Improving Braille accessibility and personnalization on Internet

Bruno Mascret
Université de Lyon, CNRS
Université Lyon 1, LIRIS, UMR5205
F-69622, France
bruno.mascret@liris.cnrs.fr

Vivien Guillet
Mission handicap, Université de Lyon
Université Lyon 1
F-69622, France
vivien.guillet@univ-lyon1.fr

Alain Mille
Université de Lyon, CNRS
Université Lyon 1, LIRIS, UMR5205
F-69622, France
alain.mille@liris.cnrs.fr

Aude Lancelle
NAT Braille support team
Chamelet, F-69620, France
audelancelle@gmail.com

ABSTRACT

This paper focuses on how to improve accessibility for Braille readers on Internet. We criticize actual technologies and show their limits in two areas : scientific Braille and Braille personnalization, especially in pedagogical situations. We present NAT Braille, a free software solution designed to respond to pedagogical specific needs. This transcriber has been made in a modular way with standard web technologies : the transcribing process uses a set of customizable XSLT transformations and several XML formats. We detail the design of NAT Braille and the technologies used for Braille transcriptions. Then we explain why NAT Braille improves personnalization in Braille rendering on Internet. We give the example of our Mozilla extension. This extension is able to transcribe web pages including MathML markup, and is set up with adapted transcription rules taking into account the user's preferences. We conclude by raising issues related to our work.

Keywords

Accessibility, Braille, Pedagogy, Web based education

1. INTRODUCTION

New technologies are increasingly making digital data access easier, especially considering the use of CMS[1], of educational platforms and many other communication tools. The amount of online information grows steadily in all fields - school, industry, business. Therefore, the CMS is a very precious publication tool for many especially for e-learning practices.

Inclusive education issues are still very active in a lot of scientific domains : assistive technologies, teaching methods, ergonomics... Concerning visually impaired students, D.Archambault [1] notices the numerous and various papers proposed to the International Council for Education of People with Visual Impairment under the topic "Inclusive Educational Practices". Moreover, he insists on the vocabulary used in English (inclusive education) and French (integration) to underline the difficulties encountered for a real *inclusion* of students into a class.

But visual impairement remains a real problem for both teacher and student, because most teachers are not skilled in Braille and only few solutions are designed to teach Braille inside a classroom. In this article, we focus on Braille and give an analysis of what a good automatic transcribing tool should propose to be suitable for inclusive education and for Internet contents.

Transcribing into Braille is not, as one should think, just a plain transcoding of characters. Braille users are mainly blind persons and transcribers. However, a transcriber is not necessarily professional - for example, a teacher with a blind student in his class needs to transcribe his documents into Braille. Conversely, a blind student writing in Braille with his computer is not able to make a transcription than his teacher can understand. Many countries, such as France, tend to integrate most blind students into general courses.

Different types of Braille codes are mentioned : the literary code (transcribing each character of the initial document), the abbreviated code (reducing the number of characters thanks to complex contraction rules), the mathematical code (transcribing all scientific notations)[4], the musical code. Each language has its own number of characters (no accents in English but many sorts of accents in other languages). Therefore, each language has its own Braille codes[10], if not each country (British and American codes

[1]Content Management System

W4A2012 - Submission Type, April 16-17, 2011, Lyon, France. Co-Located with the 21st International World Wide Web Conference.

are different). The musical Braille code is the only universal one. We sometimes use transcribing standards, describing page settings for documents, tables, notes, etc.

We propose to improve Braille support not only by using dedicated softwares in classroom but also by integrating Braille solutions in existing applications such as web browsers for example.

In the first section, we propose a critic of the existing solutions and technologies. We argue that pedagogical Braille could be used in a lot of applications, not only educational ones. We claim that using it on Internet could get students and users to improve their Braille reading skills. We show that users need personnalization for Braille according to their skills and needs.

Then we introduce NAT Braille[2][12][11], a free software solution designed to respond to pedagogical specific needs. We explain how it works and how it improves personnalization in Braille rendering on Internet. We give the example of our Mozilla extension. This extension is able to transcribe web pages including MathML markup, and is set up with adapted transcription rules taking into account the user's preferences. We conclude by raising issues related to our work.

2. CRITICS OF ACTUAL TECHNOLOGIES FOR BRAILLE ACCESSIBILITY

Digital communication tools offer many advantages to most users : time-saving on diffusion, organization, less paper use... But they create new handicap situations to visually impaired persons[8]. The accessibility of these communications media is increasingly better taken into account, thanks to specific software adaptations for visual impairment and to the development of "design for all".

However it is mainly based on restoring the structure, contents and conditioning of websites or applications. Reducing the time needed to find a specific piece of information inside a document, especially in a particular context, remains complicated and difficult [3].

Visually-impaired people only have sequential access to information (vocal reading, refreshable Braille display) and therefore cannot directly find the interesting elements, nor consider a document as a whole. The accessibility of digital contents by the document's producer is only rarely considered.

Indeed a Braille user needs an adaptation processing to read digital documents containing scientific notations. A vocal synthesizer or a screen reader can give a general idea of the document, but they do not offer an accurate understanding. Besides, they are still unable to transcribe mathematical, musical or graphical notations.

Archambault et al. [2] propose a good state of art of the main mathematical solutions. As for text transcribers, there are very few : DuxBurry's DBT[3] is the only good commercial solution for contracted Braille, Odt2Braille[1] uses the free LibLouisXML library[4] for text. BrlTTY[13] and BrlAPI[16] offer refreshable Braille features and a light support for contracted Braille and direct rendering. Even if the transcribing result is good, none of these solutions propose a pedagogical approach for contracted Braille and they offer few possibilities of customization. Moreover, only Odt2braille supports mathematics and text, and none implements chemical or musical notation.

Another criteria to keep in mind is the "time disability"[17][18]. Very often, visually impaired people waste a lot of time, even with accessible documents : they get lost throughout a document, try to find a precise paragraph, etc. Real time interactions with the document are in fact very limited, and it is up to the user to sort himself out. The communication between visually-impaired and sighted people remains uneasy since they do not share a common working space. This problem is of first importance especially for the inclusion of students into a class. Visually impaired must have the same interacting possibilities as other students during a lesson and their disabilities must be limited to allow them to follow a common pedagogical process[15]. The MaWEN project[2] proposes a tool for mathematics to limit this situation.

But for contracted Braille, there is no solution at this time to support the learning process. Moreover, most of the norms have not been revised and new terms (Internet, browser, keyboard...) have no special contractions. Users define their own contraction rules and it is difficult to set up assistive technologies with it.

The development of speech synthetizers and screen readers have improved the accessibility of text content. Simple scientific expressions may be accessed. As a result, many students do not wish to learn contracted or mathematical Braille anymore because their screen reader is skilled enough to give them access to content. This general trend becomes a real problem when students have to access to large documents or scientific ones : for complex scientific notation, Braille remains the only good solution. Moreover, at the university, the amount of document to read increases a lot. Contracted Braille may limit the amount of pages to read.

A last important point is the development of E-learning. It is a rising mode of learning, but it increases the time disabilities for visually-impaired students. Investigations on new interacting devices using 2D representations [14], haptic modalities[19] or Braille display have two main disadvantages : they are not able to manage pedagogical Braille and they don't transcribe directly scientific notations like MathMl on the web.

In the following section we introduce our contribution, NAT Braille[12][11], a free software solution designed to respond to pedagogical specific needs.

3. NAT BRAILLE, A TRANSCRIBER FOR PERSONNALIZATION

NAT Braille was created in 2005 during a university project. It was further developed for almost two years without financing. In July 2007, the software received a first financial support from the European Social Fund and was entirely supported by the French Minister of Education beteween July 2008 and April 2011. French laboratory LIRIS now supervises the project. An expert partnership with the INS HEA[5] has also been set to validate the quality of transcrip-

[2]downloadable at `http://liris.cnrs.fr/nat`; software under GLP licence.

[3]`http://www.duxburysystems.com/`

[4]`http://code.google.com/p/liblouisxml/`

[5]Institut National Supérieur de formation et de recherche pour l'éducation des jeunes Handicapés et les Enseignements Adaptés (National high institute of research and learning for impaired young and assistive education): this institute plays

tions. A fundation[6] has been created in order to follow up developments, maintain the software and set up partnerships.

This project mainly aims at solving the problems previously described, and wishes to produce a solution which could be at the same time accessible to every one, independent from special configurations, highly customizable, and potentially integrated to other systems. The motive is not to compete with transcribing centers -they are far better than any automatic software could ever be- but on the opposite to give them a tool allowing a bigger efficiency and productivity.

Furthermore, we have focused on proposing various integrations of NAT Braille in other softwares. We will essentially consider the use of NAT Braille in web browser in this article.

3.1 Working principles

Taking the different constraints into account has led us to a modular organization, based on adaptation to each type of document (format, mixed contents, encodings, etc.). The structure proposes three main modules : conversion, transcription and post-processing (see figure 1. Ideally the user gives the system a file in a given format : the conversion module conforms to the document type and produces an internal XML format file. Then the transcription module transcribes the internal file with chosen XSL filters. Finally the post-processing module manages the presentation, exportation or printing through other XSL filters. The specific role of each component allows the system to be independent during the development process. A new format would only need that a specific converter be associated to it. Transcribing filters are also independent from the initial format.

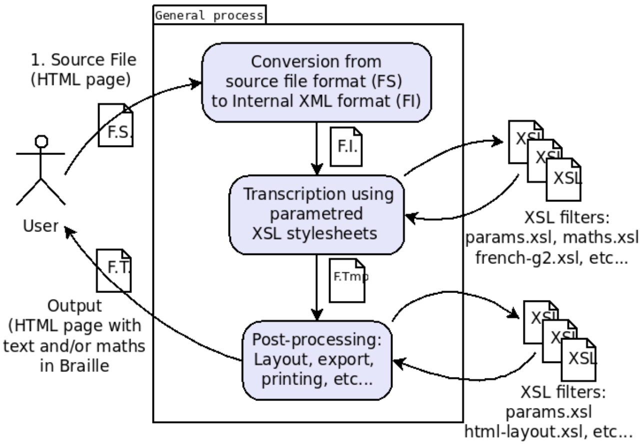

Figure 1: This figure describes the modular organization of NAT in three independant modules :conversion, transcribing and rendering. Note that users don't need to interact too much during the transcribing process.

The transcribing mechanism is original because the different filters and their specialization are interoperable. Their implementation is no longer based on dictionaries but on rules, and therefore gets as close as possible to a human reasoning when using different transcribing processes. Since these filters are interoperable, they allow each document to realize dynamically its own transcribing scenario : using abbreviated or literate Braille code, choosing encodings, choosing Braille code tables, whether transcribe mathematics or not, applying black to Braille or reverse transcription.

At the beginning of a transcription, the scenario is written according to many parameters. We will detail the most interesting features in the following section and the technical aspects in section 4. This organization allows us to propose a wide range of customizations.

One of our important principle is to fully comply with the norms.It seems to be the logical way to operate, but in fact many other softwares don't consider this essential, maybe for technical reasons or lack of tests.

3.2 Validation process and development

Since the efficient support of the French Minister of Education, we have split the original project team into two entities : research and testing. The first one, supervised by our laboratory is charged to develop the software whereas the second (INS HEA) organizes tests and provides advice. The test team is not only composed of Braille experts (most of them are members of the French Braille Norms Committee) : we have also included teachers and professors who have students in inclusive education (alone or UPI's ones). Some active users (school or university students) are invited to give their opinion as well.

This organization is really efficient because developers must consider several points of view depending on the way each tester uses Braille. It underlines the differences between a strict application of the norm and the real situation in a classroom. For example, one of the teachers who has quite good Braille skills has discovered that some notations she uses were not complying with the norm : the software she is using for mathematics has several Braille codes which do not correspond to the Braille norm.

Priorities are different for each member of our team : transcribing experts essentially focus on the quality of the transcription whereas teachers are more interested in an easy-to-deploy solution, even if the software makes a few mistakes. The research team has sometimes to deal with these contradictions.

Another important idea is that expert transcribers have acquired a lot of implicit knowledge on Braille transcription. Most of the time, they are not aware of that. This knowledge isn't written in norms and it is very hard to find which implicit rule is involved, especially for contracted Braille and mathematics. Thanks to the cooperation between the two teams, researchers have been able to propose usage rules and to include them in the transcribing process.

3.3 Adapting transcription to users

NAT is able to propose several kinds of French Braille transcription features : grade 1 or contracted (grade 2) Braille, mathematics and chemistry. Music is still under development. Mathematics and literal Braille codes (contracted or not) can be rendered into black.

But each notation contains several possibilities and parameters according to the user's skills. For example, beginners do not use the complementary rules in French grade one, nor specific trigonometric notation in mathematics. Later

a role of major importance in the French public education system

[6]this fundation is settled in the University Lyon 1 fundation

Possible user	Braille result	Rules activated
Primary school (beginner)	⠨⠇⠁⠨⠋⠕⠝⠉⠞⠊⠕⠝⠠⠄⠎⠊⠝⠦⠭⠴	basic g1
Primary school (advanced)	⠨⠇⠁⠨⠨⠋⠕⠝⠉⠞⠊⠕⠝⠠⠄⠎⠊⠝⠦⠭⠴	complementary g1
College student (beginner)	⠨⠄⠨⠨⠋⠕⠝⠉⠞⠊⠕⠝⠠⠄⠬⠦⠭⠴	trigo. & basic g2
College student (advanced)	⠨⠄⠨⠨⠋⠬⠠⠄⠬⠦⠭⠴	grade 2

Table 1: This table gives several transcriptions of the same expression "*La FONCTION* $sin(x)$" (The $sin(x)$ FUNCTION) according to the user's skill.

on they learn contracted Braille and apply these rules too.

Depending on the user's profile, we have to adapt the transcribing process to take into account the reader's skills. That is why NAT's core does not use any dictionary but is controlled with a large set of rules and parameters. Each of them can be set active or not. Table 1 shows different possible renderings and explains which rules have been involved to produce Braille : basic grade one rules don't make a difference between a full upercase word :*FONCTION* (⠨⠋⠕⠝⠉⠞⠊⠕⠝) and *La* (⠨⠇⠁) use the same simple capital prefix ⠨ ; then complementary grade 1 rules take this difference into account by using the double capital prefix ⠨⠨ for *FONCTION* (⠨⠨⠋⠕⠝⠉⠞⠊⠕⠝) ; "*sin*" (⠎⠊⠝) is represented by ⠬ in specific trigonomic notation ; "la" (the) is one of the first contracted words learnt in French (⠄) ; The contraction of "fonction" (⠋⠬) is learnt afterwards.

Most of the existing solutions propose a set of options to slightly adapt the transcription, but none is able to manage the contraction rules by activating only a subset of them. But most of the time, an unexperimented transcriber is not able to choose which rules have to be activated. Moreover, they are more than 100 signs and locutions, 800 symbols and 75 contraction rules depending on the context or the subset used (signs, symbol, other word...).

NAT is able to manage special configurations (pedagogical scenarios) containing steps which activate a set of rules. Scenarios have been made to follow the progression of Braille contraction learning methods like "Étudions l'abrégé"[5]. Each step has been checked by professional transcribers and Braille teachers to validate the quality of the produced Braille and the pedagogical consistence.

A user only has to know which step must be activated and may follow a pedagogical sequence given by someone more skilled in Braille. For example, at the beginning of November, students should begin lesson 4 : they go from step 3 to step 4 in the software.

We will now details in the next section the technical aspects of NAT Braille and more precisely the use of XML and XSL.

4. TECHNICAL ASPECTS

4.1 XML formats and rule definition

NAT uses several XML formats to represent the internal format of the document and the rule definitions.

The internal format has a fairly simple basis (see figure 2) and allows other notation standards (such as MathML) to be integrated. The smallest element is the word or the punctuation mark. All types of contents (mathematical or literary) are organized inside paragraphs which constitute the document. For the time being, the different elements' properties are represented through tag attributes, and not as being themselves tags (this is different from HTML). This way the document maintains a simple structure.

Including open and standard formats guarantees NAT to be compatible on upfront with all softwares respecting international standards and independent towards specific software distributions.

```
<!-- for math and chemical content -->
<!ENTITY % laDtdMath SYSTEM "mathml.dtd" >
%laDtdMath;
<!-- music -->
<!ENTITY % laDtdMusique SYSTEM "partwise.
  dtd" >
%laDtdMusique;

<!-- text content -->
<!ELEMENT ponctuation (#PCDATA)>
<!ELEMENT lit ((mot | ponctuation)*)>
<!ELEMENT page-break EMPTY>
<!ATTLIST mot
 hauteur  CDATA #IMPLIED
 mev   CDATA #IMPLIED
 integral CDATA #IMPLIED
 doSpace CDATA #IMPLIED
>

<!-- Element conteneurs de contenus -->
<!ELEMENT phrase ((math | lit | score-
  partwise)*)>
<!ELEMENT titre ((math | lit | score-
  partwise)*)>
<!ATTLIST titre
 niveauOrig   CDATA #IMPLIED
 niveauBraille CDATA #IMPLIED
 styleOrig    CDATA #IMPLIED
>
<!ELEMENT tableau ((col)+)>
<!ELEMENT col ((ligne)+)>
<!ELEMENT ligne ((math | lit | score-
  partwise)*)>
<!ELEMENT ol (li)+>
<!ELEMENT ul (li)+>
<!ELEMENT li (ol | ul | math | lit | score-
  partwise)*>

<!-- Element de base -->
<!ELEMENT doc ((phrase | titre | ol | ul |
  tableau | page-break)*)>
```

Figure 2: The main elements of the DTD of the internal format in NAT Braille

The XML format for transcription rules is slightly more complicated. It contains the definitions of locutions, signs, symbols and rules on these elements. Figure 3 gives examples of such definitions.

```
<!-- a locution -->
<locution>
  <noir>de suite</noir>
  <braille>&pt145;&pt456;&pt234;</braille>
</locution>
<!-- a sign -->
<signe pluriel="true">
  <noir>son</noir>
  <braille>&pt246;</braille>
</signe>
<!-- a symbol -->
<symbole>
  <noir>bonjour</noir>
  <braille>&pt12;&pt245;</braille>
</symbole>
<!-- a rule for symbol -->
<rule id="18">
  <desc>Les derives en 's' ne se terminant
     pas par "es" abregent &pt234;</desc>
  <ref>III.27, III,28</ref>
  <for>symbole</for>
  <regIn>^(.*[^e])s$</regIn>
  <regOut>($1)&pt234;</regOut>
</rule>
<!-- a general rule -->
<rule id="102">
  <desc>"logie" donne &pt123;&pt1245; en
     fin de mot</desc>
  <ref>I et L.R./P II-V</ref>
  <for>general</for>
  <regIn>(.*)(logie)(s?)$</regIn>
  <regOut>&pt123;&pt1245;</regOut>
  <regInBr>(.*)(&pt123;&pt1245;)(&pt234;?)
     $</regInBr>
  <regOutBr>$1logie$3</regOutBr>
</rule>
```

Figure 3: Examples of rule definitions for contracted Braille. The whole file for French contracted Braille contains about 4800 lines of rule definitions.

A rule may contain a simple definition (symbol, sign, locution) or complex rules using regular expressions. Complex rules have a description to allow users to understand the meaning of the rule, and to choose if they want to activate it. This rule file is then parsed by an XSL stylesheet to make sequences, lists and parameters used by the contraction algorithm. The resulting file is an XSL parameters file which will be included by the main xsl stylesheet.

We have used the same principle to define the patterns for hyphenation[7].

4.2 XSL management system

All Nat Braille's core modules consist of interoperable XSL Stylesheets. The choice of this technology is motivated by its expressiveness for XML transformations. It is also quite easy to set up XSL stylesheets with parameters and to generate custom stylesheets.

We have used two kind of XSL stylesheets :

- System styleshhets, which may receive a set of parameters and may include other system stylesheets;
- Generated stylesheets, which contain the constructed parameters and the main algorithm.

Thus the main algorithm is generated and includes the stylesheets it needs, according to the user's preferences. The table 2 gives a description of the main stylesheets of the software and the figure 4 shows the several inclusion possibilities for the transcription.

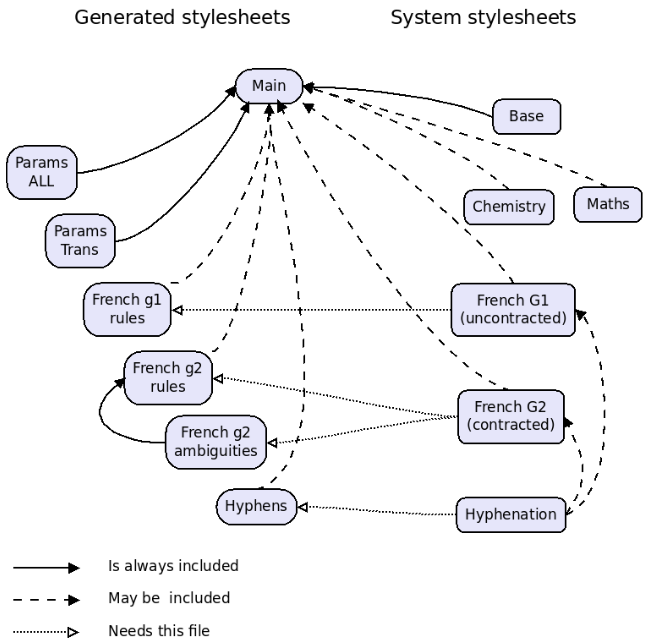

Figure 4: This figure shows the relationships between the xsl stylesheets involved in Braille transcription

5. A MOZILLA EXTENSION TO RENDER BRAILLE ON WEB PAGES

Classical assistive technologies are not able to manage pedagogical Braille on internet. Moreover, they can't transcribe MathMl content on web pages. The modular organization of NAT allowed us to propose conversion and presentation filters dedicated to web rendering. We have developed a Mozilla extension which takes text and MathMl contents on web pages, transcribes them into Braille according to the user's preferences, and then displays the result in the browser.

The main advantage of this feature is that a web page structure and layout is not changed : only the content is transcribed into Braille. It reduces the time needed to read the page, and provides efficient access to MathMl contents. Moreover, users need to set up only one software for Braille transcription : rules used to produce document or web page

[7] An explanation of the XSL implementation of Liang's algorithm in NAT can be found at `http://natbraille.free.fr/xsl/`

XSL name	kind of sheet	description
main.xsl	generated	main algorithm for transcription
paramsAll.xsl	generated	xsl params and variables, shared by both transcription and page layout
paramsTrans.xsl	generated	xsl params and variables used by transcription
paramsMep.xsl	generated	xsl params and variables used by page layout
base.xsl	system	common transcription rules for any content
fr-g1.xsl	system	French grade 1 algorithm
fr-g1-rules.xsl	system/generated	rules and set of constants for grade 1 transcription, will be generated in future
fr-g2.xsl	system	French grade 2 (contracted) algorithm
fr-g2-amb.xsl	system/generated	rules for ambiguities in grade 2 transcription, will be generated in future
fr-g2-rules.xsl	generated	rules for contractions (regular expression and definitions)
fr-maths.xsl	system	French mathematics algorithm
fr-chimie.xsl	system	French chemistry algorithm
hyphenation.xsl	system	Implementation of Liang's algorithm for hyphenation[9]
hyphens.xsl	generated	Hyphenation patterns for Liang's algorithm
...	...	...

Table 2: The main xsl stylesheets potentially involved in a Braille transcription. The main one include the system ones and the generated parameters needed to set up the transcription process. Some system files will be generated in future (rules for ambiguities and grade 1 variables) to take other languages into account.

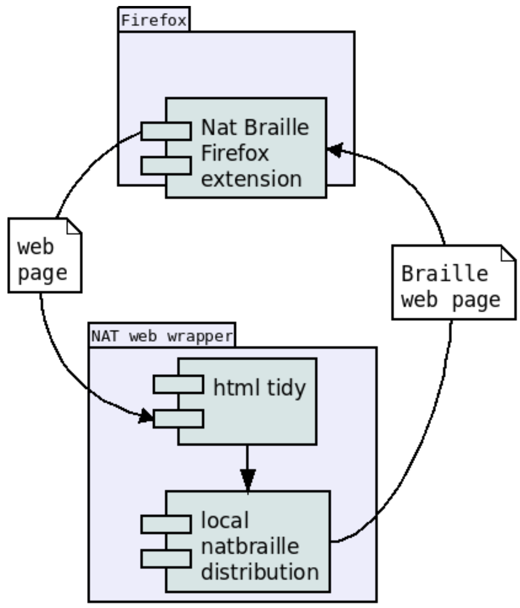

Figure 5: This figure shows the working principle of the Mozilla extension. The wrapper cleans up the body of the document and NAT Braille transcribes this HTML fragment. Then the resulting document replaces the original body of the web pages in the browser.

transcriptions are the same. This functionnality may help Braille learners access a wide range of documents, even on the Web.

5.1 Working principle

The NAT Braille firefox extension is written using the javascript Mozilla extension [8]. Figure 5 gives a representation of the working principle.

When activated, the extension is automatically triggered by mozilla each time a web page has finished downloading. A temporary local file containing the body of the downloaded page is created and its filename is passed as parameter to a NAT wrapper. This wrapper first uses HTML Tidy [9] to clean up the document. This allow to clean up the document for it to be parsable by NAT.

The wrapper then uses a local NAT installation for the transcription process, using a command line switch enabling the transcribing process to keep the original markup. The Braille translated document is put in a temporary file. When the wrapper's process ends, the extension replaces the body of the tab by fetching the content of the resulting Braille file. Mozilla refreshes automaticaly the page display which now shows the Braille translation of the document.

The writing and reading of temporary files and the execution of the NAT wrapper are done asynchronously by the extension so that the whole browser is not blocked by the translation process.

5.2 Results

Figure 6 shows an original web-page containing formulas. It has been extracted from an e-learning web-page. This web page contains several formulas which have been coded with MathMl. The page layout has been programmed in a CSS file.

The extension has immediately been called and after 3 seconds, the new page has been displayed in firefox (iceweasel in the screenshots).

Figure 7 shows the resulting page. The page structure is unchanged, nor the layout and the style, but contents have been transcribed into Braille. It is possible to transcribe only mathematics or text, depending on the default configuration choosen by the user

6. DISCUSSION

We show that NAT Braille is a good solution for inclusive education and web based education. Its modular organization helps to give a web extension able customize the Braille. However it remains a tool and would be advantaged if combined with other assistive solutions.

[8]https://developer.mozilla.org/fr/docs

[9]http://tidy.sourceforge.net/

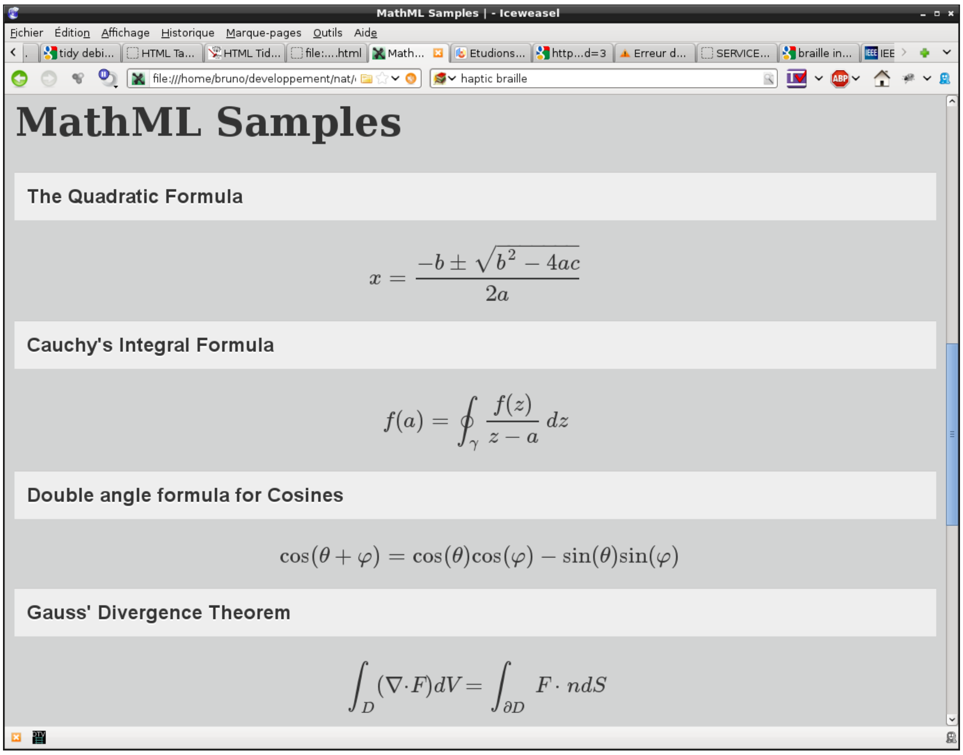

Figure 6: The original page

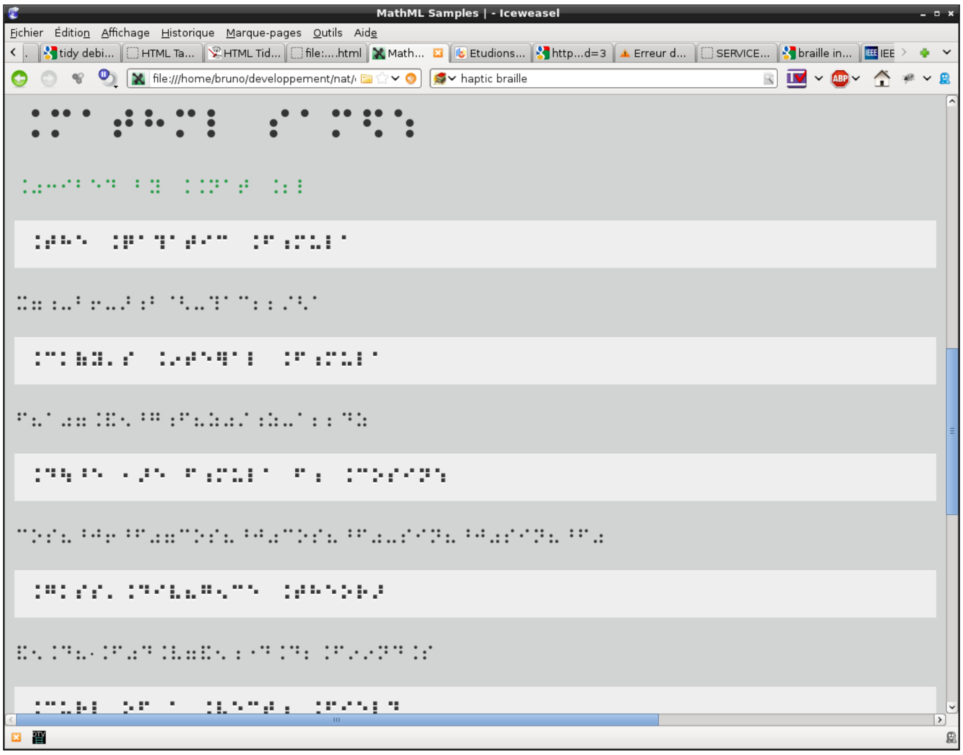

Figure 7: The resulting page

Our transcribing tool is a first step towards developing other assistive tools for Braille learning and transcribing. Some other assistive tools propose interesting features which could be combined or implemented into NAT to improve the understanding of formulas for example. A promising possibility would be to underline the current position in both Braille and black texts like in MaWEN[1].

Pedagogical scenarios could also be further developed thanks to dedicated software[7][6]. Interaction traces[3] could help their adaptation. Moreover, we are building a web service for NAT Braille to facilitate its integration in web based softwares. Finally we are developing foreign partnerships to propose NAT Braille into other languages and include NAT Braille in publishing chains for producing adaptable documents.

Acknowledgments

Authors wish to thank the project's test team for its precious help and advice, especially Marc Ollier (in charge of tests, INS HEA), Christiane Perdoux (former INS HEA teacher), Françoise Magna (in charge of the French Braille evolution council), Olga d'Amore (accessibility), Isabelle Spartialis and Alain Lacherez (inclusive education teachers in Rambouillet and Lille), and Jack Sagot (project manager at the INS HEA).

We wish to thank the other NAT's development team members : the permanent developer Frédéric "POF" Schwebel (LIRIS), and students who have increased the capabilities of the software during their training courses : Raphaël "BUPP" Mina (INSA of Lyon), Grégoire Lapicque (École Centrale de Lyon, Lucas Traoré (Institute of Technology of Villeurbanne) and Didier Erin (Lyon 1 University). We also thank all the too-numerous-to-be-mentioned free contributors and students who proposed improvements or who tested the software in a real situation.

7. REFERENCES

[1] D. Archambault. *Interaction et usages des modalités non visuelles, accessibilité des contenus complexes.* HDR, Faculté d'ingénierie, Université Pierre et Marie Curie-Paris, Paris, 2010.

[2] D. Archambault, B. Stöger, M. Batušič, C. Fahrengruber, and K. Miesenberger. Mathematical working environments for the blind : what is needed now ? In *Proceedings of ICALT Workshop on Advanced Learning Technologies for Disabled and Non-Disabled People*, Niigata, Japon, 2007.

[3] A. Cordier, B. Mascret, and A. Mille. Dynamic Case Based Reasoning for Contextual Reuse of Experience. In C. Marling, editor, *Provenance-Awareness in Case-Based Reasoning Workshop. ICCBR 2010.*, pages 69–78, July 2010.

[4] M. F. Le braille mathématique. *Revue bimestrielle du Groupement des professeurs et éducateurs d'aveugles et amblyopes*, 183, 2002.

[5] J. Le Reste and C. Perdoux. *Étudions l'abrégé.* Association Valentin Haüy, Paris, 2006.

[6] M. Lefèvre. Geppeto : une approche générique permettant d'adapter les activités des apprenants aux intentions pédagogiques de chaque enseignant. In *RJC EIAH 2010*, Lyon, France, 2010.

[7] M. Lefèvre, A. Mille, S. Jean-Daubias, and N. Guin. A meta-model to acquire relevant knowledge for interactive learning environments personalization. In *Adaptive 2009*, Athènes, Grèce, 2009.

[8] B. Leporini and M. Buzzi. Learning by e-learning: breaking down barriers and creating opportunities for the visually-impaired. In *Proceedings of the 4th international conference on Universal access in human-computer interaction: applications and services*, UAHCI'07, pages 687–696, Berlin, Heidelberg, 2007. Springer-Verlag.

[9] F. Liang. *Word Hy-phen-a-tion by Com-put-er.* PhD thesis, Stanford University, Computer Science Dept, 1983.

[10] s. C. Mackenzie. *World Braille usage : a survey of efforts towards uniformity of Braille notation.* International meeting on Braille uniformity, UNESCO, 1950, Paris, 1954.

[11] B. Mascret and A. Mille. Supporting the learning process – more than a braille transcription. In G. J. Gelderblom, M. Soede, L. Adriaens, and K. Miesenberger, editors, *Everyday Technology for Independence and Care - AAATE 2011*, volume 29, Maastricht, The Netherlands, Sept. 2011. AAATE, IOS Press.

[12] B. Mascret, A. Mille, and M. Ollier. An ideal braille transcriber? *AMSE journals, Advances in modelling*, dec 2008.

[13] N. Nair, N. Pitre, S. Doyon, and D. Mielke. Access to the console screen for blind persons using refreshable braille displays. *BRLTTY Reference Manual*, 1995-2011.

[14] M. Rotard, C. Taras, and T. Ertl. Tactile web browsing for blind people. *Multimedia Tools and Applications*, 37(1):53–69, 2008.

[15] C. Thibault and G. Leroux. To optimise the support to young visually impaired persons. In *Inclusive Educational Practices, 12th world conference*, Kuala Lumpur, Malaysia, 2006.

[16] S. Thibault and S. Hinderer. Brlapi: Simple, portable, concurrent, application-level control of braille terminals. *Arxiv preprint cs/0703044*, 2007.

[17] G. Uzan. Temps technologiques, temps individuels, temps sociaux : l'articulation des contraintes temporelles dans l'utilisation de l'informatique par des aveugles. In *Colloque Fisaf*, 2003.

[18] G. Uzan. *Ergonomie cognitive du handicap visuel : une contribution à la conception d'aides informatiques.* PhD thesis, Université René Descartes - Paris 5, 45 rue des Saints-Pères 75270 Paris Cedex 06 France, 12 2005.

[19] L. Zeng and G. Weber. Audio-haptic browser for a geographical information system. In *Computers Helping People with Special Needs*, pages 466–473. Springer, 2010.