Localisation of Vector Graphics : A Comparison in Scalable Vector Graphics Localisation with SDL Trados Studio 2015 and Inkscape 0.91

Gutwein, Andrea Melanie

Andrea Melanie Gutwein

Localisation of Vector Graphics: A Comparison in Scalable Vector Graphics Localisation with SDL Trados Studio 2015 and Inkscape 0.91

Supervisor: Lucía Morado Vázquez
Examiners: Silvia Rodríguez Vázquez, Violeta Seretan

Mémoire présenté à la Faculté de traduction et d'interprétation (Département du traitement informatique multilingue (TIM)) pour l'obtention de la Maîtrise universitaire en traduction, mention technologies de la traduction

January 2016

J'affirme avoir pris connaissance des documents d'information et de prévention du plagiat émis par l'Université de Genève et la Faculté de traduction et d'interprétation (notamment la *Directive en matière de plagiat des étudiant-e-s,* le *Règlement d'études de la Faculté de traduction et d'interprétation* ainsi que l'*Aide-mémoire à l'intention des étudiants préparant un mémoire de Ma en traduction*).

J'atteste que ce travail est le fruit d'un travail personnel et a été rédigé de manière autonome.

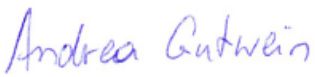
Je déclare que toutes les sources d'information utilisées sont citées de manière complète et précise, y compris les sources sur Internet.

Je suis consciente que le fait de ne pas citer une source ou de ne pas la citer correctement est constitutif de plagiat et que le plagiat est considéré comme une faute grave au sein de l'Université, passible de sanctions.

Au vu de ce qui précède, je déclare sur l'honneur que le présent travail est original.

Nom et prénom : GUTWEIN Andrea Melanie

Lieu / date / signature : Wels, le 21 Décembre 2015



Acknowledgements

First and foremost, I would like to thank my supervisor, Lucía Morado Vázquez, for her guidance, patience and encouraging words throughout this research project.

I would also like to thank my examiners Silvia Rodríguez Vázquez and Violeta Seretan for dedicating their time to the evaluation of my work, and Marianne Starlander for her guidance in the elaboration of an evaluation framework following EAGLES.

The participants in the experiment deserve special thanks for voluntarily giving their time and effort.

I would also like to thank Valeria Siano for allowing me to use a part of her work and for inspiring the evaluation design.

I will be eternally grateful to Verónica, Sarah and Julien, for letting me into their home and offering their support when I most needed it. Thank you so very much.

I am also especially thankful for the help that I received from Martin, Silvia and Annemarie, who have provided me with technical, mathematical and organisational assistance.

I would like to thank my mother, grandmother and grandfather for their love and support, and my extended family: Annemarie, Barbara, Gerald, Karoline, Martin, Sandra, Silvia and Ursula, who prove that blood does not a family make.

Finally, I am dedicating this work to my sister, Daniela, who has taught me to always strive for knowledge. Thank you for everything that you have done for me. You will never be forgotten.

Abstract

Being one of the main types of images, vector graphics play an important role in the field of image localisation. Nevertheless, the subject of vector graphics localisation has not been sufficiently researched to this date. The purpose of this Master's thesis is to determine if CAT tools or image editors (in particular SDL Trados Studio 2015 and Inkscape 0.91) are better suited for the localisation of SVG files by translation students or novice translators with little or no image localisation experience. Three characteristics laid out by the EAGLES evaluation method (functionality, usability and efficiency) and their corresponding sub-characteristics (suitability, accuracy, understandability, learnability, operability, attractiveness and time behaviour) were analysed and evaluated through feature inspection, scenario test and questionnaire. The results of the study suggest that the CAT tool SDL Trados Studio 2015 is better suited for the localisation of Scalable Vector Graphics by users with little or no experience in the field. These findings highlight the advantages of CAT tools in vector graphics localisation and can serve as a starting point for a change in the vector graphics localisation workflow with a greater use of CAT tools in this field.

Table of Contents

List of Figures

List of Tables

1. Introduction

With the rise of the Internet and its growing capabilities, multimedia content is rapidly gaining importance and "users now demand a richer, more engaging presentation than simple text and static diagrams" (Donney et al., 2008). An increase in the localisation of multimedia content, such as static and dynamic graphics, animated files, audio and video or any combinations thereof, can complicate localisation workflows and project management, and lead to additional costs (Mata Pastor, 2009a).

Currently, bitmapped images are the most commonly used image type, but their localisation can be a time-consuming and challenging task. A different image type holds many potential advantages for the field of image localisation that could help cater to the rising demand in localised content.

Vector graphics, such as Scalable Vector Graphics (SVG), which is "a language for describing two-dimensional graphics in XML" (W3C, 2011b), make use of mathematical formulas to create an image, thus offering more flexibility and accessibility than bitmapped graphics. Most importantly from a translator's or localiser's point of view, text and accessibility information are provided in the file's source code and can be easily extracted and edited.

It is possible to use different methods for the translation of text content in Scalable Vector Graphics. The translator or localiser can edit the text with a draw program, such as Inkscape, or translate the text content of the picture with a computer-assisted translation (CAT) tool, such as SDL Trados Studio, when a corresponding filter is created. The translatable text could even be altered in the source code itself, although the information in the code can be quite complex, based on the content of the image, and the translator or localiser runs the risk of accidentally altering or erasing the elements of the file.

Although image editors might seem the obvious choice for the localisation of graphics, because their focus lies on graphical editing, CAT tools offer many features that can lead to a higher quality in translations and allow professionals to work more efficiently. Thus, a greater incorporation of CAT tools into the field of image localisation could yield very promising results.

While there are various methods for the localisation of vector graphics, there is currently no research in this field that determines which is the most suitable practice. This study is motivated by the lack of research at hand and seeks to establish a preliminary study that determines the most suitable Scalable Vector Graphics localisation practice for a certain kind of users.

This research project focuses on evaluating the suitability of CAT tools and image editors for Scalable Vector Graphics localisation by novice translators or localisers with little or no experience in the field of image localisation. In particular, the tools SDL Trados Studio 2015[1] and Inkscape 0.91[2] are analysed in order to determine the most suitable tool for these target users, which will facilitate their work and help them choose a program adapted to their needs.

The study and its findings can serve as a model for similar studies involving different tools and/or target users and a starting point for a change in the workflow in the field of vector graphics localisation.

The manuscript is divided into seven chapters. Chapter one provides a brief introduction to the subject of Scalable Vector Graphics, the possible methods for translating their text content and the research objectives, including the personal motivation for the research and a description of the research questions and the hypothesis. Chapter two is dedicated to an analysis of literature surrounding vector graphics and the Scalable Vector Graphics format, focusing primarily on the differences between vector images and bitmapped images, the advantages of vector graphics and the text content elements that are used in the Scalable Vector Graphics file format. Chapter three describes the research methodology, including the creation of the Scalable Vector Graphics filter in SDL Trados Studio 2015, and the EAGLES evaluation method, which was used for the comparison of the tools. Chapter four lays out the experiment design for the evaluation, along with descriptions of the materials, files and software that were used during the experiment and of the participants' profiles. Chapter five is dedicated to the presentation of the results of the EAGLES evaluation, including a criteria weighting. A

[1] For more information on SDL Trados Studio 2015 and to purchase the tool, please visit: http://www.sdl.com/cxc/language/translation-productivity/trados-studio/

[2] For more information and to download the tool, please visit: https://inkscape.org/en/

summary of the results, as well as the conclusions, limitations and future applications of the research project are provided in chapter six.

Throughout this research project, the acronym SVG is used to refer to the 'Scalable Vector Graphics' file format.

The terms 'novice translator' or 'novice localiser' are used in this research project to denominate translation students or former students, who have graduated from a Master of Arts in Translation no more than two years ago.

The participants of the experiment are referred to as 'P' following their participant number (e.g. P1).

The names of SVG elements are written inside tag pairs (e.g. <text>) and attributes are written with single quotation marks (e.g. 'systemLanguage'), except in quotations, where the original formatting is preserved. In cases where the use of tags or quotation marks might lead to confusion, e.g. when the XPath rules of the SVG filter for SDL Trados Studio 2015 are explained (cf. section 3.3.2.3), a bold type is used to highlight text.

Any content taken from SDL websites, notably www.sdl.com and producthelp.sdl.com, and figures displaying the program SDL Trados Studio 2015 or any part of it, belong to SDL PLC and are used with written consent. All intellectual property rights are the sole and exclusive property of SDL PLC.

1.2. Research Objective

This section describes the personal motivation for the choice of the subject of this research project (section 1.2.1), the research questions (section 1.2.2), and a hypothesis for the outcome of the study (section 1.2.3).

1.2.1 Personal Motivation

The subject of image localisation is very important in the field of localisation, since graphics appear in almost every medium that requires translation (e.g. websites, video games, software, advertisement, etc.). Therefore, I consider the study and determination of the best practice in vector graphics localisation of utmost importance for the daily work of a translator and/or localiser.

I chose this subject over others, since the findings of this study can determine the quickest and most efficient SVG translation process, making the work of novice translators or localisers easier and less time-consuming. It will also be an aid to recently graduated students at the start of their careers, who need to choose a program for SVG localisation that is adapted to their specific needs.

Furthermore, I will be able to apply the results of this research project to my later career, because it will allow me to gather experience in the field of image localisation and in the creation of XML based filters.

1.2.2 Research Questions

There is a need for research in the field of vector graphics localisation. A study of the strengths and weaknesses of CAT tools and draw programs can determine the best practice for the localisation of vector graphics. The findings can render the localisation process more efficient and point out if there are any shortcomings or a need for optimisation of draw image localisation in CAT tools.

This research project seeks an answer to the following question:

Are CAT tools or image editors better suited for the localisation of SVG images by translation students or novice translators or localisers with little or no image localisation experience?

For this purpose, one of each type of tools was chosen for the study; in particular, SDL Trados Studio 2015 to represent CAT tools and Inkscape 0.91 to represent image editors.

To obtain an answer, it is necessary to examine the following sub-question: Which are the advantages and shortcomings of SDL Trados Studio 2015 and Inkscape 0.91 in the localisation of Scalable Vector Graphics files?

1.2.3 Hypothesis

A priori, the expected outcome of the evaluation is that SDL Trados Studio 2015 will be better suited for the localisation of SVG files by novice translators and localisers with little or no experience in image localisation than Inkscape 0.91. This suggestion is based on the assumption that the translation process will be easier and more efficient using a CAT tool than a vector graphics editor, since the primary function of CAT tools is to aid in the

translation process, whereas graphics editors are not specifically adapted to the needs of translators or localisers.

For instance, the adaptation of CAT tools to the needs of translators or localisers can be observed in the extraction of translatable content, which allows the user to focus solely on the translation task, without the risk of modifying elements of the graphic that are not supposed to be altered during the translation process. Another beneficial aspect is that the CAT tool SDL Trados Studio 2015 facilitates the translation with useful features, such as auto-propagation, spell-checking and ways to ensure consistency throughout the translation.

Even though vector graphics editors, such as Inkscape 0.91, offer very different advantageous aspects for the translation of SVG files (such as the possibility to view the whole graphic and to modify translatable content with the context and surrounding elements in mind), it is hypothesised that, ultimately, the advantages of a CAT tool outweigh the advantages of a graphics editor in the localisation of SVG files.

2. Literature Review

This chapter provides the reader with information on the Scalable Vector Graphics format. It includes a comparison of the different image types (section 2.1), a description of vector graphics in localisation, including its advantages over bitmapped graphics (section 2.2), a description of the basic concepts of the SVG format (section 2.3), and an overview of the text content elements used in the Scalable Vector Graphics format (section 2.4).

2.1 Types of Graphics

2-D computer graphics can be grouped into two main types: bitmapped and vector drawn (Savage and Vogel, 2013). This section provides an overview of the basic characteristics of these two types.

2.1.1 Bitmapped Graphics

Bitmapped graphics (also called 'raster graphics'[3]) are created as a pattern of elements, which are called pixels ('picture elements'). Pixels are "usually defined as small squares" and each one is "assigned a code that designates its color". A 'matrix' or 'array' is used, serving as a "mapping of the locations and colors of the pixels that compose the image" ('bitmapping'). (Savage and Vogel, 2013, pp. 125-126)

Bitmapped graphics can be divided into three main types: line art, grayscale images and color bitmapped images. Some commonly used bitmapped and metafile formats are: PICT, BMP, TIFF, JPEG, GIF, and PNG. (Savage and Vogel, 2013)

The quality of a bitmapped image is determined by "the density of the pixels, or **spatial resolution**, and the number of different colors each pixel can display, or **color resolution**". A higher spatial resolution results in better quality images, because they are "made up of many very small, closely packed pixels". (Savage and Vogel, 2013, p.128)

When the dimensions of bitmapped graphics are altered (e.g. when an image is rendered on a higher-resolution device), the image is often resampled, meaning that samples (one sample represents a pixel) are either added (upsampling) or reduced (downsampling) in the image to increase or decrease the spatial resolution of the file

[3] http://techterms.com/definition/bitmap

(Savage and Vogel, 2013). “Upsampling is used to enlarge the physical dimensions of an image on a given device”, but it “usually degrades the appearance of a graphic”, since “the computer has to guess the color values for the additional pixels” (Savage and Vogel, 2013, pp. 129-130). In other words, when a bitmapped graphic is enlarged to fit a bigger screen, the quality of the image decreases. In the same manner, when resizing a graphic in image editing programs (ibid.) or when zooming into the image, the same decrease in quality can be observed.

Given that this type of images is composed of pixels, any text that is included in a bitmapped image is not made up by alphanumeric characters and symbols, but by little dots that take on the form of letters, numbers and signs, when they are arranged in a certain way (Mata Pastor, 2009a).

2.1.2 Vector Graphics

Vector graphics (also ‘vector-drawn graphics’) are composed of vectors, which are “line[s] with a particular length, curvature, and direction”. These lines are “mathematically defined to form shapes such as rectangles, circles, triangles and other polygons”, which are combined to make up a vector graphic. “In vector-drawn graphics, the computer is given a set of commands that it executes to draw the image”. Some general-purpose vector formats are EPS, PDF and SVG (Savage and Vogel, 2013, p.137-139).

Draw programs are used to create vector graphics, which can range from very simple images to complex images or even art works (ibid.). The Scalable Vector Graphics format also allows the user to create an image by manually coding it in a notepad tool, including the definition of the commands needed to draw the image, but this way is not efficient for complex images.

The file size of vector graphics is usually much smaller compared to bitmapped graphics.

Another difference to bitmapped images is that vector graphics are ‘device-independent’, meaning that “the same file can be used with different devices without altering the size of the image”. The commands for drawing contained in a vector graphic are simply applied to the output resolutions of the different device, thus preserving the “original dimensions of the image”. This concept also allows for “smooth scaling” in vector

images without a decrease in quality, since they "are enlarged by changing the parameters of their component shapes". (Savage and Vogel, 2013, p.138)

It is possible to create vector graphics from bitmapped images through 'autotracing'. In this process, the bitmapped image is analysed for shapes, which are then mathematically defined in order to produce a vector graphic. This can be useful for simple images, but with complex images, the appearance of the bitmapped file is not always preserved (ibid.). When images with text content are autotraced, the text is also recognised as a shape and not as textual content, thus it cannot be easily edited or translated after the conversion.

However, vector graphics can effortlessly be converted to bitmapped images through 'rasterising', which "rapidly samples the vector image and saves it in bitmapped form" (Savage and Vogel, 2013, p.138). This can usually be done with a vector graphics editor, by exporting the vector graphic to a bitmapped format. It has to be taken into consideration, that all modern displays are raster-oriented, which means that all images are rasterised eventually, but there is a difference as to where the graphic is rasterised (W3C, 2011a). Bitmapped images are rasterised on the server, while vector graphics are rasterised on the client side (ibid.).

The main advantages of vector graphics over bitmapped graphics are the "smooth scaling and reshaping", the "ease of editing objects" due to layers and grouping or ungrouping of objects, the "low file sizes" and their "device-independent image resolution". The disadvantages are the "less detailed representation of complex contone images", the lack of "photo-editing capability", and "limited artistic control". (Savage and Vogel, 2013, p.140)

While vector graphics are ideal for images that are composed of "distinct lines and curves" and are especially used in fields "where accurate measurement and the ability to zoom in on a drawing to see details are essential", bitmapped graphics are "more appropriate for use with photographs" (Eisenberg and Bellamy-Royds, 2014, p.2-3).

Since vector graphics offer the ability of accurate drawing and measurements, they seem to be especially important and used in domains such as mathematics, physics or any other field that works with graphical representations of lines and curves and that does not use complex images.

Furthermore, because the SVG file format is an XML application, it "cooperates with other XML applications" that can be used to complement the functions of SVG. One example would be the generation of graphs in a mathematics textbook with SVG and other XML applications to include explanatory text or to describe equations. (Eisenberg and Bellamy-Royds, 2014, p.6)

Despite the advantages of SVG over bitmapped graphics formats, Chen and Harper observe that SVG has shown a poor adoption rate since it became a W3C recommendation in 2003 (Chen and Harper, 2008). This slow trend can probably be attributed to the fact that the tools for the creation of raster images are "widespread and generally easier to use than vector-based tools" and are "widely available", which is why, until the publication of the SVG format, web browsers only supported raster images (Eisenberg and Bellamy-Royds, 2014, p.3).

A significant increase in adoption rates is expected after the release of the format's second edition, based on other graphical formats trends (Chen and Harper, 2008). This prediction and the enormous advantages that vector graphics and the SVG format in particular present for certain domains, make it a format that will eventually become increasingly relevant for the work of localisers.

2.2 Vector Graphics and Localisation

Vector graphics, and SVG in particular, offer many advantages over bitmapped graphics from a translator's or localiser's point of view. Mata Pastor (2009a) argues that, to a great extent, the challenges in image localisation stem from the fact that IT products or websites primarily use bitmapped images. Their modification is often laborious and expensive. These bitmapped images are frequently generated from vector graphics, but the original vector graphic is often not available.

The localisation process changes drastically according to the image type. With bitmapped images containing embedded text, the localiser usually needs to extract the text manually or automatically for the translation with a CAT tool or any other tool. Some formats provide bitmapped images with layers, where the text can be isolated and translated. Nevertheless, these formats cannot be included in software components or websites and are converted or exported to a plain pixel graphic, which is more complex to edit. The localisation process is very different for vector graphics. Since localisable text

is located inside the file in the form of editable alphanumeric characters, it is relatively easy to localise the text directly or to extract it automatically. However, the *de facto* norm is still the use of embedded text in icons, buttons and other raster graphics elements in software interfaces or websites, even though it supposes an additional workload and, consequently, a higher cost. (Mata Pastor, 2009a)

Some vector graphics formats, including SVG, allow the user to create animations or interactive images. The localisation of such files follows the same principle as the static vector graphics, since the text is found in the same text content elements or in scripts that might be embedded. (Mata Pastor, 2009b)

The SVG format also presents advantages concerning globalisation. The format "handles both vector- and raster-based images, can include font information, saves text in an editable format, can retain font attributes such as bold, italic, and so on [and] it can be used as a source file for translation, a Web graphic for HTML or XHTML documentation, and a print graphic for PDF documentation". The SVG file format also "answers the call for better accessibility experiences for visually impaired users", given the capability of "zooming in and out of the images" without a loss in quality, of embedding "metadata such as title and image data" and of "enabling links for text found within the image to a dedicated HTML file for additional description". (Donney et al., 2008)

The wide range of SVG accessibility features make vector graphics potentially more accessible than bitmapped graphics: "Unlike many of the newer HTML5 technologies, SVG actually has a fairly large and well thought out accessibility component" (Andersen, 2010). This aspect and the fact that web browser SVG support is constantly growing (ibid.) will assure a more widespread use of SVG in the future, which means that translators and/or localisers will have to adapt to the new circumstances and learn how to use and manipulate vector graphics.

Donney et al. (2008) further suggest that the SVG file format "might prove adequate to reinvent the graphics translation process, bringing many benefits including improved cost-effectiveness […]". They also point out that the "potential benefits for the translation process […] are significant enough to make the topic well worth researching".

The potential of the Scalable Vector Graphics format has been apparent for some time now, as evidenced by Ultan Ó Broin (2003):

> "Scalable Vector Graphics is one of the most promising developments for image localization. The advent of SVG will remove much of the pain, especially for relatively simple graphics and anyone who relies on Web-based technology for product deployment."
>
> (Ó Broin, 2003)

Mata Pastor (2009b) highlights that flexible and open formats, such as vector graphics formats or other XML applications, are essential to the future progress of graphics design and illustration tools, also in what concerns the localisation sector. The reason for this is that they do not limit the possibilities of graphics designers, and at the same time facilitate the processing of text in multimedia content. He further points out that the success of these standards depends on their widespread adoption and implementation by the agents of the localisation sector and, notably, the developers of CAT tools[4] and localisation tools.

Much of the current literature on SVG pays particular attention to the file type in general and its use in software, web design and for accessibility purposes. To the best of our knowledge, the practical applications of SVG in the field of localisation are widely unexplored in the scientific community. This state of research therefore presents the opportunity to dive into a generally undiscovered subject and to provide new and valuable scientific information.

2.3 The format SVG (Scalable Vector Graphics)[5]

The World Wide Web Consortium (W3C) started developing the SVG format in 1998, two years after the release of general requirements for vector graphics. The SVG Working Group was formed when "six competing submissions to the W3C in the Web vector graphics area" (W3C, 2010) were made (Web Schematics, PGML, VML, Hyper Graphics

[4] It is interesting to observe that, as of this date, the market leader in CAT tools, SDL Trados Studio 2015, has not yet added the SVG format to the list of supported file types, making it necessary to create a custom filter for the translation of SVG files.

[5] This section focuses on the main SVG concepts that are necessary to acquire a basic understanding of the format.

Markup Language, WebCGM and DrawML). After the review of these submissions, it was decided to create a new language that would be "informed by, but not really based on, any of the submissions" (ibid.). On 4 September 2001[6], SVG 1.0 was released as a W3C Recommendation. The current Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second Edition)[7] was released on 16 August 2011.

Three types of graphical objects can be used in SVG: vector graphic shapes, such as paths, images and text. They can be "grouped, styled, transformed and composited into previously rendered objects". It is possible to create interactive and dynamic drawings using SVG. (W3C, 2011b)

Scalability is one of the main features of vector graphics, as mentioned in section 2.1.2. SVG is scalable in two senses of the word: it is "not limited to a single, fixed, pixel size" and it "can grow to a large number of files, a large number of users, [and] a wide variety of applications". The scalability allows SVG graphics to be resized by users or according to display resolutions, without a deterioration of the image's quality. (W3C, 2011a)

Another aspect of SVG scalability is that "SVG content can be a stand-alone graphic or can be referenced or included inside other SVG graphics", making complex illustrations by several people possible (ibid.). Like any vector graphic, SVG "contains geometric objects such as lines and curves", which makes the format more flexible in comparison with bitmapped graphics. In SVG and in vector graphics in general, bitmapped graphics can be integrated and combined with vector information. Also, the fact that the SVG format is written in XML gives it a "sound basis for internationalization, powerful structuring capability, [and] an object model", among other advantages. (ibid.)

2.4 Text Content Elements Used in SVG

This section provides an overview of the primary functions of SVG elements containing text, since they are the main focal point for translators or localisers who work with SVG. Only the most relevant attributes are mentioned in this work. For more detailed information, the W3C Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second

[6] http://www.w3.org/TR/2001/REC-SVG-20010904/

[7] http://www.w3.org/TR/SVG/

Edition)[8] can be consulted. It provides a complete description of all the attributes and properties that may be used with SVG 1.1 text content elements and their child elements (W3C, 2011c), as well as an element index (W3C, 2011d). Another useful source for the consultation of SVG elements is the Mozilla Developer Network (2013) with a list of elements including their categorisation by function.

The SVG file format offers different elements to embed text in the source code, which are categorised as text content elements or text content child elements. "A text content element is an SVG element that causes a text string to be rendered onto the canvas" and "[a] text content child element is a text content element that is allowed as a descendant of another text content element" (W3C, 2011b).

The following SVG 1.1 elements can render text (ibid.) :

- Text content elements: <altGlyph>, <textPath>, <text>, <tref>, <tspan>
- Text content child elements: <altGlyph>, <textPath>, <tref>, <tspan>

The Mozilla Developer Network (2013) also includes the elements <altGlyphDef>, <altGlyphItem>, <glyph>, <glyphRef> in the list of text content elements.

The elements below are explained starting with the primary text content element, following a decreasing order of importance. Because of its significance in localisation, a brief explanation of the <switch> element and the descriptive elements <desc> and <title> is included in this section, even though strictly speaking they are not text content elements. The order was determined following the order of appearance of these elements in the 'Text' chapter of the SVG specification[9].

2.4.1 <text>

The <text> element is the "primary method of adding text" (Eisenberg and Bellamy-Royds, 2014, p.125). It "defines a graphics element consisting of text" and it is possible to indicate "the writing direction, font specification and painting attributes which describe how exactly to render the characters" through attributes and properties (W3C, 2011c). In its

[8] http://www.w3.org/TR/SVG/

[9] http://www.w3.org/TR/SVG/text.html

simplest form, it only requires the x and y attributes, which define the starting point for the "baseline of the first character" (Eisenberg and Bellamy-Royds, 2014, p.126).

W3C (2011c) points out that a <text> element can only produce a single string of text and does not allow for automatic line breaking or word wrapping. Multiple lines can be created with a <text> element for each line of text or with a <tspan> child element. Another way is to "[e]xpress the text to be rendered in another XML namespace such as XHTML [...] embedded inline within a 'foreignObject' element", but the W3C Recommendation notes that "the exact semantics of this approach are not completely defined at this time". (ibid.)

The following example from the W3C Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second Edition) (ibid.) shown in Figure 1 features a <text> element, as well as specifications regarding the position of the text (with the 'x' and 'y' coordinates), the font and the text colour.

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
  "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="10cm" height="3cm" viewBox="0 0 1000 300"
     xmlns="http://www.w3.org/2000/svg" version="1.1">
  <desc>Example text01 - 'Hello, out there' in blue</desc>

  <text x="250" y="150"
        font-family="Verdana" font-size="55" fill="blue" >
    Hello, out there
  </text>

  <!-- Show outline of canvas using 'rect' element -->
  <rect x="1" y="1" width="998" height="298"
        fill="none" stroke="blue" stroke-width="2" />
</svg>
```

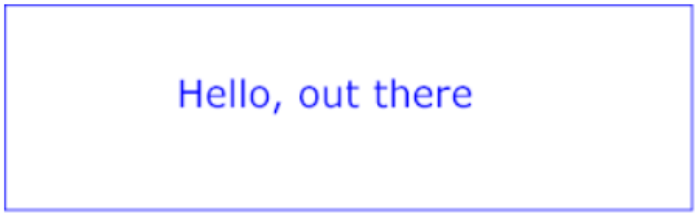


Figure 1. Example of an SVG file containing a <text> element and its final graphical representation[10]

[10] Copyright © [16 August 2011] World Wide Web Consortium, (MIT, ERCIM, Keio, Beihang). http://www.w3.org/Consortium/Legal/2015/doc-license

2.4.2 <tspan>

A <tspan> element can be used within a <text> element to adjust “text and font properties and the current text position” with “absolute or relative coordinate values” (W3C, 2011c).

Eisenberg and Bellamy-Royds (2014, pp.129-132) recommend the use of text span elements, because they “group related lines”, allowing them to be selected as a unit and adding structure to the document. The authors use the example of a sentence with italic, normal and bold text to highlight the utility of <tspan> elements for a “string with varying text attributes”. In contrast to the <text> element, <tspan> elements “remember the text position”, thus facilitating the implementation of style changes, such as “font size, color, weight, etc.” or the “positioning of individual letters”, such as “superscripts or subscripts”. (ibid.)

The example in Figure 2 illustrates the <tspan> element as used in the W3C Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second Edition) (W3C, 2011c). The <tspan> is used inside a <text> element to change the font-weight and colour of only one word in a whole sentence.

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
  "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="10cm" height="3cm" viewBox="0 0 1000 300"
     xmlns="http://www.w3.org/2000/svg" version="1.1">
  <desc>Example tspan01 - using tspan to change visual attributes</desc>

  <g font-family="Verdana" font-size="45" >
    <text x="200" y="150" fill="blue" >
      You are
        <tspan font-weight="bold" fill="red" >not</tspan>
      a banana.
    </text>
  </g>

  <!-- Show outline of canvas using 'rect' element -->
  <rect x="1" y="1" width="998" height="298"
        fill="none" stroke="blue" stroke-width="2" />
</svg>
```

Figure 2. Example of an SVG file containing a <tspan> element and its final graphical representation[11]

[11] Copyright © [16 August 2011] World Wide Web Consortium, (MIT, ERCIM, Keio, Beihang). http://www.w3.org/Consortium/Legal/2015/doc-license

2.4.3 <tref>

The <tref> element is used to reference character data content (W3C, 2011c), which has previously been defined within a <defs> element ("a container element for referenced elements" (W3C, 2011e)).

The W3C Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second Edition) provides a short example and the accompanying explanation (W3C, 2011c):

> Figure 3 "shows how to use character data from a different element as the character data for a given 'tspan' element. The first 'text' element (with id="ReferencedText") will not draw because it is part of a 'defs' element. The second 'text' element draws the string "Inline character data". The third 'text' element draws the string "Reference character data" because it includes a 'tref' element which is a reference to element "ReferencedText", and that element's character data is "Referenced character data"".
>
> (W3C, 2011c)

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
  "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="10cm" height="3cm" viewBox="0 0 1000 300" version="1.1"
     xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
  <defs>
    <text id="ReferencedText">
      Referenced character data
    </text>
  </defs>
  <desc>Example tref01 - inline vs reference text content</desc>

  <text x="100" y="100" font-size="45" fill="blue" >
    Inline character data
  </text>
  <text x="100" y="200" font-size="45" fill="red" >
    <tref xlink:href="#ReferencedText"/>
  </text>

  <!-- Show outline of canvas using 'rect' element -->
  <rect x="1" y="1" width="998" height="298"
        fill="none" stroke="blue" stroke-width="2" />
</svg>
```

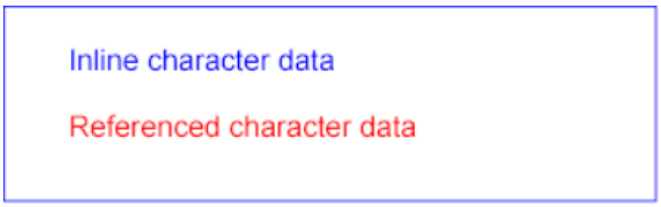


Figure 3. Example of an SVG file containing a <tref> element and its final graphical representation[12]

2.4.4 <textPath>

"In addition to text drawn in a straight line, SVG also includes the ability to place text along the shape of a 'path' element" (W3C, 2011c). The path needs to be defined and can then be accessed by the <textPath> using an 'xlink:href' attribute (Eisenberg and Bellamy-Royds, 2014).

The example shown in Figure 4, taken from the W3C Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second Edition) (W3C, 2011c), shows text that follows a cubic Bézier curve. It displays a path (called 'MyPath') that has been predefined within a <defs> element and is later referenced in the <textPath> element.

[12] Copyright © [16 August 2011] World Wide Web Consortium, (MIT, ERCIM, Keio, Beihang). http://www.w3.org/Consortium/Legal/2015/doc-license

Usually, the line of the path is not displayed if it is defined in a <defs> element. This example makes the line visible with the <use> element (Eisenberg and Bellamy-Royds, 2014, p.138). It is possible to create several kinds of paths using different commands. The example below uses the command 'C' (curveto'), which can be used to display a cubic Bézier curve. A full description of all the commands and the kinds of paths they create is provided by the W3C (2011f).

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
  "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="12cm" height="3.6cm" viewBox="0 0 1000 300" version="1.1"
     xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
  <defs>
    <path id="MyPath"
          d="M 100 200
             C 200 100 300   0 400 100
             C 500 200 600 300 700 200
             C 800 100 900 100 900 100" />
  </defs>
  <desc>Example toap01 - simple text on a path</desc>

  <use xlink:href="#MyPath" fill="none" stroke="red"  />
  <text font-family="Verdana" font-size="42.5" fill="blue" >
    <textPath xlink:href="#MyPath">
      We go up, then we go down, then up again
    </textPath>
  </text>

  <!-- Show outline of canvas using 'rect' element -->
  <rect x="1" y="1" width="998" height="298"
        fill="none" stroke="blue" stroke-width="2" />
</svg>
```

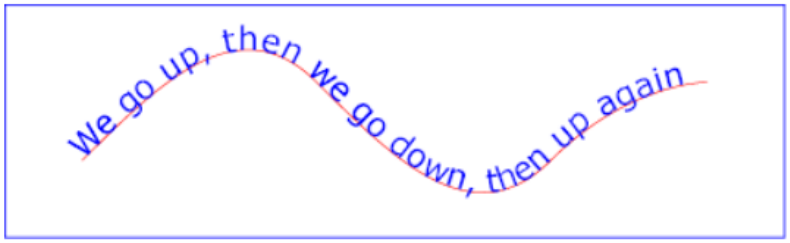


Figure 4. Example of an SVG file containing a <textPath> element and its final graphical representation[13]

[13] Copyright © [16 August 2011] World Wide Web Consortium, (MIT, ERCIM, Keio, Beihang). http://www.w3.org/Consortium/Legal/2015/doc-license

2.4.5 <altGlyph>

The <altGlyph> element is used to control glyphs ("the visible representation of a character or characters" (Eisenberg and Bellamy-Royds, 2014)) in "situations such as ligatures, special-purpose fonts (e.g., a font for music symbols) or alternate glyphs for Asian text strings where it is required that a different set of glyphs is used than the glyph(s) which normally corresponds to the given character data" (W3C, 2011c).

In other words, the <altGlyph> element is used to determine a text that is displayed, if a user agent cannot render special glyphs or fonts. In theory, the <altGlyph> element can apply to any font format, but the glyph addressing scheme was not defined for non-SVG fonts. Therefore, the <altGlyph> element is only reliable when used in combination with an SVG font. (ed__, 2015)

In SVG 1.1, 'SVG fonts' is a facility that allows users to define their own fonts and display them with simple <text> elements:

> "SVG fonts can improve the semantic richness of graphics that represent text. […] In some cases, accessibility may be enhanced by expressing the logo as a series of glyphs in an SVG font and then rendering the logo as a 'text' element which references this font."
>
> (W3C, 2011g)

Eisenberg and Bellamy-Royds (2014, p.137, pp.327-329) state the utility of 'SVG fonts' as a way of adding "special symbols that are not represented in Unicode" or "a subset of the Unicode characters without having to install an entire font". The authors also describe how to create a custom font, which can then be incorporated into the SVG with a unique 'id' for the <font> tag.

SVG 1.1 categorises this element as a text content element, but the working draft of SVG 2 has already revealed the removal of 'SVG fonts' and, consequently, the <altGlyph> element (as well as the <altGlyphDef>, <altGlyphItem> and <glyphRef> elements) in the next version (W3C, 2015a). Currently, both 'SVG fonts' and all of the above mentioned elements, which are closely linked to 'SVG fonts', are deprecated.

2.4.6 <switch>

While the <switch> element is not categorised as a text content element, but a container element ("[a]n element which can have graphics elements and other container elements as child elements" (W3C, 2011b)), it can have an important influence on the text that is displayed and therefore on the localisation process due to the possibility of the use of multiple languages. Eisenberg and Bellamy-Royds have also included this element in the chapter about text, under the section "Internationalization and Text" (2014, pp.135-137).

> "The 'switch' element evaluates the 'requiredFeatures', 'requiredExtensions' and 'systemLanguage' attributes on its direct child elements in order, and then processes and renders the first child for which these attributes evaluate to true".
>
> (W3C, 2011e)

In the case of the 'systemLanguage' attribute, its value signals the language of the text and the <switch> element then displays the text in the language of the software that is used to view the SVG. Eisenberg and Bellamy-Royds (2014, pp.135) state that SVG provides the "ability to display multiple languages in a single document" by using the <switch> element. They point out the utility of this capability in the context of brochures for an event with international visitors (ibid.), and it can be just as advantageous for providing multiple translations in a single document, creating a multilingual SVG.

Figure 5 shows a screenshot of a <switch> element from the source code of the multilingual SVG shown in Figure 6 below. The image was designed by Wikimedia Commons user Delphi234[14] and is made available under the Creative Commons CC0 1.0 Universal Public Domain Dedication[15]. The <switch> element can hold various translations, which are then rendered according to the language setting of the program used to view the SVG.

[14] https://commons.wikimedia.org/wiki/File:History_of_energy_consumption_in_the_United_States.svg

[15] To consult the Legal Code, please visit: https://creativecommons.org/publicdomain/zero/1.0/legalcode

```
<g font-size="22" transform="translate(340 70)">
<switch>
    <text systemLanguage="ar">تاريخ استهلاك الطاقة في الولايات المتحدة 1776-2013</text>
    <text systemLanguage="de">Energieverbrauch in den Vereinigten Staaten (1776-2013)</text>
    <text systemLanguage="es">Historia del consumo de energía en los Estados Unidos (1776-2013)</text>
    <text systemLanguage="fi">Historia energiankulutuksen Yhdysvalloissa (1776-2013)</text>
    <text systemLanguage="fr">Histoire de la consommation d'énergie aux Etats-Unis (1776-2013)</text>
    <text systemLanguage="it">Storia del consumo di energia negli Stati Uniti (1776-2013)</text>
    <text systemLanguage="ja">米国でのエネルギー消費量 (1776-2013)</text>
    <text systemLanguage="nl">Energieverbruik in de Verenigde Staten (1776-2013)</text>
    <text systemLanguage="pl">Historia zużycie energii w Stanach Zjednoczonych (1776-2013)</text>
    <text systemLanguage="pt">História do consumo de energia nos Estados Unidos (1776-2013)</text>
    <text systemLanguage="ru">История потребления энергии в США (1776-2013)</text>
    <text systemLanguage="sv">Historia av energiförbrukningen i USA (1776-2013)</text>
    <text systemLanguage="zh">在美國的能源消費量 (1776-2013)</text>
    <text>History of Energy Consumption in the United States (1776-2013)</text>
</switch>
</g>
```

Figure 5. Example of the source code of an SVG file containing a <switch> element with various translations

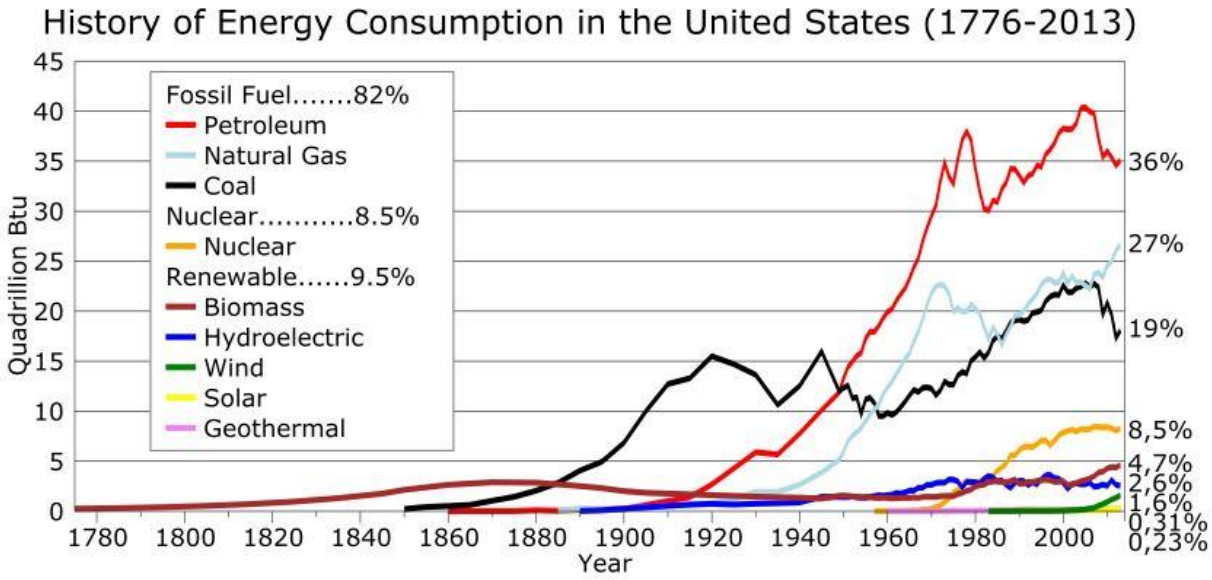


Figure 6. Scalable Vector Graphic of the History of energy consumption in the United States (1776-2013)

2.4.7 <desc> and <title>

The <desc> and <title> elements are used to provide a text-only description or title for a container element or a graphics element in SVG. These elements are typically not graphically rendered in the SVG file, but the <title> element may be displayed "as a tooltip, as the pointing device moves over particular elements". (W3C, 2011e)

Both of these elements can contain important information for accessibility purposes, which needs to be localised.

The example in Figure 7, taken from the W3C Recommendation on Scalable Vector Graphics (SVG) 1.1 (Second Edition) (W3C, 2011e), shows a <title> element and a <desc> element, which are used to describe the contents of a <g> element.

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
     version="1.1" width="4in" height="3in">
  <g>
    <title>Company sales by region</title>
    <desc>
      This is a bar chart which shows
      company sales by region.
    </desc>
    <!-- Bar chart defined as vector data -->
  </g>
</svg>
```

Figure 7. Example of an SVG file containing a <title> element and a <desc> element[16]

[16] Copyright © [16 August 2011] World Wide Web Consortium, (MIT, ERCIM, Keio, Beihang). http://www.w3.org/Consortium/Legal/2015/doc-license

3. Research Methodology

This chapter presents the methodology and the resources used in the course of this research project. It focuses on the tools that were evaluated (section 3.1), the data generation methods (section 3.2), how the SVG file type[17] for SDL Trados Studio 2015 was created (section 3.3), and a detailed description of the EAGLES method and its application to the evaluation of the tools (section 3.4).

As mentioned in section 1.2.2, the purpose of this study is to determine if CAT tools or image editors are better suited for the localisation of SVG images by translation students or novice translators or localisers with little or no image localisation experience. One tool of each type was chosen for evaluation: the CAT tool SDL Trados Studio 2015 and the vector graphics editor Inkscape 0.91 (cf. section 3.2).

To obtain an answer to the research question and test the hypothesis, an experiment was chosen as research strategy. The data for the evaluation was obtained with a method triangulation, using an experiment and questionnaires as data generation methods (cf. section 3.2).[18]

The experiment consisted of an SVG translation task from English to Spanish with SDL Trados Studio 2015 and Inkscape 0.91 that was given to six participants. A within-subjects approach was used by dividing the participants in two groups of three people each. The experiment design is further described in chapter 4.

The evaluation is based on several relevant quality characteristics set out by the EAGLES approach (cf. section 3.4). User-oriented test methods were applied for the system measurements (cf. 3.4.1): feature inspections, scenario tests and questionnaires.

[17] The program SDL Trados Studio 2015 uses the term 'file type' to denominate a filter for the display of translatable content for a certain file type. This research project employs both the terms 'file type' and 'filter'. The SDL Trados Studio 2015 terminology is only used in section 3.3 for the description of the steps needed to create a filter for a file type that is not supported in SDL Trados Studio 2015. This decision was taken in order to facilitate the filter creation process. The rest of the research project refers to this custom SVG file type as 'filter' or 'SVG filter', to be more specific and to avoid confusion.

[18] The denominations for the research strategy, data generation methods and data analysis follow the terminology set out by Oates (2006).

Both quantitative (e.g. the clicks needed during the translation process) and qualitative data (e.g. the screen recordings of the participants' on-screen work during the experiment or the translated SVG files) was obtained and analysed during the tests. The results are presented in chapter 5.

3.1 Tools for Evaluation

The tools SDL Trados Studio 2015 (section 3.1.1) and Inkscape 0.91 (section 3.1.2) were chosen for the evaluation and are presented in the following sections.

3.1.1 SDL Trados Studio 2015

SDL Trados Studio 2015 is a computer-assisted translation software (CAT tool). It is currently the leading CAT tool on the market with over 200,000[19] professional translators working with it. The tool "enables organizations to effectively manage all aspects of translation projects. Studio incorporates project management and computer-aided translation (CAT) tools for use by project managers, translators, editors, proofreaders and other language professionals" (SDL PLC, 2015a).

The CAT tool SDL Trados Studio 2015 supports over 70 different file types (SDL PLC, 2015b). SVG is not among the list of supported file types, but the program allows its users to create a customised filter that lets the program extract the translatable text (cf. section 3.3).

This tool was chosen for the evaluation, since it is the best-known and most used CAT tool on the market. Therefore, an evaluation based on this program would produce findings relevant to a greater number of CAT tool users than if a lesser used program had been chosen.

Furthermore, it was important to keep in mind the participant requirements for the choice of the tools to be evaluated. It would not have been possible to carry out an experiment for the evaluation of a CAT tool with participants without any notion of this tool, since this would probably have caused frustration or even withdrawal from the study among the participants. Therefore, the participants of the experiment needed to have at least basic notions of the CAT tool that would be included in the evaluation, given that these kinds of tools are usually more complex to use than graphics editors and require

[19] http://www.sdl.com/cxc/language/translation-productivity/trados-studio/

more effort to learn how to use them. The course 'Traduction assistée par ordinateur (TAO)' at the University of Geneva, focuses on computer-assisted translation, including the use of SDL Trados Studio 2015. This made it the ideal choice as a tool for the evaluation of SVG localisation, since participants who had already taken the course could be recruited, without the need of an explanation on how to use CAT tools.

3.1.2 Inkscape 0.91

Inkscape 0.91 is a free and open-source vector graphics editor that uses SVG as its native format (Inkscape, 2015a). The tool allows its users to create and edit SVG files and to view and alter the source code with an internal XML editor. Some of the tool's features are the creation and manipulation of objects, the management of object styles with fills and strokes, creating and editing paths, text support, including multi-line text and text on paths, and rendering of objects (Inkscape, 2015b)[20].

Inkscape 0.91 was chosen as a tool for the evaluation for various reasons: it is a well-known, free draw program that runs on Windows, Mac OS and Linux and is often featured in lists of the best open-source vector graphics editors online or as an alternative to expensive vector graphics editors that are most commonly used by professional designers.

An important consideration was the cost factor: The tool is available free of cost, thus it is accessible to a greater number of users than other tools that need to be purchased. The fact that the tool is free for use also facilitated the execution of the experiment, which was used for the evaluation, given that there was no need to acquire an expensive tool or license.

Furthermore, the program's native file format is SVG and it offers all of the functions needed for the localisation of textual content in SVG files, which made it an adequate choice for the representation of image editors in the evaluation.

Another reason is that this tool is used in the introduction to vector graphics localisation during the course 'Localisation et gestion de projet' at the University of Geneva, which introduces students to localisation and project management.

[20] A more detailed list of the features is available at https://inkscape.org/en/about/features/

3.2 Data Generation Methods

This research project uses a method triangulation, meaning that "the study uses two or more data generation methods", which "allows the findings from one method to be corroborated or questioned by comparing with the data from another method" (Oates, 2006). The empirical data for this research project was generated with two different methods: an experiment and questionnaires. They were designed in order to evaluate some of the characteristics of the CAT tool SDL Trados Studio 2015 and the vector graphics editor Inkscape 0.91 regarding the localisation of SVG images.

For the experiment, the participants were divided into two groups and asked to fill out a general questionnaire and then to translate SVG files using the aforementioned programs (cf. chapter 4). All onscreen work, as well as mouse clicks, virtual keys[21] and the duration of the translation tasks were recorded in order to yield objective results. The observations for the experiment were based on the data provided by the screen recordings, including the key logs, and the translated SVG files. The recordings of the whole experiment were also analysed for a better understanding of the correlations between the results, since they provided invaluable information on causality in the observed data. Upon completion of each translation process, a questionnaire was given to the participants to evaluate their personal experience.

The observations drawn from the experiment were compared with the data gathered from the questionnaires and they often complemented each other, thus helping understand the links between the data and the participants' actions (cf. chapter 5).

3.3 Creation of the SVG File Type for SDL Trados Studio 2015

The CAT tool SDL Trados Studio 2015 supports over 70 different file types (SDL PLC, 2015b), since the program converts all those different file formats to SDLXLIFF, which is a format used only by SDL that complies with XLIFF version 1.2. The function of filters in CAT tools is to convert any kind of format into the OASIS XML Localisation Interchange File Format (XLIFF), so that content may be processed and translated (Aranzana González, 2012). SDL Trados Studio uses filters for the more generic file types, which only display the translatable elements of the file. If the translator or localiser wants to work with a file

[21] The term 'virtual key' is used by the screen recording program BB Flashback Express 5 and refers to keyboard shortcuts (e.g. [Ctrl]+[Z]) and other special keys used for commands (e.g. 'Space' or 'Delete').

format that is currently not supported, a new file type with customised rules for the extraction of translatable content can be added, allowing the program to recognise the new file type and its textual content.

Although this particular CAT tool supports a generic XML filter, which might be used for XML based formats, it does not include a specific filter for SVG. Without a custom file type, the program does not recognise SVGs as files for translation and only adds them to a translation project as reference files. When the extension of an SVG file is changed to '.xml', SDL Trados Studio 2015 recognises the file type, but the rules for the extraction of translatable text in XML are not specifically designed for SVG and result in a subpar extraction of translatable text. A test with the unit testing SVG file with a changed extension to resemble an XML file, has demonstrated that SDL Trados Studio 2015 extracts untranslatable text that should not be shown in the 'Editor'.

Aranzana González (2012) published a guide in Spanish language concerning the basics of the creation of any kind of XML based filter that can be made with SDL Trados Studio 2009, which is still applicable to newer versions of the CAT tool. With the basics of XML filter creation in mind, an SVG filter had to include every element specific to the SVG format that could possibly contain localisable text. Only a filter including all possible text elements, thus ensuring that the CAT tool displays everything that needs to be translated, can be valid and applicable to all SVG files. The SVG specification explains all text content elements that might be used in SVG images (W3C, 2011c), making it the best starting point for the elaboration of the filter. An overview and a brief description of these elements can be found in section 2.4. Another important aspect of the localisation process is the translation of metadata, such as a title or description, which also has to be taken into account when creating a filter.

3.3.1 Recognition of Scalable Vector Graphics in SDL Trados Studio 2015

In SDL Trados Studio 2015, the 'File Types' dialogue box can be accessed by opening the 'Project Settings' when a project is opened or created (cf. Figure 8). During the creation of a new project, the SVG filter should be elaborated or imported before adding the files for the translation, since the program does not recognise the SVG format, if it has not been added beforehand, treating the documents as reference files. When the user tries to proceed without a file type that can be recognised, an error message appears on the

screen with a warning that the project must include at least one translatable file (cf. Figure 9).

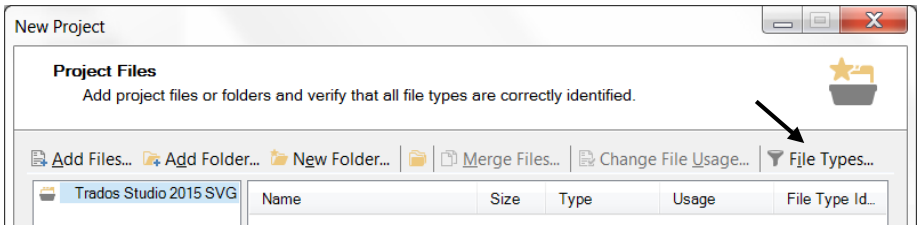


Figure 8. Accessing the 'File Types' settings during the creation of a new project in SDL Trados Studio 2015

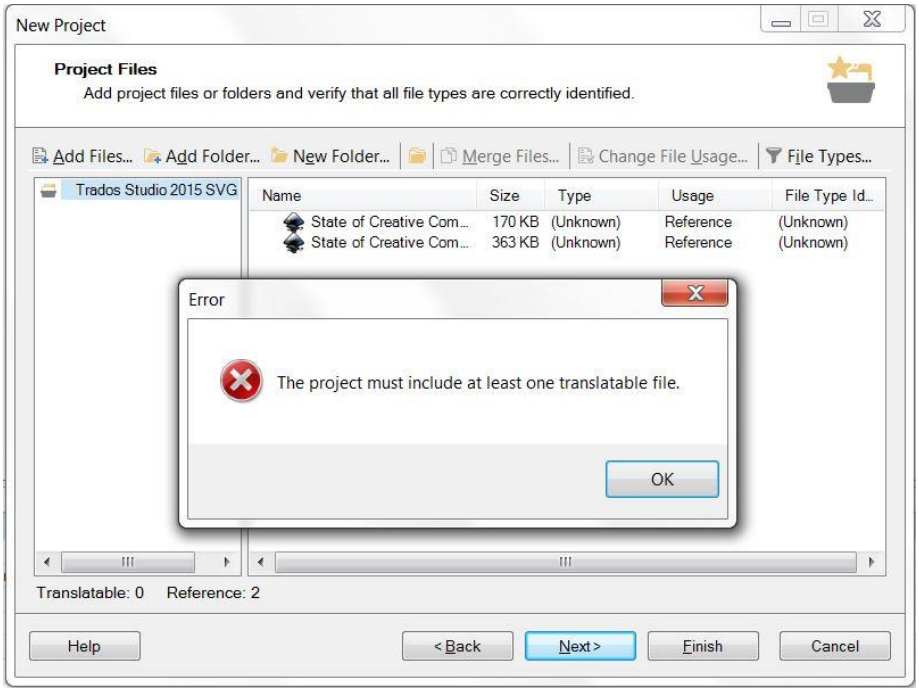


Figure 9. Adding files of an unknown type to the translation project in SDL Trados Studio 2015

If the user possesses an already existing SDLFTSETTINGS file that carries the settings for the desired file type, it is possible to import the file type in question by clicking on 'Import Settings…' in the 'File Types' dialogue box and selecting the SDLFTSETTINGS file.

Importing or adding a new file type in the 'Project Settings' will only make it available for the project that is currently in use. Another way to access the 'File Types' is by selecting 'Options' in the 'File' tab, which opens the global settings and affects all future projects created without a template (Filkin, 2014a). To import or create a new file type

that will appear in all future projects based on a template, it is necessary to edit the default project template under 'File' > 'Setup' > 'Project Templates'.

3.3.2 Adding a New File Type

When creating a new file type by clicking the 'New…' button in the 'Project Settings' or in the 'Options' window (if the user wants to add a file type to the global settings), SDL Trados Studio 2015 opens the 'Create File Type' wizard. The user has to choose one of the following file types: 'Simple Delimited Text', 'HTML', 'Regular Expression Delimited Text', 'XML (Embedded Content)' or 'XML (Legacy Embedded Content)' (Filkin, 2014b). Since SVGs are XML based, one of the two XML options is selected for the new SVG file type.

The difference between 'XML (Embedded Content)', which uses the new processor, and 'XML (Legacy Embedded Content)' lies in the way embedded content is treated. The old (legacy) processor extracts embedded content with the XML parser and processes them with a generic embedded content processor. Newer versions of SDL Trados Studio (version 2014 SP1 and later) use a "dedicated processor, specific to the type of embedded content found inside the XML document" (SDL PLC, 2015c). Either way, the user has to separately 'Enable embedded content processing' under the node 'Embedded content (Legacy)' after the new file type has been created. With the Legacy Embedded Content, the user has to define 'Tag definition rules' with regular expressions, whereas the new content processor allows the user to determine whether embedded content should be processed for 'CDATA sections' or other 'Document structure information' with default processors, which can also be configured (Filkin, 2014b)[22].

The following sections explain the steps of the 'Create File Type' wizard and the information that has to be entered to create the SVG filter.

3.3.2.1 File Type Information

The screenshot in Figure 10 depicts the interface where the user can enter general information on the new file type in SDL Trados Studio 2015. Mandatory settings are the 'File type name' and the 'File type identifier' (SDL PLC, 2015d), which have both been set

[22] Consult for further information on embedded content: http://producthelp.sdl.com/SDL_Trados_Studio_2015/client_en/About_Embedded_Content_Processors.htm and http://multifarious.filkin.com/2014/06/01/custom-xml/

to ‘SVG’ for the SVG filter. The user can provide additional information on the new file type with a ‘File type icon’ or by filling out the text boxes ‘Name of individual document’ and ‘Name of document category’. The ‘File dialog wildcard expression’ text box allows the user to enter the extension of the file name with a wildcard. In the case of Scalable Vector Graphics, the wildcard expression is ‘*.svg’, meaning that SDL Trados Studio identifies files of any given name with this extension as SVG documents and then applies this file type. When a file of this type is opened during translation, the “**Open Document** dialog box displays this expression in a drop-down list” (ibid.).

The user can also determine the name of the individual document and of the document category. Our SVG filter establishes the ‘Name of individual document’ as ‘SVG Document’ and the ‘Name of document category’ as ‘SVG Documents’. These details are optional and solely for information purposes, as they are only displayed in dialogue boxes when a file is opened or closed (ibid.).

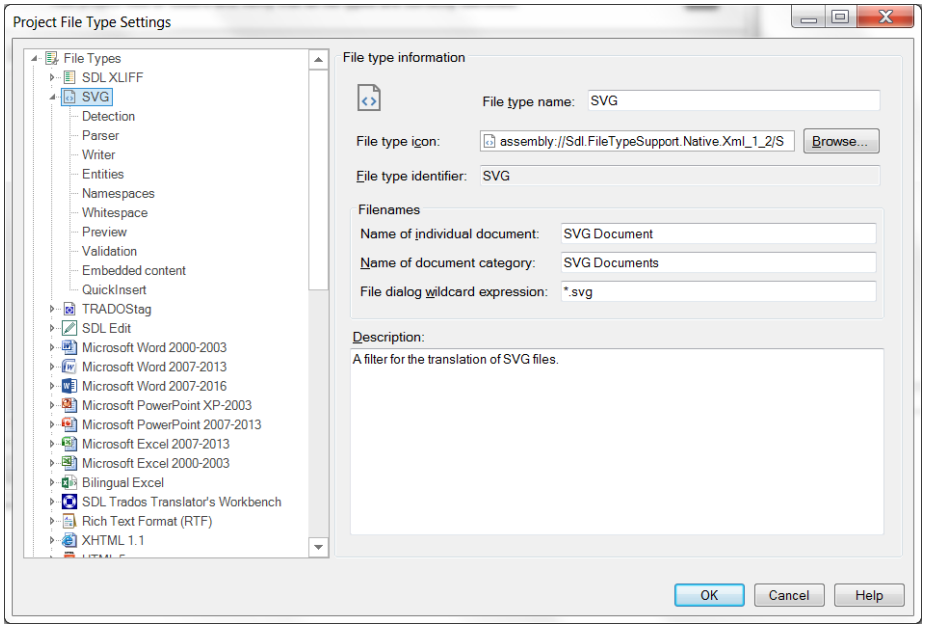


Figure 10. File type information in SDL Trados Studio 2015

3.3.2.2 XML Settings Import

XML settings can be imported in the next step. The option ‘Create an XML file type based on the default settings’ uses generic XML settings. The user can also import an XML file, which will serve as the base of the new file type filter, with the option ‘Define settings based on INI, ANL, XML, XSD, ITS or DTD rule file’. This is particularly convenient for XML

files, since the user is able to import all the elements and attributes of a custom file and can then simply select this content without having to look for the right element or attribute name.

For our filter, it is unnecessary to import a file, because the elements in SVG do not carry customised names. For this reason, the new file type can simply be based on the default settings.

3.3.2.3 Parser rules

Aside from recognising the file format, the file type separates the translatable content from the non-translatable. With the right set of rules, it lets the program know which elements contain text and therefore have to be displayed for translation and which elements do not have to be shown. This task is achieved with a set of rules in XML Path Language (XPath) indicating the elements. These are entered in the 'Parser' during the creation of a new file type with the wizard or upon completion by selecting the 'Parser' node and subsequently adding the rules.

XPath is a language that "uses path expressions to select nodes or node-sets in an XML document", which has been a W3C Recommendation since 16 November 1999 (W3Schools, 2015a). It "navigate[s] through elements and attributes in an XML document" (ibid.), such as a Scalable Vector Graphic. In SDL Trados Studio 2015, the rules are formulated with XPath expressions to designate which elements and attributes are to be translated.

SDL Trados Studio 2015 offers three options for entering the rule in the 'Parser', after the user opens the 'Add rule' dialogue box by clicking on 'Add…' (cf. Figure 11). The rules may be written directly with an XPath expression with 'Rule type' set to 'XPath' (cf. Figure 12). Another option is to select 'Element' or 'Attribute' as a rule type and to directly refer to an element or attribute (cf. Figure 13 and Figure 14). When an XPath expression that only refers to one element or attribute is entered, SDL Trados Studio 2015 automatically changes the rule type accordingly. Rules containing more than one element or attribute need to be created with a complete XPath expression.

When a new rule is added, it is necessary to indicate whether the elements or attributes that are specified with the XPath expression have to be extracted for translation or not. The user can choose among four settings for the 'Translate' property: 'Always

Translatable', 'Not translatable', 'Translatable (but not in protected content)' and 'Not specified'. The default setting is 'Translatable (but not in protected content)', meaning that the translator or localiser cannot modify content that is written inside tag pairs (e.g. the bold tag <b>). Considering that SVG provides text styling or other information inside elements, such as the <tspan> element, the user can simply choose 'Always translatable'. An example of this would be a bold word in the middle of a sentence. In this case, the sentence is written inside a <text> element and only the bold word appears inside a <tspan> element (e.g. **<text x="0" y="0">This is a <tspan style="font-weight:bold">bold</tspan> text</text>**). SDL Trados Studio 2015 simply displays all the elements separately. The 'Always translatable' option also makes sure that all the translatable text is displayed in the 'Editor'.

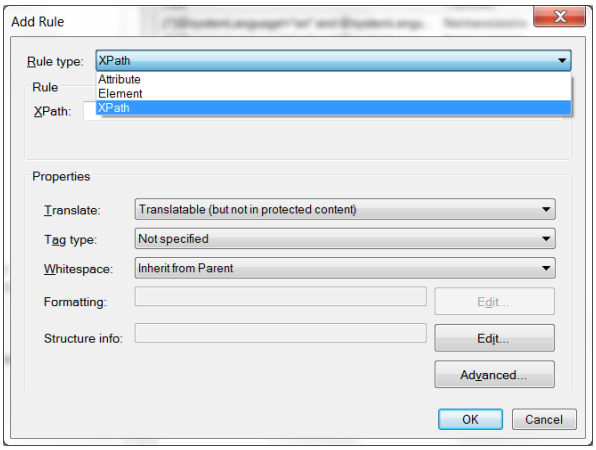


Figure 11. Rule types in SDL Trados Studio 2015

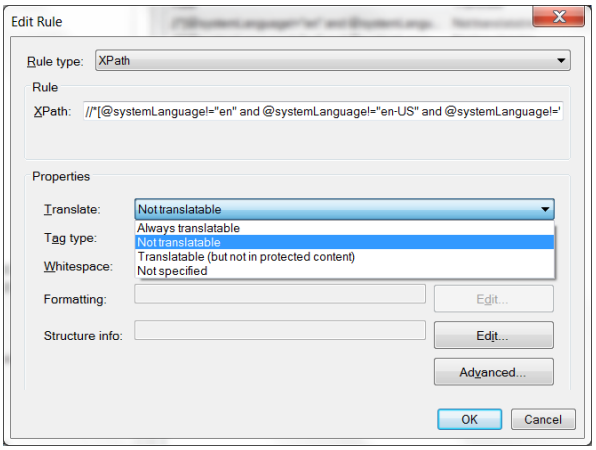


Figure 12. XPath rule type in SDL Trados Studio 2015

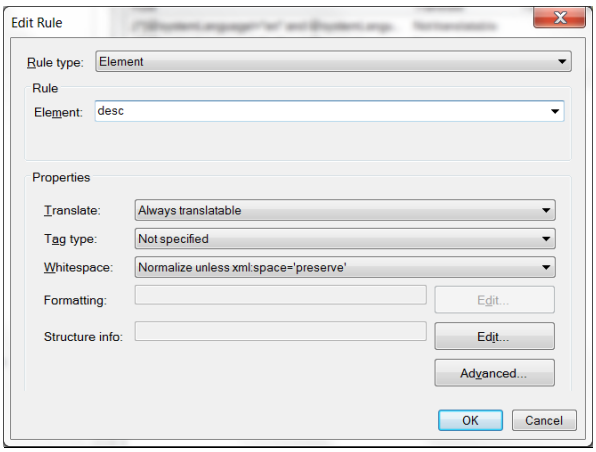


Figure 13. Element rule type in SDL Trados Studio 2015

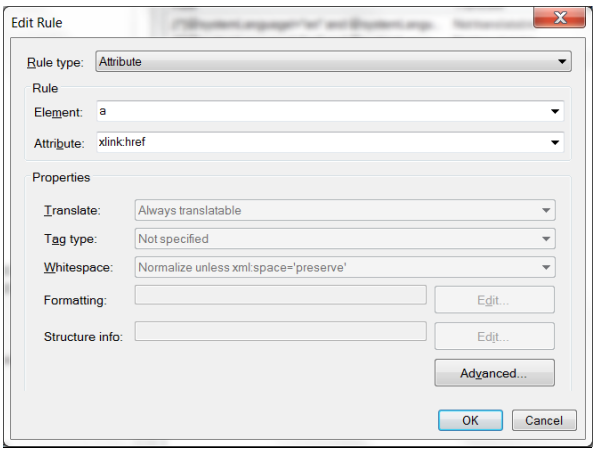


Figure 14. Attribute rule type in SDL Trados Studio 2015

In XPath, a single slash sign selects from the root node and two slash signs select "nodes in the document from the current node that match the selection no matter where they are" (W3Schools, 2015b). Accordingly, we can refer to an element with one slash (e.g. **/element**), to select a root element. A path that starts with one slash "always represents an absolute path to an element" (ibid.). An expression with two slashes (e.g. **//element**) selects all of these elements, regardless of their position in the document. Wildcards, such as an asterisk, can be used to refer to unknown elements or attributes (e.g. **//*** or **@***), meaning that the expression matches any element or attribute of the document. The symbol '@' is used to select an attribute and has to be placed right in front of it (e.g. **@attribute**). Predicates, which are written in square brackets, can be used to further limit a path and find a specific element, attribute or value of an attribute (e.g. **//element[@attribute='value']**). The value of an attribute is preceded by the operator[23] '=' and written inside single or double quotation marks. (ibid.)

The SVG filter for the source language English has to contain the rules presented below. Table 1 lists the XPath expressions of the English SVG filter and the corresponding 'Translate' settings. The screenshot of the complete filter in Figure 15 illustrates these rules.

[23] A list of operators that can be used with XPath expressions is included in the W3Schools tutorial at http://www.w3schools.com/xsl/xpath_operators.asp. The rules of the SVG filter employ the operators 'not equal' (**!=**), placed before the value of an attribute, as well as 'and', which is used to include several 'systemLanguage' attributes in one expression.

Rule No.	XPath rule	Translatable
1	//*[@systemLanguage!="en" and @systemLanguage!="en-US" and @systemLanguage!="en-GB" and @systemLanguage!="en-CA" and @systemLanguage!="en-AU" and @systemLanguage!="en-NZ" and @systemLanguage!="en-IE"]	No
2	//*[@systemLanguage!="en" and @systemLanguage!="en-US" and @systemLanguage!="en-GB" and @systemLanguage!="en-CA" and @systemLanguage!="en-AU" and @systemLanguage!="en-NZ" and @systemLanguage!="en-IE"]//*	No
3	//a/@xlink:href	Yes
4	//desc	Yes
5	//title	Yes
6	//text	Yes
7	//text//*	Yes
8	//*	No

Table 1. XPath rules of the SVG filter in English

The first rule of the filter (cf. Table 1) determines that all text written in another language than the source language (the main variants of English in our case) is not translatable. This prevents the program from displaying the text contained in these elements during the translation process in the 'Editor'. The rule refers to any element of the SVG (**//***) with 'systemLanguage' attributes other than (**!=**) English (**en**), American English (**en-US**), British English (**en-GB**), Canadian English (**en-CA**), Australian English (**en-AU**), New Zealand English (**en-NZ**) and Irish English (**en-IE**). The 'Translate' property needs to be set to 'Not translatable' for this rule, which only leaves text content of these English variants to translate.

The second rule is the same as the first one, but it also includes all the child elements (**//***) that might be enclosed inside elements with a 'systemLanguage' other than English or one of its main variants (cf. Table 1).

The country codes are used in the first and second rule, since creators of SVG files might set the value of the 'systemLanguage' attribute to a specific variant of the English language. Even though most creators may choose the generic language code 'en', the

possibility of using a country code exists. The World Wide Web Consortium (W3C) uses the Language Subtag Registry[24] provided by the Internet Assigned Numbers Authority (IANA) to determine a language tag or subtag.

The first two rules include the variants of the main countries of the Anglosphere, as established in the English Language Blog[25]. In theory, every country code of the English speaking world can be incorporated into the rules, but this would make them immensely long.

A complete SVG filter should not neglect the country codes in the SVG parser rules. Although it is possible to create a filter that encompasses only the language code, the translator or localiser should take note that some translatable elements are not shown, if the SVG file contains any country code. Eisenberg and Bellamy-Royds (2014, p.136) claim that with switch elements, "[a] match of language code alone is considered a match, and country codes are used only to "break a tie" […].

However, testing of the SVG file type has proven that it will not display translatable elements with a country code, if this code is not specifically included in the rules. For example, the XPath expression **//*[@systemLanguage!="en"]** does not render any results for the element **<text systemLanguage="en-US">**. This highlights the significance of country codes in the rules of the filter. If the translator or localiser notices that SDL Trados Studio does not display all translatable text, the source code of the file should be checked for country codes that have been omitted in the 'Parser' rules.

Rule number three dictates that **//a/@xlink:href** is translatable. The <a> element is used to create hyperlinks to a destination defined by the attribute 'xlink:href'. This rule selects every <a> element with the attribute 'xlink:href'. Here, the 'Translate' property needs to be set to 'Always translatable'.

It has to be pointed out that the working draft of the W3C Scalable Vector Graphics (SVG) 2 recommendation has deprecated the 'xlink:href' attribute in favour of 'href' without the use of the XLink namespace (W3C, 2015a). In the current SVG 1.1 version, the attribute is still specified with the namespace. Elements using the new attribute 'href' will

[24] http://www.iana.org/assignments/language-subtag-registry/language-subtag-registry

[25] http://blogs.transparent.com/english/what-is-the-anglosphere/

also recognise the attribute in the XLink namespace, for the sake of backwards compatibility (W3C, 2015b).

The fourth rule searches for the content of <desc> elements (accessed with the expression **//desc**) and defines it as translatable. Text inside a <desc> element does not appear in the SVG image, but it can be used to include descriptions of any kind (e.g. for a table or chart that can be seen in the graphic). This content is important for accessibility reasons and needs to be localised.

In the same manner as the <desc> element, the <title> element, which is the focus of the next rule, is not shown in the graphic, but contains valuable information to make an SVG more accessible. Rule number five establishes that all <title> elements (**//title**) are 'Always translatable'.

Rules number six and seven concern the <text> element. The expression **//text** selects all the <text> elements and **//text//*** matches all of the child elements, which might be enclosed inside a <text> element. Both rules need to be set to 'Always translatable'. Text content child elements, such as <tspan> and <tref>, as well as the now deprecated <altGlyph> element, are already included in the latter rule, since they can only appear inside a <text> element. Therefore, it is not necessary to make separate rules for elements that are contained in this element.

The last rule is the XPath expression **//***, which is set to 'Not translatable'. This expression selects all the elements of the document that have not been specifically addressed with the other rules. It allows the user to exclude all the elements that are not needed for the localisation of the document and to only pick out the few that are important.

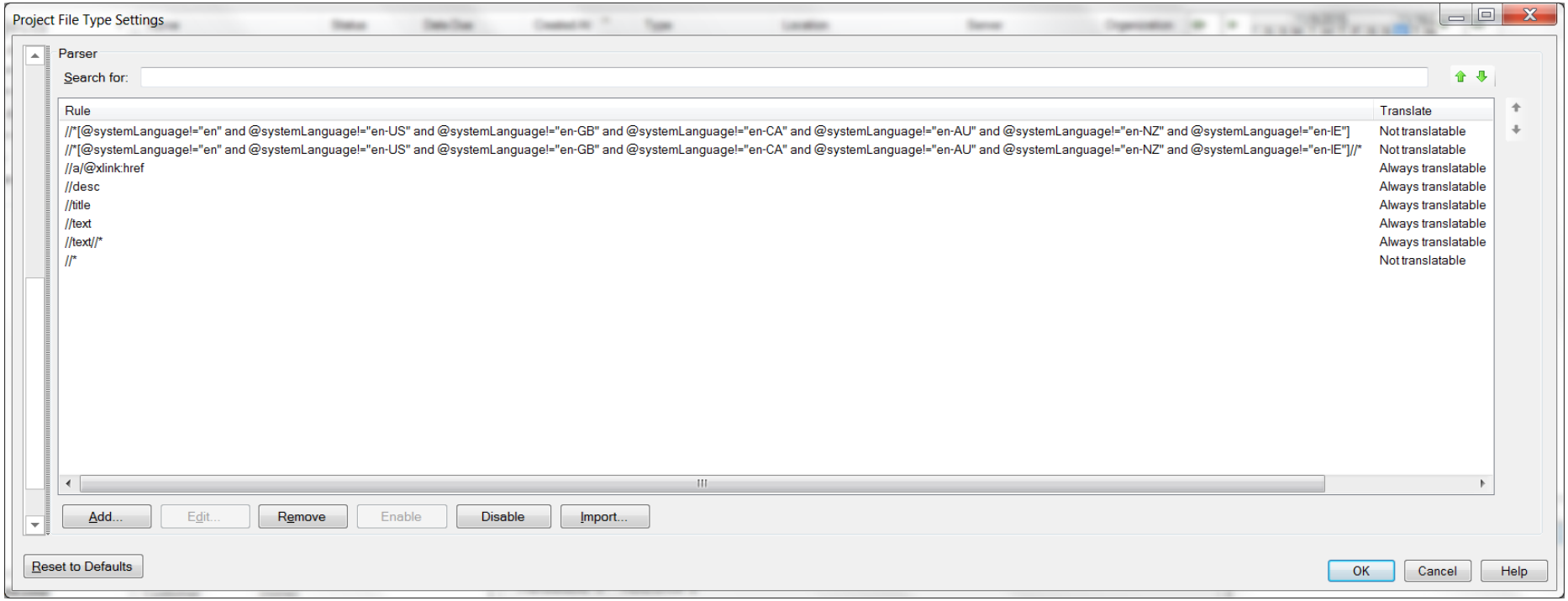


Figure 15. The final SVG filter for SDL Trados Studio 2015 (English source language)

An important aspect needs to be kept in mind when entering the rules in the 'Parser': the XPath expressions follow a hierarchical order. This means that the first rule is applied before all the others. In Figure 16 we can see an example of a rule that illustrates the significance of this hierarchy. The XPath expression dictating that //* is not translatable must be placed at the end of the rules.

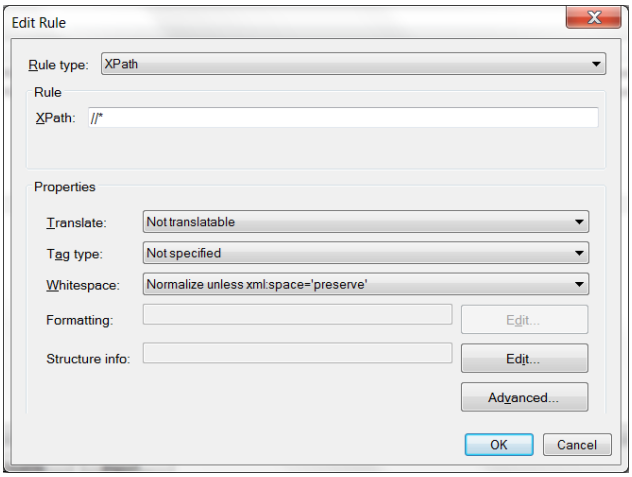


Figure 16. Last rule of the SVG filter in SDL Trados Studio 2015

As explained above, this expression refers to all the elements of the document. By placing this rule in the beginning, the program is told not to translate any elements and will not show any text. When placed at the end, though, it means that all the elements that have not been especially mentioned before this rule do not have to be shown in the editor. (Aranzana González, 2012)

During the creation of new rules, the user can also opt to change the 'Whitespace' settings, which determine how Studio handles extra whitespace. Spaces, non-breaking spaces, tabs and line breaks are considered as whitespace (SDL PLC, 2015e). Normalising these spaces can be convenient for the translator or localiser, since extra line breaks, etc. will then be eliminated from the 'Editor' view, making the text more readable.

The 'Properties' of the 'Parser' rules establish whether whitespace is kept or replaced with a single space. The options are 'Inherit from Parent' (the parent element dictates how whitespace is handled), 'Always preserve' (whitespace is not replaced), 'Normalize unless xml:space='preserve' (whitespace is only replaced when the element does not contain the attribute xml:space='preserve') or 'Always normalize' (whitespace is always replaced) (ibid.). The rules of this SVG filter establish that whitespace is normalised unless an element has the attribute xml:space='preserve'.

3.3.2.4 File Detection

It is important to supply information about the file type, so that the program will be able to recognise SVG files when they are added to the project and to apply the right filter. As mentioned above, it does not suffice to simply change the file extension of an SVG file to '.xml', since the XML filter has not been designed to suit the translatable content of elements and attributes specific to the SVG format.

In the 'Create File Type' wizard, filling out the 'File Detection' information is the last step that comes right after the rule creation (cf. section 3.3.2.3). The information required for file detection can also be entered after the creation of a new file type. On the left side of the menu, the different points containing information on the filter can be selected (cf. Figure 17). The root element 'svg' needs to be added under the menu item 'Detection', telling the program that the SVG filter is applied for files with the root element <svg> in the source code (cf. Figure 17). This information will suffice to detect the file type, but it is also possible to provide further details, such as the URI in an 'xsi:schemaLocation declaration', the 'DOCTYPE declaration with element name', a URI of an 'xmlns:declaration' or 'XPath rule matches'.

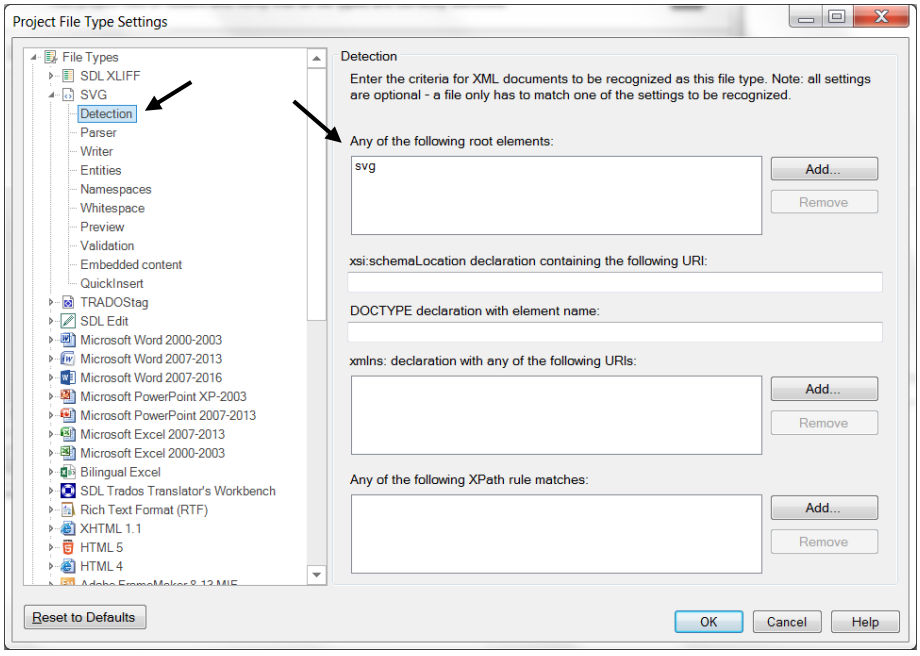


Figure 17. File type detection in the SDL Trados Studio 2015 'Project Settings'

3.3.3 Limitations of the SDL Trados Studio 2015 SVG Filter

The SVG filter manages to extract the translatable content of most common SVG files, but in some cases, the filter shows content that should not be translated. This section presents these limitations, based on a unit testing file (cf. Appendix I), which was created specifically to test the filter's capacities.

The unit testing SVG comprises text content that is translatable, marked as 'Show text' or 'Show Tref', plus a numbering of the text content. Text content that is not translatable is marked as 'Do not show text' or 'Do not show Tref' and the corresponding number (cf. Figure 18).

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
 <svg xmlns="http://www.w3.org/2000/svg"
     xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:ev="http://www.w3.org/2001/xml-events"
     version="1.1" baseProfile="full"
     width="400" height="500">
<title>Title (metadata)</title>
<desc>Description (metadata)</desc>
<defs>
<text id="tref1">Show Tref1</text>
<text id="tref2">Do not show Tref2</text>
</defs>
<defs systemLanguage="en">
<text id="tref3">Show Tref3</text>
</defs>
<defs systemLanguage="fr">
<text id="tref3">Do not show Tref3</text>
</defs>
<defs>
<path id="text_path1" x="0" y="0" d="M0 220 C100 240 200 200 200 100 C200 100 200 20 250 20"
    fill="none" stroke="silver" />
<path id="text_path2" x="0" y="0" d="M0 240 C100 260 200 200 200 100 C200 100 200 20 250 20"
    fill="none" stroke="silver" />
    <path id="text_path3" x="0" y="0" d="M0 260 C100 280 200 200 200 100 C200 100 200 20 250 20"
    fill="none" stroke="silver" />
<path id="text_path4" x="0" y="0" d="M0 280 C100 300 200 200 200 100 C200 100 200 20 250 20"
    fill="none" stroke="silver" />
<path id="text_path5" x="0" y="0" d="M0 300 C100 320 200 200 200 100 C200 100 200 20 250 20"
    fill="none" stroke="silver" />
</defs>

<text x="0" y="10">Show <tspan style="font-weight:bold">bold text</tspan> 1</text>
<text x="0" y="30">Show text 2</text>
<text systemLanguage="en" x="0" y="50">Show text 3</text>
<text systemLanguage="en-US" x="0" y="70">Show text 4</text>
<text systemLanguage="fr" x="0" y="80">Do not show text 1</text>
<text systemLanguage="de" x="0" y="90">Do not show text 2</text>
<text systemLanguage="en, fr" x="0" y="100">Show text 5</text>
```

Figure 18. First part of the source code of the unit testing SVG

The unit testing file was translated in SDL Trados Studio 2015 with the custom SVG filter. This process has revealed the flaws of the filter that are explained in the following paragraphs.

As described in section 2.4.3 (<tref>), the text contained in a <tref> is specified in the <defs> element and later referenced in a <text> element, which accesses this text. The filter displays a flaw with <tref> elements of a language that is not the source language. Information on the language of the content of a <tref> element can be directly included in this element by using the 'systemLanguage' attribute. However, it might also be the case that an SVG creator includes the language information in the <text> element, which references the <tref> (cf. Figure 20).

```
 9  <defs>
10  <text id="tref1">Show Tref1</text>
11  <text id="tref2">Do not show Tref2</text>
12  </defs>
```

Figure 19. Example of the source code of the unit testing SVG containing the references for the <tref> elements

```
83  <text systemLanguage="fr" x="0" y="430"><tref xlink:href="#tref2" /></text>
```

Figure 20. Example of the source code of the unit testing SVG containing a <tref> element

The unit testing includes the theoretical case of a <text> element with the 'systemLanguage' attribute set to French (**<text systemLanguage="fr" x="0" y="430"><tref xlink:href="#tref2" /></text>**). Normally, the content of such a <text> element is not shown in the 'Editor' view. However, because of the <tref> element, the translatable text is not written inside the <text> element, but inside the <defs> element, which (in this specific case) does not contain any information on the system language (cf. Figure 19). Since the element holding the translatable text does not contain any information on the system language, the French text is displayed in the 'Editor'. This issue does not occur when the system language is correctly specified in the <tref> element itself.

Another shortcoming of the filter is the exclusion of the content of <text> elements with more than one system language. The 'systemLanguage' attribute can have more than one value (e.g. **<text systemLanguage="en, fr" x="0" y="100">Show text 5</text>**). This example includes the system language English, so the text content should be translatable. Nonetheless, the filter does not recognise the content of such an element as translatable, meaning that it is not displayed in the 'Editor'. An interesting approach to resolve this issue would be the use of the XPath 'contains' function (e.g. **//text[contains(@systemLanguage,"en")]//***) in the 'Parser' rules. However, this method was unsuccessful when it was incorporated into the parser rules of SDL Trados Studio 2015.

The filter normally recognises the translatable and non-translatable content in a <switch> element, if the 'systemLanguage' attribute uses different language codes. When the <switch> element only includes text content elements of the same language, but with different country codes, and an element without any language indication (cf. Figure 21), the filter shows all of the options in the 'Editor'.

```
75 <switch>
76 <text systemLanguage="en-US" x="0" y="380"><tspan>Show text 21</tspan></text>
77 <text systemLanguage="en-GB" x="0" y="380"><tspan>Do not show text 12</tspan></text>
78 <text x="0" y="380"><tspan>Do not show text 13</tspan></text>
79 </switch>
```

Figure 21. Example of the source code of the unit testing SVG containing a <switch> element that is not rendered correctly with the SVG filter

A last limitation of the filter could be determined with rule number three, which dictates that **//a/@xlink:href** is translatable. In theory, an SVG author could decide to create hyperlinks with a 'systemLanguage' attribute in order to display a certain webpage depending on the user's language settings (e.g. www.google.de for systems set to German and www.google.com for systems set to English).

Figure 22 depicts the <a> elements of the unit testing file. Line 86 shows an <a> element without any system language information, whereas the following two elements include language information. When the SVG filter is applied, the content of the first and the third <a> element (lines 86 and 88) is entirely extracted for translation (including the hyperlink). The second <a> element (line 87) is not supposed to be translatable, since this element's 'systemLanguage' attribute is set to German (**systemLanguage="de"**). Although the filter does not show the text contained inside the <text> child element, the

filter does extract the information inside the 'xlink:href' attribute (**xlink:href=www.google.de**).

An approach to solve this issue was the creation of a more specific rule (**//a[@systemLanguage!="en"]/@xlink:href**), but this method was not successful either.

```
86 <a xlink:href="www.google.com"><text x="0" y="460">Show text 22 (link)</text></a>
87 <a systemLanguage="de" xlink:href="www.google.de"><text x="0" y="470">Do not show
   text 14 (link)</text></a>
88 <a systemLanguage="en" xlink:href="www.google.com"><text x="0" y="480">Show text 23
   (link)</text></a>
```

Figure 22. Example of the source code of the unit testing SVG containing the <a> elements

Lastly, it is necessary to take into account that the filter does not recognise text that has been represented graphically with shapes or paths, instead of inside a text content element. Usually, vector graphics creators will use such graphical representations of text for logos that do not have to be changed. Any text that is susceptible to change should logically be created with the use of text content elements, since it is much more laborious to create text with shapes and paths than with a simple <text> element. Also, it erases the possibility to edit text by simply writing inside the source code or with the text editing functions of a vector graphics editor, which is normally a very big advantage of Scalable Vector Graphics. The localisation procedure of such elements would be very similar to the procedure used for pixel images, thus defeating the purpose of using vector graphics in the first place. Since such elements are usually not meant to be changed, they are not regarded as translatable elements for the purpose of the evaluation carried out in this research project.

3.4 EAGLES Evaluation Method

The evaluation of the findings follows the EAGLES 7-step recipe for the "evaluation of language technology systems or components" (EAGLES, 1999), which was designed by the EAGLES Evaluation Working Group.

Starlander and Morado Vázquez (2013) carried out a case study on the critical evaluation of the "appropriateness of a CAT tool in a given scenario", examining the EAGLES approach. The study has shown the utility of the 7-step recipe when the researchers incorporated it in their own courses at the Faculty of Translation and Interpreting of the University of Geneva. Furthermore, the authors write that it is a "clear step by step method for performing a user-oriented evaluation" and that the EAGLES approach offers a "clearly context-based orientation", making it an evaluation framework that can easily be adapted to the needs of specific users and contexts. (ibid.)

The evidence presented in this study suggests that the 7-step approach is a suitable framework for the assessment of SVG localisation in SDL Trados Studio 2015 and Inkscape 0.91 by translation students or translators/localisers who have recently started their careers.

Furthermore, Valeria Siano (2015) and Cristina Peron (2013) have used this evaluation method in their Master's theses, which present similar studies containing a comparison of tools. This highlights the appropriateness of the EAGLES approach and inspired the author to use it for the evaluation laid out in this study.

3.4.1 Test Methods

The methods for system measurement under EAGLES are established in the final report of the EAGLES Evaluation of Natural Language Processing Systems (EAGLES, 1996). Since our evaluation is based on the needs of a certain user class, we are going to apply a user-oriented test type. EAGLES (ibid.) mentions three types of user oriented tests: **scenario tests**, **systematic testing** and **feature inspection**.

The scenario test is the most suitable option for a big part of our evaluation, since its primary aim is to "assess the suitability of a software product for every-day routines" (EAGLES, 1996, p.33). It is meant to test the system by letting the envisaged user perform a standardised task and is especially useful for providing detailed empirical information on the attributes of the usability characteristic, but can also be used for the sub-

characteristics suitability, accuracy, interoperability, time behaviour, resource behaviour, changeability and adaptability (ibid.).

Systematic testing "examine[s] the behaviour of software under specific conditions with particular results expected" and does not rely on users, as it can be carried out by "software engineers and/or user representatives" (EAGLES, 1996, p.33-34).

The third test type is the feature inspection in which the technical features of a piece of software are described in as much detail as possible, allowing a comparison between systems of the same type (EAGLES, 1996, p. 34-35).

In conformity with the testing instruments suggested in the EAGLES final report (1996), our evaluation employs scenario tests and feature inspections. Furthermore, it uses a logging program called BB FlashBack Express 5[26] to record how users interact with the system, a general questionnaire and an SVG localisation questionnaire for each tool, which is used to obtain the user's point of view on the SVG localisation with the programs to be evaluated.

3.4.2 The 7-Step Recipe

The EAGLES Evaluation Working Group provides a brief overview of the "7 major steps necessary to carry out a successful evaluation of language technology systems or components" (EAGLES, 1999). The EAGLES evaluation method is explained in more detail in the final report of the EAGLES Evaluation of Natural Language Processing Systems (EAGLES, 1996), which was published in October 1996.

The EAGLES framework was applied in the evaluation of SDL Trados Studio 2015 and Inkscape 0.91 for the translation of SVG files by novice translators or localisers with little or no image localisation experience. The main steps are the following (EAGLES, 1999):

1. The first step focuses on why the evaluation is being done and determines the purpose and the object of the evaluation (a system, a system component, a system in isolation or in a specific context of use).

[26] http://www.bbsoftware.co.uk/bbflashbackexpress/home.aspx

2. The second step is the elaboration of a task model, including the identification of all relevant roles and agents. This step establishes what the system is going to be used for and who will use it.
3. The third step consists of the definition of top level quality characteristics, by establishing the features of the system that need to be evaluated and if they are equally important.
4. In the fourth step the researcher produces detailed requirements for the system under evaluation based on steps two and three. In order to find a valid and reliable way of measuring how the system under evaluation performs in respect to a feature, this feature might be broken down until measurable sub-attributes are obtained.
5. Step five is dedicated to the definition of metrics that are applied to the system for the requirements produced under step four. The researcher defines how each attribute will be measured and establishes what counts as a good score, a satisfactory score or an unsatisfactory score, as well as the cut-off points. Furthermore, the relative importance of the sub-attributes (given the task model) is established at this stage.
6. Step six comprises the design of the execution of the evaluation, by developing test materials and establishing the time, circumstances and agent to carry out the measurements, as well as the form of the end result.
7. Step seven is dedicated to the actual measurements, including the comparison of the measurements with the pre-determined satisfaction ratings. The results are presented in an evaluation report.

The following sections show how the 7-step-recipe was applied to our evaluation of SDL Trados Studio 2015 and Inkscape 0.91.

3.4.2.1 Step One: Why Is the Evaluation Being Done?

The purpose of the evaluation is to analyse the suitability of two programs for the localisation of Scalable Vector Graphics. The participants of the study are students or former students who have completed the university course 'Traduction assistée par ordinateur (TAO)' or 'Localisation et gestion de projet' of the Faculty of Translation, University of Geneva.

The evaluation concerns the localisation of SVG files using the CAT tool SDL Trados Studio 2015 and the vector graphics editor Inkscape 0.91.

3.4.2.2 Step 2: Task Model

The scenario to be studied involves students of the Master of Arts in Translation at the University of Geneva or novice translators or localisers who have accomplished the aforementioned Master in the last two years, translating from English to Spanish.

The system is going to be used for the translation of text that is found inside vector graphics. Both systems may be used in an educational as well as a professional environment by translation students and professional translators or localisers, who need to work with text embedded in vector graphics. They will use the program to translate text in SVG files. The students usually have little practical experience in the localisation of vector graphics and even the more experienced translators or localisers could have little experience in this field, if they do not work with this data type on a regular basis.

3.4.2.3 Step Three: Top Level Quality Characteristics

The EAGLES framework model establishes six quality characteristics and sub-characteristics to evaluate software products, based on the standard ISO/IEC 9126[27] on the evaluation of software quality (EAGLES, 1996). The evaluation of SDL Trados Studio 2015 and Inkscape 0.91 presented in this research project is based on the criteria set out in this framework. The criteria was then adapted to better reflect the needs of the target users, which are translation students or novice translators or localisers with little or no image localisation experience. The six quality characteristics and their respective

[27] The standards ISO/IEC 9126:1991 and ISO/IEC 9126-1:2001 have now been withdrawn and revised by ISO/IEC 25010:2011.

sub-characteristics used by the EAGLES framework based on the standard are listed as follows under ISO/IEC 9126-1 (2001):

1. Functionality: "The capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions" (ISO/IEC 9126-1, 2001). The 'functionality' sub-characteristics are:
    a. Suitability
    b. Accuracy
    c. Interoperability
    d. Security
    e. Functionality Compliance
2. Reliability: "The capability of the software product to maintain a specified level of performance when used under specified conditions" (ISO/IEC 9126-1, 2001).The 'reliability' sub-characteristics are:
    a. Maturity
    b. Fault tolerance
    c. Recoverability
    d. Reliability compliance
3. Usability: "The capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions" (ISO/IEC 9126-1, 2001). The 'usability' sub-characteristics are:
    a. Understandability
    b. Learnability
    c. Operability
    d. Attractiveness
    e. Usability compliance
4. Efficiency: "The capability of the software product to provide appropriate performance, relative to the amount of the resources used, under stated conditions" (ISO/IEC 9126-1, 2001). The 'efficiency' sub-characteristics are:

a. Time behaviour

b. Resource utilisation

c. Efficiency compliance

5. Maintainability: “The capability of the software product to be modified” (ISO/IEC 9126-1, 2001). The ‘maintainability’ sub-characteristics are:

    a. Analysability

    b. Changeability

    c. Stability

    d. Testability

    e. Maintainability compliance

6. Portability: “The capability of the software product to be transferred from one environment to another” (ISO/IEC 9126-1, 2001). The ‘portability’ sub-characteristics are:

    a. Adaptability

    b. Installability

    c. Co-existence

    d. Replaceability

    e. Portability compliance

These characteristics are of a very general nature and the EAGLES group later added a seventh characteristic, customisability, which addresses the possibility to modify the system in order to satisfy the user’s needs (King and Maegaard, 1998).

The ISO/IEC 9126-1 (2001) quality characteristics and their sub-characteristics “can serve as a check-list when defining the quality model for a piece of software“ (King, 1997). According to EAGLES (1996), quality depends on given needs and therefore varies from one evaluation to another. Thus, the characteristics and sub-characteristics need to be adapted to the specific needs of the class of users considered in the evaluation.

Since our evaluation focuses on the needs of translation students or novice translators or localisers with little or no experience in the field of image localisation, the characteristics and sub-characteristics shown in Table 2 were deemed important for the evaluation.

For this particular evaluation, the characteristics 'functionality', 'usability' and 'efficiency' were chosen, as can be seen in Figure 23. The quality characteristic 'reliability' was left out of the evaluation, given that it was not carried out over a period of time and therefore cannot measure if the system is capable of maintaining its level of performance. 'Maintainability' and 'customisability' were not included in the criteria, since the possibility of modifying the software does not form part of the research objective. 'Portability' has also been excluded, since our research focuses on the translation process itself and is not aimed at determining how the software adapts to different environments.

Adding the characteristic 'cost' to this list of characteristics to be evaluated was also considered, since the price of a tool may be an important aspect for translation students or novice translators or localisers, who have just started their careers. The price can often determine if a tool is accessible to young professionals with a small budget. Though the cost of a program is an important aspect for a professional translator or localiser, it may be negligible, since a CAT tool will be a necessary purchase for the translation of files other than vector graphics, which would justify a high price. For this reason, the cost was ultimately not considered in this evaluation.

Characteristics	Sub-characteristics	Definition of the sub-characteristics according to ISO/IEC 9126-1
Functionality	Suitability	*The capability of the software product to provide an appropriate set of functions for specified tasks and user objectives.* (ISO/IEC 9126-1, 2001)
	Accuracy	*The capability of the software product to provide the right or agreed results or effects with the needed degree of precision.* (ISO/IEC 9126-1, 2001)
Usability	Understandability	*The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.* (ISO/IEC 9126-1, 2001)
	Learnability	*The capability of the software product to enable the user to learn its application.* (ISO/IEC 9126-1, 2001)
	Operability	*The capability of the software product to enable the user to operate and control it.* (ISO/IEC 9126-1, 2001)
	Attractiveness	*The capability of the software product to be attractive to the user.* (ISO/IEC 9126-1, 2001)
Efficiency	Time behaviour	*The capability of the software product to provide appropriate response and processing times and throughput rates when performing its function, under stated conditions.* (ISO/IEC 9126-1, 2001)

Table 2. Characteristics and sub-characteristics including their definitions according to ISO/IEC 9126-1 (2001) used in the evaluation

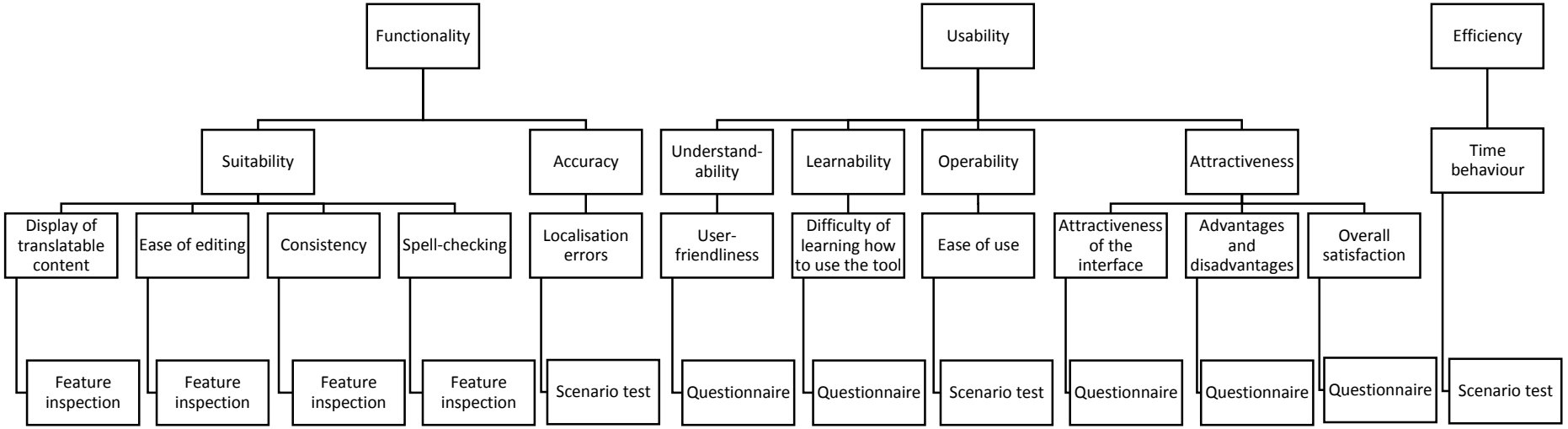


Figure 23. Hierarchical diagram of the characteristics, sub-characteristics, attributes and means of measurement used in the evaluation

3.4.2.4 Steps Four and Five: Detailed Requirements and Metrics

The quality characteristics and the respective sub-characteristics that have been chosen in the third step are not yet measurable. In order to be measured in a valid and reliable way, they need to be further specified, until measurable attributes or sub-attributes are obtained. Figure 23 sums up these characteristics and attributes, as well as their means of evaluation.

Step five of the EAGLES 7-step approach is dedicated to the design of metrics applied to these measurable attributes. This research project uses two types of evaluations: an objective evaluation concerning the time participants need for the translation, number of clicks and virtual keys, as well as if the tools display all the translatable text, their ease of editing, and the consistency and spell-checking features they offer. Furthermore, the participants of the experiment are going to give a satisfaction rating for some of the systems' attributes by answering a questionnaire for each tool.

The SVG localisation questionnaire (cf. Appendix XII and Appendix XIII) employs mixed types of questions. Some of them use a binary scale of the yes/no-type, where zero points are given for an unsatisfactory score and one point for a satisfactory one. Some are scale questions (Oates, 2006) ranging from one to five, with one being the worst and five being the best score, awarding zero points for an unsatisfactory score, one point for a satisfactory score and two points for a good score. This follows the lead of Canelli et al. (2000) by giving the total number of points if the score is good, half the available points if it is acceptable and none if the score is poor. The rest of the questions are open-ended and encourage the users to give a complete and detailed answer. The way these scales are applied varies for each attribute defined in the task model and is explained further on in this section.

A summary of the metrics and scores used in this study is provided in Table 22 (cf. Appendix XVII). Concerning the attributes that are measured with a binary score, both tools receive one point in case of a tie.

3.4.2.4.1 Measurable Attributes of the Characteristic 'Functionality'

The evaluation of the functionality of the two programs is based on measurable attributes concerning the functions that satisfy the users' needs (**suitability**) and how accurate the tool is (**accuracy**). It is necessary to state what these needs are in order to analyse the functions.

The **suitability** sub-characteristic is broken down into the following sub-attributes:

- **Display of translatable content**: Users of the two programs need to be able to see all the translatable content for the translation to be complete. Thus, the tool must give easy access to the whole translatable content by displaying all the translatable text in the source language English, even if it is hidden content used for accessibility purposes.

  This attribute is measured by feature inspection. It is not acceptable if the program does not show the whole or hidden text, therefore, any missing text content results in an unsatisfactory score (zero points). If the tool displays all translatable content, including metadata that is not seen in the picture, but included in the source code, it is awarded a satisfactory score (one point). If both visible text content and metadata are instantly accessible, without needing extra clicks to select further features, the tool is awarded a good score (two points). Overall, the tool can achieve a maximum of two points for the display of translatable content.

- **Ease of editing**: The possibility to modify translated text is important for the target users. The tool has to allow them to view the source image, as well as the localised image, and to edit translated text.

  It is of utmost importance that the translator or localiser verify the translation and view the final product. Modifications might be needed, in case the translated text does not fit the space allotted to the source text. The possibility to modify translated text without having to use another tool (e.g. to view the graphic) is extremely useful for this task.

  The option of modifying a graphic is evaluated by feature inspection. If the tool offers the possibility to view the SVG file, it receives one point. Furthermore, a tool that offers the possibility to modify translated SVG files (changing the size of

the text or including line breaks, so that the text better fits the picture) without the use of another tool is awarded one point. If no such features are available and the translator or localiser has to use other programs to view the image or to accomplish the task of editing the image, the tool receives zero points respectively.

If the SVG file needs to be edited with a different program after the translation due to space or formatting issues, the tool receives zero points. If no editing with a different program is necessary, it receives one point. Overall, the tool can achieve a maximum of three points for ease of editing.

- **Consistency**: Users need a way to ensure consistency among translations or the possibility to look up words.

  The consistent use of terminology in a translation or in several texts within a bigger translation project is a significant part of the translation process. The verification of consistency throughout a whole project can be time-consuming and difficult to do manually. CAT tools usually offer a way to ensure consistency throughout one or several texts, which can be very advantageous for translators or localisers. This attribute is evaluated with a feature inspection on a binary basis, providing one point, if the tool offers an internal dictionary, term base, translation memories or any other way that helps ensure consistency, and zero points, if no way to ensure consistency exists. Overall, the tool can achieve a maximum of one point for consistency.

- **Spell-checking**: Users need to be able to spell-check the document they are working on.

  During the translation process, a spell-checking function can be advantageous for translators or localisers, since it often catches and draws attention to errors that might easily be overlooked, such as inverted letters. This attribute is also evaluated with a feature inspection, giving one point if a spell-checking feature in the target language exists and zero if it does not. Overall, the tool can achieve a maximum of one point for spell-checking.

The evaluation of the sub-characteristic **accuracy** is determined by measuring the following sub-attribute:

- **Localisation errors**: The final result of the translation process needs to be a fully and accurately translated file without any localisation errors that might be caused due to the nature of the tool itself (e.g. the user cannot view the SVG in SDL Trados Studio 2015 and this lack of context could possibly lead to errors in localisation).

  The accuracy of the final localisation product is measured with a scenario test that determines if localisation errors are found in the translated SVG files. Each localisation error type that can be found in the participants' translated SVG files and that was produced during the translation process equals one negative point for the tool. Five possible localisation error types have been identified: the tool renders a translation with text that is too long and does not fit inside the confines of the picture or overlaps with other elements, maybe even making it unreadable. It constitutes another error, if the translated text is misplaced, meaning that the translation does not correspond to the source text, thus creating senseless phrases. If the style of the text in the source image is not preserved in the target image, regarding font, colour, etc., it constitutes another localisation error. Another error is counted when text content elements have been erased in the translated file. The last error type is given, if translatable content remains untranslated and the program is at fault. The maximum score of 5 points per participant is given, if the tool does not produce any localisation errors. Any localisation error types are subtracted from the maximum score. Overall, the tool can achieve a maximum of 30 points for accuracy.

3.4.2.4.2 Measurable Attributes of the Characteristic 'Usability'

The usability is divided into **understandability**, **learnability**, **operability** and **attractiveness**.

The **understandability** of the tools is measured by evaluating the following attribute:

- **User-friendliness**: In the SVG localisation questionnaire, the participants have to rate if the program is user-friendly and intuitive (questions 1 and 2 with a scale ranging from one to five (one being the worst and five being the best score). The scores one and two are unsatisfactory (zero points), three and four constitute a

satisfactory score (one point) and five is a good score (two points). Overall, the tool can achieve a maximum of four points for each participant (24 points) for understandability.

The **learnability** sub-characteristic is measured by determining how difficult it is to learn how to use the tool with the following attribute:

- **Difficulty of learning how to use the tool**: In the SVG localisation questionnaire, the participants rate the difficulty of learning how to use the tool (question 3) and the supposed difficulty of having to carry out the experiment without any instructions (question 4). They can choose from a scale ranging from one to five (one being the worst and five being the best score). The scores one and two are unsatisfactory (zero points), three and four constitute a satisfactory score (one point) and five is a good score (two points). They are also asked if the tool is appropriate for beginners in SVG localisation (question 5). If the answer is positive, the tool receives one point, if negative, the tool receives zero points. Overall, the tool can achieve a maximum of five points for each participant (30 points) for learnability.

The tools' **operability** is based on the **ease of use**. All functions needed for the translation process need to be easily accessible by being located within as few clicks as possible of the interface where the translation is carried out. It is evaluated with the following attribute:

- **Ease of use**: This operability attribute is measured in the scenario test with a binary score based on the number of clicks and the number of virtual keys that the participants of the experiment need during the translation process to determine the ease of use. Three evaluations are carried out in order to obtain accurate results.

  First, the clicks that are needed to get from the starting point of the interface to the functions that are needed for translation are measured in a best case scenario. The tools received a certain number of points for each participant according to the number of clicks (less than or 2 clicks: 30 points, 3-4 clicks: 24 points, 5-6 clicks: 18 points, 7-8 clicks: 12 points, 9-10 clicks: 6 point, more than 10 clicks: 0 points).

Also, the number of clicks and the number of virtual keys that the participants of the experiment need during the translation process are recorded with the program BB FlashBack Express 5. Again, the tools received a certain number of points for each participant, depending on the number of clicks that were made during the translation process (less than 100 clicks: 5 points, 101-200 clicks: 4 points, 201-300 clicks: 3 points, 301-400 clicks: 2 points, 401-500 clicks: 1 point, more than 500 clicks: 0 points). Points were also awarded depending on the number of virtual keys used during the translation process (less than or 300 virtual keys: 5 points, 301-600 virtual keys: 4 points, 601-900 virtual keys: 3 points, 901-1200 virtual keys: 2 points, 1201-1500 virtual keys: 1 point, more than 1500 virtual keys: 0 points). Overall, the tool can achieve a maximum of 90 points for operability.

The clicks and virtual keys were counted starting from the moment that the main interface of the program was opened until the last time the translated SVG file was saved. The clicks or virtual keys used to open the programs were not counted, since they varied for each participant, because SDL Trados Studio 2015 sometimes asks users for permission to collect data. Aside from that, when the program is started for the first time, a window pops up, asking the user to fill out some personal information before being granted access to the main interface. Normally, this window only appears once, but when the program is opened over the University of Geneva's server, the window always appears. Since all of these steps would not be necessary on a fully configured personal version of SDL Trados Studio 2015, it was decided to skip the steps needed to open the programs and to make the main interface of the tools the starting point for the attribute evaluations based on the translation process.

Attractiveness is measured by determining the **attractiveness of the interface**, the **advantages and disadvantages** of the tools and the **overall satisfaction**, by applying the following metrics:

- **Attractiveness of the interface**: The participants of the experiment rate the attractiveness of the tool's interface in the SVG localisation questionnaire (question 6). They can choose from a scale ranging from one to five. The scores one and two are unsatisfactory (zero points), three and four constitute a satisfactory

score (one point) and five is a good score (two points). Overall, the tool can achieve a maximum of two points for each participant (12 points) for the attractiveness of the interface.

- **Advantages and disadvantages**: In the SVG localisation questionnaire, the participants were asked to list the advantages and disadvantages of the tool during the translation of the SVG image (questions 7 and 8). These questions are open-ended and each advantage awards the tool one point, whereas each disadvantage gives the tool one negative point. In question 9 participants were asked if they would change anything about the tool. Again, each answer receives one negative point.

  In the end, points were given to the tools for each participant according to the number of advantages, disadvantages and changes that were listed. For the advantages, the following scale was applied: more than or 5 advantages: 5 points, 4 advantages: 4 points, 3 advantages: 3 points, 2 advantages: 2 points, 1 advantage: 1 point, 0 advantages: 0 points. The results for disadvantages and changes were merged, since both of them indicate a disadvantage. For the disadvantages and changes, the following scale was applied: 0 disadvantages or changes: 5 points, 1 disadvantage or change: 4 points, 2 disadvantages or changes: 3 points, 3 disadvantages or changes: 2 points, 4 disadvantage or changes: 1 point, more than or 5 disadvantages or changes: 0 points. Overall, the tool can achieve a maximum of 10 points for each participant (60 points) for advantages and disadvantages.

- **Overall satisfaction**: Questions 10-12 of the SVG localisation questionnaire measure the overall satisfaction of the user. The tool receives one point respectively, if the user enjoyed working with it and thinks that it is a good tool for the localisation of SVG files and zero points for negative answers to these questions.

  Also, the participants were asked to rate their satisfaction with the resulting translation on a scale from one to five. The scores one and two are unsatisfactory (zero points), three and four constitute a satisfactory score (one point) and five is a good score (two points).

  Overall, the tool can achieve a maximum of four points for each participant (24 points) for the overall satisfaction.

3.4.2.4.3 Measurable Attributes of the Characteristic 'Efficiency'

The efficiency of the programs is evaluated based on the **time behaviour** sub-characteristic. It is measured in order to establish if the tool facilitates a fast and time-efficient translation process, which will save the user time and money. This can be determined by timing the translation process in the scenario test (cf. section 3.4.1) following the metrics below:

- **Time behaviour**: The speed is determined by measuring the time needed for the whole translation process, starting after the main interface of the program has been accessed and ending after the target file, as well as the translation project in the case of SDL Trados Studio 2015, have been saved. The whole experiment is recorded using BB FlashBack Express 5, which captures all on-screen work and therefore allows measuring the speed of the translation.

  The tools were given points for each participant according to the minutes they needed to complete the translation process (more than or 10 minutes: 5 points, 10-15 minutes: 4 points, 15-20 minutes: 3 points, 20-25 minutes: 2 points, 25-30 minutes: 1 point, more than 30 minutes: 0 points). Overall, the tool can achieve a maximum of 5 points for each participant (30 points).

3.4.2.5 Criteria Weighting

Aside from the design of the metrics to be applied to the requirements, the EAGLES 7-step method also calls for a reflection on the relative importance of the evaluated attributes (EAGLES, 1999). Rico Pérez's adaptable model for system performance addresses this point with criteria weighting: "assigning weighted values to the characteristics previously defined so as to decide their relative importance in the evaluation of the system performance" (Rico Pérez, 2001). A weighting has already taken place with the selection of the characteristics that will be evaluated, but it can be made more precise if the separate attributes are assigned a weight according to the role they play in SVG localisation.

In order to determine an accurate weighting for the target user class, the participants of the experiment were asked to rate the importance of each attribute in relation to the task at hand: the localisation of an SVG file. The Criteria Weighting questionnaire (cf. Appendix XIV – Criteria Weighting) is dedicated to obtaining the

weighted values, which were subsequently applied to the evaluation results of each measurable attribute.

3.4.2.6 Steps Six and Seven

The sixth step of the EAGLES evaluation is dedicated to the design of the execution, which is laid out in chapter 4 (Experiment Design). The last step is treated in chapter 5 (Results).

4. Experiment Design

This chapter focuses on the design of the experiment, including information on the general concept (section 4.1), the pilot studies prior to the experiment (section 4.2), the participants' profiles (section 4.3), descriptions of the materials that were used (sections 4.4), the SVG images that were translated in the experiment (section 4.4) and the software that was used for its execution (section 4.6).

4.1 General Concept

The general concept of the experiment was the realisation of an SVG translation task from English to Spanish with SDL Trados Studio 2015 and Inkscape 0.91 respectively by novice translators or localisers with little or no image localisation experience. Upon completion of each task, the participants were asked to fill out a questionnaire to evaluate and describe the experience with each tool. The translation tasks, as well as the mouse clicks, virtual keys and keystrokes were recorded. The evaluation of the tools was based on the data gathered from the screen recordings and the questionnaires.

Spanish was selected as the target language for the experiment, since it is one of the working languages of the author and the courses of the Department of Translation Technology at the University of Geneva usually have many students with an English-Spanish language combination, which facilitated the recruitment of participants for the study.

Regarding the test conditions, the experiment followed a within-subjects design. All participants were tested under the same conditions, since they all tested the use of both SDL Trados Studio 2015 and Inkscape 0.91 for SVG localisation, allowing a comparison of their performance and results for both tools.

Overall, six participants were recruited for the experiment. They were divided into two groups, which were asked to translate an SVG file split into two parts, in order to avoid an undesired improved or degraded performance of the participants due to an order effect. The first group had to translate the first part of the SVG file (cf. Figure 24, Figure 25) using SDL Trados Studio 2015 and the second part of the graphic (cf. Figure 26 and Figure 27) with Inkscape 0.91. The second group had to translate the same SVG files,

but the order of the programs was inverted, meaning that the first part was translated with Inkscape 0.91 and the second part with SDL Trados Studio 2015.

Two pilot studies were carried out before the start of the experiment to rule out any misunderstandings and possible errors in the material and to control the duration and feasibility of the experiment.

The experiments were conducted at the computer room of the Faculty of Translation and Interpreting of the University of Geneva, except for the experiment with P1, which was carried out at the university's library due to the opening hours of the computer rooms. All of the experiments were performed on the faculty's virtual desktop (cf. section 4.6.1) in order to create the same environment for all of the participants and for the tools. SDL Trados Studio 2015 is installed on the University's server and Inkscape 0.91 was also installed on the virtual desktop for the duration of the experiment, so that the participants would use the same system for both tools.

The tests lasted from 70 minutes up to 120 minutes in some cases, even though the envisaged time should not have exceeded 90 minutes, based on the estimates drawn from the pilot studies.

Before the arrival of the participants, the virtual desktop of the Faculty of Translation and Interpreting of the University of Geneva was opened and Inkscape 0.91 and BB FlashBack Express 5 were set up.

Each participant received several information sheets and a USB drive containing the materials for the conduction of the experiment: the SVG files to be translated, the questionnaires that had to be filled out and the translation project for the work with SDL Trados Studio 2015, which already included the source SVG file and the SVG filter needed to recognise the translatable file.

Before the start of the experiment, the general concept of the experiment and the task at hand were explained to the participants. Special emphasis was made on the fact that the quality of their translations would not be evaluated and they were asked not to use dictionaries during the experiment.

The participants started by reading the information sheets and signing the consent form. This was followed by a quick demonstration of the functions used for the translation of SVG files in Inkscape 0.91. The screen recording with BB FlashBack Express 5 was activated subsequently and the participants received the instruction sheets.

First, the participants had to fill out the General Questionnaire. Then, they performed the first translation task, after which they filled out the SVG Localisation Questionnaire for the tool that they had just worked with. Upon completion of the questionnaire, they proceeded to the second translation task, after which they filled out the questionnaire for the second tool. Finally, they filled out the Criteria Weighting sheet.

In the end, the screen recording was stopped and saved and it was verified if the participants had completed all the tasks and saved all the necessary documents.

4.2 Pilot Studies

Two pilot studies were carried out prior to the execution of the experiment with the final six participants (cf. section 4.3). Their purpose was to detect and eradicate possible errors and misunderstandings in the experiment material and execution.

The participants of the pilot study shared a similar profile with the final participants, to ensure that the findings of the pilot studies could be extrapolated to the actual experiment.

4.2.1 First Pilot Study

The first pilot study was performed with the group 1 material, meaning that the participant had to translate the first part of the SVG file with SDL Trados Studio 2015 and the second part with Inkscape 0.91.

At that point of the experiment design, the SVG files for translation were divided differently and an extra paragraph had been added to the second part of the SVG file in order to harmonise the number of words between the parts. Both files contained 163 words.

The first pilot study showed that the experiment took too much time (1h 39min). In order to shorten its duration, the files were divided in a different way and no extra words were added to them.

Prior to the first pilot study, the concept of the experiment intended the screen recording to be stopped at the end of the first translation task and resumed at the beginning of the second translation task. This idea was abandoned and it was decided that the screen recording would be kept running for the whole duration of the experiment, since this would help save time, aside from being more convenient by allowing the participant to carry out the experiment without interruptions.

4.2.2 Second Pilot Study

The second pilot study was performed with the group 2 material, meaning that the participant had to translate the first part of the SVG file with Inkscape 0.91 and the second part with SDL Trados Studio 2015. It was carried out following the adaptations made after the first pilot study.

This final pilot study demonstrated that the time issue from the first pilot study had been effectively resolved, given that the participant needed much less time (1h 10min).

Another issue came to light during the second pilot study: the participant found it very difficult to work with Inkscape 0.91 and had trouble with the execution of the translation task. Given this difficulty for people with little or no image localisation experience, it was decided to include an information sheet on the translation of SVG files using Inkscape 0.91, followed by a quick demonstration of the functions (cf. section 4.4.1.3 Information Sheet SVG Translation Inkscape 0.91).

4.3 Participants in the Experiment

The general profile of the participants was that of students or recent graduates of the MA in Translation at the University of Geneva, who had little or no experience in the field of image localisation.

Novice translators or localisers were chosen as the target users of the study to facilitate the recruitment of participants. The experiment was unpaid, thus relying on voluntary participation. Given the considerable length of the experiment, it would presumably have been difficult to find voluntary participants among localisation professionals.

All of the participants were required to have taken the course 'Traduction assistée par ordinateur (TAO)' at the University of Geneva, which focuses on computer-assisted translation, in recent years. This requirement was necessary in order for the participants to be able to work with SDL Trados Studio 2015, without the need to include an explanation on how to use the CAT tool.

The participants were also required to be Spanish native speakers and to be at least proficient in English. These requirements were used to ensure the homogeneity of the participants in order to minimise the risk of disparity in the time behaviour.

A total of six participants took part in the experiment. They were divided into two groups: the first group had to translate the first part of the SVG file (cf. Figure 24, Figure 25) using SDL Trados Studio 2015 and the second part of the graphic (cf. Figure 26 and Figure 27) with Inkscape 0.91. The second group had to translate the same SVG files, but the order of the programs was inverted (the first part was translated with Inkscape 0.91 and the second part with SDL Trados Studio 2015).

The participants were aged 23 to 30, all Spanish native speakers with an at least proficient English level. Only one participant (P5) stated both Spanish and English as their native language. When the experiment was carried out, 4 out of 6 participants were students (P2, P4, P5 and P6). 2 out of 6 participants were translators (P3 and P4, the latter being both a student and a translator) and P1 identified as a recent graduate. Their language combinations in the MA in Translation at the University of Geneva all included Spanish as their first (native) language. Almost all of the participants had English as their second language, with the exception of P1, who did not have English in their language combination.

All of the participants have taken the course 'Traduction assistée par ordinateur (TAO)' at the University of Geneva, which focuses on computer-assisted translation, in recent years (P2, P3 and P4 in 2013, P5 and P6 in 2015). Only P1 stated that they had taken both this and the localisation course in 2012.

P1 to P4 have taken the course 'Localisation et gestion de projet' at the University of Geneva, which gives an introduction to localisation and project management, in the years 2012 (P1), 2013 (P3) and 2014 (P4). P5 and P6 have not taken this course. P1 took the localisation course again in 2013 as an unofficial guest student. P1 to P4 also shared the same area of specialisation in the MA in Translation: Translation Technologies. P5 was

undecided about the area of specialisation and P6 was undertaking Specialised Translation.

All of the participants were avid users of computers or similar devices, with 5 out of 6 using them at least once a day for 5 or more hours per day and the remaining participant 5-6 times a week for 3 hours per day. 5 out of 6 participants used them for work, education, communication and leisure, and out of those 5, 2 participants also added research to the list. The remaining participant stated education and leisure when asked about the context of use of computers or similar devices. Furthermore, all of the participants cited Windows as the operating system they normally use and one participant added Mac OS to the list.

When asked about their experience in the field of image localisation, one half of the participants stated that they had only academic, but no practical experience and the other half claimed that they had no experience in the field. 4 out of 6 participants claimed that they used CAT tools. The remaining two participants said that they did not use CAT tools, but the rest of their answers indicate that all of the participants use CAT tools at least in an academic context.

Each participant described their experiences with CAT tools as follows:

P1 has used SDL Trados Studio and MultiTerm in an academic and work context, during an internship, and the text alignment tool MyCat on a daily basis during another internship. The participant occasionally works with SDL Trados Studio and has some knowledge of the 2011 and 2014 versions and professional knowledge of SDL Trados Workbench (2007), but is not certified for the tool.

P2 learned how to use MultiTrans Prism and SDL Trados Studio in the university courses, but does not use them regularly, meaning that they do not have practical experience. The participant is not certified for SDL Trados Studio.

P3 started using SDL Trados Studio and MultiTrans Prims for work in a translation agency in 2011, but had taken a few courses on previous versions of SDL Trados Studio before that. The participant has used translation memories for almost all translation projects ever since (mostly working with SDL Trados Studio), and has used memoQ three or four times. The vast majority of projects this participant uses CAT tools for consists of DOC files. More precisely, P3 uses the CAT

tool SDL Trados Studio mostly for DOC or PPT files. The participant has a level 3 certification for SDL Trados Studio 2014.

P4 mainly uses SDL Trados Studio and MultiTrans Prism for freelance translations. Although the participant started using SDL Trados Studio four years ago, they did not use it very often until recently. P4 started using MultiTrans Prism two years ago and, since then, uses it on a daily basis. The participant normally translates DOC and XML files with CAT tools. The participant occasionally works with SDL Trados Studio and holds a level 2 certification for the 2015 version.

P5 started using CAT tools in the computer-assisted translation course 'TAO', but does not really use it for translations. The participant translated DOC, PDF and PPT files during the aforementioned university course. The participant only works occasionally with SDL Trados Studio and has a level 2 SDL Trados Studio 2015 certification.

P6 only has academic, but no practical experience with SDL Trados Studio, gathered from the 'TAO' course at the University of Geneva. The participant has mainly used the tool for DOC files. P6 also possesses level 2 of the SDL Trados Studio 2015 certification.

The results of the general questionnaire state that 5 out of 6 participants do not use image editing software, although P1 has some academic experience with the tool Inkscape, gathered in the localisation course at the University of Geneva in 2013 as an unofficial guest student (the participant had already taken the course the year before, but did not work with Inkscape during this course). P2 stated that they occasionally worked with Inkscape, having used it previously to create simple graphics. P3 had used Inkscape prior to the experiment in the localisation course at the University of Geneva, but did not have practical experience with the tool.

4.4 Materials

This section provides a description of the materials that were used for the execution of the experiment, listed in their order of use.

The information sheets, consent form and instruction sheets (cf. sections 4.4.1 to 4.4.3) were provided to the participants on paper, to allow them to focus solely on the translation task, without having to switch between windows to consult the instructions.

The participants were given the choice of filling out the questionnaires and the criteria weighting (cf. sections 4.4.4 and 4.4.5) on paper or electronically and all of them chose the latter option.

All of the questionnaires were inspired by the questionnaires used by Valeria Siano (2015) and Cristina Peron (2013) in their own Master's theses and adapted to the purpose of this experiment (cf. section 4.4.4).

The consent form (cf. section 4.4.2) was taken with permission in its entirety from Valeria Siano's Master's thesis (Siano, 2015) and was only slightly altered.

4.4.1 Information Sheets

The participants received the following information sheets.

4.4.1.1 General Information Sheet

A general Information Sheet (cf. Appendix II) was given to the participants at the beginning of the experiment. It contains the aim of the research project and a description of the participants' task, as well as an estimate of the duration of the experiment and further information on the participation and the treatment of data.

4.4.1.2 Information Sheet Image Localisation

An information sheet on image localisation (cf. Appendix III) was provided as an introduction to the concept of image localisation. It briefly describes the basics of image localisation and lays out the main differences between pixel and vector images.

4.4.1.3 Information Sheet SVG Translation Inkscape 0.91

This information sheet (cf. Appendix IV) was used to teach the participants the basic principles of translating SVG files with Inkscape 0.91, since they had little or no experience using the tool.

This document was followed by a short demonstration of the functions that were explained in the information sheet. For this purpose, an SVG file containing only text was created (cf. Appendix V). The style of the text resembled the styling of the text in the SVG files that had to be translated during the experiments.

4.4.2 Consent Form

The participants were given a consent form (cf. Appendix VI) that they had to sign, showing their understanding of the conditions of their participation and that their data would be collected for processing.

4.4.3 Instruction Sheets

Four different kinds of instruction sheets were given to the participants (one for each tool and group): Instruction Sheet Group 1 SDL Trados Studio 2015 (cf. Appendix VII), Instruction Sheet Group 1 Inkscape 0.91 (cf. Appendix VIII), Instruction Sheet Group 2 Inkscape 0.91 (cf. Appendix IX) and Instruction Sheet Group 2 SDL Trados Studio 2015 (cf. Appendix X). They contain step-by-step instructions for the experiment, telling the participant exactly how to proceed, where to find the files for translation and the questionnaires, and where to save the results.

The four instruction sheets are very similar in nature. The instructions mainly vary for the translation tasks with SDL Trados Studio 2015 and Inkscape 0.91. The difference between the groups was the order in which the instruction sheets were presented. Group 1 had to translate the first part of the SVG file ('State of Creative Commons_EN_Part 1') with SDL Trados Studio 2015 and the second part ('State of Creative Commons_EN_Part 2') with Inkscape 0.91. Group 2 had to translate the same files, but with an inverted order of the tools. Therefore, the name of the file that needs to be translated changes according to the group and also the instructions describing the beginning and the end of the translation tasks change for each group (e.g. the participant had to notify the experiment director to activate the screen recording before the start of the experiment, but in the second half of the experiment, this was not necessary anymore).

4.4.4 Questionnaires

All of the questionnaires generally follow the criteria for effective questions established by Peterson (2000, pp. 50-59). Questions should be:

- Brief: The questions mostly had 20 words or less, following the recommendation for the ideal length. Questions were only longer when they needed further specification in order to be understandable, and to incite the respondent to give complete answers. The questions' length ranged from 1 to 44 words.

- Relevant: All the questions and their wordings were carefully designed in order to be relevant to the overall questionnaire and its purpose.
- Unambiguous: Words with multiple meanings and words that were likely unfamiliar to the respondents were avoided.
- Specific: The questions were formulated in a clear and specific way. Whenever it was necessary, further specifications were given to make the questions more understandable. At the same time, questions that were too specific to answer were avoided.
- Objective: The questions were worded in an objective way and thus do not include parts of the answer, to avoid any possible influence on the respondent.

4.4.4.1 General Questionnaire

The General Questionnaire (cf. Appendix XI) was used to gather information on the participants' profiles. It consists of 22 questions, featuring both open questions and closed questions. The questions mainly concerned the respondents' experience with image localisation and with CAT tools and image editing software (especially SDL Trados Studio and Inkscape). This questionnaire had to be filled out before the start of the translation tasks.

4.4.4.2 SVG Localisation Questionnaires

The participants were given a questionnaire after each translation task in order to rate their experience with each tool: SVG Localisation Questionnaire SDL Trados Studio 2015 (cf. Appendix XII) and SVG Localisation Questionnaire Inkscape 0.91 (cf. Appendix XIII).

The localisation questionnaires consist of 12 questions for each tool. As mentioned in section 3.4.2.4, they use yes/no-type questions, scale questions and open questions. The questionnaires also provide some lines after each attribute, which are dedicated to further comments on each attribute and to final remarks.

4.4.5 Criteria Weighting

At the end of the experiment, the participants received a sheet used to determine the importance of each attribute used in the evaluation on a scale from one to five, with one being the least important and five being the most important (cf. 3.4.2.5 Criteria Weighting and Appendix XIV).

4.5 SVG Files for Translation

The SVG files for the experiment were procured via an Internet search. The aim of the search was to find static images in SVG format using a Creative Commons license with translatable text of a general character and containing approximately 200-300 words.

The length of the text was determined following a recommendation of Sharon O'Brien (2009) for similar kinds of experiments involving eye-tracking. She argues that "[t]he capacity of one researcher to thoroughly analyse the amount of data that translation process methods can produce is limited" and that "[t]he reasons for 200-300 word texts are numerous and valid", one example being that "participants could get tired or bored quickly, leading to a drop in motivation and an effect on the data" (O'Brien, 2009, p.261). Furthermore, images with this number of words show more continuity and context than images containing very little text. They also often present an interesting localisation challenge, depending on the language combination: adjusting the translated text, so that it fits the given space in the picture. Translations that are longer than the original text might need to be shortened or edited to achieve more space to avoid localisation errors, such as overlapping elements.

The reason for choosing Creative Commons licensed pictures is that they are free to the public and would not pose any issues of copyright infringement.

Another criterion was for the images to be static, rather than animated. Even though the filter created for SDL Trados Studio 2015 can be used for the translation of animated SVG files, they would have further complicated the experiment and deviated attention from the main focal point: the translation process.

The last criterion was to find images containing text of a general character, meaning that it could not be too specialised or technical. This factor was indispensable for obtaining an objective evaluation of the experiment. In order to evaluate the time behaviour of the translation, the participants could not spend a long time thinking about

how to translate a word, especially since participants were asked not to consult dictionaries. If the content of the images had been too difficult to understand or the sentences too complex, the objectivity of the findings might have been compromised. The readability of the SVG files was verified with a readability test tool[28], which gives a score based on the following readability indicators: Flesch Kincaid Reading Ease, Flesch Kincaid Grade Level, Gunning Fog Score, Coleman Liau Index and Automated Readability Index (ARI).

The search for SVG files meeting the aforementioned criteria proved to be quite difficult, since the images had to incorporate a sufficient amount of translatable text. In order to meet this criterion, the search was centred on obtaining infographics of the SVG file format. The results turned out to be quite limited and the search had to be carried out with very precise keywords, given the vast majority of SVG tutorials and templates for the creation of infographics on the Internet.

The following keywords provided the most useful results: 'SVG infographic –template –tutorial –create', 'SVG infographic Mozilla', 'SVG infographic Creative Commons' and the search of SVG graphics on the website commons.wikimedia.org.

Even though some of the pictures that were found complied with most of the established criteria, the search also rendered many SVG files using multilingual text, which is directly embedded in the source code with a <switch> element (cf. section 2.4).

Although it would have been possible to include such images in this research project, the design of the experiment would have required much adaptation to accommodate multilingual files. The filter for SDL Trados 2015 was conceived in a way that would only extract the text in English, but the participants would still have had to be provided with a browser or any other program in English in order to correctly view the English source text. The group using Inkscape 0.91 for the translation would also have encountered some issues: the language of the user interface would have had to be changed to English to correctly view the source text. When an SVG is translated with SDL Trados Studio 2015 or with Inkscape 0.91 (using the feature "Create and edit text objects"), the file's 'systemLanguage' attribute is erased. The English version disappears from the

[28] http://read-able.com/

source code and a new text content element without any indication of the system language (or other attributes) is created, leading to an undesirable loss of information in the file.

In a real life situation, a translator or localiser receiving a translation order for a multilingual file, would have to use a very different approach by manually adding the target language to the source code. This procedure ensures that the source language still appears in the code, instead of just replacing the source language with the target language.

Ultimately, the use of multilingual SVGs was ruled out because of all of the aforementioned difficulties that would have unnecessarily complicated the translation process.

Given the difficulties with multilingual SVGs and the abundance of SVGs containing little text, the final choice focused on two SVG images: an infographic about Mozilla in 2013[29] and another infographic about the state of Creative Commons (Lemonly, 2014). The former contains 153 translatable words and the latter 265 words. Ultimately, the Mozilla infographic was not chosen for the experiment, since the readability could not be adequately determined. The readability test only evaluated the first 200kB of the file and attributed a grade level of about 21, meaning that the content was not very readable. The readability test based on the text content of the Creative Commons infographic, however, achieved a grade level of about 14, making it easily understandable for 19 to 20 year olds. The Creative Commons infographic fulfilled all of the established criteria, which made it an adequate SVG file for the experiment.

Figure 24, Figure 25, Figure 26 and Figure 27 show the final SVG that was used for the translation task (Lemonly, 2014)[30]. The image was divided into four separate figures for better viewing, since the SVG image was too long to be included in one piece.

[29] https://static.mozilla.com/moco/en-US/images/mozilla_eoy_2013_EN.svg

[30] The source code of the SVG file can be downloaded at: https://stateof.creativecommons.org/report/

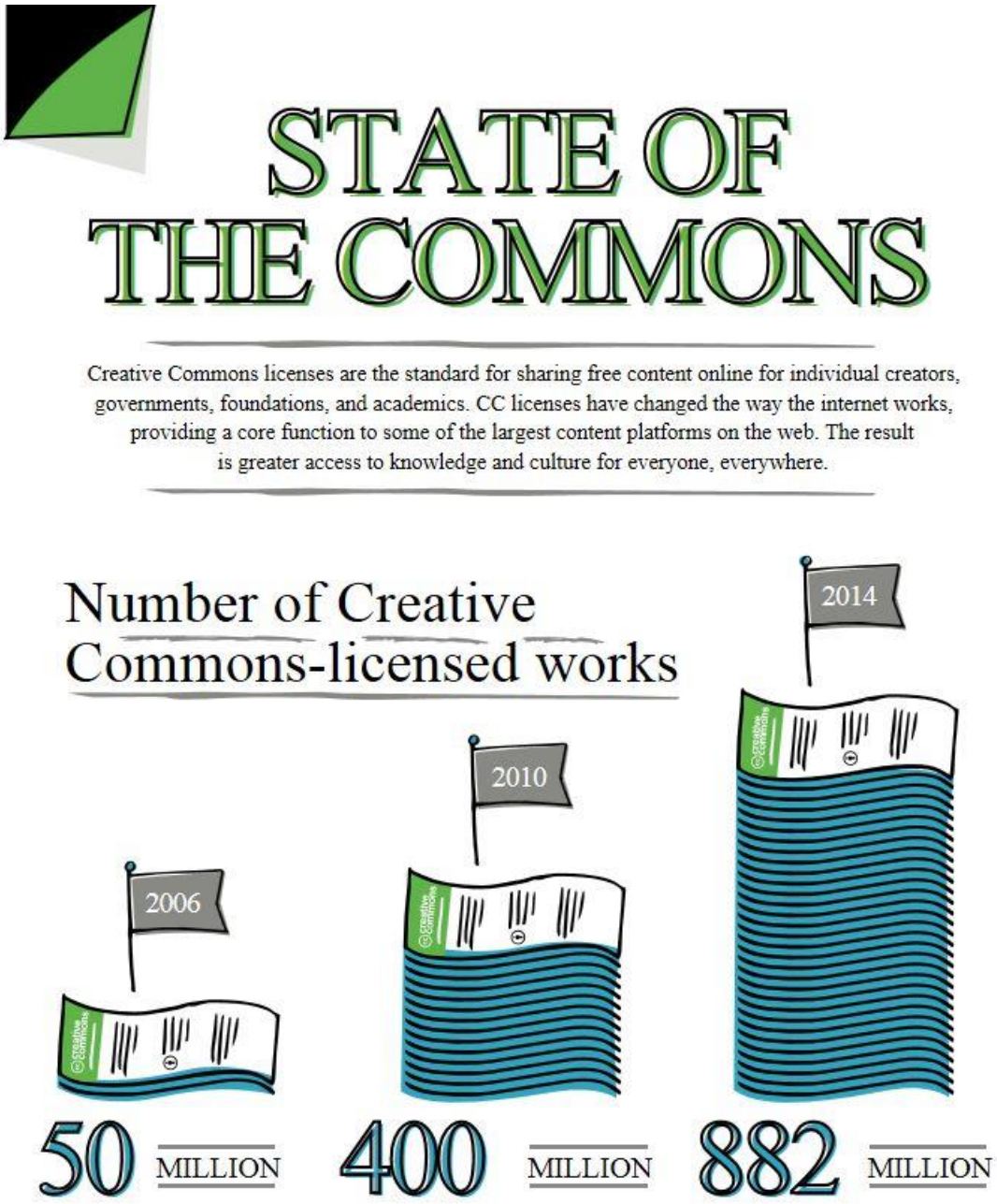


Figure 24. First part of the final SVG used in the experiment

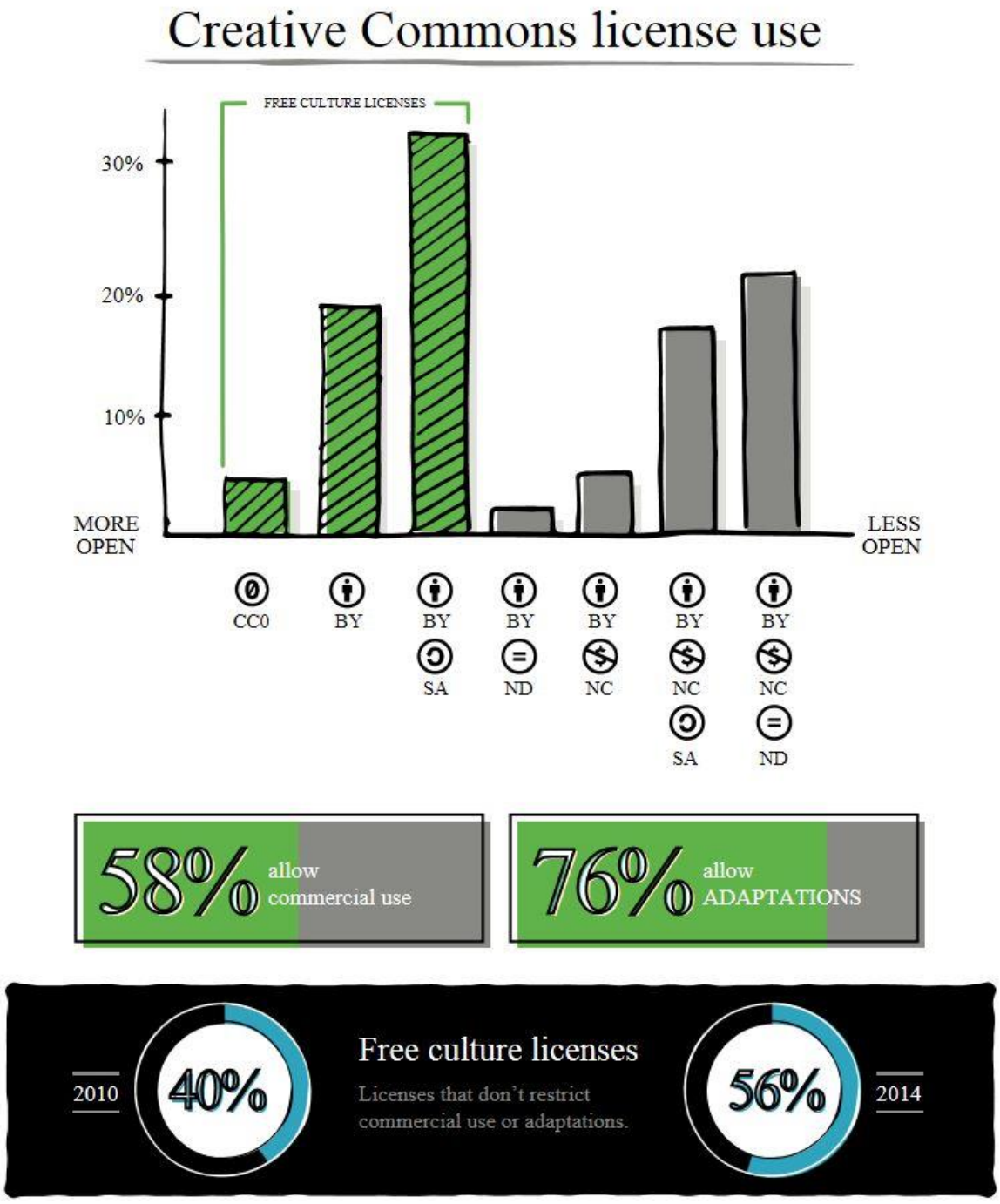


Figure 25. Second part of the final SVG used in the experiment

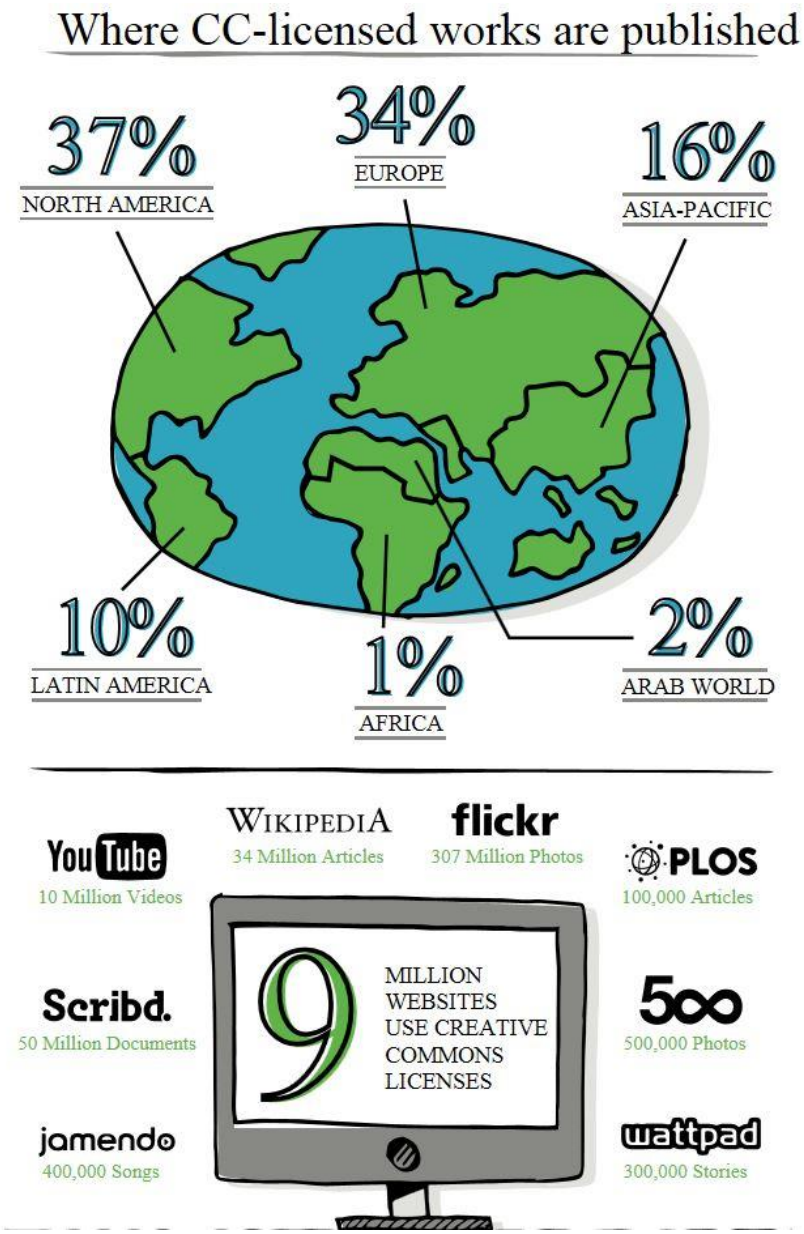


Figure 26. Third part of the final SVG used in the experiment

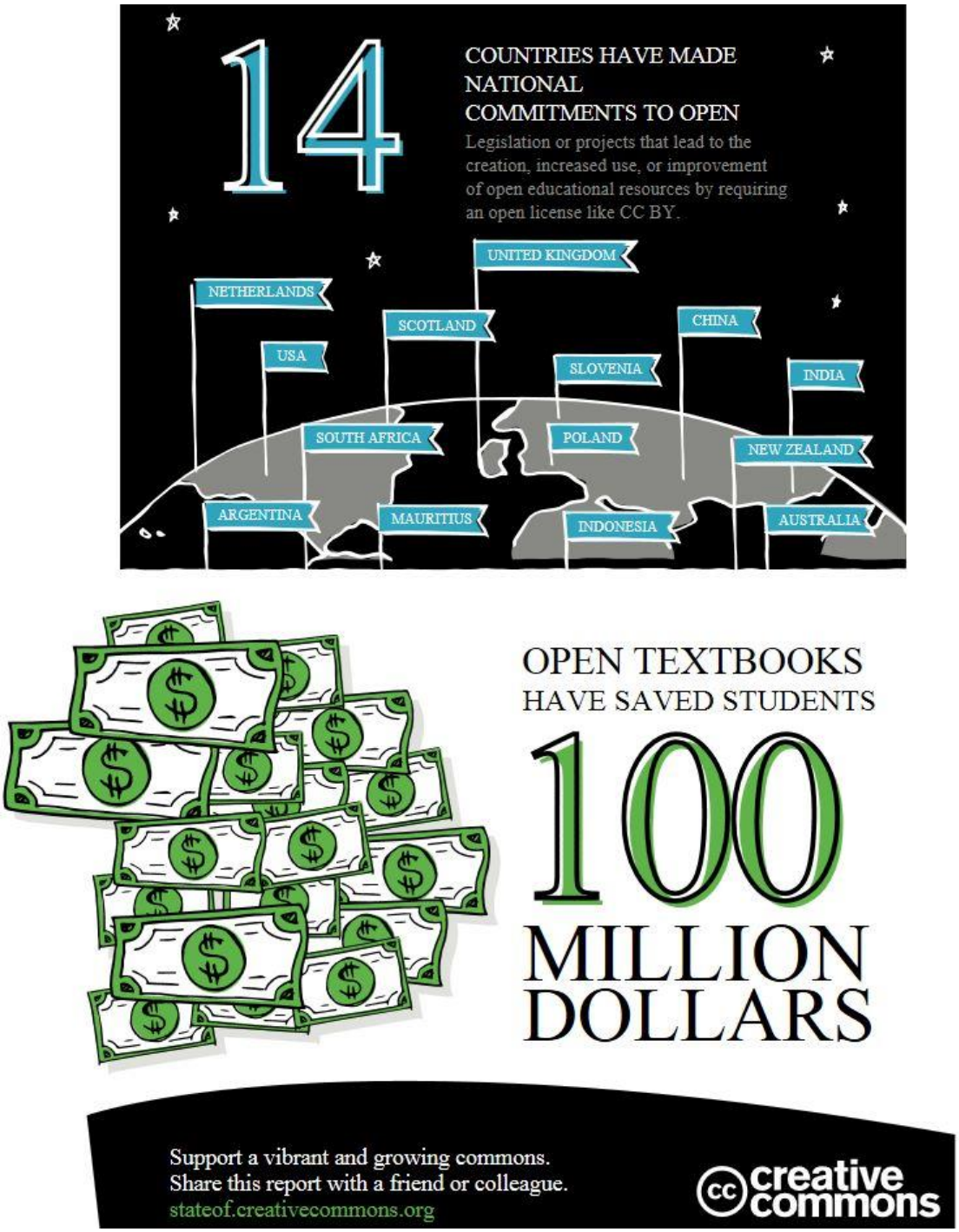


Figure 27. Fourth part of the final SVG used in the experiment

This final SVG file was essentially not adapted for the use in the experiment in order to preserve the real life effect obtained by using an actual infographic in SVG format. Only one slight alteration had to be made: two text content elements of the file were outdated and had to be changed. The <flowPara> element, which figured in the 2004 W3C SVG 1.2 Working Draft, but was later not included in the final specification, was used in the source code. Inkscape 0.91 recognises this element, but it is not supported by browsers, which means that these elements are not displayed when opened in a browser. Furthermore, the SVG filter that had been created for SDL Trados Studio 2015 did not include this outdated element, so it would not have been displayed in SDL Trados Studio 2015. These two elements were converted to current text content elements with Inkscape 0.91, making them viewable with browsers and translatable in SDL Trados Studio 2015.

The SVG file was then split into two parts in order to obtain two images (one for each group in the experiment) that are as equal as possible concerning their subject and readability. The first half of the image contains only slightly more text than the second half (137 words in the first part and 133 words in the second part). With a disparity of only 4 words between the SVG files, the graphics were similar enough in length to be used for the comparison of the tools.

The readability test with the same readability test tool[31] showed only a slight disparity of one grade level between the two parts of the picture. The first part obtained a grade level of about 14, which should be easily understood by 19 to 20 year olds, and the second part obtained a grade level of about 13, easily understood by 18 to 19 year olds. It was decided that a difference of one grade level was negligible and that the readability was still similar enough to facilitate an accurate comparison of the tools' characteristics.

4.6 Software

This section only presents the software that was used during the experiment to obtain results for the evaluation. The tools at the centre of this study, SDL Trados Studio 2015 and Inkscape 0.91, are presented in the previous chapter (cf. section 3.1).

[31] http://read-able.com/

4.6.1 VMWare Horizon View

The free software VMWare Horizon View[32] was used to connect to the University of Geneva's virtual desktop, thus giving the participants access to the tool SDL Trados Studio 2015. This software also made it possible to create the same environment for all of the participants. All of the tasks of the experiment were carried out on the virtual desktop of the Faculty of Translation and Interpreting of the University of Geneva.

The system properties of the virtual desktop were the following when the experiment was carried out:

Operating System: Windows 7 Professional

Processors: Intel(R) Xeon(R) CPU X650 @ 2.67 GHz (2 processors)

Installed memory (RAM); 3,00 GB

System type: 64-bit Operating System

4.6.2 BB FlashBack Express

The analysis of the experiment was partly based on screen recordings that were made using the screen recording software BB FlashBack Express version 5.11.0[33]. The recorder and player are freeware and the software allows users to change a range of settings for the screen recording, adapting them to their individual needs.

For the purpose of this evaluation, the options were set to record full screen and to capture keystrokes, including modifiers, such as 'Ctrl', 'Shift' or 'Alt' keys.

The files obtained with the BB FlashBack Express Recorder were then evaluated with the help of the BB FlashBack Express Player, by analysing the time behaviour, mouse clicks and virtual keys, and by drawing conclusions from the participants' on-screen work whenever it was necessary.

The virtual keys and keystrokes were extracted in the BB FlashBack Express Player, which offers the possibility to export a key log in XML format, grouping the normal keys and the virtual keys and including the time stamps for each keystroke.

[32] https://www.vmware.com/products/horizon-view/

[33] http://www.bbsoftware.co.uk/bbflashbackexpress/home.aspx

5. Results

This chapter presents the evaluation results of SDL Trados Studio 2015 and Inkscape 0.91 in relation to SVG localisation. As can be seen in Figure 23, the results were obtained by feature inspection, as well as by scenario test and questionnaires from the experiment described in the previous chapter. The evaluation follows the metrics established in chapter 3 (cf. section 3.4.2.4).

The findings of the feature inspections, which were carried out to determine the **suitability** sub-characteristic, are presented in section 5.1. Scenario tests were used for the **accuracy** (section 5.2), **operability** (section 5.5) and **time behaviour** (section 5.7) sub-characteristics and the results gathered from questionnaires were used to evaluate the **understandability** (section 5.3), the **learnability** (section 5.4) and the **attractiveness** (section 5.5) of the tools.

In the end, the chapter provides the reader with information on the criteria weighting that was applied to the results (section 5.8).

5.1 Suitability

Suitability is the "capability of the software product to provide an appropriate set of functions for specified tasks and user objectives" (ISO/IEC 9126-1, 2001). This sub-characteristic was measured with an evaluation of the attributes 'display of translatable content', 'ease of editing', 'consistency' and 'spell-checking'.

5.1.1 Display of Translatable Content

The tools' capacity of displaying translatable content is determined by feature inspection with a comparison carried out by the author based on the tools' functions.

As mentioned in section 3.3.1, **SDL Trados Studio 2015** does not recognise Scalable Vector Graphics without a custom filter that has to be either created or imported. Nevertheless, it is fairly easy and fast to create a filter that allows the program to recognise SVG files and to display the translatable content. Once a filter has been added to the default settings of SDL Trados Studio 2015, it will not be erased, meaning that the user will only have to create or import the SVG filter once to be able to translate SVGs with future projects.

Since this step only needs to be done once, this research project does not consider that SDL Trados Studio 2015 is incapable of displaying translatable content. An accurate SVG filter, such as the one provided under section 3.3.2, displays all visible text content of an SVG file (<text> elements and their children; <a> elements for links), as well as hidden content that is used to describe the picture (the <desc> and <title> elements, which should also be translated for accessibility reasons). In concordance with the established metrics, SDL Trados Studio 2015 receives the maximum score of **two points (good score)**, since the settings of the parser rules allow hidden text to be displayed alongside visible text in the 'Editor' view, without needing further clicks to translate the hidden elements.

It is important to mention that SDL Trados Studio 2015 fails to recognise text that has not been placed inside a text content element by the image creator, but is represented graphically instead (with shapes or paths). Usually, these kinds of elements are used for logos that do not have to be changed. They are excluded from the evaluation, since these elements are normally not meant to be changed and are therefore not regarded as translatable elements for the purpose of this evaluation (cf. section 3.3.3).

In the case of **Inkscape 0.91**, visible text is displayed effortlessly, but hidden text, such as the description and title, need to be accessed by clicking on 'File' and selecting the 'Document Properties…' or with the shortcut 'Shift+Ctrl+D'. The title and description can be altered under the tab 'Metadata'. Accessing this hidden text content requires three clicks, or the use of the shortcut plus one click to access the 'Metadata' tab. Following the previously established metrics, Inkscape 0.91 is given **one point (satisfactory score)**, since visible content is displayed, but hidden content is not instantly accessible in the main interface.

Overall, **SDL Trados Studio 2015** receives a final score of **two points** for the capability of displaying translatable text, whereas **Inkscape 0.91** is given a final score of **one point**.

5.1.2 Ease of Editing

This attribute is determined with a feature inspection based on the tools' ability of letting the user view and edit the SVG file.

SDL Trados Studio 2015 successfully extracts the translatable elements, but it is not able to display the whole Scalable Vector Graphic and the translator or localiser has

to use another program, such as a browser, to view the source and target files. SDL Trados Studio 2015 does offer a preview function, but in the case of SVG files, it only displays the source code and not the whole image, thus failing to provide context for the translation. This can be observed on the right-hand side of Figure 28, which shows how the SDL Trados Studio 2015 preview function displays the text. The tool receives **zero points** due to its incapability to display SVG files.

It is not only important for the translator or localiser to be able to view the text content, but also to edit it. SDL Trados Studio 2015 allows the translator or localiser to modify purely the text content of the SVG file and it does not let the user merge or split segments, as can be seen in Figure 28. This makes it difficult to correctly translate a file, since the word order of some text content elements might need to be changed in the target language due to space issues. The experiment has shown that it was challenging for the participants to provide a translation that would fit into the allotted space and at the same time respect the segments shown in SDL Trados Studio 2015. A phrase was often divided into different segments, following the original division of the text into several text content elements. The translation would often be too long or the word order had to be switched between segments, resulting in a translation that would not correspond to the source text. Consequently, another tool would have to be used to shorten the text by changing the size of the text content element, by adding line breaks or by using other means of editing. The tool is given **zero points** for the possibility of modifying an image without the use of another tool and again **zero points**, because the SVG file needs to be edited with another program after translation due to space or formatting issues.

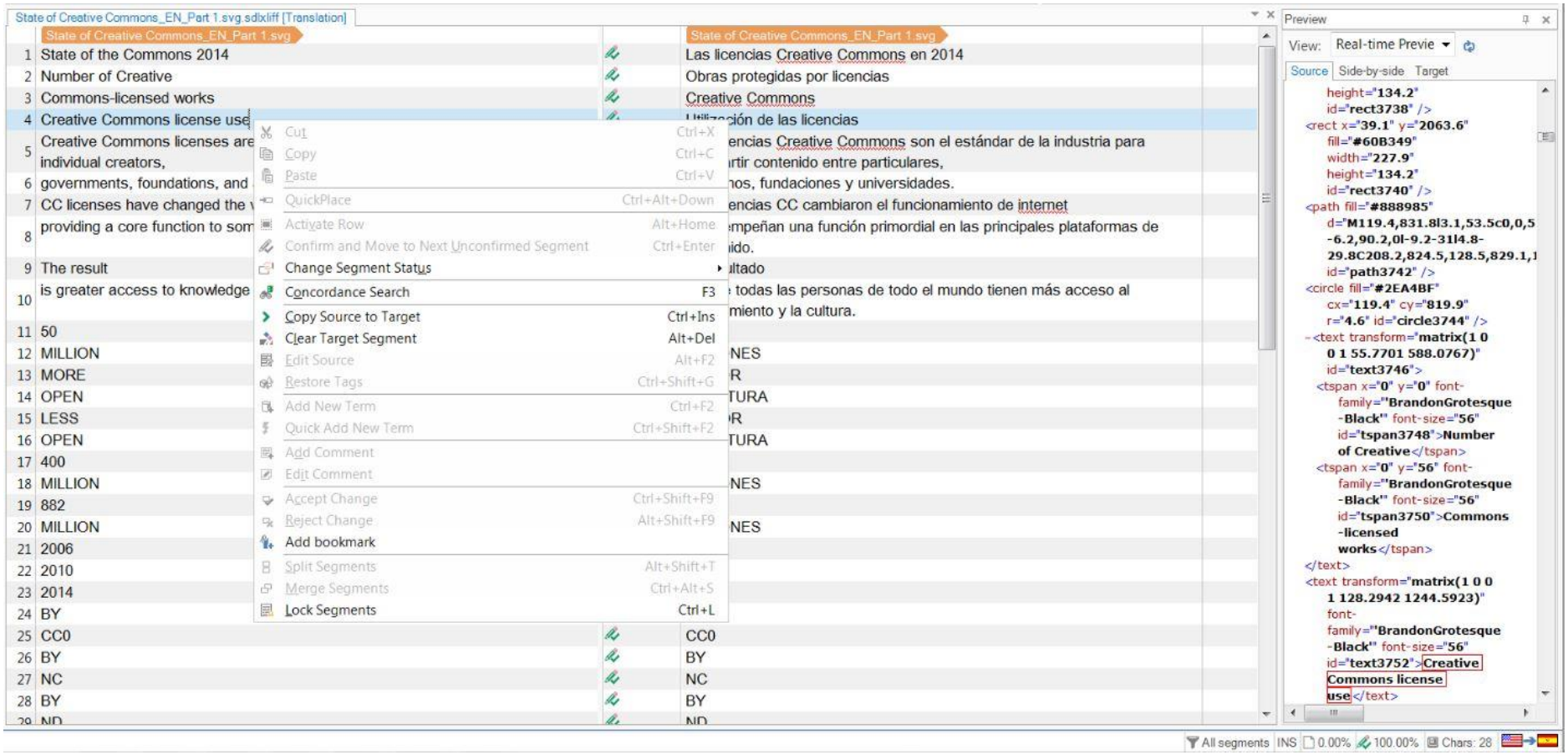


Figure 28. Excerpt of the translation of P3 in the SDL Trados Studio 2015 Editor

In **Inkscape 0.91**, the user can see the whole vector graphic with all of its elements. Therefore, the tool is given **one point** for giving the user the possibility to view the image.

Being a vector graphics editor, the main purpose of Inkscape is to allow the user to create and modify vector graphics, leading to **one point** for the tool concerning the possibility to edit SVG files without the use of another tool.

Since Inkscape 0.91 allows the user to edit the image with the same tool, there is no need to fall back on another tool to correct space or formatting issues, since the user can correct them throughout the translating process. The tool is given **one point**, because the SVG file does not have to be edited with another program after translation due to space or formatting issues.

All of the aforementioned capabilities of Inkscape 0.91 can be viewed in Figure 29, which shows the interface of Inkscape 0.91 with an excerpt of the first part of the SVG image used in the experiment. All of the elements of the SVG file are displayed, which allows the translator or localiser to know the context of the source text. On the left-hand side of the tool, the user can select from several functions used to edit the image, thus eliminating any issues that might have to be corrected after translation.

Overall, **SDL Trados Studio 2015** receives **zero points** for ease of editing, whereas **Inkscape 0.91** is awarded the maximum score for this attribute with **three points**.

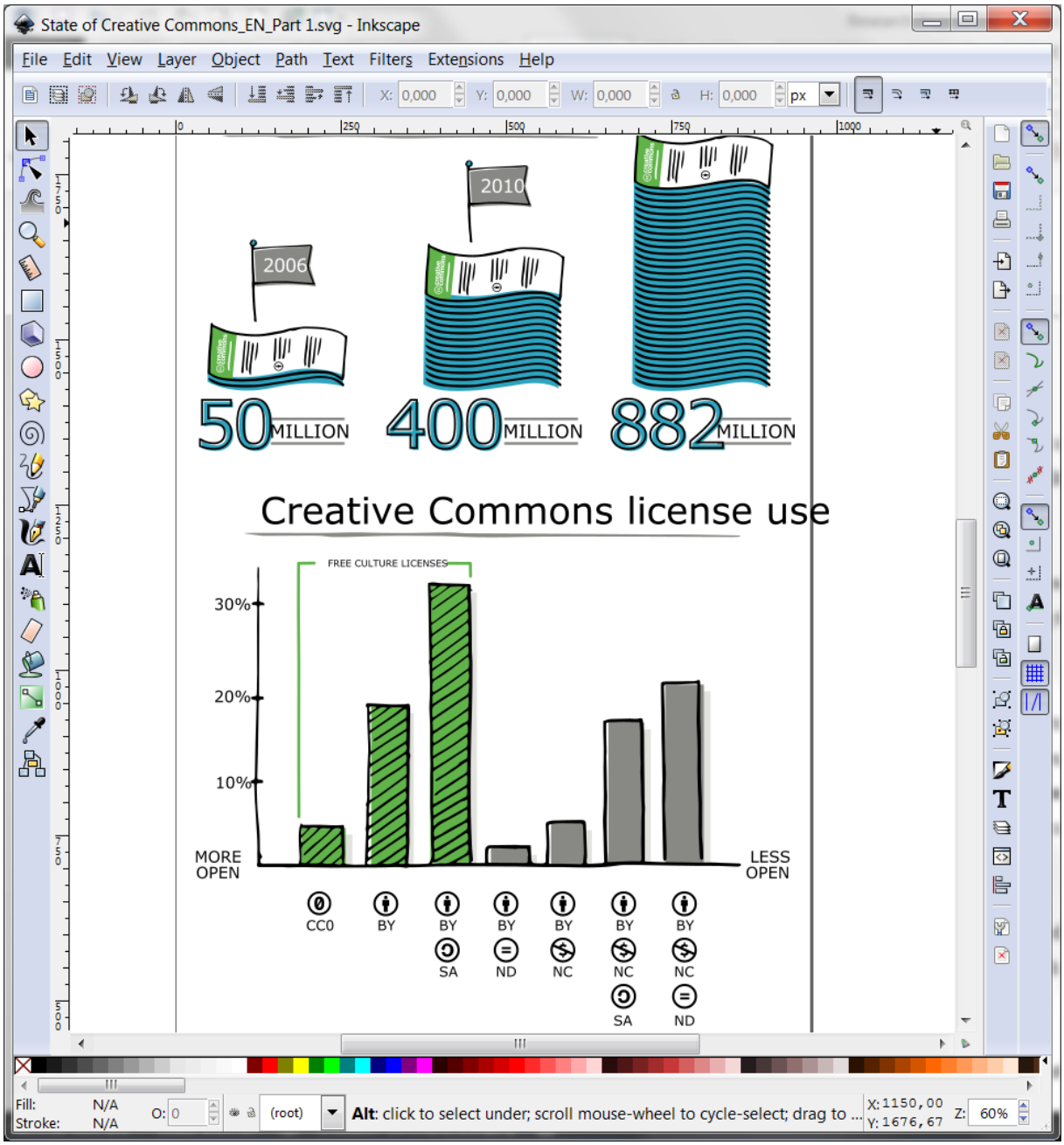


Figure 29. Screenshot of an excerpt of the first part of the SVG file used for translation opened in Inkscape 0.91

5.1.3 Consistency

The capacity of the tool to ensure consistency throughout the translation of one or several SVG files is evaluated by feature inspection.

The advantages of CAT tools are widely known and appreciated by translators and localisers and do not need to be further explained. One of these advantages is the possibility to maintain terminological consistency.

SDL Trados Studio 2015 offers several ways to ensure consistency throughout the translation. The user can work with translation memories and the integrated terminology management tool SDL MultiTerm, which can be fed with the appropriate reference texts or terminology for a specific project or can be updated on the go. The tool also allows the user to effortlessly implement terminology guidelines given by the client by integrating these guidelines into translation memories or terminology databases. Given the aforementioned options for maintaining consistent terminology throughout one or several translations, the tool receives **one point** for consistency.

The vector graphics editor **Inkscape 0.91** does not primarily cater to the needs of translators or localisers, since its focus does not lie on editing only text content elements, but on editing every element of the vector graphic. Seeing that translators or localisers are not the tool's primary clientele, it does not offer the same functions as CAT tools when it comes to maintaining consistency. Since there is no possibility of using translation memories, customised dictionaries, term bases or any other means that would ensure a consistent use of terminology, the tool receives **zero points**.

Overall, **SDL Trados Studio 2015** receives a final score of **one point** for consistency, whereas **Inkscape 0.91** is given a final score of **zero points**.

5.1.4 Spell-checking

A feature inspection is used to determine if the tool offers a spell-checking function.

Being a CAT tool, which is adapted to translators' needs, **SDL Trados Studio 2015** offers a seemingly good spell-checking function for the target language, which marks words while translating in the 'Editor' with a jagged red underline and suggests different options of how the word might be written when the underlined word is right-clicked. Also,

during the translation process or upon its completion, the translation can be spell-checked by clicking on the 'Review' tab and selecting the 'Check Spelling' feature or by pressing F7.

SDL Trados Studio uses Hunspell dictionaries for a number of languages by default and users can import dictionaries, assuring a working spell-checking function for a very wide range of target languages (SDL PLC, 2015f). The user can also opt to use a Microsoft dictionary, if Word is installed. Given this range of possibilities, SDL Trados Studio 2015 is awarded **one point** for the spell-checking attribute.

Inkscape 0.91 offers a spell-checking feature, but it is only limited to some variants of the English language, which can be accessed by selecting 'Edit' and then 'Preferences…' or pressing 'Shift+Ctrl+P'). An SVG file with English text can be spell-checked by accessing the tab 'Text' and clicking on 'Check Spelling…' or by pressing 'Ctrl+Alt+K'. Inkscape 0.91 does not offer the possibility to import dictionaries in Windows, which makes the spell-checking function useless for translators or localisers using Windows whose target language is not English. Currently, it is only possible to include dictionaries in Linux, by installing Aspell and the dictionary for the desired target language. Statistics[34] show that Windows is the most frequently used operating system, meaning that the spell-checking feature is not accessible to the vast majority of users. For the purpose of the evaluation of this feature, this research project focuses on spell-checking in the operating system used during the experiment (Windows) and the target language of the experiment (Spanish). Since no Spanish spell-checker is included in the Windows-version of Inkscape 0.91, the tool is given **zero points**.

Overall, **SDL Trados Studio 2015** receives a final score of **one point** for spell-checking, whereas **Inkscape 0.91** receives a final score of **zero points**.

[34] https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0

5.2 Accuracy

Accuracy is the "capability of the software product to provide the right or agreed results or effects with the needed degree of precision" (ISO/IEC 9126-1, 2001). This sub-attribute was measured with the attribute 'localisation errors'.

5.2.1 Localisation Errors

A scenario test was used to determine the localisation errors found in the translated files of the experiment's participants. Table 3 marks the presence of a certain kind of localisation error in the translated SVG files of each participant. A comparison of the results shown in Table 3 make it clear that certain types of localisation errors prevail in each program.

All of the translations made with SDL Trados Studio 2015 had text elements that were too long and did not fit inside the confines of the image or overlapped with other elements. It is safe to assume that this issue arose, because the segments that the tool creates (which coincide with the elements of the SVG file) cannot be separated when translating SVG files. Therefore, it is not possible to insert line breaks to divide the text. The only way of avoiding this issue would be to either edit the translated file with another tool or to shorten the translation, which might result in a loss of information.

Five out of six translations made with Inkscape 0.91 presented the same issue of text elements that were too long or overlapped. The screen recordings have shown that the participants often did not try to resolve this issue, since it was not part of their task, after all. Other times, the participants tried to insert line breaks to separate the text, but for some text content elements, this turned out to be a rather complex task, since the program created a new text content element on top of the existing element, instead of adding a line break. Only P4 produced a translation without space issues, but it is important to point out that this participant took the longest for the translation task with Inkscape 0.91 due to the editing of lines that overlapped or that were too long (cf. section 5.7).

Another localisation error was the creation of senseless phrases with SDL Trados Studio 2015 in 4 out of 6 occasions. This can most likely be attributed to the segmentation of the text and the impossibility to merge segments, as well as to the fact that SDL Trados Studio 2015 does not display the source image. Since the tool extracts text directly from

the SVG file's text content elements, it also adopts the division of text set out by the SVG's creator. Each segment corresponds to a text content element, but these elements do not necessarily have to contain whole phrases.

Furthermore, SDL Trados Studio 2015 extracts the text in the order of the presence of text content elements in the SVG file's source code. This order does not necessarily have to follow the sense in which the text is displayed in the vector graphic, since the location of each element is determined with coordinates. This means that the image creator can include the end of a phrase in a text content element that is placed before the beginning of the phrase, as long as the coordinates arrange the text in its intended word order.

In combination with the added difficulty of having to consult the source file to find out the location of the segments in the SVG file, the participants did not always realise which segments belonged together to create one phrase. It was a challenging task for the participants to translate the divided segments, since they did not appear in the right order. This apparently made it difficult for them to see the context and realise which segments belonged together, resulting in nonsensical translations. Figure 30 shows an excerpt of the SDL Trados Studio 2015 translation of P2, which illustrate the word order issue. Segments 65, 66 and 67 all form the beginning of a phrase, but these three segments appear at the very end of SDL Trados Studio 2015's 'Editor'. The rest of the phrase appears in segments that are located above the beginning of the phrase.

Figure 30. Segments with a wrong word order in SDL Trados Studio 2015 and their graphical representation in the translated SVG file (translation provided by P2)

This localisation error did not occur in the translations of two participants (P3 and P4). Since P3 edited some of the segments after an inspection of the translated file, it was necessary to determine if any localisation errors had been eliminated in the editing

process or if they did not exist in the first place. The screen recording showed that the participant correctly identified the difficulties of the separated segments and managed to create a translation without any mistranslations. It is important to point out that this participant frequently consulted the source file to get a better idea of the context and location of the segments during the translation with SDL Trados Studio 2015. The constant verification of the context probably led to a correct translation without nonsensical phrases. This interpretation is reinforced by the fact that this error did not occur in the translations made with Inkscape 0.91.

P4 also managed to produce a target file without illogical translations. The screen recording showed that the participant correctly identified that some segments belonged together, even without consulting the source image. As a result, P4 treated some of the segments as one unit and translated the text accordingly, changing the word order whenever it was necessary, or leaving out a segment, because the information had to be added to another segment.

As mentioned above, this localisation error was not present in the translations made with Inkscape 0.91, given the possibility to see the source text when translating with this tool.

The third error type in SDL Trados Studio 2015 was that some content remained untranslated. In the translation of P6, the headline was not translated. The screen recording showed that the participant had translated the segment and then tried to proceed to copy source segments containing numbers into their respective target segments, using 'Ctrl+Ins'. The participant forgot or failed to select the right source segment and accidentally copied the source segment of the headline into the target segment, thus overwriting the translation. Supposedly, the same error appeared in the translation of P4, where only 98% of the text were translated (the country names and part of one sentence were still in English). Even though at first sight this seemed to be a localisation error created by the tool, a look at the screen recording revealed that the participant copied the source segments into the target segments using 'Ctrl+Ins' on each of the untranslated elements. Since the participant selected all of the segments seemingly purposefully, there was no indication that their actions could have been arbitrary. This leads to the assumption that the participant made a conscious choice to not translate the

elements, meaning that the program is not at fault and that the error has not been counted in the evaluation.

In Inkscape 0.91, P2 left some text untranslated and commented that they "didn't enjoy working with Inkscape" and were "not able to finish the translation" in the final remarks of the SVG localisation questionnaire. The screen recording showed that the participant seems to leave parts of an element untranslated on purpose, which means that the tool did not cause this error.

Aside from the space issues, another type of error was present in the translations with Inkscape 0.91. All of the translations showed changes in the text styling, especially changes in colour and font size or in the alignment of the text. The style was kept when participants translated with the 'Create and edit text objects' function, but the screen recordings showed that they had difficulties with the creation of line breaks or with selecting the right element or word to translate. They often switched to the 'Text and Font…' function, which allowed them to easily create line breaks and insert the translation. Nevertheless, this function sometimes altered the style of the text when the changes were applied.

The last localisation error that was identified in Inkscape 0.91 occurred during the translation process of P3. The screen recording showed that the participant erased the contents of a text content element, before proceeding to the translation. Apparently, this was done in order to be able to see the translation. In this case, the text colour was white on a black background, but when the translation became too long, the text would leave the confines of the image. The white text was then displayed over a white background, making it unreadable. Having already erased the source text, the participant seemed to have forgotten part of the translatable content, so the resulting translation was incomplete.

Overall, 11 localisation errors were counted in the translations from SDL Trados Studio 2015 and 12 localisation errors in the translations from Inkscape 0.91. Following the metrics, the tool receives a maximum score of 30 points if no localisation errors are found. The number of errors is subtracted from the maximum score. **SDL Trados Studio 2015** achieved a final score of **19 points** in the evaluation of the localisation errors, whereas **Inkscape 0.91** obtained **18 points**.

	SDL Trados Studio 2015						Inkscape 0.91					
Localisation error	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**
The translated text does not fit in the confines of the image or overlaps with other elements	x	x	x	x	x	x	x	x	x		x	x
The translated text is not in the same place as the source element, creating senseless phrases	x	x			x	x						
The style of the text is not the same as in the source image (font, colour, etc.)							x	x	x	x	x	x
Text content elements have been erased in the translation									x			
Some translatable content has not been translated				(x)		x		(x)				
Score	**11 errors 19 points**						**12 errors 18 points**					

Table 3. Localisation errors committed by participants using SDL Trados Studio 2015 and Inkscape 0.91

5.3 Understandability

Understandability is the "capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use" (ISO/IEC 9126-1, 2001). This sub-characteristic was measured with an evaluation of the attribute 'user-friendliness'.

5.3.1 User-friendliness

The user-friendliness of the tools was measured with two questions (**Q1** and **Q2**) of the localisation questionnaires concerning the user-friendliness. The results of the evaluation are presented in Table 4.

The participants had to rate if the program was user-friendly and intuitive with a scale ranging from one to five (one being the worst and five being the best score). The scores one and two are unsatisfactory (zero points), three and four constitute a satisfactory score (one point) and five is a good score (two points).

Q1 asked the participants to rate if the tools were user-friendly, with features that are easy to use. SDL Trados Studio 2015 obtained a much higher score ($\bar{x}$ =4, s = 0.63) than Inkscape 0.91 ($\bar{x}$ = 2.83, s = 1.17).

In **Q2**, the participants had to rate if the interface of the tools was intuitive. Again, SDL Trados Studio 2015 obtained a higher score ($\bar{x}$ = 4.17, s = 0.75) than Inkscape 0.91 ($\bar{x}$ = 3.17, s = 1.17).

P1 remarked that SDL Trados Studio 2015 is intuitive, but that it has "features that are not that user-friendly, for instance that a segment always start[s] with a capital letter".

Regarding Inkscape 0.91, P2 commented that "previous experience with Inkscape [was] necessary" and P3 that "[i]t's rather difficult to select the text you want to modify and it is uncomfortable to work inside the text box once you're there". P4 thought that it was "user friendly because the icons are similar to those of other editing tools. But the function of each icon sometimes is not so clear, or obliges you to take an extra step". P5 added that the tool "seems to be very [user-]friendly but sometimes it is difficult to work with it and it takes a long time to finish the task".

Overall, **SDL Trados Studio 2015** achieved a final score of **15 points** in the evaluation of the user-friendliness, whereas Inkscape 0.91 obtained a final score of **9 points**.

Participant	Question Number	Points SDL Trados Studio 2015	Points Inkscape 0.91	Score SDL Trados Studio 2015	Score Inkscape 0.91
P1	Q1	4	4	**2**	**2**
	Q2	4	4		
P2	Q1	4	2	**2**	**0**
	Q2	3	1		
P3	Q1	4	1	**2**	**1**
	Q2	4	3		
P4	Q1	4	3	**2**	**2**
	Q2	4	3		
P5	Q1	3	3	**3**	**2**
	Q2	5	4		
P6	Q1	5	4	**4**	**2**
	Q2	5	4		
Final Score				**15**	**9**
Average ($\bar{x}$)	Q1	4	2.83		
	Q2	4.17	3.17		
Standard Deviation (s)	Q1	0.63	1.17		
	Q2	0.75	1.17		

Table 4. Results of the understandability evaluation

5.4 Learnability

Learnability is the "capability of the software product to enable the user to learn its application" (ISO/IEC 9126-1, 2001). This sub-characteristic was measured with an evaluation of the attribute 'difficulty of learning how to use the tool'.

5.4.1 Difficulty of Learning How to Use the Tool

The evaluation of the difficulty of learning how to use the tool is carried out to determine the tools' learnability. It was measured with three questions (**Q3**, **Q4**, **Q5**) of the localisation questionnaires. The results of the evaluation are presented in Table 5.

Q3 had the participants rate the difficulty of learning how to use the tool. SDL Trados Studio 2015 obtained the same average score and standard deviation as Inkscape 0.91 ($\bar{x} = 4$, $s = 0.89$).

In **Q4**, participants had to indicate how difficult it would have been to carry out the experiment without any previous instructions on how to use the tool. Inkscape 0.91 obtained a slightly better rating ($\bar{x} = 2.83$, $s = 1.47$) than SDL Trados Studio 2015 ($\bar{x} = 2$, $s = 0.89$).

Q5 was used to determine if a tool was appropriate for beginners in SVG localisation. 5 out of 6 participants thought that Inkscape was an appropriate tool for beginners in SVG localisation, whereas 4 out of 6 participants thought the same of SDL Trados Studio 2015.

In the comment section on the learnability of SDL Trados Studio 2015, P1 stated that "SDL Trados is an intuitive tool and easy to use. It is recommended, however, to have some previous knowledge of the different features. If not, it can be difficult to perform the task and to get familiarised with the tool". P2 commented that "[i]t allows the person to focus on the text". P4 remarked that the difficulty of learning how to use the tool might depend "on whether students had previous knowledge of [T]rados. For someone that already translates with the system, localising in this way is pretty much the same".

Comments on the learnability of Inkscape 0.91 were that "[t]he user should have some previous knowledge of the tool. As oppos[ed to] Trados, where [it] is recommended to have some knowledge in order to work easily, I think that Inkscape could be more difficult for beginners" (P1). P2 thought that "people need to know how to create images

and texts within the images to after[wards] use it for localisation". P4 commented that "[a]lthough it is annoying the way in which the system changes the place of objects, or the fact that sometimes it can be difficult to select elements that overlap, it can be a goo[d] tool to start learning localisation, since it is very visual and introduces students [to] the first obstacles of localising images and text with layers." P5 added that "[a]lthough Inkscape has its problems[,] I think, without a doubt it is really easy to learn how to use it. I didn't have any experience and I could develop all my tasks pretty easily because the interface is designed in a way that […] makes it clear for you how to do your tasks".

Overall, **Inkscape 0.91** earned a better final score for learnability with a total of **17 points**, in comparison with a total of **14 points** for **SDL Trados Studio 2015**.

Participant	Question Number	Points SDL Trados Studio 2015	Points Inkscape 0.91	Score SDL Trados Studio 2015	Score Inkscape 0.91
P1	Q3	5	4	**3**	**2**
	Q4	2	2		
	Q5	Yes	Yes		
P2	Q3	4	4	**2**	**1**
	Q4	1	1		
	Q5	Yes	No		
P3	Q3	5	3	**2**	**3**
	Q4	1	3		
	Q5	No	Yes		
P4	Q3	4	3	**3**	**3**
	Q4	3	4		
	Q5	Yes	Yes		
P5	Q3	3	5	**1**	**5**
	Q4	2	5		
	Q5	No	Yes		
P6	Q3	3	5	**3**	**3**
	Q4	3	2		
	Q5	Yes	Yes		
Final Score				**14**	**17**
Average ($\bar{x}$)	Q3	4	4		
	Q4	2	2.83		
Standard Deviation (s)	Q3	0.89	0.89		
	Q4	0.89	1.47		

Table 5. Results of the learnability evaluation

5.5 Operability

Operability is the "capability of the software product to enable the user to operate and control it" (ISO/IEC 9126-1, 2001). This sub-characteristic was measured with an evaluation of the attribute 'ease of use'.

5.5.1 Ease of Use

The operability of the tools regarding SVG localisation is evaluated by measuring the ease of use. The clicks needed to access the functions for translation were counted. Then, an analysis of the mouse clicks and virtual keys used during the translation process was carried out with a scenario test based on the recordings of the experiment with BB FlashBack Express 5.

5.5.1.1 Number of Clicks

Two different calculations were applied for the number of clicks: the clicks required to access the functions needed for translation and the clicks made by the participants during the translation tasks of the experiment.

To determine the clicks needed to access the functions for the translation of SVG files in SDL Trados Studio 2015 and Inkscape 0.91, a best case scenario was recreated. The starting point of the click count was the main interface or the welcome screen of each tool.

The evaluation takes into account the path that allows the translator or localiser to use the least clicks without the use of keyboard shortcuts. Double-clicks were counted as two clicks. The results are summarised in Table 6.

First, the clicks needed to open the file for translation were counted. In SDL Trados Studio 2015, it is not possible to simply open an SVG file without the use of a translation project. For the purposes of this evaluation, an already existing translation project was considered as the file to be translated.

In SDL Trados Studio, the user needs 4 clicks to open an existing translation project: one click to open the 'File' menu, another one on 'Open Project', a third click to select the file, and a last click on 'Open'.

In the same manner, the user also needs 4 clicks to open an SVG file in Inkscape 0.91: one click to open the 'File' menu, another one on 'Open', a third click to select the file, and a last click on 'Open'.

The clicks needed to access the functions for the translation were counted after the file had been opened until the moment that the user could start typing the translation.

In SDL Trados Studio 2015, the user needs 4 clicks: a double-click on the project, then a double-click on the file for translation to access the 'Editor'. The first segment is already selected and the user can begin typing without the need of further clicks.

In Inkscape 0.91, two functions can be used for the translation of SVG files. With the 'Create and edit text objects' function, the user needs 2 clicks to be able to start translating: a click on the icon and a click to select a text content element. With the 'Text and Font…' function, 5 clicks are needed: a click on 'Text', then on 'Text and Font…', a click to select the 'Text' tab, another click to select a text content element and one last click on the function's text box to be able to type.

Action	**SDL Trados Studio 2015**	**Inkscape 0.91**
Opening the file for translation	4 clicks	4 clicks
Accessing the functions used for translation	Editor: 4 clicks	'Create and edit text objects': 2 clicks 'Text and Font…': 5 clicks
Sum of clicks	8 clicks	7.5 clicks
Average ($\bar{x}$)	**4**	**3.67**
Final Score	**12**	**12**

Table 6. Evaluation of the number of clicks needed to access the translation functions

The final score was obtained from the sum of clicks, since Inkscape 0.91 offers two different functions for the translation and SDL Trados Studio 2015 only has the option of translating in the 'Editor' view. The results show that a sum of 8 clicks is needed with SDL Trados Studio 2015 and 7.5 clicks (2 options with different values) with Inkscape 0.91 to access the functions for the translation.

Overall, **SDL Trados Studio 2015** earned a final score of **12 points** in this evaluation, and **Inkscape 0.91** also obtained **12 points**.

A second evaluation was carried out for the number of clicks, focusing on the translation task. Table 7 and Table 8 show the number of clicks made during the translation tasks of the experiment for each participant. The results were divided into the two groups of the experiment in order to compare the clicks for the same images. Again, the double-clicks are counted as two clicks.

Participant	Clicks SDL Trados Studio 2015		Clicks Inkscape 0.91		Score SDL Trados Studio 2015	Score Inkscape 0.91
	Single	Double	Single	Double		
P1	54	17	260	60	**5**	**2**
P3	182	18	117	1	**3**	**4**
P5	31	4	205	3	**5**	**3**
Average ($\bar{x}$)	89	13	194	21.33		
Standard Deviation (s)	81.36	7.81	72.13	33.5		

Table 7. Evaluation of the number of clicks of the translation task for group 1

Participant	Clicks SDL Trados Studio 2015		Clicks Inkscape 0.91		Score SDL Trados Studio 2015	Score Inkscape 0.91
	Single	Double	Single	Double		
P2	26	7	238	30	**5**	**3**
P4	165	7	501	17	**4**	**0**
P6	118	6	293	10	**4**	**2**
Average ($\bar{x}$)	103	6.67	344	19		
Standard Deviation (s)	70.7	0.58	138.72	10.15		

Table 8. Evaluation of the number of clicks of the translation task for group 2

The results show that the participants used less clicks with SDL Trados Studio 2015 ($\bar{x}$ = 89, s = 81.36 in group 1 and $\bar{x}$ = 103, s = 70.7 in group 2) and double-clicks ($\bar{x}$ = 13, s = 7.81 in group 1 and $\bar{x}$ = 6.67, s = 0.58 in group 2) than with Inkscape 0.91 (single-clicks: $\bar{x}$ = 194, s = 72.13 in group 1 and $\bar{x}$ = 344, s = 138.72 in group 2; double-clicks: $\bar{x}$ =21.33, s = 35.5 in group 1 and $\bar{x}$ = 19, s = 10.5 in group 2).

The big disparity between the tools, concerning the clicks the participants used during the translation, stems from the different nature of the tools. Being a CAT tool, SDL Trados Studio 2015 is adapted to the needs of translators, thus the translatable text is extracted and can be translated directly. It is possible to select the target segments by clicking on them, although this is not necessary, since the 'Ctrl+Ret' shortcut confirms a segment and automatically moves to the next one. This kept the SDL Trados Studio 2015 click count very low in most of the participants' experiments. The screen recordings have revealed that the participants mostly used clicks to consult the source image for context or to select already translated segments to make changes. In Inkscape 0.91, the user needs to select each text content element to modify the text. This led to a considerably higher click count for the tool.

The SDL Trados Studio 2015 translations of P3 and P4 had more clicks than the others. P3 took more time for the translation, because this user edited the translation after viewing the translated file (cf. section 5.7). This practice not only influenced the time behaviour, but also the number of clicks. The screen recording for P4 showed that the participant selected the segments by clicking on them, even though they had already been selected with the shortcut 'Ctrl+Ret'. P4 also needed further clicks, because the user tried to consult the preview function and to merge segments.

The results also show that P4 needed considerably more clicks than the other participants in Inkscape 0.91. The user edited the text content elements during the translation process, so that they would not overlap, which explains the high click count.

The high standard deviation for the double-clicks used in Inkscape 0.91 stems from the participants' use of double-clicks to select elements, even though a single-click would have sufficed.

Overall, **SDL Trados Studio 2015** achieved a final score of **26 points** in the analysis of the number of clicks made during the translation process, whereas **Inkscape 0.91** obtained **14 points**.

5.5.1.2 Number of Keystrokes

The following sections are dedicated to the analysis of the keystrokes, including a description of the factors that had an impact on the key logs (cf. section 5.5.1.2.1) and the analysis of the virtual keys that were used by the participants of the experiment during the translation tasks (cf. section 5.5.1.2.2).

5.5.1.2.1 Factors that Influenced the Key Logs

The keystroke analysis is based on the recordings of BB FlashBack Express 5. The screen recording tool allows its users to export the key log as an XML file containing all of the normal keystrokes and virtual keys with their respective timestamps. The virtual keys were then grouped and counted using the XML key log. In BB Flashback Express 5, 'virtual keys' are keyboard shortcuts (e.g. [Ctrl]+[Z]) and other special keys used for commands (e.g. 'Space' or 'Delete'), whereas 'normal keys' are alphanumeric characters (e.g. the letter 'A').

An issue with the key log was encountered due to the participants' keyboard language settings during the experiment. All of the participants could choose their preferred language. P1 and P5 had their settings changed to Spanish. P6 started with the keyboard set to French, but had some difficulties finding the right keys and changed the settings to Spanish after a few minutes of working with Inkscape 0.91. For the translation with SDL Trados Studio 2015, P6 did not want to change the settings to Spanish again, leaving them at the default French setting. P2, P3 and P4 had their keyboards set to French throughout the entire experiment. The language settings proved to have an impact on the way the keys were registered. Participants working with a Spanish keyboard would have the 'z' and 'y' keys inverted, which had an impact on the recording of the shortcuts 'Ctrl+Z' and 'Ctrl+Y'. In the tables below, the results have been adapted to reflect the real commands that were given to the program (e.g. the key log showed that P5 used the shortcut 'Ctrl+Y' 104 times with Inkscape 0.91, but the command that was given to the program corresponded to 'Ctrl+Z').

Another difference that arose from the change in language settings was the way special characters, such as accented letters, were registered. Participants using the French keyboard had to use the key combination 'Ctrl+Alt' to create the special characters used in the Spanish alphabet. This key combination was included in Table 17 and Table 18 (cf.

Appendix XV) as a shortcut in order to give a complete overview of the used keys. Nevertheless, it was omitted from the results displayed in Table 9 and Table 10, which were used to obtain a score for the operability evaluation. Their omission ensures an equal calculation of the results of the virtual keys and the special characters are still reflected in the total count of the normal keys.

Aside from the aspects mentioned in this section, no further impact on the results was noticed.

5.5.1.2.2 Virtual Key Analysis

Table 9 and Table 10 provide an overview of the number of virtual keys and normal keys, as well as the total number of keystrokes (sum of virtual keys and normal keys) that the participants of each group used. The results were separated by group to allow for a better comparison of the keystrokes, since group one started translating the first image with SDL Trados Studio 2015, whereas the second group translated the same image using Inkscape 0.91.

The evaluation focuses on the analysis of the virtual keys, instead of the normal keystrokes that the participants made during the experiment. This decision was taken, since the virtual keys are a better reflection of the tool's ease of use, while the normal keystrokes could primarily give information on the participants' typing skills and the way they translate. Generally speaking, the participants used quite a similar amount of normal keys in the experiments (cf. 'Keys' in Table 9 and Table 10). P3 used more keys for the translation with SDL Trados Studio 2015, which is in accordance with the participant's decision to edit the file after translation. The amount of keys used by P2 in Inkscape 0.91 is comparably low, which is most likely a result of the text that has been left untranslated.

Table 17 and Table 18 (cf. Appendix XV) present a more detailed description of the virtual keys by providing a list of all the shortcuts and other special keys that the participants used during the translation process. In both tables, the names of the virtual keys follow the terminology used by BB FlashBack Express 5.

The count of the shortcuts provided in Table 9 and Table 10, as well as the lists of virtual keys and other special keys in Table 17 and Table 18 (cf. Appendix XV) include all of the wrong shortcuts that were used by the participants during the experiment (marked with 'wrong shortcut' in the description). These were made when the participants pressed

the wrong keys for a shortcut or when two virtual keys were used together, but did not occupy a special function. They were not omitted, since they are a normal part of the writing process on a keyboard. Some examples of such shortcuts are '[Shift]+[Spc]' and '[Shift]+[BkSp]', which were created when a participant was writing in capital letters with the 'Shift' key pressed and then used the 'Space' or 'Backspace' key. Another example for these wrong shortcuts are '[Ctrl]+[Shf]' and '[Shift]+[Ctrl]', which actually refer to the same key combination, but in a different order (in the former, the 'Ctrl' key was pressed first and the latter one uses the 'Shift' key first). These key combinations do not have a function, but were registered in the screen recordings just before other shortcuts were used when the participants took a bit longer to press the remaining key of the desired shortcut.

In some cases, the differences of virtual keys used with each tool were quite pronounced. Also, the results have shown that some participants worked primarily with virtual keys, whereas others preferred using clicks to navigate the programs.

The differences between the tools concerning the use of virtual keys provide insight into the translation processes for each tool. P3 used many different virtual keys for the translation. The results show that this participant used the shortcut '[Alt]+[TAB]' (allows the user to go to another open window) 35 times with SDL Trados Studio 2015 and only once with Inkscape 0.91. The screen recording showed that P3 used this shortcut during the translation with SDL Trados Studio 2015 to view the English SVG file, supposedly to gather more context. The fact that P3 only used this shortcut once with Inkscape 0.91 demonstrates that the participant had all the necessary context for the translation, since the source file is displayed in the same tool.

The disparity between the use of the virtual keys 'Del' (delete) or 'Bksp' (backspace) to erase words also needs to be pointed out. With the exception of P3, all of the participants needed to erase by far more letters or words in Inkscape 0.91 than in SDL Trados Studio 2015. For example, P5 used the 'Backspace' key 91 times in SDL Trados Studio 2015 and 498 times in Inkscape 0.91 and P6 used the 'Delete' key 529 times in Inkscape 0.91, but not even once in SDL Trados Studio 2015. This contributed greatly to the disparity in both participants' final virtual key count.

This effect can most likely be attributed to the fact that in Inkscape 0.91, the source text needs to be erased from the image, while SDL Trados Studio 2015 lets the user enter the translation without the need to erase the source text. Only P3 used both virtual keys slightly more often in SDL Trados Studio 2015. The most probable reason for this, is that the user changed the initial translations more often in SDL Trados Studio 2015 than in Inkscape 0.91. A second possible reason for the elevated virtual key count is that P3 edited the text of the target SVG file in SDL Trados Studio 2015 after having completed the translation, thus erasing many words with the 'Delete' and 'Backspace' keys.

Another interesting aspect that can be noticed by looking at the use of virtual keys, is that the participants frequently needed the shortcut 'Ctrl+Z' during the translation with Inkscape 0.91. The screen recordings show that participants used this shortcut primarily to bring back text content elements that had accidentally been erased (which also deleted the style settings of the element) when the participant deleted the source text before typing the translation. In order to maintain the same text style, the user had to leave at least one character of the text element intact, so that the element would not be erased completely.

Furthermore, the participants sometimes had difficulty with the 'Text and Font…' function in Inkscape 0.91. When they applied the text changes, the style of the text content element would often change drastically. The participants frequently undid their last operations with the shortcut 'Ctrl+Z', to remedy any unwanted style change.

In comparison, P6 used this shortcut 268 times in Inkscape 0.91, but only used it once with SDL Trados Studio 2015, to bring back a translation that had accidentally been overwritten with the source text of the corresponding segment due to the prior use of the shortcut 'Ctrl+Ins'.

An elevated use of the 'Control' key can also be noticed in the translations of P3 and P4 in SDL Trados Studio 2015. The screen recordings show that this is primarily due to the longer translation process (cf. section 5.7). Also, P4 tried to merge the segments various times, which led to a much higher use of the 'Shift' key, since the participant pressed it while selecting the segments that they wanted to merge.

Overall, **SDL Trados Studio 2015** used less virtual keys in 5 out of 6 occasions and achieved a final score of **19 point** in the virtual key analysis, whereas **Inkscape 0.91** obtained a final score of **12 points**.

Participant	Virtual Keys SDL Trados Studio 2015	Virtual Keys Inkscape 0.91	Keys SDL Trados Studio 2015	Keys Inkscape 0.91	Total SDL Trados Studio 2015	Total Inkscape 0.91
P1	468	542	898	731	1366	1273
P3	1367	941	1015	871	2382	1812
P5	373	1186	725	637	1098	1823
Average ($\bar{x}$)	736	889.67	879.33	746.33	1615.33	1636
Standard Deviation (s)	548.52	325.05	145.90	117.75	677.34	314.41

Table 9. Overview of the number of virtual keys, normal keys and total number of keystrokes used by group 1

Participant	Virtual Keys SDL Trados Studio 2015	Virtual Keys Inkscape 0.91	Keys SDL Trados Studio 2015	Keys Inkscape 0.91	Total SDL Trados Studio 2015	Total Inkscape 0.91
P2	465	547	856	356	1321	903
P4	1180	1659	658	982	1838	2641
P6	660	1738	785	879	1445	2617
Average ($\bar{x}$)	768.33	1314.67	766.33	739	1534.67	2053.67
Standard Deviation (s)	369.61	665.99	100.31	335.66	269.91	996.58

Table 10. Overview of the number of virtual keys, normal keys and total number of keystrokes used by group 2

5.6 Attractiveness

Attractiveness is the "capability of the software product to be attractive to the user" (ISO/IEC 9126-1, 2001). This sub-characteristic was measured with an evaluation of the attributes 'attractiveness of the interface', 'advantages and disadvantages' and 'overall satisfaction'.

5.6.1 Attractiveness of the Interface

In Q6 of the localisation questionnaires, participants were asked to rate how attractive (pleasing to the eye) the interface of the tool is. The findings are displayed in Table 11.

The results show that SDL Trados Studio 2015 has achieved a higher rating ($\bar{x} = 4.17$, $s = 0.98$) than Inkscape 0.91 ($\bar{x} = 3.5$, $s = 1.38$) in terms of attractiveness.

P1 commented: "The interface is clear and easy to use. I think this version of Trados has been improved in comparison with previous versions, being the actual one more user-friendly".

Overall, **SDL Trados Studio 2015** achieved a final score of **9 points** in the evaluation of the attractiveness, whereas **Inkscape 0.91** obtained a final score of **6 points**.

Participant	Question Number	Rating SDL Trados Studio 2015	Rating Inkscape 0.91	Score SDL Trados Studio 2015	Score Inkscape 0.91
P1	Q6	5	4	**2**	**1**
P2	Q6	3	5	**1**	**2**
P3	Q6	5	1	**2**	**0**
P4	Q6	4	4	**1**	**1**
P5	Q6	3	4	**1**	**1**
P6	Q6	5	3	**2**	**1**
Final Score				**9**	**6**
Average ($\bar{x}$)	Q6	4.17	3.5		
Standard Deviation (s)	Q6	0.98	1.38		

Table 11. Results of the attractiveness evaluation

5.6.2 Advantages and Disadvantages

Three questions (**Q7**, **Q8**, **Q9**) were used to evaluate the advantages and disadvantages of the tools in order to determine their overall attractiveness. The evaluation results are shown in Table 12 below.

The participants were asked to list the advantages of the tools that they had noticed during the translation of the SVG images in Q7 and the disadvantages in Q8. On average, the participants listed 2.5 advantages for both SDL Trados Studio 2015 (s = 1.05) and Inkscape 0.91 (s = 1.38). Table 19 (cf. Appendix XVI) provides an overview of the participants' answers given for Q7.

According to the participant's answers, the main advantages of SDL Trados Studio 2015 were the reuse of previously translated segments and auto-propagation of numbers, the well-known CAT tool environment, the similarity of the translation of SVG files to the translation of other files, and that there is no risk of moving or changing elements.

For Inkscape 0.91, the main advantages were the possibility to work directly on the image and to see the results, the easy-to-use interface after getting used to it, and the

possibility to modify sentences and to easily move and edit elements. In response to **Q8**, they listed on average 2.17 disadvantages for Inkscape 0.91 (s = 0.98) and 2.33 for SDL Trados Studio 2015 (s = 1.97). Table 20 (cf. Appendix XVI) provides an overview of the participants' answers given for Q8.

The main disadvantages for SDL Trados Studio 2015 listed by the participants were the division of sentences into different segments that cannot be merged, the incapability of viewing the image, and that sentences started with capital letters.

For Inkscape 0.91, the main disadvantages were the difficulty of selecting the right segment, the risk of damaging or changing the original image, that the original format was changed, and the difficulty of creating line breaks.

Q9 asked the participants to state if they would change anything about the tools concerning SVG translation. On average, the participants would have changed one thing for SDL Trados Studio 2015 ($\bar{x}$ = -1, s = 0.63). In the case of Inkscape 0.91, all of them stated exactly one thing that they would change about the tool ($\bar{x}$ = -1, s = 0). Table 21 (cf. Appendix XVI) provides an overview of the participants' answers given for Q9.

Participants mainly listed the possibility of visualising the image in the editor and the ability to merge segments as desirable changes for the tool SDL Trados Studio 2015.

For Inkscape 0.91, they would want a better capability of selecting text, or more complete, word-processor-like text boxes, and a text-editing feature adapted to the use by translators or localisers. Table 12 contains the numeric representation of the results and the scores that were given in the evaluation of the advantages and disadvantages of the tools according to the pre-established metrics.

Overall, **SDL Trados Studio 2015** achieved a final score of **27 points** in the evaluation of the advantages and disadvantages, whereas **Inkscape 0.91** obtained a final score of **26 points**.

Participant	Question Number	Points SDL Trados Studio 2015	Points Inkscape 0.91	Score SDL Trados Studio 2015	Score Inkscape 0.91
P1	Q7	3	4	**3**	**4**
	Q8	-6	-2	**0**	**2**
	Q9	-1	-1		
P2	Q7	2	2	**2**	**2**
	Q8	-1	-2	**3**	**2**
	Q9	-1	-1		
P3	Q7	4	1	**4**	**1**
	Q8	-1	-2	**2**	**2**
	Q9	-2	-1		
P4	Q7	2	3	**2**	**3**
	Q8	-2	-4	**3**	**0**
	Q9	0	-1		
P5	Q7	3	4	**3**	**4**
	Q8	-3	-2	**1**	**2**
	Q9	-1	-1		
P6	Q7	1	1	**1**	**1**
	Q8	-1	-1	**3**	**3**
	Q9	-1	-1		
Final Score				**27**	**26**
Average ($\bar{x}$)	Q7	2.5	2.5		
	Q8	-2.33	-2.17		
	Q9	-1	-1		
Standard Deviation (s)	Q7	1.05	1.38		
	Q8	1.97	0.98		
	Q9	0.63	0		

Table 12. Results of the advantages and disadvantages evaluation[35]

[35] The points for question 8 and 9 were merged, since the changes form part of and give information on the disadvantages.

5.6.3 Overall Satisfaction

The overall satisfaction was determined with three questions (**Q10**, **Q11**, **Q12**) in the localisation questionnaires. The results are presented in Table 13.

Q10 asked the participants if they enjoyed working with the tool. All of the participants enjoyed working with SDL Trados Studio 2015 and 3 out of 6 enjoyed working with Inkscape 0.91.

In **Q11**, the participants were asked if they thought that the tool was a good tool for the translation of SVG files. 4 out of 6 participants gave an affirmative answer for SDL Trados Studio 2015 and 3 out of 6 for Inkscape 0.91.

Q12 had the participants rate how satisfied they were with the resulting translations. SDL Trados Studio 2015 obtained a slightly higher score ($\bar{x} = 3.33$, $s = 0.82$) than Inkscape 0.91 ($\bar{x} = 3.17$, $s = 0.75$).

In relation to the overall satisfaction with SDL Trados Studio 2015, P1 remarked that "[i]n the final result, there are different cases where the text and the images are overlapped. As the user can't see a preview of the result before finishing the translation, he can't see that there would be [a need] to rearrange the text in order to avoid those cases".

Concerning the overall satisfaction with Inkscape 0.91, the same participant (P1) wrote: "The fact of being able to see the image when the user is working is basic and very important to have an idea of the final result. The user can reformulate the translation if it's too long, modify the boxes, etc."

Overall, **SDL Trados Studio 2015** achieved a final score of **15 points** in the evaluation of the overall satisfaction, whereas **Inkscape 0.91** obtained a final score of **11 points**.

Participant	Question Number	Rating SDL Trados Studio 2015	Rating Inkscape 0.91	Score SDL Trados Studio 2015	Score Inkscape 0.91
P1	Q10	Yes	Yes	**2**	**3**
	Q11	No	Yes		
	Q12	3	4		
P2	Q10	Yes	No	**3**	**1**
	Q11	Yes	No		
	Q12	4	3		
P3	Q10	Yes	No	**1**	**0**
	Q11	No	No		
	Q12	2	2		
P4	Q10	Yes	No	**3**	**1**
	Q11	Yes	No		
	Q12	4	3		
P5	Q10	Yes	Yes	**3**	**3**
	Q11	Yes	Yes		
	Q12	4	3		
P6	Q10	Yes	Yes	**3**	**3**
	Q11	Yes	Yes		
	Q12	3	4		
Final Score				**15**	**11**
Average ($\bar{x}$)	Q12	3.33	3.17		
Standard Deviation (s)	Q12	0.82	0.75		

Table 13. Results of the overall satisfaction evaluation

5.7 Time Behaviour

Time behaviour is the "capability of the software product to provide appropriate response and processing times and throughput rates when performing its function, under stated conditions" (ISO/IEC 9126-1, 2001).

The efficiency of SDL Trados Studio 2015 and Inkscape 0.91 regarding SVG localisation is evaluated by scenario test, measuring the time participants took to carry out the translation task. The time behaviour analysis is conducted on the basis of the recordings of the experiment with BB FlashBack Express 5.

Table 14 contains the results for the time behaviour in the translation tasks for each participant.

Participant	Time Behaviour SDL Trados Studio 2015	Time Behaviour Inkscape 0.91	Score SDL Trados Studio 2015	Score Inkscape 0.91
P1	17m45s	31m2s	**3**	**0**
P2	13m42s	17m49s	**4**	**3**
P3	30m49s	17m13s	**0**	**3**
P4	17m58s	39m27s	**3**	**0**
P5	12m37s	24m55s	**4**	**2**
P6	21m27s	30m15s	**2**	**0**
Final score			**16**	**8**
Average ($\bar{x}$)	19m3s	26m47s		
Standard Deviation (s)	6m35s	8m33s		

Table 14. Results of the time behaviour evaluation

The results gathered from the scenario tests show that the participants needed less time for the translation task using SDL Trados Studio 2015 ($\bar{x}$ = 19m3s, s = 6m35s) than using Inkscape 0.91 ($\bar{x}$ = 26m47s, s = 8m33s).

In a comparison of the results of the first group (P1, P3, P5), the participants took roughly the same time for the translation task, but SDL Trados Studio 2015 proved to be more efficient ($\bar{x}$ = 20m23s, s = 9m22s) than Inkscape 0.91 ($\bar{x}$ = 24m22s, s = 6m55s).

The results obtained from the second group (P2, P4, P6) show more disparity between translation times. Participants were clearly faster using SDL Trados Studio 2015 ($\bar{x}$ = 17m41s, s = 3m53s) than Inkscape 0.91 ($\bar{x}$ = 29m10s, s = 10m50s).

Out of all the tests, only P3 was faster using Inkscape 0.91 than SDL Trados Studio 2015. A look at the screen recording revealed that after having completed the translation process in SDL Trados Studio 2015, P3 went back several times to correct the segments after viewing the target file, even though this was not part of the participant's task. The user seemed to be dissatisfied with the initial translation from SDL Trados Studio 2015 after seeing the translation in context, thus going back to correct segments. Also, this participant often looked up the SVG file to locate the SDL Trados Studio 2015 segments in the source image. These practices have greatly contributed to the disparity in the translation times, but would not have affected the attribute's final score. If the participant's translation process would have been counted until the first save of the complete translation, the translation time with SDL Trados Studio 2015 would have been 20 minutes and 6 seconds, resulting in an average of $\bar{x}$ =17m16s and a standard deviation of s = 3m29s. Table 14 takes into account the result of the participant's translation process including the editing process that was carried out, since the user apparently was not satisfied enough with the translated SVG file to consider that the translation process was already complete.

P4 took the longest with Inkscape 0.91, but it is important to mention that this user edited the SVG image during the translation process, eliminating some space issues, so that the text content elements would not overlap. This task is also reflected in the localisation errors, since P4's translation did not show any space issues.

Overall, **SDL Trados Studio 2015** was the fastest tool in 5 out of 6 occasions and is awarded a final score of **16 points** in the evaluation of the time behaviour, whereas **Inkscape 0.91** is given a final score of **8 points**.

5.8 Criteria Weighting

As mentioned in section 3.4.2.5, not all the attributes are equally important for the needs of beginners in SVG localisation. In order to establish which attributes are most important and which are least important, the participants of the experiment were asked to rate the importance of each attribute for the localisation of Scalable Vector Graphics. Table 15 shows the results, the average rating ($\bar{x}$), and the assigned weight factor for each sub-characteristic or attribute.

Sub-characteristic or Attribute	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**$\bar{x}$**	**Weight factor**
Display of translatable content	5	5	5	5	3	2	4.17	**8.04**
Ease of editing	5	5	5	5	4	5	4.83	**9.33**
Consistency	5	5	4	5	4	5	4.67	**9.01**
Spell-checking	5	4	4	5	4	3	4.17	**8.04**
Localisation errors	5	5	5	5	4	5	4.83	**9.33**
User-friendliness	4	4	5	4	3	4	4	**7.72**
Learnability	4	5	5	4	4	3	4.17	**8.04**
Ease of use	4	5	4	5	5	4	4.5	**8.69**
Attractiveness of the interface	3	4	4	3	3	1	3	**5.79**
Time behaviour	5	5	5	5	5	4	4.83	**9.33**

Table 15. Results of the Criteria Weighting Questionnaire for each sub-characteristic or attribute

The highest rated attributes are ‘ease of editing’, ‘localisation errors’ and ‘time behaviour’, whereas ‘user-friendliness and ‘attractiveness of the interface’ received the lowest ratings. This could be interpreted as a positive reflection of the users’ expertise in the field of computer-assisted translation, since people who do not frequently rely on programs would most likely tend to favour ‘user-friendliness’ or ‘attractiveness’ over other functionalities.

Surprisingly, the rating of the attribute 'display of translatable content' is not among the highest. It is interesting to see that P1 to P4 have given the highest rating to this attribute, while it was rated quite lowly by P5 (3 points) and P6 (2 points). This difference in the ratings might be attributed to the localisation experience of the participants. P1 to P4 were all students or former students specialised in Translation Technologies who have taken the course on localisation 'Localisation et gestion de projet' at the University of Geneva. P5 and P6 were students in their first year, without a specialisation in Translation Technologies, who have not taken the localisation course, but only the course on computer-assisted translation 'Traduction assistée par ordinateur (TAO)'. This lack of familiarisation in the field of localisation could explain the lower ratings for an attribute that would usually be ranked among the most important ones.

For the purpose of the SVG localisation evaluation of this research project, the value assigned to each feature is calculated as the sum of the participants' ratings for each feature.

To calculate the weighted values, the attributes' assigned values are harmonised to 1 (=100%). The weight factor is calculated by multiplying each value by a constant (C).

The attributes 'advantages and disadvantages' and 'overall satisfaction' have not been assigned a weight, since it was not possible to rate their importance. They have been weighted with an average value of 8.33% ($\frac{1}{12}$, because 12 is the number of attributes in the evaluation). This average weighting has been given in order to be consistent in the application of the weighting.

So, for the criteria weighting assigned by the participants to the 10 attributes shown in Table 15, only 1 - $\frac{2}{12}$ (=83.33%) is left, and therefore, this value influences the constant C.

This constant is calculated by the division of the sum of the attributes' assigned values (V) following the formula $C = \frac{100\% - 2 * 8.33\%}{\sum V}$. The sum of the weighted values equals 1 - $\frac{2}{12}$ (=83.33%).

Following the aforementioned formula, the constant is calculated this way: $C = \frac{83.33\%}{259}$. The resulting constant C = 0.00322 is applied to the final scores of each attribute gathered from the evaluation of the two tools (cf. section 6.1, Table 16).

6. Summary and Conclusions

This chapter provides the reader with a summary of the results of the evaluation (section 6.1), the conclusions that were drawn from the results (section 6.2), the limitations of this study (section 6.3), and some ideas for possible future research related to this study (section 6.4).

6.1 Summary of the Results

Table 16 provides a summary of the final scores for SDL Trados Studio 2015 and Inkscape 0.91, which were obtained from the evaluation in the previous chapter (cf. chapter 5). The table also reflects the weight assigned to each attribute (cf. Table 16) in the form of the weighted values. The sum of the weighted values determines the final outcome of the evaluation of SDL Trados Studio 2015 and Inkscape 0.91 concerning SVG localisation. The total value could easily be compared to other programs, as the optimum value for a perfect program is, logically, 100%.

Attribute	Final Score SDL Trados Studio 2015	Final Score Inkscape 0.91	Harmonised Final Score SDL Trados Studio 2015	Harmonised Final Score Inkscape 0.91	Weighted Value SDL Trados Studio 2015	Weighted Value Inkscape 0.91
Display of translatable content	2 of 2	1 of 2	100%	50%	**8%**	**4%**
Ease of editing	0 of 3	3 of 3	0%	100%	**0%**	**9%**
Consistency	1 of 1	0 of 1	100%	0%	**9%**	**0%**
Spell-checking	1 of 1	0 of 1	100%	0%	**8%**	**0%**
Localisation errors	19 of 30	18 of 30	63%	60%	**6%**	**6%**
User-friendliness	15 of 24	9 of 24	63%	38%	**5%**	**3%**
Difficulty of learning how to use the tool	14 of 30	17 of 30	47%	57%	**4%**	**5%**
Ease of use	57 of 90	38 of 90	63%	42%	**6%**	**4%**
Attractiveness of the interface	9 of 12	6 of 12	75%	50%	**4%**	**3%**
Advantages and disadvantages	27 of 60	26 of 60	45%	43%	**4%**	**4%**
Overall satisfaction	15 of 24	11 of 24	63%	46%	**5%**	**4%**
Time behaviour	16 of 30	8 of 30	53%	26%	**5%**	**2%**
Total					**64%**	**44%**

Table 16. Summary of the final scores and weighted values for each tool per attribute

6.2 Conclusions

The aim of this research project was to find out whether SDL Trados Studio 2015 or Inkscape 0.91 is better suited for the localisation of SVG files by users with little or no experience in image localisation.

First, the basics of the SVG file format were described for a better understanding of its structure and functions. This section focused on the text content elements, which are at the centre of SVG translation.

Information on the research methodology included a description of the creation of the SVG filter for SDL Trados Studio 2015, which is a crucial part of the experiment, since the custom filter allows the program to display the textual content in SVG files, thus facilitating the execution of the experiment. Subsequently, an overview of the EAGLES evaluation method was given and the model was applied to the SVG localisation evaluation of SDL Trados Studio 2015 and Inkscape 0.91.

Three characteristics were chosen for the evaluation ('functionality', 'usability' and 'efficiency'), which were divided into seven sub-characteristics ('suitability', 'accuracy', 'understandability', 'learnability', 'operability', 'attractiveness' and 'time behaviour'). A last division was applied to the first six sub-characteristics, to obtain eleven measurable attributes ('display of translatable content', 'ease of editing', 'consistency', 'spell-checking', 'localisation errors', 'user-friendliness', 'difficulty of learning how to use the tool', 'ease of use', 'attractiveness of the interface', 'advantages and disadvantages' and 'overall satisfaction').

The evaluation of the tools was conducted using different means of measurement. The 'suitability' was evaluated by feature inspection, which is used to examine the technical features of a system. Scenario tests were carried out for the sub-characteristics 'accuracy', 'operability' and 'time behaviour', taking the form of an experiment with six participants, who were asked to do two SVG translation tasks. In the course of the experiment, questionnaires were used to determine the 'understandability', 'learnability' and the 'attractiveness' of the tools.

The findings of the evaluation (cf. chapter 5 and Table 16) lead to the conclusion that, with a total score of 64% of the optimum performance, the CAT tool SDL Trados Studio 2015 is better suited for the localisation of Scalable Vector Graphics by users with little or no experience in the field of image localisation than the vector graphics editor Inkscape 0.91, which achieved a total score of 44% of the optimum performance.

The CAT tool obtained higher scores for the attributes 'display of translatable content', 'consistency', 'spell-checking', 'user-friendliness', 'ease of use', 'attractiveness of the interface', 'overall satisfaction' and 'time behaviour'.

Inkscape 0.91 achieved higher scores than SDL Trados Studio 2015 for the attributes 'ease of editing', 'difficulty of learning how to use the tool', and the same scores for 'localisation errors' and 'advantages and disadvantages'.

The differences are especially prominent when it comes to the attributes 'consistency', 'spell-checking' and 'time behaviour'. Even though Inkscape 0.91 offers more ease of editing than SDL Trados Studio 2015 and it is less difficult to learn how to use the tool, the features that are more specific to CAT tools, such as the ones that were mentioned above, seem to have had a significant impact on the other attributes, such as time behaviour and ease of use, since the participants could operate the tool without much effort.

The final results of the study support the hypothesis, which suggested that SDL Trados Studio 2015 would be better suited for the localisation of SVG files by novice translators or localisers with little or no image localisation experience than Inkscape 0.91.

There is some disparity between the tools, which leads to the conclusion that both programs have their pros and cons for the translation of SVG files. Even though image editors offer the capability of viewing and editing graphics, the study has shown that novice translators or localisers prefer the familiar environment and translation-specific features of CAT tools and achieve better results using this type of tools. The criteria weighting further illustrates the high appreciation for CAT tool features in SVG localisation.

Although CAT tools have proven to be better suited for the SVG localisation by this type of users, the final scores show that image editors, such as Inkscape 0.91, can also be used for the localisation of vector graphics, despite their lack of CAT tool specific features.

A final aspect can be taken into consideration: the results of the 'accuracy' evaluation found localisation errors in the translations made with SDL Trados Studio 2015 that could be remedied by the possibility of viewing the source file and editing the translated text content elements. This was evidenced by the data on 'accuracy' recollected from P3's translation task in SDL Trados Studio 2015. The participant frequently consulted the source file, which led to a translation without senseless phrases. Since Inkscape 0.91 obtained much better results than SDL Trados Studio 2015 for the 'ease of editing' attribute, a combination of both tools might yield the best results in the translation of SVG files. Further research is needed to support or contradict this consideration.

6.3 Limitations

The study laid out in this research project presents some limitations, which are explained in this section.

First and foremost, it is important to point out that the findings have limited external validity due to the reduced number of participants. With the context being that of a Master's thesis, the study relied on the voluntary participation of students.

Time was also a significant factor that limited the study. Further characteristics could have been analysed to enrich the evaluation of the tools and achieve a more exhaustive study. It would also have been possible to add more attributes to the evaluated characteristics to obtain a more accurate reflection of the tools' capabilities. However, doing so would have resulted in longer experiment durations and some time constraints had to be implemented in the experiment to prevent fatigue of the participants or a withdrawal from the study.

Another consideration is the difference between participants regarding the knowledge of the evaluated tools, which was caused by the scarce availability of voluntary participants with a relevant profile. Some participants had never worked with the 2015 version of SDL Trados Studio, while others had SDL certifications for this version. Even though the newer versions of the tool are very similar, there is still some disparity that might have been addressed if the study had been carried out on a larger scale with more participants.

The choice of the target user may present a further limitation for the study, since it focused on novice translators and localisers, whereas the results might have been different, had the focus been placed on localisation professionals.

Considering the aforementioned limitations, this research project serves as a preliminary study, which can be used for more exhaustive studies in the future (cf. section 6.4).

6.4 Future Applications

As mentioned in the previous section (cf. section 6.3), this research project could serve as a basis for future studies. This section provides some ideas for future research that could complete the findings.

It would be possible to extend the concept of the study to include further characteristics or attributes and achieve a more exhaustive evaluation. One possibility would be to create a study focused solely on the editing process. Another possible application would be the repetition of the experiment with a greater number of participants in order to verify if the results would still be relevant. It would also be interesting to conduct a similar evaluation of the tools with different target users, such as localisers with much experience in the field of vector graphics localisation.

The concept of this study could be applied to different tools, featuring other CAT tools and vector graphics editors, or focusing solely on a comparison of the functionalities in CAT tools or in vector graphics editors.

In the future, it would be possible to repeat the experiment with newer releases of the tools or of SVG itself, to determine whether changes in the tools' functions or the functionalities of SVG affect the results. In the same manner, it would be possible to carry out a study including interactive or multilingual SVG files, which were excluded in this research project.

In conclusion, the findings of the evaluation suggest that a combination of SDL Trados Studio 2015 for the translation process and Inkscape 0.91 for the editing process might be ideal. A study could be conducted to support or contradict this assumption.

Bibliography

Andersen, A., 2010. *WebAIM: Future Web Accessibility: SVG*. [online] Available at: <http://webaim.org/blog/future-web-accessibility-svg/> [Accessed 20 July 2015].

Aranzana González, R., 2012. *Creación de filtros para documentos XML en la plataforma SDL Trados Studio 2009. Guía práctica para traductores.* Universidad de Salamanca. Available at: <http://gredos.usal.es/jspui/bitstream/10366/120767/1/TFG_RAranzanaGonzalez.pdf> [Accessed on 9 April 2015].

Canelli, M. and Grasso, D. and King, M., 2000. Methods and Metrics for the Evaluation of Dictation Systems: A Case Study. In: *Second International Conference on Language Resources and Evaluation (LREC)*. Athens, Greece, vol. 3, pp. 1325-1331. Available at: <http://www.lrec-conf.org/proceedings/lrec2000/pdf/56.pdf> [Accessed 13 November 2015].

Chen, Alex Q. and Simon Harper. 2008. *Web Evolution: Method and Materials*. Report. Manchester: School of Computer Science, The University of Manchester.

Donney, J. and Murphy, S. and Sacre, C. and Scholz, A. and Walters, D., 2008. Globalization of Graphics: Delineating a Research into Using the Scalable Vector Graphics File Format to Improve the Translation of Graphics. In: *SIGDOC '08: Proceedings of the 26th Annual ACM International Conference on Design of Communication*. New York, N.Y.: ACM, pp. 87–92.

EAGLES, 1999. *The EAGLES 7-step recipe*. [online] Available at: <http://www.issco.unige.ch/en/research/projects/eagles/ewg99/7steps.html> [Accessed 26 May 2015].

EAGLES, 1996. *EAGLES Evaluation of Natural Language Processing Systems FINAL REPORT*. [online] Available at: <http://www.issco.unige.ch/en/research/projects/ewg96/ewg96.html> [Accessed 4 November 2015].

ed__, 2015. Channel: #svg. *Internet Relay Chat*. [personal conversation online] 29 October 2015. Available at: <http://irc.lc/mozilla/svg/> [Accessed 29 October 2015].

Eisenberg, J.D. and Bellamy-Royds, A., 2014. *SVG Essentials*. 2nd ed. Sebastopol, CA: O'Reilly Media, Inc. Available at: <http://commons.oreilly.com/wiki/index.php/SVG_Essentials> [Accessed 15 July 2015].

Filkin, P., 2014a. *Those Project Settings!*. [online] Available at: <http://multifarious.filkin.com/2014/01/24/those-project-settings/> [Accessed 23 October 2015].

Filkin, P., 2014b. *Why do we need custom XML filetypes?*. [online] Available at: <http://multifarious.filkin.com/2014/06/01/custom-xml/> [Accessed 23 October 2015].

Globalization and Localization Association (GALA), 2015. *What is Localization?*. [online] Available at: <https://www.gala-global.org/language-industry/intro-language-industry/what-localization> [Accessed 18 November 2015].

Inkscape, 2015a. *Overview*. [online] Available at: <https://inkscape.org/en/about/overview/> [Accessed 14 December 2015].

Inkscape, 2015b. *Features of Inkscape*. [online] Available at: <https://inkscape.org/en/about/features/> [Accessed 14 December 2015].

ISO/IEC 9126-1: *Software engineering — Product quality — Part 1: Quality model* (2001). Geneva: International Organization for Standardization.

King, M., 1997. Evaluation Design: the EAGLES framework. In: *Konvens 97 Proceedings* Available at: <http://www.cst.dk/eagles/konvens2.html> [Accessed 5 November 15].

King, M., Maegaard, B., 1998, *Issues in Natural Language Systems Evaluation*, *Proceedings of the 1st LREC Conference*, Granada, Spain, vol.1, 28-30 May, pp. 225-230.

Lemonly, 2014. *State of the Commons*. [image online] Available at: <https://stateof.creativecommons.org/report/> [Accessed 18 September 2015]

Mata Pastor, M., 2009a. Algunas pautas para el tratamiento de imágenes y contenido gráfico en proyectos de localización (I). *Entreculturas*, (1), pp. 513–532.

Mata Pastor, M., 2009b. Algunas pautas para el tratamiento de imágenes y contenido gráfico en proyectos de localización (II). *Entreculturas*, (1), pp. 533–569.

Mozilla Developer Network, 2013. *SVG element reference*. [online] Available at: <https://developer.mozilla.org/en-US/docs/Web/SVG/Element> [Accessed 17 October 2015].

Oates, B.J., 2006. *Researching Information Systems and Computing*. SAGE Publications Ltd.

O'Brien, S., 2009. Eye tracking in translation process research: methodological challenges and solutions. In: Mees, I.M. and Alves, F. and Gopferich, S., (eds.) *Methodology, Technology and Innovation in Translation Process Research*. Copenhagen Studies in Language, 38. Samfundslitteratur, Copenhagen, pp. 251–266. Available at: <http://doras.dcu.ie/17157/1/CSL_38_Methodologies.pdf> [Accessed 21 May 2015]

Ó Broin, U., 2003. Image Localization and New Technology. *MultiLingual Computing & Technology*, 14(6).

Peron, C., 2013. *Évaluation d'une plate-forme de localisation : le cas de Reverso Localize*. M.A. Univ. Genève.

Peterson, R.A., 2000. *Constructing Effective Questionnaires*. London: SAGE Publications.

Rico Pérez, C., 2001. Reproducible models for CAT tools evaluation: A user-oriented perspective. In: *Translating and the Computer 23*. 29-30 November 2001. London: Aslib. Available at: <http://www.mt-archive.info/Aslib-2001-Rico.pdf> [Accessed 6 November 2015].

Savage, T.M. and Vogel, K.E., 2013. *An Introduction to Digital Multimedia*. 2nd ed. Burlington, Mass.: Jones & Bartlett Learning.

SDL PLC, 2015a. *Overview: Introducing SDL Trados Studio*. [online] Available at: <http://producthelp.sdl.com/SDL_Trados_Studio_2015/client_en/introducing/ini ntroducingsdltradosstudiooverview.htm> [Accessed 14 December 2015].

SDL PLC, 2015b. *SDL Trados Studio 2015 Languages and Filters*. [online] Available at: <http://www.sdl.com/download/sdl-trados-studio-2015-languages-and-filters/83425/> [Accessed 16 November 2015].

SDL PLC, 2015c. *Legacy Embedded Content vs Embedded Content Processors*. [online] Available at: <http://producthelp.sdl.com/SDL_Trados_Studio_2015/client_en/Legacy_Embedd ed_Content_vs_Embedded_Conent_Processors.htm> [Accessed 29 October 2015].

SDL PLC, 2015d. *Creating a File Type*. [online] Available at: <http://producthelp.sdl.com/SDL_Trados_Studio_2015/client_en/FileTypes/Creat ing-a-File-Type.html> [Accessed 16 November 2015].

SDL PLC, 2015e. *XML Whitespace*. [online] Available at: <http://producthelp.sdl.com/SDL_Trados_Studio_2015/client_en/FileTypes/XML-Whitespace-Normalization.html> [Accessed 28 October 2015].

SDL PLC, 2015f. *About the Hunspell Spell Checker*. [online] Available at: <http://producthelp.sdl.com/SDL_Trados_Studio_2015/client_en/setting_preferences/Check_Spelling/HunspellSpellChecker.htm> [Accessed 2 December 2015].

Siano, V., 2015. *A Comparative Study of Two FOSS Localisation Tools : Pootle and Virtaal.* M.A. Univ. Genève.

Starlander, M. and Morado Vázquez, L., 2013. Training translation students to evaluate CAT tools using Eagles: a case study. In: *Translating and the Computer 35*. London: Aslib. Available at: < http://archive-ouverte.unige.ch/unige:35622> [Accessed 16 October 2015].

W3C, 2015a. *Changes from SVG 1.1 — SVG 2*. [online] Available at: <http://www.w3.org/TR/SVG2/changes.html> [Accessed 19 October 2015].

W3C, 2015b. *Linking — SVG 2*. [online]. Available at: <http://www.w3.org/TR/SVG2/linking.html> [Accessed 11 December 2015].

W3C, 2011a. *Concepts – SVG 1.1 (Second Edition)*. [online] Available at: <http://www.w3.org/TR/SVG/concepts.html> [Accessed 16 December 2015].

W3C, 2011b. *Introduction – SVG 1.1 (Second Edition)*. [online] Available at: <http://www.w3.org/TR/SVG/intro.html> [Accessed 17 October 2015].

W3C, 2011c. *Text – SVG 1.1 (Second Edition)*. [online]. Available at: <http://www.w3.org/TR/SVG/text.html> [Accessed 17 October 2015].

W3C, 2011d. *Element Index – SVG 1.1 (Second Edition)*. [online] Available at: <http://www.w3.org/TR/SVG/eltindex.html> [Accessed 18 October 2015].

W3C, 2011e. *Document Structure – SVG 1.1 (Second Edition)*. [online] Available at: <http://www.w3.org/TR/SVG/struct.html> [Accessed 18 October 2015].

W3C, 2011f. *Paths – SVG 1.1 (Second Edition)*. [online] Available at: <http://www.w3.org/TR/SVG/paths.html> [Accessed 19 October 2015].

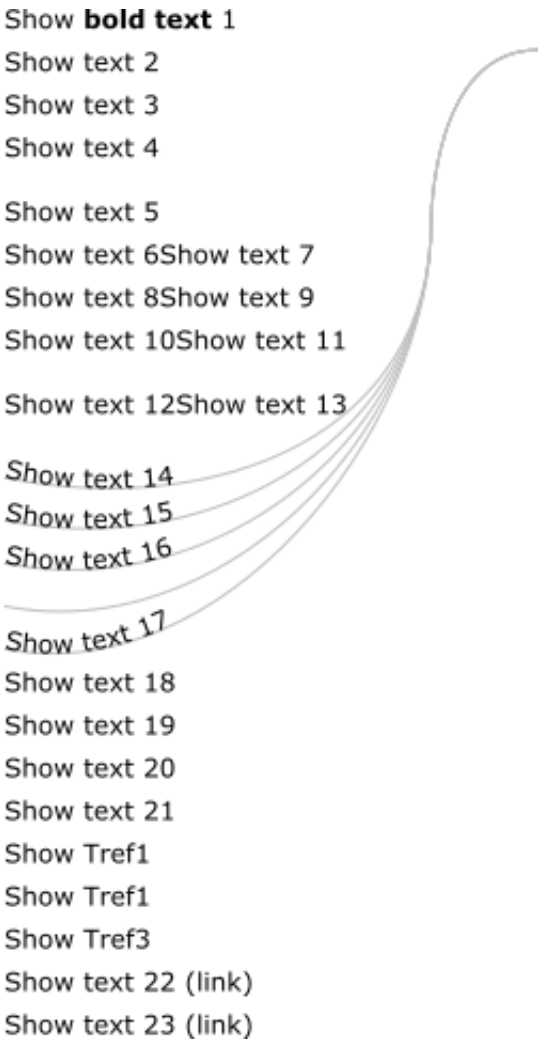
W3C, 2011g. *Fonts – SVG 1.1 (Second Edition)*. [online] Available at: <http://www.w3.org/TR/SVG/fonts.html> [Accessed 19 October 2015].

W3C, 2010. *Secret Origin of SVG*. [online]. Available at: <http://www.w3.org/Graphics/SVG/WG/wiki/Secret_Origin_of_SVG> [Accessed 16 December 2015].

W3Schools, 2015a. *XPath Tutorial*. [online] Available at: <http://www.w3schools.com/xsl/xpath_intro.asp> [Accessed 22 October 2015].

W3Schools, 2015b. *XPath Syntax*. [online] Available at: <http://www.w3schools.com/xsl/xpath_syntax.asp> [Accessed 25 October 2015].

Appendix I – Unit Testing SVG and Source Code




```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
 <svg xmlns="http://www.w3.org/2000/svg"
    xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:ev="http://www.w3.org/2001/xml-events"
    version="1.1" baseProfile="full"
    width="400" height="500">
<title>Title (metadata)</title>
<desc>Description (metadata)</desc>
<defs>
<text id="tref1">Show Tref1</text>
<text id="tref2">Do not show Tref2</text>
</defs>
<defs systemLanguage="en">
<text id="tref3">Show Tref3</text>
```

```
</defs>
<defs systemLanguage="fr">
<text id="tref3">Do not show Tref3</text>
</defs>
<defs>
<path id="text_path1" x="0" y="0" d="M0 220 C100 240 200 200 200 100 C200 100 200
20 250 20"
  fill="none" stroke="silver" />
<path id="text_path2" x="0" y="0" d="M0 240 C100 260 200 200 200 100 C200 100 200
20 250 20"
  fill="none" stroke="silver" />
<path id="text_path3" x="0" y="0" d="M0 260 C100 280 200 200 200 100 C200 100 200
20 250 20"
  fill="none" stroke="silver" />
<path id="text_path4" x="0" y="0" d="M0 280 C100 300 200 200 200 100 C200 100 200
20 250 20"
  fill="none" stroke="silver" />
<path id="text_path5" x="0" y="0" d="M0 300 C100 320 200 200 200 100 C200 100 200
20 250 20"
  fill="none" stroke="silver" />
</defs>


<text x="0" y="10">Show <tspan style="font-weight:bold">bold text</tspan> 1</text>
<text x="0" y="30">Show text 2</text>
<text systemLanguage="en" x="0" y="50">Show text 3</text>
<text systemLanguage="en-US" x="0" y="70">Show text 4</text>
<text systemLanguage="fr" x="0" y="80">Do not show text 1</text>
<text systemLanguage="de" x="0" y="90">Do not show text 2</text>
<text systemLanguage="en, fr" x="0" y="100">Show text 5</text>

<text x="0" y="120"><tspan>Show text 6</tspan> <tspan>Show text 7</tspan></text>
<text systemLanguage="en" x="0" y="140"><tspan>Show text 8</tspan> <tspan>Show
text 9</tspan></text>
<text systemLanguage="en-US" x="0" y="160"><tspan>Show text 10</tspan>
<tspan>Show text 11</tspan></text>
<text systemLanguage="fr" x="0" y="170"><tspan>Do not show text 3</tspan>
<tspan>Do not show text 4</tspan></text>
<text systemLanguage="en, fr" x="0" y="190"><tspan>Show text 12</tspan>
<tspan>Show text 13</tspan></text>

<use xlink:href="#text_path1" />
<text><textPath xlink:href="#text_path1">Show text 14</textPath></text>
<use xlink:href="#text_path2" />
<text systemLanguage="en"><textPath xlink:href="#text_path2">Show text
15</textPath></text>
<use xlink:href="#text_path3" />
<text systemLanguage="en-US"><textPath xlink:href="#text_path3">Show text
16</textPath></text>
```

```
<use xlink:href="#text_path4" />
<text systemLanguage="it"><textPath xlink:href="#text_path4">Do not show text
5</textPath></text>
<use xlink:href="#text_path5" />
<text systemLanguage="en-US"><textPath xlink:href="#text_path5">Show text
17</textPath></text>

<switch>
<text systemLanguage="fr" x="0" y="320">Do not show text 6</text>
<text systemLanguage="it" x="0" y="320">Do not show text 7</text>
<text x="0" y="320">Show text 18</text>
</switch>

<switch>
   <g systemLanguage="en"><text x="0" y="340">Show text 19</text></g>
   <g systemLanguage="fr"><text x="0" y="340">Do not show text 8</text></g>
   <g systemLanguage="it"><text x="0" y="340">Do not show text 9</text></g>
</switch>

<switch>
<text systemLanguage="fr" x="0" y="360"><tspan>Do not show text 10</tspan></text>
<text systemLanguage="it" x="0" y="360"><tspan>Do not show text 11</tspan></text>
<text x="0" y="360"><tspan>Show text 20</tspan></text>
</switch>

<switch>
<text systemLanguage="en-US" x="0" y="380"><tspan>Show text 21</tspan></text>
<text systemLanguage="en-GB" x="0" y="380"><tspan>Do not show text
12</tspan></text>
<text x="0" y="380"><tspan>Do not show text 13</tspan></text>
</switch>

<text x="0" y="400"><tref xlink:href="#tref1" /></text>
<text systemLanguage="en" x="0" y="420"><tref xlink:href="#tref1" /></text>
<text systemLanguage="fr" x="0" y="430"><tref xlink:href="#tref2" /></text>
<text x="0" y="440"><tref xlink:href="#tref3" /></text>

<a xlink:href="www.google.com"><text x="0" y="460">Show text 22 (link)</text></a>
<a systemLanguage="de" xlink:href="www.google.de"><text x="0" y="470">Do not
show text 14 (link)</text></a>
<a systemLanguage="en" xlink:href="www.google.com"><text x="0" y="480">Show text
23 (link)</text></a>
</svg>
```

Appendix II – General Information Sheet

INFORMATION SHEET

Localisation of vector graphics: A comparison in Scalable Vector Graphics localisation with SDL Trados Studio 2015 and Inkscape 0.91

The aim of this research project is to determine which tool between SDL Trados Studio 2015 or Inkscape 0.91 is the most adequate for the localisation of Scalable Vector Graphics (SVG) for users with no or little experience in the field of image localisation.

Your task in the experiment is the translation of two SVG files with the help of SDL Trados Studio 2015 and Inkscape 0.91 (one vector graphic for each program). The quality of your translation will not be evaluated.

Your on-screen work is recorded with BB FlashBack Express 5 and you will receive several questionnaires to evaluate your experience. You will be given a general questionnaire at the beginning of the experiment and after each translation process a questionnaire on your experience with each tool. At the end of the experiment, you are given a short questionnaire to assign a weight to the evaluated criteria.

The experiment will last approximately 90 minutes.

Your participation in the experiment is voluntary and you can withdraw at any time.

The data gathered from the translation process and questionnaires is evaluated by participant number. Your name only needs to be stated in the consent form. Personal data is treated confidentially and will not be revealed. Any information that could identify you is excluded from the thesis and will not be made public.

If you have any questions, please contact the researcher, Andrea Gutwein:
andrea.gutwein@etu.unige.ch

Appendix III – Information Sheet Image Localisation

INFORMATION SHEET

Image Localisation

Localisation is "the process of adapting a product or content to a specific locale or market" with the aim of giving "a product the look and feel of having been created specifically for a target market, no matter their language, culture, or location" (GALA, 2015).

According to the Globalization & Localization Association (GALA) (ibid.), the localization includes translation and possibly also the following points:

- *Adapting graphics to target markets*
- *Modifying content to suit the tastes and consumption habits of other markets*
- *Adapting design and layout to properly display translated text*
- *Converting to local requirements (such as currencies and units of measure)*
- *Using proper local formats for dates, addresses, and phone numbers*
- *Addressing local regulations and legal requirements*

Images are of utmost importance in digital content and they might need to be culturally and linguistically adapted. There are two types of images:

- **Pixel images**, such as JPEG, GIF and BMP. These are defined by the position and colour of pixels and modifying their size reduces their quality.
- **Vector images**, such as AI (Adobe Illustrator) and SVG (Scalable Vector Graphics). These are defined by mathematical formulas and modifying their size does not affect their quality. Localising a vector image is relatively easy because the text is normally isolated.

During this experiment, you will translate two SVG images using the CAT tool SDL Trados Studio 2015 and the vector graphics editor Inkscape 0.91.

Appendix IV – Information Sheet SVG Translation Inkscape 0.91

UNIVERSITÉ DE GENÈVE
FACULTÉ DE TRADUCTION ET D'INTERPRÉTATION

INFORMATION SHEET

SVG Translation in Inkscape 0.91

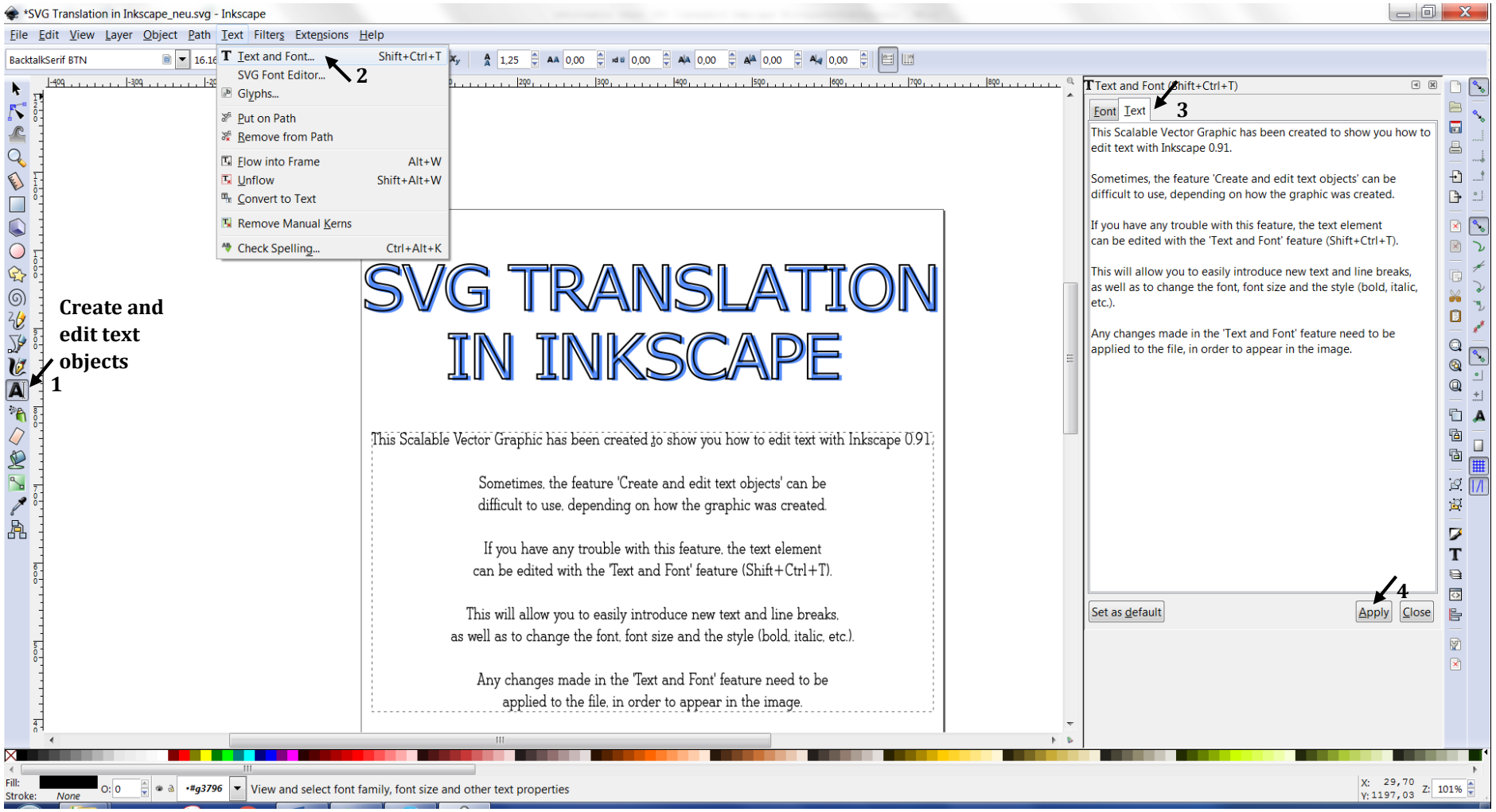
If you want to translate an SVG file in Inkscape, you can select the feature 'Create and Edit text objects' (marked by the symbol  , number 1 in the following screenshot) on the left-hand menu bar. Simply click on a text element and write over the text. Be careful to select the line that you want to translate. Sometimes, this feature can be difficult to use, depending on how the graphic was created.

If you cannot make line breaks with the 'Create and Edit text objects' feature, it might be easier to directly change the text with the 'Text and Font' feature (Shift+Ctrl+T, number 2 in the following screenshot).

In the 'Text and Font' feature, you can select the 'Text' tab (number 3 in the following screenshot) to view and edit the text of the selected element. It will allow you to easily introduce new text and line breaks, as well as to change the font, font size and the style (bold, italic, etc.). Any changes made with the 'Text and Font' feature need to be applied to the file by clicking the 'Apply' button (number 4 in the following screenshot) in order to appear in the image.

Some graphics may superimpose two or more text elements to create a special font, e.g. for headlines. In this case, it is necessary to translate all of the elements.

A quick demonstration will show you how to translate an SVG file with these features.


*SVG Translation in Inkscape_neu.svg - Inkscape
File Edit View Layer Object Path Text Filters Extensions Help
Text and Font... Shift+Ctrl+T
SVG Font Editor...
Glyphs...
Put on Path
Remove from Path
Flow into Frame Alt+W
Unflow Shift+Alt+W
Convert to Text
Remove Manual Kerns
Check Spelling... Ctrl+Alt+K
2
Create and edit text objects
1
SVG TRANSLATION IN INKSCAPE
This Scalable Vector Graphic has been created to show you how to edit text with Inkscape 0.91.
Sometimes, the feature 'Create and edit text objects' can be difficult to use, depending on how the graphic was created.
If you have any trouble with this feature, the text element can be edited with the 'Text and Font' feature (Shift+Ctrl+T).
This will allow you to easily introduce new text and line breaks, as well as to change the font, font size and the style (bold, italic, etc.).
Any changes made in the 'Text and Font' feature need to be applied to the file, in order to appear in the image.
Text and Font (Shift+Ctrl+T)
Font Text
3
This Scalable Vector Graphic has been created to show you how to edit text with Inkscape 0.91.
Sometimes, the feature 'Create and edit text objects' can be difficult to use, depending on how the graphic was created.
If you have any trouble with this feature, the text element can be edited with the 'Text and Font' feature (Shift+Ctrl+T).
This will allow you to easily introduce new text and line breaks, as well as to change the font, font size and the style (bold, italic, etc.).
Any changes made in the 'Text and Font' feature need to be applied to the file, in order to appear in the image.
Set as default
4
Apply
Close
Fill:
Stroke: None
View and select font family, font size and other text properties

Appendix V – SVG for the Demonstration of SVG Translation in Inkscape 0.91

This Scalable Vector Graphic has been created to show you how to edit text with Inkscape 0.91.

Sometimes, the feature 'Create and edit text objects' can be difficult to use, depending on how the graphic was created.

If you have any trouble with this feature, the text element can be edited with the 'Text and Font' feature (Shift+Ctrl+T).

This will allow you to easily introduce new text and line breaks, as well as to change the font, font size and the style (bold, italic, etc.).

Any changes made in the 'Text and Font' feature need to be applied to the file, in order to appear in the image.

Appendix VI – Consent Form

CONSENT FORM

Localisation of vector graphics: A comparison in Scalable Vector Graphics localisation with SDL Trados Studio 2015 and Inkscape 0.91

Researcher: Andrea Gutwein
Faculty of Translation and Interpreting (FTI)
MA in Translation with specialisation in Translation Technologies

Supervisor: Lucía Morado Vázquez
Examiners: Silvia Rodríguez Vázquez, Violeta Seretan

Please tick the following boxes if you are ready to start:

☐ I confirm that I have read and understood the information sheet for the study. I have had the opportunity to consider the information and to ask questions that were answered satisfactorily.

☐ I understand that my participation is voluntary and that I am free to withdraw from the study at any time, without having to give a reason and without any consequence.

☐ I understand that data collected throughout this study may be stored and processed by the researcher.

☐ I understand that any information gathered during the study will remain confidential and no information that identifies me will be made publicly available.

_____	_____	_____
Participant's name	Date	Signature
_____	_____	_____
Researcher's name	Date	Signature

Appendix VII – Instruction Sheet Group 1 SDL Trados Studio 2015

INSTRUCTION SHEET GROUP 1

SDL Trados Studio 2015

The researcher will log into the FTI-W7-64 server with the VMWare Horizon Client and install the necessary software before your arrival. The USB drive that you have been given contains a translation project and the SVG file that has to be translated ('State of Creative Commons_EN_Part 1', located in the folder 'SVG for Translation'). You may open and view the SVG file using a browser (e.g. Internet Explorer) if you need more context for your translation. Please read the instructions carefully before starting the experiment. If you have any questions, do not hesitate to ask.

1. When you are ready to start the experiment, notify the experiment director to activate the program that will record your work.

2. Fill out the 'General Questionnaire', which is located in the folder 'Questionnaires'. Do not forget to save the changes.

3. Open the **FTI-W7-64** server with the VMWare Horizon Client.

4. Open SDL Trados Studio 2015 and open the project 'SVG Localisation SDL Trados Studio 2015 Part 1', which is located in the folder 'Translation Project'. It already contains the SVG file that you have to translate ('State of Creative Commons_EN_Part 1').

5. Translate the SVG file in the 'Editor'. You can confirm translated segments by pressing Ctrl+Enter.

6. When the translation has been completed, save the project (Ctrl+S) and the target file (SAVE TARGET AS; Shift+F12) in the folder 'Results' on the USB drive and view the resulting translation to assess your satisfaction in the questionnaire.

7. Fill out the 'SVG Localisation Questionnaire SDL Trados Studio 2015', which is located in the folder 'Questionnaires'. Do not forget to save the changes.

8. Please proceed to the translation with Inkscape 0.91 with the ‘Instruction Sheet Group 1 Inkscape 0.91’.

Appendix VIII – Instruction Sheet Group 1 Inkscape 0.91

UNIVERSITÉ DE GENÈVE
FACULTÉ DE TRADUCTION ET D'INTERPRÉTATION

INSTRUCTION SHEET GROUP 1

Inkscape 0.91

The USB drive that you have been given contains the SVG file that has to be translated ('State of Creative Commons_EN_Part 2', located in the folder 'SVG for Translation'). You may open the SVG file using a browser (e.g. Internet Explorer) if you want to view the file. Please read the instructions carefully before starting the translation. If you have any questions, do not hesitate to ask.

1. Open the **FTI-W7-64** server with the VMWare Horizon Client.

2. Open Inkscape 0.91 and open the SVG image called 'State of Creative Commons_EN_Part 2', which is located in the folder 'SVG for translation'.

3. Translate the SVG file. You can use the feature 'Create and Edit text objects' (marked by the symbol ) on the left-hand menu bar to write over the text.

4. When the translation has been completed, save the translated file in the folder 'Results' on the USB drive (**DO NOT use 'Shift+Ctrl+S', since this will stop the screen recording**). View the resulting translation to assess your satisfaction in the questionnaire.

5. Fill out the 'SVG Localisation Questionnaire Inkscape 0.91' and the 'Criteria Weighting Questionnaire', which are located in the folder 'Questionnaires'. Do not forget to save the changes.

6. When you have completed steps 1-5, notify the experiment director to save the screen recording and to collect the USB drive if you have already completed both translations and filled out the questionnaires.

Thank you for your time and participation!

Appendix IX – Instruction Sheet Group 2 Inkscape 0.91

UNIVERSITÉ DE GENÈVE
FACULTÉ DE TRADUCTION ET D'INTERPRÉTATION

INSTRUCTION SHEET GROUP 2

Inkscape 0.91

The researcher will log into the FTI-W7-64 server with the VMWare Horizon Client and install the necessary software before your arrival. The USB drive that you have been given contains the SVG file that has to be translated ('State of Creative Commons_EN_Part 1', located in the folder 'SVG for Translation'). You may open the SVG file using a browser (e.g. Internet Explorer) if you want to view the file. Please read the instructions carefully before starting the experiment. If you have any questions, do not hesitate to ask.

1. When you are ready to start the experiment, notify the experiment director to activate the program that will record your work.

2. Fill out the 'General Questionnaire', which is located in the folder 'Questionnaires'. Do not forget to save the changes.

3. Open the **FTI-W7-64** server with the VMWare Horizon Client.

4. Open Inkscape 0.91 and open the SVG image called 'State of Creative Commons_EN_Part 1', which is located in the folder 'SVG for translation'.

5. Translate the SVG file. You can use the feature 'Create and Edit text objects' (marked by the symbol ) on the left-hand menu bar to write over the text.

6. When the translation has been completed, save the translated file in the folder 'Results' on the USB drive. **DO NOT use 'Shift+Ctrl+S', since this will stop the screen recording.** View the resulting translation to assess your satisfaction in the questionnaire.

7. Fill out the 'SVG Localisation Questionnaire Inkscape 0.91', which is located in the folder 'Questionnaires'. Do not forget to save the changes.

8. Please proceed to the translation with SDL Trados Studio 2015 with the ‘Instruction Sheet Group 2 SDL Trados Studio 2015’.

Appendix X – Instruction Sheet Group 2 SDL Trados Studio 2015

INSTRUCTION SHEET GROUP 2

SDL Trados Studio 2015

The USB drive that you have been given contains a translation project and the SVG file that has to be translated ('State of Creative Commons_EN_Part 2', located in the folder 'SVG for Translation'). You may open and view the SVG file using a browser (e.g. Internet Explorer) if you need more context for your translation. Please read the instructions carefully before starting the translation. If you have any questions, do not hesitate to ask.

1. Open the **FTI-W7-64** server with the VMWare Horizon Client.

2. Open SDL Trados Studio 2015 and open the project 'SVG Localisation SDL Trados Studio 2015 Part 2', which is located in the folder 'Translation Project'. It already contains the SVG file that you have to translate ('State of Creative Commons_EN_Part 2').

3. Translate the SVG file in the 'Editor'. You can confirm translated segments by pressing Ctrl+Enter.

4. When the translation has been completed, save the project (Ctrl+S) and the target file (SAVE TARGET AS; Shift+F12) in the folder 'Results' on the USB drive and view the resulting translation to assess your satisfaction in the questionnaire.

5. Fill out the 'SVG Localisation Questionnaire SDL Trados Studio 2015' and the 'Criteria Weighting Questionnaire', which are located in the folder 'Questionnaires'. Do not forget to save the changes.

6. When you have completed steps 1-5, notify the experiment director to save the screen recording and to collect the USB drive if you have already completed both translations and filled out the questionnaires.

Thank you for your time and participation!

Appendix XI – General Questionnaire

GENERAL QUESTIONNAIRE

Please answer this questionnaire, which is used to gather information on your profile.

Participant Number: Date:

1. Age: _____

2. Native Language(s): _____

3. English level:

☐ Native speaker

☐ Proficient

☐ Advanced

☐ Intermediate

☐ Elementary

☐ Other

Please specify: _____

4. Current position:

☐ Student

☐ Localiser

☐ Translator

☐ Other

Please specify: _____

5. When did you take the 'Traduction assistée par ordinateur (TAO)' course at the University of Geneva?

_____

6. When did you take the 'Localisation et gestion de projet' course at the University of Geneva?

_____

7. Which is/was your area of specialisation in the MA in Translation at the University of Geneva?

_____

8. What is/was your language combination at the MA in Translation at the University of Geneva?

_____

COMPUTER SKILLS

9. How often do you use a computer or a similar device (such as a tablet)?

☐ At least once a day

☐ 5-6 times a week

☐ 2-4 times a week

☐ At least once a month

☐ Less than once a month

☐ Other

Please specify: _____

10. How many hours per day do you use a computer or a similar device (such as a tablet)?

☐ 5 or more hours per day

☐ 4 hours per day

☐ 3 hours per day

☐ 2 hours per day

☐ 1 hour per day

☐ Less than 1 hour per day

11. In which context do you use a computer or a similar device? (multiple answers possible)

☐ Work

☐ Education

☐ Research

☐ Communication

☐ Leisure

☐ Other

Please specify: _____

12. Which Operating System do you normally use? (multiple answers possible)

☐ Windows

☐ Mac OS

☐ Linux

☐ Other

Please specify: _____

USER SECTION

13. How much experience do you have in the field of image localisation?

☐ I am a professional (image) localiser.

☐ I occasionally work in the field of image localisation.

☐ I only have academic, but no practical experience.

☐ I don't have any experience in the field of image localisation.

☐ Other

Please specify: _____

14. Do you use CAT tools?

☐ Yes ☐ No

15. If yes, please describe in a few lines your experience with CAT-tools (which program(s) do you normally use, when did you start using them, frequency of use, for which types of files do you normally use them, etc.)

_____

_____

_____

_____

16. How much experience do you have with the CAT tool SDL Trados Studio?

☐ I use SDL Trados Studio professionally or often work with it.

☐ I occasionally work with SDL Trados Studio.

☐ I only have academic, but no practical experience.

☐ I don't have any experience with SDL Trados Studio.

☐ Other

Please specify: _____

17. Do you have a SDL Trados Studio certification? Please indicate the level and the version of SDL Trados Studio.

☐ I do not have a SDL Trados Studio certification.

☐ Level 1

☐ Level 2

☐ Level 3

☐ Please specify the version of your SDL Trados Studio certification (e.g. 2015):

_____

18. Please describe in a few lines your experience with SDL Trados Studio (when did you start using it, frequency of use, for which type of files do you normally use it, etc.)

_____

_____

_____

_____

19. Do you use image editing software?

☐ Yes ☐ No

20. If yes, please describe in a few lines your experience with image editing software (which program(s) do you normally use, when did you start using them, frequency of use, for which types of images – pixel or vector – do you normally use them, etc.)

_____

_____

_____

_____

21. How much experience do you have with the vector graphics software Inkscape?

☐ I use Inkscape professionally or often work with it.

☐ I occasionally work with Inkscape.

☐ I only have academic, but no practical experience.

☐ I don't have any experience with Inkscape.

☐ Other

Please specify: _____

22. Please describe in a few lines your experience with Inkscape (when did you start using it, frequency of use, etc.)

_____

_____

_____

_____

Appendix XII – SVG Localisation Questionnaire SDL Trados Studio 2015

SVG Localisation Questionnaire

SDL Trados Studio 2015

Please answer this questionnaire to help evaluate the tools that were used for the translation of an SVG file.

Participant Number: Date:

USABILITY

User-friendliness

1. Is SDL Trados Studio 2015 user-friendly, with features that are easy to use?

☐	☐	☐	☐	☐
1 Not user-friendly	2	3	4	5 Very user-friendly

2. Is the interface of SDL Trados Studio 2015 intuitive?

☐	☐	☐	☐	☐
1 Not intuitive	2	3	4	5 Very intuitive

Further comments:

_____

_____

Difficulty of learning how to use the tool

3. How difficult was it to learn how to use SDL Trados Studio 2015?

☐	☐	☐	☐	☐
1	2	3	4	5
Very difficult				Very easy

4. How difficult would it have been to carry out this experiment with SDL Trados Studio 2015 without any previous instructions on how to use the tool (a university course on its functioning)?

☐	☐	☐	☐	☐
1	2	3	4	5
Very difficult				Very easy

5. Do you think SDL Trados Studio 2015 is an appropriate tool for beginners in SVG localisation?

☐	☐
Yes	No

Further comments:

_____

_____

Attractiveness

6. How attractive (pleasing to the eye) is the interface of SDL Trados Studio 2015?

☐	☐	☐	☐	☐
1	2	3	4	5
Not attractive				Very attractive

Further comments:

_____

_____

Advantages and disadvantages

7. Please list the advantages of SDL Trados Studio 2015 that you have noticed during the translation of the SVG image.

_____

_____

_____

_____

8. Please list the disadvantages of SDL Trados Studio 2015 that you have noticed during the translation of the SVG image.

_____

_____

_____

_____

9. Would you change anything about SDL Trados Studio 2015 concerning SVG translation?

_____

_____

_____

_____

Further comments:

_____

_____

Overall satisfaction

10. Did you enjoy working with SDL Trados Studio 2015?

☐ Yes ☐ No

11. Do you think that SDL Trados Studio 2015 is a good tool for the translation of SVG files?

☐ Yes ☐ No

12. How satisfied are you with the resulting translation from SDL Trados Studio 2015?

☐	☐	☐	☐	☐
1	2	3	4	5
Not satisfied				Very satisfied

Further comments:

_____

_____

Further remarks on SVG localisation in SDL Trados Studio 2015 (optional):

_____

_____

_____

_____

Appendix XIII – SVG Localisation Questionnaire Inkscape 0.91

SVG Localisation Questionnaire

Inkscape 0.91

Please answer this questionnaire to help evaluate the tools that were used for the translation of an SVG file.

Participant Number: Date:

USABILITY

User-friendliness

1. Is Inkscape 0.91 user-friendly, with features that are easy to use?

☐	☐	☐	☐	☐
1 Not user-friendly	2	3	4	5 Very user-friendly

2. Is the interface of Inkscape 0.91 intuitive?

☐	☐	☐	☐	☐
1 Not intuitive	2	3	4	5 Very intuitive

Further comments:

_____

_____

Difficulty of learning how to use the tool

3. How difficult was it to learn how to use Inkscape 0.91?

☐	☐	☐	☐	☐
1	2	3	4	5
Very difficult				Very easy

4. How difficult would it have been to carry out this experiment with Inkscape 0.91 without any previous instructions on how to use the tool (instruction sheet and/or a university course on its functioning)?

☐	☐	☐	☐	☐
1	2	3	4	5
Very difficult				Very easy

5. Do you think Inkscape 0.91 is an appropriate tool for beginners in SVG localisation?

☐	☐
Yes	No

Further comments:

_____

_____

Attractiveness

6. How attractive (pleasing to the eye) is the interface of Inkscape 0.91?

☐	☐	☐	☐	☐
1	2	3	4	5
Not attractive				Very attractive

Further comments:

_____

_____

Advantages and disadvantages

7. Please list the advantages of Inkscape 0.91 that you have noticed during the translation of the SVG image.

_____

_____

_____

_____

8. Please list the disadvantages of Inkscape 0.91 that you have noticed during the translation of the SVG image.

_____

_____

_____

_____

9. Would you change anything about Inkscape 0.91 concerning SVG translation?

_____

_____

_____

_____

Further comments:

_____

_____

Overall satisfaction

10. Did you enjoy working with Inkscape 0.91?

☐ Yes ☐ No

11. Do you think that Inkscape 0.91 is a good tool for the translation of SVG files?

☐ Yes ☐ No

12. How satisfied are you with the resulting translation from Inkscape 0.91?

☐	☐	☐	☐	☐
1	2	3	4	5
Not satisfied				Very satisfied

Further comments:

_____

_____

Appendix XIV – Criteria Weighting

UNIVERSITÉ DE GENÈVE
FACULTÉ DE TRADUCTION ET D'INTERPRÉTATION

CRITERIA WEIGHTING

Participant Number: Date:

Please give a score from 1 (not important) to 5 (very important) to the following characteristics or attributes in relation to their importance for SVG localisation. Do you think they play a big or a minor role in SVG localisation?

This exercise is not linked to any program and is used to determine the general importance of the below characteristics for SVG localisation. Ask yourself, for example, if it is more important for SVG localisation that a program shows all the translatable content or that it has a spell-checking feature, and assign your points accordingly.

Sub-characteristic or Attribute	**1**	**2**	**3**	**4**	**5**
Display of translatable content (showing all the translatable text)					
Ease of editing (possibility of modifying the file after the translation has been completed)					
Consistency (possibility of ensuring consistency)					
Spell-checking					
Accuracy (no localisation errors caused by the program)					
User-friendliness (intuitive use of the functions)					
Learnability (difficulty of learning how to use the tool)					
Ease of use (functions easily accessible with few clicks)					
Attractiveness of the interface					
Time behaviour (speed)					

Thank you for your time and participation!

Appendix XV – Lists of Shortcuts and Other Special Keys for 'Operability' Evaluation

	Virtual Keys SDL Trados Studio 2015	Description[36]	P1	P2	P3	P4	P5	P6
Shortcuts	[Alt]+[§]	Wrong shortcut			1			
	[Alt]+[1]	Used to write the letter 'ñ'			1			
	[Alt]+[4]	Used to write the letter 'ñ'			1			
	[Alt]+[6]	Used to write the letter 'ñ'			1			
	[Alt]+[TAB]	Go to next window			35			
	[Ctrl]+[#]	Wrong shortcut						2
	[Ctrl]+[,]	Open a drop-down list of placeables (SDL Trados Studio)						1
	[Ctrl]+[5]	Wrong shortcut		1				
	[Ctrl]+[Alt]	Used before special characters		13	10	9		42
	[Ctrl]+[Alt]+[D]	Change tag display mode (SDL Trados Studio)						1
	[Ctrl]+[Alt]+[S]	Merge segments (SDL Trados Studio)						6

[36] Some descriptions were taken from http://www.w3schools.com/tags/ref_keyboardshortcuts.asp and https://en.wikipedia.org/wiki/Table_of_keyboard_shortcuts

	[Ctrl]+[ArwLft]	Move one word to the left			164			
	[Ctrl]+[ArwRght]	Move one word to the right			172			
	[Ctrl]+[BkSp]	Erase word to the left			26			
	[Ctrl]+[C]	Copy the selection			1			2
	[Ctrl]+[Del]	Erase word to the right		1	2			
	[Ctrl]+[H]	Replace (SDL Trados Studio)						1
	[Ctrl]+[HOME]	Erase line to the left			1			
	[Ctrl]+[Ins]	Copy source to target (SDL Trados Studio)			11	55		17
	[Ctrl]+[Ret]	Confirm and move to next unconfirmed segment (SDL Trados Studio)	71	69	88	87	66	65
	[Ctrl]+[Shf]	Wrong shortcut			28			3
	[Ctrl]+[Shift]+[,]	Wrong shortcut						3
	[Ctrl]+[Shift]+[ArwLft]	Select words to the left			13			
	[Ctrl]+[Shift]+[ArwRght]	Select words to the right			42			
	[Ctrl]+[V]	Paste contents of clipboard			4			3
	[Ctrl]+[X]	Cut the selection			4			

	[Ctrl]+[Z]	Undo the last operation						1
	[Shift]+[ArwLft]	Select characters to the left			2	4		
	[Shift]+[ArwRght]	Select characters to the right				20		
	[Shift]+[BkSp]	Wrong shortcut		1		2		
	[Shift]+[Ctrl]	Wrong shortcut			3	1		
	[Shift]+[END]	Select content until the end of the line			1			
	[Shift]+[F12]	Save target as (SDL Trados Studio)			5			
	[Shift]+[Ins]	Wrong shortcut						1
	[Shift]+[Ret]	Insert line break						1
	[Shift]+[Spc]	Wrong shortcut			3			1
Other Special Keys	ALT	Alt key			35			
	ArwDwn	Down Arrow			13	4		
	ArwLft	Left Arrow	2	5	21	99		19
	ArwRght	Right Arrow	1		26	105		43
	ArwUp	Up Arrow			39			2
	BkSp	Backspace	111	92	85	73	91	157
	CpLk	Caps Lock	2	7	4	4	58	9

Ctrl	Control	86	114	317	454	59	141
Del	Delete	1	5	8			
END	Go to end of line		5	4			
Esc	Escape			4			
HOME	Go to start of line		2	3			
Ins	Insert				9	2	
Ret	Return/Enter		1			2	11
Shf	Shift	61	59	41	168	1	79
Spc	Space	132	102	158	95	94	90
TABULATOR	Tab key	1					1

Table 17. List of shortcuts and other special keys used by the participants in SDL Trados Studio 2015

	Virtual Keys Inkscape 0.91	Description[37]	P1	P2	P3	P4	P5	P6
Shortcuts	[Alt]+[TAB]	Go to next window			1			
	[Ctrl]+[ä]	Wrong shortcut						1
	[Ctrl]+[Alt]	Used before special characters		3	21	13		1
	[Ctrl]+[Alt]+[ä]	Wrong shortcut						1
	[Ctrl]+[ArwLft]	Move one word to the left			53			
	[Ctrl]+[ArwRght]	Move one word to the right	3		41			
	[Ctrl]+[ArwUp]	Move to beginning of line break			1			
	[Ctrl]+[BkSp]	Erase word to the left			14			
	[Ctrl]+[C]	Copy the selection			1			
	[Ctrl]+[E]	Focus and select						2
	[Ctrl]+[M]	Wrong shortcut		1				
	[Ctrl]+[Ret]	Confirm and move to next unconfirmed segment (SDL Trados Studio)					2	

[37] Some descriptions were taken from http://www.w3schools.com/tags/ref_keyboardshortcuts.asp and https://en.wikipedia.org/wiki/Table_of_keyboard_shortcuts

[Ctrl]+[Shf]	Wrong shortcut	9		21			2
[Ctrl]+[Shift]+[ä]	Wrong shortcut						3
[Ctrl]+[Shift]+[ArwLft]	Select words to the left	7		1			
[Ctrl]+[Shift]+[ArwRght]	Select words to the right	16		85			
[Ctrl]+[Shift]+[END]	Select text between the cursor and the end of the document			1			
[Ctrl]+[ß]	Wrong shortcut						1
[Ctrl]+[T]	Wrong shortcut			1			
[Ctrl]+[V]	Paste contents of clipboard			1			
[Ctrl]+[Y]	Redo the last operation	1			8		6
[Ctrl]+[Z]	Undo the last operation	18	23	99	110	104	268
[Shift]+[ArwDwn]	Select lines downwards			3	5		
[Shift]+[ArwLft]	Select characters to the left			12	81		
[Shift]+[ArwRght]	Select characters to the right		11	15	445		
[Shift]+[Ctrl]	Wrong shortcut			9			1
[Shift]+[END]	Select content until the end of the line			29			

	[Shift]+[HOME]	Select text between the cursor and the beginning of the current line			3			
	[Shift]+[Ret]	Insert line break		1			12	
Other Special Keys	ALT	Alt key			1			
	ArwDwn	Down Arrow	2	21	9	13	25	45
	ArwLft	Left Arrow	5	20	38	101	35	135
	ArwRght	Right Arrow	39	69	24	251	281	137
	ArwUp	Up Arrow	3	3	10	2	3	10
	BkSp	Backspace	159	88	69	277	498	336
	CpLk	Caps Lock	6	4	8	6	22	10
	Ctrl	Control	31	33	145	98	88	36
	Del	Delete	14	190	15			529
	END	Go to end of line		1	10			
	HOME	Go to start of line			23			
	Ins	Insert		1	1		1	
	Ret	Return/Enter	17	14	13	31	13	20
	Shf	Shift	122	22	71	82	12	29
	Spc	Space	90	46	111	149	90	164
	TABULATOR	Tab key			2			2

Table 18. List of shortcuts and other special keys used by the participants in Inkscape 0.91

Appendix XVI – Lists of Advantages, Disadvantages and Changes for 'Advantages and Disadvantages' Evaluation

Part. No.	Advantages SDL Trados Studio 2015	Advantages Inkscape 0.91
P1	- Clear interface - It shows the matches, segments already translated which is helpful - The translation of images is similar to the translation [of] any text (for a translator used to use [T]rados [it] is easy to use)	- The main advantage is that you can see the image and you work on the image, so the user can rearrange the text, boxes, etc, in order to obtain a better result - The possibility of using the box to write the text outside the image is very useful - Once getting used to the interface, it is easy to use - The user can modify the sentences, in order to have a better result (as oppos[ed to] [T]rados, when, for instance, sometimes the translator has to maybe reformulate the translation because the sentence has been divided into two segments)
P2	- Edition is easy - [W]e have less chances of damaging the image while translating	- We can edit changes that we made rather easily - We can see the results in the image directly in terms of space and colors
P3	- Isolates text - Reuses previously translated segments - Automatically translates numbers (fewer chances to make mistakes)	- Not many advantages. There was no need to change some text, such as some country names like Argentina or Indonesia

	- Harmonization of corrected segments (I originally translated and confirmed the first segment containing just a percentage without adding the article in Spanish, I then realized I had to add the article so I changed it, and Trados added the articles to all other segments containing percentages; this favours the overall coherence of the translation)	
P4	- The translator can work in a well-known environment (considering that most translators work with some sort of CAT tool like Trados or similar, but not so much with image editors) - [T]he translator does not have to worry about moving elements	- Intuitive - [A]llows to move elements around - [T]wo functions for editing text
P5	- Trados always guesses what you're going to translate because it takes into account your previous translations and that is pretty helpful - If you know how to use it, you can translate really fast - It gives you [A]uto[Su]ggest dictionaries and termbases if you need them	- Easy to use - You can move elements like images and the text pretty easily - It is friendly to the eye - You easily guess how to use it
P6	- There is no need to be clicking at the text and dealing with the image at all, you can be totally concentrated in the content	- Translating with the 'text' tab was very easy

Table 19. List of the advantages given by the participants

Part. No.	Disadvantages SDL Trados Studio 2015	Disadvantages Inkscape 0.91
P1	- Segments start with capital letters - Sentences that are divided in different segments - The format is not copied from source to target text, the translator has to remember to adapt the format (i.e. Capital letters) - The image is not available, the translator has to open it aside - No preview of the result of the translation of the image - The translation of images is similar to the translation [of] any text (in this case [it] is a disadvantage because it is not a regular text, the translator can't see the image while he's translating)	- The interface could be a bit more difficult at the beginning - When translating in the box, the color of the letters is not adapted, the user has to change it after applying the changes
P2	- It does not allow to change the order of the sentences which may mean that the result[ing] sentences in the target text won't make any sense	- It is difficult to select the correct image to translate - We can damage the original image if we do not know how to use the software
P3	- Could not merge segments	- Difficulty to select text - Difficulty to work inside the textbox
P4	- [B]roken segments - [T]ranslators have to open the image separately to know where the text that they are translating will be placed in the image (maybe the preview function would help if the file type was customized)	- One of the functions to edit text does not keep the original format - [H]ard to move elements around when they overlap - [H]aving to change function to brake [sic] lines and then move the elements

		- [T]he translator can accidentally move elements when trying to organize text boxes, and sometimes it can be difficult to place them back exactly where they were originally
P5	- I don't think the interface is [user-]friendly, if you don't know the program it is probably going to be pretty difficult to understand what you have to do, so I don't think this [is] a program for beginners. - Sometimes it separates segments that when you translate in Spanish need to be together in order to understand the whole meaning and sure you can merge segments but I think it takes time and you also need to understand Trados to know how to do it - I think Trados is more for professional translators	- It doesn't guess your translation or autopropagate your translations - It takes too long to complete a simple task because you have to do so much editing
P6	- The segments are incredibly short and there were moments when I could've used the segment merging to make a proper translation that would save correctly in the TM. Since this wasn't possible you had to choose between having wrong entries in your TM or having a wrong translation	- Translating with the '[C]reate and edit text objects' was more difficult in terms of line breaks

Table 20. List of the disadvantages given by the participants

Part. No.	Changes SDL Trados Studio 2015	Changes Inkscape 0.91
P1	- It'll be useful to be able to see the image when translating	- Sometimes it is difficult to select the text in the image, especially when the text is small, so the user has to "fight" to select the correct text
P2	- I would add a tool that will allow me to visualize the image in the editor	- I would create a window that allows me to do the localisation of the text only (similar to the option 3 in the explanation sheet) where the text can be approved before adding it to the image
P3	- […] I would add information that allows the translator to identify the content of each segment with a part of the document, just as when you translate a Word document you can see when a paragraph starts and ends or whether you are translating a cell of a table - […] [I]t would also be useful to merge segments into larger, more coherent, units of sense	- I would like more complete, bigger, word-processor-like text boxes that can be easily selected
P4		- The 'create and edit text' function could work in a similar way to the text edit function in Paint, for example: that allows you to write and move the text around without

		having to change to a different function
P5	- Maybe it would be better if they came with a [...] more friendly [sic] and more easy to use [interface].	- I think it would be better if Inkscape would guess what word you are going to use in your translation based on your previous choices
P6	- The ability to merge segments	- When translating the text, color would change in random occasions causing it to sometimes blend in with the background color and add a few steps to the assignment

Table 21. List of the changes that participants would want for the tools

Appendix XVII – Summary of the Metrics and Scores

Attribute	Metrics	Score	Score SDL Trados Studio 2015	Score Inkscape 0.91
Display of translatable content	All translatable text is shown and hidden text can be accessed without further clicks	2 points	**2 points**	**1 point**
	All translatable text is shown	1 point		
	Text content elements are missing or metadata cannot be accessed	0 points		
Ease of editing	SVG files can be viewed with the tool	1 point	0 points	1 point
	SVG files cannot be viewed with the tool	0 points		
	SVG files can be edited without the use of another tool	1 point	0 points	1 point
	SVG files cannot be edited without the use of another tool	0 points		
	SVG files do not need to be edited with another tool after translation due to space or formatting issues	1 point	0 points **Overall: 0 points**	1 point **Overall: 3 points**
	SVG files need to be edited with another tool after translation due to space or formatting issues	0 points		

Consistency	The tool offers a way to ensure consistency (internal dictionary, term base, translation memory, etc.)	1 point	**1 point**	**0 points**
	The tool does not offer a way to ensure consistency (internal dictionary, term base, translation memory, etc.)	0 points		
Spell-checking	The tool offers a spell-checking feature in the target language	1 point	**1 point**	**0 points**
	The tool does not offer a spell-checking feature in the target language	0 points		
Localisation errors	The translated text does not fit in the confines of the image or overlaps with other elements	Each localisation error: - 1 point for each participant Maximum points for no localisation errors (per participant): 5 points Localisation errors are subtracted from the maximum score.	11 errors (all participants) **Overall: 19 points**	12 errors (all participants) **Overall: 18 points**
	The translated text is not in the same place as the source element, creating senseless phrases			
	The style of the text is not the same as in the source image (font, colour, etc.)			
	Text content elements or parts of the text have been erased in the translation			
	Some translatable content has not been translated (remaining in the source language)			

User-friendliness	The tool is user-friendly (Q1)	Rating: (for each participant) 5: 2 points 3 or 4: 1 point 1 or 2: 0 points	Average Q1: 4 Average Q2: 4.17 **Overall: 15 points**	Average Q1: 2.83 Average Q2: 3.17 **Overall: 9 points**
	The tool's interface is intuitive (Q2)	Rating: (for each participant) 5: 2 points 3 or 4: 1 point 1 or 2: 0 points		
Difficulty of learning how to use the tool	Difficulty of learning how to use the tool (Q3)	Rating: (for each participant) 5: 2 points 3 or 4: 1 point 1 or 2: 0 points	Average Q3: 4 Average Q4: 2 Q5: 5 points **Overall: 14 points**	Average Q3: 4 Average Q4: 2.83 Q5: 4 points **Overall: 17 points**
	Supposed difficulty of carrying out the experiment without previous instructions (Q4)	Rating: (for each participant) 5: 2 points 3 or 4: 1 point 1 or 2: 0 points		
	The tool is appropriate for beginners in SVG localisation (Q5)	1 point (for each participant)		
	The tool is not appropriate for beginners in SVG localisation (Q5)	0 points (for each participant)		

Ease of use	Clicks needed to reach the functions for the translation of SVG files from the starting point of the interface	<2 clicks: 30 points 3-4 clicks: 24 points 5-6 clicks: 18 points 7-8 clicks: 12 points 9-10 clicks: 6 point >10 clicks: 0 points	12 points	12 points
	Clicks used in the translation process	For each participant: <100 clicks: 5 points 101-200 clicks: 4 points 201-300 clicks: 3 points 301-400 clicks: 2 points 401-500 clicks: 1 point > 500 clicks: 0 points	26 points	14 points
	Virtual keys used in the translation process	For each participant: <300 virtual keys: 5 points 301-600 virtual keys: 4 points 601-900 virtual keys: 3 points 901-1200 virtual keys: 2 points 1201-1500 virtual keys: 1 point >1500 virtual keys: 0 points	19 points **Overall: 57 points**	12 points **Overall: 38 points**

Attractiveness of the interface	The tool's interface is attractive (Q6)	Rating: (for each participant) 5: 2 points 3 or 4: 1 point 1 or 2: 0 points	Average Q6: 4.17 **Overall: 9 points**	Average Q6: 3.5 **Overall: 6 points**
Advantages and disadvantages	Advantages of the tool (Q7)	≥5 advantages: 5 points 4 advantages: 4 points 3 advantages: 3 points 2 advantages: 2 points 1 advantage: 1 point 0 advantages: 0 points	Average Q7: 2.5 Average Q8: -2.33 Average Q9: -1 **Overall: 27 points**	Average Q7: 2.5 Average Q8: -2.17 Average Q9: -1 **Overall: 26 points**
	Disadvantages of the tool (Q8)	0 disadvantages or changes: 5 points 1 disadvantage or change: 4 points 2 disadvantages or changes: 3 points 3 disadvantages or changes: 2 points 4 disadvantage or changes: 1 point ≥5 disadvantages or changes: 0 points		
	Need to change anything about the tool (Q9)			

Overall satisfaction	It is enjoyable to work with the tool (Q10)	1 point (for each participant)	6 points	3 points
	It is not enjoyable to work with the tool (Q10)	0 points (for each participant)		
	The program is a good tool for the translation of SVG files (Q11)	1 point (for each participant)	4 points	3 points
	The program is not a good tool for the translation of SVG files (Q11)	0 points (for each participant)		
	Satisfaction with the resulting translation (Q12)	Rating: (for each participant) 5: 2 points 3 or 4: 1 point 1 or 2: 0 points	Average Q12: 3.33 **Overall: 15 points**	Average Q12: 3.17 **Overall: 11 points**
Time behaviour	Time needed for the translation process	≤10 minutes: 5 points 10-15 minutes: 4 points 15-20 minutes: 3 points 20-25 minutes: 2 points 25-30 minutes: 1 point >30 minutes: 0 points	Average time: 19m3s **Overall: 16 points**	Average time: 26m47s **Overall: 8 points**

Table 22. Summary of the metrics and scores used for the evaluation of SVG localisation in SDL Trados Studio 2015 and Inkscape 0.91