Journal of Assistive Technologies

ASVG - Accessible Scalable Vector Graphics: intention trees to make charts more accessible and usable

Markus Weninger Gerald Ortner Tobias Hahn Olaf Drümmer Klaus Miesenberger

Project report

ASVG – Accessible Scalable Vector Graphics: intention trees to make charts more accessible and usable

Markus Weninger, Gerald Ortner, Tobias Hahn, Olaf Drümmer and Klaus Miesenberger

Markus Weninger, Gerald Ortner and Tobias Hahn, all are based at the Johannes Kepler University Linz, Linz, Austria. Olaf Drümmer is based at callas software, Berlin, Germany. Professor Klaus Miesenberger is based at the Johannes Kepler University Linz, Linz, Austria.

Abstract

Purpose – *The purpose of this paper is to enhance accessibility of graphical information in particular for blind and visually handicapped people.*

Design/methodology/approach – *Prototype development based on an intense analysis of the state of the art and potential technologies with later on heuristic analysis of different approaches to enrich graphical information for better accessibility.*

Findings – *A novel approach to enhance accessibility named "Intention Tree" for enhancing accessibility. It allows integrating descriptive and navigation information into standard Scalable Vector Graphics and also mechanisms to analyse and aggregate data.*

Research limitations/implications – *The approach promises interesting new tools for better accessing and navigating graphical information with potential not only for blind and visually handicapped people.*

Practical implications – *Design prototype for further development.*

Social implications – *Potential for better social inclusion and participation.*

Originality/value – *This paper presents a novel and new approach for enhanced accessibility and usability and a new technique for authoring graphical information.*

Keywords *Accessibility, Usability, Accessibile SVG, Design for all, People with disabilities, Scalable vector graphics*

Paper type *Research paper*

1. Abstract

This paper proposes an innovative technique called intention tree, which supports improving accessibility of graphical information. By providing a prototype implementation for Scalable Vector Graphics (SVG) called Accessible SVG (ASVG) it demonstrates how to achieve improved accessibility and usability beyond the most widespread state of the art of using alternative text or longdesc attributes.

Received 8 October 2015
Accepted 8 October 2015

This work was supported by callas software through its CEO Olaf Drümmer both financially and in terms of content. It was supervised by the University Linz, Institute Integriert Studieren, in the frame of the European Coding for a Cause competition (C4C, www.projectpossibility.org) with its finals held at the 13th AAATE Conference in September 2015 in Budapest (www.aaate2015.eu).

2. Introduction and state of the art

With computers becoming consumer electronic the use of documents in electronic formats and the World Wide Web became ubiquitous and using them can be seen as a standard cultural technique. These developments not only started a fundamental change in almost any domain of our societies, it also allows removing barriers for people with disabilities and moving towards an inclusive society. With the help of Assistive Technologies (AT) they get independent access to the same newspapers, books, scientific papers or any other information, system or service as everybody else. Therefore the power of the web and e-documents "is in its universality. Access by everyone regardless of disability is an essential aspect" (Berners-Lee, n.d.).

The above-mentioned ATs are efficient for reading out or presenting textual content and even allow navigating through simple tables. But nested tables and complex data on the other hand (see Table I) can get very confusing and frustrating to read for people with disabilities, in particular with visual impairments.

For sighted people complex data are usually presented through various kinds of charts in a visual manner for better and faster understanding. Already in 1786 William Playfair invented line graphs and bar charts (Playfair, 1786/1801) which became a standard way of presenting data and complex relations in parallel "at a glance". ATs tend to allow access to data in a linear manner only with restricted usability compared to this visual access.

There are scientific approaches to analyse the images and to interpret the content by heuristics (Savva *et al.*, 2011). These approaches are promising but a lot of work is still needed. At the moment the only way to make images/charts accessible is to use so called "alt(ernative) texts" or "longdesc" attributes (WCAG 2.0). These alt-texts give a more or less brief textual description of what can be seen on the image/chart but do not convey the overview and intuitive easy navigation as the visual presentation. For certain graphic types like, e.g. SVG (W3C) also single graphic elements like a single bar in a bar chart could hold such an alt text (Dardailler, 2002).

This approach has a few downsides. First it asks for a lot of extra work for the author of the document when done in detail. Furthermore charts can be read in different ways as every person analyses the chart individually to receive the information most important to her/him as fast as possible. Even when the chart information is described in detail and stored as an alt-text information may be lost as charts might be presented in only one way – the way the alt text was authored. Including the table-structured data into the document next to the chart is currently the best way to provide detailed information for blind and vision-impaired people. The information might be basically accessible but it considerably lacks in terms of non-visual usability.

3. Proposing ASVG for better access

The above analysis defines key requirements to improve accessibility and usability of charts for blind and visually impaired people. First of all the information loss that occurs when the raw data are transformed into a visual representation has to be limited. Therefore certain information about the chart needs to be stored as metadata alongside the graphics, preferably embedded into the same file. Furthermore the chart needs to be interactively traversable by the user. Users of ATs should be able to navigate through the single chart elements like bars in a bar chart or data points in a line chart in an intuitive way. This would also allow the AT to highlight the currently selected chart element(s) by, e.g. magnifying it to support vision-impaired people. Having this additional metadata stored automatically with the chart at creation time would also avoid any extra workload for chart/document creators.

Table I Table-structured data

	1994		*2004*		*2014*	
	Male	*Female*	*Male*	*Female*	*Male*	*Female*
Burgenland	1.355	1.265	1.117	1.093	1.159	1.024
Carinthia	3.200	3.075	2.482	2.363	2.365	2.243
Lower Austria	8.528	8.173	7.360	6.962	7.291	7.005
Upper Austria	8.745	8.483	7.128	6.809	7.463	6.978
Salzburg	3.299	3.039	2.699	2.555	2.807	2.638
Styria	6.603	6.342	5.342	5.122	5.414	5.002
Tirol	4.412	4.173	3.635	3.403	3.595	3.471
Vorarlberg	2.436	2.223	2.037	2.005	2.075	1.932
Vienna	8.834	8.230	8.740	8.116	9.993	9.267

3.1 Technologies for improved access

Based on these goals different technologies were analysed to determine how they allow addressing these requirements. The project started with an intense experimental phase in which different approaches using HTML5 combined with Cascading Style Sheets (CSS) and JavaScript (Mozilla Developer Network, n.d.) were tried out and checked for flexibility and their general usability. All these approaches turned out as unsatisfactory mainly due to missing a general data structure that would fit all kinds of charts. It would result in a complex construction process for each type of chart. Furthermore redundancies in the stored data and the mixing of graphics and embedded metadata could lead to unnecessary overhead and file size increase.

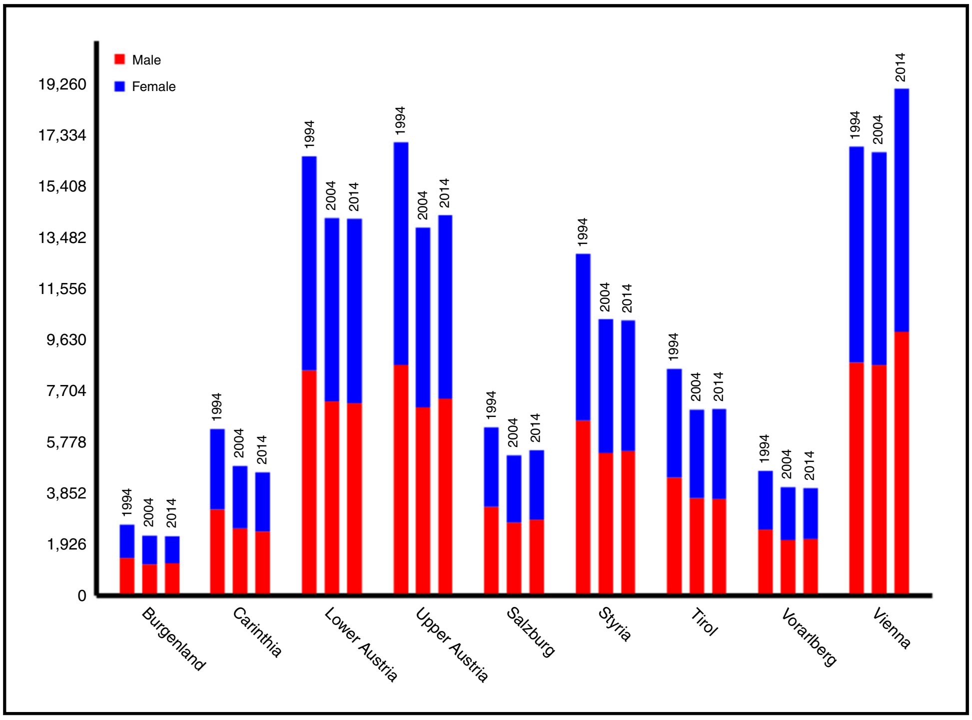
After this experimental phase the focus was on a technology which is more in line with all our requirements. SVG (W3C) is a language for describing two-dimensional graphics using XML. SVG (W3C) allows three types of graphic objects: vector graphic shapes, images and text (Duce *et al.*, 2002). The most important feature of SVG (W3C) is to insert additional embedded data to the image, which is hidden for the viewer, but enables developers to store information within the graphic. This technique is seen as a viable solution for our purpose of enhancing accessibility.

3.2 Metadata enrichment for ASVGs: intention tree

After selecting the technology to store metadata with the graphical representation the next step was to define the data structure and elements to be used. By developing use cases and trying different approaches in a rapid prototyping style we gained better insight into the problem: what information has to be stored and in which format this can be done. The primary information which is to be stored is the value of each graphical object (bar, pie slice, etc.), the secondary is the relation between the graphical objects (siblings, children, parent, etc.).

The value of each graphical object can easily be stored together with the object due to the flexible nature of SVG (W3C) and its extensibility (Figure 1).

Figure 1 A typical chart with multiple levels



Explaining the relation between the graphical objects, as our heuristic evaluation showed, is more or less straight forward. Every author of a chart has a specific intention how people should "read" the chart, which means that the author has a certain order in mind in which people should look at and analyse the chart. For example, in a bar chart, most people at first intuitively split up the received graphical information along the *x*-axis. This splits up into smaller, more easily consumable information chunks. If the chart has multiple levels, for example first split by year, then by state, those chunks can again be separated into smaller pieces. This can be repeated until the information is split up to singular graphical objects. In this way the chart can be accessed visually part by part. This is currently not possible for blind people and often very hard for people with vision impairments who would need better overview and guidance in this process.

The set of metadata should store the information about the creator's intention to enable ATs to navigate through charts in a similar way as a seeing person. We therefore propose a concept called "Intention Tree". This intention tree gets embedded in the SVG (W3C) as an XML (W3C) tree. XML (W3C) is a standard, especially suited for tree-structured information, and plays an important role in the exchange of a wide variety of data on the web and elsewhere. An example for the intention tree of a bar chart about the birth rates in the states of Austria could look like the following (not formatted as XML (W3C) for reading purpose):

```
* Chart
** State: Vienna
*** Year: 1994
**** Gender: Male
**** Gender: Female
*** Year: 2004
**** Gender: Male
**** Gender: Female
*** Year: 2014
**** Gender: Male
**** Gender: Female
** State: Upper Austria
...
```

This tree structure can now be processed and navigated with the help of AT – by going up, down, left or right in the tree – which was one of the project's main goals. We refer to each of those lines as "intention node". Such a node can have one "parent" and one/several/no "children". A node without children is called "leaf" as reference to a tree, which starts at the root and splits up further and further into branches and finally ends at its leaves. Leaves have one special property: they represent a single graphical object in a chart (e.g. a bar in a bar chart).

If an AT now navigates through this intention tree it knows exactly which graphical objects represent the currently selected intention node. This allows, e.g. graphically emphasising the selected node by highlighting the related graphical objects in the chart (e.g. drawing a border around the respective bars, changing contrast). This covers a part of a general usability issue – being able to highlight certain information in the chart while navigating through it.

But highlighting is not enough, because this is again only visual information and not helpful for blind people. As stated before we store the value of each graphical object. So if we navigate to a leaf (which references a graphical object) the intention tree exposes the value to the AT, which can present it to the user using audio or Braille.

The next step is to enable nodes, which reference multiple graphical objects (like the node "Vienna, 1994" in the above example, which has two graphical objects as children, one for "Male" and one for "Female"), to establish a value that can be presented. This shows one of the biggest advantages of the intention tree technique: it is extendable with "attributes". This means, as our understanding of how to make charts more accessible grows, the intention tree can grow with it. We propose the so called "aggregation mode". The aggregation mode describes how nodes shall process the values of their child nodes to establish their own value. Two typical aggregation modes are "Sum" and "Average", telling a node that its value is either calculated by summing up the child values or taking the child values' average. This way, nodes also can be presented to the user with a value.

3.3 Improving accessibility and usability: an intention tree example

In the following example an intention tree annotated with aggregation attributes is shown:

```
* Chart
** State: Vienna; Aggregation: Average
*** Year: 1994; Aggregation: Sum
**** Gender: Male
**** Gender: Female
*** Year: 2004; Aggregation: Sum
**** Gender: Male
**** Gender: Female
*** Year: 2014; Aggregation: Sum
**** Gender: Male
**** Gender: Female
** State: Upper Austria; Aggregation: Average
...
```

The intention tree provides a value for each node in the tree to ATs, focuses (highlights) the respective graphical objects and allows navigating the tree structure.

The following comparisons demonstrate the gained potential of the intention tree technique. All examples are based on the official birth rates in Austria, grouped by the different states, years, and the babies' gender (Statistics about Birth Rates in Austria), Statistics Austria, n.d.

First of all let us compare how the files are built and stored. A chart without the intention tree technique would most likely be built by using some already existing charting library using the data and plotting it, storing it in an arbitrary image format. When using such a chart in a document or website, authors who are aware of accessibility requirements would add an alt text or a "longdesc" attribute (longdesc), (W3C). Most probably this alt text would be just a short description and could read as: "Statistics about birth rates in Austria". For the sake of fairness let us assume the author of the graph thinks about what he wants to say with the graph and provides a more fitting alt text like: "Statistic about birth rates in Austria, showing Vienna with an 8 times higher birth rate than Burgenland" which still would be rather vague (Vázquez *et al.*, 2014).

In comparison, to be able to include an intention tree, the used charting library has to support the creation of the intention tree during chart generation. The framework, which was developed as part of this project provides functionality to generate the intention tree, which illustrates the ease of adapting existing charting libraries to this new technique. The chart's visual representation would be the same as in the previous, poorly accessible approach – but this time the file format of choice is SVG (W3C), which is based on combining graphical primitives (rectangles, circles, etc.) as opposed to arbitrary pixel-based data formats. The main difference is the hidden, embedded intention tree which provides the meta-information on how the creator intended to navigate through the chart and on how to obtain data and information from it. Due to this information the chart creator does not have to provide an alt text anymore.

The alt text approach has multiple disadvantages. One is that users are most often confined to one conclusion. They cannot, for example, look at the different states and compare their birth rates on their own, or find the state with the highest birth rates. If this information is not present as part of the alt text it is lost. Also the information about the graphical arrangement, which mostly reflects the creators' intention on how to read the chart, is lost. By using an intention tree, people who want to look at the chart can navigate to the values they want to compare, for example first navigating to Vienna and then comparing it with Burgenland.

At this stage of the project this is not as straight forward as the usage of an alt text, as tool support for the intention tree is not yet implemented and integrated into authoring tools. But already at this stage the following core improvements are to be noted:

1. users gain much more flexibility on what information they receive and on how to navigate it;
2. when relying on alt texts this information would have to be written down for each state-to-state-comparison, resulting in far too much linear text; and
3. intention trees allow retrieving much more information about a chart on demand: minimum values, maximum values, local highs, local lows, or certain trends (e.g. an upward trend in the birth rate).

Compiling all this information into an alt text with pure linear access would reduce readability and usability. While alt text may be preferable for some use cases (very simple graphics), most often the intention tree approach provides improved accessibility and usability. With some further improvements in the future, which will be explained in the next section, the intention tree approach is seen as providing better information with improved usability and accessibility without negatively impacting the effort for authors.

4. Outlook and future work

The intention tree could easily be amended with information on the entrance point. This starting information would set the node or the comparison, which should be shown when starting to read the chart. This way the chart's most important information would be shown immediately without the need for any further navigation. This would most probably be the information, which would otherwise be stated in the alt text. Also, this feature could be implemented more generally: some pre-defined positions or comparisons could be added to the intention tree, making it even easier for the author to underline important information and proposing navigation paths through the chart while still keeping the user's full freedom to navigate on her/his own. For example, if the author wants to show that Burgenland has a higher birth rate than Vorarlberg, but less then Vienna, then the comparison Burgenland/Vorarlberg and Burgenland/Vienna could be saved as two pre-defined positions in the graph.

All possibilities on how to use the intention tree mentioned in the use cases are currently implemented in a working prototype. To allow widespread adoption plug-ins for existing technologies are to be implemented. A system to navigate through charts using an intention tree could be developed in JavaScript (Mozilla Developer Network n.d.) which could work for all kinds of browsers. ATs like screen readers could be enriched with plug-ins and scripting to recognise the current position and movement within the intention tree and react with the respective actions (like reading out the current value). In fact, first steps are already taken to explore the possibilities to implement a similar technology by callas software (pdfGoHTML by Callas Software, n.d.) for presenting mathematical formulae in the widespread Portable Document Format (PDF).

Another feature still to be implemented, for which alt text might still be preferable currently with regard to our prototype, is the textual description of graphs. By generally enhancing the data retrieval out of the intention tree we could go beyond the currently implemented aggregation modes so that the user can get, e.g. maxima, minima or median values. Such features could be added and enhanced as textual description in the chart development process. For example constant increase values, the development over time, local highs and lows could be calculated and presented to the user based on the information stored within the chart.

Another goal is to support other chart and graphic types. The intention tree is based on an abstract layer, so it can be adapted and used in different scenarios. The current prototype supports only SVG (W3C) and bar charts to prove the functionality provided by the intention tree. This could be extended to line charts, pie charts or even completely different types of graphics. One could think of a world map, which splits up into continents, regions, countries, states, provinces, cities and villages. This is another typical tree-structure, which could make use of the intention tree technology.

Finally let us take the approach one step further: charts or other graphics with an intention tree are not only useful for blind or vision-impaired people. They can improve usability for everybody. The new navigation possibilities through charts and images may even help those who do not recognise that they are using an accessibility feature. Maybe by selecting just some part of information out of a big chart which may be calculated based on the intention tree, maybe by teaching students different subjects like geography, coding, etc., by using interactive graphics which are based on an intention tree. As seen in schools, accessibility technology is often and successfully used in mainstream environments (Lindström, 2009). Thus, we do see a strong potential for and beyond accessibility.

References

Berners-Lee, T. (n.d.), "W3C Director and inventor of the World Wide Web", available at: www.w3.org/standards/webdesign/accessibility (accessed 23 September 2015).

Callas Software (n.d.), "pdfGoHTML", available at: www.callassoftware.com/en/products/pdfgohtml (accessed 7 October 2015).

Dardailler, H. (2002), "SVG linearization and accessibility", *Computer Graphics Forum*, Vol. 21 No. 4, pp. 777-86.

Duce, D., Herman, I. and Hopgood, B. (2002), "Web 2D graphics file formats", *Computer Graphics Forum*, Vol. 21 No. 1, pp. 43-64.

Lindström, H. (2009), "Use of assistive technology devices in mainstream schools: students' perspective", *The American Journal of Occupational Therapy*, Vol. 63 No. 4, pp. 463-72, available at: www.researchgate.net/profile/Helena_Hemmingsson/publication/26768514_Use_of_assistive_technology_devices_in_mainstream_schools_students'_perspective/links/0deec5270fee33c363000000.pdf (accessed 7 October 2015).

Mozilla Developer Network (n.d.), "JavaScript", available at: https://developer.mozilla.org/en-US/docs/Web/JavaScript (accessed 4 October 2015).

Playfair, W. (1786/1801), in Debrett, J. and Wallis, J. (Eds), *The Commercial and Political Atlas*, 1st and 3rd ed., London.

Savva, K., Chhajta, F.-F. and Agrawala, H. (2011), "ReVision: automated classification, analysis and redesign of chart images", *UIST '11 Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, pp. 393-402, available at: http://vis.stanford.edu/files/2011-ReVision-UIST.pdf (accessed 4 October 2015).

Statistics Austria (n.d.), "Geborene (Birth Rates in Austria)", available at: www.statistik.at/web_de/statistiken/menschen_und_gesellschaft/bevoelkerung/geborene/index.html (accessed 23 September 2015).

Vázquez S.R., Bouillon, P. and Bolfing, A. (2014), "Applying accessibility-oriented controlled language (CL) rules to improve appropriateness of text alternatives for images: an exploratory study", *LREC 2014 Proceedings*, pp. 4139-46, available at: www.lrec-conf.org/proceedings/lrec2014/pdf/640_Paper.pdf (accessed 4 October 2015).

WCAG 2.0. available at: www.w3.org/TR/WCAG20/ (accessed 4 October 2015).

XML. available at: www.w3.org/XML/ (accessed 23 September 2015).

Further reading

World Wide Web Consortium (W3C) (n.d.-a), "Cascading style sheets home page", available at: www.w3.org/Style/CSS/ (accessed 4 October 2015).

World Wide Web Consortium (W3C) (n.d.-b), "HTML5, a vocabulary and associated APIs for HTML and XHTML, W3C recommendation 28 October 2014", available at: www.w3.org/TR/html5/ (accessed 4 October 2015).

World Wide Web Consortium (W3C) (n.d.-c), "Web accessibility tutorials, guidance on how to create websites that meet WCAG, complex images", available at: www.w3.org/WAI/tutorials/images/complex (accessed 4 October 2015).

World Wide Web Consortium (W3C) (n.d.-d), "Scalable Vector Graphics (SVG) 1.1 (Second Edition)", W3C Recommendation 16 August 2011, available at: www.w3.org/TR/SVG11/ (accessed 4 October 2015).

World Wide Web Consortium (W3C) (n.d.-e), "Web content accessibility guidelines (WCAG) 2.0", W3C Recommendation 11 December 2008", available at: www.w3.org/TR/WCAG20/ (accessed 4 October 2015).

World Wide Web Consortium (W3C) (n.d.-f), "Extensible markup language (XML)", available at: www.w3.org/XML/ (accessed 4 October 2015).

Corresponding author

Professor Klaus Miesenberger can be contacted at: klaus.miesenberger@jku.at