Exceptional Texts on the Multilingual Web

Gavin Brelstaff
CRS4
Loc. Piscina Manna, Ed. 1
09010 Pula (CA) Sardinia
Italy
gjb @ crs4.it

Francesca Chessa
DUMAS
University of Sassari
Via Roma 151, 07100 (SS)
Italy
fch @ uniss.it

ABSTRACT
Great writers help keep a language efficient for discourse of all kinds. In doing so they produce exceptional texts which may defy Statistical Machine Translation by employing uncommon idiom. Such "turns of phrase" can enter into a Nation's collective memory and form the basis from which compassion and conviction are conveyed during important national discourse. Communities that unite across language barriers have no such robust basis for discourse. Here we describe a Multilingual Web prototype application that promotes appreciation of exceptional texts by non-native readers. The application allows dual column original/translation texts (in Open Office format) to be imported into the translator's browser, to be manually aligned for semantic correspondence, to be aligned with an audio reading, and then saved as HTML5 for subsequent presentation to non-native readers. We hope to provide a new way of experiencing exceptional texts (poetry, here) that transmits their significance without incurring extraneous distraction. We motivate, outline and illustrate our application in action.

Categories and Subject Descriptors
H.5.3 [**Group and Organization Interfaces**]: Web-based interaction. K.4,3 [**Organizational Impacts**]: Computer-supported collaborative work. J.5 [**Arts and Humanities**]: Language translation, Linguistics, Literature. I.7,2 [**Document Preparation**]: Hypertext/hypermedia, .

Keywords
Multilingual Web; ODT, HTML5 Audio; Parallel texts; Text alignment; Browser-based application; Javascript; Literary culture; Poetry

1. MOTIVATION
When Tim Berners-Lee urged us to think about *A Magna Carta for the Web*, in his 2014 TED-Talk, he said he wants a web that is "a really good basis for democracy" [1]. If such a basis is to emerge a *really* profound discourse is needed between the nascent citizens of the web. Any discourse between disparate peoples is difficult, even when equality of rights is an explicit objective [2]. Witness how hard it is to make political progress in the amicable context of the expanded European community. The EC's website welcomes its citizens in 24 different languages and Members of the European Parliament are free to speak in any of those languages [3]. Yet not one politician is capable of addressing the greater European public with anything like compassion or conviction – the language barrier simply prevents most Europeans from receiving the direct impact of any compelling message. This is not a matter to be naively solved by statistical machine translation (SMT) which is designed to deliver the most prevailing/common-place version of any candidate phrase [4]: Exceptional texts are what is required in order to achieve effective impact for important arguments – and our national leaders do adopt, or even coin, an uncommon idiom. This linguistic capacity derives from, and depends on, the literary context in which they are immersed. Idiom, here is not a mere poetical artifice attached to information being transmitted: When idiom is good, it constitutes an efficient *aide-mémoire* for a whole people. For Aristotle, good poetry could instil a sense of civic morality. Great writers create exceptional texts that extend the idiom for their, and future, generations. As Ezra Pound once wrote:

> "Good writers are those who keep the language efficient. ... If a nation's literature declines, the nation atrophies and decays. Your legislator can't legislate for the public good, your commander can't command, your populace (if you be a democratic country) can't instruct its 'representatives', save by language. … The statesman cannot govern, the scientist cannot participate his discoveries, men cannot agree on wise action without language, and all their deeds and conditions are affected by the defects or virtues of idiom." [5]

If we are going to provide a basis for democracy capable of evolving beyond the confines of nation-states serious consideration needs to be given to how the Multilingual Web might convey the "virtues of idiom" between disparate peoples – i.e. those who do not share a common first language – since as Pound put it [5]:

> "The sum of human wisdom is not contained in any one language, and no single language is capable of expressing all forms and degrees of human comprehension."

So a wiser discourse, and a wiser community, might be one involving citizens fortified by the idiom of great writers in *all* its languages – made equally accessible. Here we explore how such accessibility might be fostered via a multilingual community on the web. Exceptional texts merit exceptional measures. Many literary texts are already freely available to readers around the globe – as a result of the growth of the web, e.g. [6]. Yet, understandably, true access gets inhibited by two big show

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2743005

-stoppers: (1) the language barrier that prevents most readers ever getting started; and (2) a manifest public disengagement with literary texts, even when well translated.

1.1 The Language Barrier

Impact-making translations generally require slow human, rather than rapid machine, translation. Time is needed by translators to deliberate on, and compromise between, the various semantic, metrical and melodic aspects. SMT can help produce a rough first draft, but collaboration with native-language speakers is a better starting point. The final translation should involve first language speakers of the receiving language. This is a classical "no centre, all edge" scenario [7], in which there is no conceivable central competence – all relevant literary/linguistic competence resides at the edges, between each pair of languages; Indeed, for many years, translators and authors have enthusiastically worked at these "edges" simply by sending word-processor documents to each other via email – a robust practice not to lightly be disrupted. To use the web-application we describe below they do, however, need to be able to layout the original and completed translation side-by-side on the page in dual columns and save to Open Document format (ODT) [8]. The web-application does not rely on a central server: being entirely browser-based – but it can also be run as service when one wants to compile an archive of the documents which may then be presented to a world of readers – see below.

1.2 Public disengagement

In 1986, W.H. Auden wrote [9] that the public still found poetry "indigestible" while it had "learned how to consume even the greatest fiction as if it were a can of soup". In an era of the filter bubble, TL;DR and so many audio/visual attractions on the web, public disengagement has further widened. Auden was able to console himself that "one must either enter into poetry by a personal encounter or else leave it alone", we aim here to recruit web technology to draw the public into this enabling personal encounter – without resorting to the televisual tendency of treating a poet's words as the voice over of a attention-competing video stream. We hope our approach is more engaging than the a plain, auto-cue style presentations – e.g. [10].

2. PROJECT OVERVIEW

We aim to provide a new reading experience – which in its most striking variety - allows you to listen to a poem being read in its original language while you are shown the parts of the text that are semantically equivalent to what is currently being spoken. Our aim is to achieve this entirely within the modern web browser – via a dedicated web application that provides for both production and presentation. An important part of the production is the alignment of the sentences, phrases and words of the texts – so that semantic equivalence becomes available.

2.1 Production process

Figure 1 gives an overview of production processes in three labeled stages:

A. The Translator selects, or receives, the original language text and, in their own good time, they produce a translation that they are satisfied with.

B. When the translation is complete they can start up a word processor and enter in both texts: laying out the page into two columns – the original text in the left column and the translation in the right. At this point they can apply italic, bold or underline styling to words in the text. They can also chose whether the text should be aligned centrally or not. Next, they select each column in turn and assign to it the appropriate spell checking language. The example in the figure has the right column assigned to English and the left to Sardinian (where some words are found not in the *LibreOffice* dictionary). Finally they save the file in ODT format – which is also an option in *MS Word*.

C. The Translator then starts our Web Application in order to first import the ODT document and check its appearance before continuing. Any variations in font size or family will have been discarded while the columns and the aforementioned rich text styling should have been retained. The language-code symbols should appear at top-left of each column (SR and EN in our example). Now the two parallel texts may be aligned. The first part is semi-automatic: sentences and phrases are aligned following a set of rules that either depend on punctuation marks (known as Lexical context), or by stanza and line breaks (known as Lineal context). The Translator selects between Lexical and Lineal using the drop down menu. Next, alignment is carried out manually at the word level. Here one or more words (not necessarily contiguous) in one column will map on to one or more word in the other column. The translator clicks on these words in turn – each one becoming shown to be selected via a dotted boundary – and then the translator applies a Control-Click to fix (or undo) the alignment. They are also able to attribute a color to each such alignment: green when they consider the equivalence is literal; yellow when it is paraphrastic; and red when it is approximate in any other way. The example in the figure shows that the word "arriving" is a paraphrase of "sa rezida'e". As the Translator moves the cursor over any word on the page that has already been aligned, it becomes highlighted again in synchrony with each other words involved in its alignment. To see all the existing alignments at once the Translator can toggle on the "Notes" button:– it does two things (1) adds a footnote-like superscript to each aligned word whereby aligned words share the same superscript, (2) draws a box around the word in the color it was assigned during alignment. The effect is like that seen in **Figure 2**. Once the Translator is satisfied that all alignments have been properly made they press the "Save" button this saves the aligned document in a HTML5 format – either to disk or, when the Web Application is served from the web, to a location on its server. In either case it can be retrieved for further alignment sessions, if and when desired.

Figure 2 shows how the HTML5 output can be rendered. Since it is self-contained (no Javascript, only inline CSS, no external links) it is very portable. It can be printed to paper, sent by email or, as in our example, presented in a standalone iframe. Here the colored boxes and superscript labels are evident. Note also, an inline CSS trick (:hover::after) allows a simple tool-tip type alignment cue as the cursor passes over the word ("*omine*" in our example) We consider HTML5 a much more versatile archive format compared to the subset of the TEI schema we had previously adopted [13]

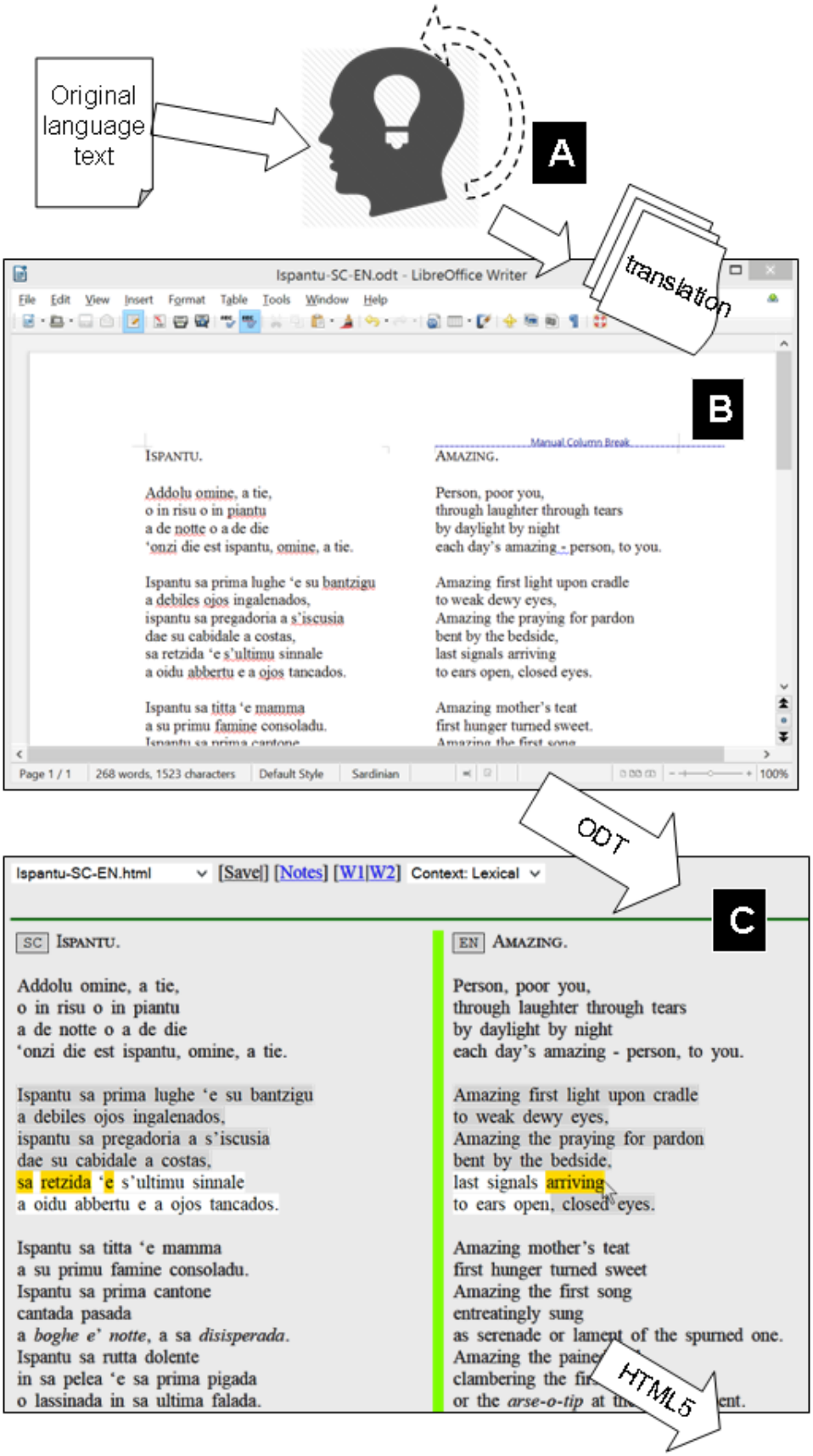

Figure 1. Overview of the production process showing stage A, B, C – see main text.

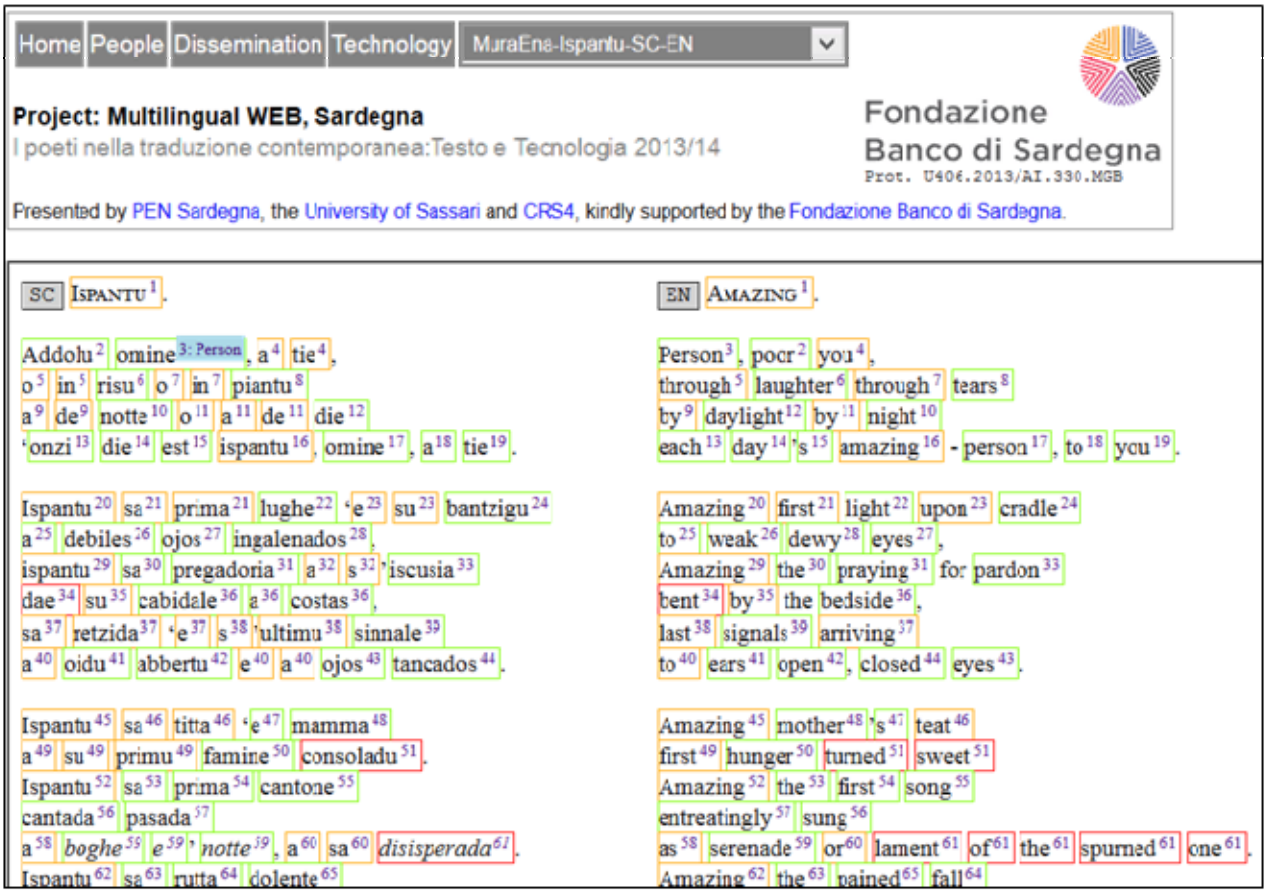

Figure 2. The result of the alignment task is in an HTML5 format that allows it to be visualised outside of the web application, using a standard browser. Aligned words share the same superscript; The color of the boxes indicate the type of equivalence involved – see main text.

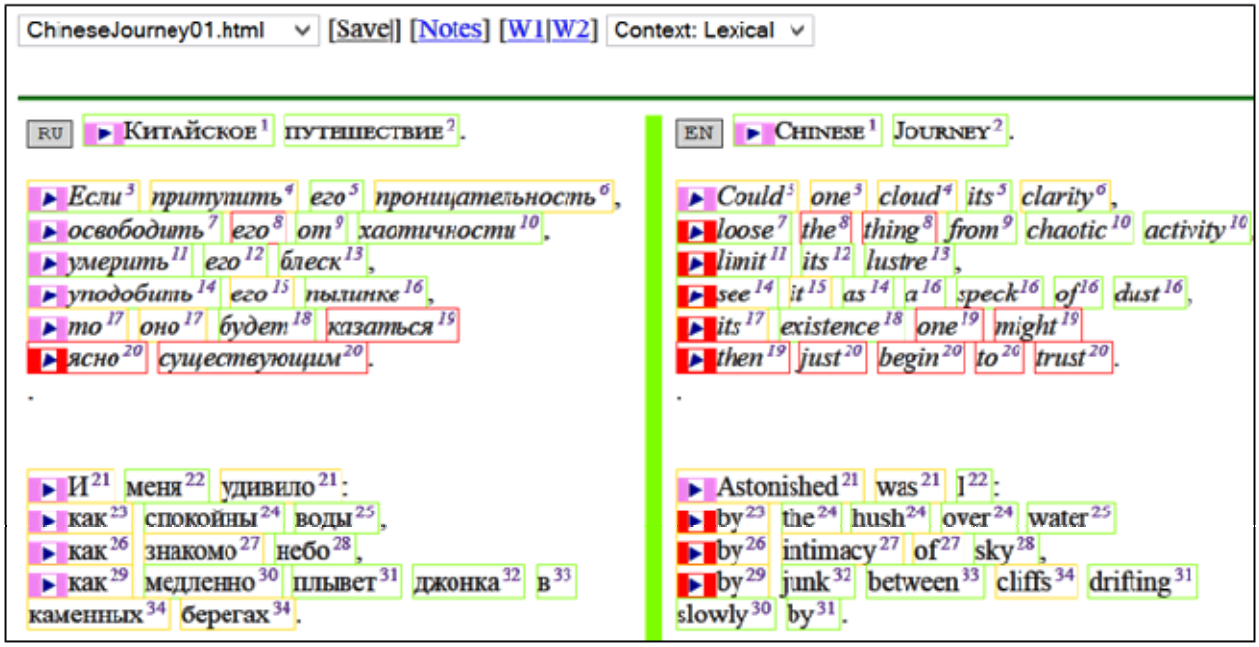

Figure 3. Experiments with automated audio alignment. The symbol [▶] indicates where spaces between words on the page and in the audio tracks have been found to correspond. Work in progress.

2.2 Some technological details

In order to pass from stage B to C in Figure 1 it is necessary to script the browser so that it can read and render ODT. ODT is a zipped archive containing the document content in XML. Our Web Application uses Gildas Lormeau's zip.js JavaScript library [12] to unzip the ODT and then applies the web browser's XSL transform to simultaneously render the document structure as a hierarchy of HTML5 elements and decorate it using CSS assigned to class attributes of those elements. In particular, each single word is represented by a span element and may have an attribute "sup" that indicates its subscript value; attribute "n" that codes the ordinal number of the elements it is joined to by an equivalence alignment.; an attribute "class" that indicates the type of equivalence as "literal", "parap' or "approx". In the case of subsequent audio alignment an addition attribute is given that specifies the time the word starts to be spoken – e.g. start="2.4s".

The jQuery JavaScript library [13] greatly simplifies the scripting of the user interaction required for the alignment task. We have also developed a minimalistic node.js webserver to provide a REST interface for accessing and archiving the ODT and HTML5 documents.

2.3 Audio alignment

Once semantic alignment is complete it is possible to walk through the page automatically – starting at the top of one of the columns and stepping sequentially word-by-word through the whole text. (by pressings the W1 or W2 button seen in Figure 1B) As each word becomes highlighted so do the corresponding words in the other column. This provides a novel reading experience that cannot be reproduced here on paper! It walks through the words at a constant speed – which is rather artificial. It would be more satisfactory to proceed at a real reading pace: which motivates the experimental phase of our project – the alignment of our written text to readings recorded as digital audio tracks – i.e. text-audio synchronization at a word-by-word level, [14]. By scripting HTML5-Audio it is now possible to read and analyze audio data in the browser (MP3/Ogg format). Ideally Voice-to-text technology, such as *Siri* or *Google Now*, would allow us to obtain the time each word starts in the audio track. But such information is not readily available to all-comers on the web. Instead we are experimenting with a simplistic approach: We extract only the duration of the words and the gaps between them. Then we look for corresponding cues in the rendered text. For languages like Chinese this will not work, but in many languages. length of word or its number of vowels correlates somewhat with its spoken duration. Furthermore, gaps tend to be longest after full-stops, stanza-breaks; while commas and line-breaks can also produce notable gaps. We have implemented a heuristic algorithm for matching audio and written text . Thus we can tentatively assign start-times to word elements in our HTML5 documents. The example in Figure 3 shows the symbol [▸] where start-times have been satisfactorily computed. Clicking on that symbol plays the audio starting at that word. We need to refine our algorithm to estimate beginnings of words in the more difficult condition when audio remains at relatively high level even between words. If this proves possible we will be closer to our final goal as stated at the beginning of Section 2.

3. CONCLUSION

The Web Application described here, although still work-in-progress, illustrates that developments in browser technology can be of great practical application in the Multilingual Web field. Our goal remains to build a compelling way to experience and appreciate exceptional texts.

4. ACKNOWLEDGMENTS

This work is kindly supported by the *Fondazione Banco di Sardegna* (Prot. U406.2013/AI.330.MGB). Thanks also go to Gianluigi Zanetti of CRS4 & Prof Nicola Tanda of *Pen Sardegna*.

5. REFERENCES

[1] Berners-Lee, T. 2014., *A Magna Carta for the Web*, TED-Talk, Aug 2014 http://www.ted.com/talks/tim_berners_lee_a_magna_carta_for_the_web/transcript.

[2] Kennedy, H. 2015. *A Modern Magna Carta*, BBC Radio 4, broadcast 2-2-2015 http://www.bbc.co.uk/programmes/b050zy47.

[3] EU, 2014. *European Parliament website*. as available from 25-05-2014. http://doi.acm.org/10.1145/90417.90738 and http://www.europarl.europa.eu/aboutparliament/en/007e69770f/Multilingualism.html

[4] Koehn, P., 2013. *Open Problems in Machine Translation* University of Edinburgh, 25-3-2013 http://www.youtube.com/watch?v=6UVgFjJeFGY

[5] Pound, E. 1934. *ABC of Reading*. George Routledge Limited, London, UK.

[6] Poetry Foundation, 61 West Superior Street, Chicago, IL 60654 USA, http://www.poetryfoundation.org.

[7] Mayfield, R. 2005. *Social-Oriented Architecture*, 2-6-2005, http://ross.typepad.com/blog/2005/06/socialoriented_.html.

[8] OASIS, A. Z. 2006. Open Document Format, Standard ISO/IEC 26300:2006/Amd 1:2012 http://www.opendocumentformat.org. and http://www.libreoffice.org

[9] Auden, W.H. 1986. "A Short Defense of Poetry", *New York Review Of Books,*– 30-1-1986

[10] John Adams, 2015. *Read Along Classics* Channel on YouTube*, since 2012* http://www.youtube.com/user/ReadAlongClassics

[11] Chessa, F., Brelstaff, G. 2011. Going beyond Google Translate? in *Facing Complexity - CHItaly 2011*, Volume 9, page 108--113 – 2011

[12] Lormeau, G. 2014. Zip.js- *A JavaScript library to zip and unzip files*. http://gildas-lormeau.github.io/zip.js

[13] Resig, J. 2006,jQuery: *The Write Less, Do More, JavaScript Library*. http://jquery.com/

[14] Brelstaff, G. Chessa, F. Multilingual Mark-Up of Text-Audio Synchronization at a Word-by-Word Level *W3C Workshop - Making the Multilingual Web Work* 12-3-2013, Rome, Italy http://www.multilingualweb.eu/documents/rome-workshop/rome-program