

“Le Prononreur” by Marc Laurencelle

Computer science and mathematics
Design
Intermediate Age 16
Séminaire Saint-Joseph
Mauricie, Centre-du-Québec
Presented at the Science Fair in 2002

Project summary

“Le Prononreur” is a screen reader that transforms a typed French text or file into speech. The software analyzes each word and syllable, then emits the corresponding string of phonetic syllables. The system could read to the visually impaired or provide computer narration or a voice help function in a compact medium.

Project Report

WHAT IS IT?

It is a programmed system that, essentially, transforms a written text into speech. Given text typed on the keyboard or a text file, it splits the words into syllables, which it then processes, unlike most other utilities, which process whole words. In addition, the user can personalize the system by providing new sounds or modifying the way in which words are split.

To develop the software, I used Delphi, a language very similar to Pascal, which runs in a Windows environment. Despite its complexity, the system has a relatively short source code (about 950 lines). In the past year, I have worked dozens of hours on the project.

WHAT HARDWARE DO YOU NEED?

Basically, you need a computer with Windows and a monitor to see the text and the analysis. A relatively powerful processor (200 MHz or more) and a few megabytes of RAM are recommended for optimum performance.

If the text is to be typed in, you will also need a keyboard. A mouse will make using the software easier. And, of course, you will need speakers and a sound card to hear the speech.

WHAT SOFTWARE DO YOU NEED?

The executable file must be installed on the computer. A text file containing a list of words, syllables and word endings associated with a given sound must exist in order for the program to run: this is the main list. Optional text files can also be used: a list of word endings associated with the *liaison* effect (a condition under which silent consonants are pronounced), a list of word endings to be ignored in the detection of whole words.

Sound files corresponding to each sound defined in the main list (syllable/sound pairs) are indispensable for voice generation and are grouped together in a single directory. A subdirectory can be added for the sounds associated with each letter of the alphabet, without accents. The screen reader processes these files at the time of log-in.

LISTS

The main list is essential for processing whole words and splitting words into syllables.

Additional lists are optional. The lists are structured as follows.

➤ Whole words

Strings corresponding to a whole word may be associated with one or more consonant sounds at the end of the word that are transferred to the beginning of the next word (*enchaînement*). Some word endings, however, such as plural suffixes (which are normally not pronounced in French), may be ignored and defined in the appropriate list.

➤ Syllables and word endings

Each element (syllable or word ending) is associated with a sound. In order to be accepted, however, the elements must meet the conditions defined in the same line.

This will be explained in more detail later on. In addition, an element may appear several times in the list with varying sounds and conditions.

Throughout the analysis, the system seeks an element corresponding to the accumulated word segment. When it finds one, it verifies whether it meets the conditions and, if so, it memorizes it. If a later element also meets the conditions, it replaces the last result with its new values. Thus, the user defines similar elements in sequence, from the general to the specific, so that the general rule does not prevent the consideration of the vagaries of the language.

The analysis continues with a segment plus a new letter, up to N letters (default: N = 5). If no elements have been memorized once the value N is attained, the N letters are named one by one and the analysis begins again at the next letter.

➤ CONDITIONS

The conditions are variable, and can involve what precedes a segment, what follows it, or the entire word. The following are examples of conditions and their application:

<u>Condition (in code)</u>	<u>Meaning</u>
[+] SCC	Must be followed by two consonants
-SCV	Must not be followed by a consonant and a vowel
±SC; ±SV or SLCV	May be followed by a consonant or a vowel. The letter "L" means that the following letters or symbols form a list of possibilities to be checked one by one rather than a string.
-SR	Must not be the last character in the string

PLRC"a"	Must be the first character in the string or preceded by a consonant or the letter "a"
-M"salut"	The word must not be "salut"
-PR"ti"	Must not be preceded by "ti" at the beginning of the word

Definitions of the type "Syllable: Associated sound { Condition 1; ...; Condition *n*"

lu:lu	The syllable "lu" must produce the sound "lu"
ne:ne	The syllable "ne" must produce the sound "ne"
sa:sà	The syllable "sa" must produce the sound "sà"
sai:sè	The syllable "sai" must produce the sound "sè"
sain:sin{-SLV"n"	The syllable "sain" must produce the sound "sin" only if the segment is not followed by a vowel or the letter "n"
ÿsai:sé	The ending "sai" must produce the sound "sé". The symbol "ÿ" identifies the element as a word ending.
üc'est:sè	The word "c'est" must produce the sound "sè". The symbol "ü" introduces the definition of a whole word.
üwindows:wi/n'/dô/z'	The word "windows" must produce the following string of sounds: "wi", "n'", "dô" and "z"
ÿt:	The ending "t" must be silent. If no sound is defined, the element is silent.

Word splitting: The following table illustrates how certain words are split in accordance with the above definitions as follows: "word ? split":

salut ?	sà / lu	saine ?	sè / ne
sait ?	sè	saints ?	sin / t / s
saint ?	sin	sabre ?	sà / b / r / e

POTENTIAL USES

Some people could benefit from this system, for example, those with a partial or total vision loss. With a few adaptations, the program could read a Web site, e-mail, text files, or even an e-book.

Video game designers could save enormous amounts of disk space by having the program read aloud coded text in alphanumeric characters rather than using huge sound files.

PROBLEMS AND DIFFICULTIES

French is a wonderful language and its syllabic structure enables the program to process it effectively. I believe that a language like English would be more difficult to convert given its irregularities and lack of a simple structure, although some exceptions in French are also difficult to process. With an extended main list, however, the system could process whole words, thereby ensuring their proper pronunciation.

PLANNED IMPROVEMENTS AND CONCLUSION

Because of their unusual pronunciation, some words are very difficult to process.

Sometimes the words are not split in an optimal fashion, and updating lists becomes an arduous process. It might be possible to develop a list manager to facilitate the task.

Also, new types and conditions could be incorporated into the program to facilitate processing.

The sound files produce artificial series of syllables if they are strung together by concatenation; the sounds should coalesce. All the sounds could be loaded into memory when the program is launched then, after analysis of a sentence, the syllable sounds could be strung together. Other effects could be added later: emphasis in tone, reading speed, etc.