institute ISG (Industry Specification Group) (Guillemin, 2011). ETSI now shares further development of the standard with the TerminOrgs (Terminology for Large Organizations) component of LTAC Global (TerminOrgs, LTAC 2014), which enjoys significant joint membership with the old LISA/OSCAR group and with ISO's Technical Committee 37, Sub-Committee 3, *Systems to manage terminology, knowledge and content*.

Both the TBX-Basic and the parent TBX standard are available on the TerminOrgs site (TerminOrgs, 2014). Another important source of TBX and LISA-related information is the GALA/CRISP (Globalization and Localization Association/Collaborative Research, Innovation, and Standards Program), whose mission is to provide a clearing house for information on language industry standards, including the latest (last) versions of LISA's TMX, TBX, and SRX (GALA, 2015). The standards are also available from ttt.org, along with a significant collection of utilities and sample files (TerminOrgs (previously the LISA terminology SIG) at ttt.org, 2015).

A further source of information under development is the TBXinfo website at www.tbxinfo.net, which is slated to provide a full range of support materials concerning the TBX standard (ISO 30042) and its various forms and dialects (TBX-Default, TBX-Basic, TBX-min). For referencing TBX in web-related xml documents, the namespace is `http://iso.org/ns/tbx`. Many of the TBX data categories (DCs) are already available via persistent identifiers (PIDs) of the form:

`https://www.isocat.org/datcat/DC-[xxxx]`,

where `[xxxx]` represents the unique ID of a given DC in the Data Category Registry (see below). Anyone wishing to examine a sample TBX database may download the IATE (*InterActive Terminology for Europe*) termbase, which contains 8 million terms in 24 European languages. The intention of this massive download is to enable users to integrate IATE data into local terminology management systems and Translation Environment Tools (TEnTs).

Work is ongoing to issue an updated version of the full TBX standard, with the goal of introducing enhancements while at the same time maintaining reverse compatibility in order to protect legacy data.

In parallel with the development of TBX, ISO TC 37 has also developed a Data Category Registry (DCR) designed as a dynamic repository of data category specifications, which houses not only TBX-related data categories originally listed in ISO 12620:1999, but several thousand data categories used in a wide range of language resources (ISO 12620:1999; ISOcat, 2015). Originally sponsored by The Max Planck Institute for Psycholinguistics in Nijmegen, The Netherlands, the ISOcat resource has recently changed venue, but remains accessible as a static representation at http://www.isocat.org. It is currently available as a static repository, but plans are under way (at the time of writing, early 2015) for its resurrection as an active data resource residing in the TermWeb environment.

The rather confusing collection of different organizations reflects the need to bring together the essential experts in the field in openly available forums, as some industry organizations are closed to non-paying members, and some industry experts have not affiliated with official standards bodies. ISO standards are desirable on the one hand because they are required in some official venues, but the ISO model contrasts the policy of free and open standards that prevails in the Internet and World Wide Web environment. As a consequence, we could follow the standard being repositioned several times, in order to ensure both the international weight and validity of an ISO standard and the free availability of most components, particularly all processable components, of the standard.

2.1 Scenarios for XLIFF<->TBX Interoperability

The following scenarios describe some of the possible use cases, in which conversion between TBX and XLIFF Glossary data, and *vice versa* will contribute to improved localisation productivity

and/or quality.

- At the beginning of a project, the described mapping will enable agents and users to populate the XLIFF glossary module with data from an existing TBX-Basic compatible termbase, using a conversion utility or web service designed for that purpose.
- An XLIFF-compatible Translation Environment Tool (TEnT) or Computer Assisted Translation (CAT) tool that does not feature an interactive interface with a companion termbase can allow translators to mark terms while translating and automatically store them in the XLIFF glossary module. After completion of the translation, the glossary module data can be harvested for new terms to add to any terminological database using the same TBX mapping in the opposite direction. This procedure can comprise an approval workflow for updating obsolete and adding of new entries in an existing TBX compliant termbase. In both cases, contextual examples can come directly from the bitext, that is the segments or units in which the terms were used can be featured as such examples.
- In the event that one or more TEnTs in the localisation production chain does not have an interactive termbase available to the translator, XLIFF Glossary can be used for terminological support of translators and editors working with the XLIFF file.
- Even in case, interactive termbases are available to translators, the glossary module can be used to provide just the locally relevant terminology and as working space for just the locally relevant terminology for the project.
- Terminology suggestions collected via the glossary module can be used as seed terminology in target or even the source language to jump start terminology management and termbase setup efforts, where it did not previously exist. A scenario more common in the industry than professional terminologists are willing to believe.

All of the above and many more possible scenarios make use of at least two of the four possible facilitated interactions

1. Termbase data and metadata enrich XLIFF using the mapping
2. Translation agents (human and machine) are informed by the seeded data, which helps them make better decisions
3. Human or text analysis agents enter or update data and metadata in the module during the translation process
4. The wider terminology management process consumes data and metadata introduced or curated through the module using the mapping.

3. XLIFF Glossary Module

As defined in XLIFF Version 2.0 (Comerford, T., Filip, D., Raya, R.M., Savourel, Y., Eds. 2014; Savourel, Y., 2014), the XLIFF Glossary Module is a namespace based extension optionally embedded in an XLIFF 2.0 file. The `<glossary>` element is the root element of the module and is only mandatory upon inclusion of the module in an XLIFF file. The module allows the inclusion of simple glossaries and in its current form comprises the following elements: `<glossary>`, `<glossEntry>`, `<term>` (the term occurring in a given context in the source text), `<translation>` (one or more possible target language equivalents) and `<definition>`.

A glossary node can contain one or more `<glossEntry>` elements, and each `<glossEntry>` must contain exactly one `<term>` element. It is accompanied by all relevant information pertaining to this single term as used in a specific translatable text context, including an optional definition, reference to the usage within the translatable text at hand, and possibly multiple translations. Since it only contains information on a single term in the given context, an XLIFF `<glossEntry>` complies with the TMF/TBX requirement that a `<termEntry>` treat a single concept.

Obviously, if users wish to document multiple locally relevant terms, there can be multiple `<glossEntry>` elements in each glossary node. It is interesting to note that in contrast to term entries in termbases, there is only one term reflecting a single given instance of the term in a specific context. There can, however, be multiple equivalents in the case of multiple existing or proposed translations. Should a situation occur in which multiple source terms represent a single concept, there are a few ways to convey this within the Glossary Module:

1. While each `<glossEntry>` can only point to a single source occurrence of the term within the same `<unit>`, XLIFF core term annotation can be used to reference a `<glossEntry>` the other way round, that is from the source text, for

instance in the following cases:

a) The same term has been used more than once in the same <unit> element.

b) Different lemmas of the same term have been used in the same <unit> element.

c) Different synonymous terms have been used throughout the <unit>, <file> or the entire XLIFF file.

2 Use identical definitions in the <definition> elements of synonymous term entries, possibly mention the other synonymous terms in the definition.

3 Use an external termbase concept identifier (ideally a dereferencable URL) to link synonyms. This information can be sent through a dedicated extended attribute or included in the module's own source attribute that is free text and does not have any prescribed semantics.

4 Introduce an extended element to express the synonymy relationship without an external reference. Such element could carry a list of fragment identifiers that point to synonymous terms within the same <unit>, <file> or <xliff> element.

Method 1., possibly combined with method 2., will ensure maximum interoperability along the bitext roundtrip. Information conveyed via methods 3. or 4. would not be interoperable during the XLIFF roundtrip without a pre-agreed handshake mechanism, may be nevertheless critical for terminology post-processing in the termbase environment. If method 3 or 4 has to be used for the sake of automated terminology management outside of the XLIFF based bitext roundtrip, the interoperability during the XLIFF roundtrip should still be ensured using 1 and/or 2.

4. TBX-Basic

As noted above, the root element of TBX-Basic is <martif>, as it is based on the original SGML MARTIF standard (see above). A <martif> element contains a <martifHeader> element and a <text> element. As the XLIFF Glossary Module does not contain any data categories that would map to the <martifHeader>, only the <text> element will be discussed; see the TBX-Basic guidelines on the Terminorgs site for detailed information on <martifHeader>. The <text>

Element/Attribute Name	Description
<glossary>	This is the Glossary Module container element at <unit> level that can contain an arbitrary number of locally relevant glossary entries.
<glossEntry>	Single glossary entry element wrapping a single source term and all related data and metadata, It is extensible by elements and attributes from other namespaces.
<term>	contains one term and only one term, single word or a multi word expression.
<translation>	contains a translation of the sibling <term> element content in the XLIFF file's target language; multiple translations can be proposed as variants or synonyms within the same entry.
<definition>	contains a definition of the concept represented by the term.
ref	IRI that identifies the term as a text span within source or target translatable text of the same <unit> element.
	May be used on <glossEntry> or <translation>
source	free text indicating the origin of the <term>, <translation>, or <definition> content.

Table 1: XLIFF Glossary Elements and Attributes

element contains a <body> element, which contains the terminological entries of the TBX file and is organized as illustrated in Figure 1 and Appendix I.

As it is language specific, the <langSet> element must include an xml:lang attribute representing the language or locale to which it refers in compliance with IETF BCP 47 (2009). At its simplest, such a language code may comprise just the two letter ISO 639-1 code (e.g., "en" for English). It is also commonly combined with ISO 3166 country codes to provide more specific regional information (e.g., "fr-CA for Canadian French), or combined with other information such as script codes. Importantly for the mapping, XLIFF also uses BCP 47 as the norm to indicate its source and target languages at the the <xliff> element level by setting the srcLang ang trgLang attributes. The srcLang attribute determines the language of the Glossary Module <term> element, while the trgLang attribute determines the language of the <translation> elements.

Each <langSet> element contains at least one <tig> (term information group) element. The <tig> element provides all of the specific information on a term such as contextual examples, part of speech, and so forth. In TBX entries, definitions are often anchored either at the <termEntry> or <langSet> levels because they generally pertain to the whole entry or to a specific language.

The <tig> element must include at least one <term>, which contains a plain text representation of a term associated with the <termEntry> concept. Aside from <term>, there are several other elements that may be used to provide additional information, but none are mandatory. Nevertheless *Part of Speech* (<termNote type="partOfSpeech">) is highly recommended, and TerminOrgs maintains that it should be used in all cases to optimize repurposability of the termbase (see Appendix I for more specific information).

5. The Mapping

The following table maps the data categories available in the XLIFF Glossary Module to those in TBX-Basic based on their respective semantics. This mapping has been proposed as a way to enable interoperability between the two formats by laying down a foundation upon which file conversion applications and web services could be based. Because XLIFF <glossEntry> is extensible by attributes and elements from other namespaces, obviously a maximalist one-to-one mapping is possible that could roundtrip all TBX data categories in the TBX namespace elements and attributes. However, such endeavour is not necessary and not even advisable or desirable. Such a full mapping would clutter the minimalistic glossary module with unnecessary information, which would thus undermine the benefit of providing just the locally relevant terminology with the necessary minimum of metadata.

Moreover, default XLIFF and Glossary Module features are expressive enough to roundtrip all mandatory TBX-Basic data categories. Thus this mapping does not consider extensibility allowed in the XLIFF Glossary Module and focuses only on the default elements and attributes specified in the XLIFF standard (Comerford, T., Filip, D., Raya, R.M., Savourel, Y., Eds. 2014; Savourel, Y., 2014). A conversion routine between the two files is under development and will be made available at http://www.tbxinfo.net/tbx-downloads/. A simple example of XLIFF module data, which has been converted to TBX-Basic using this mapping may be found in Appendix III.

The ref attribute actually points to the exact marker delimited span of text that contains just the term within a ; so typically the whole enclosing content will be used as the context content in TBX.

Occasionally a term may span more than one element. If this happens, there must be something wrong going on:

1. either the term is not really a term, or
2. wrong segmentation has been applied.
3. Or authors have erroneously used structural implements for an ad hoc line break, which caused a correct segmentation rule to break a term erroneously.

Nevertheless, such situations do happen and the mapping needs to have a way how handle them. Thus when converting an XLIFF term that spans more than one element, concatenation of all spanned elements will be needed as context for TBX in those cases.

Data category	Representation	Description
Context	<descrip type=”context”>	comprises a sample sentence to show contextual usage of the term
Created by	<transac type="transactionType"> creation </transac>">	appears in a <transacGrp> and accompanied by a <transacNote> specifying the creator’s name and date
Creation date	<date>	appears in the <transacGrp> containing <transac type="transactionType"> creation </transac>
Cross Reference	<ref type=”crossReference” target=”element_id”>	points to another entry or term within the same TBX-Basic file
Customer	<admin type=”customerSubset”>	identifies term that may be required for specific customers
Definition	<descrip type=”definition”>	defines the concept represented by the terms in the term entry
External cross-reference	<xref type=”externalCrossReference” target=”external_id>	points to external reference or explanatory text such as a website link
Figure	<xref type="xGraphic" target="file_location"> description of graphic </xref>	Reference (URI, URL, or local file path) external to the TBX file. The reference is the target value and the description is the element value
Gender	<termNote type=”grammaticalGender”>	indicates grammatical relationships between words in sentences
		Permissible values:
		masculine
		feminine
		neuter
		other
Geographical Usage	<termNote type=”geographicalUsage”>	indicates geographical area of usage (best implemented as a picklist). Should either use ISO 3166 country codes or IETF BCP 47
Last modified by	<transac type="transactionType"> modification </transac>	appears in a <transacGrp> and accompanied by a <transacNote> specifying the modifier’s name and date

Table 2 part 1: TBX-Basic data categories and their representations

Last modification author	<transacNote type="responsibility" target='person_id'>[creator name]	appears in the <transacGrp> containing <transac type="transactionType">modification</transac>. person_id refers to the specific ID given a person in the backmatter
	</transacNote>	
Last modified date	<date>	appears in the <transacGrp> containing <transac type="transactionType">modification</transac>
Note	<note>	any kind of note
Part of Speech	<termNote type="partOfSpeech">	associated with a category assigned to a word based on its grammatical and semantic properties
		Permissible values:
		noun (www.isocat.org/datcat/DC-1333)
		verb (www.isocat.org/datcat/DC-1424)
		adjective (www.isocat.org/datcat/DC-1230)
		adverb (www.isocat.org/datcat/DC-1232)
		properNoun (www.isocat.org/datcat/DC-384)
		other (www.isocat.org/datcat/DC-4336)
Project	<admin type="projectSubset">	identifies terms which may be required for specific jobs/projects
Source of Context	<admin type="source">	indicates the source of context sample. should be found in the <descripGrp> containing context
Source of Definition	<admin type="source">	describes the source of the definition; appears in in the <descripGrp> containing definition

Table 2 part 2: TBX-Basic data categories and their representations

Source of Term	<admin type="source">	indicates the source of the term; appears in in the <descripGrp> containing definition
Term Location	<termNote type="termLocation">	records the location in a user interface where the term occurs, such as <list item> or <button label>
Term Type	<termNote type="termType">	attribute assigned to a term indicating its form; permissible values:
		fullForm
		acronym
		abbreviation
		shortForm
		variant
		phrase
Usage Status	<termNote type="administrativeStatus">	indicates whether a term is approved for use or not
		Permissible values (note they are simplified in TBX-Basic):
		preferred (www.isocat.org/datcat/DC-72)
		admitted (www.isocat.org/datcat/DC-73)
		notRecommended (www.isocat.org/datcat/DC-74)
		obsolete (www.isocat.org/datcat/DC-75)

Table 2 part 3: TBX-Basic data categories and their representations

XLIFF Elements and Attributes	**TBX-Basic**	**Comment**
<glossEntry>	<termEntry>	
<term>	<term>	
<translation>	<term>	<term> belonging to the target language's <langSet>
<definition>	<descrip type="definition">	
ref	<descrip type="context">	see Page 28 Column 2 Paragraph 3
source	<admin type="source">	

Table 3: XLIFF-TBX Mapping-

6. Conclusion

When mapping TBX-Basic mandatory data cetegories to XLIFF core and Glossary Module, we meet with a fairly straightforward match. There is perhaps one subtlety worth noting. While TBX-Basic requires definition or context information for a concept entry to be valid, XLIFF Glossary module requires either a definition or a translation for a valid glossary entry. Hence in cases when XLIFF glossary entries are not provided with a definition as they don't have to be. The valid TBX-Basic entry needs to extract context information. That is however always present in the underlying XLIFF core bitext. Thus the bidirectional mapping is feature complete. If a particular process needs to make use of optional TBX categories, these can be always roundtripped using XLIFF core and Glossary module extension points. This aspect has not been however discussed except as a brief mention as an option for handling source synonymy.

Terminologists and lexicographers have been making for a long time the distinction between lexicographical resources and terminological resources, asserting that lexicographical entries are word-centred with potentially many associated senses, while terminological entries are concept-centred with potentially many terms (synonyms) and target language equivalents. In contrast to these traditional models, the XLIFF Glossary Module entry documents a single term embedded in the context of the source language component of a bitext and provides the option to link that term to one or more potential target language equivalents (Figure 2). This paper demonstrates that this model is mappable to the TBX interchange model (specifically TBX-Basic) because a single term in a single context comprises one feature complete facet of a concept-oriented terminological entry. This mapping, together with the appropriate utilities, will enable users working in a variety of technical writing and localisation environments to utilize context-grounded terminological information across applications and platforms.

References

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2014) *XLIFF Version 2.0* [online], OASIS Standard. Available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-os.html [accessed 22 Aug 2014].

Cover, R. (2002) Technology Reports: Text Encoding Initiative (TEI). Available at:

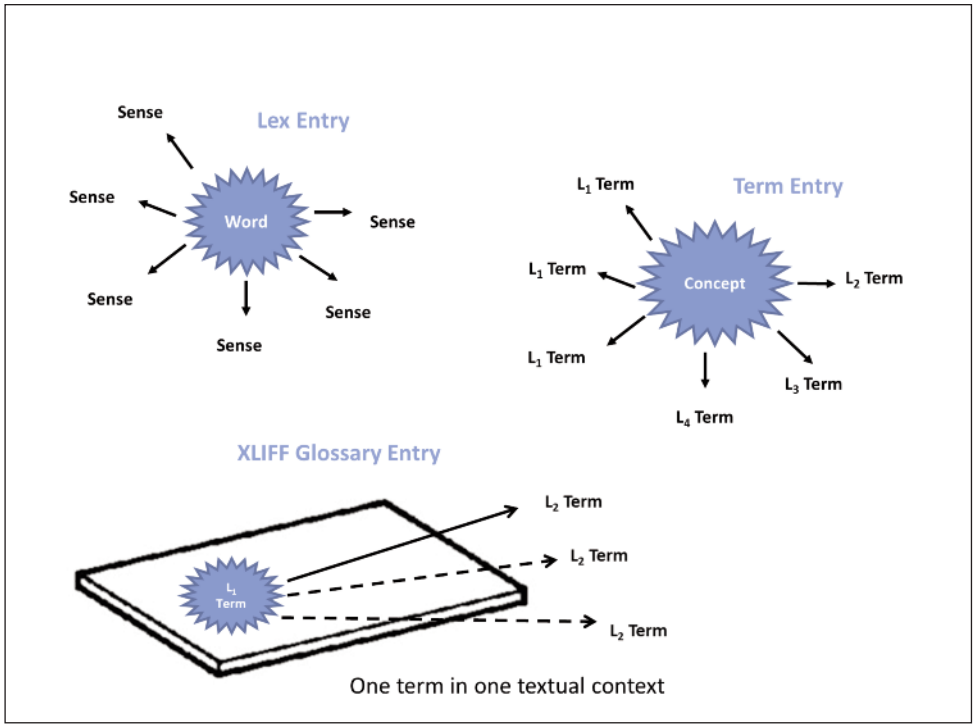


Figure 2: Lexical, Terminological, and XLIFF Structural Model

http://xml.coverpages.org/tei.html [accessed 18 Jan 2015]

Cuddihy, K. (2011) LISA Intellectual Property Turned Over to ETSI. STC Notebook. Available at: http://notebook.stc.org/lisa-intellectual-property-turned-over-to-etsi/ [accessed 17 Jan. 2015]

Ethnologue. (2015) Ethnologue Languages of the World: Browse by Language Name, Language Code, Language Family, Map Title. Available at http://www.ethnologue.com/browse [accessed 20 Jan 2015] Note: all ISO 639 codes are available in the entries for each language.

GALA (2015) LISA OSCAR Standards. Available at: http://www.gala-global.org/lisa-oscar-standards [accessed 18 Jan 2015]

Guillemin, P. (ETSI Secretariat) (2011) Correspondence: In the beginning WhP and ETSI Technical Committee Human Factors...Available at: http://docbox.etsi.org/ISG/Open/ISGLIS/LocWorld-SantaClara/Patrick%20GUILLEMIN%20TEXT%20%20in%20C1%20v2.pdf [accessed 18 Jan 2-15]

IATE. (2014) InterActive Terminology for Europe. Available at: http://iate.europa.eu/; http://iate.europa.eu/tbxPageDownload.do [accessed 18 Jan 2015]

IETF BCP 47. (2009) Tags for Identifying Languages. Available at: https://tools.ietf.org/html/bcp47 [accessed 19 Jan 2015]

ISO 639. (Varies) Family of Language Code standards: see *Ethnologue*.

ISO 3166. (2015) Country Codes Online Browsing Platform (OBP). Available at: https://www.iso.org/obp/ui/#search [accessed 19 Jan 2015]

ISO 12200:1999 Computer applications in terminology – Machine-readable terminology interchange format (MARTIF) – Negotiated interchange. Withdrawn. Geneva: ISO.

ISO 12620. (1999) Computer Applications in Terminology – Data Categories. Geneva: ISO. Withdrawn.

ISO 12620. (2009) Terminology and other language and content resources – Specification of data categories and management of a Data Category Registry for language resources. Geneva: ISO.

ISO 16642. (2003) Computer applications in terminology – Terminological markup framework (TMF). Geneva: ISO.

ISO 30042. (2008) *Systems to manage terminology, knowledge and content – TermBase eXchange (TBX)*. Geneva: ISO.

ISO. (2015) ISOcat Data Category Registry. Available at: http://www.ISOcat.org [accessed 18 Jan 2015]

Kemps-Snijders, M.; Windhouwer, M.; and Wright, S.E. ISOcat: An ISO 12620:2009 Data Category Registry. Available at: http://www.slideshare.net/mwindhouwer/isocat-an-iso-126202009-data-category-registry [accessed 18 Jan 2015]

Lommel, A. (2007) "OSCAR Standards for Localization and Globalization Environments" Available at: http://www.ttt.org/TC37/ISO%20Conference%202007_files/Arle_LISA%20standards.pdf [accessed 17 Jan 2015]

LTAC Global (Language Technology and Authoring Consortium) Available at: http://www.ltacglobal.org/ [accessed 18 Jan 2015]

OASIS. (2014) XLIFF Version 2.0: OASIS Standard. See Comerford et al. above.

SALT. (1998-2002) Standards-based Access to multilingual Lexicons and Terminologies. Available at: https://web.archive.org/web/20090319040215/http://www.loria.fr/projets/SALT/saltsite.html [Accessed 17 Jan 2015 via Wayback Machine]

Savourel, Y. (2014) An Introduction to XLIFF 2.0. Multilingual, pp 42-47. Available at: http://dig.multilingual.com/201406/8B19207B6B20FA6ADBAB2612383D9EEF/201406.pdf [accessed 2015-01-20]

TBXinfo. Available at: www.tbxinfo.net [accessed 18 Jan 2015]

TBX-Basic. (2015) [See: TerminOrgs, ETSI, tbxinfo.net]

Text Encoding Initiative. (1994) Part 3: Base Tag Sets, 13: Terminological Databases. In: Sperberg-McQueen, C. M., and Burnard, L, Eds. *Guidelines for Electronic Text Encoding and Interchange*.P3 Revised reprint, Oxford, May 1999. Available at: http://quod.lib.umich.edu/cgi/t/tei/tei-idx?type=HTML&rgn=DIV1&byte=1158058 [accessed 16 Jan 2015]

TerminOrgs (Terminology for Large Organizations) (2014) TBX-Basic Version 3.1; Termbase eXchange (TBX). Available at: http://www.terminorgs.net/downloads/TBX_Basic_Version_3.1.pdf [accessed 18 Jan 2015]`

Terminorgs/LISA. (2014) An Archive of Oscar Standards: Termbase eXchange (TBX). Available at: http://www.ttt.org/oscarStandards/tbx/ [accessed 18 Jan 2015]

TermWeb. (2015) http://www.interverbumtech.com/ (See also ISOcat.org above)

Appendix I - TBX-Basic Implementation Guide

This Appendix describes the elements required to create a valid TBX-Basic file. TBX-Basic can be validated using the TBX Checker with the TBX-Basic DTD file (TBXBasiccoreStructV02.dtd) and XCS file (TBXBasicXCSV02.xcs). Each of these items can be found in the TBX-Basic Package at the website: http://www.tbxinfo.net/tbx-downloads/

The prescribed file structure is shown in figure 3:

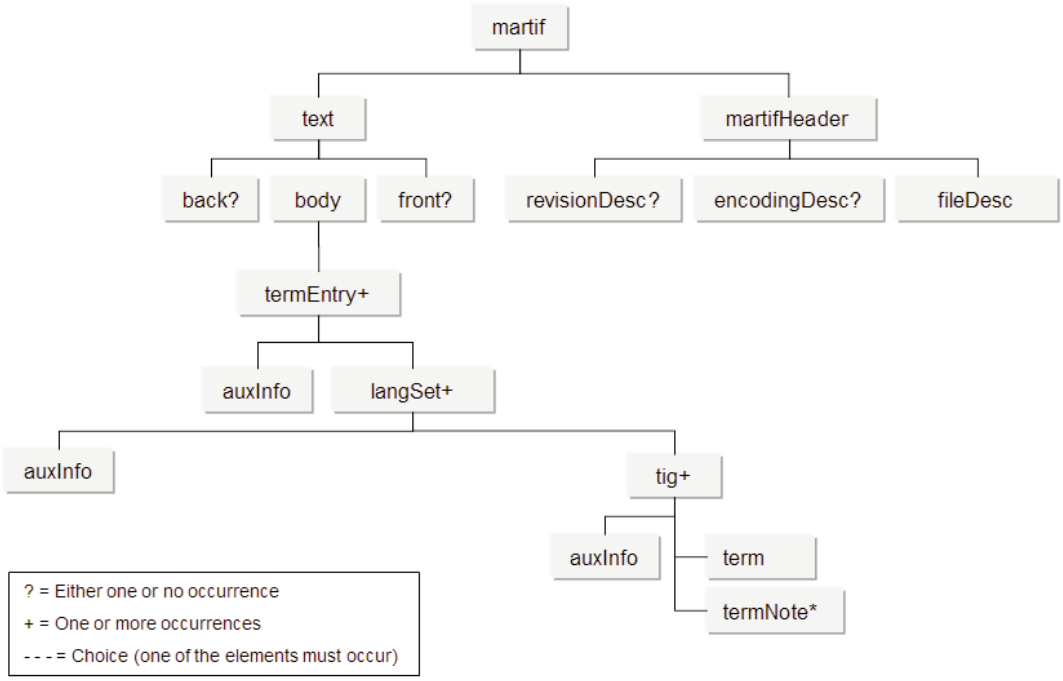


Figure 3 TBX-Basis structure

Other constraints:

- The <back> element is required if internal references in <body> (such as in creator or modifier) point to the ID of a person listed in the back matter. The auxInfo box represents the meta data-categories representations such as <descrip>, <descripGrp>, <admin>, <adminGrp>, <xref>, etc.
- One of definition or context is required.

Appendix II - XLIFF Core + Glossary Module tree and Constraints

(see http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-os.html)

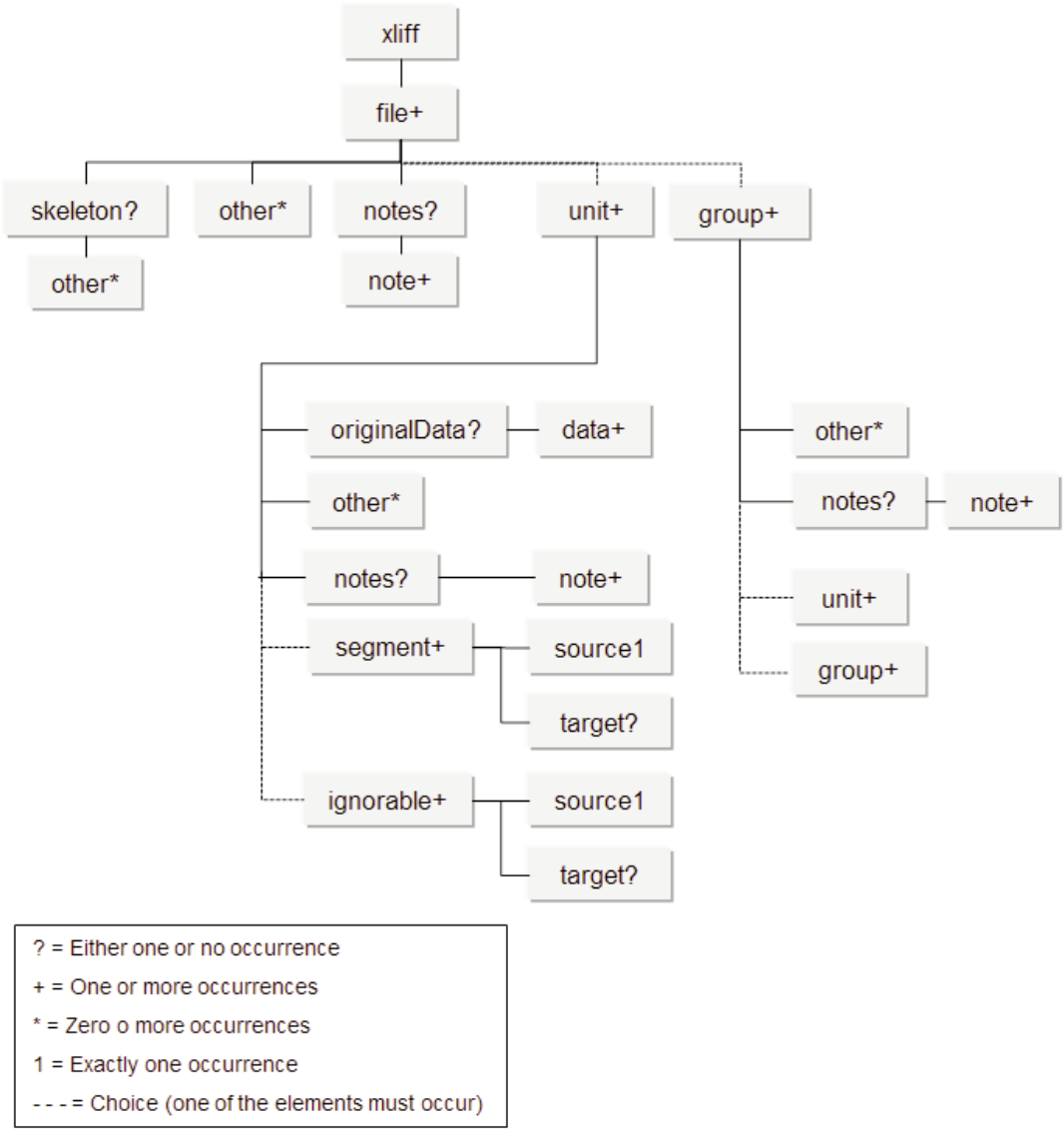


Figure 4: XLIFF Core structure

Source terms appear within the `<source>` children of `<segement>` elements, translated terms appear within their `<target>` siblings.

In order to reference terms in context from the Glossary Module, the term spans need to be delimited using the XLIFF core term annotation, making use of `<mrk>` elements or `<sm/>/<em/>` pairs.

The `<glossary>` wrapper is allowed at each `<unit>` element. The Glossary Module structure is shown in figure 5.

Constraints

- A `<glossEntry>` element MUST contain a `<translation>` or a `<definition>` element to be valid.

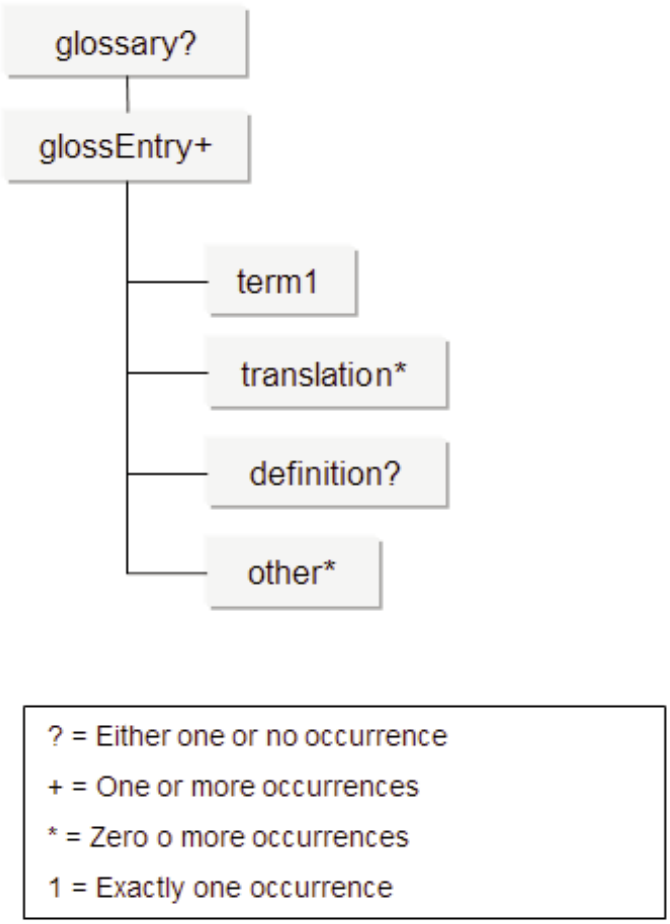


Figure 5 - XLIFF Glossary Module structure

Appendix III XLIFF Glossary Module to TBX-Basic sample conversion

These files can be downloaded at: http://www.tbxinfo.net/tbx-downloads/

XLIFF File

```
<?xml version="1.0" encoding="UTF-8"?>
<xliff xmlns="urn:oasis:names:tc:xliff:document:2.0" version="2.0" srcLang="en"
trgLang="de"
     xmlns:gls="urn:oasis:names:tc:xliff:glossary:2.0">
     <file id="f1">
     <unit id="1">
          <gls:glossary>
               <gls:glossEntry ref="#m1">
                    <gls:term source="publicTermbase">TAB key
                    </gls:term>
                    <gls:translation id="1" source="myTermbase">Tabstopptaste
                    </gls:translation>
                    <gls:translation ref="#m2" source="myTermbase">TAB-TASTE
                    </gls:translation>
                    <gls:definition source="publicTermbase">A keyboard key that
istraditionally used to insert tab characters into a document.
                    </gls:definition>
               </gls:glossEntry>
          </gls:glossary>
          <segment>
               <source>Press the <mrk id="m1" type="term">TAB key</mrk>.
               </source>
               <target>Drücken Sie die <mrk id="m2" type="term">TAB-TASTE</mrk>.
               </target>
          </segment>
     </unit>
     </file>
</xliff>
```

TBX File

```
<?xml version='1.0'?>
<!DOCTYPE martif SYSTEM "TBXBasiccoreStructV02.dtd">
<!— THIS FILE MAKES USE OF THE TBX NAMESPACE —>
<martif type="TBX-Basic" xml:lang="en-US" xmlns="iso.org/ns/tbx/2016">
      <martifHeader>
            <fileDesc>
                  <titleStmt>
                        <title>XLIFF 2.0 Glossary Module to TBX-Basic
Demonstration</title>
                  </titleStmt>
                  <sourceDesc>
                        <p>
                              This is a demonstration of a potential mapping from
the glossary module of XLIFF 2.0
                               to TBX-Basic.
                        </p>
                  </sourceDesc>
            </fileDesc>
            <encodingDesc>
                  <p type="XCSURI">TBXBasicXCSV02.xcs
                  </p>
            </encodingDesc>
      </martifHeader>
      <text>
            <body>
                  <termEntry>
                        <langSet xml:lang="en">
                              <tig>
                                    <term>TAB Key</term>
                                    <admin type='source'>publicTermbase</admin>
                                    <descripGrp>
                                          <descrip type='definition'>A keyboard key
that is
                                             traditionally used to insert tab characters
into a document.
                                 </descrip>
                                          <admin
type='source'>publicTermbase</admin>
                                    </descripGrp>
                                    <descripGrp>
                  <!— Here the segments were pulled from <segment> and used as data for
an 'example' —>
                                          <descrip type='context'>Press the TAB
key.</descrip>
                                    </descripGrp>
                              </tig>
                        </langSet>
                        <langSet xml:lang="de">
                              <tig>
                                    <term>Tabstoptaste</term>
                                    <admin type='source'>myTermbase</admin>
                              </tig>
                              <tig>
                                    <term>TAB-TASTE</term>
                                    <admin type='source'>myTermbase</admin>
                                    <descripGrp>
                        <!— Here the segments were pulled from <segment> and used as data
for an 'example' —>
                                          <descrip type='context'>Drücken Sie die
TAB-TASTE</descrip>
                                    </descripGrp>
                              </tig>
                        </langSet>
                  </termEntry>
            </body>
      </text>
</martif>
```

Using Semantic Mappings to Manage Heterogeneity in XLIFF Interoperability

Dave Lewis, Rob Brennan, Alan Meehan, Declan O'Sullivan
CNGL Centre for Global Intelligent Content, Knowledge and Data Engineering Group,
School of Computer Science and Statistics, Trinity College Dublin, Ireland
dave.lewis@scss.tcd.ie, rob.brennan@scss.tcd.ie. meehanal@scss.tcd.ie, declan.osullivan@scss.tcd.ie

Abstract

The XLIFF 1.2 standard features several extension points that have served to complicate the full interoperability of translation content meta-data between tools from different vendors. Many vendors' specific extensions are in common use. XLIFF profiles promoted by individual large tool vendors or by consortia of smaller vendors (e.g. Interoperability Now!) attempt to reduce this complexity. However, as no one profile dominates, the overall result is that many XLIFF profiles are now in common use that extend the original standard in different ways. The XLIFF 2.0 standard attempts to control the evolution of extensions through the managed definition of new modules. However, until XLIFF 2.0 fully supplants the use of XLIFF 1.2 and its variants, tools vendors and language service providers will have to handle a range of different XLIFF formats and manage heterogeneity in the use of meta-data that impairs its use in automating steps in the localisation workflow.

Managing the mapping of different XLIFF profiles to an internal representation requires therefore, either extensive coding knowledge, or the understanding and maintenance of a wide range of different XSL Transforms. In this work we describe an alternative approach to handling the design, implementation and maintenance of meta-data mappings using semantic web technologies.

Keywords: *Semantic Mapping, Interoperability, Multilingual Web, XLIFF, RDF*

The localization industry is built on heterogeneous tool-chains with strong interoperability requirements. The XLIFF (XML Localization Interchange File Format) standard was established to enable greater interoperability between tools from different vendors. The XLIFF 1.2 (Savourel et al. 2008) standard has included several extension points to its structure with the aim to help provide greater interoperability between tools. However, these extensions have caused confusion among tool vendors and are rarely utilized. Instead, individual tool vendors have established their own extensions and as a result, many different extensions are in use causing complex interoperability issues. Specific XLIFF profiles, promoted by individual large tool vendors or by a consortium of smaller vendors, attempt to reduce the complexity and interoperability issues. Since no one profile dominates, the result is that many XLIFF profiles are now in use, which deviate from the XLIFF 1.2 standard in different ways. The XLIFF 2.0 (Comerford et al. 2014) standard attempts to control the evolution of existing extensions through the managed definition of new modules. Until the XLIFF 2.0 standard fully supplants the XLIFF 1.2 standard and the XLIFF profiles already in existence, tool vendors and language service providers still have to cope with the interoperability issues caused by the multitude of XLIFF formats in existence.

In this work, we present an alternative approach of overcoming XLIFF interoperability using Semantic Web technologies. In previous work (Lewis et al. 2012), a process is described how the use of Extensive Stylesheet Language Transformations (XSLT) (Kay 2007) at different points in the localization workflow can be used to uplift multilingual content and meta-data into a Resource Description Framework (RDF) (Cyganiak et al. 2014), Linked Data (Bizer et al. 2009) representation, also known as Linked Language and Localization data or L3Data for short. This provides a decentralized representation of the data, publishable on the web, where it can be shared among

localization enterprises for mutual benefit. Such benefits include access to a larger pool of language resources to aid in translation services and large datasets to train Statistical Machine Translation (SMT) tools. Interoperability issues are still present within the L3Data as multiple heterogeneous domain and tool-specific vocabularies are often employed within the RDF. However, the use of semantic mappings (Euzenat & Shvaiko 2013) can be employed to reduce this heterogeneity by transforming the L3Data from one vocabulary to another.

Our mapping representation, which we presented in (Meehan et al. 2014), is an RDF-based mapping representation that can be used to represent mappings between different L3Data vocabularies. The mapping representation uses a combination of SPARQL Inferencing Notation (SPIN) (Knublauch 2013) and meta-data. The executable specification associated with the mapping representation is a SPARQL (Harris & Seaborne 2013) construct query, which is executable on any standard SPARQL endpoint. The objective of the mapping representation is to provide a more agile approach to translation workflows and greater interoperability between software tools by allowing specific tool vendors to publish mappings, alongside the L3Data that they publish. This allows consumers of the L3Data to discover these mappings, through the use of SPARQL queries and execute them via a SPARQL processor.

Our use case is a Language Technology retraining workflow where publishing mappings leads to new opportunities for interoperability for the retraining of machine translation tools. Figure 1 below displays the process where a piece of HTML source content is acted upon by specific tools in a localization workflow. An XLIFF file is used to record the processing that the source content undergoes at each step of the workflow. At the end of the workflow, a custom tool using the XSLT language is used to uplift the data in the XLIFF file, to an L3Data representation, using the Global Intelligent Content (GLOBIC) semantic model (Brennan & Lewis 2014) vocabulary and store it in a triple store. This L3Data represents details such as the *source* and *target* of text content that underwent a Machine Translation (MT) process, which tool carried out the MT process, *post edits* and *quality estimates* associated with translated content. By building up L3Data in the triple store, it becomes a rich source of MT training data. The retraining aspect of the workflow involves retrieving content to be fed back into the SMT tool. This is achieved by querying the triple store for translated content with a quality estimate over a certain threshold value. SMT tools from different vendors, looking to utilize this L3Data for retraining purposes, need to have it mapped to a vocabulary they recognize. In Figure 1, the *MT tool* is unaware of the GLOBIC vocabulary, it is designed to consume data according to the Internationalization Tag Set (ITS) (Filip et al. 2013) vocabulary. The *Quality Estimate (QE)* and *Post Edited (PE) data* that is represented in GLOBIC must be mapped to an ITS representation for the *MT tool* to use it. Our mapping representation can be used in this situation since it is stored alongside the L3Data in the triple store. Mappings between the GLOBIC and ITS vocabularies can be discovered by a user/tool, through SPARQL queries

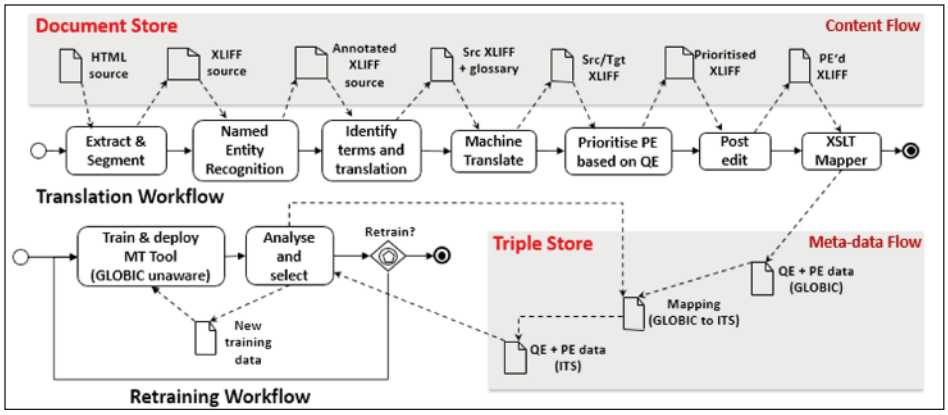


Figure 1. Language Technology Retraining Workflow

and executed. This will transform the L3Data, allowing the SMT tool to consume it.

References

Bizer, C., Heath, T. & Berners-Lee, T. (2009). Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3), 1-22.

Brennan, R. & Lewis, D. (2014). *The Global Intelligent Content Semantic Model Specification*. [Online] available: https://www.scss.tcd.ie/~meehanal/gic.ttl [Accessed 20 Jan 2015].

Comerford, T., Filip, D., Raya, R. & Savourei, Y. (2014). *XLIFF Version 2.0*. [Online] available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html [Accessed 20 Jan 2015].

Cyganiak, R., Wood, D. & Lanthaler, M. (2014). *RDF 1.1 Concepts and Abstract Syntax*. [Online] available: http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/ [Accessed 20 Jan 2015].

Euzenat, J. & Shvaiko, P. (2013). Classifications of Ontology Matching Techniques. *Ontology Matching*, pp.79-84.

Filip, D. et al.(2013). *Internationalization Tag Set (ITS) Version 2.0*. [Online] available: http://www.w3.org/TR/its20/ [Accessed 20 Jan 2015].

Harris, S. & Seaborne, A. (2013). *SPARQL 1.1 Query Language*. [Online] available: http://www.w3.org/TR/sparql11-query/ [Accessed 20 Jan 2015].

Kay, M. (2007). *XSL Transformations (XSLT) Version 2.0*. [Online] available: http://www.w3.org/TR/xslt20/ [Accessed 20 Jan 2015].

Knublauch, H. (2013). *SPIN - SPARQL Syntax*. [Online] available: http://spinrdf.org/sp.html [Accessed 20 Jan 2015].

Lewis, D. et al. (2012). On Using Linked Data for Language Resource Sharing in the Long Tail of the Localisation Market., 2012. LREC.

Meehan, A., Brennan, R., Lewis, D. & O'Sullivan, D. (2014). Mapping Representation based on Metadata and SPIN for Localization Workflows. *In Proceedings of the Second International Workshop on Semantic Web Enterprise Adoption and Best Practice at ESWC*.

Savourel, Y., Reid, J., Jewtushenko, T. & Raya, R. (2008). *XLIFF Version 1.2*. [Online] available: http://docs.oasis-open.org/xliff/v1.2/os/xliff-core.html [Accessed 20 Jan 2015].

Advanced Validation Techniques for XLIFF 2

Soroush Saadatfar and David Filip
Localisation Research Centre, CNGL,
University of Limerick, Ireland
soroush.saadatfar@ul.ie**,** david.filip@ul.ie

Abstract

This paper aims to present an overview of validation issues for the XML Localization Interchange File Format (XLIFF), while XLIFF 1.2 validation is mentioned, the paper concentrates on validation of XLIFF 2.0 and successors (XLIFF 2). The goal is to propose an optimal set of DSDL (Document Schema Definition Languages) (Brown 2010) compliant methods to provide a set of XLIFF TC (and in due course OASIS and ISO) guaranteed standardized machine readable artefacts that would support validation of all XLIFF 2 normative statements. The discussed methods include XSD files, proposed possible schemas in the Relax NG schema language, Schematron rules, and finally a master NVDL (Namespace-based Validation Dispatching Language) file that brings all the necessary and guaranteed methods together. Development of a lightweight demonstrator RESTful web service that wraps all the proposed DSDL methods in one validator platform is also discussed and foreshadowed.

Keywords: XLIFF, validation, Relax NG, XSD, Schematron, NVDL, DSDL, XML, RESTful, web-service, web service

1. Introduction

According to the W3C definition, Extensible Mark-up Language (XML) is designed to describe data by providing a flexible text format derived from SGML (Standard Generalized Mark-up Language) (Bray et. al. Eds. 2008). As XML is a software- and hardware-independent format for carrying information, it has become an important tool for information exchange among applications. XML is meant to be easily understood by humans, yet – at the same time – machine-readable (w3schools 2014).

XML provides a minimum set of rules for creating user-defined tags to describe the stored and exchanged data. Therefore, XML – as a metalanguage – provides logical space for creating XML vocabularies. In turn, instances of XML vocabularies need to be validated against a specific vocabulary's schema. An XML schema is a document, which describes the structure of the document and indicates the order of elements with admissible and mandatory attributes and their values. A number of schema languages are available, which can check the XML instances against a variety of rules; from very basic ones – such as hierarchical structure or order of elements –, to complicated conditions, restrictions, and even admissible states in a workflow progression.

XSD (XML Schema Definition) (Thompson et. al. Eds. 2004) is the most popular schema language that is being used widely by XML consumers. In spite of its popularity, XSD is not expressive enough for XLIFF 2 and is not able to target many constraints including complex and advanced rules. Although, the latest version of XSD, 1.1 (Gao et. al. Eds. 2012), can handle many issues omitted in XSD 1.0 (Delima et. al. 2009), it is not broadly implemented and used (Quin et. al. 2012).

Relax NG schema language (Clark et. al. Eds. 2001) can be considered the next step in XML validation. It provides more possibilities for describing complex conditions than XSD. In addition, Relax NG is easier to learn and its expressions and syntax are more intuitive and user-friendly (Quin et. al. 2009; Vlist 2011). This schema has the potential to provide validation solutions to some of the XLIFF 2 constraints that XSD 1.0 cannot express.

Finally, the most powerful and expressive schema language for XML validation is Schematron (Jelliffe Ed. 2002). It can define the most complicated rules and constraints of XML standards and vocabularies. Schematron also provides room for delivering user-defined diagnostics and customized error messages, which can be enriched by detailed information about the objects that failed to comply with any of the validation rules.

All of the mentioned validation techniques are gathered and together make DSDL (Document

Schema Definition Languages). As a multipart ISO standard, DSDL defines a set of schema languages for XML validation (Brown 2010).

2. XLIFF Validation

XLIFF 2.0, as an OASIS standard, is presented in the specification created by the Technical Committee (Comerford, T., Filip, D., Raya, R.M., Savourel, Y.; Eds., 2014). XLIFF contains a large variety of different constraints and rules, which an XLIFF instance must not violate in order to be valid, it also addresses various application conformance targets with its processing requirements. Because of varied requirements for validation expressivity, different techniques for automated validation need to be used to cover the specification in full.

There are a number of tools already developed and available for the purpose of validating different versions of XLIFF. The first tool that addresses XLIFF validation beyond XSD was designed for XLIFF 1.2, it is the *XLIFFChecker* (Raya, 2012). We are however primarily concerned with XLIFF validation, because XLIFF 2.0 does provide, in its prose, specification statements that allow for more advanced DSDL validation.

The third party validators currently available for the 2.0 edition are *XLIFF 2.0 XMarker Checker* (Schnabel) and *Okapi-lynx* (Savourel). Unlike the approach used in this paper, these validators are not DSDL-based, except the basic XSD validation. Their validation methods that go beyond a simple XSD check are implementation dependent in the above. On the plus side, both of the cited XLIFF 2 validators

```
<optional>
   <choice>
      <group>
         <attribute name="type">
            <ref name="atttype"/>
         </attribute>
         <attribute name="subType">
            <ref name="attsubType"/>
         </attribute>
      </group>
      <attribute name="type">
         <ref name="atttype"/>
      </attribute>
   </choice>
</optional>
```

Listing 1: `subType` dependency in Relax NG

do provide validation for XLIFF 2.0 fragment identifiers, which goes beyond the scope of this paper. This DSDL based approach addresses the wellformedness of an XLIF fragment identifier only as far as it is required to check attributes of the URI or IRI type.

The main target of this paper is to identify and elaborate validation methods and artefacts on the basis of XML standardised (implementation independent) methods of validation, i.e. DSDL schema languages. DSDL artefacts that cover automated validation for a maximal subset of a XLIFF 2 specification are suitable for becoming XLIFF TC deliverables (as part of the multipart standard product according to OASIS definitions) in XLIFF 2.1 and successor editions.

XSD schema – the first part of DSDL – for XLIFF 2.0 core is already provided by as a part of the XLIFF 2.0 multipart OASIS standard. However, this schema is able to provide only very basic validation and moreover, many patterns that pass XSD validation are in fact violating normative statements of the prose specification. For instance, many attributes which are encoded as optional in the XSD schema, may be in fact required or forbidden conditionally, dependent on specific values of other attributes, availability of specific content or elements etc. These advanced constraints are explained in detail further in this section, as we are tackling them with other DSDL methods.

As the first step, the Relax NG schema for XLIFF 2.0 core was developed (Saadatfar 2014). Relax NG schema for XLIFF validation allows one to conduct more detailed document checks than the original XSD. For instance, Listing 1 illustrates how it is possible to handle the constraint for the `subType` attribute (used in some of XLIFF inline elements). The XLIFF specification states:

If the attribute `subType` is used, the attribute `type` must be specified as well.

The schema is defining two valid cases (`<choice>` element allows either of its children to be valid); first if both `type` and `subType` attributes are present in the element, and the second case where `type` appears only. `<ref>` elements will describe the content of each declared node later in the schema.

Despite its advantages, Relax NG is not expressive enough to describe all the normative content of an XLIFF 2 specification. For instance, it does not

support default values for attributes directly, DTD compatibility annotations must be embedded for this purpose. Definition of some rules may turn into extremely long code which is very hard to read and maintain. For instance, according to the specification, the `trgLang` attribute of `<xliff>` element is required if and only if the XLIFF document contains a `<target>` element. This statement could be targeted in Relax NG by defining two patterns for `<xliff>`. This approach would duplicate most of the schema and therefore does not present a practical solution.

For such complex constraints, Schematron offers a suitable solution, the rule-based validation. Listing 2 illustrates the rule for the earlier mentioned statement;

```
<iso:rule context="xlf:target"
 see="http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-
 os.html#xliff">
   <iso:let name="parent-name" value="name(..)"/>
   <iso:let name="unit-id" value="ancestor::xlf:unit/@id"/>
   <iso:assert test="ancestor::xlf:xliff/@trgLang"
    diagnostics="general spec-quote">
      The XLIFF document contains a &lt;<iso:name/>&gt; element, but the
      trgLang attribute of &lt;xliff&gt; element is missing.
   </iso:assert>
</iso:rule>
…
<iso:diagnostic id="general">
   The &lt;<iso:value-of select="name()"/>&gt; was found in the
   &lt;<iso:value-of select="$parent-name"/>&gt; element.
   The id of enclosing &lt;unit&gt; is: '<iso:value-of select="$unit-
   id"/>'.
</iso:diagnostic>
<iso:diagnostic id="spec-quote">
   The trgLang [for &lt;xliff&gt; element] attribute is
   <iso:emph>required</iso:emph> if and only if the  XLIFF Document
   contains &lt;target&gt; elements that are children of &lt;segment&gt; or
   &lt;ignorable&gt;
</iso:diagnostic>
```

Listing 2: `trgLang` check in Schematron

Schematron uses XPath language to access elements inside documents. This approach helps to track errors at all levels.

The rule in listing 3 first seeks `<target>` elements inside the document (as `context` attribute defines) and then applies the constraint; i.e. if the file does not contain any `<target>`, the rule will be ignored as it is not compulsory anymore. After the element was found, at the `<assert>` element the validation test is conducted (assertion). The expression inside the `test` attribute will return "true" if `<xliff>` element has `trgLang` attribute and otherwise, an error will be raised. In the case of an error, the message placed inside `<assert>` will be shown as well as a link (inside `see` attribute) to the XLIFF specification. The `<name>` element returns the name of the picked node (`target` in this case). This element becomes very helpful when dealing with several nodes at the same time and it will be discussed later. Finally, beside all that, thanks to `<let>` elements, which are variables in Schematron, we can save additional information about the error. In this example, we are retrieving the `id` attribute of the enclosing `<unit>` element as well as the immediate parent of the `<target>` element which caused the error (it can be either `<segment>` or `<ignorable>`). Then by taking advantage of diagnostic mechanism of Schematron, an advanced diagnostic message is formed and delivered to the user which makes the error tracking and further fixing much faster and more efficient. It is notable that messages are human-made and are considered individually rather than machine-generated general errors.

One of the other examples that offers a good

```
<element name="skeleton">
   <choice>
      <group>
         <attribute name="href">
            <data type="anyURI"/>
         </attribute>
         <empty/>
      </group>
      <group>
         <interleave>
            <text/>
            <zeroOrMore>
               <ref name="anyElement/>
            </zeroOrMore>
         </interleave>
      </group>
   </choice>
</element>
```

Listing 3: `skeleton` "pseudo-solution", Relax NG

comparison between the Relax NG and Schematron approaches is the `<skeleton>` element's constraint. The `<skeleton>` element must contain either an `href` attribute and no text (empty) or text without the attribute. Listing 3 shows how this issue seems to have been resolved in the Relax NG schema;

Schema again is giving two valid scenarios; either the element is empty and has the attribute, or it is not empty and no attributes are present.

However, both of the following invalid patterns will pass the validation against this schema; `<skeleton/>` and `<skeleton> </skeleton>`. The point is that an empty element or a white-space is still considered text in Relax NG, which allows the mentioned patterns to pass although in fact not valid according the normative prose specification.

Listing 3 shows how this constraint can be addressed by a Schematron rule defining all the possible cases.

This rule perfectly targets the `<skeleton>` constraint by describing it within the expression inside the `test` attribute.

Schematron can cover all remaining XLIFF normative statements including the most important and sophisticated ones, i.e. uniqueness of `id` attribute values among elements in different uniqueness scopes of an XLIFF document.

ID-uniqueness in XLIFF 2.0 comes in three distinct levels of complexity. First, when children elements (i.e. siblings to each other) must have unique `id` values within their parent element. This is the case of `<file>` elements within the root `<xliff>` element. The second level then requires elements grandchildren to have unique identifiers within the entire grandparent scope: the case of `<data>` elements within `<unit>` elements. These first two types are fairly easy to handle, so we will discuss in detail only the third and most complex one.

Unit level in XLIFF is a logical container for a relatively independent portion of content. The extracted text along with codes, represented by XLIFF inline elements, are stored in `<source>` elements, and later the translations will be added to `<target>` elements, both grandchildren of `<unit>`; each pair of the `<source>` and `<target>` siblings is placed within a `<segment>` or an `<ignorable>` element in order to capture segmentation of the content unit.

The constraints and other provisions for the unit level are provided and explained in many different places in the specification, and moreover many of them are conditional to specific governing attribute values.

```
<iso:rule context="xlf:skeleton"
 see="http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-
 os.html#skeleton">
   <iso:assert test="(normalize-space(.)='' and @href) or (not(@href) and
    normalize-space(.)!='')">
      The &lt;skeleton&gt; element must either contain href attribute and
      be empty, or contain text with no href attribute.
   </iso:assert>
</iso:rule>
```

Listing 4: `<skeleton>` solution, Schematron

```
<iso:rule context="xlf:segment[@id] | xlf:ignorable[@id]">
   <iso:let name="id" value="@id"/>
   <iso:let name="f-check"
value="following::xlf:*[ancestor::xlf:unit[@id=$unit-id]][@id=$id]"/>
   <iso:let name="p-check"
value="preceding::xlf:*[ancestor::xlf:unit[@id=$unit-id] and
(ancestor::xlf:segment or ancestor::xlf:ignorable)][@id=$id]"/>
   <iso:let name="d-check" value="descendant::xlf:*[@id=$id]"/>
   <iso:assert test="count($f-check)=0 and count($p-check)=0
and count($d-check)"> id duplication found.
   </iso:assert>
</iso:rule>
```

Listing 5: `id` uniqueness, first step, Schematron

The most general, though, is that any `id` attribute must be unique among all `<segment>`, `<ignorable>`, `<mrk>`, `<sm>`, `<pc>`, `<sc>`, `<ec>`, or `<ph>` elements within the enclosing `<unit>` element. But, the inline elements inside `<target>` must use the duplicate `id` values of their corresponding inline elements in the `<source>` sibling, as long as such corresponding elements do exist, given the conditional logic of specific governing attributes.

The trickiest validity conditions to check occur here. The translated text of a content unit may be presented in a different order, compared to the original text. This means that the `<target>` sibling content does not need to logically or linguistically correspond to the content of its <source> sibling, provided that a different order has been explicitly specifies through the `order` attribute that is optional at the `<target>` element.

The `order` attribute defines the actual position of its `<target>` element in the sequence of the unit content. Therefore the corresponding inline elements must be searched for inside the relevant `<source>` that can be a "cousin" (same grandparent, different parent) of the `<target>`. The value of the `order` attribute is a positive integer between 1 and N, where N is the number of `<segment>` and `<ignorable>` elements within the unit. Finally, inline elements that have their `canDelete` attribute set to 'no', must appear in the corresponding `<target>`, but other elements may be deleted, because the default value of the `canDelete` attribute is 'yes'.

As the first step for this task, we will check the outermost elements of the unit; `<segment>` and `<ignorable>`. The `id` is optional for these elements, so only those with `id` attributes will be selected and checked for duplication within the unit. Listing 5 demonstrates this step.

Here, the variables pick the elements which follow the node but still within the same `<unit>`, its preceding elements and all its descendants. There must not be any node matching this pattern.

As the second step, all inline elements inside `<source>` elements will be checked for duplication among themselves, which is shown in the Listing 6.

For the demonstration purposes, the rules were shortened for inclusion in this paper; the actual rules provided on the XLIFF TC SVN repository provide detailed error messages that are rigorously based on the normative statements of the specification.

From now on (for the third ID-uniqueness validation step), we are going to assume that the validated XLIFF instance is in the final stage of validation (going to merge back into the original format as the next workflow stage), which means that all rules that apply to the `<target>` element are enforced, while this element is strictly speaking optional (from the specifications point of view) at any other stage.

This third step consists itself of sub-steps. First, we check whether values (explicit and implicit) of the `order` attributes are legal (between 1 and N) and then for their uniqueness using the method explained earlier. Then, all inline elements inside the `<target>` will be verified against all of other inline elements except themselves; they are allowed to have one element duplicating their `id` in the corresponding `<source>`, we will look for them a bit later.

```
<iso:rule context="xlf:source//xlf:ph | xlf:source//xlf:pc |
xlf:source//xlf:mrk | xlf:source//xlf:sc | xlf:source//xlf:ec[@id] |
xlf:source//xlf:sm">
   <iso:let name="id-attirbute" value="@id"/>
   <iso:let name="parent-unit" value="ancestor::xlf:unit/@id"/>
   <iso:let name="ph-check"
value="preceding::xlf:ph[ancestor::xlf:unit[@id=$parent-unit] and
ancestor::xlf:source][@id=$id-attirbute]"/>
   <iso:let name="pc-check"
value="preceding::xlf:pc[ancestor::xlf:unit[@id=$parent-unit] and
ancestor::xlf:source][@id=$id-attirbute]"/>
…
   <iso:let name="descendant-check" value="descendant::xlf:*[@id=$id-
attirbute]"/>
   <iso:assert test=”count($ph-check)=0”>id duplication found!
   </iso:asser>
</iso:rule>
```

Listing 6: Inline `id` uniqueness check, Schematron

```
<iso:rule context="xlf:unit">
   <iso:let name="id" value="@id"/>
   <iso:let name="seg-ig-number" value="count(descendant::xlf:segment |
descendant::xlf:ignorable)"/>
   <iso:let name="invalid" value="descendant::xlf:target[@order>$seg-ig-
number]"/>
   <iso:let name="disordered"
value="descendant::xlf:target[@order!=count(preceding::xlf:segment[ancestor
::xlf:unit[@id=$id]] |
preceding::xlf:ignorable[ancestor::xlf:unit[@id=$id]])+1]/@order"/>



   <iso:let name="tester"
value="descendant::xlf:target[count(preceding::xlf:segment[ancestor::xlf:un
it[@id=$id]]|
preceding::xlf:ignorable[ancestor::xlf:unit[@id=$id]])+1=$disordered][@orde
r]"/>
   <iso:assert test="count($invalid)=0">
      The value used for order attribute of &lt;target&gt; element is
greater than number of &lt;segment&gt; and &lt;ignorable&gt; elements
within the enclosing &lt;unit&gt; element. Number of invalid &lt;target&gt;
elements found: '<iso:value-of select="count($invalid)"/>'
   </iso:assert>
   <iso:assert test="count($tester)=count($disordered)">
      Those &lt;target&gt; elements whos position is taken by order
attribute of other &lt;target&gt; element are missing or do not contain
order attribute.
   </iso:assert>
</iso:rule>
```

Listing 7: `order` constraints described in Schematron

The next sub-step of the third step is to check if the `order` attributes are used correctly within units. If a `<target>` element's natural order in the unit sequence has been overridden by explicitly specifying a different position via the `order` attribute, the `<target>`, which had occurred at the natural position corresponding to the former `<target>` element's explicitly set order, the latter has been "displaced", and therefore has to use its own `order` attribute to specify another available position within the unit. And so forth, until all target segments have explicitly set available positions, or otherwise occur on positions with their natural order not "displaced" by any other explicitly set `order` attributes . Listing 7 demonstrates the rule for validating this.

This rule will first identify those `order` attributes that use an illegal positive integer. The natural positions within the sequence are given by the ordinal numbers designating the positions of the `<source>` elements within the given unit, so the rule searches for `<target>` elements, which are not at the natural position (the `order` attribute may actually match the actual position), and saves the values, then checks for `<target>` elements at those positions that in turn must have an explicitly set `order` attribute. This loop continues until the rule verifies that all of the `<target>` elements are ordered properly or until a violation is identified.

After the corresponding `<source>` elements have been identified, previously described methods are used to match legal inline ID duplications between target and source content.

3. Web Service

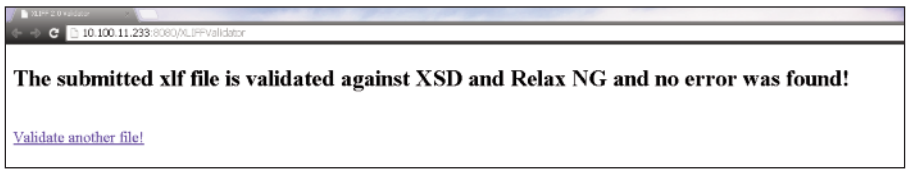
To implement Relax NG and XSD validation in Java, the Java library developed by the Thai Open Source Software Center, *Jing* (Clark, 2008) can be used. This library, as well as other standard Java libraries for XML processing, provides sufficient tools to develop a Java programme, which can validate XML documents against the given schemas; XSD or RNG by user choice. This programme also returns a list of identified errors in case of a failed validation.

Based on the above, a RESTful web service was developed that consumes (receives) an *.xlf file within an HTTP POST request and produces (returns) the validation result (along with the error report, if any errors or warnings occurred) in JSON format. This web service can be called to validate XLIFF instances against the XSD or RNG schemas or both. The functionality of the web service is illustrated with the screenshots below, with an XLIFF file passing the validation and another failing.

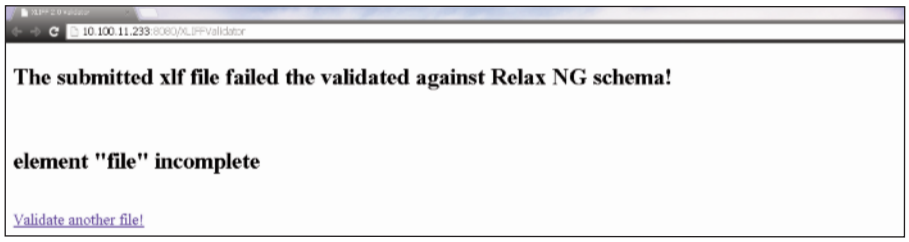
Integrating the Schematron rules is planned as the next step and will give more informative messages in case of a validation failure.

4. Next Steps towards the full advanced validation of XLIFF 2

By using the above demonstrated Schematron methods and techniques, the XLIFF core can be described in full, and hence fully validated with DSDL methods only. The expressivity of Schematron (together with XSD) is sufficient to guarantee the full validation of the XLIFF core and modules including the fragment identifiers. All DSDL based methods of XLIFF validation originating from this research are being proposed as the technical solution (normative



Picture 1: The instance passes the validation



Picture 2: The instance fails the validation

artefacts to become part of the multipart standard) for the *Advanced Validation* feature of XLIFF 2.1 via the OASIS XLIFF TC.

To be able to check arbitrary XLIFF files including modules (and extensions), against all provided schemas (schemas for extensions can be also integrated if provided), constraints and processing requirement, and in order to handle dynamic validation, NVDL (Namespace-based Validation Dispatching Language) needs to be used. This will provide a suitable mapping tool for the advanced validation of different parts of XLIFF document against different namespaces, using mixed DSDL methods and schema languages. Eventually, also NVDL based techniques will be embedded in the validation web service that is being developed based on this research. The web service, however, is not expected to achieve any normative status through the XLIFF TC due to being implementation dependent. Nevertheless, the web service should be made publicly available by the CNGL within the first half of 2015 in order to provide easy access to the advanced validation methods for the standard's end users.

References

Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler E., Yergeau, F. (Eds.) (2008) *XML Version 1.0* (Fifth Edition) [online], available: http://www.w3.org/TR/2008/PER-xml-20080205/ [accessed 15 Mar 2015].

Brown, A. (2010) *Document Schema Definition Languages* [online], available: http://www.dsdl.org/ [accessed 12 Mar 2015].

Clark J. (2008) *A RELAX NG validator in Java* [Computer program], available: http://www.thaiopensource.com/relaxng/jing.html [accessed 1 Nov 2014].

Clark, J., Makoto, M. (Eds.) (2001) *RELAX NG* [online], ISO standard, available: https://www.oasis-open.org/committees/relax-ng/spec-20011203.html [accessed 10 Nov 2014].

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.) (2014) *XLIFF Version 2.0* [online], OASIS Standard. ed, Standard, OASIS, available: http://docs.oasis-open.org/xliff/xliff-core/v2.0/os/xliff-core-v2.0-os.html [accessed 22 Aug 2014].

Delima, N., Gao, S., Glavassevich, M., Noaman, K. (2009) *XML Schema 1.1, Part 2: An introduction to XML Schema 1.1* [online] available: http://www.ibm.com/developerworks/library/x-xml11pt2/ [accessed 20 Mar 2015].

Gao, S., Sperberg-McQueen, C.M., Thompson, H.S. (Eds.) (2012) *XSD 1.1, w3c* [online] available: http://www.w3.org/TR/xmlschema11-1/ [accessed 12 Mar 2015].

Jelliffe, R. (Ed.) (2002), *ISO Schematron*,[online], ISO standard, available: http://www.schematron.com/spec.html [accessed 16 Dec 2014].

Quin, L.R.E., Fawcett, J., Ayers, D. (2012) *Beginning XML, 5th ed*., Indianapolis: John Wiley & Sons.

Raya, M.R. (2012) *XLIFFChecker (Version 1.0-8)* [Computer program], available: http://www.maxprograms.com/products/xliffchecker.html [accessed 13 Mar 2015].

Saadatfar, S., *Relax NG Schema for XLIFF 2.0 core* [Computer program], available: http://tools.oasis-open.org/version-control/browse/wsvn/xliff/branches/Soroush [accessed 1 Feb 2015]

Savourel, Y., *Okapi-lynx XLIFF 2.0 Validation* [Computer program], available: http://okapi-lynx.appspot.com/validation [accessed 18 Mar 2015].

Schnabel, B., *XLIFF 2.0 XMarker Checker* [Computer program], available: http://xmarker.com/node/13 [accessed 13 Mar 2015].

Thompson, H.S., Beech, D., Malony, M., Mendelsohn, N. (Eds.) (2004) *XML Schema*, *w3c* [online], available: http://www.w3.org/TR/xmlschema-1/ [accessed 12 Mar 2015].

Vlist , E. (2003) *RELAX NG*, Sebastopol: O'Reilly & Associates.

w3schools *XML Tutorial* [online], available: http://www.w3schools.com/xml/ [accessed 19 Dec 2014].

Towards a CAT tool agnostic standard for User Activity Data

John Moran, Dave Lewis
Trinity College Dublin,
Dublin, Ireland
Moranj3@cs.tcd.ie**,** Dave.lewis@cs.tcd.ie

Abstract

The dominance of the source word pricing model combined with the fact that most translators work as freelancers has led to a scenario in which until recently most buyers (direct and intermediary) who work with freelancers neither knew nor cared how many words per hour the translators they hire translate. However, this situation is beginning to change. Machine translation has shown that it is possible for translation requesters to impact positively on words per hour productivity. In addition to classical full-sentence MT, advances in various typeahead technologies have resulted in a situation in which a number of options are available to impact positively on a translator's working speed and terminological consistency with previous translations. Finally, evidence is beginning to emerge that productivity gains can be achieved where translators use Automatic Speech Recognition to dictate rather that type the target text. In this paper we will provide a brief overview of these technologies and use cases the impact on translator productivity and describe an architecture to gather translation process data to measure their impact from working translators in a maximally unobtrusive way. We propose an open-standard for User Activity Data in CAT tools (CAT-UAD) so that they can work in any CAT tool that implements this standard and outline a technical architecture to gather such data conveniently and a privacy model that respects translator, intermediary and end-client data sharing concerns and discuss various A/B testing scenarios that can be tested using Segment Level A/B testing.

Keywords: *CAT-UAD, iOmegaT, translator productivity, machine translation post-editing, SLAB testing*

1. Introduction

A number of studies have shown that MT can be a productivity aid for human translators, e.g. Plitt & Masselot (2010), Roukos et al. (2012) and (Moran, Saam, et al. 2014). Often the term *post-editing* is used to describe this use case, but the reality is more complex.

In fact, MT can be presented in a number of ways:

Full-sentence pre-populated MT

This is the typical post-editing scenario in which a target segment is pre-populated using MT. Unless the MT proposal is fit-for-purpose, action is required on the part of the translator to delete or improve it. Usually this can be done quickly using a keystroke combination. A variation on this theme is adaptive MT where post-editing patterns are identified and applied automatically without the need to retrain the underlying language model. A second variation on this theme is the Example Based Machine Translation (EBMT) paradigm where text fragments are glued together with some morphological processing on the edges. Unlike rule-based or statistical MT this is usually carried out in the CAT tool itself. An example of this can be found in DejaVuX' auto-assemble technology (Atril 2015).

MT-as-reference

In this scenario the translator can glance at an MT proposal in a side pane and insert the proposal in the target segment with a keyboard shortcut if useful. However, even if the translator does not consider it useful enough to bootstrap the translation of the segment it may contain terminology that is useful and hence save on research or thinking time. Anecdotally, this workflow works well with Automatic Speech Recognition (ASR). A translator dictates parts of the proposal into the target segment. This may reduce the temptation to compromise on word order and so reduce the impact of MT on style. Unfortunately, very little research has been carried out on this use case.

Type-ahead technologies

As a feature in CAT tools predictive typing has been a common feature in desktop-based CAT tools for some time. For example in Trados Studio (SDL, 2015) the feature is referred to as Auto-suggest and in MemoQ (Kilgray 2015) as the Muse function. Proposals may be statistically generated from bitext using bilingual terminology extraction. It is also possible to cull false positives to create a smaller termbase[1] to reduce the annoyance they cause.

A second approach is to compile a terminology database over a long period of time. This termbase does not necessarily contain terms. It may contain any words or multi-word fragments a translator guesses will arise again. In this case a 30 second investment to save an entry in the termbase may save five minutes of typing or research over a few years. Intuitively it seems likely that this approach saves more time for translators who are specialized than generalists.

A third and more recent approach is to connect to an MT system from the CAT tool for typeahead purposes. Research into type-ahead technologies including Interactive Machine Translation (IMT) dates back to the 1990's (Foster et al. 1997). Proposals appear ahead of the cursor as the translator types and can be accepted using a keystroke. Generally, one problem with IMT is that it is difficult to evaluate in an academic context as traditional automated metrics like BLEU scores (Papineni et al. 2002) do not apply.

Finally, technology is not the only factor governing translator productivity. It is possible to increase translator productivity by requiring that translators ignore stylistic factors in their work and focus on fidelity (light post-editing). Productivity gains can still be achieved in full post-editing where little or no quality degradation is accepted e.g. Plitt & Masselot (2010) and Moran et al. (2014) but they are lower.

Lack of accurate productivity speed ratios can become critical when MT is used as a reason to give a discount (in addition to discounts for translation memory matches). Where a translation requester asks for an unfair discount that overshoots the utility of the MT this may only become obvious after some time. In this case, once a project has been accepted it may be too late for the translator to reverse the discount. However, the translator may decide not to take on future projects that involve MT discounts from that client again (even though they may be fair). Clearly, unfair discounts are not in the interest of any stakeholder. A better approach is that taken at IBM where MT utility is measured over a long period on a large or ongoing project and a discount is negotiated once both parties agree that the utility measure is accurate (Roukos et al. 2012).

2. Automatic Speech Recognition

Though MT and type-ahead technologies can be beneficial from a productivity perspective, it is likely that on average Automatic Speech Recognition (ASR) has a greater impact (outside of light PE contexts). Certainly, financially it is in the interest of the translator. Discounts for post-editing are often requested in a similar manner to discounts for translation memory matches. Where a translator uses dictation software they bring the productivity enhancing technology to the table so discounts are neither requested, nor are they likely to be granted.

Dictation of written translation (or sight translation) is not a new phenomenon. For example, the now infamous Alpac Report (Pierce et al. 1966) described how translators were highly productive when dictating translations to be typed by human transcriptionists. In a 2001 ITI Survey (a UK-based translators union) with 430 respondents approximately 30 used a typist (Aparicio, A., Benis, M., & Cross 2001). More recently ASR software may have begun to replace human typists and to have found new users. In a recent survey (CIOL & ITI 2011 p.4) 10% reported using ASR, of which 94% used Dragon Naturally Speaking (Nuance 2015). Unfortunately, productivity gains reports from ASR are not as well reported as those for MT. In the introduction to an online tutorial Jim Wardell, an experienced professional translator comments how he has been able to double his earnings over his working lifetime using dictation (Wardell, 2014). In a recent survey of ASR use by translators with 47 respondents, the average reported productivity increase was 110.56% (though the median was 35%) (Ciobanu 2014).

However, as there is no means of tracking working

speed over long periods of time in any commercial CAT tool the impact of training and practice are unavailable. For example, techniques used to train interpreters may be useful in sight translation. Also the impact of the recognition quality of the ASR system on translator productivity is unknown. This information gap may also help to explain why there is so little take up of dictation software by translators. It may also explain why there is little or no focus on translators by dictation software publishers who according to Reddy et al. (2009) could improve accuracy by 32% using context derived from the translation task. Finally, it is worth noting that the health gains to be had from this mode of text input (e.g. lower risk of Repetitive Strain Injury) means that even if productivity gains were negligible it would still be worth using the technology.

3. Previous work

A number of means of measuring translation speed exist. Web-based testing platforms that like TAUS DQF (TAUS, 2015) and TransCentre (Denkowski & Lavie 2012) do not provide most of the features found in CAT tools (e.g. a concordance function or translation memory matching) so they can only be used to gather small samples. However, unlike most CAT tools they can provide a Segment Level A/B (SLAB) testing scenario where translation speed in segments without MT (A) are compared to segments with MT (B). An overview of other similar systems and approaches is described in Moran, Saam, et al. (2014).

Our approach is most similar to IBM TM2 (Roukos et al. 2012) which gathers translation process data at the segment level from within a well-featured desktop-based CAT tool.

4. CAT-UAD – A standard format to record User Activity Data

In (Moran, Lewis, et al. 2014) we describe how User Activity Data is gathered in iOmegaT and give an example of the data in XML. In future work we plan to publish a formal specification for CAT-UAD but for the purposes of this paper it can be thought of as a format that records how a translator interacted with a CAT tool during the normal course of their work in an XML format that can be replayed and analysed later (which explains the video camera icon in Figure 1). The XML records details of segment editing sessions as events and context. It also records when a translator returns to a segment (thus taking self-review time into account).

Translators generally use CAT tools for many hours per day and though they may use more than one anecdotal evidence suggests they normally they have a preference for the CAT tool they use most. Although it is likely that most translators do not use all the features that sophisticated desktop-based CAT tools provide in their daily work, nonetheless, anecdotally at least, resistance to using new web-based CAT tools expressed on Internet forums and social media indicates familiarity impacts on productivity.

This is mirrored in our experience. For example, asking a freelance translator familiar with Trados to work in an unfamiliar CAT tool called OmegaT (omegat.org 2015) for a few days to carry out an MT productivity tool is possible but it is not viable for longer periods, e.g. weeks, months or indeed years.

Nonetheless, OmegaT is a well-featured CAT tool as evidenced by the fact that it is commonly used. Download statistics from Sourceforge (the code repository from which it is downloaded) indicate that downloads will soon exceed 10,000 per month and over 2000 users are registered on the user support e-mail list. In its ten-year existence downloads have doubled approximately every four years. However, a recent survey of translators by proz.com (a website for translators) indicated that OmegaT was being used by under 10% of respondents. In contrast various versions of Trados make up the majority of translators with WordFast (Wordfast LLC 2015) and MemoQ in second and third place. Thus, to record and report on the utility of machine translation in terms of translation speed or the effectiveness of training translators in the use of dictation software in a CAT tool agnostic manner, a new data standard is required so that CAT tool developers can log the data in a convenient manner. Also, unlike, for example, the current speed report in MemoQ, time series reports can also be reported at a supra project level (i.e. longitudinally).

Figure 1 shows an overview of how this architecture would look.

In terms of the client-side data collection, OmegaT is shown without the “i” prefix (iOmegaT) as we plan to merge our instrumentation code into the main

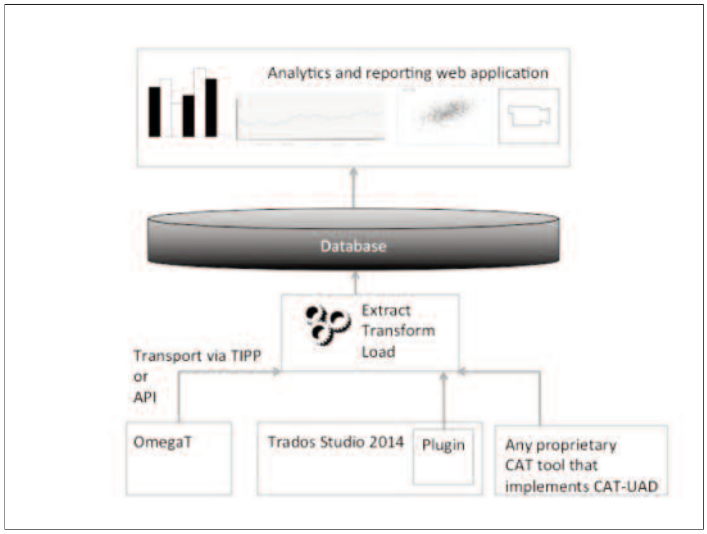


Figure 1: Architectural overview of reporting platform for translation productivity data

OmegaT codebase when the web-based reporting platform has been developed. It is important that a free open-source CAT tool remain central to the platform as this is a maximally flexible option and will make it easier for researchers to carry out reproducible research using SLAB testing in the field, e.g. into various techniques and strategies for interactive MT. Currently, the OmegaT project is the container for the CAT-UAD but this may become a live API. Also, it is possible that it could be added to the TIPP specification (or simply added to the folder structure)[2].

Recent changes to the Trados Studio 2014 Application Programming Interface suggest that a plugin to gather at least some of the data we gather with iOmegaT can be gathered in Trados. However, APIs are not as flexible as open-source applications so it is likely that some A/B testing scenarios that can be implemented in OmegaT will not be possible in Trados Studio.

Finally, conversations with both web-based and desktop based CAT tool publishers suggest there are grounds for cautious optimism that the CAT-UAD standard can be implemented in other proprietary CAT tools once it is formally defined.

In terms of the server-side implementation, the current iOmegaT Translator Productivity Testbench uses console based applications that can be installed locally on a PC. These applications extract, transform and load (ETL) the data gathered from the CAT-UAD files that are stored in the iOmegaT project containers in XML format.

Similarly the web-based reporting platform will be locally installable so all data remains private. In addition a cloud-based option will be available for convenience, albeit with some loss of data privacy.

We have not outlined exact implementation details (e.g. so-called Big Data technologies). However, it is worth noting that recent advances in cloud-computing and data processing provide a number of templates for high volume processing of log data at low cost.

5. Privacy models

Figure 2 shows how the privacy settings could be defined in a CAT tool.

The nature of the translation industry is that translators can be located in almost any jurisdiction so we will use Germany as an example. The recording of User Activity Data in a CAT tool (and in particular translation speed) is a form of workplace monitoring. For translators who are employees pursuant to §87, Subsection 1, No. 1, Works Council Constitution Act (Betriebsverfassungsgesetz - BetrVG) this should be discussed with the relevant works council. For this reason sharing of CAT-UAD should be deactivated by default.

Also, a translator may wish to share translation speed data or other User Activity Data with a third-party

(e.g. a company that provides training and support for dictation software). This can be done without infringing a non-disclosure agreement (NDA) with the agency or end client as Words Per Hour and other data identifying the ASR system, MT system or IMT algorithm being used is unrelated to the text being translated. However an option to share linguistic data is required as in some circumstances, e.g. where the reporting application is hosted with the agency it may be useful to include linguistic data and the NDA is not being infringed. Finally, if a translator wishes to remain anonymous or (more likely) an agency wishes to preserve translator anonymity from a client (a larger translation agency or end buyer) requesting a discount for MT post-editing, it should be possible to do so using an anonymous ID in the Username field.

7. Summary

In this paper we have presented a number of technologies that can impact on translator productivity. We outlined some means by which translation speed can be measured and showed why a dual strategy of adapting an open-source CAT tool (e.g. to test different IMT scenarios) and instrumenting existing proprietary CAT tools to be maximally unobtrusive to the translators who do not use OmegaT regularly. The latter strategy should make it possible to record translation speed data longitudinally to the benefit of computation linguistics researchers, translators, intermediaries and end buyers.

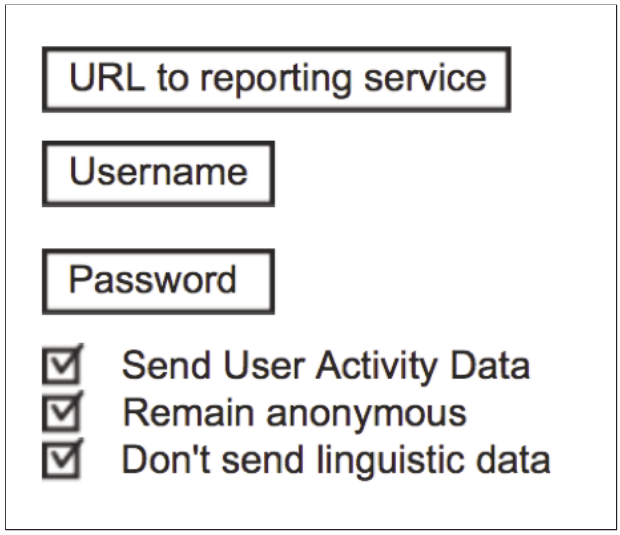


Figure 2: Proposed privacy settings in a CAT tool

6. Future SLAB testing scenarios

In our work to date we have focused on two segment categories, target segments pre-populated with full-sentence MT and empty segments (which we call HT or Human Translation). However, many other SLAB tests are conceivable. For example, dictation with and without MT, two MT systems blind tested against each other (e.g. with two different language models or two different MT providers), two IMT algorithms blind tested against each other, IMT versus HT and even dictating every second segment. Even small improvements should be visible given enough User Activity Data.

References

Atril, 2015. Company website [online], available: http://www.atril.com [accessed 1 Mar 2015].

Aparicio, A., Benis, M., & Cross, G., 2001. ITI 2001 Rates & Salaries Survey.

Ciobanu, D., 2014. Of Dragons and Speech Recognition Wizards and Apprentices. *Tradumàtica*, (12). available: http://revistes.uab.cat/tradumatica/article/view/n12-ciobanu/pdf [accessed 1 Mar 2015].

CIOL & ITI, 2011. *2011 Rates and Salaries Survey for Translators and Interpreters*,

Denkowski, M. & Lavie, A., 2012. TransCenter: Web-Based Translation Research Suite. In *Workshop on Post-Editing Technology and Practice Demo Session*. San Diego, available: http://www.cs.cmu.edu/~mdenkows/transcenter/ [accessed July 9, 2014].

Elming, J., Winther Balling, L. & Carl, M., 2014. Investigating User Behaviour in Post-editing and Translation using the CASMACAT Workbench. In S. O'Brien et al., eds. *Post-editing of Machine Translation*. Newcastle upon Tyne, pp. 147–169.

Foster, G., Isabelle, P. & Plamondon, P., 1997. Target-text mediated interactive machine translation. *Machine Translation*, 12(1-2), pp.175–194.
Kilgray, 2015. Company website [online], available: http://www.kilgray.com [accessed 1 Mar 2015]

Linport, 2015. website [online], available: http://www. ort.org [accessed 1 Mar 2015]

Moran, J., Lewis, D. & Saam, C., 2014. Analysis of Post-editing Data: A Productivity Field Test using an Instrumented CAT Tool. In S. O'Brien et al., eds. *Post-editing of Machine Translation*. Newcastle upon Tyne, pp. 126–146.

Moran, J., Saam, C. & Lewis, D., 2014. Towards desktop-based CAT tool instrumenta-tion. In Third Workshop on Post-Editing Technology and Practice. Vancouver, p. 99.

Nuance, 2015. Company website [online], available: http://www.nuance.com [accessed 1 Mar 2015].

OmegaT, 2015. Open-source project website [online], available: http://www.omegat.org [accessed 1 Mar 2015].

Papineni, K. et al., 2002. BLEU : a Method for Automatic Evaluation of Machine Translation. *Computational Linguistics*, (July), pp.311–318.

Pierce, J.R. et al., 1966. Computers in translation and linguistics (ALPAC report). report 1416. *National Academy of Sciences/National Research Council*.

Plitt, M. & Masselot, F., 2010. A Productivity Test of Statistical Machine Translation Post-Editing in a Typical Localisation Context. *The Prague Bulletin of Mathematical Linguistics*, (93), pp.7–16.

Reddy, A. et al., 2009. Incorporating knowledge of source language text in a system for dictation of document translations. *Proceedings of the twelfth Machine Translation Summit*.

Roukos, S., Ittycheriah, A. & Xu, J.-M., 2012. Document-Specific statistical machine translation for improving human translation productivity. In *Computational Linguistics and Intelligent Text Processing*. Springer, pp. 25–39.

SDL, 2015. Company website [online], available: http://www.sdl.com [accessed 1 Mar 2015].

TAUS, 2015. Dynamic Quality Framework [online], available: https://evaluate.taus.net/evaluate/dqf/dynamic-quality-framework [accessed 1 Mar 2015]

Wardell, J., 2014. Using Dragon NaturallySpeaking Speech Recognition Software to Maximize Speed and Quality in memoQ [online], available: https://www.youtube.com/watch?v=VWQOwBUS-kM [accessed 1 Mar 2015].

Wordfast LLC, 2015. Company website [online], available: http://www.wordfast.net [accessed 1 Mar 2015].

Guidelines for Authors

Localisation Focus
The International Journal of Localisation
Deadline for submissions for VOL 14 Issue 2 is 31 August 2015

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. 14 Issue 1 (2015)

SPECIAL STANDARDS ISSUE 2

CONTENTS

Editorial
David Filip & Dave Lewis 3

Research articles:

Teaching XLIFF to translators and localisers
Lucía Morado Vázquez, Jesús Torres del Rey …….....4

Leveraging NLP Technologies and Linked Open Data to Create Better CAT Tools
Chris Hokamp 14

XLIFF 2.0 and the Evolution of a Standard
Chase Tingley 19

Interoperability of XLIFF 2.0 Glossary Module and TBX-Basic
James Hayes, Sue Ellen Wright, David Filip, Alan Melby, Detlef Reineke 23

Using Semantic Mappings to Manage Heterogeneity in XLIFF Interoperability
Dave Lewis, Rob Brennan, Alan Meehan, Declan O'Sullivan 40

Advanced Validation Techniques for XLIFF 2
Soroush Saadatfar, David Filip 43

Towards a CAT tool agnostic standard for User Activity Data
John Moran, Dave Lewis 51